



INTERNATIONAL TELECOMMUNICATION UNION

**ITU-T**

TELECOMMUNICATION  
STANDARDIZATION SECTOR  
OF ITU

**X.224**

(11/93)

**DATA NETWORKS AND OPEN SYSTEM  
COMMUNICATIONS**

**OPEN SYSTEMS INTERCONNECTION –  
CONNECTION-MODE PROTOCOL SPECIFICATIONS**

---

**PROTOCOL FOR PROVIDING THE OSI  
CONNECTION-MODE TRANSPORT SERVICE**

**ITU-T Recommendation X.224**

(Previously "CCITT Recommendation")

---

## FOREWORD

The ITU-T (Telecommunication Standardization Sector) is a permanent organ of the International Telecommunication Union (ITU). The ITU-T is responsible for studying technical, operating and tariff questions and issuing Recommendations on them with a view to standardizing telecommunications on a worldwide basis.

The World Telecommunication Standardization Conference (WTSC), which meets every four years, establishes the topics for study by the ITU-T Study Groups which, in their turn, produce Recommendations on these topics.

The approval of Recommendations by the Members of the ITU-T is covered by the procedure laid down in WTSC Resolution No. 1 (Helsinki, March 1-12, 1993).

ITU-T Recommendation X.224 was revised by ITU-T Study Group 7 (1993-1996) and was approved under the WTSC Resolution No. 1 procedure on the 16th of November 1993.

---

### NOTE

In this Recommendation, the expression "Administration" is used for conciseness to indicate both a telecommunication administration and a recognized operating agency.

© ITU 1994

All rights reserved. No part of this publication may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm, without permission in writing from the ITU.

**ITU-T X-SERIES RECOMMENDATIONS**  
**DATA NETWORKS AND OPEN SYSTEM COMMUNICATIONS**  
(February 1994)

**ORGANIZATION OF X-SERIES RECOMMENDATIONS**

| Subject area                                | Recommendation series |
|---|-----------------------|
| <b>PUBLIC DATA NETWORKS</b>                 |                       |
| Services and facilities                     | X.1-X.19              |
| Interfaces                                  | X.20-X.49             |
| Transmission, signalling and switching      | X.50-X.89             |
| Network aspects                             | X.90-X.149            |
| Maintenance                                 | X.150-X.179           |
| Administrative arrangements                 | X.180-X.199           |
| <b>OPEN SYSTEMS INTERCONNECTION</b>         |                       |
| Model and notation                          | X.200-X.209           |
| Service definitions                         | X.210-X.219           |
| Connection-mode protocol specifications     | X.220-X.229           |
| Connectionless-mode protocol specifications | X.230-X.239           |
| PICS proformas                              | X.240-X.259           |
| Protocol identification                     | X.260-X.269           |
| Security protocols                          | X.270-X.279           |
| Layer managed objects                       | X.280-X.289           |
| Conformance testing                         | X.290-X.299           |
| <b>INTERWORKING BETWEEN NETWORKS</b>        |                       |
| General                                     | X.300-X.349           |
| Mobile data transmission systems            | X.350-X.369           |
| Management                                  | X.370-X.399           |
| <b>MESSAGE HANDLING SYSTEMS</b>             | X.400-X.499           |
| <b>DIRECTORY</b>                            | X.500-X.599           |
| <b>OSI NETWORKING AND SYSTEM ASPECTS</b>    |                       |
| Networking                                  | X.600-X.649           |
| Naming, addressing and registration         | X.650-X.679           |
| Abstract Syntax Notation One (ASN.1)        | X.680-X.699           |
| <b>OSI MANAGEMENT</b>                       | X.700-X.799           |
| <b>SECURITY</b>                             | X.800-X.849           |
| <b>OSI APPLICATIONS</b>                     |                       |
| Commitment, concurrency and recovery        | X.850-X.859           |
| Transaction processing                      | X.860-X.879           |
| Remote operations                           | X.880-X.899           |
| <b>OPEN DISTRIBUTED PROCESSING</b>          | X.900-X.999           |



## CONTENTS

|   | <i>Page</i> |
|---|-------------|
| 1 Scope.....  | 1           |
| 2 References.....   | 1           |
| 2.1 Identical Recommendations   International Standards.....                              | 2           |
| 2.2 Paired Recommendations   International Standards equivalent in technical content..... | 2           |
| 3 Definitions.....  | 2           |
| 4 Abbreviations.....  | 4           |
| 4.1 Data units.....   | 4           |
| 4.2 Types of Transport Protocol data units.....   | 4           |
| 4.3 TPDU Fields.....  | 4           |
| 4.4 Times and associated variables.....   | 5           |
| 4.5 Miscellaneous.....  | 5           |
| 5 Overview of the Transport Protocol.....   | 5           |
| 5.1 Service provided by the Transport Layer.....  | 6           |
| 5.2 Service assumed from the Network Layer.....   | 6           |
| 5.3 Functions of the Transport Layer.....   | 8           |
| 5.4 Classes and options when operating over CONS.....                                     | 9           |
| 5.5 Characteristics of class 4 transport protocol when operating over CLNS.....           | 11          |
| 5.6 Model of the Transport Layer.....   | 11          |
| 6 Elements of procedure.....  | 12          |
| 6.1 Use of the network service.....   | 12          |
| 6.2 Transport protocol data unit (TPDU) transfer.....                                     | 13          |
| 6.3 Segmenting and reassembling.....  | 14          |
| 6.4 Concatenation and separation.....   | 14          |
| 6.5 Connection establishment.....   | 15          |
| 6.6 Connection refusal.....   | 21          |
| 6.7 Normal release.....   | 21          |
| 6.8 Error release when operating over CONS.....   | 23          |
| 6.9 Association of TPDU's with transport connections.....                                 | 24          |
| 6.10 Data TPDU numbering.....   | 27          |
| 6.11 Expedited data transfer.....   | 28          |
| 6.12 Reassignment after failure when operating over CONS.....                             | 29          |
| 6.13 Retention and acknowledgement of TPDU's.....   | 30          |
| 6.14 Resynchronization.....   | 32          |
| 6.15 Multiplexing and demultiplexing when operating over CONS.....                        | 34          |
| 6.16 Explicit flow control.....   | 34          |
| 6.17 Checksum.....  | 35          |
| 6.18 Frozen references.....   | 36          |
| 6.19 Retransmission on time-out.....  | 37          |
| 6.20 Resequencing.....  | 37          |
| 6.21 Inactivity control.....  | 37          |
| 6.22 Treatment of protocol errors.....  | 38          |
| 6.23 Splitting and recombining when operating over CONS.....                              | 39          |
| 7 Protocol classes.....   | 39          |

|         |  |     |
|---------|--|-----|
| 8       | Specification for class 0: Simple class .....                          | 41  |
|         | 8.1 Functions of class 0 .....   | 41  |
|         | 8.2 Procedures for class 0 .....                                       | 41  |
| 9       | Specification for class 1: Basic error recovery class .....            | 41  |
|         | 9.1 Functions of class 1 .....   | 41  |
|         | 9.2 Procedures for class 1 .....                                       | 42  |
| 10      | Specification for class 2: Multiplexing class .....                    | 43  |
|         | 10.1 Functions of class 2 .....  | 43  |
|         | 10.2 Procedures for class 2 .....                                      | 43  |
| 11      | Specification for class 3: Error recovery and multiplexing class ..... | 45  |
|         | 11.1 Functions of class 3 .....  | 45  |
|         | 11.2 Procedures for class 3 .....                                      | 45  |
| 12      | Specification for class 4: Error detection and recovery class .....    | 47  |
|         | 12.1 Functions of class 4 .....  | 47  |
|         | 12.2 Procedures for class 4 .....                                      | 47  |
| 13      | Structure and encoding of TPDU's .....                                 | 59  |
|         | 13.1 Validity .....  | 59  |
|         | 13.2 Structure .....   | 60  |
|         | 13.3 Connection request (CR) TPDU .....                                | 62  |
|         | 13.4 Connection Confirm (CC) TPDU .....                                | 66  |
|         | 13.5 Disconnect Request (DR) TPDU .....                                | 67  |
|         | 13.6 Disconnect Confirm (DC) TPDU .....                                | 68  |
|         | 13.7 Data (DT) TPDU .....  | 69  |
|         | 13.8 Expedited data (ED) TPDU .....                                    | 70  |
|         | 13.9 Data acknowledgement (AK) TPDU .....                              | 71  |
|         | 13.10 Expedited data acknowledgement (EA) TPDU .....                   | 72  |
|         | 13.11 Reject (RJ) TPDU .....   | 73  |
|         | 13.12 TPDU error (ER) TPDU .....                                       | 74  |
| 14      | Conformance .....  | 75  |
| Annex A | – State tables .....   | 76  |
|         | A.1 General .....  | 76  |
|         | A.2 Conventions .....  | 76  |
|         | A.3 Tables .....   | 77  |
|         | A.4 State tables for classes 0 and 2 .....                             | 79  |
|         | A.5 State tables for classes 1 and 3 .....                             | 82  |
|         | A.6 State tables for class 4 over CONS .....                           | 88  |
|         | A.7 State tables for class 4 over CLNS .....                           | 94  |
| Annex B | – Network connection management subprotocol .....                      | 98  |
|         | B.1 Introduction .....   | 98  |
|         | B.2 Scope .....  | 98  |
|         | B.3 Definitions .....  | 98  |
|         | B.4 Abbreviations .....  | 99  |
|         | B.5 Overview of the protocol .....                                     | 99  |
|         | B.6 Elements of procedure .....  | 100 |
|         | B.7 Protocol operation .....   | 105 |
|         | B.8 Structure and encoding of TPDU's .....                             | 109 |
|         | B.9 Conformance .....  | 111 |
|         | B.10 State Table .....   | 112 |
|         | B.11 Diagram for NCMS protocol operation .....                         | 115 |

|  | <i>Page</i> |
|--|-------------|
| Annex C – PICS Proforma .....  | 118         |
| C.1    General.....  | 118         |
| C.2    Identification.....   | 118         |
| C.3    Indices used in this annex .....  | 119         |
| C.4    Based standard/recommendation conformance .....   | 121         |
| C.5    General statement of conformance .....  | 121         |
| C.6    Protocol implementation.....  | 121         |
| C.7    NCMS functions .....  | 121         |
| C.8    Initiator/responder capability for protocol classes 0 - 4.....  | 122         |
| C.9    Supported functions.....  | 122         |
| C.10   Supported TPDUs.....  | 128         |
| C.11   Supported parameters of issued TPDUs.....   | 129         |
| C.12   Supported parameters for received TPDUs .....   | 136         |
| C.13   User data in issued TPDUs.....  | 137         |
| C.14   User data in received TPDUs.....  | 137         |
| C.15   Negotiation .....   | 138         |
| C.16   Error handling.....   | 142         |
| C.17   Timers and protocol parameters.....   | 143         |
| Annex D – Checksum Algorithms.....   | 144         |
| D.1    Symbols .....   | 144         |
| D.2    Arithmetic conventions.....   | 144         |
| D.3    Algorithm for generating checksum parameters.....   | 144         |
| D.4    Algorithm for checking checksum parameters .....  | 144         |
| Annex E – State tables for operation of class 4 over connection-mode and connectionless-mode network services..... | 145         |
| E.1    General.....  | 145         |
| E.2    Conventions .....   | 145         |
| E.3    Tables.....   | 145         |
| E.4    State tables for Class 4.....   | 145         |
| Appendix I – Differences between Recommendation X.224 and ISO/IEC 8073:1992 .....                                  | 154         |
| I.1    Assignment to network connections .....   | 154         |
| I.2    Conformance.....  | 154         |
| I.3    Class negotiation.....  | 154         |
| I.4    Precedence .....  | 154         |
| I.5    Difference between Recommendation X.224 and ISO/IEC 8073 .....  | 154         |

## **SUMMARY**

This Recommendation specifies a common encoding and a five classes of transport protocol procedures which provide connection-mode transport service over either connection-mode or connectionless-mode network service and to be used with different network qualities of service.



## Introduction

This Recommendation is one of a set of Recommendations produced to facilitate the interconnection of information processing systems. This set of Recommendations covers the services and protocols required to achieve such interconnection.

The Transport Protocol is positioned with respect to other related Recommendations by the layers defined in the Reference Model for Open Systems Interconnection (CCITT Rec. X.200 | ISO 7498). It is most closely related to, and lies within the field of application of the Transport Service (ITU-T Rec. X.214 | ISO/IEC 8072). It also uses and makes reference to the Network Service Standard (CCITT Rec. X.213 | ISO/IEC 8348), whose provisions it assumes in order to accomplish the transport protocol's aims. The interrelationship of these Recommendations is illustrated in Figure 1.

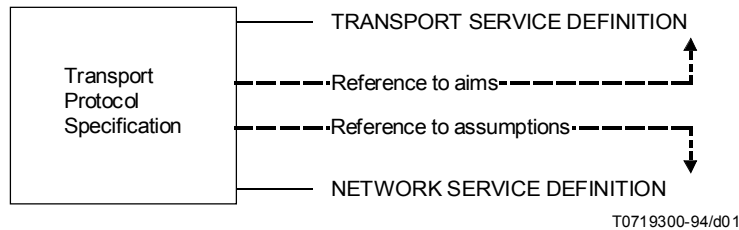


FIGURE 1/X.224

### Relationship between the Transport Protocol and adjacent services

This Recommendation 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 (each class provides a different set of functions for enhancement of service qualities).

This Recommendation defines mechanisms that can be used to optimize network tariffs and enhance the following qualities of service:

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

It does not require an implementation to use all of these mechanisms, nor does it define methods for measuring achieved quality of service or criteria for deciding when to release transport connections following quality of service degradation.

The primary aim of this Recommendation is to provide a set of rules for communication expressed in terms of the procedures to be carried out by peer entities at the time of communication. These rules for communication are intended to provide a sound basis for development in order to serve a variety of purposes i.e.:

- a) as a guide for implementors and designers;
- b) for use in the testing and procurement of equipment;
- c) as part of an agreement for the admittance of systems into the open systems environment;
- d) as a refinement of the understanding of OSI.

As it is expected that the initial users of this Recommendation will be designers and implementors of equipment this Recommendation contains, in notes or in annexes, guidance on the implementation of the procedures defined herein.

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 Recommendation 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 Recommendation 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.

This Recommendation contains a clause on conformance of equipment claiming to implement the procedures in this Recommendation. To evaluate conformance of a particular implementation, it is necessary to have a statement of which capabilities and options have been implemented for a given OSI protocol. Such a statement is called a Protocol Implementation Conformance Statement (PICS). A PICS proforma is provided in Annex C. Attention is drawn to the fact that this Recommendation does not contain any tests to demonstrate this conformance.

The variations and options available within this Recommendation are essential as they 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 Recommendation with statements of the options provided or required or with statements of the intended purpose of provision or use.

## **PROTOCOL FOR PROVIDING THE OSI CONNECTION-MODE TRANSPORT SERVICE<sup>1)</sup>**

*(Malaga-Torremolinos, 1984; amended at Melbourne, 1988, and at Geneva, 1993)*

### **1 Scope**

This Recommendation specifies

- a) five classes of procedures when operating over the connection-mode network service:
  - 1) class 0: simple class;
  - 2) class 1: basic error recovery class;
  - 3) class 2: multiplexing class;
  - 4) class 3: error recovery and multiplexing class;
  - 5) class 4: error detection and recovery class;for the connection-mode transfer of data and control information from one transport entity to a peer transport entity;
- b) one class (class 4) of procedure when operating over the connectionless-mode network service;
- c) the means of negotiating the class of procedures to be used by the transport entities;
- d) the structure and encoding of the transport protocol data units used for the transfer of data and control information.

The procedures are defined in terms of

- i) the interactions between peer transport entities through the exchange of transport protocol data units;
- ii) the interactions between a transport entity and the transport service user in the same system through the exchange of transport service primitives;
- iii) the interactions between a transport entity and the network service provider through the exchange of network service primitives.

These procedures are defined in the main text of this Recommendation supplemented by state tables in Annex A.

These procedures are applicable to instances of communication between systems which support the Transport Layer of the OSI Reference Model and which wish to interconnect in an open systems environment.

This Recommendation specifies, in clause 14, conformance requirements for systems implementing these procedures and provides the PICS proforma in compliance with the relevant requirements, and in accordance with the relevant guidance, given in CCITT Rec. X.291 | ISO/IEC 9646-2. It does not contain tests which can be used to demonstrate this conformance.

### **2 References**

The following Recommendations and International Standards contain provisions, which through references in the text, constitute provisions of this Recommendation. At the time of publication, the editions indicated were valid. All Recommendations and Standards are subject to revision, and parties to agreements based on this Recommendation are encouraged to investigate the possibility of applying the most recent edition of the Recommendations and Standards

---

<sup>1)</sup> Recommendation X.224 and ISO/IEC 8073 (Information technology – Telecommunications and information exchange between systems – Open Systems Interconnection – Protocol for providing the connection-mode transport service) are technically aligned.

listed below. Members of IEC and ISO maintain registers of currently valid International Standards. The Telecommunication Standardization Bureau of the ITU maintains a list of the currently valid ITU-T Recommendations.

## 2.1 Identical Recommendations | International Standards

- CCITT Recommendation X.213 (1992) | ISO/IEC 8348:1992, *Information technology – Network service definition for Open Systems Interconnection*.
- ITU-T Recommendation X.214 (1993) | ISO/IEC 8072:1993, *Information technology – Transport service definition for Open Systems Interconnection*.

## 2.2 Paired Recommendations | International Standards equivalent in technical content

- CCITT Recommendation X.290 (1992), *OSI Conformance testing methodology and framework for protocol Recommendations for CCITT applications – General concepts*.  
ISO/IEC 9646-1:1991, *Information technology – Open Systems Interconnection – Conformance testing methodology and framework – Part 1: General concepts*.
- CCITT Recommendation X.291 (1992), *OSI Conformance testing methodology and framework for protocol Recommendations for CCITT applications – Abstract test suite specification*.  
ISO/IEC 9646-2:1991, *Information technology – Open Systems Interconnection – Conformance testing methodology and framework – Part 2: Abstract test suite specification*.
- CCITT Recommendation X.650 (1992), *Open Systems Interconnections (OSI) – Reference model for naming and addressing*.  
ISO/IEC 7498-3:1991, *Information processing systems – Open Systems Interconnection – Basic Reference Model – Part 3: Naming and addressing*.
- ITU-T Recommendation X.264 (1993), *Transport protocol identification mechanism*.  
ISO/IEC 11570:1992, *Information technology – Telecommunications and information exchange between systems – Open Systems Interconnection – Transport protocol identification mechanism*.
- CCITT Recommendation X.200 (1988), *Reference model of Open Systems Interconnection for CCITT applications*.  
ISO 7498:1984, *Information processing systems – Open Systems Interconnection – Basic Reference Model*.

## 3 Definitions

NOTE – The definitions contained in this clause make use of abbreviations defined in clause 4.

**3.1** This Recommendation is based on the concepts developed in CCITT Rec. X.200 | ISO 7498 and CCITT Rec. X.650 | ISO/IEC 7498-3 and makes use of the following terms defined in them:

- concatenation and separation;
- segmenting and reassembling;
- multiplexing and demultiplexing;
- splitting and recombining;
- flow control;
- connectionless-mode transmission;
- nil selector value.

**3.2** For the purposes of this Recommendation, the following definitions apply:

**3.2.1 equipment:** Hardware or software or a combination of both; it need not be physically distinct within a computer system.

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

- 3.2.3 network service provider:** An abstract machine that models the totality of the entities providing the network service, as viewed by a transport entity.
- 3.2.4 local matter:** A decision made by a system concerning its behavior in the Transport Layer that is not subject to the requirements of this protocol.
- 3.2.5 initiator:** A transport entity that initiates a CR TPDU.
- 3.2.6 responder:** A transport entity with whom an initiator wishes to establish a transport connection.  
NOTE – Initiator and responder are defined with respect to a single transport connection. A transport entity can be both an initiator and responder simultaneously.
- 3.2.7 sending transport entity:** A transport entity that sends a given TPDU.
- 3.2.8 receiving transport entity:** A transport entity that receives a given TPDU.
- 3.2.9 preferred class:** The protocol class that the initiator indicates in a CR TPDU as its first choice for use over the transport connection.
- 3.2.10 alternative class:** A protocol class that the initiator indicates in a CR TPDU as an alternative choice for use over the transport connection.
- 3.2.11 proposed class:** A preferred class or an alternative class.
- 3.2.12 selected class:** The protocol class that the responder indicates in a CC TPDU that it has chosen for use over the transport connection.
- 3.2.13 proposed parameter:** The value for a parameter that the initiator indicates in a CR TPDU that it wishes to use over the transport connection.
- 3.2.14 selected parameter:** The value for a parameter that the responder indicates in a CC TPDU that it has chosen for use over the transport connection.
- 3.2.15 error indication:** An N-RESET indication, or an N-DISCONNECT indication with a reason code indicating an error, that a transport entity receives from the NS-provider.
- 3.2.16 invalid TPDU:** A TPDU that does not comply with the requirements of this Recommendation for structure and encoding.
- 3.2.17 protocol error:** A TPDU whose use does not comply with the procedures for the class.
- 3.2.18 sequence number:**
- a) the number in the TPDU-NR field of a DT TPDU that indicates the order in which the DT TPDU was transmitted by a transport entity;
  - b) the number in the YR-TU-NR field of an AK or RJ TPDU that indicates the sequence number of the next DT TPDU expected to be received by a transport entity.
- 3.2.19 transmit window:** The set of consecutive sequence numbers which a transport entity has been authorized by its peer entity to send at a given time on a given transport connection.
- 3.2.20 lower window edge:** The lowest sequence number in a transmit window.
- 3.2.21 upper window edge:** The sequence number which is one greater than the highest sequence number in the transmit window.
- 3.2.22 upper window edge allocated to the peer entity:** The value that a transport entity communicates to its peer entity to be interpreted as its new upper window edge.
- 3.2.23 closed window:** A transmit window that contains no sequence number.
- 3.2.24 window information:** Information contained in a TPDU relating to the upper and the lower window edges.
- 3.2.25 frozen reference:** A reference that is not available for assignment to a connection because of the requirements of 6.18.
- 3.2.26 unassigned reference:** A reference that is neither currently in use for identifying a transport connection nor which is in a frozen state.

**3.2.27 transparent (data):** TS-user data that is transferred intact between transport entities and which is unavailable for use by the transport entities.

**3.2.28 owner (of a network connection):** The transport entity that issued the N-CONNECT request leading to the creation of that network connection. Only applicable when operating over the connection-mode network service.

**3.2.29 retained TPDU:** A TPDU that is subject to the retransmission procedure or retention and acknowledgement procedure and is available for possible retransmission.

**3.3** This Recommendation uses the following terms defined in CCITT Rec. X.213 | ISO/IEC 8348:

- a) connection-mode network service;
- b) connectionless-mode network service.

**3.4** This Recommendation uses the following terms defined in CCITT Rec. X.290 | ISO/IEC 9646-1:

- a) PICS proforma;
- b) protocol implementation conformance statement (PICS).

## 4 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 Reject TPDU

ER 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)

SRC-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) |
| YR-EDTU-NR | ED TPDU number response (field)  |
| ROA        | Request of acknowledgement mark  |

#### 4.4 Times and associated variables

|                       |   |
|-----------------------|---|
| <i>TI</i>             | Local retransmission time                       |
| <i>N</i>              | The maximum number of transmissions             |
| <i>L</i>              | Time bound on reference and sequence number     |
| <i>I</i>              | Inactivity time                                 |
| <i>W</i>              | Window time                                     |
| <i>TTR</i>            | Time to try reassignment/resynchronization      |
| <i>TWR</i>            | Time to wait for reassignment/resynchronization |
| <i>TS1</i>            | Supervisory timer 1                             |
| <i>TS2</i>            | Supervisory timer 2                             |
| <i>M<sub>LR</sub></i> | NSDU lifetime local-to-remote                   |
| <i>M<sub>RL</sub></i> | NSDU lifetime remote-to-local                   |
| <i>E<sub>LR</sub></i> | Expected maximum transit delay local-to-remote  |
| <i>E<sub>RL</sub></i> | Expected maximum transit delay remote-to-local  |
| <i>R</i>              | Persistence time                                |
| <i>A<sub>L</sub></i>  | Local acknowledgement time                      |
| <i>A<sub>R</sub></i>  | Remote acknowledgement time                     |
| <i>I<sub>L</sub></i>  | Local inactivity time                           |
| <i>I<sub>R</sub></i>  | Remote inactivity time                          |

#### 4.5 Miscellaneous

|             |                                     |
|-------------|-------------------------------------|
| TS-user     | Transport-service user              |
| TSAP        | Transport-service-access-point      |
| NS-provider | Network service provider            |
| NSAP        | Network-service-access-point        |
| QOS         | Quality of service                  |
| CLNS        | Connectionless-mode network service |
| CONS        | Connection-mode network service     |

## 5 Overview of the Transport Protocol

NOTE – This overview is not exhaustive and has been provided for guidance.

## 5.1 Service provided by the Transport Layer

The protocol specified in this Recommendation supports the Transport Service defined in ITU-T Rec. X.214 | ISO/IEC 8072.

Information is transferred to and from the TS-user in the transport service primitives listed in Table 1.

TABLE 1/X.224

### Transport service primitives

| Primitives       |                    | Parameters   |
|------------------|--------------------|--|
| T-CONNECT        | request indication | Called address<br>Calling address<br>Expedited data option<br>Quality of service<br>TS-user data |
| T-CONNECT        | response confirm   | Responding address<br>Quality of service<br>Expedited data option<br>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<br>TS-user-data  |

## 5.2 Service assumed from the Network Layer

The protocol specified in this Recommendation assumes the use of the Network Service defined in CCITT Rec. X.213 | ISO/IEC 8348.

When operating over CONS, information is transferred to and from the NS-provider in the network service primitives listed in Table 2a). When operating over CLNS, information is transferred to and from the NS-provider in the network service primitives listed in Table 2b).

### NOTES

1 The parameters listed in Table 2a) are those in the current connection-mode network service (see CCITT Rec. X.213 | ISO/IEC 8348).

2 The parameters listed in Table 2b) are those in the current connectionless-mode network service (see CCITT Rec. X.213 | ISO/IEC 8348).

3 The way the parameters are exchanged between the transport entity and the NS-provider is a local matter.



TABLE 2a/X.224

## Connection-mode network service primitives

| Primitives         |  | X/Y    | Parameters  | W/X/Y/Z   |
|--------------------|--|--------|---|---|
| N-CONNECT          | request  | X      | Called address<br>Calling address   | X<br>X  |
|                    | indication   | X      | Receipt confirmation selection<br>Expedited data selection<br>QOS parameter set<br>NS-user-data | Y<br>Y<br>X<br>Z  |
| N-CONNECT          | response<br>confirm  | X      | Responding address<br>Receipt confirmation selection  | X   |
|                    |  | X      |   | Expedited data selection<br>QOS parameter set<br>NS-user-data |
| N-DATA             | request  | X      | N-user-data<br>Confirmation request   | X   |
|                    | indication   | X      |   | Y   |
| N-DATA ACKNOWLEDGE |  |        |   |   |
|                    | request<br>indication  | Y<br>Y |   |   |
| N-EXPEDITED DATA   | request<br>indication  | Y<br>Y | NS-user-data  | Y   |
| N-RESET            | request<br>indication  | X      | Reason<br>Originator<br>Reason  | W   |
|                    |  | X      |   | W<br>W  |
| N-RESET            | response<br>confirm  | X      | –<br>–  |   |
|                    |  | X      |   |   |
| N-DISCONNECT       | request  | X      | Reason<br>NS-user-data<br>Responding address  | W<br>Z<br>Z   |
|                    | indication   | X      | Originator<br>Reason<br>NS-user-data<br>Responding address                                      | W<br>W<br>Z<br>Z  |
| W                  | The usage of this parameter is a local matter, e.g. for diagnostic or to decide whether to attempt resynchronization.                      |        |   |   |
| 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.  |        |   |   |

TABLE 2b/X.224

**Connectionless-mode network service primitives**

| Primitives | X/Y  | Parameters  | W/X/Y/Z          |
|------------|--|---|------------------|
| N-UNITDATA | request  | Source address<br>Destination address<br>Quality of service<br>NS-user-data | X<br>X<br>X<br>X |
|            | indication   | Source address<br>Destination address<br>Quality of service<br>NS-user-data | X<br>X<br>X<br>X |
| W          | The usage of this parameter is a local matter, e.g. for diagnostic or to decide whether to attempt resynchronization.                      |   |                  |
| 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.  |   |                  |

**5.3 Functions of the Transport Layer****5.3.1 Overview of functions**

The functions in the Transport Layer are those necessary to bridge the gap between the services available from the Network Layer and those to be offered to the TS-users.

The functions in the Transport Layer are concerned with the enhancement of quality of service, including aspects of cost optimization.

These functions are grouped below into those used at all times during a transport connection and those concerned with connection establishment, data transfer and release.

NOTE – This Recommendation does not include the following functions which are under consideration for inclusion in future editions of this Recommendation:

- a) encryption;
- b) accounting mechanisms;
- c) status exchanges and monitoring of QOS;
- d) blocking;
- e) temporary release of network connections;
- f) alternative checksum algorithm.

**5.3.1.1 Functions used at all times**

The following functions, depending upon the selected class and options, are used at all times during a transport connection:

- a) *Transmission of TPDU's* (see 6.2 and 6.9).
- b) *Multiplexing and demultiplexing* (see 6.15) – A function used only when operating over CONS to share a single network connection between two or more transport connections.
- c) *Error detections* (see 6.10, 6.13 and 6.17) – A function used to detect the loss, corruption, duplication, misordering, or misdelivery of TPDU's.
- d) *Error recovery* (see 6.12, 6.14, 6.18, 6.19, 6.20, 6.21, and 6.22) – A function used to recover from detected and signalled errors.

### 5.3.1.2 Connection establishment

The purpose of connection establishment is to establish a transport connection between two TS-users. The following functions of the Transport Layer during this phase match the TS-users' requested quality of service with the services offered by the Network Layer:

- a) select the network service which best matches the requirement of the TS-user taking into account charges for various services (see 6.5);
- b) decide whether to multiplex multiple transport connections onto a single network connection only when operating over CONS (see 6.5);
- c) establish the optimum TPDU size (see 6.5);
- d) select the functions that will be operational upon entering the data transfer phase (see 6.5);
- e) map transport addresses onto network addresses;
- f) provide a means to distinguish between two different transport connections (see 6.5);
- g) transport of TS-user data (see 6.5);
- h) exchange values of Inactivity timers (see 6.5).

### 5.3.1.3 Data transfer

The purpose of data transfer is to permit duplex transmission of TSDUs between the two TS-users connected by the transport connection. This purpose is achieved by means of two-way simultaneous communication and by the following functions, some of which are used or not used in accordance with the result of the selection performed in connection establishment.

- a) *Concatenation and separation* (see 6.4) – A function used to collect several TPDU's into a single NSDU at the sending transport entity and to separate the TPDU's at the receiving transport entity.
- b) *Segmenting and reassembling* (see 6.3) – A function used to segment a single data TSDU into multiple TPDU's at the sending transport entity and to reassemble them into their original format at the receiving transport entity.
- c) *Splitting and recombining* (see 6.23) – A function allowing, only when operating over CONS, the simultaneous use of two or more network connections to support the same transport connection.
- d) *Flow control* (see 6.16) – A function used to regulate the flow of TPDU's between two transport entities on one transport connection.
- e) *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.
- f) *Expedited data* (see 6.11) – A function used to bypass the flow control of normal data TPDU. Expedited data TPDU flow is controlled by separate flow control.
- g) *TSDU delimiting* (see 6.3) – A function used to determine the beginning and ending of a TSDU.

### 5.3.1.4 Release

The purpose of release (see 6.7 and 6.8) is to provide disconnection of the transport connection, regardless of the current activity.

## 5.4 Classes and options when operating over CONS

### 5.4.1 General

The functions of the Transport Layer have been organized into classes and options.

A class defines a set of functions. Options define those functions within a class which may or may not be used.

This Recommendation defines five classes of protocol:

- a) class 0: simple class;
- b) class 1: basic error recovery class;
- c) class 2: multiplexing class;

- d) class 3: error recovery and multiplexing class;
- e) class 4: error detection and recovery class.

#### NOTES

- 1 Transport connections of classes 2, 3 and 4 may be multiplexed together onto the same network connection.
- 2 Classes 0 to 3 do not specify mechanisms to detect unsignalled network transmission failures.

### 5.4.2 Negotiation

The use of classes and options is negotiated during connection establishment. The choice made by the transport entities will depend upon

- a) the TS-users' requirements expressed via T-CONNECT service primitives;
- b) the quality of the available network services;
- c) the user required service versus cost ratio acceptable to the TS-user.

### 5.4.3 Choice of network connection

The following list classifies network services in terms of quality with respect to error behavior in relation to user requirements; its main purpose is to provide a basis for the decision regarding which class of transport protocol should be used in conjunction with given network connection:

- a) *Type A* – Network connection with acceptable residual error rate (for example, not signalled by disconnect or reset) and acceptable rate of signalled errors.
- b) *Type B* – Network connections with acceptable residual error rate (for example, not signalled by disconnect or reset) but unacceptable rate of signalled errors.
- c) *Type C* – Network connections with unacceptable residual error rate.

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

### 5.4.4 Characteristics of class 0

Class 0 provides the simplest type of transport connection and is fully compatible with the CCITT Recommendation T.70 for teletex terminals.

Class 0 has been designed to be used with type A network connections.

### 5.4.5 Characteristics of class 1

Class 1 provides a basic transport connection with minimal overheads.

The main purpose of the class is to recover from network disconnect or reset.

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

### 5.4.6 Characteristics of class 2

#### 5.4.6.1 General

Class 2 provides a way to multiplex several transport connections onto a single network connection. This class has been designed to be used with type A network connections.

#### 5.4.6.2 Use of explicit flow control

The objective is to provide flow control to help avoid congestion at transport-connection-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.

### **5.4.6.3 Non-use of explicit flow control**

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

### **5.4.7 Characteristics of class 3**

Class 3 provides the characteristics of class 2 plus the ability to recover from network disconnect or reset. Selection of this class is usually based upon reliability criteria. Class 3 has been designed to be used with type B network connections.

### **5.4.8 Characteristics of class 4**

Class 4 provides the characteristics of class 3, plus the capability to detect and recover from errors which occur as a result of the low grade of service available from the NS-provider. The kind of errors to be detected include: TPDU loss, TPDU delivery out of sequence, TPDU duplication and TPDU corruption. These errors may affect control TPDUs as well as data TPDUs.

This class also provides for increased throughput capability and additional resilience against network failure.

Class 4 has been designed to be used with type C network connections.

## **5.5 Characteristics of class 4 transport protocol when operating over CLNS**

In operation over a connectionless-mode network service the class 4 transport protocol provides flow control between communicating peer transport entities, the capability to detect and recover from errors which occur as a result of a low grade of service available from the NS-provider, and resilience from failure of the peer entity. The kinds of error to be detected include: TPDU loss, TPDU delivery out of sequence, TPDU duplication and TPDU corruption. These errors may affect control TPDUs as well as data TPDUs.

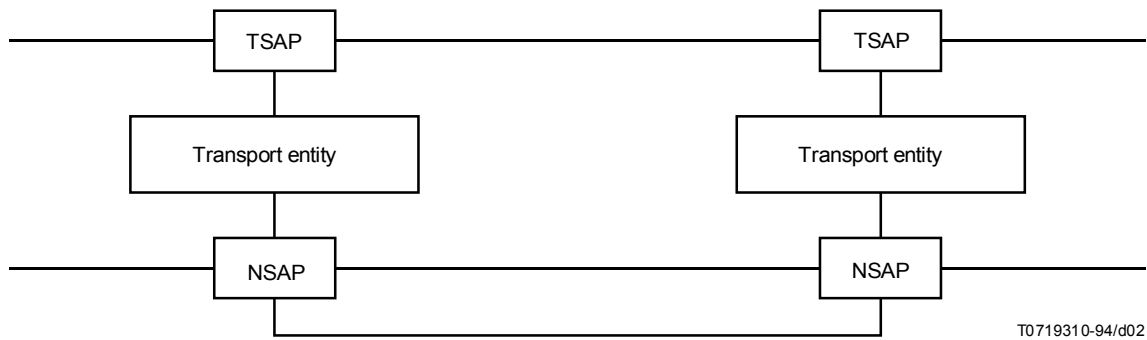
NOTE – The transport entity is incapable of distinguishing between failure of the network service and failure of the peer entity, except optionally, by some local means, in the case of the failure of the local interface to the network service (e.g. in the failure of the local transceiver on a local area network).

There is no indication given to the transport entity about the ability of the network entity to fulfill the service requirements given in the N-UNITDATA primitive. However, it can be a local matter to make transport entities aware of the availability and characteristics (QOS) of connectionless-mode network services, as the corresponding NSAP associations, exist logically by the nature of the connectionless-mode network service and may be recognized by network entities.

## **5.6 Model of the Transport Layer**

A transport entity communicates with its TS-users through one or more TSAPs by means of the service primitives as defined by the transport service definition (see ITU-T Rec. X.214 | ISO/IEC 8072). Service primitives will cause or be the result of transport protocol data unit exchanges between the peer transport entities supporting a transport connection. These protocol exchanges are effected using the services of the Network Layer as defined by the network service definition (see CCITT Rec. X.213 | ISO/IEC 8348) through one or more NSAPs. (See Figure 2.)

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



NOTE – For the purposes of illustration, Figure 2 shows only one TSAP and one NSAP for each transport entity. In certain instances, more than one TSAP and/or more than one NSAP may be associated with a particular transport entity.

FIGURE 2/X.224  
Model of the transport layer

## 6 Elements of procedure

This clause contains elements of procedure which are used in the specification of protocol classes in clauses 7 to 12. These elements are not meaningful on their own.

The procedures define the transfer of TPDU's whose structure and coding is specified in clause 13. Transport entities shall accept and respond to any TPDU received in a valid NSDU and may issue TPDU's initiating specific elements of procedure specified in this clause.

NOTE – Where network service primitives, TPDU's and parameters used are not significant for a particular element of procedure, they have not been included in the specification.

### 6.1 Use of the network service

#### 6.1.1 Assignment to network connection when operating over CONS

This procedure is used only when operating over the connection-mode network service.

##### 6.1.1.1 Purpose

The procedure is used in all classes to assign transport connections to network connections.

##### 6.1.1.2 Network service primitives

The procedure uses the following network service primitives:

- a) N-CONNECT;
- b) N-DISCONNECT.

##### 6.1.1.3 Procedure

Each transport connection shall be assigned to a network connection. The initiator may assign the transport connection to an existing network connection of which it is the owner or to a new network connection (see Note 1) which it creates for this purpose.

The initiator shall not assign or reassign the transport connection to an existing network connection if the protocol class(es) proposed for the class in use for the transport connection are incompatible with the current usage of the network connection with respect to multiplexing (see Note 2).

During the resynchronization (see 6.14) and reassignment after failure (see 6.12) procedures, the transport entity may reassign a transport connection to another network connection joining the same NSAPs, provided that it is the owner of the network connection and that the transport connection is assigned to only one network connection at any given time.

During the splitting procedure (see 6.23), a transport entity may assign a transport connection to any additional network connection joining the same NSAPs, provided that it is the owner of the network connection and that either the network connection does not have another transport connection assigned to it; or multiplexing is possible on the network connection.

The responder becomes aware of the assignment when it receives:

- a) a CR TPDU during the connection establishment procedure (see 6.5); or
- b) an RJ TPDU or a retransmitted CR or DR TPDU during the resynchronization (see 6.14) and reassignment after failure (see 6.12) procedures; or
- c) any TPDU when splitting (see 6.23) is used.

#### NOTES

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

2 An existing network connection may also not be suitable if, for example, the quality of service requested for the transport connection cannot be attained by using or enhancing the network connection.

3 A network connection with no transport connection(s) assigned to it, may be available after initial establishment, or because all of the transport connections previously assigned to it have been released. It is recommended that only the owner of such a network connection should release it. Furthermore, it is recommended that it not be released immediately after the transmission of the final TPDU of a transport connection; either a DR TPDU in response to CR TPDU or a DC TPDU in response to DR TPDU. An appropriate delay will allow the TPDU concerned to reach the other transport entity allowing the freeing of any resources associated with the transport connection concerned.

4 After the failure of a network connection, transport connections which were previously multiplexed together may be assigned to different network connections, and vice versa.

5 The transport protocol identification procedures specified in X.264 may need to be considered in conjunction with this procedure.

### **6.1.2 Transmission over CLNS**

This procedure is used only when operating over the connectionless-mode network service.

#### **6.1.2.1 Purpose**

The procedure is used to transmit TPDU's over the connectionless-mode network service.

#### **6.1.2.2 Network service primitives**

The procedure makes use of the following network service primitive:

N-UNITDATA.

#### **6.1.2.3 Procedure**

Each TPDU shall be transmitted in a single invocation of the connectionless-mode network service, over a pre-existing association between a pair of NSAPs. The association is considered by transport entities as permanently established and available.

### **6.2 Transport protocol data unit (TPDU) transfer**

#### **6.2.1 Purpose**

The TPDU transfer procedure is used in all classes to convey transport protocol data units in user data fields of network service primitives.

#### **6.2.2 Network service primitives**

The procedure uses the following network service primitives when operating over CONS:

- a) N-DATA;
- b) N-EXPEDITED DATA.

The procedure uses the following network service primitive when operating over CLNS:

N-UNITDATA.

### **6.2.3 Procedure**

The transport protocol data units (TPDUs) defined for the protocol are listed in 4.2.

When operating over CLNS, the transport entities shall transmit and receive all TPDUs as NS-user data parameters of N-UNITDATA primitives.

When operating over CONS and when the network expedited variant has been selected for class 1, the transport entities shall transmit and receive ED and EA TPDUs as NS-user data parameters of N-EXPEDITED DATA primitives.

In all other cases, transport entities shall transmit and receive TPDUs as NS-user data parameters of N-DATA primitives.

When a TPDU is put into an NS-user data parameter, the significance of the bits within an octet and the order of octets within a TPDU shall be defined in 13.2.

NOTE – TPDUs may be concatenated (see 6.4).

## **6.3 Segmenting and reassembling**

### **6.3.1 Purpose**

The segmenting and reassembling procedure is used in all classes to map TSDUs onto TPDUs.

### **6.3.2 TPDU and parameter used**

The procedure makes use of the following TPDU and parameter:

DT TPDU

– end of TSDU.

### **6.3.3 Procedure**

A transport entity shall map a TSDU onto an ordered sequence of one or more DT TPDUs. This sequence shall not be interrupted by other DT TPDUs on the same transport connection.

All DT TPDUs except the last DT TPDU in a sequence greater than one shall have a length of data greater than zero.

NOTES

- 1 The EOT parameter of a DT TPDU indicates whether or not there are subsequent DT TPDUs in the sequence.
- 2 There is no requirement that the DT TPDUs shall be of the maximum length selected during connection establishment.

## **6.4 Concatenation and separation**

### **6.4.1 Purpose**

The procedure for concatenation and separation is used in classes 1, 2, 3 and 4 to convey multiple TPDUs in one NSDU.

### **6.4.2 Procedure**

A transport entity may concatenate TPDUs from the same or different transport connections, while maintaining the order of TPDUs for a given transport connection compatible with the protocol operation.

A valid set of concatenated TPDUs may contain

- a) any number of TPDUs from the following list: AK, EA, RJ, ER, DC TPDUs, provided that these TPDUs come from different transport connections;
- b) no more than one TPDU from the following list: CR, DR, CC, DT, ED TPDUs; if this TPDU is present, it shall be placed last in the set of concatenated TPDUs.



A transport entity shall accept a valid set of concatenated TPDU's.

NOTES

- 1 The TPDU's within a concatenated set may be distinguished by means of the length indicator parameter.
- 2 The end of a TPDU containing data is indicated by the termination of the NSDU.
- 3 When operating over CONS, the number of concatenated TPDU's referred to in 6.4.2 a) is bounded by the maximum number of transport connections which are multiplexed together except during assignment or reassignment.

When operating over CLNS, the number of TPDU's that may be concatenated is bounded by the number of transport connections established between two NSAP's and/or the maximum available NSDU size.

## **6.5 Connection establishment**

### **6.5.1 Purpose**

The procedure for connection establishment is used in all classes to create a new transport connection.

### **6.5.2 Network service primitives**

When operating over CONS, the procedure uses the following network service primitive:

N-DATA.

When operating over CLNS, the procedure uses the following network service primitive:

N-UNITDATA.

### **6.5.3 TPDU's and parameters used**

The procedure uses the following TPDU's and parameters:

- a) CR TPDU
  - CDT;
  - DST-REF (set to zero);
  - SRC-REF;
  - CLASS and OPTIONS (i.e. preferred class, use of extended format, non-use of explicit flow control in class 2);
  - calling TSAP-ID;
  - called TSAP-ID;
  - TPDU size (proposed);
  - preferred maximum TPDU size (proposed);
  - version number;
  - protection parameter;
  - checksum;
  - additional option selection (i.e. use of network expedited in class 1, use of receipt confirmation in class 1, non-use of checksum in class 4, use of transport expedited data transfer service, use of selective acknowledgement, use of request acknowledgement);
  - alternative protocol class(es);
  - acknowledgement time;
  - inactivity time;
  - throughput (proposed);
  - residual error rate (proposed);
  - priority (proposed);
  - transit delay (proposed);
  - reassignment time;
  - user data;

- b) CC TPDU
  - CDT;
  - DST-REF;
  - SRC-REF;
  - CLASS and OPTIONS (selected);
  - calling TSAP-ID;
  - called TSAP-ID;
  - TPDU size (selected);
  - the preferred maximum TPDU size (selected);
  - protection parameter;
  - checksum;
  - additional option selection (selected);
  - acknowledgement time;
  - inactivity time;
  - throughput (selected);
  - residual error rate (selected);
  - priority (selected);
  - transit delay (selected);
  - user data.

#### 6.5.4 Procedure for operating over CONS

A transport connection is established by means of one transport entity (the initiator) transmitting a CR TPDU to the other transport entity (the responder), which replies with a CC TPDU.

Before sending the CR TPDU, the initiator assigns the transport connection being created to one (or more if the splitting procedure is being used) network connection(s). It is this set of network connections over which the TPDU's are sent.

NOTE 1 – Even if the initiator assigns the transport connection to more than one network connection, all the CR TPDU's (if repeated) or DR TPDU's with DST-REF set to zero which are sent prior to the receipt of the CC TPDU shall be sent on the same network connection, unless an N-DISCONNECT indication is received. (This is necessary because the remote entity may not support class 4 and therefore may not recognize splitting.) If the initiator has made other assignments, it will use them only after receipt of a class 4 CC TPDU (see also the splitting procedure 6.23).

During this exchange, all information and parameters needed for the transport entities to operate shall be exchanged or negotiated.

NOTE 2 – Except in class 4, it is recommended that the initiator starts an optional timer *TSI* at the time the CR TPDU is sent. This timer should be stopped when the connection is considered as accepted or refused or unsuccessful. If the timer expires, the initiator should reset or disconnect the network connection, and in classes 1 and 3, freeze the reference (see 6.18). For all other transport connection(s) multiplexed on the same network connection, the procedures for reset or disconnect as appropriate should be followed.

When an unexpected duplicated CR TPDU is received (with class 4 as preferred class) it shall be ignored in classes 0, 1, 2, and 3 and a CC TPDU shall be returned in class 4.

After receiving the CC TPDU for a class which includes the procedure for retention until acknowledgement of TPDU's the initiator shall acknowledge the CC TPDU as defined in Table 5 (see 6.13).

When the network expedited variant of the expedited data transfer (see 6.11) has been agreed (possible in class 1 only), the responder shall not send an ED TPDU before the CC TPDU is acknowledged.

The following information is exchanged:

- a) *References* – Each transport entity chooses a reference to be used by the peer entity which is 16 bits long and which is arbitrary under the following restrictions:
  - 1) it shall not already be in use nor frozen (see 6.18);
  - 2) it shall not be zero.

This mechanism is symmetrical and 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 matter.

- b) *Calling and called TSAP-IDs (optional)* – When either network address unambiguously defines the transport address, this information may be omitted.
- c) *Initial credit* – Only relevant for classes which include the explicit flow control function.
- d) *User data* – Not available if class 0 is the preferred class (see Note 3). Up to 32 octets in other classes.

NOTE 3 – If class 0 is a valid response according to Table 3, inclusion of user data in the CR TPDU may cause the responding entity to refuse the connection (for example, if it only supports class 0).

TABLE 3/X.224

**Valid responses corresponding to the preferred class and any alternative class proposed in the CR TPDU**

| Preferred class | Alternative class |                    |              |                 |              |              |
|-----------------|-------------------|--------------------|--------------|-----------------|--------------|--------------|
|                 | 0                 | 1                  | 2            | 3               | 4            | None         |
| 0               | Not valid         | Not valid          | Not valid    | Not valid       | Not valid    | Class 0      |
| 1               | Class 1 or 0      | Class 1 or 0       | Not valid    | Not valid       | Not valid    | Class 1 or 0 |
| 2               | Class 2 or 0      | Not valid          | Class 2      | Not valid       | Not valid    | Class 2      |
| 3               | Class 3, 2 or 0   | Class 3, 2, 1 or 0 | Class 3 or 2 | Class 3 or 2    | Not valid    | Class 3 or 2 |
| 4               | Class 4, 2 or 0   | Class 4, 2, 1 or 0 | Class 4 or 2 | Class 4, 3 or 2 | Class 4 or 2 | Class 4 or 2 |

- e) *Acknowledgement time* – Only in class 4.
- f) *Checksum parameter* – Only in class 4.
- g) *Protection parameter* – This parameter and its semantics are user defined.
- h) *Inactivity time* – Only in class 4. The inactivity time parameter shall not be included in a CC TPDU if it was not present in the corresponding CR TPDU.

The following negotiations take place:

- i) *Protocol class* – The initiator shall propose a preferred class and may propose any number of alternative classes which permit a valid response as defined in Table 3. The initiator should assume when it sends the CR TPDU that its preferred class will be agreed to, and commence the procedures associated with that class, except that if class 0 or class 1 is an alternative class, multiplexing shall not commence until a CC TPDU selecting the use of classes 2, 3 or 4 has been received.

NOTE 4 – This means, for example, that when the preferred class includes resynchronization (see 6.14) the resynchronization will occur if a reset is signalled during connection establishment.

The responder shall select one class defined in Table 3 as a valid response corresponding to the preferred class and to the class(es), if any, contained in the alternative class parameter of the CR TPDU. It shall indicate the selected class in the CC TPDU and shall follow the procedures for the selected class.

If the preferred class is not selected, then on receipt of the CC TPDU the initiator shall adjust its operation according to the procedures of the selected class.

NOTE 5 – The valid responses indicated in Table 3 result from both explicit negotiation, whereby each of the classes proposed is a valid response, and implicit negotiation whereby

- a) if class 3 or 4 is proposed, then class 2 is a valid response;
- b) if class 1 is proposed, then class 0 is a valid response.

NOTE 6 – Negotiation from class 2 to class 1 and from any class to an higher-numbered class is not valid.

NOTE 7 – Redundant combinations are not a protocol error.

- j) *TPDU size* – The initiator may propose a maximum size for TPDU's, and the responder may accept this value or respond with any value between 128 and the proposed value in the set of values available (see 13.3.4 b).

NOTE 8 – The length of the CR TPDU does not exceed 128 octets (see 13.3).

NOTE 9 – The transport entities may have knowledge, by some local means, of the maximum available NSDU size.

- k) *Preferred maximum TPDU size* – The value of this parameter, multiplied by 128, yields the proposed or accepted maximum TPDU size in octets. The initiator may proposed a preferred maximum size for TPDU's and the responder may accept this value or respond with a smaller value.

NOTE 10 – If this parameter is used in a CR TPDU without also including the TPDU size parameter, this will result in a maximum TPDU size of 128 octets being selected if the remote entity does not recognize the preferred TPDU size parameter. Therefore, it is recommended that both parameters be included in the CR TPDU.

If the preferred maximum TPDU size parameter is present in a CR TPDU the responder shall

- either: ignore the preferred maximum TPDU size parameter and follow TPDU size negotiation as defined in 6.5.4 j);
- or: use the preferred maximum TPDU size parameter to determine the maximum TPDU size requested by the initiator and ignore the TPDU size parameter. In this case the responder shall use the preferred maximum TPDU size parameter in the CC TPDU and shall not include the TPDU size parameter in the CC TPDU.

If the preferred maximum TPDU size parameter is not present in the CR TPDU it shall not be included in the corresponding CC TPDU. In this case TPDU size negotiation is as defined in 6.5.4 j).

- l) *Normal or extended format* – Either normal or extended is available. When extended is used, this applies to CDT, TPDU-NR, ED-TPDU-NR, YR-TU-NR and YR-EDTU-NR parameters.
- m) *Checksum selection* – This defines whether or not TPDU's of the connection are to include a checksum.
- n) *Quality of service parameters* – This defines the throughput, transit delay, priority and residual error rate.

NOTE 11 – The transport service defines transit delay as requiring a previously stated average TSDU size as a basis for any specification. This protocol as specified in 13.3.4 m), uses a value at 128 octets. Conversion to and from specifications based upon some other value is a local matter.

- o) The non-use of explicit flow control in class 2.
- p) The use of network receipt confirmation and network expedited when class 1 is to be used.
- q) *Use of expedited data transfer service* – This allows both TS-users to negotiate the use or non-use of the expedited data transport service as defined in the transport service (see ITU-T Rec. X.214 | ISO/IEC 8072).
- r) *The use of selective acknowledgement* – This allows the transport entities to decide whether to use procedures that allow acknowledgement of DT TPDU's that are received out-of-sequence (only in class 4).
- s) *The use of request acknowledgement* – This allows both transport entities to negotiate the use or non-use of the request acknowledgement facility specified in 6.13.4.2 (only in classes 1, 3, 4).

The following information is sent only in the CR TPDU:

- t) *Version number* – This defines the version of the transport protocol standard used for this connection.
- u) *Reassignment time parameter* – This indicates the time for which the initiator will persist in following the reassignment after failure procedure.

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 an alternative choice as defined in Table 4.

When a parameter [which is valid for the proposed class(es)] is absent and a default value is defined in this Recommendation, this is equivalent to the presence of the parameter with the default value.

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 shall also request or agree (respectively) to the use of explicit flow control.

TABLE 4/X.224

**Negotiation of options during connection establishment**

| Option   | Proposal made by the initiator | Valid selection by the responder |
|--|--------------------------------|----------------------------------|
| Transport expedited data transfer service (classes 1, 2, 3, 4 only)  | Yes<br>No                      | Yes or No<br>No                  |
| Use of receipt confirmation (class 1 only)   | Yes<br>No                      | Yes or No<br>No                  |
| Use of the network expedited variant (class 1 only)  | Yes<br>No                      | Yes or No<br>No                  |
| Non-use of checksum (class 4 only)   | Yes<br>No                      | Yes or No<br>No                  |
| Non-use of explicit flow control (class 2 only)  | Yes<br>No                      | Yes or No<br>No                  |
| Use of extended format (classes 2, 3, 4 only)  | Yes<br>No                      | Yes or No<br>No                  |
| Use of selective acknowledgement (class 4 only)  | Yes<br>No                      | Yes or No<br>No                  |
| Use of request acknowledgement (classes 1, 3, 4 only)  | Yes<br>No                      | Yes or No<br>No                  |
| NOTE – Table 4 defines the procedures for negotiation of options. This negotiation has been designed such that if the initiator proposes the mandatory implementation option specified in clause 14, the responder has to accept use of this option over the transport connection except for the use of the transport expedited data transfer service which may be rejected by the TS-user. If the initiator proposes a non-mandatory implementation option, the responder is entitled to select use of the mandatory implementation option for use over the transport connection. |                                |                                  |

**6.5.5 Procedure for operating over CLNS**

A transport connection is established by means of one transport entity (the initiator) transmitting a CR TPDU to the other transport entity (the responder), which replies with a CC TPDU. During this exchange, all information and parameters needed for the transport entities to operate shall be exchanged or negotiated. When an unexpected duplicated CR TPDU is received (with class 4 as preferred class) a CC TPDU shall be returned.

After receiving the CC TPDU, the initiator shall acknowledge the CC TPDU as defined in Table 5 (see 6.13).

The following information is exchanged:

- a) *References* – Each transport entity chooses a reference to be used by the peer entity which is 16 bits long and which is arbitrary under the following restrictions:
  - 1) it shall not already be in use nor frozen (see 6.18);
  - 2) it shall not be zero.

This mechanism is symmetrical and provides identification of the transport connection itself. The range of references used for transport connections, in a given transport entity, is a local matter.

- b) *Called and calling TSAP-IDs* (optional) – Indicates the calling and called transport service access points. When either network address unambiguously defines the transport address, this information may be omitted.
- c) *Initial credit*.
- d) *User data* – Up to 32 octets.
- e) *Acknowledgment time*.
- f) *Checksum parameter*.
- g) *Protection parameter* – This parameter and its semantics are user defined.
- h) *Inactivity time* – The inactivity time parameter shall not be included in a CC TPDU if it was not present in the corresponding CR TPDU.
- i) *Protocol class* – Class 4 is the only valid value for the preferred protocol class proposed by the initiator, and for the class selected by the responder. An alternative class is not permitted.

The following negotiations take place:

- j) *TPDU size* – The initiator may propose a maximum size for TPDU in the set of values available [see 13.3.4 b)]. This value may be limited by the maximum available NSDU size if known, and cannot exceed the maximum NSDU size for connectionless-mode network service as defined in CCITT Rec. X.213 | ISO/IEC 8348. The responder may accept this value or respond with any value between 128 and the proposed value in the set of values available [see 13.3.4 b)].

NOTE 1 – The length of the CR TPDU does not exceed 128 octets (see 13.3).

NOTE 2 – The transport entities may have knowledge, by some local means, of the maximum available NSDU size.

- k) *Preferred maximum TPDU size* – The value of this parameter, multiplied by 128, yields the proposed or accepted maximum TPDU size in octets. The initiator may propose a preferred maximum size for TPDU and the responder may accept this value or respond with a smaller value.

NOTE 3 – If this parameter is used in a CR TPDU without also including the TPDU size parameter, this will result in a maximum TPDU size of 128 octets being selected if the remote entity does not recognize the preferred TPDU size parameter. Therefore, it is recommended that both parameters be included in the CR TPDU.

If the preferred maximum TPDU size parameter is present in a CR TPDU the responder shall

- either: ignore the preferred maximum TPDU size parameter and follow TPDU size negotiation as defined in 6.5.5 j);
- or: use the preferred maximum TPDU size parameter to determine the maximum TPDU size requested by the initiator and ignore the TPDU size parameter. In this case the responder shall use the preferred maximum TPDU size parameter in the CC TPDU and shall not include the TPDU size parameter in the CC TPDU.

If the preferred maximum TPDU size parameter is not present in the CR TPDU it shall not be included in the corresponding CC TPDU. In this case TPDU size negotiation is as defined in 6.5.5 j).

- l) *Normal or extended format* – Either normal or extended is available. When extended is used, this applies to CDT, TPDU-NR, ED TPDU-NR, YR-TU-NR and YR-EDTU-NR parameters.
- m) *Checksum selection* – This defines whether or not TPDU of the connection are to include a checksum.
- n) *Quality of service parameters* – This defines the throughput, transit delay, priority and residual error rate.

NOTE 4 – The transport service defines transit delay as requiring a previously stated average TSDU size as a basis for any specification. This protocol as specified in 13.3.4 m), uses a value at 128 octets. Conversion to and from specifications based upon some other value is a local matter.

- o) *Use of expedited data transfer service* – This allows both TS-users to negotiate the use or non-use of the expedited data transport service as defined in the transport service (ITU-T Rec. X.214 | ISO/IEC 8072).

- p) *The use of selective acknowledgement* – This allows the transport entities to decide whether to use procedures that allow acknowledgement of DT TPDU's that are received out-of-sequence.
- q) *The use of request acknowledgement* – This allows both transport entities to negotiate the use or non-use of the request acknowledgement facility specified in 6.13.4.3.

The following information is sent only in the CR TPDU:

- r) *Version number* – This defines the version of the transport protocol standard used for this connection.

## 6.6 Connection refusal

### 6.6.1 Purpose

The connection refusal procedure is used in all classes when a transport entity refuses a transport connection in response to a CR TPDU.

### 6.6.2 TPDU's and parameters used

The procedure uses the follow TPDU's and parameters:

- a) DR TPDU
  - SRC-REF;
  - reason;
  - user data;
- b) ER TPDU
  - reject cause;
  - invalid TPDU.

### 6.6.3 Procedure

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

If a DR TPDU is received, the initiator shall regard the connection as released.

The responder shall respond to an invalid CR TPDU by sending an ER or DR TPDU. If an ER TPDU is received in response to a CR TPDU, the initiator shall regard the connection as released.

#### NOTES

- 1 When the invalid CR TPDU can be identified as having class 0 as the preferred class, it is recommended to respond with an ER TPDU. For all other invalid CR TPDU's either an ER TPDU or DR TPDU may be sent.
- 2 If the optional supervisory timer *TSI* has been set for this connection, then the initiator should stop the timer on receipt of the DR or ER TPDU.
- 3 It is a local matter whether the initiator releases the network connection if no transport connections are now assigned to it.

## 6.7 Normal release

### 6.7.1 Normal release when operating over CONS

#### 6.7.1.1 Purpose

The release procedure is used by a transport entity in order to terminate a transport connection. The implicit variant is used only in class 0. The explicit variant is used in classes 1, 2, 3 and 4.

#### NOTES

- 1 When the implicit variant is used (i.e. in class 0), the lifetime of the transport connection is directly correlated with the lifetime of the network connection.
- 2 The use of the explicit variant of the release procedure enables the transport connection to be released independently of the underlying network connection.

### 6.7.1.2 Network service primitives

The procedure uses the following network service primitives:

- a) N-DISCONNECT;
- b) N-DATA.

### 6.7.1.3 TPDU and parameters used

The procedure uses the following TPDU and parameters

- a) DR TPDU
  - reason;
  - user data;
  - SRC-REF;
  - DST-REF;
- b) DC TPDU.

### 6.7.1.4 Procedure for implicit variant

In the implicit variant either transport entity disconnects a transport connection by disconnecting the network connection to which it is assigned. When a transport entity receives an N-DISCONNECT this should be considered as the release of the transport connection.

### 6.7.1.5 Procedure for explicit variant

When the release of a transport connection is to be initiated, a transport entity

- a) If it has previously sent or received a CC TPDU (see Note 3) shall
  - 1) send a DR TPDU;
  - 2) discard all subsequently received TPDU's other than a DR or DC TPDU;
  - 3) consider the transport connection released on receipt of a DR or DC TPDU.
- b) If a) is not applicable and if there is an outstanding CR TPDU, it shall
  - 1) for classes other than class 4 wait for the acknowledgement of the outstanding CR TPDU; if it receives a CC TPDU, it shall follow the procedures in 6.7.1.5 a);
  - 2) for class 4 either send a DR TPDU with a zero value in the DST-REF field or follow the procedure in 6.7.1.5 b) 1). In the former case further receipt of a CC TPDU specifying class 4 will be ignored. Receipt of CC TPDU with another class will be processed as follows: If the class is 0 the network connection shall be disconnected, otherwise a DR TPDU with the DST-REF field set to the value of the SRC-REF field of the received CC TPDU shall be sent and the release procedure of the class is continued.

A transport entity that receives a DR TPDU shall

- c) If it has previously sent a DR TPDU for the same transport connection, consider the transport connection released.
  - d) If it has previously sent a CR TPDU that has not been acknowledged by a CC TPDU, consider the connection refused (see 6.6),  
if the SRC-REF is not zero a DC TPDU shall be sent using the SRC-REF as the DST-REF.
- NOTE 1 – In this case, the DR has been associated regardless of its SRC-REF field (see 6.9.1.4 and 6.9.2.4).
- e) If c) and d) are not applicable, send a DC TPDU and consider the transport connection released. If the received DR has the DST-REF field set to zero, then a DC with SRC-REF set to zero shall be sent, regardless of the local reference.

NOTE 2 – If the entity receiving such a DR TPDU has previously decided to negotiate down the class, this entity is always entitled to consider such a DR TPDU as spurious. Since no association has been made the transport connection is not released at the responder side but the CC TPDU, when sent, will be answered by a DR TPDU (spurious CC TPDU).

NOTE 3 – This requirement ensures that the transport entity is aware of the remote reference for the transport connection.

NOTE 4 – When the transport connection is considered as released, the local reference is either available for re-use or is frozen (see 6.18).



NOTE 5 – After the release of a transport connection, the network connection can be released or retained to enable its re-use for the assignment of other transport connections (see 6.1.1).

NOTE 6 – Except in class 4, it is recommended that, if a transport entity does not receive acknowledgement of a DR TPDU within time TS2, it should either reset or disconnect the network connection, and freeze the reference when appropriate (see 6.18). For all other transport connection(s) multiplexed on this network connection the procedures for reset or disconnect as appropriate should be followed.

NOTE 7 – When a transport entity is waiting for a CC TPDU before sending a DR TPDU and the network connection is reset or released, it should consider the transport connection released and, in classes other than classes 0 and 2, freeze the reference (see 6.18).

## **6.7.2 Normal release when operating over CLNS**

### **6.7.2.1 Purpose**

The release procedure is used by a transport entity in order to terminate a transport connection.

### **6.7.2.2 Network service primitives**

The procedure makes use of the following network service primitive:

N-UNITDATA.

### **6.7.2.3 TPDU and parameters used**

The procedure uses the following TPDU and parameters

- a) DR TPDU;
  - reason;
  - user data;
  - SRC-REF;
  - DST-REF;
- b) DC TPDU.

### **6.7.2.4 Procedure**

When the release of a transport connection is to be initiated, a transport entity shall send a DR TPDU and shall discard all subsequently received TPDU except for a DR or a DC TPDU.

On the receipt of a DR or a DC TPDU, it shall consider the transport connection to be released and the local reference shall be frozen (see 6.18). If a CC TPDU has been previously sent or received by the transport connection, then the remote reference is known and shall be used for the DST-REF in the DR TPDU to be sent. If the remote reference is not known, then the DST-REF in the DR TPDU may be set to zero, or the entity may wait until a CC TPDU is received before sending the DR TPDU.

NOTE – In case that the entity decides to wait for the arrival of the CC TPDU for the connection, deadlock could result from a CC TPDU that never arrives. Such a deadlock is prevented by the expiration of the CR TPDU retransmission counter, which forces the DR TPDU to be sent.

A transport entity which receives a DR TPDU shall

- a) consider the transport connection to be released if it has previously sent a DR TPDU for that connection;
- b) consider the transport connection to be refused (see 6.6) if it has previously sent a CR TPDU for that connection and no CC TPDU has been received in acknowledgment;
- c) consider the transport connection to be released and send a DC TPDU in all other cases. If the received DR TPDU has the DST-REF field set to zero, then a DC TPDU with SRC-REF set to zero shall be sent, regardless of the local reference.

## **6.8 Error release when operating over CONS**

### **6.8.1 Purpose**

This procedure is used only in classes 0 and 2 to release a transport connection on the receipt of an N-DISCONNECT or N-RESET indication.

## 6.8.2 Network service primitives

The procedure uses the following service primitives:

- a) N-DISCONNECT request;
- b) N-DISCONNECT indication;
- c) N-RESET indication;
- d) N-RESET response.

## 6.8.3 Procedure

When, on the network connection to which a transport connection is assigned, an N-DISCONNECT or N-RESET indication is received, both transport entities shall consider that the transport connection is released and so inform the TS-users.

On receipt of an N-RESET indication:

- in class 0, an N-DISCONNECT request shall be issued;
- in class 2, it is a local choice to issue an N-RESET response or an N-DISCONNECT request; one of these primitives shall be issued. However, if the Network Connection has other Transport Connections of a different class assigned to it, the error recovery procedure of that class shall be used to determine which primitive is issued.

## 6.9 Association of TPDU with transport connections

### 6.9.1 Association of TPDU with transport connections when operating over CONS

#### 6.9.1.1 Purpose

This procedure is used in all classes to interpret a received NSDU as TPDU(s) and, if possible, to associate each such TPDU with a transport connection.

#### 6.9.1.2 Network service primitives

The procedure uses the following network service primitives:

- a) N-DATA indication;
- b) N-EXPEDITED DATA indication;
- c) N-RESET request;
- d) N-DISCONNECT request.

#### 6.9.1.3 TPDU and parameters used

The procedure uses the following TPDU and parameters:

- a) any TPDU except CR TPDU, DT TPDU in classes 0 or 1 and AK TPDU in class 1
  - DST-REF;
- b) CR, CC, DR and DC TPDU
  - SRC-REF;
- c) DT TPDU in classes 0 or 1 and AK TPDU in class 1.

#### 6.9.1.4 Procedures

##### 6.9.1.4.1 Identification of TPDU

If the received NSDU or expedited NSDU cannot be decoded (i.e. does not contain one or more correct TPDU) or is corrupted (i.e. contains a TPDU with a wrong checksum), then the transport entity shall

- a) if the network connection on which the error is detected has a class 0 or 1 transport connection assigned to it, treat as a protocol error (see 6.22) for that transport connection;

- b) otherwise:
  - 1) if the NSDU can be decoded but contains corrupted TPDU, discard the TPDU (class 4 only) and optionally apply 6.9.1.4.1 b) 2);
  - 2) if the NSDU cannot be decoded issue an N-RESET or N-DISCONNECT request for the network connection and for all the transport connections assigned to this network connection (if any), apply the procedures defined for handling of network signalled reset or disconnect.

If the NSDU can be decoded and is not corrupted, the transport entity shall

- a) if the network connection on which the NSDU was received has a class 0 transport connection assigned to it, consider the NSDU as forming one TPDU and associate the TPDU with the transport connection (see 6.9.1.4.2);
- b) otherwise, invoke the separation procedures and for each of the individual TPDU in the order in which they appear in the NSDU apply the procedure defined in 6.9.1.4.2.

#### 6.9.1.4.2 Association of individual TPDU

If the received TPDU is a CR TPDU, and if the SRC-REF parameter and the remote NSAP indicate an existing transport connection at that receiving entity, then the CR TPDU is associated with that transport connection, otherwise it is processed as requesting the creation of a new transport connection.

If the received TPDU is a DT TPDU and the network connection has no TC assigned to it, and the DT TPDU is a class 0 or class 1 TPDU (as recognized by the absence of a DST-REF field), then the TPDU should be ignored.

Otherwise, the DST-REF parameter of the TPDU is used to identify the transport connection. The following cases are distinguished:

- a) If the DST-REF is not allocated to a transport connection, then no association with a transport connection is made and there are three cases:
  - 1) If the TPDU is a CC TPDU the transport entity shall respond on the same network connection with a DR TPDU. The SRC-REF of the DR TPDU may be either 0 or the DST-REF from the received CC TPDU.
  - 2) If the TPDU is a DR TPDU the transport entity shall respond on the same network connection with a DC TPDU; except in the case that the DR is carrying an SRC-REF set to zero, then no DC TPDU shall be sent, or in the case where the transport entity only supports class 0 then the network connection shall be disconnected.
  - 3) If the TPDU is neither a CC or DR it shall be discarded.
- b) If the DST-REF is allocated to a transport connection, but the TPDU is received on a network connection to which this connection has not been assigned, then there are four cases:
  - 1) If the transport connection is of class 4 and if the TPDU is received on a network connection with the same pair of NSAPs as that of the CR TPDU, then the TPDU is associated with this transport connection and considered as performing assignment.
  - 2) If the transport connection is not assigned to any network connection (waiting for reassignment after failure) and if the TPDU is received on a network connection with the same pair of NSAPs as that of the CR TPDU, then the association with that transport connection is made, except in the case of DC, DR and CC TPDU which are respectively described in 6.9.1.4.2 c), d), e).
  - 3) In classes 1 and 3, it is also possible to receive a TPDU performing reassignment prior to the notification of the disconnect of the current network connection (i.e. the transport connection is assigned to a network connection, but a TPDU containing the appropriate DST-REF is received on another network connection). In this case it is recommended that the transport entity:
    - issue an N-DISCONNECT request on the network connection to which the transport connection is currently assigned,
    - apply to all transport connections assigned to this network connection the procedure for processing a received N-DISCONNECT indication,
    - and then process the TPDU performing reassignment.

- 4) Otherwise, the TPDU is considered as having a DST-REF not allocated to a transport connection [case a)].
- c) If the TPDU is a DC TPDU then it is associated with the transport connection to which the DST-REF is allocated, unless the SRC-REF is not the expected one, in which case the DC TPDU is discarded.
- d) If the TPDU is a DR TPDU, then there are four cases:
  - 1) If the SRC-REF is not as expected then a DC TPDU with DST-REF equal to the SRC-REF of the received DR TPDU is sent back and no association is made, except that in the case where the transport entity only supports class 0 and cannot transmit a DC TPDU, it disconnects the network connection instead of transmitting a DC TPDU.
  - 2) If a CR TPDU is unacknowledged then the DR TPDU is associated with the transport connection, regardless of the value of its SRC-REF parameter.
  - 3) If the transport entity implements class 4 and if the DST-REF is zero and there is an unacknowledged CC TPDU or T-CONNECT RESPONSE is awaited, then the DR TPDU shall be associated with the transport connection holding the SRC-REF as the remote reference.
  - 4) Otherwise, the DR TPDU is associated with the transport connection identified by the DST-REF parameter.
- e) If the TPDU is a CC TPDU whose DST-REF parameter identifies an open connection (one for which a CC TPDU has been previously received), and the SRC-REF in the CC TPDU does not match the remote reference, then a DR TPDU is sent back with DST-REF equal to the SRC-REF of the received CC TPDU and no association is made.
- f) If none of the above cases apply then the TPDU is associated with the transport connection identified by the DST-REF parameter.

## **6.9.2 Association of TPDU with transport connections when operating over CLNS**

### **6.9.2.1 Purpose**

This procedure is used to interpret a received NSDU as TPDU(s) and, if possible, to associate each such TPDU with a transport connection.

### **6.9.2.2 Network service primitives**

This procedure makes use of the following network service primitive:

N-UNITDATA.

### **6.9.2.3 TPDU and parameters used**

This procedure makes use of the following TPDU and parameters:

- a) all TPDU except CR TPDU;
  - DST-REF;
- b) CR, CC, DR and DC TPDU;
  - SRC-REF.

### **6.9.2.4 Procedures**

#### **6.9.2.4.1 Identification of TPDU**

If the received NSDU cannot be decoded (i.e. does not contain one or more correct TPDU) or is corrupted (i.e. contains a TPDU with a wrong checksum), then the transport entity shall ignore (discard) the TPDU. If the NSDU can be decoded and is not corrupted, the transport entity shall invoke the separation procedures and for each of the individual TPDU in the order in which they appear in the NSDU apply the procedure in 6.9.2.4.2.

#### 6.9.2.4.2 Association of individual TPDU

Association of a received TPDU with a transport connection is generally performed by attempting to match the DST-REF in the received TPDU and the NSAP pair over which it was received with those of an existing transport connection. There are three exceptions to this general procedure: when the received TPDU is a CR TPDU, the SRC-REF is used instead of the DST-REF; when the received TPDU is either a DR or a DC TPDU, the SRC-REF is used in addition to the DST-REF; and when the received TPDU is a CC TPDU, whose DST-REF parameter identifies an open connection (one for which a CC TPDU has been previously received), then the SRC-REF is used in addition to the DST-REF.

The following actions shall be taken in consequence to the inability to match the TPDU to an existing transport connection:

- a) for a CR TPDU, a new transport connection shall be created;
- b) for a CC TPDU, a DR TPDU shall be sent using the SRC-REF and DST-REF from the received CC TPDU as the DST-REF and SRC-REF, respectively, of the DR TPDU;
- c) for a DR TPDU, there are four cases:
  - 1) if a CR TPDU is unacknowledged for the connection identified by the DST-REF in the DR TPDU, then the DR TPDU is associated with that connection regardless of the SRC-REF in the DR TPDU;
  - 2) if the CR TPDU for the connection identified by the DST-REF of the DR TPDU has been acknowledged and the SRC-REF is not as expected, then a DC TPDU using the SRC-REF of the DR TPDU as DST-REF is sent and no association is made;
  - 3) if the DST-REF in the DR TPDU is zero and there is an unacknowledged CC TPDU or a T-CONNECT response is awaited for a transport connection holding remote reference equal to the SRC-REF of the DR TPDU, then the DR TPDU is associated with that transport connection;
  - 4) in all other situations, the DR TPDU is associated with the transport connection identified by the DST-REF of the DR TPDU;
- d) for all other TPDU types, the TPDU is discarded.

### 6.10 Data TPDU numbering

#### 6.10.1 Purpose

Data TPDU numbering is used in classes 1, 2 (except when the non-use of explicit flow control option is selected), 3 and 4. Its purpose is to enable the use of recovery, flow control and resequencing functions.

#### 6.10.2 TPDU and parameters used

The procedure uses the following TPDU and parameter:

DT TPDU

- TPDU-NR.

#### 6.10.3 Procedure

A transport entity shall allocate the sequence number zero to the TPDU-NR of the first DT TPDU which it transmits for a transport connection. For subsequent DT TPDU's sent on the same transport connection, the transport entity shall allocate a sequence number one greater than the previous one.

When a DT TPDU is retransmitted, the TPDU-NR parameter shall have the same value as in the first transmission of that DT TPDU.

Modulo  $2^7$  arithmetic shall be used when normal formats have been selected and modulo  $2^{31}$  arithmetic shall be used when extended formats have been selected. In this Recommendation the relationships "greater than" and "less than" apply to a set of contiguous TPDU numbers whose range is less than the modulus and whose starting and finishing numbers are known. 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**

### **6.11.1 Expedited data transfer when operating over CONS**

#### **6.11.1.1 Purpose**

Expedited data transfer procedures are selected during connection establishment. The network normal data variant may be used in classes 1, 2, 3 and 4. The network expedited variant is only used in class 1.

#### **6.11.1.2 Network service primitives**

The procedure uses the following network service primitives:

- a) N-DATA;
- b) N-EXPEDITED DATA.

#### **6.11.1.3 TPDUs and parameters used**

The procedure uses the following TPDUs and parameters:

- a) ED TPDU
  - ED TPDU-NR;
- b) EA TPDU
  - YR-EDTU-NR.

#### **6.11.1.4 Procedures**

The TS-user data parameter of each T-EXPEDITED DATA request shall be conveyed as the data field of an Expedited Data (ED) TPDU.

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

No more than one ED TPDU shall remain unacknowledged at any time for each direction of a transport connection.

An ED TPDU with a zero length data field shall be treated as a protocol error.

#### NOTES

1 The network normal data variant is used, except when the network expedited variant (available in class 1 only), has been agreed, in which case ED and EA TPDUs are conveyed in the data fields of N-EXPEDITED DATA primitives (see 6.2.3).

2 No TPDU can be transmitted using the network expedited variant until the CC TPDU becomes acknowledged, to prevent the network expedited variant from overtaking the CC TPDU.

### **6.11.2 Expedited data transfer when operating over CLNS**

#### **6.11.2.1 Purpose**

Expedited data transfer procedures are selected during connection establishment.

#### **6.11.2.2 Network service primitives**

The procedure makes use of the following network service primitive:

N-UNITDATA.

#### **6.11.2.3 TPDUs and parameters used**

The procedure makes use of the following TPDUs and parameters:

- a) ED TPDU
  - ED TPDU-NR;
- b) EA TPDU
  - YR-EDTU-NR.

#### 6.11.2.4 Procedures

The TS-user data parameter of each T-EXPEDITED DATA request shall be conveyed as the data field of an Expedited Data (ED) TPDU.

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

No more than one ED TPDU shall remain unacknowledged at any time for each direction of a transport connection.

An ED TPDU with a zero length data field shall be treated as a protocol error (see 6.22).

### 6.12 Reassignment after failure when operating over CONS

#### 6.12.1 Purpose

The reassignment after failure procedure is used in classes 1 and 3 to commence recovery from an NS-provider signalled disconnect.

#### 6.12.2 Network service primitives

The procedure uses the following network service primitive:

N-DISCONNECT indication.

#### 6.12.3 Procedure

When an N-DISCONNECT indication is received for the network connection to which a transport connection is assigned, the initiator shall apply one of the following alternatives:

- a) If the TTR timer has not already run out and no DR TPDU is retained
  - 1) assign the transport connection to a different network connection (see 6.1.1) and start its TTR timer if not already started;
  - 2) while waiting for the completion of assignment if
    - an N-DISCONNECT indication is received, repeat the procedure from 6.12.3 a);
    - the TTR timer expires, begin procedure 6.12.3 b);
  - 3) when reassignment is complete perform active resynchronization by executing the procedure described in 6.14.4.1, and, if 6.14.4.1 b) has been performed, wait for the next event as follows:
    - if a valid TPDU is received as the result of the resynchronization, stop the TTR timer, or
    - if TTR runs out, wait for the next event, or
    - if an N-DISCONNECT indication is received, begin either procedure 6.12.3 a) or 6.12.3 b) depending on the TTR timer.
- b) If the TTR timer has run out, consider the transport connection as released and freeze the reference (see 6.18).
- c) If a DR TPDU is retained and the TTR timer has not run out, then follow the actions in either 6.12.3 a) or 6.12.3 b).

NOTE – After TTR expires and while waiting for the next event, it is recommended that the initiator set a timer with a value equal to TWR. If this timer expires before the next event, the initiator should begin the procedure in 6.12.3 b).

The responder shall start its TWR timer if not already started. The arrival of the first TPDU related to the transport connection (because of resynchronization by the initiator) completes the reassignment after failure procedure. The TWR timer is stopped and the responder shall continue with resynchronization (see 6.14). If reassignment does not take place within this time, the transport connection is considered released and the reference is frozen (see 6.18).

#### 6.12.4 Timers

The reassignment after failure procedure uses two timers:

- a) TTR, the time to try reassignment/resynchronization timer;
- b) TWR, the time to wait for reassignment/ resynchronization timer.

The TTR timer is used by the initiator. Its value shall not exceed *2 min* minus the sum of the maximum disconnect propagation delay and the maximum transit delay of the network connections (see Note 1). The value for the TTR timer may be indicated in the CR TPDU.

The TWR timer is used by the responder. If the reassignment time parameter is present in the CR TPDU, the TWR timer value shall be greater than the sum of the TTR timer plus the maximum disconnect propagation delay plus the maximum transit delay of the network connections.

If the reassignment time parameter is not present in the CR TPDU, a default value of *2 min* shall be used for the TWR timer.

##### NOTES

- 1 Provided that the required quality of service is met, TTR may be set to zero (i.e. no reassignment). This may be done, for example, if the rate of NS-provider generated disconnects is very low.
- 2 Inclusion of the reassignment time parameter in the CR TPDU allows the responder to use a TWR value of less than *2 min*.
- 3 If the optional TS1 and TS2 timers are used, it is recommended
  - a) to stop TS1 or TS2 if running when TTR or TWR is started;
  - b) to restart TS1 or TS2 if necessary when the corresponding TPDU (CR TPDU or DR TPDU respectively) is repeated;
  - c) to select for TS1 and TS2 values greater than TTR.

### 6.13 Retention and acknowledgement of TPDUs

#### 6.13.1 Purpose

The retention and acknowledgement of TPDUs procedure is used in classes 1, 3 and 4 to enable and minimize retransmission after possible loss of TPDUs.

The confirmation of receipt variant is used only in class 1 when it has been agreed during connection establishment (see the Note).

The AK variant is used in classes 3 and 4 and also in class 1 when the confirmation of receipt variant has not been agreed during connection establishment. In addition, in class 4, the option of using selective acknowledgement may be agreed to during connection establishment.

The request acknowledgement procedure is selected during connection establishment and may be used in classes 3 and 4, and in class 1 when the confirmation of receipt variant has not been agreed during connection establishment. It allows a transport entity to request acknowledgement of retained DT TPDUs by setting the ROA parameter in a transmitted DT TPDU.

NOTE – Use of the confirmation of receipt variant depends on the availability of the network layer receipt confirmation service and the expected cost reduction.

#### 6.13.2 Network service primitives

When operating over CONS, the procedure uses the following network service primitives:

- a) N-DATA;
- b) N-DATA ACKNOWLEDGE.

When operating over CLNS, the procedure uses the following network service primitive:

N-UNIT DATA.



### 6.13.3 TPDU and parameters used

The procedure uses the following TPDU and parameters:

- a) CR, CC, DR and DC TPDU;
- b) AK TPDU
  - YR-TU-NR
  - selective acknowledgement parameters;
- c) RJ TPDU
  - YR-TU-NR;
- d) DT TPDU
  - TPDU-NR;
- e) ED TPDU
  - ED-TPDU-NR;
- f) EA TPDU
  - YR-EDTU-NR.

### 6.13.4 Procedures

#### 6.13.4.1 Retention until acknowledgement of TPDU

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

CR, CC, DR, DT and ED TPDU.

except in the following case: if a DR TPDU is sent in response to a CR TPDU there is no need to retain a copy of the DR TPDU.

A copy of each of these TPDU shall be retained until

- a) it is acknowledged, as specified in Table 5; or
- b) the transport connection is released.

TABLE 5/X.224

**Acknowledgement of TPDU**

| Retained TPDU | Variant                         | Retained until acknowledged by:  |
|---------------|---------------------------------|--|
| CR            | Both                            | CC, DR or ER TPDU  |
| DR            | Both                            | DC or DR (in case of collision) TPDU   |
| CC            | Confirmation of receipt variant | N-DATA ACKNOWLEDGE indication, RJ, DT, EA or ED TPDU   |
| CC            | AK variant                      | RJ, DT, AK, ED or EA TPDU  |
| DT            | Confirmation of receipt variant | N-DATA ACKNOWLEDGE indication corresponding to an N-DATA request which conveyed, or came after, the DT TPDU  |
| DT            | AK variant                      | AK or RJ TPDU for which the YR-TU-NR is greater than TPDU-NR in the DT TPDU. In case of selective acknowledgement, if the selective acknowledgement parameters in the AK TPDU include the TPDU-NR of the DT TPDU |
| ED            | Both                            | EA TPDU for which the YR-EDTU-NR is equal to the ED-TPDU-NR in the ED TPDU   |

#### 6.13.4.2 Confirmation of receipt variant

In the confirmation of receipt variant, applicable only in class 1, transport entities shall

- a) set the confirmation request parameter only if the data parameter contains a CC or DT TPDU (see Notes 1 and 2 in 6.13.4.4);
- b) issue an N-DATA ACKNOWLEDGE request when it receives an N-DATA indication with the confirmation request parameter set.

#### 6.13.4.3 Request of acknowledgement option

If the request acknowledgement procedure has been negotiated, transport entities

- a) may request acknowledgement of retained DT TPDU by setting the ROA parameter in a transmitted DT TPDU. The decision as to when the sending transport entity should request acknowledgement is a local matter (see Note 4 in 6.13.4.4);
- b) on receipt of a DT TPDU with the ROA parameter set shall transmit an AK TPDU containing up-to-date window information.

#### 6.13.4.4 Selective acknowledgement option

If the selective acknowledgement option has been negotiated, transport entities

- a) may include selective acknowledgement parameters in a transmitted AK TPDU. These selective acknowledgement parameters, if included, shall contain acknowledgement of blocks of TPDU not acknowledged by the YR-TU-NR field of the AK TPDU. This procedure allows transport entities to acknowledge DT TPDU that are within the window but that are not in sequence;
- b) on receipt of an AK TPDU containing selective acknowledgement parameter(s) shall discard the DT TPDU specified.

NOTES (Notes 1 to 3 only apply when operating over CONS)

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 TPDU.

2 Use of the confirmation request parameter may affect the quality of network service.

3 In class 3, and in class 1, when use of explicit AK variant is selected, if a transport entity does not send an AK TPDU after reception of each DT TPDU, it is recommended that it

- starts a timer after reception of DT TPDU;
- sends an AK TPDU with up-to-date window information at expiration of the timer if an AK TPDU with the same window information has not been previously sent.

Selection of the value of this timer is a local matter but may affect performance.

4 It is recommended that, if the sending transport entity has a restriction in the number of DT TPDU that it can retain, then it set the ROA parameter to avoid a delay in transmitting DT TPDU due to the remote transport entity operating an AK withholding policy.

### 6.14 Resynchronization

#### 6.14.1 Purpose

The resynchronization procedures are used in classes 1 and 3 to restore the transport connection to normal after a reset or during reassignment after failure according to 6.12.

#### 6.14.2 Network service primitives

The procedure uses the following network service primitive:

N-RESET indication.

#### 6.14.3 TPDU and parameters used

The procedure uses the following TPDU and parameters:

- a) CR, DR, CC and DC TPDU;
- b) RJ TPDU
  - YR-TU-NR;

- c) DT TPDU
  - TPDU-NR;
- d) ED TPDU
  - ED-TPDU-NR;
- e) EA TPDU
  - YR-EDTU-NT.

#### 6.14.4 Procedure

A transport entity which is notified of the occurrence of a N-RESET shall:

- a) if the transport entity is the responder, carry out the passive resynchronization procedure (see 6.14.4.2);
- b) if the transport entity has elected not to reassign, do nothing;
- c) otherwise, execute the active resynchronization procedure described in 6.14.4.1 and, if 6.14.4.1 b) has been performed, wait for the next event as follows:
  - if a valid TPDU is received as the result of the resynchronization, stop the TTR timer, or
  - if TTR runs out, wait for the next event, or
  - if an N-RESET indication is received, perform 6.14.4.

##### 6.14.4.1 Active resynchronization procedures

The transport entity shall carry out one of the following actions:

- a) if the TTR timer has been previously started and has run out (i.e. no valid TPDU has been received), the procedures defined in 6.12.3 a) 3) shall apply;
- b) otherwise, the TTR timer shall be started (unless it is already running) and the first which become applicable of the following actions shall be taken:
  - 1) if a CR TPDU is unacknowledged, then the transport entity shall retransmit it;
  - 2) if a DR TPDU is unacknowledged, then the transport entity shall retransmit it;
  - 3) otherwise, the transport entity shall carry out the data resynchronization procedures (6.14.4.3).

##### 6.14.4.2 Passive resynchronization procedures

The transport entity shall not send any TPDU until a TPDU has been received. The transport entity shall start its TWR timer if it has not already been started (due to a previous N-DISCONNECT or N-RESET indication). If the timer runs out prior to the receipt of a valid TPDU which commences resynchronization (i.e. CR or DR or ED or RJ TPDU), the transport connection is considered as released and the reference is frozen (see 6.18).

When a valid TPDU is received the transport entity shall stop its TWR timer and carry out one of the following appropriate actions, depending on the TPDU:

- a) if it is a DR TPDU, then the transport entity shall send a DC TPDU;
- b) if it is a repeated CR TPDU (see Note 1) the transport entity shall carry out the appropriate action from the following:
  - 1) if a CC TPDU has already been sent, and acknowledged: treat as a protocol error;
  - 2) if the responder wants to release the transport connection or refuse the CR TPDU: (re)transmit the DR TPDU, setting the source reference to zero;
  - 3) if the T-CONNECT response has not yet been received from the user: take no action;
  - 4) otherwise: (re)transmit the CC TPDU, followed by retransmission of any unacknowledged ED TPDU (see Note 2) and retransmission of the unacknowledged DT TPDU, subject to any applicable flow control procedures.

## NOTES

1 A repeated CR TPDU can be identified by being on a network connection with the appropriate network addresses and having a correct source reference.

2 The transport entity should not use network expedited until the CC TPDU is acknowledged (see 6.5). This rule prevents the network expedited from overtaking the CC TPDU.

- c) if it is an RJ or ED TPDU, then one of the following actions shall be taken:
  - 1) if a DR TPDU is unacknowledged, then the transport entity shall retransmit it;
  - 2) if a CC TPDU is unacknowledged, the RJ or ED TPDU shall be considered as acknowledging the CC TPDU, and the transport entity shall carry out the data resynchronization procedures (6.14.4.3);
  - 3) otherwise, the transport entity shall carry out the data resynchronization procedures (6.14.4.3).

### 6.14.4.3 Data resynchronization procedures

The transport entity shall carry out the following actions in the following order:

- a) (re)transmit any ED TPDU which is unacknowledged.
- b) transmit an RJ TPDU with YR-TU-NR field set to the TPDU-NR of the next expected DT TPDU;
- c) wait for the next TPDU from the other transport entity, unless an RJ or DR TPDU has already been received; if a DR TPDU is received the transport entity shall send a DC TPDU, freeze the reference, inform the TS-user of the disconnection and take no further action [i.e. it shall not follow the procedures in 6.14.4.3 d)]. If an RJ TPDU is received, the procedure of 6.14.4.3 d) shall be followed. If an ED TPDU is received the procedures as described in 6.11 shall be followed. If it is a duplicated ED-TPDU the transport entity shall acknowledge it with an EA TPDU, discard the duplicated ED TPDU and wait again for the next TPDU;
- d) (re)transmit any DT TPDUs which are unacknowledged, subject to any applicable flow control procedures (see the Note).

NOTE – The RJ TPDU may have reduced the credit.

## 6.15 Multiplexing and demultiplexing when operating over CONS

### 6.15.1 Purpose

The multiplexing and demultiplexing procedures are used in classes 2, 3 and 4 to allow several transport connections to share a network connection at the same time.

### 6.15.2 TPDUs and parameters used

The procedure uses the following TPDUs and parameters:

CC, DR, DC, DT, AK, ED, EA, RJ, and ER TPDUs

– DST-REF.

### 6.15.3 Procedure

The transport entities shall be able to send and receive on the same network connection TPDUs belonging to different transport connections.

#### NOTES

1 When performing demultiplexing, the transport connection to which the TPDUs apply is determined by the procedures defined in 6.9.

2 Multiplexing allows the concatenation of TPDUs belonging to different transport connections to be transferred in the same N-DATA primitive (see 6.4).

## 6.16 Explicit flow control

### 6.16.1 Purpose

The explicit flow control procedure is used in classes 2, 3 and 4 to regulate the flow of DT TPDUs independently of the flow control in the other layers.

### 6.16.2 TPDUs and parameters used

The procedure uses the following TPDUs and parameters:

- a) CR, CC, AK and RJ TPDUs
  - CDT;
- b) DT TPDU
  - TPDU-NR;
  - ROA;
- c) AK TPDU
  - YR-TU-NR;
  - subsequence number;
  - flow control confirmation;
  - selective acknowledgement parameters;
- d) RJ TPDU
  - YR-TU-NR.

### 6.16.3 Procedure

The procedures differ in different classes. They are defined in the clauses specifying the separate classes.

## 6.17 Checksum

### 6.17.1 Purpose

The checksum procedure is used to detect corruption of TPDUs by the NS-provider.

NOTE – Although a checksum algorithm has to be adapted to the type of errors expected on the network connection, at present, only one algorithm is defined.

### 6.17.2 TPDUs and parameters used

The procedure uses the following TPDUs and parameters:

- All TPDUs
  - checksum.

### 6.17.3 Procedure

The checksum shall be used only in class 4. It shall always be used for the CR TPDU, and shall be used for all other TPDUs unless the non-use of the checksum was selected during connection establishment.

The sending transport entity shall transmit TPDUs with the checksum parameter set such that the following formulae are satisfied:

$$\sum_{i=1}^L a_i = 0 \text{ (modulo 255)}$$

$$\sum_{i=1}^L ia_i = 0 \text{ (modulo 255)}$$

where

- $i$  is the number (i.e. position) of an octet within the TPDU (see 13.2);
- $a_i$  is the value of octet in position  $i$ ;
- $L$  is the length of TPDU in octets.

A transport entity which receives a TPDU for a transport connection for which the use of the checksum has been agreed and which does not satisfy the above formulae shall discard the TPDU (see also Note 2).

When a spurious TPDU is received and an answer has to be sent, the transport entity shall

- a) if it supports the checksum algorithm and the received TPDU contains a checksum parameter, include a checksum parameter in the answering TPDU; or
- b) in all other cases, not include a checksum parameter in the answering TPDU.

An entity not supporting the checksum may always suppose that a CR TPDU with class 4 proposed is correct and therefore negotiate down to a class lower than 4.

#### NOTES

- 1 An efficient algorithm for determining the checksum parameters is given in Annex B.
- 2 If the checksum is incorrect, it is impossible to know with certainty to which transport connection the TPDU is related; further action may be required dependent on the type of network service in use (see 6.9.1 for CONS and 6.9.2 for CLNS).
- 3 The checksum proposed is easy to calculate and so will not impose a heavy burden on implementations. However, it will not detect insertion or loss of leading or trailing zeros and will not detect some octets misordering.
- 4 When CONS is used and a TPDU is received on a network connection, it is impossible to know with certainty that only class 4 transport connections use this network connection as it may be a TPDU performing reassignment.

Consequently, the only way to check the validity is as follows:

- a) if the network connection is used by a class 0 or class 1 transport connection, there is no checksum;
- b) examine the TPDU code;
- c) deduce the fixed part length;
- d) from LI, deduce the variable part;
- e) go through parameters and if the checksum parameter is found, then verify it;
- f) if it is incorrect, then assume that transport connection is class 4 and drop it;
- g) if it is correct, then associate the TPDU with a transport connection; if the transport connection uses the checksum, it is correct; otherwise, it shall be considered as a protocol error.

## 6.18 Frozen references

### 6.18.1 Purpose

This procedure shall be used in order to prevent re-use of a reference while TPDUs associated with the old use of the reference may still exist.

### 6.18.2 Procedure

When a transport entity determines that a particular connection is released it shall place the reference which it has allocated to the connection in a frozen state according to the procedures of the class. While frozen, the reference shall not be re-used.

NOTE – The frozen reference procedure is necessary because retransmission or misordering can cause TPDUs bearing a reference to arrive at an entity after it has released the connection for which it allocated the reference. Retransmission, for example, can arise when the class includes either resynchronization (see 6.14) or retransmission on time-out (see 6.19).

#### 6.18.2.1 Procedure for classes 0 and 2

This Recommendation does not specify frozen reference procedures for classes 0 and 2.

NOTE – For consistency with other classes, references may be frozen as a local matter.

#### 6.18.2.2 Procedure for classes 1 and 3

The frozen reference procedure is used except in the following cases (see Note 1):

- a) when the transport entity receives a DC TPDU in response to a DR TPDU which it has sent (see Note 2);
- b) when the transport entity sends a DR or ER TPDU in response to a CR TPDU which it has received (see Note 3);
- c) when the transport entity has considered the connection to be released after the expiration of the TWR timer (see Note 4);

- d) when the transport entity receives a DR or ER TPDU in response to a CR TPDU which it has sent;
- e) when the reference is zero.

The period of time for which the reference remains frozen shall be greater than the TWR time.

#### NOTES

- 1 However, even in these cases, for consistency freezing the reference may be done as a local decision.
- 2 When the DC TPDU is received it is certain that the other transport entity considers the connection released.
- 3 When the DR or ER TPDU is sent the peer transport entity has not been informed of any reference assignment and thus cannot possibly make use of a reference (this includes the case where a CC TPDU was sent, but was lost).
- 4 In c) the transport entity has already effectively frozen the reference for an adequate period.

### **6.18.2.3 Procedure for class 4**

The frozen reference procedure shall be used in class 4. The period for which the reference remains frozen shall be greater than  $L$  (see 12.2.1.1.6).

## **6.19 Retransmission on time-out**

### **6.19.1 Purpose**

The procedure is used in class 4 to cope with unsignalled loss of TPDUs by the NS provider.

### **6.19.2 TPDUs used**

The procedure uses the following TPDUs:

CR, CC, DR, DT, ED, AK TPDUs.

### **6.19.3 Procedure**

The procedure is specified in the procedures for class 4 [see 12.2.1.2 i) and 12.2.1.3 g)].

## **6.20 Resequencing**

### **6.20.1 Purpose**

The resequencing procedure is used in class 4 to cope with misordering of TPDUs by the network service provider.

### **6.20.2 TPDUs and parameters used**

The procedure uses the following TPDUs and parameters:

- a) DT TPDU
  - TPDU-NR;
- b) ED TPDU
  - ED TPDU-NR.

### **6.20.3 Procedure**

The procedure is specified in the procedures for class 4 (see 12.2.3.5).

## **6.21 Inactivity control**

### **6.21.1 Purpose**

The inactivity control procedure is used in class 4 to cope with unsignalled termination of a network connection when using CONS and the failure of a remote transport entity when using CONS or CLNS.

## 6.21.2 Procedure

The procedure is specified in the procedures for class 4 (see 12.2.3.3).

## 6.22 Treatment of protocol errors

### 6.22.1 Treatment of protocol errors when operating over CONS

#### 6.22.1.1 Purpose

The procedure for treatment of protocol errors is used in all classes to deal with invalid TPDU.

#### 6.22.1.2 TPDU and parameters used

The procedure uses the following TPDU and parameters:

- a) ER TPDU
  - reject cause;
  - invalid TPDU;
- b) DR TPDU
  - reason code.

#### 6.22.1.3 Procedure

A transport entity that receives a TPDU that can be associated to a transport connection and is invalid or constitutes a protocol error (see 3.2.16 and 3.2.17) shall take one of the following actions so as not to jeopardize any other transport connections not assigned to that network connection:

- a) transmit an ER TPDU;
- b) reset or close the network connection; or
- c) invoke the release procedures appropriate to the class.

Under certain circumstances it is also possible to discard the TPDU.

If an ER TPDU is sent in class 0 it shall contain the octets of the invalid TPDU up to and including the octet where the error was detected (see Notes 3, 4 and 5).

If the TPDU cannot be associated with a particular transport connection the transport entity shall follow the procedures in 6.9.

#### NOTES

1 In general, no further action is specified for the receiver of the ER TPDU but it is recommended that it initiates the release procedure appropriate to the class. If the ER TPDU has been received as an answer to a CR TPDU then the connection is regarded as released (see 6.6).

2 Care should be taken by a transport entity receiving several invalid TPDU or ER TPDU to avoid looping if the error is generated repeatedly.

3 If the invalid received TPDU is greater than the selected maximum TPDU size-inclusion in the invalid TPDU parameter of the ER TPDU may not be possible.

4 It is recommended that the sender of the ER TPDU starts an optional timer TS2 to ensure the release of the connection. If the timer expires, the transport entity shall initiate the release procedures appropriate to the class. The timer should be stopped when a DR TPDU or an N-DISCONNECT indication is received.

5 In classes other than 0, it is recommended that the invalid TPDU be also included in the ER TPDU.

### 6.22.2 Treatment of protocol errors when operating over CLNS

#### 6.22.2.1 Purpose

The procedure for treatment of protocol errors is used to deal with invalid TPDU.



### 6.22.2.2 TPDU and parameters used

The procedure uses the following TPDU and parameters:

- a) ER TPDU
  - reject cause;
  - invalid TPDU;
- b) DR TPDU
  - reason.

### 6.22.2.3 Procedure

Invalid TPDU and protocol errors shall be ignored (no action and TPDU discarded, or responded to with an ER TPDU), except for the following case: a CC TPDU is received in which the class field does not specify class 4 and a previously sent CR TPDU has not yet been acknowledged. In this case, the transport connection shall be terminated (see 6.7).

NOTE – It is recommended that the sender of the ER TPDU starts an optional timer TS2 to ensure the release of the connection. If the timer expires, the transport entity shall initiate the release procedure appropriate to class 4. The timer should be stopped when a DR TPDU is received.

## 6.23 Splitting and recombining when operating over CONS

### 6.23.1 Purpose

This procedure is used only in class 4 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.

### 6.23.2 Procedure

When this procedure is being used, a transport connection may be assigned (see 6.1) to multiple network connections (see Note 1). TPDU for the connection may be sent over any such network connection.

If the use of class 4 is not accepted by the remote transport entity following the negotiation rules, then no network connection except that over which the CR TPDU was sent may have the transport connection assigned to it.

#### NOTES

- 1 The resequencing function of class 4 (see 6.20) is used to ensure that TPDU are processed in the correct sequence.
- 2 Either transport entity may assign the connection to further network connections of which it is the owner at any time during the life of the transport connection, provided the following constraints are respected:
  - the initiator does not start splitting before having received the CC TPDU;
  - as soon as a new assignment is carried out it is recommended to send a TPDU on this network connection in order to make the remote entity aware of this assignment.
- 3 A transport entity performing splitting should ensure that TPDU are sent at intervals on each supporting network connection, for example, by sending successive TPDU on successive network connections, where the set of network connections is used cyclically.

When splitting is used the inactivity control procedure defined in 12.2.3.3 will not normally detect unsignalled network connection failure. Any method of monitoring network connections to detect such failure is a local matter.

## 7 Protocol classes

Table 6 gives an overview of which elements of procedure are included in each class. In certain cases, the elements of procedure within different classes are not identical and, for this reason, Table 6 cannot be considered as part of the definitive specification of the protocol.

TABLE 6/X.224

## Allocation of elements of procedures within classes

| Protocol mechanism  | Cross-reference | Variant or Option   | 0 | 1                             | 2                        | 3          | 4<br>CONS       | 4<br>CLNS       |
|---|-----------------|---|---|-------------------------------|--------------------------|------------|-----------------|-----------------|
| Assignment to network connection  | 6.1.1           |   | x | x                             | x                        | x          | x               |                 |
| TPDU transfer   | 6.2             |   | x | x                             | x                        | x          | x               | x               |
| Segmenting and reassembling   | 6.3             |   | x | x                             | x                        | x          | x               | x               |
| Concatenation and separation  | 6.4             |   |   | x                             | x                        | x          | x               | x               |
| Connection establishment  | 6.5             |   | x | x                             | x                        | x          | x               | x               |
| Connection refusal  | 6.6             |   | x | x                             | x                        | x          | x               | x               |
| Normal release  | 6.7             | Implicit<br>Explicit  | x | x                             | x                        | x          | x               | x               |
| Error release   | 6.8             |   | x |                               | x                        |            |                 |                 |
| Association of TPDU's with transport connection   | 6.9             |   | x | x                             | x                        | x          | x               | x               |
| TPDU numbering  | 6.10            | Normal<br>Extended  |   | x                             | m (Note 1)<br>o (Note 1) | m<br>o     | m<br>o          | m<br>o          |
| Expedited data transfer   | 6.11            | Network Normal<br>Network<br>Expedited  |   | m<br>ao                       | x (Note 1)               | x          | x               | x               |
| Reassignment after failure  | 6.12            |   |   | x                             |                          | x          | (Note 3)        |                 |
| Retention and acknowledgement of TPDU's   | 6.13            | Confirmation of receipt<br>AK<br>Use of selective acknowledgement<br>Use of request acknowledgement |   | ao<br><br>m<br><br>o (Note 4) |                          | x<br><br>o | x<br><br>o<br>o | x<br><br>o<br>o |
| Resynchronization   | 6.14            |   |   | x                             |                          | x          | (Note 3)        |                 |
| Multiplexing and demultiplexing   | 6.15            |   |   |                               | x (Note 2)               | x          | x               |                 |
| Explicit flow control (with)<br>Explicit flow control (without)   | 6.16            |   | x | x                             | m<br>o                   | x          | x               | x               |
| Checksum (use of)<br>Checksum (non-use of)  | 6.17            |   | x | x                             | x                        | x          | m<br>o          | m<br>o          |
| Frozen references   | 6.18            |   |   | x                             |                          | x          | x               | x               |
| Retransmission on time-out  | 6.19            |   |   |                               |                          |            | x               | x               |
| Resequencing  | 6.20            |   |   |                               |                          |            | x               | x               |
| Inactivity control  | 6.21            |   |   |                               |                          |            | x               | x               |
| Treatment of protocol errors  | 6.22            |   | x | x                             | x                        | x          | x               | x               |
| Splitting and recombining   | 6.23            |   |   |                               |                          |            | x               |                 |
| <p>x Procedure always included in class<br/>Not applicable</p> <p>m Negotiable procedure whose implementation in equipment is mandatory</p> <p>o Negotiable procedure whose implementation in equipment is optional</p> <p>ao Negotiable procedure whose implementation in equipment is optional and where use depends of availability within the network service</p> <p>NOTES</p> <p>1 Not applicable in class 2 when non-use of explicit flow control is selected.</p> <p>2 Multiplexing may lead to degradation of the quality of service if the non-use of explicit flow control has been selected.</p> <p>3 This function is provided in class 4 using procedures other than those used in the cross-reference.</p> <p>4 This option is not applicable in class 1, when the confirmation of receipt variant has been selected.</p> |                 |   |   |                               |                          |            |                 |                 |

## **8 Specification for class 0: Simple class**

### **8.1 Functions of class 0**

Class 0 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.

### **8.2 Procedures for class 0**

#### **8.2.1 Procedures applicable at all times**

The transport entities shall use the following procedures:

- a) TPDU transfer (see 6.2);
- b) association of TPDU with transport connections (see 6.9);
- c) treatment of protocol errors (see 6.22);
- d) error release (see 6.8).

#### **8.2.2 Connection establishment**

The transport entities shall use the following procedures:

- a) assignment to network connection (see 6.1.1); then
- b) connection establishment (see 6.5) and, if appropriate, connection refusal (see 6.6);

subject to the following constraints:

- 1) the CR and CC TPDU shall contain no parameter fields in the variable part of the header other than those for TSAP-ID, maximum TPDU size, and preferred maximum TPDU size;
- 2) the CR and CC TPDU shall not contain a data field.

#### **8.2.3 Data transfer**

The transport entities shall use the segmenting and reassembling procedure (see 6.3).

#### **8.2.4 Release**

The transport entities shall use the implicit variant of the normal release procedure (see 6.7.1.4).

NOTE – The lifetime of the transport connection is directly correlated with the lifetime of the network connection.

## **9 Specification for class 1: Basic error recovery class**

### **9.1 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 TS-user.

## 9.2 Procedures for class 1

### 9.2.1 Procedures applicable at all times

The transport entities shall use the following procedures:

- a) TPDU transfer (see 6.2);
- b) association of TPDU with transport connections (see 6.9);
- c) treatment of protocol errors (see 6.22);
- d) reassignment after failure (see 6.12);
- e) resynchronization (see 6.14), or reassignment after failure (see 6.12) together with resynchronization (see 6.14);
- f) concatenation and separation (see 6.4);
- g) retention and acknowledgement of TPDU's (see 6.13); the variant used, AK or confirmation of receipt, shall be as selected during connection establishment (see the notes);
- h) frozen references (see 6.18).

#### NOTES

1 The negotiation of the variant of retention and acknowledgement of TPDU's procedure to be used over the transport connection has been designed such that if the initiator proposes the use of the AK variant (i.e. the mandatory implementation option), the responder has to accept use of this option and if the initiator proposes use of the confirmation of receipt variant the responder is entitled to select use of the AK variant.

2 The AK variant makes use of AK TPDU's to release copies of retained DT TPDU's. The CDT parameter of AK TPDU's in class 1 is not significant, and is set to 1111.

3 The confirmation of receipt variant is restricted to this class and its use depends on the availability of the network layer receipt confirmation service, and the expected cost reduction.

### 9.2.2 Connection establishment

The transport entities shall use the following procedures:

- a) assignment to network connection (see 6.1.1); then
- b) connection establishment (see 6.5) and, if appropriate, connection refusal (see 6.6).

### 9.2.3 Data transfer

#### 9.2.3.1 General

The sending transport entity shall use the following procedures:

- a) segmenting (see 6.3); then
- b) the normal format variant of DT TPDU numbering (see 6.10).

The receiving transport entity shall use the following procedures:

- 1) the normal format variant of DT TPDU numbering (see 6.10); then
- 2) reassembling (see 6.3).

NOTE – The decision to issue an N-RESET request in order to force the remote entity to carry out the resynchronization (see 6.14) may be made on a local basis.

#### 9.2.3.2 Expedited data

The transport entities shall use either the network normal data or the network expedited variants of the expedited data transfer procedure (see 6.11) if their use has been selected during connection establishment (see Note 1).

The sending transport entity shall not allocate the same EDTPDU-NR to successive ED TPDU's (see Notes 2 and 3).

When acknowledging an ED TPDU by sending an EA TPDU the transport entity shall put into the YR-EDTU-NR parameter of the EA TPDU the value received in the ED-TPDU-NR parameter of the ED TPDU.

## NOTES

1 The negotiation of the variant of expedited data transfer procedure to be used over the transport connection has been designed such that if the initiator proposes the use of the network normal data variant (i.e. the mandatory implementation option), the responder has to accept use of this option and if the initiator proposes use of the network expedited variant, the responder is entitled to select use of the network normal data variant.

2 This numbering enables the receiving transport entity to discard repeated ED TPDUs when resynchronization (see 6.14) has taken place.

3 No other significance is attached to the ED-TPDU-NR parameter. It is recommended, but not essential, that the values used be consecutive modulo 128.

4 The use of RJ TPDUs during resynchronization (see 6.14) can lead to retransmission. Thus, the receipt of a duplicate ED TPDU is possible. Such an ED TPDU is discarded.

### 9.2.4 Release

The transport entities shall use the explicit variant of the release procedure (see 6.7.1.5).

## 10 Specification for class 2: Multiplexing class

### 10.1 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 connection resets or disconnects, the transport connection is terminated without the transport release procedure and the TS-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 that the expedited data transfer is available.

### 10.2 Procedures for class 2

#### 10.2.1 Procedures applicable at all times

The transport entities shall use the following procedures

- a) association of TPDUs with transport connection (see 6.9);
- b) TPDU transfer (see 6.2);
- c) treatment of protocol errors (see 6.22.1);
- d) concatenation and separation (see 6.4);
- e) error release (see 6.8).

Additionally the transport entities may use the following procedures:

- f) multiplexing and demultiplexing (see 6.15).

#### 10.2.2 Connection establishment

The transport entities may use the following procedures:

- a) assignment to network connection (see 6.1.1); then
- b) connection establishment (see 6.5) and, if applicable, connection refusal (see 6.6).

#### 10.2.3 Data transfer when non-use of explicit flow control has been selected

If this option has been selected as a result of the connection establishment, the transport entities shall use the segmenting procedure (see 6.3).

The TPDU-NR field of DT TPDUs is not significant and may take any value.

NOTE – Expedited data transfer is not applicable (see 6.5).

## 10.2.4 Data transfer when use of explicit flow control has been selected

### 10.2.4.1 General

The sending transport entity shall use the following procedures:

- a) segmenting (see 6.3); then
- b) DT TPDU numbering (see 6.10).

The receiving transport entity shall use the following procedures:

- 1) DT TPDU numbering (see 6.10); if a DT TPDU is received which is out of sequence it shall be treated as a protocol error; then
- 2) reassembling (see 6.3).

The variant of the DT TPDU numbering which is used by both transport entities shall be that which was agreed at connection establishment.

### 10.2.4.2 Flow control

The transport entities shall send an initial credit (which may be zero) in the CDT field of the CR or CC TPDU. This credit represents the initial value of the upper window edge allocated to the peer entity.

The transport entity that receives the CR or the CC TPDU shall consider its lower window edge as zero, and its upper window edge as the value of the CDT field in the received TPDU.

In order to authorize the transmission of DT TPDU, by its peer, a transport entity may transmit an AK TPDU at any time, subject to the following constraints:

- a) the YR-TU-NR parameter shall be at most one greater than the TPDU-NR field of the last received DT TPDU or shall be zero if no DT TPDU has been received;
- b) if an AK TPDU has previously been sent the value of the YR-TU-NR parameter shall not be lower than that in the previously sent AK TPDU;
- c) the sum of the YR-TU-NR and CDT fields shall not be less than the upper window edge allocated to the remote entity (see Note 1).

A transport entity which receives an AK TPDU shall consider the YR-TU-NR field as its new lower window edge, and the sum of YR-TU-NR and CDT as its new upper window edge. If either of these have been reduced or if the lower window edge has become more than one greater than the TPDU-NR of the last transmitted DT TPDU, this shall be treated as a protocol error (see 6.22.1).

A transport entity shall not send a DT TPDU with a TPDU-NR outside of the transmit window (see Notes 2 and 3).

#### NOTES

- 1 This means that credit reduction is not applicable.
- 2 This means that a transport entity is required to stop sending if the TPDU-NR field of the next DT TPDU which would be sent would be the upper window edge. Sending of DT TPDU may be resumed if an AK TPDU is received which increases the upper window edge.
- 3 The rate at which a transport entity progresses the upper window edge allocated to its peer entity constrains the throughput attainable on the transport connection.

### 10.2.4.3 Expedited data

The transport entities shall follow the network normal data variant of the expedited data transfer procedure in 6.11.1 if its use has been agreed during connection establishment. ED and EA TPDU are not subject to the flow control procedures in 10.2.4.2. The ED-TPDU-NR and YR-ETDU-NR fields of ED and EA TPDU respectively are not significant and may take any value.

## 10.2.5 Release

The transport entities shall use the explicit variant of the release procedure in 6.7.1.

## **11 Specification for class 3: Error recovery and multiplexing class**

### **11.1 Functions of class 3**

Class 3 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 TS-user.

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

### **11.2 Procedures for class 3**

#### **11.2.1 Procedures applicable at all times**

The transport entities shall use the following procedures:

- a) association of TPDU's with transport connections (see 6.9);
- b) TPDU transfer (see 6.2) and retention and acknowledgement of TPDU's (AK variant only) (see 6.13);
- c) treatment of protocol errors (see 6.22.1);
- d) concatenation and separation (see 6.4);
- e) reassignment after failure (see 6.12), together with resynchronization (see 6.14);
- f) frozen references (see 6.18).

Additionally, the transport entities may use the following procedure:

- g) multiplexing and demultiplexing (see 6.15).

#### **11.2.2 Connection establishment**

The transport entities shall use the following procedures:

- a) assignment to network connections (see 6.1.1); then
- b) connection establishment (see 6.5) and, if appropriate, connection refusal (see 6.6).

#### **11.2.3 Data transfer**

##### **11.2.3.1 General**

The sending transport entity shall use the following procedures:

- a) segmenting (see 6.3); then
- b) DT TPDU numbering (see 6.10); after receipt of an RJ TPDU (see 11.2.3.2) the next DT TPDU to be sent may have a value which is not the previous value of TPDU-NR plus one.

The receiving transport entity shall use the following procedures:

- 1) DT TPDU numbering (see 6.10); the TPDU-NR field of each received DT TPDU shall be treated as a protocol error if it exceeds the greatest value received in a previous DT TPDU by more than one (see the Note); then
- 2) Reassembling (see 6.3); duplicated TPDU's shall be eliminated before reassembling is performed.

NOTE – The use of RJ TPDU's (see 11.2.3.2) can lead to retransmission and reduction of credit. Thus the receipt of a DT TPDU which is a duplicate, or which is greater than or equal to the upper window edge allocated to the peer entity, is possible and is therefore not treated as a protocol error.

##### **11.2.3.2 Use of an RJ TPDU**

A transport entity may send an RJ TPDU at any time in order to invite retransmission or to reduce the upper window edge allocated to the peer entity (see Note 1).

When an RJ TPDU is sent, the following constraints shall be respected:

- a) the YR-TU-NR parameter shall be at most one greater than the greatest value received in a previous DT TPDU, or shall be zero if no DT TPDU has yet been received (see Note 2);
- b) if an AK or RJ TPDU has been sent previously the YR-TU-NR parameter shall not be lower than that in the AK or RJ TPDU sent previously.

When a transport entity receives an RJ TPDU (see Note 3):

- c) the next DT TPDU to be transmitted, or retransmitted, shall be that for which the value of the TPDU-NR parameter is equal to the value of the YR-TU-NR parameter of the RJ TPDU;
- d) the sum of the values of the YR-TU-NR and CDT parameters of the RJ TPDU becomes the new upper window edge (see Note 4).

#### NOTES

- 1 An RJ TPDU can also be sent as part of the resynchronization (see 6.14) and reassignment after failure (see 6.12) procedures.
- 2 It is recommended that the YR-TU-NR parameter be equal to the TPDU-NR parameter of the next expected DT TPDU.
- 3 These rules are a subset of those specified for the case when an RJ TPDU is received during resynchronization (see 6.14) and reassignment after failure (see 6.12).
- 4 This means that an RJ TPDU can be used to reduce the upper window edge allocated to the peer entity (credit reduction).

### 11.2.3.3 Flow control

The procedures shall be as defined in 10.2.4.2, except that

- a) a credit reduction may lead to the reception of a DT TPDU with a TPDU-NR parameter whose value is not, but would have been less than the upper window edge allocated to the remote entity prior to the credit reduction. This shall not be treated as a protocol error;
- b) receipt of an AK TPDU which sets the lower window edge more than one greater than the TPDU-NR of the last transmitted DT TPDU shall not be treated as a protocol error, provided that all acknowledged DT TPDU's have been previously transmitted (see Notes 1 and 2).

#### NOTES

- 1 This can only occur during retransmission following receipt of an RJ TPDU.
- 2 The transport entity may either continue retransmission as before or retransmit only those DT TPDU's, not acknowledged by the AK TPDU. In either case, copies of the acknowledged DT TPDU's need not be retained.

### 11.2.3.4 Expedited data

The transport entities shall follow the network normal data variant of the expedited data transfer procedure in 6.11.1 if its use has been agreed during connection establishment.

The sending transport entity shall not allocate the same ED-TPDU-NR to successive ED TPDU's.

The receiving transport entity shall transmit an EA TPDU with the same value in its YR-EDTU-NR parameter. If, and only if, this number is different from that of the ED TPDU perceived previously, shall it generate a T-EXPEDITED DATA indication to convey the data to the TS-user (see Note 2).

#### NOTES

- 1 No other significance is attached to the ED-TPDU-NR parameter. It is recommended, but not essential that the values be consecutive modulo  $2^n$ , where  $n$  is the number of bits of the parameter.
- 2 This procedure ensures that the TS-user does not receive data corresponding to the same ED TPDU more than once.

### 11.2.4 Release

The transport entities shall use the explicit variant of the release procedure in 6.7.1.



## 12 Specification for class 4: Error detection and recovery class

### 12.1 Functions of class 4

#### 12.1.1 Functions of class 4 when operating over CONS

Class 4 provides the functionality of class 3, plus the ability to detect and recover from lost, duplicated, or out of sequence TPDU's without involving the TS-user.

This detection of errors is made by extended use of the DT TPDU numbering of class 2 and class 3, by time-out mechanisms, and by additional procedures.

Class 4 detects signalled and unsignalled network failures (i.e. resets or disconnects or inactivity) and recovers from these failures by using time-out mechanisms.

This class detects and recovers from damaged TPDU's by using a checksum mechanism. The checksum mechanism shall be available but its use or its non-use is subject to negotiation.

This class also provides additional resilience against network failure and increased throughput capability by allowing a transport connection to make use of multiple network connections.

#### 12.1.2 Functions of class 4 when operating over CLNS

Class 4 provides flow control between peer transport entities, the capability to detect and recover from errors which occur as a result of a low grade service available from the network service provider and resilience from failure of the peer entity – the kind of errors to be detected include: TPDU loss, TPDU delivery out of sequence, TPDU duplication and TPDU corruption – these errors may affect control TPDU's as well as data TPDU's.

The detection of errors is made by use of TPDU numbering on DT, AK, ED and EA TPDU's, by time-out mechanisms and additional procedures such as the use of a checksum mechanism. The use of the checksum mechanism shall be available but its use or its non-use is subject to negotiation.

### 12.2 Procedures for class 4

#### 12.2.1 Procedures available at all times

##### 12.2.1.1 Timers used at all times

This subclause defines timers that apply at all times in class 4. These timers are listed in Table 7.

This Recommendation does not define specific values for the timers, and the derivations described in this subclause are not mandatory. The values should be chosen so that the required quality of service can be provided, given the known characteristics of the network.

Timers that apply only to specific procedures are defined under the appropriate procedure.

##### 12.2.1.1.1 NSDU lifetime ( $M_{LR}$ , $M_{RL}$ )

The Network Layer is assumed to provide, as an aspect of its grade of service, for a bound on the maximum lifetime of NSDU's in the network. This value may be different in each direction of transfer through a network between two transport entities. The values, for both directions of transfer, are assumed to be known by the transport entities. The maximum NSDU lifetime local-to-remote ( $M_{LR}$ ) is the maximum time which may elapse between the transmission of an NSDU from the local transport entity to the network and receipt of any copy of the NSDU from the network at the remote transport entity. The maximum NSDU lifetime remote-to-local ( $M_{RL}$ ) is the maximum time which may elapse between the transmission of an NSDU from the remote transport entity to the network and receipt of any copy of the NSDU from the network at the local transport entity.

TABLE 7/X.224

**Time parameters related to the operation of class 4**

| Symbol   | Name   | Definition   |
|----------|--|--|
| $M_{LR}$ | NSDU lifetime local-to-remote                  | A time bound for the maximum time which may elapse between the transmission of an NSDU by a local transport entity and the receipt of any copy of it by a remote peer entity.                    |
| $M_{RL}$ | NSDU lifetime remote-to-local                  | A time bound for the maximum time which may elapse between the transmission of an NSDU from a remote transport entity and the receipt of any copy of it by the local peer entity.                |
| $E_{LR}$ | Expected maximum transit delay local-to-remote | A time bound for the maximum delay suffered by all but a small proportion of NSDUs transferred from the local transport entity to a remote peer entity.  |
| $E_{RL}$ | Expected maximum transit delay remote-to-local | A time bound for the maximum delay suffered by all but a small proportion of NSDUs transferred from a remote peer entity to the local transport entity.  |
| $A_L$    | Local acknowledgement time                     | A time bound for the maximum time which can elapse between the receipt of a TPDU by the local transport entity from the network layer and the transmission of the corresponding acknowledgement. |
| $A_R$    | Remote acknowledgement time                    | As $A_L$ , but for the remote entity.  |
| $TI$     | Local retransmission time                      | A time bound for the maximum time the local transport entity will wait for acknowledgement before retransmitting a TPDU.   |
| $R$      | Persistence time                               | A time bound for the maximum time the local transport entity will continue to transmit a TPDU that requires acknowledgement.   |
| $N$      | Maximum number of transmissions                | A time bound for the maximum number of times which the local transport entity will continue to transmit a TPDU that requires acknowledgement.  |
| $L$      | Time bound on references and sequence numbers  | A time bound for the maximum time between the transmission of a TPDU and the receipt of any acknowledgement relating to it.  |
| $I_L$    | Local inactivity time                          | A lower bound for the time after which the local transport entity will, if it does not receive a TPDU, initiate the release procedure to terminate the transport connection. (See Note)          |
| $I_R$    | Remote inactivity time                         | A lower bound for the time after which the remote transport entity will, if it does not receive a TPDU, initiate the release procedure to terminate the transport connection. (See Note)         |
| $W$      | Window time                                    | A time bound for the maximum time a transport entity will wait before retransmitting up-to-date window information.  |

NOTE – This parameter is required for protection against unsignalled failures.

**12.2.1.1.2 Expected maximum transit delay ( $E_{LR}$ ,  $E_{RL}$ )**

The Network Layer is assumed to provide, as an aspect of its grade of service, an expected maximum transit delay for NSDUs in the network. This value may be different in each direction of transfer through a network between two transport entities. The values, for both directions of transfer, are assumed to be known by the transport entities. The expected maximum transit delay local-to-remote ( $E_{LR}$ ) is the maximum delay suffered by all but a small proportion of NSDUs transferred through the network from the local transport entity to the remote transport entity. The expected maximum transit delay remote-to-local ( $E_{RL}$ ) is the maximum delay suffered by all but a small proportion of NSDUs transferred through the network from the remote transport entity to the local transport entity.

### 12.2.1.1.3 Acknowledgement time ( $A_R, A_L$ )

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 transmission 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  $A_R$ .

### 12.2.1.1.4 Local retransmission time ( $TI$ )

The local transport entity is assumed to maintain a bound on the time it will wait for an acknowledgement before retransmitting the TPDU. Its value is given by

$$TI = E_{LR} + E_{RL} + A_R + x$$

where

$E_{LR}$  is the expected maximum transit delay local-to-remote;

$E_{RL}$  is the expected maximum transit delay remote-to-local;

$A_R$  is the remote acknowledgement time;

$x$  is the local processing time for a TPDU.

NOTE – During connection establishment the value of  $A_R$  is not known. In this case a suitable bound for  $TI$  may be established either by estimating (or having a priori knowledge of)  $A_R$  or by applying a suitable algorithm to the transport connection establishment delay QOS parameter.

### 12.2.1.1.5 Persistence time ( $R$ )

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 acknowledgement and which is not outside the current transmit window, even after credit reduction. This value is referred to as  $R$ .

The value is clearly related to the time elapsed between retransmission,  $TI$ , and the maximum number of transmissions,  $N$ . It is not less than  $TI \cdot (N - 1) + x$ , where  $x$  is a small quantity to allow for additional internal delays, the granularity of the mechanism used to implement  $TI$ , etc. 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.

### 12.2.1.1.6 Time bound of references and sequence numbers ( $L$ )

A bound for the maximum time between the decision to transmit a TPDU and the receipt of any acknowledgement relating to it ( $L$ ) is given by:

$$L = M_{LR} + M_{RL} + R + A_R$$

where

$M_{LR}$  is the NSDU lifetime local-to-remote;

$M_{RL}$  is the NSDU lifetime remote-to-local;

$R$  is the persistence time;

$A_R$  is the remote acknowledgement time.

It is necessary to wait for a period of time before reusing any reference or sequence number in order to avoid confusion when a TPDU referring to it is duplicated or delayed.

The period of time during which the sequence numbers for DT TPDUs should be frozen is the period  $L$ , starting from the time when the sequence number has fallen below the lower window edge.

#### NOTES

1 In practice, the value of  $L$  may be too 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.

2 The relationships between times discussed above are illustrated in Figures 3 and 4.

### 12.2.1.1.7 Inactivity timer ( $I_L, I_R$ )

Any transport entity is assumed to provide a lower bound for the time which can elapse without receipt of a TPDU before it will initiate the release procedure to terminate the transport connection. This value is referred to as  $I_L$ . The corresponding time given by the remote transport entity is referred to as  $I_R$ .

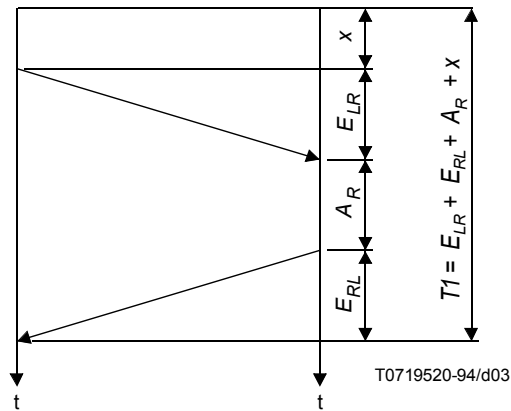


FIGURE 3/X.224

Interrelationship of times for the average case in class 4

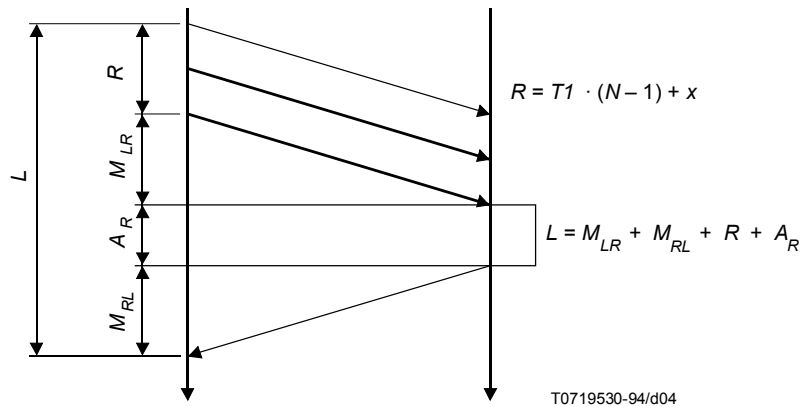


FIGURE 4/X.224

Interrelationship of times for maximum delay in class 4

### 12.2.1.2 General procedures when operating over CONS

The transport entity shall use the following procedures:

- a) TPDU transfer (see 6.2);
- b) association of TPDU's with transport connections (see 6.9.1);
- c) treatment of protocol errors (see 6.22.1);
- d) checksum (see 6.17);
- e) splitting and recombining (see 6.23);
- f) multiplexing and demultiplexing (see 6.15);

- g) retention and acknowledgement of TPDU's (see 6.13);
- h) frozen references (see 6.18);
- i) retransmission procedures; when a transport entity has some outstanding TPDU's that require acknowledgement, it will check that no  $TI$  interval elapses without the arrival of a TPDU that acknowledges at least one of the outstanding TPDU's. If the timer expires, the first TPDU is retransmitted and the timer is restarted except if the TPDU to be retransmitted is a DT TPDU and is outside the transmit window due to credit reduction. Retransmission of a TPDU is subject to the availability of a network connection. If no network connection is available, and the retransmission timer runs out, then the retransmission counter may be incremented without sending the TPDU subject to the retransmission procedure. After  $N$  transmissions (i.e.  $N - 1$  retransmissions) it is assumed that useful two-way communication is no longer possible and the release procedure is used, and the TS-user is informed;

#### NOTES

- 1 This procedure may be implemented by different means. For example:
  - a) one interval is associated with each TPDU. If the timer expires the associated TPDU will be retransmitted and the timer  $TI$  will be restarted for all subsequent TPDU's; or
  - b) one interval is associated with each transport connection:
    - 1) if the transport entity transmits a TPDU requiring acknowledgement, it starts timer  $TI$ ;
    - 2) if the transport entity receives a TPDU that acknowledges one of the TPDU's to be acknowledged, it restarts timer  $TI$  unless the received TPDU is an AK which explicitly closes the transmit window;
    - 3) if the transport entity receives a TPDU that acknowledges the last TPDU to be acknowledged, it stops timer  $TI$ .

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

2 For DT TPDU's it is a local choice to retransmit either only the first DT TPDU or all TPDU's waiting for an acknowledgement up to the upper window edge.

3 It is recommended that after  $N$  transmissions, the transport entity waits  $TI + W + M_{RL}$  in order to provide a higher possibility of receiving an acknowledgement before entering the release phase. For other TPDU types which may be retransmitted, it is recommended that after  $N$  transmissions the transport entity waits  $TI + M_{RL}$  in order to provide a greater possibility of receiving the expected reply.

4 If use of selective acknowledgement has been negotiated, a selective acknowledgement implicitly identifies DT TPDU's not received. Since such a DT TPDU could be a lost DT TPDU, or simply a delayed DT TPDU, it is a local matter whether DT TPDU's not acknowledged in a selective acknowledgement should be retransmitted immediately.

- j) concatenation and separation (see 6.4).

### 12.2.1.3 General procedures when operating over CLNS

The transport entity shall use the following procedures:

- a) TPDU transfer (see 6.2);
- b) association of TPDU's with transport connections (see 6.9.1);
- c) treatment of protocol errors (see 6.22.2);
- d) checksum (see 6.17);
- e) retention and acknowledgment of TPDU's (see 6.13);
- f) frozen references (see 6.18);
- g) retransmission procedures; when a transport entity has some outstanding TPDU's that require acknowledgment, it will check that no  $TI$  interval elapses without the arrival of a TPDU that acknowledges at least one of the outstanding TPDU's.

If the timer expires, except if the TPDU to be retransmitted is a DT TPDU and it is outside the transmit window due to credit reduction, the first TPDU is retransmitted and the timer is restarted. After  $N$  transmissions (i.e.  $N - 1$  retransmissions) it is assumed that useful two-way communication is no longer possible and the release procedure is used, and the TS-user is informed;

## NOTES

- 1 This procedure may be implemented by different means. For example:
  - a) one interval is associated with each TPDU. If the timer expires the associated TPDU will be transmitted and the timer  $TI$  will be restarted for all subsequent TPDU's; or
  - b) one interval is associated with each transport connection:
    - 1) if the transport entity transmits a TPDU requiring acknowledgment, it starts timer  $TI$ ;
    - 2) if the transport entity receives a TPDU that acknowledges one of the TPDU's to be acknowledged, it restarts timer  $TI$  unless the received TPDU is an AK which explicitly closes the transmit window;
    - 3) if the transport entity receives a TPDU that acknowledges the last TPDU to be acknowledged, it stops timer  $TI$ .

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

2 For DT TPDU's it is a local choice to retransmit either only the first DT TPDU or all TPDU's waiting for an acknowledgment up to the upper window edge.

3 It is recommended that after  $N$  transmissions, the transport entity waits  $TI + W + M_{RL}$  to provide a higher possibility for receiving an acknowledgment before entering the release phase. For other TPDU types which may be retransmitted, it is recommended that after  $N$  transmissions the transport entity waits  $TI + M_{RL}$  to provide a higher possibility of receiving the expected reply.

4 If use of selective acknowledgement has been negotiated, a selective acknowledgement implicitly identifies DT TPDU's not received. Since such a DT TPDU could be a lost DT TPDU, or simply a delayed DT TPDU, it is a local matter whether DT TPDU's not acknowledged in a selective acknowledgement should be retransmitted immediately.

- h) concatenation and separation (see 6.4).

### 12.2.2 Procedures for connection establishment

#### 12.2.2.1 Timers used in connection establishment

There are no timers specific to connection establishment.

#### 12.2.2.2 General procedures when operating over CONS

The transport entities shall use the following procedures:

- a) assignment to network connection (see 6.1.1);

When a network connection to which the transport connection is assigned is released (NDISind received):

  - 1) if a CC TPDU is awaited the initiator shall perform a new assignment according to QOS and the retransmission procedure (i.e. not sending the CR TPDU for more than  $N \cdot TI$ );
  - 2) if there is at least one other network connection to which the transport connection is assigned both initiator and acceptor may either perform a new assignment or continue operation using one of the remaining network connections;
  - 3) if the transport connection becomes unassigned the acceptor may either perform a new assignment or wait (there is no risk of deadlock as either  $TI$  or  $I_L$  will be running); the initiator shall perform a new assignment (except in the closing state);
- b) connection establishment (see 6.5) and if appropriate connection refusal (see 6.6) together with the additional procedures:
  - 1) a connection is not considered established until the successful completion of a 3-way TPDU exchange. The sender of a CR TPDU shall respond to the corresponding CC TPDU by immediately sending a DT, ED, DR, or AK TPDU;
  - 2) as a result of duplication or transmission, 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, or is waiting for the T-CONNECT response from the TS-user, the receiving transport entity shall discard such a TPDU. Otherwise a CC TPDU shall be transmitted;
  - 3) 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 shall only acknowledge the duplicate CC TPDU according to the procedure in 12.2.2.2.b)1);

- 4) a CC TPDU may be received specifying a reference which is in the frozen state. The response to such a TPDU shall be a DR TPDU;
- 5) the retransmission procedures (see 12.2.1.2) are used for both the CR TPDU and CC TPDU.

NOTE – After receiving a CR TPDU, it is recommended that the transport entity enforce a time limit upon the transport service user so that late acceptance of the transport connection will not cause a delayed CC TPDU to be sent.

### 12.2.2.3 General procedures when operating over CLNS

The transport entity shall use the procedure of connection establishment (see 6.5) and if appropriate connection refusal (see 6.6) together with the additional procedures:

- 1) a connection is not considered established until the successful completion of a three-way TPDU exchange. The sender of a CR TPDU shall respond to the corresponding CC TPDU by immediately sending a DT, ED, DR or AK TPDU;
- 2) as a result of duplication or retransmission, 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 three-way TPDU exchange procedure, or is waiting for the T-CONNECT response from the TS-user, the receiving transport entity shall discard such a TPDU. Otherwise a CC TPDU shall be transmitted;
- 3) 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 shall only acknowledge the duplicate CC TPDU according to the procedure in 12.2.2.3.1);
- 4) a CC TPDU may be received specifying a reference which is in the frozen state. The response to such a TPDU shall be a DR TPDU;
- 5) the retransmission procedures (see 12.2.1.3) are used for both the CR TPDU and CC TPDU.

NOTE – After receiving a CR TPDU, it is recommended that the transport entity enforce a time limit upon the transport service user so that late acceptance of the transport connection will not cause a delayed CC TPDU to be sent.

## 12.2.3 Procedures for data transfer

### 12.2.3.1 Timers used in data transfer

#### 12.2.3.1.1 Timers used in data transfer when operating over CONS

The data transfer procedures use one additional timer:

- Window timer ( $W$ )

A transport entity maintains a timer interval to ensure that there is a bound on the maximum interval between window updates.

NOTE – A suitable upper bound value for  $W$  is such that  $W < I_R - E_{LR}$ . It is recommended that the value for  $W$  be sufficiently less than  $(I_R - E_{LR})$  such that the inactivity control procedure in 12.2.3.3 can be operated having regard to the possibility of TPDU loss.

#### 12.2.3.1.2 Timers used in data transfer when operating over CLNS

The data transfer procedures use one additional timer:

- Window timer ( $W$ )

A transport entity maintains a timer interval to ensure that there is a bound on the maximum interval between window updates.

NOTE – A suitable upper band value for  $W$  is such that  $W < I_R - E_{LR}$ . It is recommended that the value for  $W$  be sufficiently less than  $(I_R - E_{LR})$  such that the inactivity control procedure in 12.2.3.3 can be operated having regard to the possibility of TPDU loss.

### 12.2.3.2 General procedures for data transfer

The transport entities shall use the following procedures:

- a) inactivity control (see 6.21);
- b) expedited data (see 6.11);
- c) explicit flow control (see 6.16).

The sending transport entity shall use the following procedures in the following order:

- 1) segmenting (see 6.3);
- 2) DT TPDU numbering (see 6.10).

The receiving transport entity shall use the following procedures in the following order:

- DT TPDU numbering (see 6.10);
- resequencing (see 6.20);
- reassembling (see 6.3).

### 12.2.3.3 Inactivity control

If the interval of the inactivity timer *I* expires without receipt of some TPDU, the transport entity shall initiate the release procedures. To prevent expiration of the remote transport entity's inactivity timer 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 procedures (see 12.2.3.8) ensure that this requirement is met.

NOTE – It is likely that the release procedure initiated due to the expiration of the inactivity timer will fail, as such expiration indicates probable failure of the supporting network connection or of the remote transport entity.

### 12.2.3.4 Expedited data

#### 12.2.3.4.1 Expedited data when operating over CONS

The transport entities shall follow the network normal data variant of the expedited data transfer procedures (see 6.11.1), if the use of the transport expedited service option has been agreed during connection establishment.

The ED TPDU shall have a TPDU-NR which is allocated from a separate sequence space from that of the DT TPDUs.

A transport entity shall allocate the sequence number zero to the ED TPDU-NR of the first ED TPDU which it transmits for a transport connection. For subsequent ED TPDUs sent on the same transport connection, the transport entity shall allocate a sequence number one greater than the previous one.

Modulo  $2^7$  arithmetic shall be used when normal formats have been selected and modulo  $2^{31}$  arithmetic shall be used when extended formats have been selected.

The receiving transport entity shall transmit an EA TPDU with the same sequence number in its YR-EDTU-NR field. If this number is one greater than in the previously received in-sequence ED TPDU, the receiving transport entity shall transfer the data in the ED TPDU to the TS-user.

If a transport entity does not receive an EA TPDU in acknowledgement to an ED TPDU it shall follow the retransmission procedures (see Note and 12.2.1.2).

The sender of an ED TPDU shall not send any new DT TPDU created from a T-DATA request subsequent to the T-EXPEDITED DATA request, until it receives the EA TPDU.

NOTE – This procedure ensures that ED TPDUs are delivered to the TS-user in sequence and that the TS-user does not receive data corresponding to the same ED TPDU more than once. Also it guarantees the arrival of the ED TPDU before any data subsequently sent by the TS user.

#### 12.2.3.4.2 Expedited data when operating over CLNS

The transport entities shall follow the expedited data transfer procedures in 6.11.2, if the use of the transport expedited data service option has been agreed during connection establishment.

The ED TPDU shall have a TPDU-NR which is allocated from a separate sequence space from that of the DT TPDUs.



A transport entity shall allocate the sequence number zero to the ED TPDU-NR of the first ED TPDU which it transmits for a transport connection. For subsequent ED TPDU's sent on the same transport connection, the transport entity shall allocate a sequence number one greater than the previous one.

Modulo  $2^7$  arithmetic shall be used when normal formats have been selected and modulo  $2^{31}$  arithmetic shall be used when extended formats have been selected.

The receiving transport entity shall transmit an EA TPDU with the same sequence number in its YR-EDTU-NR field. If this number is one greater than in the previously received in-sequence ED TPDU, the receiving transport entity shall transfer the data in the ED TPDU to the TS-user.

If a transport entity does not receive an EA TPDU in acknowledgment to an ED TPDU it shall follow the retransmission procedures (see Note and 12.2.1.3).

The sender of an ED TPDU shall not send any new DT TPDU created from a T-DATA request subsequent to the T-EXPEDITED DATA request, until it receives the EA TPDU.

NOTE – This procedure ensures that ED TPDU's are delivered to the TS-user in sequence and that the TS-user does not receive data corresponding to the same ED TPDU more than once. Also it guarantees the arrival of the ED TPDU before any data subsequently sent by the TS user.

### 12.2.3.5 Resequencing

The receiving transport entity shall deliver all DT TPDU's to the TS-user in the order specified by the sequence number field.

DT TPDU's received out-of-sequence but within the transmit window shall not be delivered to the TS-user until all in-sequence TPDU's have been received. DT TPDU's received out-of-sequence and outside the transmit window shall be discarded but may result in transmission of an AK TPDU with up-to-date window information (see 12.2.3.8). If the selective acknowledgement option has been agreed to at connection establishment, DT TPDU's that have been selectively acknowledged shall be retained by the receiving transport entity until delivered to the TS-user. They shall be retained even if the selectively acknowledged DT TPDU's later fall outside the transmit window due to a subsequent credit reduction.

NOTE – It is recommended that the transport entity sending the AK TPDU maintains a bound on the number of times a DT TPDU is selectively acknowledged in order to reduce the processing at the transport entity receiving the AK TPDU.

Duplicate TPDU's can be detected because the sequence number matches that of previously received TPDU's. Sequence numbers shall 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 TPDU's shall 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 shall be discarded.

### 12.2.3.6 Explicit flow control

The transport entities shall send an initial credit (which may take the value 0) in the CDT field of the CR TPDU or CC TPDU. This credit represents the initial value of the upper window edge of the peer entity.

The transport entity which receives the CR TPDU or CC TPDU shall consider its lower window edge as zero and its upper window edge as the value in the CDT field in the received TPDU.

In order to authorize the transmission of DT TPDU's by its peer, a transport entity may transmit an AK TPDU at any time.

The sequence number of an AK TPDU shall not exceed the sequence number of the next expected DT TPDU, i.e. it shall not be greater than the highest sequence number of a received DT TPDU, plus one.

A transport entity may send a duplicate AK TPDU containing the same sequence number, CDT, and subsequence number field at any time.

A transport entity may increase or decrease the upper window edge at any time.

A transport entity which receives an AK TPDU shall consider the value of the YR-TU-NR field as its new lower window edge if it is greater than any previously received in a YR-TU-NR field, and the sum of YR-TU-NR and CDT as its new upper window edge subject to the procedures for sequencing AK TPDUs (see 12.2.3.8). A transport entity shall not transmit or retransmit a DT TPDU with a sequence number outside the transmit window.

### 12.2.3.7 Sequencing of received AK TPDUs

To allow a receiving transport entity to properly sequence a series of AK TPDUs that all contain the same sequence number and thereby use the correct CDT value, AK TPDUs may contain a subsequence parameter. For the purpose of determining the correct sequence of AK TPDUs, the absence of the subsequence parameter shall be equivalent to the value of the parameter set to zero.

An AK TPDU is defined to be in sequence if

- a) the sequence number is greater than any previously received AK TPDU, or
- b) the sequence number is equal to the highest in any previously received AK TPDU, and the subsequence parameter is greater than in any previously received AK TPDU having the same value for YR-TU-NR field, or
- c) the sequence number and subsequence parameter are both equal to the highest in any previously received AK TPDU and the credit field is greater than or equal to that in any previously received AK TPDU having the same YR-TU-NR field.

When the receiving transport entity recognizes an out-of-sequence AK TPDU it shall discard it.

### 12.2.3.8 Procedures for transmission of AK TPDUs

#### 12.2.3.8.1 Transmission of AK TPDUs

An in-sequence DT TPDU shall be acknowledged within time  $A_L$ , by the transmission of an AK TPDU whose YR-TU-NR parameter is set to at least the sequence number of the received DT TPDU plus one. If the selective acknowledgement option has been agreed to at connection establishment, out of sequence DT TPDUs may also be acknowledged within time  $A_L$ . The YR-TU-NR parameter shall be set to one greater than the highest sequence number of an in-sequence DT TPDU and the selective acknowledgement parameter will be appropriately set.

An AK TPDU shall be transmitted containing up-to-date window information if

- a) a DT TPDU is received whose sequence number is lower than the lower window edge, but greater than or equal to the lower window edge minus the maximum credit value ever given for this transport connection, or
- b) a DT TPDU is received whose sequence number is above the current upper window edge, but following credit reduction is within the upper window edge which has been granted and then withdrawn.

NOTE 1 – A simpler implementation may send an AK TPDU upon reception of any DT TPDU outside the transmit window.

NOTE 2 – The procedure a) is required so that loss of an AK TPDU is correctly recovered, i.e. when the sender of the DT TPDU retransmits it following non-receipt of an acknowledgement.

NOTE 3 – The procedure b) is required due to the possibility of loss of the AK TPDU indicating the upper window edge reduction, which could otherwise cause incorrect termination of the transport connection.

NOTE 4 – Wherever procedures a) and b) are invoked and selective acknowledgement option is being used, the selective acknowledgement parameters, if required, of the AK TPDU will be appropriately set.

A transport entity shall not allow an interval  $W$  to pass without the transmission of an AK TPDU. If the transport entity is not using the procedure following setting CDT to zero (see 12.2.3.8.3) or reduction of the upper window edge (see 12.2.3.8.4), and does not have to acknowledge receipt of any DT TPDU, then it shall achieve this by retransmission of the most recent AK TPDU, with up-to-date window information.

NOTE 5 – The use of the procedures defined in 12.2.3.8.3 and 12.2.3.8.4 is optional for any transport entity. The protocol operates correctly either with or without these procedures which are defined to enhance the efficiency of its operation.

#### **12.2.3.8.2 Sequence control for transmission of AK TPDU**

To allow the receiving transport entity to process AK TPDU in the correct sequence, as described in 12.2.3.7, the subsequence parameter shall be included following reduction of CDT. If the value of the subsequence number to be transmitted is zero, then the parameter should be omitted.

The value of the subsequence parameter, if used, shall be zero (either explicitly or by absence of the parameter) if the sequence number is greater than the parameter in previous AK TPDU, sent by the transport entity.

If the sequence number is the same as the previous AK TPDU sent and the CDT field is equal to or greater than the CDT field in the previous AK TPDU sent then the subsequence parameter, if used, shall be equal to that in the previously sent AK TPDU.

If the sequence number is the same as the previous AK TPDU sent and the CDT field is less than the value of the CDT field in the previous AK TPDU sent then the subsequence parameter, if used, shall be one greater than the value in the previous AK TPDU.

NOTE – If a transport entity never reduces credit, then it does not need to use the subsequence number.

#### **12.2.3.8.3 Retransmission of AK TPDU after CDT set to zero**

Due to the possibility of loss of AK TPDU, the upper window edge as perceived by the transport entity transmitting an AK TPDU may differ from that perceived by the intended recipient. To avoid the possibility of extra delay, the retransmission procedure (see 12.2.1.2 and 12.2.1.3) can be followed for an AK TPDU, if it opens the transmit window which has previously been closed by sending an AK TPDU with CDT field set to zero.

The retransmission procedure, if used, terminates and the procedure in 12.2.3.8.1 is used when

- a) an AK TPDU is received containing the flow control confirmation parameter, whose lower window edge and your subsequence fields are equal to the sequence number and subsequence number in the retained AK TPDU and whose credit field is not zero;
- b) an AK TPDU is transmitted with a sequence number higher than that in the retained AK TPDU, due to reception of a DT TPDU whose sequence number is equal to the lower window edge;
- c) *N* transmissions of the retained AK TPDU have taken place. In this case the transport entity shall continue to transmit the AK TPDU at an interval of *W*.

An AK TPDU which is subject to the retransmission procedure shall not contain the flow control confirmation parameter. If it is required to transmit this parameter concurrently, an additional AK TPDU shall be transmitted having the same values in the sequence, subsequence (if applicable) and credit fields.

#### **12.2.3.8.4 Retransmission procedures following reduction of the upper window edge**

This subclause specifies the procedure for retransmission of AK TPDU after a transport entity has reduced the upper window edge (see 12.2.3.6). This procedure is used until the lower window edge exceeds the highest value of the upper window edge ever transmitted (i.e. the value existing at the time of credit reduction, unless a higher value is retained from a previous credit reduction).

The retransmission procedure should be followed for any AK TPDU which increases the upper window edge, unless it is known that the remote transport entity has an open window. This is known if

- a flow control confirmation (FCC) parameter has been received corresponding to an AK TPDU transmitted following the most recent credit reduction, and;
- this FCC parameter conveys an upper window edge value (i.e. the sum of the lower window edge and credit fields) which is greater than the lower window edge of the transmitted AK TPDU.

This retransmission procedure for any particular AK TPDU shall terminate when

- a) an AK TPDU is received containing the flow control confirmation parameter, whose lower window edge and your subsequence fields are equal to the lower window edge and subsequence number in the retained AK TPDU; or
- b)  $N$  transmissions of the retained AK TPDU have taken place. In this case the transport entity shall continue to transmit the AK TPDU at an interval of  $W$ .

An AK TPDU which is subject to the retransmission procedure shall not contain the flow control confirmation parameter. If it is required to transmit this parameter concurrently, an additional AK TPDU shall be transmitted having the same values in the sequence, subsequence (if applicable) and credit fields.

NOTE – Retransmission of AK TPDU is normally not necessary, except following explicit closing of the window (i.e. transmission of an AK TPDU with CDT field set to zero). If data are available for transmission, the retransmission procedure for DT TPDU will ensure that an AK TPDU is received granting further credit where this is available; following credit reduction, this may no longer be so, because retransmission may be inhibited by the credit reduction. The rules described in this clause avoid extra delay.

The rules for determining whether to apply the retransmission procedure to an AK TPDU may be expressed alternatively as follows. Let

- LWE = Lower window edge  
UWE = Upper window edge  
KUWE = Lower bound on upper window edge held by remote transport entity.

The retransmission procedure is to be applied whenever:

$$(UWE > LWE) \text{ and } (KUWE = LWE)$$

i.e. when the window is opened and it is not known definitely that the remote transport entity is aware of this.

KUWE is maintained as follows:

When credit is reduced, KUWE is set to LWE. Subsequently, it is increased only upon receipt of a valid flow control confirmation (i.e. one which matches the retained lower window edge and subsequence). In this case KUWE is set to the implied upper window edge of the flow control confirmation, i.e. the sum of its lower window edge and your credit fields. By using this method, it can be ensured that KUWE is always less than or equal to the actual upper window edge used by the transmitter of DT TPDU.

### 12.2.3.9 Use of flow control confirmation parameter

An AK TPDU containing a flow control confirmation parameter may be transmitted at any time. The lower window edge, your subsequence and your credit fields shall be set to the same values as the corresponding fields in the most recently received in-sequence AK TPDU.

An AK TPDU containing a flow control confirmation parameter should be transmitted whenever

- a) a duplicate AK TPDU is received, with the value of YR-TU-NR, CDT, and subsequence fields equal to the most recently received in sequence AK TPDU, but not itself containing the flow control confirmation parameter;
- b) an AK TPDU is received which increases the upper window edge but not the lower window edge, and the upper window edge was formerly equal to the lower window edge; or
- c) an AK TPDU is received which increases the upper window edge but not the lower window edge, and the lower window edge is lower than the highest value of the upper window edge received and subsequently reduced (i.e. following credit reduction).

## 12.2.4 Procedures for release

### 12.2.4.1 Timers used for release

There are no timers used only for release.

### 12.2.4.2 General procedures for release

The transport entity shall use the explicit variant of normal release (see 6.7).

Although the retransmission procedure also apply to the DR TPDU in the release phase, the transport entity may however consider that the transport connection has been released if it would be necessary to open a new network connection in order to retransmit the DR TPDU.

## 13 Structure and encoding of TPDUs

### 13.1 Validity

Table 8 specifies those TPDUs which are valid for each class and the code for each TPDU.

TABLE 8/X.224

#### TPDU codes

|                                   | Validity within classes |     |    |   |   | See sub-clause | Code      |
|-----------------------------------|-------------------------|-----|----|---|---|----------------|-----------|
|                                   | 0                       | 1   | 2  | 3 | 4 |                |           |
| CR Connection request             | x                       | x   | x  | x | x | 13.3           | 1110 xxxx |
| CC Connection confirm             | x                       | x   | x  | x | x | 13.4           | 1101 xxxx |
| DR Disconnect request             | x                       | x   | x  | x | x | 13.5           | 1000 0000 |
| DC Disconnect confirm             |                         | x   | x  | x | x | 13.6           | 1100 0000 |
| DT Data                           | x                       | x   | x  | x | x | 13.7           | 1111 000y |
| ED Expedited data                 |                         | x   | NF | x | x | 13.8           | 0001 0000 |
| AK Data acknowledgement           |                         | NRC | NF | x | x | 13.9           | 0110 zzzz |
| EA Expedited data acknowledgement |                         | x   | NF | x | x | 13.10          | 0010 0000 |
| RJ Reject                         |                         | x   |    | x |   | 13.11          | 0101 zzzz |
| ER TPDU error                     | x                       | x   | x  | x | x | 13.12          | 0111 0000 |
| Not available (see the Note)      |                         |     |    |   |   | --             | 0000 0000 |
|                                   |                         |     |    |   |   | --             | 0011 0000 |
|                                   |                         |     |    |   |   | --             | 1001 xxxx |
|                                   |                         |     |    |   |   | --             | 1010 xxxx |

xxxx (bits 4 to 1) Used to signal the CDT (set to 0000 in classes 0 to 1).

zzzz (bits 4 to 1) Used to signal CDT in classes 2, 3, 4 set to 1111 in class 1.

y (bit 1) Used to signal ROA if the request acknowledgement procedure has been agreed at connection establishment (classes 1, 3, 4 only). This bit shall be set to 0 if the request acknowledgement procedure has not been agreed.

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

NRC Not available when the receipt confirmation option is selected.

NOTE – These codes are already in use in related protocols defined by standards organizations other than ITU-T and ISO/IEC.

## 13.2 Structure

All the transport protocol data units (TPDUs) shall contain an integral number of octets. The octets in a TPDU are numbered starting from 1 and increasing in the order they are put into an NDSU. The bits in an octet are numbered from 1 to 8, where bit 1 is the lowest order bit.

When consecutive octets are used to represent a binary number, the lower octet number has the most significant value.

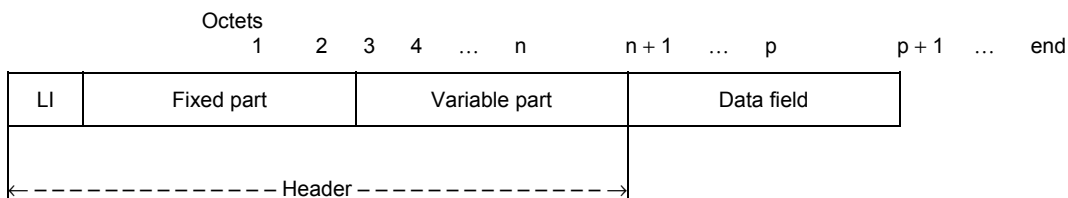
### NOTES

- 1 The numbering of bits within an octet is a convention local to this Recommendation.
- 2 The use of the terms “high order” and “low order” is common to this Recommendation and to adjacent layer standards.
- 3 The use of the above conventions does not affect the order of bit transmission on a serial communications link.
- 4 As described in 6.2.3, both transport entities respect these bit and octet ordering conventions, thus allowing communication to take place.
- 5 In this clause the encoding of TPDUs is represented in the following form:
  - a) octets are shown with the lowest numbered octet to the left; higher numbered octets being further to the right;
  - b) within an octet, bits are shown with bit 8 to the left and bit 1 to the right.

TPDUs shall contain, in the following order:

- a) the header, comprising
  - 1) the length indicator (LI) field;
  - 2) the fixed part;
  - 3) the variable part, if present;
- b) the data field, if present.

The structure is illustrated below:



### 13.2.1 Length indicator field

The field is contained in the first octet of the TPDUs. The length is indicated by a binary number, with a maximum value of 254 (1111 1110). The length indicated shall be the header length in octets including parameters, but excluding the length indicator field and user data, if any. The value 255 (1111 1111) is reserved for possible extensions.

If the length indicated exceeds or is equal to the size of the NS-user data which is present, this is a protocol error.

### 13.2.2 Fixed part

#### 13.2.2.1 General

The fixed part contains frequently occurring parameters including the code of the TPDU. The length and the structure of the fixed part are defined by the TPDU code and in certain cases by the protocol class and the formats in use (normal or extended). If any of the parameters of the fixed part have an invalid value, or if the fixed part cannot be contained within the header (as defined by LI), this is a protocol error.

NOTE – In general, the TPDU code defines the fixed part unambiguously. However, different variants may exist for the same TPDU code (see normal and extended formats).

### 13.2.2.2 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 Connection request
- 1101 xxxx Connection confirm
- 1111 000y Data
- 0101 xxxx Reject
- 0110 xxxx Data acknowledgement

where

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

y (bit 1) is used to signal ROA if the request acknowledgement has been agreed at connection establishment (class 1, 3, 4 only). This bit shall be set to 0 if the request acknowledgement procedure has not been agreed.

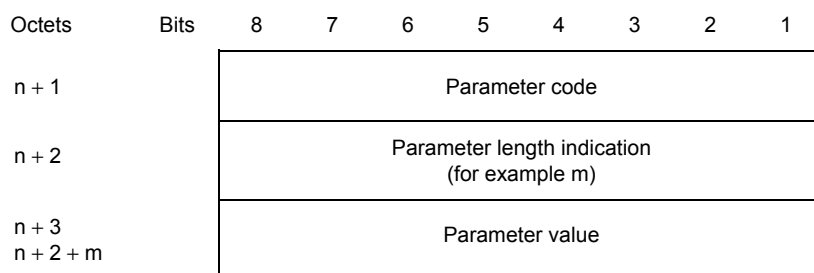
Only those codes defined in 13.1 are valid.

### 13.2.3 Variable part

The variable part is used to define less frequently used parameters. If the variable part is present, it shall contain one or more parameters.

NOTE 1 – The number of parameters that may be contained in the variable part is indicated by the length of the variable part which is LI minus the length of the fixed part.

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



The parameter code field is coded in binary.

NOTE 2 – Without extensions, it provides a maximum number of 255 different parameters. However, as noted below, bits 8 and 7 cannot take every possible value, so the practical maximum number of different parameters is less. Parameter code 1111 1111 is reserved for possible extensions of the parameter code.

The parameter length indication indicates the length, in octets, of the parameter value field.

NOTE 3 – 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 parameter 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 succeeding parameter, the maximum value of m decreases.

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

No parameter code uses bits 8 and 7 with the value 00.

The parameters defined in the variable part may be in any order. If any parameter is duplicated then the last value shall be used. A parameter not defined in this Recommendation shall be treated as a protocol error in any received TPDU except a CR TPDU; in a CR TPDU it shall be ignored. A called TSAP-ID parameter in a CC TPDU with a length indicator set to zero shall be treated as having the “nil selector value” (see 9.5.2 of CCITT Rec. X.650 | ISO/IEC 7498-3). If the responding transport entity selects a class for which a parameter of the CR TPDU is not defined, it may ignore this parameter, except if it is the class and option parameter, or the alternative protocol class parameter which shall always be interpreted. A parameter defined in this Recommendation but having an invalid value shall be

treated as a protocol error in any received TPDU except a CR TPDU. In a CR TPDU it shall be treated as a protocol error if it is either the class and option parameter or the alternative class parameter; bits 8 to 7, and bits 6 to 1, if not meaningful for the class proposed, of the additional options parameter shall be ignored; otherwise it shall be either ignored or treated as a protocol error.

### 13.2.3.1 Checksum parameter (class 4 only)

All TPDU types may contain a 16-bit checksum parameter in their variable part. This parameter shall be present in a CR TPDU and shall be present in all other TPDUs 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 6.17.

### 13.2.4 Data field

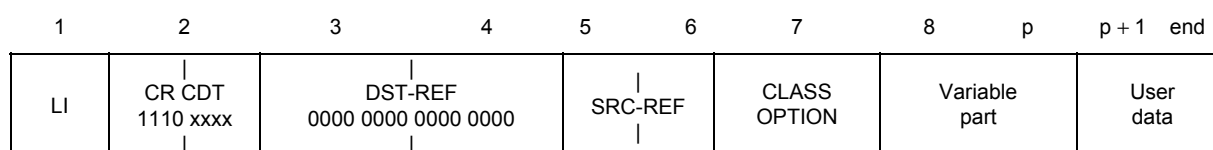
This field contains transparent user data. Restrictions on its size are noted for each TPDU.

## 13.3 Connection request (CR) TPDU

The length of the CR TPDU shall not exceed 128 octets.

### 13.3.1 Structure

The structure of the CR TPDU shall be as follows:



### 13.3.2 LI

See 13.2.1.

### 13.3.3 Fixed part (octets 2 to 7)

The structure of this part shall contain:

- a) CR: Connection request code: 1110. Bits 8 to 5 of octet 2;
- b) CDT: Initial credit allocation (set to 0000 in classes 0 and 1 when specified as preferred class). Bits 4 to 1 of octet 2;
- c) DST-REF: Set to zero;
- d) SRC-REF: Reference selected by the transport entity initiating the CR TPDU to identify the requested transport connection;
- e) CLASS and OPTION: Bits 8 to 5 of octet 7 define the preferred transport protocol class to be operated over the requested transport connection. When operating over CONS, this field shall take one of the following values:

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

When operating over CLNS, this field shall take the value 0100 to indicate class 4.



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

Bits 4 to 1 of octet 7 define options to be used on the requested transport connection as follows:

| Bit | Option   |
|-----|--|
| 4   | = 0 Always   |
| 3   | = 0 Always   |
| 2   | = 0 Use of normal formats in all classes<br>= 1 Use of extended formats in classes 2, 3, and 4 |
| 1   | = 0 Use of explicit flow control in class 2<br>= 1 No use of explicit flow control in class 2  |

Bits related to options particular to a class are not meaningful if that class is not proposed and may therefore take any value.

NOTES

- 1 The connection establishment procedure (see 6.5) does not permit a given CR TPDU to request use of transport expedited data transfer service (additional option parameter) and not 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.

**13.3.4 Variable part (octets 8 to p)**

The following parameters are permitted in the variable part:

a) *Transport Service Access Point Identifier (TSAP-ID)*

Parameter code: 1100 0001 for the identifier of the calling TSAP  
1100 0010 for the identifier of the called TSAP

Parameter length: not defined in this Recommendation

Parameter value: identifier of the calling or called TSAP respectively.

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

b) *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: 1100 0000

Parameter length: 1 octet

Parameter value:

0000 1101 8 192 octets (not allowed in class 0)

0000 1100 4 096 octets (not allowed in class 0)

0000 1011 2 048 octets

0000 1010 1 024 octets

0000 1001 512 octets

0000 1000 256 octets

0000 0111 128 octets

Default value: 0000 0111 (128 octets).

c) *Preferred maximum 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: 1111 0000

Parameter length: up to 4

Parameter value: a binary value. The binary value indicates the maximum TPDU size, expressed as a multiple of 128 octets [see 6.5.4 k) and 6.5.5 k)]. This binary value shall be greater than or equal to 1.

d) *Version number* (not used if class 0 is the preferred class)

Parameter code: 1100 0100

Parameter length: 1 octet

Parameter value field: 0000 0001

Default value: 0000 0001 (not used in class 0).

e) *Protection parameters* (not used is class 0 is the preferred class)

This parameter is user defined.

Parameter code: 1100 0101

Parameter length: user defined

Parameter value: user defined.

f) *Checksum* (used only if class 4 is the preferred class) (see 13.2.3.1)

This parameter shall 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.

g) *Additional option selection* (not used if class 0 is the preferred class)

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

Parameter code: 1100 0110

Parameter length: 1

Parameter value:

| Bit | Option   |
|-----|--|
| 6   | = 1 Use of request acknowledgement in class 1, 3, 4<br>= 0 Non-use of request acknowledgement in classes 1, 3, 4                 |
| 5   | = 1 Use of selective acknowledgement in class 4<br>= 0 Non-use of selective acknowledgement in class 4                           |
| 4   | = 1 Use of network expedited in class 1<br>= 0 Non-use of network expedited in class 1   |
| 3   | = 1 Use of receipt confirmation in class 1<br>= 0 Use of explicit AK variant in class 1  |
| 2   | = 0 16-bit checksum defined in 6.17 shall be used in class 4<br>= 1 16-bit checksum defined in 6.17 shall not be used in class 4 |
| 1   | = 1 Use of transport expedited data transfer service<br>= 0 Non-use of transport expedited data transfer service                 |

Default value: 0000 0001.

Bits 8 and 7 shall be set to zero when sending the TPDU and ignored upon receipt.

Bits related to options particular to a class are not meaningful if that class is not proposed and may therefore take any value.

- h) *Alternative protocol class(es)* (not used if class 0 is the preferred class or when operating over CLNS)
- Parameter code: 1100 0111
- Parameter length:  $n$
- Parameter value: Encoded as a sequence of single octets; each encoded as for octet 7 but with bits 4 to 1 set to zero (i.e. no alternative option selections permitted).
- i) *Acknowledgement time* (used only if class 4 is the preferred class)
- This parameter conveys the maximum acknowledgement time  $A_L$  to the remote transport entity. It is an indication only, and is not subject to negotiation (see 12.2.1.1.3).
- Parameter code: 1000 0101
- Parameter length: 2
- Parameter value:  $n$ , a binary number where  $n$  is the maximum acknowledgement time, expressed in milliseconds.
- j) *Throughput* (not used if class 0 is the preferred class)
- Parameter code: 1000 1001
- Parameter length: 12 or 24
- Parameter value:
- 1st 12 octets: maximum throughput, as follows:
- first 3 octets: target value, calling-called user direction;
  - second 3 octets: minimum acceptable, calling-called user direction;
  - third 3 octets: target value, called-calling user direction;
  - fourth 3 octets: minimum acceptable, called-calling user direction.
- 2nd 12 octets (optional): average throughput, as follows:
- fifth 3 octets: target value, calling-called user direction;
  - sixth 3 octets: minimum acceptable, calling-called user direction;
  - seventh 3 octets: target value, called-calling user direction;
  - eighth 3 octets: minimum acceptable, called-calling user direction.
- Where average throughput is omitted, it is considered to have the same value as the maximum throughput. Values are expressed in octets per second.
- k) *Residual error rate* (not used if class 0 is the preferred class)
- Parameter code: 1000 0110
- Parameter length: 3
- Parameter value:
- 1st octet: target value, power of 10;
  - 2nd octet: minimum acceptable, power of 10;
  - 3rd octet: TSDU size of interest, expressed as a power of 2.
- l) *Priority* (not used if class 0 is the preferred class)
- Parameter code: 1000 0111
- Parameter length: 2
- Parameter value: integer (0 is the highest priority)
- m) *Transit delay* (not used if class 0 is the preferred class)
- Parameter code: 1000 1000
- Parameter length: 8

Parameter value:

- first 2 octets: target value, calling-called user direction;
- second 2 octets: maximum acceptable, calling-called user direction;
- third 2 octets: target value, called-calling user direction;
- fourth 2 octets: maximum acceptable, called-calling user direction.

Values are expressed in milliseconds, and are based upon a TSDU size of 128 octets.

- n) *Reassignment time* (not used if class 0 or 2 is the preferred class; if class 4 is preferred and class 3 is an alternate, it may be used)

This parameter conveys the Time to Try Reassignment (TTR) which shall be used when following the procedure for reassignment after failure (see 6.12).

Parameter code: 1000 1011

Parameter length: 2

Parameter value:  $n$ , a binary number where  $n$  is the TTR value expressed in seconds.

- o) *Inactivity timer* (used only if class 4 is the preferred or selected class)

This parameter conveys the inactivity timer  $I_L$  to the remote transport entity. It is an indication only, and is not subject to negotiation (see 12.2.1.1.7).

Parameter code: 1111 0010

Parameter length: 4

Parameter value: a binary value. This binary value indicates the inactivity time expressed in milliseconds.

### 13.3.5 User data (octets $p + 1$ to the end)

No user data are permitted in class 0, and are optional in other classes. Where permitted, they shall not exceed 32 octets.

## 13.4 Connection Confirm (CC) TPDU

### 13.4.1 Structure

The structure of the CC TPDU shall be as follows:

|    |                         |                 |                 |                 |                  |   |   |              |         |     |
|----|-------------------------|-----------------|-----------------|-----------------|------------------|---|---|--------------|---------|-----|
| 1  | 2                       | 3               | 4               | 5               | 6                | 7 | 8 | $p$          | $p + 1$ | end |
| LI | <br>CC CDT<br>1101 xxxx | <br>DST-REF<br> | <br>SRC-REF<br> | CLASS<br>OPTION | Variable<br>part |   |   | User<br>data |         |     |

### 13.4.2 LI

See 13.2.1.

### 13.4.3 Fixed part (octets 2 to 7)

The fixed part shall contain

- a) CC: Connection confirm code: 1101. Bits 8 to 5 of octet 2;
- b) CDT: Initial credit allocation (set to 0000 in classes 0 and 1). Bits 4 to 1 of octet 2;
- c) DST-REF: Reference identifying the requested transport connection at the remote transport entity;
- d) SRC-REF: Reference selected by the transport entity initiating the CC TPDU to identify the confirmed transport connection;
- e) CLASS OPTION: Defines the selected transport protocol class and option to be operated over the accepted transport connection according to the negotiation rules specified in 6.5.

### 13.4.4 Variable part (octets 8 to p)

The parameters are defined in 13.3.4 and are subject to the constraints stated in 6.5 (connection establishment). Parameters ruled out by selection of an alternative class and option shall not be present.

### 13.4.5 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, they shall not exceed 32 octets. The user data are subject to the constraints of the negotiation rules (see 6.5).

## 13.5 Disconnect Request (DR) TPDU

### 13.5.1 Structure

The structure of the DR TPDU shall be as follows:

|    |                 |         |         |        |                  |   |   |              |       |     |
|----|-----------------|---------|---------|--------|------------------|---|---|--------------|-------|-----|
| 1  | 2               | 3       | 4       | 5      | 6                | 7 | 8 | p            | p + 1 | end |
| LI | DR<br>1000 0000 | DST-REF | SRC-REF | REASON | Variable<br>part |   |   | User<br>data |       |     |

### 13.5.2 LI

See 13.2.1.

### 13.5.3 Fixed part (octets 2 to 7)

The fixed part shall contain

- a) DR: Disconnect request code: 1000 0000;
- b) DST-REF: Reference identifying the transport connection at the remote transport entity;
- c) SRC-REF: Reference identifying the transport connection at the transport entity initiating the TPDU. Value zero when reference is unassigned;
- d) REASON: Defines the reason for disconnecting the transport connection. This field shall take one of the following values:

The following values may be used for classes 1 to 4:

- 1) 128 + 0 Normal disconnect initiated by session entity
- 2) 128 + 1 Remote transport entity congestion at connect request time
- 3) \*128 + 2 Connection negotiation failed [i.e. proposed class(es) not supported]
- 4) 128 + 3 Duplicate source reference detected for the same pair of NSAPs.
- 5) 128 + 4 Mismatched references
- 6) 128 + 5 Protocol error
- 7) 128 + 6 Not used
- 8) 128 + 7 Reference overflow
- 9) 128 + 8 Connection request refused on this network connection
- 10) 128 + 9 Not used
- 11) 128 + 10 Header or parameter length invalid.

The following values can be used for all classes:

- 12) 0 Reason not specified
- 13) 1 Congestion at TSAP

- 14) \*2            Session entity not attached to TSAP
- 15) \*3            Address unknown.

NOTE – Reasons marked with an asterisk (\*) may be reported to the TS-user as persistent, other reasons as transient.

**13.5.4 Variable part (octets 8 to p)**

The variable may contain

- a) a parameter allowing additional information related to the clearing of the connection;
  - Parameter code:    1110 0000
  - Parameter length: any value provided that the length of the DR TPDU does not exceed the maximum agreed TPDU size or 128 when the DR TPDU is used during the connection refusal procedure.
  - Parameter value:    additional information; the content of this field is user defined.
- b) checksum (see 13.2.3.1).

**13.5.5 User data (octets p + 1 to the end)**

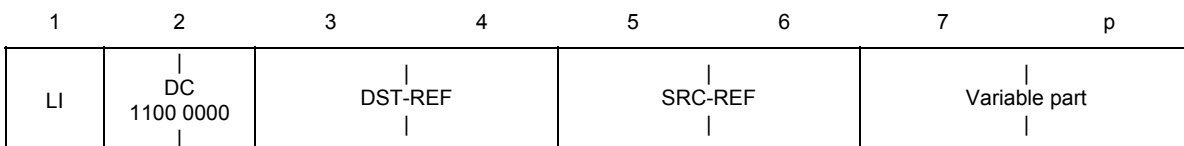
This field shall not exceed 64 octets and is used to carry TS-user data. The successful transfer of this data is not guaranteed by the transport protocol. When a DR TPDU is used in class 0 it shall not contain this field.

**13.6 Disconnect Confirm (DC) TPDU**

This TPDU shall not be used in class 0.

**13.6.1 Structure**

The structure of the DC TPDU shall be as follows:



**13.6.2 LI**

See 13.2.1.

**13.6.3 Fixed part (octets 2 to 6)**

The fixed part shall contain

- a) DC: Disconnect confirm code: 1100 0000;
- b) DST-REF: see 13.4.3;
- c) SRC-REF: see 13.4.3.

**13.6.4 Variable part**

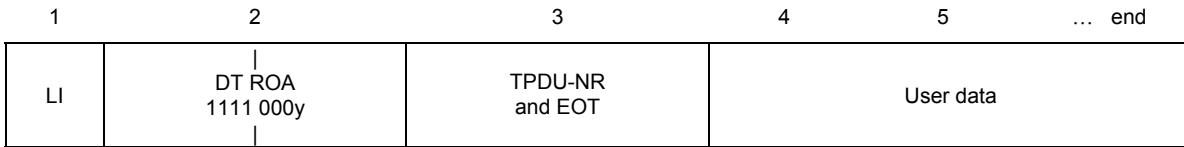
The variable part shall contain the checksum parameter if the condition defined in 13.2.3.1 applies.

## 13.7 Data (DT) TPDU

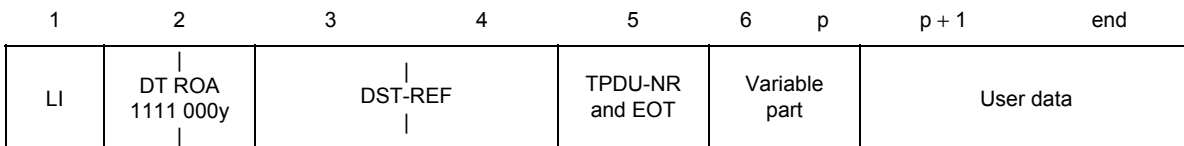
### 13.7.1 Structure

Depending on the class and the option the DT TPDU shall have one of the following structures:

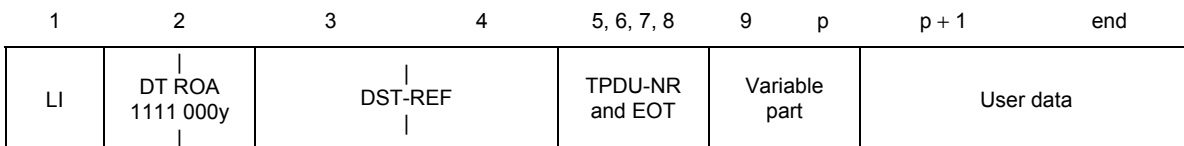
a) *Normal format for classes 0 and 1*



b) *Normal format for classes 2, 3 and 4*



c) *Extended format for use in classes 2, 3 and 4 when selected during connection establishment*



### 13.7.2 LI

See 13.2.1.

### 13.7.3 Fixed part

The fixed part shall contain

- a) DT: Data transfer code: bits 8 to 5 shall be set to 1111. Bits 4 to 2 shall be set to zero.
- b) ROA: Request of acknowledgement mark: If the request acknowledgement procedure has not been agreed during connection establishment, bit 1 shall be set to 0 in all DT TPDUs.  
When the request acknowledgement procedure has been agreed during connection establishment, bit 1 (ROA) is used to request acknowledgement in classes 1, 3, and 4. When set to one, ROA indicates that the sending transport entity requests an acknowledgement from the receiving transport entity. Otherwise ROA is set to zero.
- c) DST-REF: See 13.4.3;
- d) EOT: When set to ONE, it indicates that the current DT TPDU is the last data unit of a complete DT TPDU sequence (end of TSDU). EOT is bit 8 of octet 3 in class 0 and 1 and bit 8 of octet 5 for classes 2, 3 and 4;
- e) TPDU-NR: TPDU send sequence number (zero in class 0). May take any value in class 2 without explicit flow control. TPDU-NR is bits 7 to 1 of octet 3 for classes 0 and 1, bits 7 to 1 of octet 5 for normal formats in classes 2, 3, and 4 and bits 7 to 1 of octet 5 together with octets 6, 7 and 8 for extended format.

NOTE – Depending on the class, the fixed part of the DT TPDU uses the following octets:

- classes 0 and 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.

### 13.7.4 Variable part

The variable part shall contain the checksum parameter if the condition defined in 13.2.3.1 applies.

### 13.7.5 User data field

This field contains data of the TSDU being transmitted.

NOTE – 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 present, may further reduce the size of the user data field.

## 13.8 Expedited data (ED) TPDU

This ED TPDU shall not be used in class 0 or in class 2 when the no explicit flow control option is selected or when the expedited data transfer service has not been selected for the connection.

### 13.8.1 Structure

Depending on the format negotiated at connection establishment the ED TPDU shall have one of the following structures:

a) *Normal format (classes 1, 2, 3, 4)*

|    |                 |         |                       |                  |           |   |       |     |
|----|-----------------|---------|-----------------------|------------------|-----------|---|-------|-----|
| 1  | 2               | 3       | 4                     | 5                | 6         | p | p + 1 | end |
| LI | ED<br>0001 0000 | DST-REF | ED TPDU-NR<br>and EOT | Variable<br>part | User data |   |       |     |

b) *Extended format (for use in classes 2, 3 and 4 when selected during connection establishment)*

|    |                 |         |                       |                  |           |   |       |     |
|----|-----------------|---------|-----------------------|------------------|-----------|---|-------|-----|
| 1  | 2               | 3       | 4                     | 5, 6, 7, 8       | 9         | p | p + 1 | end |
| LI | ED<br>0001 0000 | DST-REF | ED TPDU-NR<br>and EOT | Variable<br>part | User data |   |       |     |

### 13.8.2 LI

See 13.2.1.

### 13.8.3 Fixed part

The fixed part shall contain

- a) ED: Expedited data code: 0001 0000;
- b) DST-REF: see 13.4.3;
- c) ED TPDU-NR: Expedited TPDU identification number. ED TPDU-NR is used in classes 1, 3 and 4 and may take any value in class 2. For normal formats bits 7 to 1 of octet 5 and for extended formats bits 7 to 1 of octet 5 together with octets 6, 7 and 8;
- d) EOT: End of TSDU always set to 1 (bit 8 of octet 5).

NOTE – Depending on the format the fixed part shall be either octets 2 to 5 or 2 to 8.

### 13.8.4 Variable part

The variable part shall contain the checksum parameter if the condition defined in 13.2.3.1 applies.

### 13.8.5 User data field

This field contains an expedited TSDU (1 to 16 octets).



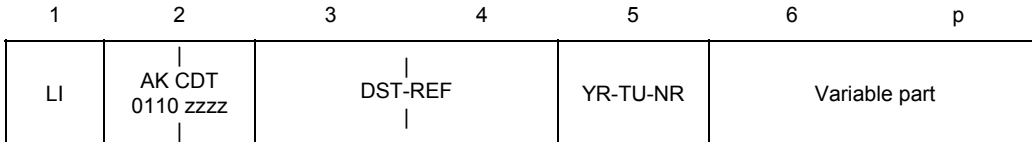
### 13.9 Data acknowledgement (AK) TPDU

This TPDU shall not be used in class 0 or in class 2 when the no explicit flow control option is selected, nor for class 1 when the network receipt confirmation option is selected.

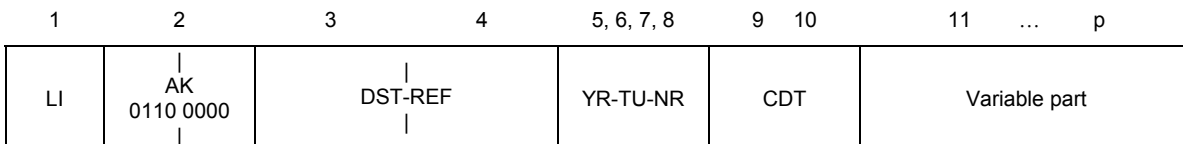
#### 13.9.1 Structure

Depending on the class and option agreed the AK TPDU shall have one of the following structures:

- a) *Normal format (classes 1, 2, 3, 4)*



- b) *Extended format (for use in classes 2, 3 and 4 when selected during connection establishment)*



#### 13.9.2 LI

See 13.2.1.

#### 13.9.3 Fixed part

The fixed part shall contain (in octets 2 to 5 when normal format is used or in octets 2 to 10) the following parameters:

- a) AK: Acknowledgement code: 0110;
- b) CDT: Credit value (set to 1111 in class 1). CDT is bits 4 to 1 of octet 2 for normal formats and octets 9 and 10 for extended formats;
- c) DST-REF: See 13.4.3;
- d) YR-TU-NR: Sequence number indicating the next expected DT TPDU number. For normal formats, bits 7 to 1 of octet 5; bit 8 of octet 5 is not significant and shall take the value 0. For extended formats, bits 7 to 1 of octet 5 together with octets 6, 7 and 8; bit 8 of octet 5 is not significant and shall take the value 0.

#### 13.9.4 Variable part

The variable part contains the following parameters:

- a) Checksum if the condition in 13.2.3.1 applies;
- b) Subsequence number when optionally used under the conditions defined in class 4. This parameter is used to ensure that AK TPDU's are processed in the correct sequence. If it is absent, this is equivalent to transmitting the parameter with a value of zero.  
 Parameter code: 1000 1010  
 Parameter length: 2  
 Parameter value: 16-bit subsequence number;
- c) Flow control confirmation when optionally used under the conditions defined in class 4. 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 12.2.3.9).

Parameter code: 1000 1100

Parameter length: 8

Parameter value: defined as follows:

- 1) Lower window edge (32 bits). Bit 8 of octet 1 of the parameter value field is set to zero, the remainder contains the YR-TU-NR value of the received AK TPDU. When normal format has been selected, only the least significant seven bits (bits 1 to 7 of octet 4 of the parameter value field) of this field are significant.
  - 2) Your subsequence (16 bits). Contains the value of the subsequence parameter of the received AK TPDU, or zero if this parameter was not present.
  - 3) Your credit (16 bits). Contains the value of the CDT field of the received AK TPDU. When normal format has been selected, only the least significant four bits (bits 1 to 4 of octet 2 of the Your Credit field) of this field are significant.
- d) Selective acknowledgement parameters when optionally used, under conditions defined in class 4, to acknowledge out of sequence DT TPDU received by the entity transmitting the AK TPDU. All consecutive DT TPDU received shall be acknowledged by a single block. Different groups of DT TPDU that are consecutive within group but not across groups shall be acknowledged using separate blocks [e.g., if DT TPDU numbers 3, 4, 5, 7, 8, 12, 13, 14, 15 and 17 are received with 3 the first out-of-sequence DT TPDU, then 3, 4, 5 form one group, 7, 8 another and 12, 13, 14, 15 a third, and 17 a fourth. The corresponding blocks would be (3, 5), (7, 8), (12, 15) and (17, 17)].

Parameter code: 1000 1111

Parameter length: 2n (normal format) or 8n (extended format) octets where n is the number of distinct blocks being selectively acknowledged. This length is constrained by the maximum header size of 254 octets.

Parameter value: In the normal format, the first octet of a pair of two octets shall represent the lower edge and the second octet the upper edge of each block. Bit 8 of each octet is set to zero, the remainder represents the sequence number of the edge.

In the extended format, the first four octets of a set of eight octets represent the lower edge and the subsequent four octets represent the upper edge. For each edge of four octets, Bit 8 of the first octet is not significant and is set to 0; bits 7 to 1 of the first octet together with the second, third and fourth octets represent the sequence number.

Whether normal or extended formats are used, each set of two or eight octets may be repeated as many times as there are blocks to be acknowledged.

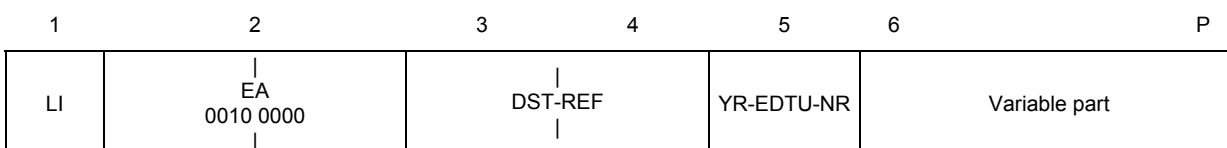
### 13.10 Expedited data acknowledgement (EA) TPDU

The EA TPDU shall not be used for class 0, or for class 2 when the “no explicit flow control” option is selected, or when the expedited data transfer service has not been selected for the connection.

#### 13.10.1 Structure

Depending on the option (normal or extended format) the TPDU structure shall be:

- a) Normal format (classes 1, 2, 3, 4)



b) *Extended format (for use in classes 2, 3 and 4 if selected during connection establishment)*

|    |                 |         |   |            |               |   |
|----|-----------------|---------|---|------------|---------------|---|
| 1  | 2               | 3       | 4 | 5, 6, 7, 8 | 9             | P |
| LI | EA<br>0010 0000 | DST-REF |   | YR-EDTU-NR | Variable part |   |

### 13.10.2 LI

See 13.2.1.

### 13.10.3 Fixed part

The fixed part shall contain (in octets 2 to 5 when normal format is used or in octets 2 to 8) the following parameters:

- a) EA: Expedited acknowledgement code: 0010 0000;
- b) DST-REF: See 13.4.3;
- c) YR-EDTU-NR: Identification of the ED TPDU being acknowledged. May take any value in class 2. For normal formats bits 7 to 1 of octet 5; bit 8 of octet 5 is not significant and shall take the value 0. For extended formats, bits 7 to 1 of octet 5 together with octets 6, 7 and 8; bit 8 of octet 5 is not significant and shall take the value 0.

### 13.10.4 Variable part

The variable part may contain the checksum parameter (see 13.2.3.1).

## 13.11 Reject (RJ) TPDU

The RJ TPDU shall not be used in classes 0, 2, and 4.

### 13.11.1 Structure

The RJ TPDU shall have one of the following formats:

- a) *Normal format (classes 1 and 3)*

|    |                     |         |   |          |
|----|---------------------|---------|---|----------|
| 1  | 2                   | 3       | 4 | 5        |
| LI | RJ CDT<br>0101 zzzz | DST-REF |   | YR-TU-NR |

- b) *Extended format (for use in class 3 if selected during connection establishment)*

|    |                 |         |   |            |     |    |
|----|-----------------|---------|---|------------|-----|----|
| 1  | 2               | 3       | 4 | 5, 6, 7, 8 | 9   | 10 |
| LI | RJ<br>0101 0000 | DST-REF |   | YR-TU-NR   | CDT |    |

### 13.11.2 LI

See 13.2.1.

### 13.11.3 Fixed part

The fixed part shall contain (in octets 2 to 5 when normal format is used or in octets 2 to 10) the following parameters:

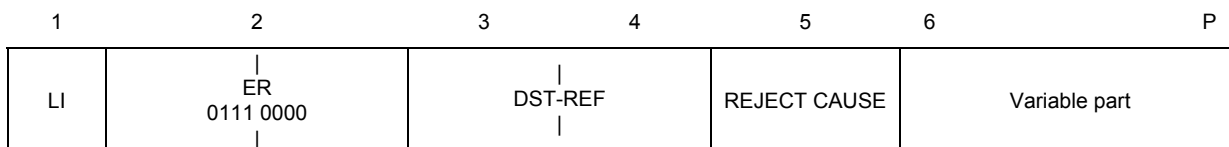
- a) RJ: Reject code: 0101. Bits 8 to 5 of octet 2;
- b) CDT: Credit value (set to 1111 in class 1). For normal formats bits 4 to 1 of octet 2 and for extended formats octets 9 and 10;
- c) DST-REF: See 13.4.3;
- d) YR-TU-NR: Sequence number indicating the next expected TPDU from which retransmission should occur. For normal formats, bits 7 to 1 of octet 5, bit 8 of octet 5 is not significant and shall take the value 0. For extended formats, bits 7 to 1 of octet 5 together with octets 6, 7 and 8; bit 8 of octet 5 is not significant and shall take the value 0.

### 13.11.4 Variable part

There is no variable part for this TPDU type.

## 13.12 TPDU error (ER) TPDU

### 13.12.1 Structure



### 13.12.2 LI

See 13.2.1.

### 13.12.3 Fixed part

The fixed part shall contain the following parameters:

- a) ER: TPDU Error code: 0111 0000;
- b) DST-REF: See 13.4.3;
- c) REJECT CAUSE:
  - 0000 0000 Reason not specified
  - 0000 0001 Invalid parameter code
  - 0000 0010 Invalid TPDU type
  - 0000 0011 Invalid parameter value.

### 13.12.4 Variable part

The variable part may contain the following parameters:

- a) *Invalid TPDU*
  - Parameter code: 1100 0001
  - Parameter length: number of octets of the value field
  - Parameter value: contains the bit pattern of the rejected TPDU header up to and including the octet which caused the rejection. This parameter is mandatory in class 0.
- b) *Checksum*
  - This parameter shall be present if the condition in 13.2.3.1 applies.

## 14 Conformance

**14.1** A system claiming to implement the procedures specified in this Recommendation shall comply with the requirements in 14.2-14.4.

**14.2** The system shall implement class 0. This implies operation over CONS.

**14.3** If the system implements class 3 or class 4, it shall also implement class 2.

**14.4** For each class which the system claims to implement, the system shall be capable of

- a) initiating CR TPDU or responding to CR TPDU with CC TPDU or both. Where a CR TPDU proposing class 2, 3 or 4 is initiated, class 0 shall be explicitly indicated as an alternative class except if there is already one (or several) transport connection(s) assigned to the network connection (multiplexing being possible);
- b) responding to any other TPDU and operating network service in accordance with the procedures for the class;
- c) operating all the procedures for the class listed as mandatory in Table 9;
- d) operating those procedures for the class listed as optional in Table 9 for which conformance is claimed;
- e) handling all TPDU of lengths up to the lesser value of
  - 1) the maximum length for the class if the preferred maximum TPDU size parameter is not implemented [see 13.3.4 b)];
  - 2) the maximum for which conformance is claimed (see Note 2);

### NOTES

1 The procedures for classes 0 to 4 are specified in clauses 8 to 12 respectively. The procedures refer to the elements of procedures specified in clause 6.

2 The requirement in 14.4 e) indicates that TPDU size of 128 octets is always implemented.

TABLE 9/X.224

### Provision of options

| Procedures   | Class 0        | Class 1        | Class 2        | Class 3        | Class 4        |
|--|----------------|----------------|----------------|----------------|----------------|
| TPDU with checksum                                   | Not applicable | Not applicable | Not applicable | Not applicable | Mandatory      |
| TPDU without checksum                                | Mandatory      | Mandatory      | Mandatory      | Mandatory      | Optional       |
| Expedited data transfer                              | Not applicable | Mandatory      | Mandatory      | Mandatory      | Mandatory      |
| No expedited data transfer                           | Mandatory      | Mandatory      | Mandatory      | Mandatory      | Mandatory      |
| Flow control in class 2                              | Not applicable | Not applicable | Mandatory      | Not applicable | Not applicable |
| No flow control in class 2                           | Not applicable | Not applicable | Optional       | Not applicable | Not applicable |
| Normal formats                                       | Mandatory      | Mandatory      | Mandatory      | Mandatory      | Mandatory      |
| Extended formats                                     | Not applicable | Not applicable | Optional       | Optional       | Optional       |
| Use of receipt confirmation in class 1               | Not applicable | Optional       | Not applicable | Not applicable | Not applicable |
| No use of receipt confirmation in class 1            | Not applicable | Mandatory      | Not applicable | Not applicable | Not applicable |
| Use of network expedited in class 1                  | Not applicable | Optional       | Not applicable | Not applicable | Not applicable |
| No use of network expedited in class 1               | Not applicable | Mandatory      | Not applicable | Not applicable | Not applicable |
| Use of selective acknowledgement in class 4          | Not applicable | Not applicable | Not applicable | Not applicable | Optional       |
| Use of request of acknowledgement in classes 1, 3, 4 | Not applicable | Optional       | Not applicable | Optional       | Optional       |

#### 14.5 Claims of conformance shall state

- a) which class or classes of protocol are implemented;
- b) whether class 4 can be operated over the connectionless-mode network service;
- c) whether the system is capable of initiating or responding to CR TPDU's or both;
- d) which of the procedures listed as optional in Table 9 are implemented;
- e) for each class, the maximum size of TPDU implemented [see 13.3.4 b) and 13.3.4 c)]. If the preferred maximum TPDU size parameter is not implemented the value shall be chosen from the following list and all values in the list which are less than this maximum shall be implemented:

128, 256, 512, 1 024, 2 048, 4 096 or 8 192 octets.

If the preferred maximum TPDU size parameter is implemented, any maximum size of TPDU that is a multiple of 128 octets is allowed. All values, except 0, that are a multiple of 128 octets, less than the maximum claimed shall be implemented.

**14.6** The supplier of a protocol implementation which is claimed to conform to this Recommendation shall complete a copy of the PICS proforma provided in Annex D and shall provide the information necessary to identify both the supplier and the implementation.

## **Annex A** (normative)

### **State tables**

(This annex forms an integral part of this Recommendation)

#### **A.1 General**

This annex provides a more precise description of the protocol.

The state tables also define the mapping between service and protocol events that TS-users can expect.

This annex describes the transport protocol in terms of state tables. The state tables show the state of a transport connection, the events that occur in the protocol, the actions taken and the resultant state.

The state tables describe only the operation of a single transport connection. They do not necessarily describe all possible combinations of sequences of events at transport and network service boundary, nor do they describe the exact mapping between TPDU's and NSDU's.

#### **A.2 Conventions**

**A.2.1** Incoming events are represented in the state tables by their abbreviated name, as defined in Table A.1.

**A.2.2** States are represented in the tables by their abbreviated name, as defined in Table A.2.

**A.2.3** The intersection of each state and event which is invalid is left blank. The action to be taken in this case shall be one of the following:

- a) for an event related to the transport service (i.e. coming from the TS-user), take no action;
- b) for an event related to a received TPDU, follow the procedure for treatment of protocol errors (see 6.22) if the state of the supporting network connection makes it possible;
- c) for an event falling into neither of the above categories (including those which are impossible by the definition of the behaviour of the transport entity or NS-provider), take no action.

**A.2.4** At each intersection of state and event which is valid the state tables specify an action which may include one of the following:

- a) one action constituted of a list of any number of outgoing events (none, one, or more) given by their abbreviated name defined in Table A.3 followed by the abbreviated name of the resultant state (see Table A.2);
- b) conditional actions separated by a semi-colon (;). Each conditional action contains a predicate followed by a colon (:) and by an action as defined in a). The predicates are boolean expressions given by their abbreviated name and defined in the clauses related to the state tables of each class. Only the action corresponding to the predicate which is true shall be taken.

**A.2.5** The state tables also include

- a) informal comments giving explanatory materials;
- b) references to notes using the following notation: (note number);
- c) references to other actions defined in separate tables using the following notation: [action number].

### A.3 Tables

Table A.1 specifies that names and abbreviated names of the incoming events, classified as TS-user events, NS-provider events or TPDU events.

TABLE A.1/X.224

#### Incoming events

| Abbreviated name | Category    | Name                                |
|------------------|-------------|-------------------------------------|
| TCONreq          | TS-user     | T-CONNECT Request primitive         |
| TCONresp         | TS-user     | T-CONNECT Response primitive        |
| TDTreq           | TS-user     | T-DATA Request primitive            |
| TEXreq           | TS-user     | T-EXPEDITED DATA Request primitive  |
| TDISreq          | TS-user     | T-DISCONNECT Request primitive      |
| NDISind          | NS-provider | N-DISCONNECT Indication primitive   |
| NCONconf         | NS-provider | N-CONNECT Confirm primitive         |
| NRSTind          | NS-provider | N-RESET Indication primitive        |
| CR               | TPDU        | Connection Request TPDU             |
| CC               | TPDU        | Connection Confirm TPDU             |
| DR               | TPDU        | Disconnect Request TPDU             |
| DC               | TPDU        | Disconnect Confirm TPDU             |
| AK               | TPDU        | Data Acknowledgement TPDU           |
| EA               | TPDU        | Expedited Data Acknowledgement TPDU |
| DT               | TPDU        | Data TPDU                           |
| ED               | TPDU        | Expedited Data TPDU                 |
| ER               | TPDU        | TPDU Error TPDU                     |
| RJ               | TPDU        | Reject TPDU                         |

Table A.2 specifies the names and abbreviated names of the states.

TABLE A.2/X.224

**States**

| Abbreviated name | Name  |
|------------------|---|
| WFNC             | Wait for network connection   |
| WFCC             | Wait for the CC TPDU  |
| WBCL             | Wait before releasing (wait for CC TPDU before sending the TPDU DR) |
| OPEN             | Transport connection is open  |
| CLOSING          | Release in progress   |
| WFTRESP          | Wait for T-CONNECT response   |
| CLOSED           | Transport connection is closed                                      |
| WFNC-R           | Wait for network connection and reassignment in progress            |
| WFCC-R           | Wait for CC TPDU and reassignment in progress                       |
| WBCL-R           | Wait before releasing and reassignment in progress                  |
| OPEN-R           | Open and reassignment in progress                                   |
| OPEN-WR          | Open and wait for reassignment                                      |
| CLOSING-R        | Release in progress and reassignment in progress                    |
| CLOSING-WR       | Release in progress and wait for reassignment                       |
| WFTRESP-WR       | Wait for T-CONNECT response and wait for reassignment               |
| WBCL-WR          | Wait before releasing and wait for reassignment                     |
| WBOC             | Wait before open complete (CC is unacknowledged)                    |
| WBOC-WR          | Wait before open complete and wait for reassignment                 |
| CLOSING BOC      | Wait before open complete and release in progress                   |
| CLOSING BOC-WR   | Idem and wait for reassignment                                      |
| AKWAIT           | Waiting for acknowledgement of CC TPDU                              |
| REFWAIT          | Waiting for frozen reference time                                   |



Table A.3 specifies the names and abbreviated names of the outgoing events classified as TS-provider events, NS-user events or TPDU events.

TABLE A.3/X.224

**Outgoing event**

| Abbreviated name | Category    | Name                                  |
|------------------|-------------|---------------------------------------|
| TCONind          | TS-provider | T-CONNECT Indication primitive        |
| TCONconf         | TS-provider | T-CONNECT Confirm primitive           |
| TDTind           | TS-provider | T-DATA Indication primitive           |
| TEXind           | TS-provider | T-EXPEDITED DATA Indication primitive |
| TDISind          | TS-provider | T-DISCONNECT Indication primitive     |
| NDISreq          | NS-user     | N-DISCONNECT Request primitive        |
| NRSTresp         | NS-user     | N-RESET Response primitive            |
| NCONreq          | NS-user     | N-CONNECT Request primitive           |
| CR               | TPDU        | Connection Request TPDU               |
| CC               | TPDU        | Connection Confirm TPDU               |
| DR               | TPDU        | Disconnect Request TPDU               |
| DC               | TPDU        | Disconnect Confirm TPDU               |
| AK               | TPDU        | Data Acknowledgement TPDU             |
| EA               | TPDU        | Expedited Data Acknowledgement TPDU   |
| DT               | TPDU        | Data TPDU                             |
| ED               | TPDU        | Expedited Data TPDU                   |
| ER               | TPDU        | TPDU Error TPDU                       |
| RJ               | TPDU        | Reject TPDU                           |

**A.4 State tables for classes 0 and 2**

This clause provides a more precise description of a transport entity for a transport connection of class 0 or class 2.

The description uses predicates defined in Table A.4, and specific actions defined in Table A.5.

The description does not include a complete specification of the data transfer procedures but makes reference to the specification of the classes (see clause 8 and 10). Table A.6 gives the state automata for classes 0 and 2.

TABLE A.4/X.224

**Predicates for classes 0 and 2**

| Name | Description                                       |
|------|---|
| P0   | T-CONNECT request unacceptable                    |
| P1   | Unacceptable CR TPDU                              |
| P2   | No network connection available                   |
| P3   | Network connection available and open             |
| P4   | Network connection available and open in progress |
| P5   | Class in class 0 (class selected in CC)           |
| P6   | Unacceptable CC                                   |
| P7   | Class is class 2                                  |
| P8   | Acceptable CC                                     |
| P9   | Class 4 CR  |
| P10  | Local choice                                      |

TABLE A.5/X.224

**Specific actions for classes 0 and 2**

| Name | Description   |
|------|---|
| [1]  | If the network connection is not used by another transport connection assigned to it, it may be disconnected. (See 6.1.1.3 Note 3)  |
| [2]  | See 6.22 (receipt of an ER TPDU)  |
| [3]  | See data transfer procedures of the class   |
| [4]  | See expedited data transfer procedure of the class  |
| [5]  | An N-RESET response has to be issued once for the network connection if the network connection has not been released. In class 0, an N-DISCONNECT request has to be issued. |
| [6]  | The DC TPDU contains a SRC-REF field set to zero and a DST-REF field set to the SRC-REF of the DR TPDU received.  |

TABLE A.6/X.224

State table for classes 0 and 2

| State Event | WFNC                          | WFCC  | WBCL (Class 2 only)                       | OPEN                                  | CLOSING (Class 2 only) | WFTRESP                | CLOSED   |
|-------------|-------------------------------|---|---|---------------------------------------|------------------------|------------------------|--|
| TCONreq     |                               |   |   |                                       |                        |                        | P0: TDISind CLOSED;<br>P2: NCONreq WFNC;<br>P3: CR WFCC;<br>P4: WFNC |
| TCONresp    |                               |   |   |                                       |                        | CC OPEN                |  |
| TDTrreq     |                               |   |   | [3] OPEN                              |                        |                        |  |
| TEXreq      | DOES NOT EXIST IN CLASS 0     |   |   |                                       |                        |                        |  |
|             |                               |   |   | [4] OPEN                              |                        |                        |  |
| TDISreq     | [1] CLOSED                    | not P7: NDISreq CLOSED;<br>P7: WBCL   |   | P5: NDISreq CLOSED;<br>P7: DR CLOSING |                        | DR CLOSED              |  |
| NCONconf    | CR WFCC                       |   |   |                                       |                        |                        |  |
| NRSTind     |                               | TDISind [1] [5] CLOSED  | [1] [5] CLOSED                            | TDISind [1] [5] CLOSED                | [1] [5] CLOSED         | TDISind [1] [5] CLOSED |  |
| NDISind     | TDISind CLOSED                | TDISind CLOSED  | CLOSED                                    | TDISind CLOSED                        | CLOSED                 | TDISind CLOSED         |  |
| CR          |                               |   |   | P9: OPEN                              | P9: CLOSING            | P9: WFTRESP            | P1: DR (1) CLOSED;<br>not P1: TCONind WFTRESP                        |
| DR          |                               | TDISind   |   | P5: (2)                               |                        | P10: DC [6] (5)        | CLOSED (4)   |
|             |                               | [1] CLOSED  | [1] CLOSED                                | P7: DC TDISind CLOSED                 | [1] CLOSED             | TDISind CLOSED         | DC CLOSED  |
| DC          | DOES NOT EXIST IN CLASS 0 (2) |   |   |                                       |                        |                        |  |
|             |                               |   |   |                                       | P7: [1] CLOSED         |                        |  |
| CC          |                               | P8: TCONconf OPEN;<br>P6 and P5: TDISind NDISreq CLOSED;<br>P6 and P7: TDISind DR CLOSING | P5: (3) NDISreq CLOSED;<br>P7: DR CLOSING |                                       | CLOSING                |                        | DR CLOSED  |
| AK          | DOES NOT EXIST IN CLASS 0 (2) |   |   |                                       |                        |                        |  |
|             |                               |   |   | [3] OPEN                              | CLOSING                |                        |  |
| EA          | DOES NOT EXIST IN CLASS 0 (2) |   |   |                                       |                        |                        |  |
|             |                               |   |   | [4] OPEN                              | CLOSING                |                        |  |

TABLE A.6/X.224 (end)

**State table for classes 0 and 2**

| State<br>Event   | WFNC                          | WFCC                  | WBCL<br>(Class 2<br>only) | OPEN     | CLOSING<br>(Class 2<br>only) | WFTRESP | CLOSED |
|--|-------------------------------|-----------------------|---------------------------|----------|------------------------------|---------|--------|
| ED   | DOES NOT EXIST IN CLASS 0 (2) |                       |                           |          |                              |         | CLOSED |
|  |                               |                       |                           | [4] OPEN | CLOSING                      |         |        |
| DT   |                               |                       |                           | [3] OPEN | CLOSING                      |         | CLOSED |
| ER   |                               | TDISind<br>[1] CLOSED | [1] CLOSED                | (2)      | (2)                          |         | CLOSED |
| (1) An ER TPDU should be sent in certain cases (see 6.6).<br>(2) If received it should be processed as a protocol error (see 6.22).<br>(3) A CR with class 2 has been sent and a CC class 0 is received.<br>(4) If DC is not available (i.e. class 0 only implemented) or SRC-REF is zero.<br>(5) This happens only when the preferred class of the CR TPDU received is class 4. |                               |                       |                           |          |                              |         |        |

**A.5 State tables for classes 1 and 3**

This clause provides a more precise description of a transport entity for a transport connection of class 1 or 3.

The description uses the predicates defined in Table A.7.

Specific actions are defined in Table A.8 and specific additional notes are given in Table A.9.

The description does not include a complete specification of the data transfer but makes reference to the specification of the classes (see clauses 9 and 11). Table A.10 gives the state automata for classes 1 and 3.

TABLE A.7/X.224

**Predicates for classes 1 and 3**

| Name | Description  |
|------|--|
| P0   | T-CONNECT Request unacceptable   |
| P1   | No available network connection can be used for assignment or reassignment                                     |
| P2   | A network connection can be used for assignment or reassignment; the network connection opening is in progress |
| P3   | A network connection can be used for assignment or reassignment; the network connection is open                |
| P4   | TTR timer has previously run out   |
| P5   | Local choice   |
| P6   | Initiator of the transport connection  |
| P7   | Unacceptable CR TPDU   |
| P8   | TWR is running   |
| P9   | Class 4 CR   |
| P10  | Class selected in CC is class 0 or 2   |

TABLE A.8/X.224

**Specific actions for classes 1 and 3**

| Name   | Description   |
|--|---|
| [1]  | The network connection can be disconnected if not used by any transport connection assigned to it   |
| [2]  | Retransmit expedited data which are unacknowledged or which have been stored when waiting for reassignment (if any). If an RJ TPDU has been received, enable also data TPDU transmission (if any). If an ED was received, handle according to procedures for class if not a duplicate |
| [3]  | Network connection can be disconnected if not used by any transport connection and was locally opened   |
| [4]  | Start TWR timer if not already running. Disable sending DT TPDUs until an RJ TPDU is received (see Note 3)  |
| [5]  | Stop TWR timer  |
| [6]  | Issue an N-RESET response if not already done   |
| [7]  | See data transfer procedure for the class   |
| [8]  | Start TTR timer if not already running. The sending credit is also set to zero in order not to send DT TPDUs until an RJ TPDU is received.  |
| [9]  | Stop TTR timer if running or remove information that TTR timer has run out (see Notes 1 and 2)  |
| [10]   | Store information that TTR timer has run out (see Note 1)   |
| [11]   | Store request   |
| [12]   | See state table appropriate to the class selected in the CC TPDU  |
| [13]   | Close the network connection to which the transport connection is currently assigned, apply to all transport connections assigned to this network connection the procedure for processing NDISind and then process the TPDU reassignment  |
| [14]   | The DC TPDU contains a SRC-REF field set to zero and a DST-REF field set to the SRC-REF of the DR TPDU received   |
| <p>NOTES</p> <p>1 This information is used by predicate P4.</p> <p>2 This action is not performed if the transport entity is the responder or if neither reassignment nor resynchronization is in progress.</p> <p>3 The method of disabling transmission of DT TPDUs is a local matter. In class 3 for example, it may be effected by setting credit to zero. In class 1, this may be effected by setting of a boolean indicator.</p> |   |

TABLE A.9/X.224

**Specific notes for classes 1 and 3**

| Name | Description   |
|------|---|
| (1)  | Any TPDU except DR and CC having an unknown destination reference   |
| (2)  | CC TPDU having an unknown destination reference or a mismatched source reference  |
| (3)  | CR TPDU which is not duplicated but rejected. If the CR TPDU is duplicated, ignore it   |
| (4)  | Or send any DT or ED TPDU waiting for transmission or use N-DATA ACKNOWLEDGE request if available and selected (class 1 only)   |
| (5)  | Same as for (9) and issue a T-DISCONNECT indication   |
| (6)  | If the resultant state is CLOSED, the reference should be frozen except in the cases described in 6.18  |
| (7)  | An ER TPDU should be sent in the cases defined in 6.6   |
| (8)  | Receipt of a DC TPDU is a protocol error since DC cannot be used for reassignment. It is recommended to stop the TWR timer ([5]) and to consider the transport connection as released (CLOSED STATE)  |
| (9)  | Receipt of one of these TPDUs in this state is a protocol error. It is recommended to stop the TWR timer ([5]), and send a DR TPDU and enter the closing state  |
| (10) | Or a DR with mismatched source reference has been received  |
| (11) | Receipt of CR in this state is only valid if the TPDU is received on a network connection to which the transport connection is not assigned. It is recommended to apply action [13]   |
| (12) | Receipt of this TPDU in this state is possible either on the network connection to which the transport connection is currently assigned or on another network connection (for the responder only). In the former case the action is as stated in the state table. In the latter case it is recommended to apply action [13] |
| (13) | This happens only when the preferred class of the CR TPDU received is class 4   |

TABLE A.10/X.224

**State table for classes 1 and 3  
(First part: connection - responder side)**

| State Event  | CLOSED   | WFTRESP                                  | WFTRESP-WR            | WBCL-WR              | WBOC                                | WBOC-WR                     | CLOSING BOC                            | CLOSING BOC-WR           |
|--|--|--|-----------------------|----------------------|-------------------------------------|-----------------------------|--|--------------------------|
| TDISreq  |  | DR<br>CLOSED<br>(6)                      | WBCL-WR               |                      | DR<br>CLOSING<br>BOC                | CLOSING<br>BOC-WR           |  |                          |
| TCONresp   |  | P10: (12);<br>not P10: CC<br>WBOC        | WBOC-WR               |                      |                                     |                             |  |                          |
| NRSTind  |  | [4] [6]<br>WFTRESP-<br>WR                | [6]<br>WFTRESP-<br>WR | [6]<br>WBCL-WR       | [4] [6]<br>WBOC-WR                  | [6]<br>WBOC-WR              | [4] [6]<br>CLOSING<br>BOC-WR           | [6]<br>CLOSING<br>BOC-WR |
| NDISind  |  | [4]<br>WFTRESP-<br>WR                    | WFTRESP-<br>WR        | WBCL-WR              | [4]<br>WBOC-WR                      | WBOC-WR                     | [4]<br>CLOSING<br>BOC-WR               | CLOSING<br>BOC-WR        |
| CR   | P7: DR<br>(3 and 7)<br>CLOSED (6);<br>not P7:<br>TCONind;<br>WFTRESP | P9:<br>WFTRESP;<br>not P9: (11)          | [5]<br>WFTRESP        | [5] DR<br>CLOSED (6) | P9: WBOC;<br>not P9: (11)           | [5] CC<br>WBOC              | P9:<br>CLOSING<br>BOC;<br>not P9: (11) | DR [5]<br>CLOSED (6)     |
| DR   | DC<br>CLOSED   | P5: DC [14]<br>(13)<br>TDISind<br>CLOSED |                       |                      | TDISind<br>DC<br>CLOSED<br>(6) (12) | DC (5)<br>TDISind<br>CLOSED | CLOSED<br>(6) (12)                     | [5] DC<br>CLOSED (6)     |
| RJ or ED   | CLOSED   |  |                       |                      | OPEN [7]<br>(12)                    | [5] [2]<br>RJ OPEN          | CLOSING<br>(12)                        | [5] DR<br>CLOSING        |
| DC   | CLOSED   |  |                       |                      |                                     |                             | CLOSED                                 | (8)                      |
| First TPDU<br>other than<br>CR, DR,<br>DC, ED,<br>ER or RJ | CLOSED   |  |                       |                      | OPEN [7]                            |                             | CLOSING                                | (9)                      |
| TWR<br>time-out  |  |  | TDISind<br>CLOSED (6) | CLOSED (6)           |                                     | TDISind<br>CLOSED (6)       |  | CLOSED (6)               |
| TDTreq   |  |  |                       |                      | [7]<br>WBOC                         | [11]<br>WBOC-WR             |  |                          |
| TEXreq   |  |  |                       |                      | [7]<br>WBOC                         | [11]<br>WBOC-WR             |  |                          |
| ER   |  |  |                       |                      | TDISind DC<br>CLOSING<br>BOC        |                             | CLOSED (6)                             |                          |

TABLE A.10/X.224 (cont'd)

**State table for classes 1 and 3  
(Second part: connection initiator side)**

| State Event     | CLOSED  | WFNC   | WFNC-R  | WFCC  | WFCC-R   | WBCL  | WBCL-R   |
|-----------------|---|--|---|---|--|---|--|
| TCONreq         | P0: TDISind<br>CLOSED; not<br>(P0 and P1):<br>NCONreq<br>WFNC;<br>not (P0 and<br>P2): WFNC;<br>not (P0 and<br>P3): CR<br>WFCC |  |   |   |  |   |  |
| NCONconf        |   | CR WFCC  | CR WFCC   |   | CR WFCC  |   | CR WBCL  |
| NRSTind         |   |  |   | P4: TDISind [6]<br>(6) [1]<br>CLOSED;<br>not P4: CR [6]<br>[8] WFCC   |  | P4: [6] CLOSED<br>[1];<br>not P4: CR [6] [8]<br>WBCL  |  |
| NDISind         |   | P1: NCONreq<br>WFNC-R [8];<br>P2: [8]<br>WFNC-R;<br>P3: CR [8]<br>WFCC | P1:<br>NCONreq<br>WFNC-R;<br>P2: WFNC-R;<br>P3: CR WFCC | P4: TDISind<br>CLOSED (6);<br>(not P4) and<br>P1: [8]<br>NCONreq<br>WFCC-R;<br>(not P4) and<br>P2: [8]<br>WFCC-R;<br>(not P4) and<br>P3: [8] CR<br>WFCC | P1: NCONreq<br>WFCC-R;<br>P2: WFCC-R;<br>P3: CR WFCC | P4 or P5: [1 and<br>9] (6) CLOSED;<br>(not (P4 or P5))<br>and P1: [8]<br>NCONreq<br>WBCL-R;<br>(not (P4 or P5))<br>and P2: [8]<br>WBCL-R;<br>(not (P4 or P5))<br>and P3: [8] CR<br>WBCL | P5: CLOSED (6)<br>[9];<br>(not P5) and P1:<br>NCONreq<br>WBCL-R;<br>(not P5) and P2:<br>WBCL-R;<br>(not P5) and P3:<br>CR WBCL |
| TDISreq         |   | [1] CLOSED<br>(6)  | [1] CLOSED<br>(6) [9]                                   | WBCL  | P5: CLOSED<br>(6) [1 and 9];<br>not P5:<br>WBCL-R    |   |  |
| DR              | (10)<br>DC<br>CLOSED (12)   |  |   | TDISind<br>[1] [9]<br>CLOSED (6)  |  | [1] [9]<br>CLOSED (6)   |  |
| CC              | DR<br>CLOSED  |  |   | P10: [12];<br>not P10:<br>TCONconf<br>AK (4) OPEN<br>[9]  |  | P10: [12];<br>not P10: DR<br>[9]<br>CLOSING   |  |
| (1)             | CLOSED  |  |   |   |  |   |  |
| (2)             | DR<br>CLOSED  |  |   |   |  |   |  |
| TTR<br>time-out |   |  | TDISind<br>[1]<br>CLOSED (6)                            | [10]  | TDISind<br>[1]<br>CLOSED (6)                         | [10]  | [1]<br>CLOSED (6)  |
| ER              |   |  |   | TDISind<br>[1] [9]<br>CLOSED (6)  |  | [1] [9]<br>CLOSED (6)   |  |



TABLE A.10/X.224 (end)

**State table for classes 1 and 3**  
**(Third part: OPEN and CLOSING states)**

| State<br>Event        | OPEN  | OPEN-R  | OPEN-WR                      | CLOSING  | CLOSING-R  | CLOSING-WR           |
|-----------------------|---|---|------------------------------|--|--|----------------------|
| NCONconf              |   | RJ [2]<br>OPEN  |                              |  | DR<br>CLOSING  |                      |
| TDISreq               | P8: CLOSING;<br>not P8: DR CLOSING  | CLOSING-R   | CLOSING-WR                   |  |  |                      |
| NRSTind               | P6 and P4: (6) [6] [3]<br>TDISind CLOSED;<br>(P6 and not P4): [6] [2]<br>[8] RJ OPEN;<br>not P6: [4 and 6]<br>OPEN  |   |                              | P6 and P4: (6) [6]<br>[3] CLOSED;<br>P6 and not P4: [6]<br>[8] DR CLOSING;<br>not P6: [4, 6]<br>CLOSING  |  |                      |
| NDISind               | P6 and P4: TDISind<br>CLOSED (6);<br>(P6 and not P4) and<br>P1: [8] NCONreq<br>OPEN-R;<br>(P6 and not P4) and<br>P2: [8] OPEN-R;<br>(P6 and not P4) and<br>P3: [8] [2] RJ OPEN;<br>not P6: [4]<br>OPEN-WR | P1: NCONreq<br>OPEN-R;<br>P2: OPEN-R;<br>P3: [2] RJ<br>OPEN |                              | P6 and (P5 or P4):<br>CLOSED (6);<br>P6 and not (P4 or<br>P5) and P1: [8]<br>NCONreq<br>CLOSING-R;<br>P6 and not (P4 or P5)<br>and P2: [8]<br>CLOSING-R;<br>P6 and not (P4 or<br>P5) and P3: [8]<br>DR CLOSING;<br>not P6: [4]<br>CLOSING-WR | P5: CLOSED (6);<br>(not P5 and P1):<br>NCONreq<br>CLOSING-R;<br>(not P5) and P2:<br>CLOSING-R;<br>(not P5) and P3:<br>DR CLOSING |                      |
| RJ or ED              | P8: [5] [2] RJ OPEN;<br>not P8: [7] [9] OPEN<br>(12)  |   | RJ [5 and 2]<br>OPEN         | P8: [5] DR<br>CLOSING;<br>not P8: [9]<br>CLOSING (12)  |  | DR [5]<br>CLOSING    |
| TWR time-<br>out      | TDISind (6) CLOSED  |   | TDISind (6)<br>CLOSED        | CLOSED (6)   |  | CLOSED (6)           |
| DR                    | P8: TDISind DC (6)<br>[5] CLOSED;<br>not P8: TDISind DC<br>(6) [9] CLOSED (12)  |   | TDISind DC [5]<br>CLOSED (6) | P8: [5] DC (6)<br>CLOSED;<br>not P8: [3] [9] (6)<br>CLOSED (12)  |  | [5] CLOSED<br>(6) DC |
| DC                    |   |   |                              | P8: (8);<br>not P8: [3] [9]<br>CLOSED (6)  |  | (8)                  |
| DT, AK, or<br>EA TPDU | [7] OPEN  |   | (5)                          | CLOSING  |  | (9)                  |
| TTR time-<br>out      | [10]  | TDISind<br>CLOSED [1] (6)                                   |                              | [10]   | CLOSED<br>[1] (6)  |                      |
| TDTreq                | P8: [11] OPEN;<br>not P8: [7] OPEN  | [11]<br>OPEN-R  | [11]<br>OPEN-WR              |  |  |                      |
| TEXreq                | P8: [11] OPEN;<br>not P8: [7] OPEN  | [11]<br>OPEN-R  | [11]<br>OPEN-WR              |  |  |                      |
| ER                    | TDISind<br>DR CLOSING   |   | TDISind<br>DR CLOSING        | CLOSED<br>(6)  |  | CLOSED<br>(6)        |

## A.6 State tables for class 4 over CONS

This clause provides a more precise description of a class 4 Transport Connection.

Tables A.11, A.12 and A.13 give the predicates, actions and notes for class 4 respectively.

Table A.14 is the state table for a class 4 transport connection.

The following assumptions and notations are used:

- a) the state of every network connection is known as being open or opening (i.e. a NCONreq has been issued and the NCONconf is awaited);
- b) for each transport connection the transport entity maintains the set of network connections to which the transport is assigned. A network connection in this set is either in open or opening state;
- c) when an N-CONNECT confirmation, N-RESET indication or N-DISCONNECT indication is received this event is associated with the transport connection if the network connection belongs to the set;
- d) when an N-DISCONNECT is received, the network connection becomes unexisting and is therefore withdrawn from the set. When a NCONconf is received the state of the nc becomes “open”;

NOTE – This is not shown by an explicit action in the state table. Conversely adding a network connection to a set and setting its state to “opening” is shown by an explicit action.

- e) when the state goes back to CLOSED or REFWAIT state, it is assumed that all timers are stopped (if running), the count is set to zero and the set becomes empty;
- f) when a TPDU is received the network connection on which it has been received is assumed to be known;
- g) the variable “current-nc” is used to designate either the network connection on which a TPDU has been received or the network connection which has been chosen for a new assignment (either an existing one or a new one which is created);
- h) the following variables are also used:

local-ref: the reference (local) of the TC is chosen when sending the CR or when accepting a CR;

remote-ref: the reference of remote entity is initially set to zero and initialized when processing the CC except if the CC is ignored;

SRC-REF: designates the corresponding field of the received TPDU;

DST-REF: designates the corresponding field of the received TPDU;

src-ref, dst-ref: designates the corresponding field of the sent TPDU;

count: designates the number of times a TPDU has been sent (retransmissions);

- i) the data transfer phase is not completely described in the state table but refers to the main text;
- j) a spontaneous event called “new network connection assignment” has been introduced. It may occur at any time provided P1 or P2 are true (see Table A.11) and the remote ref is not zero (i.e. a CR TPDU has been received or a CC TPDU has been received and processed);
- k) when an N-RESET indication is received, an N-RESET response is issued.

TABLE A.11/X.224

**Predicates for class 4 over CONS**

| Name | Description  |
|------|--|
| P0   | T-CONNECT Request is acceptable  |
| P1   | An assignment can be done to a suitable network connection (either open or opening)  |
| P2   | It is possible to open a new network connection  |
| P3   | Local choice   |
| P4   | A CR TPDU has never been sent  |
| P5   | The transport entity is the initiator and the set of network connections is now empty (i.e. a new assignment shall be done) or a new assignment is decided as a local choice |
| P6   | Local choice not to perform a new assignment if the set of network connections is empty (for Closing state only)   |
| P7   | Count = maximum  |
| P8   | Acceptable CR TPDU   |
| P9   | Acceptable class 4 CC TPDU   |
| P10  | Unacceptable class 4 CC TPDU   |
| P11  | CC TPDU not specifying class 4   |

TABLE A.12/X.224

**Specific actions for class 4 over CONS**

| Name | Description  |
|------|--|
| [0]  | Set reference timer  |
| [1]  | Count = count + 1  |
| [2]  | Count = 0  |
| [3]  | Set retransmission timer   |
| [4]  | Stop retransmission timer if running   |
| [5]  | Set window timer   |
| [6]  | Stop window timer if running   |
| [7]  | Set inactivity timer   |
| [8]  | Stop inactivity timer  |
| [9]  | Set initial credit for sending according to the received CR/CC TPDU  |
| [10] | Set initial credit for controlling reception according the the sent CR/CC TPDU   |
| [11] | Send the CR TPDU if there is a network connection in the open state in the set   |
| [12] | Add the current network connection to the set, if not already included   |
| [13] | The current network connection is now in opening state   |
| [14] | Send the CC TPDU if a network connection in the open state is in the set   |
| [15] | Send the DR TPDU if a network connection in the open state is in the set. This DR TPDU is sent with SRC-REF = local-ref and DST-REF = remote-ref (may be zero) |
| [16] | Send the DR TPDU if a network connection in the open state is in the set. The DR TPDU is sent with SRC-REF = 0 and DST-REF = remote-ref                        |
| [17] | Send a TPDU according to data transfer procedure   |
| [18] | See state table of the class specified in the CC TPDU (refer to data transfer)   |
| [19] | See state table of the class (refer to release procedure): send a DR TPDU if the class is not 0, otherwise issue an N-DISCONNECT request                       |
| [20] | Store request and exercise flow control to the user  |
| [21] | Send a DR TPDU with SRC-REF field set to zero  |
| [22] | Send a DC TPDU except if the SRC-REF field of the received DR TPDU is equal to zero  |

TABLE A.13/X.224

**Specific notes for class 4 over CONS**

| Name | Description  |
|------|--|
| (1)  | Not possible as no set of Network Connections is associated with this transport connection   |
| (2)  | It is also possible to remain in the same state (T1 is still running) until <ul style="list-style-type: none"> <li>– a CC TPDU is received which performs a new assignment;</li> <li>– a new assignment is tried (spontaneous event);</li> <li>– T1 runs out and the count is equal to the maximal value</li> </ul>  |
| (3)  | No new assignment was possible: if the set is empty, the transport entity waits until a new assignment is received, or can be locally performed (spontaneous event)  |
| (4)  | It is also possible to perform a new assignment. (This may be done in triggering the event “new network connection assignment”)  |
| (5)  | Not a duplicated CR TPDU. If the CR TPDU is duplicated, ignore it  |
| (6)  | Since a new network connection is now assigned, it is recommended that the appropriate TPDU be sent on this network connection (if open) in order to make the remote entity aware of this assignment. It is also possible to allow the normal retransmission procedures to cause the TPDU to be sent; however, the first TPDU available for sending should be sent on the new network connection |
| (7)  | As a local choice it is also possible to apply the following: [0], TDISind, REFWAIT  |
| (8)  | Association to this transport connection is carried out regardless of the SRC-REF field. If the SRC-REF is not zero, a DC TPDU is sent back  |
| (9)  | At least an AK TPDU shall be sent if the transport entity is in the initiator in order to ensure that the responder will complete its three-way handshake  |
| (10) | If association has been made, and DST-REF is zero, then the DC TPDU contains a SRC-REF field set to zero   |
| (11) | If the CLOSING state has been entered, coming from WFCC state, the remote-ref is zero. The SRC-REF field of the CC TPDU is ignored (i.e. if the DR TPDU is retransmitted, it will be with DST-REF field set to zero)   |
| (12) | If the CLOSING state has been entered, coming from WFCC state, the remote-ref (which is zero) shall be set with SRC-REF in order to comply with the release procedure of the negotiated class  |
| (13) | The DR TPDU may be either repeated immediately or when T1 will run out   |
| (14) | If the set is empty, this event may be used as a criteria for triggering the event “new network connection assignment”   |
| (15) | Previously stored T-DATA or T-EXPEDITED DATA requests are ready for processing according to data transfer procedures   |
| (16) | See data transfer procedures   |
| (17) | When an N-RESET indication is received, an N-RESET response has to be issued once independent of the state automata  |

TABLE A.14/X.224

**Class 4 connection/disconnection over CONS**

| State Event                       | REFWAIT | CLOSED  | WFCC  | WBCL  | OPEN   | WFTRESP   | AKWAIT   | CLOSING  |
|-----------------------------------|---------|---|---|---|--|---|--|--|
| TCONreq                           |         | not P0:<br>TDISind<br>CLOSED;<br>P0 and P1:<br>[12, 1, 3, 10<br>and 11]<br>WFCC;<br>P0 and not<br>P1 and P2:<br>[13, 12, 1, 3<br>and 10]<br>NCONreq<br>WFCC;<br>P0 and not<br>P2: TDISind<br>CLOSED |   |   |  |   |  |  |
| TCONresp                          |         |   |   |   |  | [3, 2, 1, 10<br>and 14]<br>AKWAIT   |  |  |
| TDISreq                           |         |   | P4: CLOSED;<br>(not P4) and<br>P3: WBCL;<br>(not P4) and<br>(not P3): [4, 3,<br>2, 1 and 15]<br>CLOSING                             |   | [6, 8, 4, 3, 2,<br>1 and 15]<br>CLOSING  | [16]<br>CLOSED  | [4, 3, 2, 1<br>and 15]<br>CLOSING  |  |
| NDISind                           | (1)     | (1)   | P1: [12]<br>WFCC;<br>(not P1) and<br>P2: [13 and 12]<br>NCONreq<br>WFCC;<br>(not P1) and<br>(not P2): [0]<br>[2] TDISind<br>REFWAIT | P3: [0]<br>REFWAIT;<br>(not P3) and<br>P1: [12 and<br>11] WBCL;<br>(not P3) and<br>(not P1) and<br>P2: [13 and<br>12]<br>NCONreq<br>WBCL;<br>(not P3) and<br>(not P1) and<br>(not P2): [0]<br>REFWAIT | P5 and P1:<br>[12 and 17]<br>(6) OPEN;<br>P5 and (not<br>P1) and P2:<br>[13 and 12]<br>NCONreq<br>OPEN;<br>P5 and (not<br>P1) and (not<br>P2): OPEN<br>(3);<br>not P5:<br>OPEN | WFTRESP<br>(4)  | P5 and P1:<br>[12 and 14]<br>(6)<br>AKWAIT;<br>P5 and (not<br>P1) and P2:<br>[13 and 12]<br>NCONreq<br>AKWAIT;<br>P5 and (not<br>P1) and (not<br>P2):<br>AKWAIT<br>(3);<br>not P5:<br>AKWAIT | P6: [0]<br>REFWAIT;<br>(not P6) and P5<br>and P1: [12 and<br>15] CLOSING<br>(6);<br>(not P6) and P5<br>and (not P1) and<br>P2: [13 and 12]<br>NCONreq<br>CLOSING;<br>(not P6) and P5<br>and (not P1) and<br>(not P2):<br>CLOSING (3);<br>(not P6) and (not<br>P5): CLOSING |
| NRSTind                           |         |   | (17)  | (17)  | (17)   | (17)  | (17)   | (17)   |
| TDTreq<br>TEXreq                  |         |   |   |   | (16) OPEN  |   | [20]<br>AKWAIT   |  |
| NCONconf                          | (1)     | (1)   | CR<br>WFCC (6)  | CR<br>WBCL (6)  | [17]<br>OPEN (6)   | WFTRESP   | CC AKWAIT<br>(6)   | [15] CLOSING<br>(6)  |
| New network connection assignment |         |   |   |   | P1: [12 and<br>17] OPEN<br>(6);<br>(not P1) and<br>P2: [13 and<br>12]<br>NCONreq<br>OPEN   | P1: [12]<br>WFTRESP<br>(6);<br>(not P1) and<br>P2: [13 and<br>12]<br>NCONreq<br>WFTRESP | P1: [12 and<br>14] (6)<br>AKWAIT;<br>(not P1) and<br>P2: [13 and<br>12]<br>NCONreq<br>AKWAIT   | P1: [12 and 15]<br>(6) CLOSING;<br>(not P1) and P2:<br>[13 and 12]<br>NCONreq<br>CLOSING   |

TABLE A.14/X.224 (end)

## Class 4 connection/disconnection over CONS

| State<br>Event   | REFWAIT      | CLOSED  | WFCC   | WBCL   | OPEN  | WFTRESP                | AKWAIT   | CLOSING   |
|------------------|--------------|---|--|--|---|------------------------|--|---|
| Retrans-timer    |              |   | P7 and P3: [0] TDISind REFWAIT; P7 and (not P3): [3, 2, 1 and 15] TDISind CLOSING (14); not P7: [1, 3 and 11] WFCC | P7 and P3: [0] REFWAIT; P7 and (not P3): [3, 2, 1 and 15] CLOSING (14); not P7: [1, 3 and 11] WBCL | P7: [6, 8, 3, 2, 1 and 15] TDISind CLOSING (14); not P7: (16) (14) OPEN |                        | P7: [3, 2, 1 and 15] TDISind (14) CLOSING; not P7: [1, 3 and 14] (14) AKWAIT | P7: [0] REFWAIT; not P7: [1, 3 and 15] (14) CLOSING |
| Inactivity-timer |              |   |  |  | [6, 4, 3, 2, 1 and 15] TDISind CLOSING (7)                              |                        |  |   |
| Reference timer  | CLOSED       |   |  |  |   |                        |  |   |
| CR               |              | not P8: [21] CLOSED (5); P8: [9 and 12] TCONind WFTRESP (5) |  |  | [12, 8 and 7] OPEN  | [12] WFTRESP           | [12 and 14] AKWAIT   | [12] CLOSING (13)                                   |
| CC               | DR REFWAIT   | DR CLOSED   | P9: [12, 9, 2, 4, 5, 7 and 17] TCONconf (9) OPEN; P10: [12, 4, 3, 2, 1 and 15] TDISind CLOSING; P11: [18]          | P11: [19]; not P11: [12, 2, 4, 3, 1 and 15] CLOSING  | [12, 17, 8 and 7] (9) OPEN  |                        |  | P11: [19] (12); not P11: [12] CLOSING (11)          |
| ER               | REFWAIT      | CLOSED  | [0] TDISind REFWAIT  | [0] REFWAIT  | [12, 6, 8, 4, 3, 2, 1 and 15] TDISind CLOSING                           |                        | [12, 4, 3, 2, 1 and 15] TDISind CLOSING                                      | [0] REFWAIT   |
| DR               | [22] REFWAIT | [22] CLOSED   | (8) TDISind [0] REFWAIT  | (8) [0] REFWAIT  | DC (10) [0] TDISind REFWAIT   | DC (10) TDISind CLOSED | DC (10) [0] TDISind REFWAIT  | [0] REFWAIT   |
| DC               | REFWAIT      | CLOSED  |  |  |   |                        |  | [0] REFWAIT   |
| EA               | REFWAIT      | CLOSED  |  |  | [12, 8 and 7] OPEN (16)   |                        |  | [12] CLOSING (13)                                   |
| DT/AK/ED         | REFWAIT      | CLOSED  |  |  | [12, 8 and 7] OPEN (16)   |                        | [12 and 7] OPEN (15) (16)  | [12] CLOSING (13)                                   |

## A.7 State tables for class 4 over CLNS

This clause provides a more precise description of a class 4 transport connection when operating over CLNS.

Tables A.15, A.16 and A.17 give the predicates, actions and notes for class 4 respectively.

Table A.18 is the state table for a class 4 transport connection when operating over CLNS.

The following assumption and notations are used

- a) local-ref: the reference (local) of the TC is chosen when sending the CR or when accepting a CR;  
remote-ref: the reference of the remote entity is initially set to zero and initialized when processing the CC except if the CC is ignored;  
SRC-REF: designates the corresponding field of the received TPDU;  
DST-REF: designates the corresponding field of the received TPDU;  
src-ref, dst-ref: designates the corresponding fields of the sent TPDU;  
count: designates the number of times a TPDU has been sent (retransmissions);
- b) the data transfer phase is not completely described in the state table but refers to the main text;
- c) it is assumed that the network service is continuously available;

The operations resulting from signalled inaccessibility of the network service are a local matter.

TABLE A.15/X.224

### Predicates for class 4 over CLNS

| Name | Description                     |
|------|---------------------------------|
| P0   | T-CONNECT request is acceptable |
| P3   | Local choice                    |
| P7   | Count = maximum                 |
| P8   | Acceptable CR TPDU              |
| P9   | Acceptable class 4 CC TPDU      |



TABLE A.16/X.224

**Specific actions for class 4 over CLNS**

| Name | Description  |
|------|--|
| [0]  | Set reference timer  |
| [1]  | Count = count + 1  |
| [2]  | Count = 0  |
| [3]  | Set retransmission timer   |
| [4]  | Stop retransmission timer if running   |
| [5]  | Set window timer   |
| [6]  | Stop window timer if running   |
| [7]  | Set inactivity timer   |
| [8]  | Stop inactivity timer if running   |
| [9]  | Set initial credit for sending according to the received CR/CC TPDU                                    |
| [10] | Set initial credit for controlling reception according to the sent CR/CC TPDU                          |
| [15] | Send the DR TPDU. This DR TPDU is sent with src-ref = local-ref and dst-ref = remote-ref (may be zero) |
| [16] | Send the DR TPDU. The DR TPDU is sent with src-ref = 0 and dst-ref = remote-ref                        |
| [17] | Send a TPDU according to data transfer procedure   |
| [20] | Store request and exercise flow control to the user  |
| [21] | Send a DC TPDU with SRC-REF equal to zero  |
| [22] | Send a DC TPDU except if the SRC-REF field of the received DR TPDU is equal to zero                    |
| [23] | Send a DR TPDU with src-ref = local-ref and dst-ref = SRC-REF in CC TPDU                               |

TABLE A.17/X.224

**Specific notes for class 4 over CLNS**

| Name | Description  |
|------|--|
| (5)  | Not a duplicated CR TPDU. If the CR TPDU is duplicated, ignore it  |
| (7)  | As a local choice it is also possible to apply the following [0], TDISind, REFWAIT   |
| (8)  | Association to this Transport connection is done regardless of the SRC-REF field. If SRC-REF is not zero, a DC TPDU is set back  |
| (9)  | At least an AK TPDU shall be sent if the transport entity is the initiator in order to ensure that the responder will complete its three-way handshake   |
| (10) | If association has been made, and DST-REF is zero, then the DC TPDU contains a src-ref field set to zero   |
| (11) | If the CLOSING state has been entered, coming from WFCC state, the remote-ref is zero. The SRC-REF field of the CC TPDU is ignored (i.e. if the DR TPDU is retransmitted, it will be with the dst-ref field set to zero) |
| (13) | The DR TPDU may be either repeated immediately or when T1 will run out.  |
| (15) | Previously stored T-DATA or T-EXPEDITED-DATA requests are ready for processing according to data transfer procedures   |
| (16) | See data transfer procedures   |

TABLE A.18/X.224

**Class 4 connection/disconnection over CLNS**

| State<br>Event       | REFWAIT | CLOSED  | WFCC  | WBCL   | OPEN  | WFTRESP                    | AKWAIT   | CLOSING  |
|----------------------|---------|---|---|--|---|----------------------------|--|--|
| TCONreq              |         | not P0:<br>TDisind<br>CLOSED;<br>P0:<br>[1,3,10] CR<br>WFCC               |   |  |   |                            |  |  |
| TCONresp             |         |   |   |  |   | [3,2,1,10]<br>CC<br>AKWAIT |  |  |
| TDisreq              |         |   | P3:<br>WBCL;<br>not P3:<br>[4,3,2,1,15]<br>CLOSING  |  | [6,8,4,3,<br>2,1,15]<br>CLOSING   | [16]<br>CLOSED             | [4,3,2,1,15]<br>CLOSING  |  |
| TDTreq<br>TEXreq     |         |   |   |  | (16)<br>OPEN  |                            | [20]<br>AKWAIT   |  |
| Retrans-<br>timer    |         |   | P7 and P3:<br>[0]<br>TDisind<br>REFWAIT;<br>P7 and (not P3):<br>[3,2,1,15]<br>TDisind<br>CLOSING;<br>not P7:<br>[1,3], CR<br>WFCC | P7 and P3:<br>[0]<br>REFWAIT;<br>P7 and<br>(not P3):<br>[3,2,1,15]<br>CLOSING;<br>not P7:<br>[1,3], CR<br>WBCL | P7:<br>[6,8,3,2,1,15]<br>TDisind<br>CLOSING;<br>not P7:<br>(16)<br>OPEN |                            | P7:<br>[3,2,1,15]<br>TDisind<br>CLOSING;<br>not P7:<br>[1,3], CC<br>AKWAIT | P7:<br>[0]<br>REFWAIT;<br>not P7:<br>[1,3,15]<br>CLOSING |
| Inactivity-<br>Timer |         |   |   |  | [6,4,3,2,1,15]<br>TDisind<br>CLOSING<br>(7)                             |                            |  |  |
| Reference-<br>timer  | CLOSED  |   |   |  |   |                            |  |  |
| CR                   |         | not P8:<br>[21]<br>CLOSED;<br>P8:<br>[1,9,3]<br>TCONind<br>WFTRESP<br>(5) |   |  | [8,7]<br>OPEN   | WFTRESP                    | CC<br>AKWAIT   | CLOSING<br>(13)  |

TABLE A.18/X.224 (end)

**Class 4 connection/disconnection over CLNS**

| State<br>Event | REFWAIT         | CLOSED         | WFCC   | WBCL                           | OPEN                                       | WFTRESP                      | AKWAIT                               | CLOSING                |
|----------------|-----------------|----------------|--|--------------------------------|--|------------------------------|--------------------------------------|------------------------|
| CC             | DR<br>REFWAIT   | DR<br>CLOSED   | P9:<br>[9,2,4,5,7,17]<br>TCONconf<br>(9)<br>OPEN;<br>not P9:<br>[4,3,2,1,23]<br>TDisInd<br>CLOSING | P9:<br>[2,4,3,1,15]<br>CLOSING | [17,8,7]<br>(9)<br>OPEN                    |                              |                                      | P9:<br>(11)<br>CLOSING |
| ER             | REFWAIT         | CLOSED         | [0]<br>TDisInd<br>REFWAIT  | [0]<br>REFWAIT                 | [6,8,4,3,2,<br>1,15]<br>TDisInd<br>CLOSING |                              | [4,3,2,1,15]<br>TDisInd<br>CLOSING   | [0]<br>REFWAIT         |
| DR             | [22]<br>REFWAIT | [22]<br>CLOSED | (8)<br>[0]<br>TDisInd<br>REFWAIT   | (8)<br>[0]<br>REFWAIT          | DC (10)<br>[0]<br>TDisInd<br>REFWAIT       | DC (10)<br>TDisInd<br>CLOSED | DC (10)<br>[0]<br>TDisInd<br>REFWAIT | [0]<br>REFWAIT         |
| DC             | REFWAIT         | CLOSED         |  |                                |  |                              |                                      | [0]<br>REFWAIT         |
| EA             | REFWAIT         | CLOSED         |  |                                | [8,7]<br>OPEN (16)                         |                              |                                      | CLOSING<br>(13)        |
| DT/AK/ED       | REFWAIT         | CLOSED         |  |                                | [8,7]<br>OPEN (16)                         |                              | [7]<br>OPEN<br>(15) (16)             | CLOSING<br>(13)        |

## **Annex B** (normative)

### **Network connection management subprotocol**

(This annex forms an integral part of this Recommendation)

#### **B.1 Introduction**

The objectives of this annex are to

- a) Provide for more flexibility in the use of the network connections established between two cooperating transport entities, thus enlarging the field of application of the transport protocol as presently defined in the main body of this Recommendation. In particular it allows for an optimization of the use of the network connections by allowing both transport entities at each end of a network connection to assign and reassign transport connections to a network connection;
- b) Allow more information to be sent explaining why a network connection is released in order to be able to optimize recovery.

The protocol described in this annex is called the network connection management subprotocol (NCMS).

The procedures defined in this annex are optional extensions to the main body of this Recommendation.

#### **B.2 Scope**

The procedures specified in this annex are an extension of the basic procedure defined in the main body of this Recommendation and therefore do not prevent communication between transport entities conforming to this Recommendation with this annex and those conforming to this Recommendation without this annex.

The basic network connection management that is specified in the main body of this Recommendation allows for assignment or reassignment of transport connections on an existing network connection by its owner, who is currently restricted to be the transport entity that initiated this network connection. This annex describes the procedures necessary to extend this basic management to permit the peer transport entity (i.e. the acceptor of a network connection) to become also the owner of the network connection and consequently to be able to assign or reassign transport connections to it.

When performing multiplexing of transport connection this feature allows a network connection to be fully shared, thus increasing the scope of the multiplexing classes of the transport protocol (i.e. classes 2, 3 and 4).

In order to control the number of shared network connections that peer entities are willing to use simultaneously (one or more), a mechanism is provided to resolve collisions when simultaneous network connection establishments occur, especially in the case of recovery after network failure.

#### **B.3 Definitions**

For the purposes of this annex, the following definitions apply.

NOTE – The definitions contained in this clause make use of abbreviations defined in clause B.4.

**B.3.1 owner (of a network connection):** The transport entity that issued the N-CONNECT request leading to the creation of the network connection if the NCM TPDU is not used or the transport entity (possibly both) which is designated to have the right of performing assignment in accordance with the NC-RIGHT field of the NCM TPDU when the NCM TPDU is used (see B.6.2.2).

NOTE – This definition extends the definition of the owner of the network connection given in 3.2.28.

**B.3.2 network connection reference (or nc-reference):** An identifier which is associated with a network connection and used to resolve collisions when network connections are reopened.

## **B.4 Abbreviations**

### **B.4.1 Types of transport-protocol-data-units**

|           |   |
|-----------|---|
| NCM TPDU  | Network connection management TPDU              |
| DIAG TPDU | Diagnostic TPDU                                 |
| NCCM TPDU | Network connection management confirmation TPDU |

The following TPDU is used by this annex and is defined by ITU-T Rec. X.264 | ISO/IEC 11570, Transport protocol identification mechanism:

|         |                                |
|---------|--------------------------------|
| UN TPDU | Use of network connection TPDU |
|---------|--------------------------------|

### **B.4.2 TPDU fields**

|          |  |
|----------|--|
| NC-REF   | Network connection reference (field)   |
| NC-TYPE  | Network connection type (field)        |
| NC-RIGHT | Network right (field)                  |
| LI       | Length indicator (field)               |
| NC-PREF  | Network connection preference (field)  |
| NC-COL   | Network connection collision indicator |
| NC-REC   | Network connection recovery indicator  |

The following fields of the UN TPDU are used by this annex and are defined by ITU-T Rec. X.264 | ISO/IEC 11570:

|        |                             |
|--------|-----------------------------|
| SHARE  | Sharing option (field)      |
| PRT-ID | Protocol identifier (field) |

### **B.4.3 Timers**

|        |   |
|--------|---|
| TTR-NC | Time to try to reopen a network connection using a given NC-REF |
| TPD-NC | Time to consider a given NC-REF as pending                      |
| TFR-NC | Time to consider a given NC-REF as frozen                       |

### **B.4.4 Miscellaneous**

|      |   |
|------|---|
| NCMS | Network connection management subprotocol             |
| NSAP | Network-service-access-point                          |
| AA   | Assignment right to all                               |
| SA   | Sender has assignment right                           |
| RA   | Receiver has assignment right                         |
| AFI  | Authority and Format Identifier (of the NSAP address) |
| IDI  | Initial Domain Identifier (of the NSAP address)       |
| DSP  | Domain specific part (of the NSAP address)            |

## **B.5 Overview of the protocol**

NCMS allows for:

- a) identification of the protocol to be used on top of a given network connection;

NOTE – The use of NSAP addresses as it is defined in CCITT Rec. X.650 | ISO/IEC 7498-3 provides greater flexibility in distinguishing between OSI and non-OSI users of the network service. If however the use of NSAPs incurs unacceptable penalties, for example where each NSAP is charged for by the network provider, then the protocol identification mechanism (see ITU-T Rec. X.264 | ISO/IEC 11570) is available.

- b) Explicit designation of the transport entity (or entities) which has the right to assign transport connection(s) to a specific network connection and is therefore considered as the (co-)owner of the network connection;
- c) Resolution of connection establishment collisions when a network connection is first established or recovered after failure.

NCMS assumes the use of the network service defined in CCITT Rec. X.213 | ISO/IEC 8348.

When operating NCMS the transport entities use only the network service primitives listed in Table B.1 (the other network service primitives are used as defined in 5.2).

TABLE B.1/X.224

**Network Service Primitives Used for NCMS Operation**

| Primitives  |            | Parameters                     | A/B/C |
|---|------------|--------------------------------|-------|
| N-CONNECT   | request    | Called address                 | A     |
|   | indication | Calling address                | A     |
|   | response   | NS user-data                   | B     |
|   | confirm    | QOS parameter set              | A     |
|   |            | Responding address             | A     |
|   |            | Receipt confirmation selection | A     |
| N-DISCONNECT  | request    | NS user-data                   | C     |
|   | indication | Originator                     | C     |
|   |            | Reason                         | A     |
| <p>A This parameter is used in accordance with the procedures specified in the main body of this Recommendation.</p> <p>B When operating NCMS this parameter is used in request and indication and in response and confirmation if the NCMC TPDU is used.</p> <p>C This parameter may be optionally used when operating NCMS.</p> |            |                                |       |

**B.6 Elements of procedure**

**B.6.1 TPDU transfer**

The transport-protocol-data-units (TPDUs) defined for this annex are listed in B.4.1.

The transport entities shall transmit and receive the UN (see ITU-T Rec. X.264 | ISO/IEC 11570) and NCM TPDU in the NS-user data parameter of the N-CONNECT request and indication primitive only.

The sending transport entity shall:

- a) Either not transmit any TPDU in the NS-user data parameter of the N-CONNECT request primitive;
- b) Or transmit the UN TPDU (see ITU-T Rec. X.264 | ISO/IEC 11570) followed by the NCM TPDU in the NS-user data parameter of the N-CONNECT request primitive.

When used the DIAG TPDU is transmitted in the NS-user data parameter of N-DISCONNECT primitive.

When used the NCMC TPDU is transmitted in the NS-user data parameter of N-CONNECT response and confirmation primitive.

**B.6.2 Network connection management**

**B.6.2.1 General**

When the procedure described in B.6.1 b) is used

- a) The sending transport entity shall use the procedure described below together with the procedure defined in the main body of this Recommendation;
- b) The receiving transport entity shall
  - 1) either ignore the NCM TPDU and operate the procedure described in the main body of this Recommendation;
  - 2) or recognize and process the NCM TPDU and therefore operate the procedure described below together with those defined in the main body of this Recommendation.

When a transport entity has processed an NCM TPDU received from a given NSAP [see B.6.2.1 b)2)] it shall process further NCM TPDU's received from the same NSAP.

**B.6.2.2 Assignment right**

When an N-CONNECT request primitive is issued by a transport entity to request the opening of a new network connection, the transport entity may choose whether or not to include the NCM TPDU in the NS-user data parameter of the primitive. The recipient may choose not to process the NCM TPDU and to operate the procedures defined in the main body of this Recommendation instead.

The owner(s) may use the network connection for assigning or reassigning transport connections with the following restrictions:

- a) a transport entity which is the owner of the network connection shall not assign a transport connection with a preferred class 0 or 1 if its peer is also the owner of the network connection (see Note 2);
- b) a transport entity which is the owner of the network connection can assign a transport connection with an alternative class 0 or 1, but shall not, when receiving a CR TPDU proposing 0 or 1 as an alternative class, select one of these classes (see Note 3).

A transport entity shall be designated the "owner" of a network connection according to the Table B.2.

TABLE B.2/X.224

**Determination of assignment rights**

| Event  | Entity | Network connection initiator | Network connection responder |
|--|--------|------------------------------|------------------------------|
| No NCM sent                                    |        | Y                            | N                            |
| NCM sent but not processed<br>Right = SA or AA |        | Y                            | N                            |
| NCM sent but not processed<br>Right = RA       |        | N<br>(see Note 4)            | N<br>(see Note 4)            |
| NCM sent and processed<br>Right = SA           |        | Y                            | N                            |
| NCM sent and processed<br>Right = RA           |        | N                            | Y                            |
| NCM sent and processed<br>Right = AA           |        | Y                            | Y                            |
| Y Owner<br>N Not owner                         |        |                              |                              |

NOTE 1 – The use of a network connection by a called transport entity to initiate new transport connections should only be made when the called transport entity is adequately assured of the true identity of the calling transport entity (i.e. there is trust in the calling NSAP identification provided by the Network Layer) or the data to be transferred is not sensitive.

NOTE 2 – This gives the guarantee that transport connections of classes 0 or 1 cannot be opened simultaneously at both ends of a network connection.

NOTE 3 – This allows a transport entity which has sent the NCM TPDU to still propose classes 0 or 1 as an alternative class. If the peer transport entity has not processed the NCM TPDU it may still select class 0 or class 1.

NOTE 4 – Use of NC-RIGHT with the NCM TPDU allows explicit control of assignment rights whilst also permitting both entities to be able to recover a failed network connection. This is not possible when the NCM is used.

Provided that the restriction stated in B.6.2.2 a) and B.6.2.2 b) are respected, both transport entities at each end of the network connection shall follow the procedures defined in the main body of this Recommendation, except that the owner of the network connection is defined as in B.6.2.2.

NOTE 5 – The transport protocol defined in the main body of this Recommendation makes use of the definition of owner of a network connection for defining the entity which can perform assignment and reassignment.

### **B.6.2.3 Network connection reference (nc-reference) management**

When a transport entity elects to use the NCM TPDU it shall keep track of the nc-references used in the NCM TPDU sent or received in the NS-user data parameter of the N-CONNECT request or indication primitives.

An nc-reference is associated with

- a) the pair of NSAPs addresses involved in the network connection on which the NCM TPDU has been transferred;
- b) the source of the allocation: the nc-reference has been remotely or locally allocated.

The nc-reference is exchanged as the NC-REF parameter of the NCM TPDU. The NC-TYPE parameter of the NCM TPDU indicates the source of allocation:

- 1) NC-TYPE set to NEW indicates a new nc-reference allocated by the sender of the NCM TPDU,
- 2) NC-TYPE set to MY indicates a recovery using an nc-reference previously allocated by the sender of the NCM TPDU,
- 3) NC-TYPE set to YOURS indicates a recovery using an nc-reference previously allocated by the receiver of the NCM TPDU;

NOTE 1 – Use of the NC-TYPE MY permits the explicit distinction between the two cases where the NC initiator either has or has not received the N-CONNECT confirm.

- c) the state of the nc-reference which can be
  - 1) OPEN: There is one network connection associated with the nc-reference and for which an N-CONNECT confirm has been received or an N-CONNECT response sent, and no subsequent N-DISCONNECT primitive exchanged,
  - 2) OPENING: There is one network connection for which an N-CONNECT confirm is awaited and the nc-reference has never been previously in the OPEN state,
  - 3) RECOVER: There is one network connection associated with the nc-reference for which an N-CONNECT confirm is awaited and the nc-reference has previously been in the OPEN state,
  - 4) PENDING: There is no network connection associated with the nc-reference;



- d) the assignment right allocated for the use of the network connection associated with the nc-reference. It can be
- 1) my-side: The local transport entity is the only owner of the network connection,
  - 2) remote-side: The remote transport entity is the only owner of the network connection,
  - 3) both-sides: Both the local and the remote transport entities are the owners of the network connection;

NOTE 2 – Due to the collision and recovery mechanisms, it is possible that different network connections – initiated either by the local or the remote transport entity – be consecutively associated to the same nc-reference. The assignment rights are attached with the nc-reference and remain unchanged independent of whichever transport entity is the initiator of the network connection currently used.

- e) The preference to be used in the collision resolution mechanism (see B.6.2.5). This value is equal to the value of the field of the last NCM TPDU sent and is only significant when an N-CONNECT confirm is awaited (i.e. the nc-reference is either in the OPENING or in the RECOVERY state).

When an nc-reference which has been locally allocated is no longer needed, the nc-reference shall not be reused before a TFR-NC period of time. No information is associated with this frozen reference other than the TRF-NC timer and will be considered unknown if received in an NCM TPDU.

NOTE 3 – In order to prevent that, in collision cases, two nc-references have the same value, it is required to allocate values at random, for example based on the time of their allocation.

#### **B.6.2.4 Timers**

The network connection management procedure makes use of the following timers:

- a) the TTR-NC timer defines the period of time that shall not be exceeded when reopening a network connection associated with a given nc-reference after the receipt of an N-DISCONNECT indication in OPENING or RECOVERY state, TTR NC shall be less than TPD-NC by at least the sum of the maximum disconnection and maximum connection propagation delays of the network service;
- b) the TPD-NC timer defines the minimum time a transport entity shall maintain an nc-reference in the PENDING state. A value of 2 min is used for TPD-NC;
- c) the TFR-NC timer defines the minimum time that shall elapse before an entity may reuse a locally allocated nc-reference. A value of 2 min is used for TFR-NC.

#### **B.6.2.5 Association of a received NCM TPDU with a known nc-reference**

When an NCM TPDU is received according to B.6.1 c) and B.6.2 and processed (a transport entity may always elect to process or ignore an NCM TPDU), the NCM TPDU is associated with an existing nc-reference if one of the following holds:

- a) either the three following conditions are met:
  - 1) the reference number received in the NC-REF parameter is the same as the one stored; and
  - 2) the pair of NSAPs addresses in the N-CONNECT indication in which the NCM TPDU was received is the same as those stored with the reference; and
  - 3) the parameter received in the NCM TPDU indicates the same source of allocation as that stored with the nc-reference, as indicated in Table B.3;

TABLE B.3/X.224

**Matching source of nc-reference allocation**

| Stored Source<br>NC-TYPE                                  | Remote | Local |
|---|--------|-------|
| NEW   | S      | D     |
| YOUR  | D      | S     |
| MY  | S      | D     |
| S Same allocation source<br>D Different allocation source |        |       |

- b) or there is no nc-reference known by the transport entity corresponding to B.6.2.5 a)1) above and the following three conditions hold:
- 1) the NC-TYPE parameter has the value NEW;
  - 2) there is an nc-reference, locally allocated, joining the same pair of NSAPs addresses, in OPENING state and having the assignment right defined as follows:
    - assignment right is “my-side” and the RIGHT field of the received NCM TPDU holds the value RA (receiver has assignment right);
    - or assignment right is “remote-sides” and the RIGHT field of the received NCM TPDU holds the value SA (sender has assignment right);
    - or assignment right is “both-sides” and the RIGHT field of the received NCM TPDU holds the value AA (assignment right to all);
  - 3) acceptance of both network connections would result in the establishment of more connections than the transport entity is prepared to support.

An NCM TPDU which is not associated but carries a value different from NEW in the TYPE parameter has to be considered an error.

**B.6.2.6 Collision****B.6.2.6.1 Collision cases**

A collision is detected when:

- a) an NCM TPDU is associated with a known nc-reference (see B.6.2.5) and;
- b) there is an N-CONNECT confirm pending for the network connection used for the nc-reference.

NOTE – In other words a collision is an association with an nc-reference either in OPENING or RECOVERY state.

**B.6.2.6.2 Collision resolution mechanism****B.6.2.6.2.1 Collision winner**

When a collision occurs, one of the two network connections (i.e. the one which is currently used for the nc-reference and the one carrying the NCM TPDU which is associated with the nc-reference) has to be disconnected.

In general the state of the nc-reference determines which network connection shall be disconnected.

However, in the following two cases:

- a) the NCM has been associated with an nc-reference in RECOVERY state according to B.6.2.5 a) and the TYPE parameter has a value different from NEW;
  - NOTE – In this case both ends are in RECOVERY state.
- b) or the NCM has been associated according to B.6.2.5 b).
  - NOTE – In this case both ends are in OPENING state.

The following procedure shall be used to determine if the local transport entity is the collision winner or the collision loser. The local entity is the winner if

- a) the state of the nc-reference is OPENING and the local allocated nc-reference has a lower value (nc-reference to be treated as a 16-bit integer) than the nc-reference of the received NCM TPDU. In the case when both references are equal, both network connections are disconnected, i.e. refused, and both transport entities choose another nc-reference and try (eventually) again;
- b) the state of the nc-reference is RECOVERY and the preference attached to nc-reference is higher than the one contained in the NC-PREF field of the received NCM TPDU;
- c) the state of the nc-reference is RECOVERY and the preference attached to nc-reference is equal to the one contained in the NC-PREF field of the received NCM TPDU and either:
  - 1) the source of allocation of the nc-reference is local and the value of the NC-REC field of the first sent NCM TPDU (i.e. which had NC-TYPE = NEW) was “please do not recover”, or
  - 2) the source of allocation of the nc-reference is remote and the value of the NC-REC field of the first sent NCM TPDU (i.e. which had NC-TYPE = NEW) is “please recover”.

### **B.6.3 NCM confirmation**

When an NCM TPDU has been sent with RIGHT set to RA the transport entity which receives and processes the NCM TPDU shall, if accepting the incoming network connection, transmit an NCMC TPDU in the NS-user data parameter of the N-CONNECT response.

NOTE 1 – The NCMC TPDU is sent only if

- a) The incoming network connection is accepted; and
- b) The received NCM TPDU has the TYPE field set to NEW; and
- c) The received NCM TPDU has the RIGHT field set to RA.

NOTE 2 – This mechanism avoids possible useless freezing of resources (network connection) when the peer-entity ignores an NCM TPDU which gives exclusive assignment.

If an N-CONNECT confirmation is received after having an NCM TPDU with RIGHT set to RA, which does not carry an NCMC TPDU, the initiator shall disconnect the network connection.

NOTE 3 – The absence of NCMC TPDU indicates that the peer-entity did not process the NCM TPDU.

## **B.7 Protocol operation**

### **B.7.1 Receipt of an N-CONNECT indication**

The recipient of an N-CONNECT indication which either does not contain an NCM TPDU or contains an NCM TPDU which it chooses to ignore shall follow the procedures described in the main body of this Recommendation. If the NCM TPDU is to be processed then the transport entity shall apply the procedure for association of NCM TPDU to a known nc-reference (see B.6.2.5). If the NCM TPDU is associated then the transport entity shall apply either the procedure describe in B.7.3 a) or B.7.3 b) or B.7.4.2 b) or B.7.4.3 c) or B.7.4.2 or B.7.5 depending upon the state of the nc-reference. Otherwise the procedure in B.7.2 applies.

### **B.7.2 Passive network connection establishment with NCM TPDU**

The transport entity may either decide to refuse the incoming network connection (i.e. issue an N-DISCONNECT request) or accept the network connection.

If the transport entity elects to accept the network connection it shall

- a) issue an N-CONNECT response; if the RIGHT field of the received NCM TPDU contains the value RA the NCMC TPDU shall be transmitted in the NS-user data parameter of the N-CONNECT response;
- b) note the nc-reference and the pair of NSAPs;
- c) note that the nc-reference has been remotely allocated;

- d) note the assignment rights as my-side if RA has been received in the RIGHT field of the NCM TPDU, as remote-side if SA has been received or both-sides if AA has been received;
- e) put the nc-reference into the OPEN state and use it for assignment or reassignment if it is (one of) the owner(s).

### B.7.3 Active network connection establishment with NCM TPDU

The transport entity, which elects to use the NCMS procedure when opening a network connection shall send an N-CONNECT request with the UN TPDU (see ITU-T Rec. X.264 | ISO/IEC 11570) and NCM TPDU contained in the NS-user data parameter. The NCM TPDU parameters are set as follows:

- NC-REF contains the selected reference which shall neither be used for any other network connection between the same pair of NSAPs nor frozen;
- NC-TYPE is set to NEW;
- NC-RIGHT is set to SA, RA, or AA;
- NC-PREF is set to low, medium or high according to the preference of the initiator to keep this connection in case of collision.

NOTE – The selection of this value can be based on the knowledge of the correspondence between the expected QOS, the cost when reverse charging is used and other optimization considerations.

The initiator shall store the nc-reference together with the pair of NSAPs to be joined by the network connection being established, the value of the NC-PREF parameter sent, the ownership of the network connection and the source of the nc-reference (locally allocated in this case).

The state of the nc-reference shall be set to OPENING.

The initiator shall wait for an N-CONNECT confirm to complete the establishment. If the assignment rights are “remote side” (i.e. an NCM TPDU with the RIGHT parameter having the value RA was sent) the received N-CONNECT confirm shall contain an NCMC TPDU in its user data parameter otherwise the transport entity shall disconnect the network connection. If one of the following cases occurs, the initiator shall perform the action specified:

- a) if an NCM TPDU is received and associated according to B.6.2.5 b) (TYPE = NEW) the transport entity shall apply one of the following:
  - 1) if the local transport entity is the winner (see B.6.2.6.2) the incoming network connection is disconnected (i.e. an N-DISCONNECT request is sent in response to the incoming N-CONNECT indication) and the nc-reference remains in the OPENING state;
  - 2) if the local entity is the loser (see B.6.2.6.2) the network connection which was local opened is disconnected and the incoming network connection is accepted (i.e. an N-CONNECT response is issued). If the received NCM TPDU contains the value RA in its RIGHT field the NCMC TPDU shall be transmitted in the NS-user data parameter of the N-CONNECT response. The nc-reference which has been locally allocated is frozen for a TFR-NC period of time (and then released) and the transport entity keeps track of the nc-reference contained in the NC-REF parameter of the incoming NCM TPDU as remotely allocated and in OPEN state. The network connection is considered as open and ready for use as described in the main body of this Recommendation, according to the assignment rights.  
  
Any transport connections assigned to the disconnected connection shall be reassigned:
- b) if an NCM TPDU is received with a TYPE parameter different from NEW and is associated, the transport entity shall
  - 1) issue an N-DISCONNECT request for the network connection for which the N-CONNECT confirm is awaited;
  - 2) respond to the incoming N-CONNECT indication by an N-CONNECT response;
  - 3) place the nc-reference in the OPEN state and consider the network connection as ready for assignment or reassignment according to the assignment rights;
- c) if an N-DISCONNECT indication is received, the transport entity may decide either to give up or try to reopen a network connection by issuing an N-CONNECT request containing an NCM TPDU (see B.6.1.2 and B.6.2) which is a copy of the previously sent NCM TPDU, except that the NC-PREF parameter may

be different. The decision whether a new network connection has to be opened or not is a local matter subject to the following constraints:

- 1) when the first N-DISCONNECT indication is received, the entity shall start its TTR-NC timer and stop it when receiving the corresponding N-CONNECT confirm or N-CONNECT indication carrying an NCM TPDU which is associated and processed as described above. When the timer runs out the transport entity shall not try to open a network connection again if a new N-DISCONNECT indication is received;
- 2) if the network connection is intended to be used for transport connections allowing recovery, the network connection has to be reopened in accordance with the agreed upon quality of service of the supported transport connection(s).

When recovery is not performed or has stopped (i.e. a new N-DISCONNECT is received and TTR-NC has run out) the nc-reference of the network connection is placed in a PENDING state for a TPD-NC period of time. During this period the transport entity may receive an incoming NCM TPDU having this nc-reference (see B.7.4.3).

#### **B.7.4 Network connection recovery**

##### **B.7.4.1 Receipt of an N-DISCONNECT indication**

When a network connection which was established using NCMS is disconnected (i.e. an N-DISCONNECT is received) the transport entity shall either

- a) elect not to reopen the network connection, place the nc-reference in the PENDING state for a TPD-NC period of time, and apply the procedure described in B.7.4.3; or
- b) attempt to reopen the network connection by following the procedure described in B.7.4.2.

Alternative b) is subject to the same constraints as described in B.7.3 c).

In all cases the transport entity shall apply the procedure corresponding to the receipt of an N-DISCONNECT indication to all transport connections assigned to the network connection.

##### **B.7.4.2 Active recovery procedure**

The transport entity shall open a network connection by putting the nc-reference into the RECOVERY state and sending an NCM TPDU in the NS-user data parameter of the N-CONNECT request according to B.6.1 c) and B.6.2 with the following parameters:

- a) NC-REF is set to the value of the nc-reference associated with the network connection;
- b) NC-TYPE is set to MY if the nc-reference was locally allocated, to YOURS if the nc-reference was remotely allocated;
- c) NC-PREF is set to the desired value (see B.7.3);
- d) NC-RIGHT may take any value;  
NOTE – NC-RIGHT is not significant in an NCM TPDU performing recovery.
- e) NC-REC is set to the desired value.

The transport entity shall then apply one of the following:

- a) if an N-DISCONNECT is received apply B.7.4.1;
- b) if an NCM TPDU with type NEW is received and associated the incoming network connection is rejected;
- c) if an NCM with type different from NEW is received the collision winner is determined according to B.6.2.6.2 and:
  - 1) if the transport entity is the winner the incoming network connection is rejected, or
  - 2) if the transport entity is the loser the incoming network connection is accepted (i.e. send an N-CONNECT response), the nc-reference is placed in the OPEN state and is ready for use according to the assignment rights. The network connection for which an N-CONNECT confirm was awaited is disconnected by issuing an N-DISCONNECT request.

### **B.7.4.3 Passive recovery procedure**

If an N-CONNECT indication is received which carries an NCM TPDU, which is associated to the nc-reference, the transport entity shall send an N-CONNECT response (with an NCMC TPDU in the NS-user data parameter if the received NCM TPDU is of type NEW and has the right field set to the value RA) and put the nc-reference into the OPEN state and consider it as ready for assignment according to the assignment rights.

If the TPD-NC timer expires and if the nc-reference was remotely allocated, then the transport entity does not keep track of it any longer; if the nc-reference was locally allocated, then the transport entity shall not reuse the reference until a TFR-NC period of time has elapsed.

### **B.7.5 Remotely initiated recovery**

When a transport entity receives an NCM TPDU which is associated with an nc-reference in OPEN state it shall

- a) accept the incoming network connection and issue an N-CONNECT response; if the received NCM TPDU is of type NEW and has the RIGHT field set to RA, the NCMC TPDU shall be transmitted in the NS-user data parameter of the N-CONNECT response;
- b) issue an N-DISCONNECT request for the network connection which was associated with the nc-reference;
- c) apply to all transport connections assigned to this network connection the procedure defined in the main body of this Recommendation for processing an N-DISCONNECT indication.

### **B.7.6 Optimization principles**

#### **B.7.6.1 Use of the NC-REC indicator**

Although the recovery protocol is symmetrical, it should be noted that a transport entity is always allowed not to initiate recovery by putting the nc-reference into the PENDING state.

NOTE 1 – Not initiating recovery is equivalent to having a value of zero for the TTR-NC timer.

In order to avoid unnecessary recovery being performed or recovery being delayed the NC-REC field of the NCM TPDU should be set as follows:

- a) 0 (please do not recover): indicates that the sender does not rely on recovery being performed by the receiver, and intends to recover even if not required for its own need;
- b) 1 (please recover): indicates that the sender expects a recovery to be done by the receiver, and does not intend to recover if it does not need it.

When the nc-reference is in the OPEN state, the NC-REC field of the associated NCM TPDU gives both entities a view of the recovery intention of the partner.

When a transport entity has to initiate recovery, i.e. an N-DISCONNECT indication has been received in OPENING or RECOVERY state and TTR-NC has not run out it is recommended that

- a) if the entity has received an NCM TPDU with the NC-REC field set to “please recover”, it should try to recover even when this is not necessary for its own assignment needs;
- b) if the entity has received an NCM TPDU with the NC-REC field set to “please don’t recover”, it should not initiate recovery if this is not required for its own assignment needs;
- c) if the entity has sent an NCM TPDU with the NC-REC field set to “please don’t recover”, it should initiate recovery even if not required for its own assignment needs;
- d) if the entity has sent an NCM TPDU with the NC-REC field set to “please recover”, it should not initiate recovery if not required for its own assignment needs.

NOTE 2 – As a local choice, it is possible to implement a symmetrical recovery mechanism by setting the NC-REC field to the value “please recover” and initiating recovery even though not required for the local assignment needs.

### B.7.6.2 Use of NS-user data parameter in N-DISCONNECT primitive

The reason code in the N-DISCONNECT primitive does not adequately specify enough information to completely optimize the connection recovery mechanisms since the values defined in the network service (CCITT Rec. X.213 | ISO/IEC 8348) do not distinguish between the cases where recovery is desirable immediately or not and do not provide adequate diagnostic information. Thus the NS-user data parameter of the N-DISCONNECT request may be used and may contain a DIAG TPDU.

When a network connection is no longer needed, it is recommended that only the owner(s) may disconnect it, and put a DIAG TPDU with code 1 into the NS-user data parameter of the N-DISCONNECT request primitive.

### B.7.7 Releasing a network connection

Either entity may release a network connection at any time by issuing an N-DISCONNECT request. It is recommended to use the DIAG TPDU in order to optimize this procedure as described in B.7.6.2.

If the remote transport entity has assignment rights the nc-reference shall be placed in the PENDING state after the network connection has been released.

If the remote transport entity does not have assignment rights the nc-reference may, as a matter of local choice:

- a) be put in the PENDING state; or
- b) if locally allocated be frozen; or
- c) if remotely allocated be made unknown.

## B.8 Structure and encoding of TPDU

### B.8.1 Validity

Table B.4 specifies the TPDU valid for this annex.

TABLE B.4/X.224

#### TPDU codes

| Name   | Code      |
|--|-----------|
| NCM, network connection management               | 0000 0010 |
| DIAG, diagnostic                                 | 0000 0011 |
| NCMC, network connection management confirmation | 0000 0100 |

### B.8.2 Structure

The structure is defined in 13.2.

### B.8.3 Network connection management (NCM) TPDU

#### B.8.3.1 Structure

| 1  | 2                | 3      | 4                  | 5                            | 6 |
|----|------------------|--------|--------------------|------------------------------|---|
| LI | NCM<br>0000 0010 | NC-REF | NC-TYPE<br>NC-PREF | NC-COL<br>NC-REC<br>NC-RIGHT |   |

### B.8.3.2 LI

See 13.2.1.

#### B.8.3.3 Fixed part

The fixed part shall contain

- a) NCM: NCM TPDU code: 0000 0010;
- b) NC-REF: the nc-reference;
- c) NC-TYPE: indicates the type of the nc-reference which is sent. NC-TYPE consists of bits 8 and 7 of octet 5 and may have the value 00 (NEW), 01 (MY) or 10 (YOURS), Value 11 is reserved;
- d) NC-PREF: indicates the preference the initiator has to keep the network connection in case of collision. NC-REF is bits 6-1 of octet 5:  
 000000 highest preference,  
 000001 medium,  
 000011 lowest preference;
- e) NC-COL: indicates the collision algorithm to be used. NC-COL is bit 8 of octet 6. Only one value is defined (0): resolution of collision when N-CONNECT indication is received;
- f) NC-REC: indicates the recovery optimization option. NC-REC is bit 7 of octet 6:  
 0 please do not recover,  
 1 please recover;
- g) NC-RIGHT: indicates the kind of right of use given by the entity to its peer. NC-RIGHT is bits 6-1 of octet 6:  
 000001 SA,  
 000010 RA,  
 000011 AA.

#### B.8.3.4 Variable part

There is no variable part.

### B.8.4 Diagnostic (DIAG) TPDU

This TPDU is only transferred in the NS-user data parameter of an N-DISCONNECT. It provides diagnostic information. Sending and/or processing this TPDU is optional.

#### B.8.4.1 Structure

| 1  | 2                 | 3    |
|----|-------------------|------|
| LI | DIAG<br>0000 0011 | CODE |

### B.8.4.2 LI

See 13.2.1.

#### B.8.4.3 Fixed part

The fixed part shall contain

- a) DIAG: DIAG TPDU code 0000 0011;



- b) CODE: indicates the reason for disconnecting the network connection. The following values shall be used:
- 0 – Collision detection resolution
  - 1 – Network connection no longer needed
  - 2 – Unrecognized NC-REF (do not try to recover with this NC-REF)
  - 3 – Network connection cannot be accepted (temporary congestion)
  - 4 – A new network connection cannot be accepted again (long-term congestion or shutdown in progress)

#### B.8.4.4 Variable part

There is no variable part.

### B.8.5 Network connection management confirm (NCMC) TPDU

#### B.8.5.1 Structure

| 1  | 2                 |
|----|-------------------|
| LI | NCMC<br>0000 0100 |

#### B.8.5.2 LI

See 13.2.1.

#### B.8.5.3 Fixed part

The fixed part shall contain the NCMC TPDU code: 0000 0100.

#### B.8.5.4 Variable part

There is no variable part.

## B.9 Conformance

**B.9.1** When initiating a network connection a transport entity shall either

- a) not use the NS-user data parameter of N-CONNECT request primitive and operate using the protocol of the main body of this Recommendation on this network connection; or
- b) include in the NS-user data parameter of the N-CONNECT request a UN TPDU (see ITU-T Rec. X.264 | ISO/IEC 11570) with the PRT-ID field set to the value 01 followed by an NCM TPDU and operate the NCMS procedure together with those specified in the main body of this Recommendation.

**B.9.2** When processing an N-CONNECT indication, a transport entity shall either

- a) operate using the protocol of the main body of this Recommendation if no user data is present or if it is not claimed that the implementation supports the NCMS procedure; or
- b) operate using the protocol of the main body of this Recommendation together with the network connection management procedure if the UN [with the PRT-ID field set to the value 01 (see ITU-T Rec. X.264 | ISO/IEC 11570)] and NCM TPDU are present; or
- c) operate using the protocol of the main body of this Recommendation but ignore the NCM TPDU if the UN [with the PRT-ID field set to the value 01 (see ITU-T Rec. X.264 | ISO/IEC 11570)] and NCM TPDU are present.

## B.10 State Table

The state Tables B.5, B.6, B.7, B.8, B.9 and B.10 define the states of a network connection reference as maintained by a single transport entity obeying the procedure of this annex. Due to failures and recoveries of network connections this reference may over a period of time be associated with many network connections, one at a time. When an NCM TPDU is received the association procedure (see B.6.2.5) is applied first.

TABLE B.5/X.224

### Events

| Event          | Description  |
|----------------|--|
| NCMNEWrec      | An N-CONNECT indication containing an NCM TPDU with NC-TYPE = NEW is received        |
| NCMNOTNEWrec   | An N-CONNECT indication containing an NCM TPDU with NC-TYPE not NEW is received      |
| NDISind        | An N-DISCONNECT indication   |
| Collision      | A collision in the opening state as a result of association as described in 6.3.5 b) |
| TPD-NCexp      | The timer TPD-NC expires   |
| TTR-NRexp      | The timer TTR-NC expires   |
| Local decision | The transport entity may choose to initiate this transition                          |
| Any TPDU       | Receipt of any TPDU on the network connection  |
| NCONconf       | An C-CONNECT confirmation  |

TABLE B.6/X.224

**Actions**

| Action    | Description  |
|-----------|--|
| NCONreq   | Issue an N-CONNECT request to the network service  |
| NCMNEW    | Send an NCM TPDU with the NCONreq with NC-TYPE = NEW and locally allocated nc-reference  |
| NCMNOTNEW | Send an NCM TPDU with the NCONreq with NC-TYPE set to show the original source of allocation of the reference  |
| NDISreq   | Issue an N-DISCONNECT request to the network service   |
| NCONresp  | Issue an N-DISCONNECT response to the network service  |
| [1]       | Start TPD-NC timer   |
| [2]       | Start TTR-NC timer if not already running  |
| [3]       | Freeze the nc-reference for TFR-NC if locally allocated  |
| [4]       | The remotely initiated connection has been the winner. Re-assign any TCs from the loser and process the incoming NCM TPDU as an NCMNEW in the CLOSED state for the winning reference |
| [5]       | Stop TTR-NC if running otherwise remove information that it has expired  |
| [6]       | Stop TPD-NC  |
| [7]       | Record with the nc-reference that a TPDU has been received   |
| [8]       | Store information that TTR-NC has expired  |
| [9]       | If the received NCM TPDU has the RIGHT field set to the value RA, a NCMC TPDU is transmitted in the NS-user data parameter of N-CONNECT response                                     |

TABLE B.7/X.224

**Predicates**

| Predicate | Description  |
|-----------|--|
| P1        | Incoming network connection unacceptable or the local entity is the winner of a collision    |
| P2        | Remote transport entity is not an owner of the network connection and local choice           |
| P3        | Local choice not to recover or TTR-NC has previously expired                                 |
| P4        | Local choice not to recover  |
| P5        | The remotely initiated network connection is the winner of collision resolution              |
| P6        | A TPDU has been received on a network connection associated with this nc-reference [see (7)] |
| P7        | Assignment rights are "remote side" and N-CONNECT confirmation does not carry an NCMC TPDU   |

TABLE B.8/X.224

**Notes**

| Note | Description  |
|------|--|
| (1)  | DIAG TPDU with CODE = 0 may be sent                                      |
| (2)  | The new connection is retained and the old connection disconnected       |
| (3)  | Repeat the previous NCM except that NC-PREF may be different             |
| (4)  | This is a protocol error   |
| (5)  | Discard the “loser” after collision resolution                           |
| (6)  | The incoming network connection is disconnected and the old one retained |

TABLE B.9/X.224

**States**

| Note     | Description  |
|----------|--|
| CLOSED   | Network connection is closed   |
| OPENING  | Network connection requested but not yet confirmed                         |
| OPEN     | Network connection is open   |
| RECOVERY | Attempting recovery of a failed network connection                         |
| PENDING  | A non-owner of the network connection is waiting for recovery by the owner |

TABLE B.10/X.224

State Table

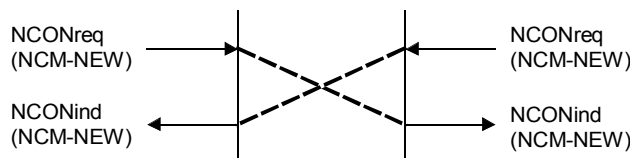
| State          | CLOSED  | OPENING   | OPEN  | RECOVERY   | PENDING  |
|----------------|---|---|---|--|--|
| Event          |   |   |   |  |  |
| NCMNEWrec      | P1: NDISreq CLOSED;<br>not P1: [9]<br>NCONresp OPEN |   | P6: (4) NDISreq OPEN;<br>not P6: [9]<br>NCONresp NDISreq (2) OPEN                       | NDISreq (6) RECOVERY   | P6: (4) NDISreq PENDING;<br>not P6: [9]<br>NCONresp [6] OPEN |
| NCMNOTNEWrec   | (4) NDISreq   | NCONresp (2) NDISreq OPEN   | NCONresp (2) NDISreq OPEN   | P1: NDISreq (1) RECOVERY;<br>not P1: NCONresp NDISreq (2) (5) OPEN | NCONresp [6] OPEN  |
| Local decision | NCMNEW OPENING                                      |   |   |  |  |
| NDISind        |   | Not P3: [2]<br>NCMNEW (3) OPENING;<br>P2 and P3: [3] CLOSED;<br>(not P2) and P3: [1, 5] PENDING | Not P4: NCMNOTNEW [2] RECOVERY;<br>P2 and P4: [3] CLOSED;<br>P4 and not P2: [1] PENDING | Not P3: NCMNOTNEW RECOVERY;<br>P3: [1, 5] PENDING                  |  |
| Collision      |   | P1: NDISreq (1) OPENING;<br>non P1: NDISreq (2) [4, 3] CLOSED                                   |   |  |  |
| TPD-NCexp      |   |   |   |  | [3] CLOSED   |
| Any TPDU       |   |   | [7] OPEN  |  |  |
| TTR-NCexp      |   | [8] OPENING   |   | [8] RECOVERY   |  |
| NCONconf       |   | P7: [3] CLOSED;<br>not P7: [5] OPEN   |   | [5] OPEN   |  |

**B.11 Diagram for NCMS protocol operation**

This clause provides some tutorial information by giving examples of collision cases (see B.11.1) and remotely initiated recovery (see B.11.2). This clause is informative.

**B.11.1 Collision case**

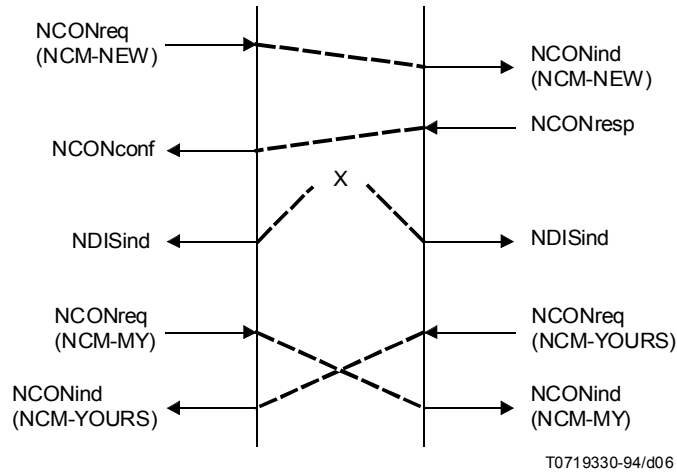
**B.11.1.1 Both ends detect a collision on OPENING state with TYPE = NEW**



T0719320-94/d05

The references are different but both ends have decided to associate the received NCM TPDU according to B.6.2.5 b).

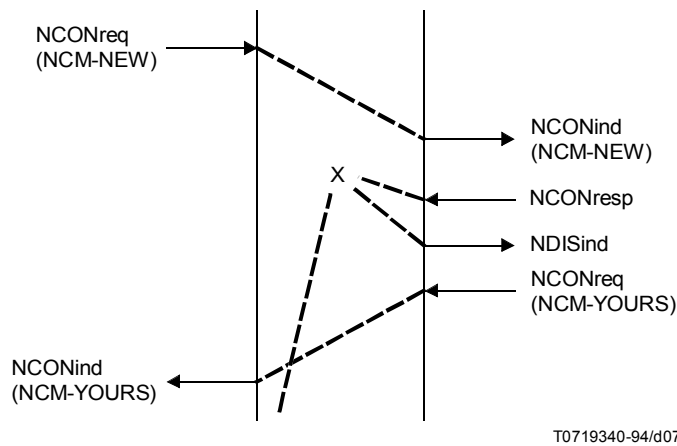
**B.11.1.2** Both ends detect a collision in RECOVERY state



Both entities will use the collision resolution algorithm and one of the two network connections will be disconnected.

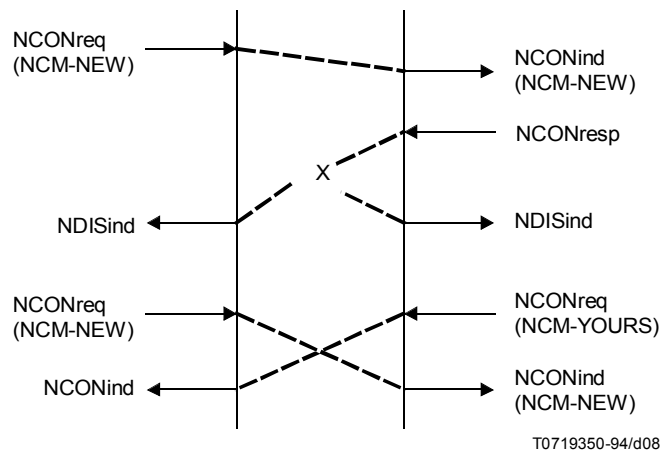
**B.11.1.3** The initiator detects a collision in OPENING state with TYPE different from NEW

**B.11.1.3.1** The other end is in RECOVERY state



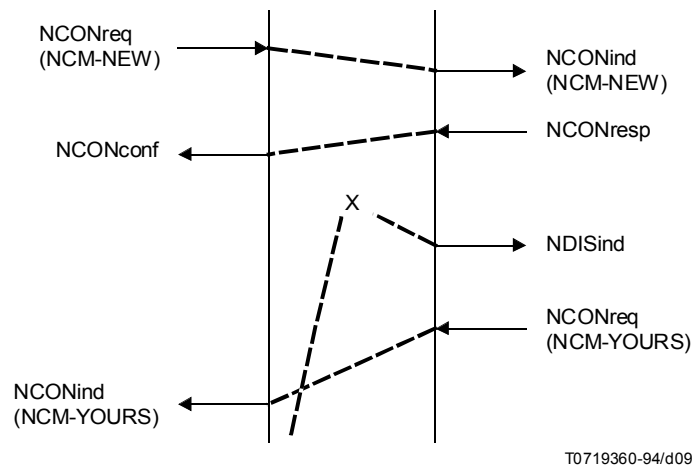
The entity in OPENING state (left side) accepts the incoming network connection and disconnects the PENDING one.

**B.11.1.3.2** The other end detects a collision in RECOVERY state

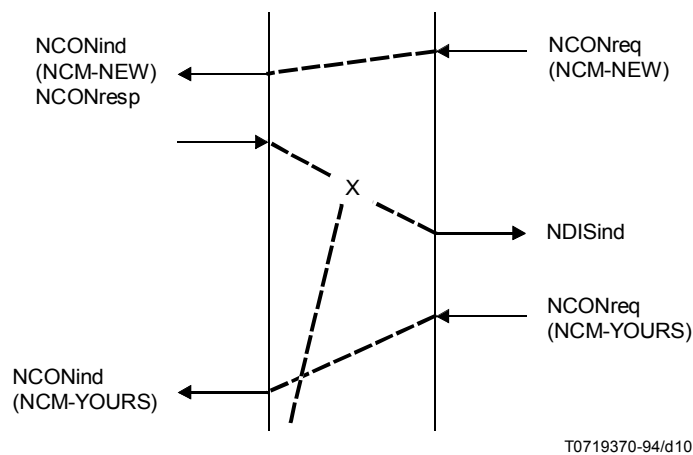


Both entities disconnect the network connection initiated by the entity on the left.

**B.11.2** Remotely initiated recovery



The entity on the left detects an incoming network connection in OPEN state and disconnects the old connection.



The entity on the left detects an incoming network connection in OPEN state and disconnects the old connection.

**Annex C<sup>2)</sup>**  
(normative)

**PICS Proforma**

(This annex forms an integral part of this Recommendation)

**C.1 General**

**C.1.1 Symbols used**

Status symbols:

M Mandatory.

O Optional to implement. If implemented the feature may or may not be used.

O.<n> Optional but support of at least one of the group of options labelled by the same numeral <n> in this PICS proforma is required.

<index>: This predicate symbol means that the status following it applies only when the PICS states that the feature identified by the index is supported. In the simplest case, <index> is the identifying tag of a single PICS item. <index> may also be a Boolean expression composed of several indices.

<index>.: When this group predicate is true the associated clause should be completed.

Support symbols:

Yes Supported.

No Not supported.

N/A Not applicable.

**C.1.2 Instructions for completing the PICS proforma**

The main part of the PICS proforma is a fixed-format questionnaire divided into a number of clauses. Answers to the questionnaire are to be provided in the rightmost column either by simply marking an answer to indicate a restricted choice (such as Yes or No) or by entering a value of a range of values or entering what action is taken.

**C.2 Identification**

**C.2.1 Implementation identification**

|  |  |
|--|--|
| Supplier   |  |
| Contact point for queries about the PICS   |  |
| Implementation Name(s) and Version(s)  |  |
| Other information necessary for full identification – e.g. name(s) and version(s) of machines and/or operating systems; System Name(s)                                   |  |
| NOTES  |  |
| 1 Only the first three items are required for all implementations; other information may be completed as appropriate in meeting the requirement for full identification. |  |
| 2 The terms Name and Version should be interpreted appropriately to correspond with a supplier's terminology (e.g. Type, Series, Model)                                  |  |

---

<sup>2)</sup> Copyright release for PICS proforma

Users of this Recommendation may freely reproduce the PICS proforma in this annex so that it can be used for its intended purpose and may further publish the completed PICS.



### C.2.2 Protocol Summary

|  |  |
|--|--|
| Identification of protocol specification   | ISO/IEC 8073:1992 (E)<br>ITU-T Rec. X.224 Reference Number: X.224 (1993) |
| Identification of Amendments and Corrigenda to this PICS proforma which have been completed as part of this PICS | ITU-T Rec. X.224 (1993)   ISO/IEC 8073:1992                              |
| Protocol Version(s) supported  | Version 1  |
| Have any Exception items been required?  | No [ ] Yes [ ]   |
| (The answer Yes means that the implementation does not conform to ITU-T Rec. X.224 (1993)   ISO/IEC 8073:1992)   |  |

|                   |  |
|-------------------|--|
| Date of statement |  |
|-------------------|--|

### C.3 Indices used in this annex

|             |          |
|-------------|----------|
| A .....     | C.6.1    |
| C .....     | C.6.2    |
| C4L .....   | C.6.2    |
| CCT .....   | C.5      |
| DRCC .....  | C.14     |
| DRCR .....  | C.14     |
| DRDR .....  | C14      |
| D1ICC ..... | C.13.1   |
| D1ICR ..... | C.13.1   |
| D1IDR ..... | C.13.1   |
| D2ICC ..... | C.13.2   |
| D2ICR ..... | C.13.2   |
| D2IDR ..... | C.13.2   |
| D3ICC ..... | C.13.3   |
| D3ICR ..... | C.13.3   |
| D3IDR ..... | C.13.3   |
| D4ICC ..... | C.13.4   |
| D4ICR ..... | C.13.4   |
| D4IDR ..... | C.13.4   |
| IC .....    | C.11.1.2 |
| ICR .....   | C.11.2   |
| IR .....    | C.8      |
| I0CC .....  | C.11.3   |
| I0CR .....  | C.11.3   |
| I0DR .....  | C.11.3   |
| I1CC .....  | C.11.4   |
| I1CR .....  | C.11.4   |
| I1DR .....  | C.11.4   |
| I1DT .....  | C.11.4   |
| I1ER .....  | C.11.4   |
| I2CC .....  | C.11.5   |
| I2CR .....  | C.11.5   |
| I2DR .....  | C.11.5   |
| I2ER .....  | C.11.5   |
| I3CC .....  | C.11.6   |
| I3CR .....  | C.11.6   |
| I3DR .....  | C.11.6   |

|              |          |
|--------------|----------|
| I3DT .....   | C.11.6   |
| I3ER .....   | C.11.6   |
| I4AK .....   | C.11.7   |
| I4CC .....   | C.11.7   |
| I4CR .....   | C.11.7   |
| I4DR .....   | C.11.7   |
| I4DT .....   | C.11.7   |
| I4ER .....   | C.11.7   |
| ISO .....    | C.5      |
| N .....      | C.7      |
| NAC .....    | C.15.1   |
| NEF .....    | C.15.4   |
| NC .....     | C.15.1   |
| NUC .....    | C.15.6   |
| NUF .....    | C.15.7   |
| OT .....     | C.17     |
| PE .....     | C.16.1   |
| PE4L .....   | C.16.1   |
| RC .....     | C.15.2   |
| RC4a .....   | C.15.2   |
| RN .....     | C.12.1.2 |
| ROA .....    | C.15.11  |
| RR .....     | C.16.2   |
| R4AKch ..... | C.12.2   |
| R4CCch ..... | C.12.2   |
| R4DCch ..... | C.12.2   |
| R4DRch ..... | C.12.2   |
| R4DTch ..... | C.12.2   |
| R4EAch ..... | C.12.2   |
| R4EDch ..... | C.12.2   |
| R4ERch ..... | C.12.2   |
| SER .....    | C.10     |
| SER4L .....  | C.10     |
| SN .....     | C.10     |
| ST .....     | C.10     |
| TA .....     | C.17     |
| TED .....    | C.15.5   |
| TS .....     | C.15.3   |
| T0F .....    | C.9.1    |
| T0S .....    | C.15.3   |
| T1F .....    | C.9.2    |
| T1S .....    | C.15.3   |
| T2F .....    | C.9.3    |
| T2S .....    | C.15.3   |
| T3F .....    | C.9.4    |
| T3S .....    | C.15.3   |
| T4F .....    | C.9.5    |
| T4S .....    | C.15.3   |
| UI .....     | C.16.3   |
| UNED .....   | C.15.9   |
| UNRC .....   | C.15.8   |
| USA .....    | C.15.10  |

#### C.4 Based standard/recommendation conformance

|  |     |    |
|--|-----|----|
| Does the implementation claim conformance to ISO/IEC 8073?     | Yes | No |
| Does the implementation claim conformance to ITU-T Rec. X.224? | Yes | No |

#### C.5 General statement of conformance

|   |   |     |    |
|---|---|-----|----|
| ISO   | Are all mandatory features of ISO/IEC 8073 implemented?     | Yes | No |
| CCT   | Are all mandatory features of ITU-T Rec. X.224 implemented? | Yes | No |
| NOTE – Answering “No” to this question indicates non-conformance to the Recommendation/International Standard |   |     |    |

#### C.6 Protocol implementation

##### C.6.1 Annex B – NCMS

| Index |  | References | Status | Support |
|-------|--|------------|--------|---------|
| A1    | Network connection management procedures | Annex B    | O      | Yes No  |

##### C.6.2 Classes implemented

| Index | Class                       | References | Status              | Support |
|-------|-----------------------------|------------|---------------------|---------|
| C0    | Class 0                     | 14         | ISO:O.1<br>CCT:M    | Yes No  |
| C1    | Class 1                     | 14         | C0:O                | Yes No  |
| C2    | Class 2                     | 14         | ISO:O.1<br>CCT:O    | Yes No  |
| C3    | Class 3                     | 14         | C2:O                | Yes No  |
| C4    | Class 4 operation over CONS | 14         | C2:O                | Yes No  |
| C4L   | Class 4 operation over CLNS | 14         | ISO:C2:O<br>CCT:N/A | Yes No  |

#### C.7 NCMS functions

| Index | Item                               | References        | Status | Support |
|-------|------------------------------------|-------------------|--------|---------|
| N2    | Network connection management      | B.6.2.1           | O      | Yes No  |
| N3    | Diagnostic                         | B.7.6.2,<br>B.7.7 | O      | Yes No  |
| N4    | Active network connection recovery | B.7.4.2           | O      | Yes No  |

The following is mandatory if the predicate is true

| Index | Item  | References | Status         | Support |
|-------|---|------------|----------------|---------|
| N5    | Passive network connection recovery   | B.7.4.3    | N2 OR N4:<br>M | Yes No  |
| N6    | Is an NCM TPDU with assignment right set to RA always rejected with N-DISCONNECT request? | B.6.3      | O              | Yes No  |

### C.8 Initiator/responder capability for protocol classes 0 - 4

| Index |                       | References | Status | Support |
|-------|-----------------------|------------|--------|---------|
| IR1   | Initiating CR TPDU    | 14.4 a)    | O.2    | Yes No  |
| IR2   | Responding to CR TPDU | 14.4 a)    | O.2    | Yes No  |

### C.9 Supported functions

#### C.9.1 Supported functions for class 0 (C0:)

The following functions are mandatory if class 0 is supported

| Index | Function   | References | Status | Support |
|-------|--|------------|--------|---------|
| T0F1  | Assignment to network connection when operating over CONS              | 6.1.1      | M      | Yes     |
| T0F2  | TPDU transfer  | 6.2        | M      | Yes     |
| T0F3  | Segmenting   | 6.3        | M      | Yes     |
| T0F4  | Reassembling   | 6.3        | M      | Yes     |
| T0F5  | Connection establishment   | 6.5        | M      | Yes     |
| T0F6  | Connection refusal   | 6.6        | M      | Yes     |
| T0F7  | Normal release when operating over CONS (implicit)                     | 6.7.1      | M      | Yes     |
| T0F8  | Error release when operating over CONS                                 | 6.8        | M      | Yes     |
| T0F9  | Association of TPDU with Transport connection when operating over CONS | 6.9.1      | M      | Yes     |
| T0F10 | Treatment of protocol errors when operating over CONS                  | 6.22.1     | M      | Yes     |

### C.9.2 Supported functions for class 1 (C1::)

The following functions are mandatory if class 1 is supported

| Index | Function  | References | Status | Support |
|-------|---|------------|--------|---------|
| T1F1  | Assignment to network connection when operating over CONS                                 | 6.1.1      | M      | Yes     |
| T1F2  | TPDU transfer   | 6.2        | M      | Yes     |
| T1F3  | Segmenting  | 6.3        | M      | Yes     |
| T1F4  | Reassembling  | 6.3        | M      | Yes     |
| T1F5  | Separation  | 6.4        | M      | Yes     |
| T1F6  | Connection establishment  | 6.5        | M      | Yes     |
| T1F7  | Connection refusal  | 6.6        | M      | Yes     |
| T1F8  | Normal release when operating over CONS (explicit)  | 6.7.1      | M      | Yes     |
| T1F9  | Association of TPDU's with Transport connections when operating over CONS                 | 6.9.1      | M      | Yes     |
| T1F10 | Data TPDU numbering (normal)  | 6.10       | M      | Yes     |
| T1F11 | Expedited data transfer when operating over CONS (Network normal)                         | 6.11.1     | M      | Yes     |
| T1F12 | Reassignment after failure when operating over CONS                                       | 6.12       | M      | Yes     |
| T1F13 | Retention and acknowledgement of TPDU's<br>Retention until acknowledgement of TPDU's (AK) | 6.13.4.1   | M      | Yes     |
| T1F14 | Resynchronization   | 6.14       | M      | Yes     |
| T1F15 | Frozen references   | 6.18       | M      | Yes     |
| T1F16 | Treatment of protocol errors when operating over CONS                                     | 6.22.1     | M      | Yes     |

The following functions are optional if class 1 is supported

| Index | Function  | References | Status          | Support |
|-------|---|------------|-----------------|---------|
| T1F17 | Concatenation   | 6.4        | O               | Yes No  |
| T1F18 | Expedited data transfer when operating over CONS (Network expedited)      | 6.11.1     | O               | Yes No  |
| T1F19 | Retention and acknowledgement of TPDU's<br>Confirmation of Receipt        | 6.13.4.2   | not T1F20:<br>O | Yes No  |
| T1F20 | Retention and acknowledgement of TPDU's<br>Use of request acknowledgement | 6.13.4.3   | not T1F19:<br>O | Yes No  |

### C.9.3 Supported functions for class 2 (C2::)

The following functions are mandatory if class 2 is supported

| Index | Function  | References | Status | Support |
|-------|---|------------|--------|---------|
| T2F1  | Assignment to network connection when operating over CONS               | 6.1.1      | M      | Yes     |
| T2F2  | TPDU transfer   | 6.2        | M      | Yes     |
| T2F3  | Segmenting  | 6.3        | M      | Yes     |
| T2F4  | Reassembling  | 6.3        | M      | Yes     |
| T2F5  | Separation  | 6.4        | M      | Yes     |
| T2F6  | Connection establishment  | 6.5        | M      | Yes     |
| T2F7  | Connection refusal  | 6.6        | M      | Yes     |
| T2F8  | Normal release when operating over CONS (explicit)                      | 6.7.1      | M      | Yes     |
| T1F9  | Error release when operating over CONS                                  | 6.8        | M      | Yes     |
| T2F10 | Association of TPDU with Transport connections when operating over CONS | 6.9.1      | M      | Yes     |
| T2F11 | Data TPDU numbering (normal)  | 6.10       | M      | Yes     |
| T2F12 | Expedited data transfer when operating over CONS (Network normal)       | 6.11.1     | M      | Yes     |
| T2F13 | Demultiplexing when operating over CONS                                 | 6.15       | M      | Yes     |
| T2F14 | Explicit flow control (with)  | 6.16       | M      | Yes     |
| T2F15 | Treatment of protocol errors when operating over CONS                   | 6.22.1     | M      | Yes     |
| T2F16 | Multiplexing when operating over CONS                                   | 6.15       | M      | Yes     |

The following functions or elements of procedure are optional if class 2 is supported

| Index | Function                        | References | Status | Support |
|-------|---------------------------------|------------|--------|---------|
| T2F17 | Concatenation                   | 6.4        | O      | Yes No  |
| T2F18 | Data TPDU numbering (extended)  | 6.10       | O      | Yes No  |
| T2F19 | Explicit flow control (without) | 6.16       | O      | Yes No  |

#### C.9.4 Supported functions for class 3 (C3::)

The following functions are mandatory if class 3 is supported

| Index | Function  | References | Status | Support |
|-------|---|------------|--------|---------|
| T3F1  | Assignment to network connection when operating over CONS                                 | 6.1.1      | M      | Yes     |
| T3F2  | TPDU transfer   | 6.2        | M      | Yes     |
| T3F3  | Segmenting  | 6.3        | M      | Yes     |
| T3F4  | Reassembling  | 6.3        | M      | Yes     |
| T3F5  | Separation  | 6.4        | M      | Yes     |
| T3F6  | Connection establishment  | 6.5        | M      | Yes     |
| T3F7  | Connection refusal  | 6.6        | M      | Yes     |
| T3F8  | Normal release when operating over CONS (explicit)  | 6.7.1      | M      | Yes     |
| T3F9  | Association of TPDU's with Transport connections when operating over CONS                 | 6.9.1      | M      | Yes     |
| T3F10 | Data TPDU numbering (normal)  | 6.10       | M      | Yes     |
| T3F11 | Expedited data transfer when operating over CONS (Network normal)                         | 6.11.1     | M      | Yes     |
| T3F12 | Reassignment after failure when operating over CONS                                       | 6.12       | M      | Yes     |
| T3F13 | Retention and acknowledgement of TPDU's<br>Retention until acknowledgement of TPDU's (AK) | 6.13.4.1   | M      | Yes     |
| T3F14 | Resynchronization   | 6.14       | M      | Yes     |
| T3F15 | Demultiplexing when operating over CONS   | 6.15       | M      | Yes     |
| T3F16 | Explicit flow control   | 6.16       | M      | Yes     |
| T3F17 | Frozen references   | 6.18       | M      | Yes     |
| T3F18 | Treatment of protocol errors when operating over CONS                                     | 6.22.1     | M      | Yes     |
| T3F19 | Multiplexing when operating over CONS   | 6.15       | M      | Yes     |

The following functions are optional if class 3 is supported

| Index | Function  | References | Status | Support |
|-------|---|------------|--------|---------|
| T3F20 | Concatenation   | 6.4        | O      | Yes No  |
| T3F21 | Data TPDU numbering (extended)  | 6.10       | O      | Yes No  |
| T3F22 | Retention and acknowledgement of TPDU's<br>Use of request acknowledgement | 6.13.4.3   | O      | Yes No  |

### C.9.5 Supported functions for class 4 (C4 or C4L::)

The following functions are mandatory

| Index | Function  | References | Status | Support |
|-------|---|------------|--------|---------|
| T4F1  | TPDU transfer   | 6.2        | M      | Yes     |
| T4F2  | Segmenting  | 6.3        | M      | Yes     |
| T4F3  | Reassembling  | 6.3        | M      | Yes     |
| T4F4  | Separation  | 6.4        | M      | Yes     |
| T4F5  | Connection establishment  | 6.5        | M      | Yes     |
| T4F6  | Connection refusal  | 6.6        | M      | Yes     |
| T4F7  | Data TPDU numbering (normal)  | 6.10       | M      | Yes     |
| T4F8  | Retention and acknowledgement of TPDU's<br>Retention until acknowledgement of TPDU's (AK) | 6.13.4.1   | M      | Yes     |
| T4F9  | Explicit flow control   | 6.16       | M      | Yes     |
| T4F10 | Checksum  | 6.17       | M      | Yes     |
| T4F11 | Frozen references   | 6.18       | M      | Yes     |
| T4F12 | Retransmission on time-out  | 6.19       | M      | Yes     |
| T4F13 | Resequencing  | 6.20       | M      | Yes     |
| T4F14 | Inactivity control  | 6.21       | M      | Yes     |

The following functions are mandatory if class 4 is operated over CONS

| Index | Function  | References | Status | Support |
|-------|---|------------|--------|---------|
| T4F15 | Assignment to network connection when operating over CONS                 | 6.1.1      | M      | Yes     |
| T4F16 | Normal release when operating over CONS (explicit)                        | 6.7.1      | M      | Yes     |
| T4F17 | Association of TPDU's with Transport connections when operating over CONS | 6.9.1      | M      | Yes     |
| T4F18 | Expedited data transfer when operating over CONS (Network normal)         | 6.11.1     | M      | Yes     |
| T4F19 | Multiplexing when operating over CONS                                     | 6.15       | M      | Yes     |
| T4F20 | Demultiplexing when operating over CONS                                   | 6.15       | M      | Yes     |
| T4F21 | Treatment of protocol errors when operating over CONS                     | 6.22.1     | M      | Yes     |
| T4F22 | Recombining when operating over CONS                                      | 6.23       | M      | Yes     |



The following functions are mandatory if class 4 is operated over CLNS

| Index | Function   | References | Status | Support |
|-------|--|------------|--------|---------|
| T4F23 | Transmission over CLNS   | 6.1.2      | M      | Yes     |
| T4F24 | Normal release when operating over CLNS (explicit)                       | 6.7.2      | M      | Yes     |
| T4F25 | Association of TPDU's with Transport connection when operating over CLNS | 6.9.2      | M      | Yes     |
| T4F26 | Expedited data transfer when operating over CLNS (Network normal)        | 6.11.2     | M      | Yes     |
| T4F27 | Treatment of protocol errors when operating over CLNS                    | 6.22.2     | M      | Yes     |

The following functions are optional

| Index | Function  | References | Status | Support |
|-------|---|------------|--------|---------|
| T4F28 | Data TPDU numbering (extended)  | 6.10       | O      | Yes No  |
| T4F29 | Non-use of checksum   | 6.17       | O      | Yes No  |
| T4F30 | Concatenation   | 6.4        | O      | Yes No  |
| T4F31 | Retention and acknowledgement of TPDU's<br>Use of selective acknowledgement | 6.13.4.4   | O      | Yes No  |
| T4F32 | Retention and acknowledgement of TPDU's<br>Use of request acknowledgement   | 6.13.4.3   | O      | Yes No  |

The following functions are optional if class 4 is operated over CONS

| Index | Function   | References | Status | Support |
|-------|--|------------|--------|---------|
| T4F33 | Splitting and recombining when operating over CONS | 6.23       | O      | Yes No  |

## C.10 Supported TPDU

The following TPDU and the parameters which constitute their fixed parts are mandatory if a corresponding predicate in the status column is true

| Index | TPDUs |                           | References | Status                           | Support |
|-------|-------|---------------------------|------------|----------------------------------|---------|
| ST1   | CR    | supported on transmission | 13.1       | IR1:M                            | Yes No  |
| ST2   | CR    | supported on receipt      | 13.1       | IR2:M                            | Yes No  |
| ST3   | CC    | supported on transmission | 13.1       | IR2:M                            | Yes No  |
| ST4   | CC    | supported on receipt      | 13.1       | IR1:M                            | Yes No  |
| ST5   | DR    | supported on transmission | 13.1       | IR2:M                            | Yes No  |
| ST6   | DR    | supported on receipt      | 13.1       | IR1:M                            | Yes No  |
| ST7   | DC    | supported on transmission | 13.1       | C1 OR C2 OR C3<br>OR C4 OR C4L:M | Yes No  |
| ST8   | DC    | supported on receipt      | 13.1       | C1 OR C2 OR C3<br>OR C4 OR C4L:M | Yes No  |
| ST9   | DT    | supported on transmission | 13.1       | M                                | Yes     |
| ST10  | DT    | supported on receipt      | 13.1       | M                                | Yes     |
| ST11  | ED    | supported on transmission | 13.1       | C1 OR C2 OR C3<br>OR C4 OR C4L:M | Yes No  |
| ST12  | ED    | supported on receipt      | 13.1       | C1 OR C2 OR C3<br>OR C4 OR C4L:M | Yes No  |
| ST13  | AK    | supported on transmission | 13.1       | C1 OR C2 OR C3<br>OR C4 OR C4L:M | Yes No  |
| ST14  | AK    | supported on receipt      | 13.1       | C1 OR C2 OR C3<br>OR C4 OR C4L:M | Yes No  |
| ST15  | EA    | supported on transmission | 13.1       | C1 OR C2 OR C3<br>OR C4 OR C4L:M | Yes No  |
| ST16  | EA    | supported on receipt      | 13.1       | C1 OR C2 OR C3<br>OR C4 OR C4L:M | Yes No  |
| ST17  | RJ    | supported on transmission | 13.1       | C1 OR C3:M                       | Yes No  |
| ST18  | RJ    | supported on receipt      | 13.1       | C1 OR C3:M                       | Yes No  |
| ST19  | ER    | supported on receipt      | 13.1       | M                                | Yes     |

State for which classes, if any, ER is supported on transmission

| Index | Class             | References | Status | Support |
|-------|-------------------|------------|--------|---------|
| SER0  | Class 0           | 6.22.1     | O      | Yes No  |
| SER1  | Class 1           | 6.22.1     | O      | Yes No  |
| SER2  | Class 2           | 6.22.1     | O      | Yes No  |
| SER3  | Class 3           | 6.22.1     | O      | Yes No  |
| SER4  | Class 4 over CONS | 6.22.1     | O      | Yes No  |
| SER4L | Class 4 over CLNS | 6.22.2     | O      | Yes No  |

The following TPDU's are mandatory if a corresponding predicate in the status column is true

| Index  | TPDU's |                           | References | Status           | Support |
|--|--------|---------------------------|------------|------------------|---------|
| SN3  | NCM    | supported on transmission | B.8.1      | N2:M             | Yes No  |
| SN4  | NCM    | supported on receipt      | B.8.1      | N2:M             | Yes No  |
| SN5  | DIAG   | supported on transmission | B.8.1      | N3:M             | Yes No  |
| SN6  | DIAG   | supported on receipt      | B.8.1      | N3:M             | Yes No  |
| SN7  | NCMC   | supported on transmission | B.8.1      | SN4 AND NOT N6:M | Yes No  |
| SN8  | NCMC   | supported on receipt      | B.8.1      | P1:M             | Yes No  |
| P1: SN3 and the only supported value in IC5 is "receiver". |        |                           |            |                  |         |

## C.11 Supported parameters of issued TPDU's

### C.11.1 Supported parameters for NCMS (A1::)

#### C.11.1.1 NCM TPDU (SN3::)

What are the allowed values of the following parameters for an NCM TPDU?

| Index | Supported parameters | References | Allowed values          | Supported values |
|-------|----------------------|------------|-------------------------|------------------|
| IC1   | NC-type              | B.8.3.3 c) | New, My, Yours          |                  |
| IC2   | NC-preference        | B.8.3.3 d) | Highest, Medium, Lowest |                  |
| IC3   | NC-collision         | B.8.3.3 e) | Resolution              |                  |
| IC4   | NC-recovery          | B.8.3.3 f) | Do not, Do              |                  |
| IC5   | NC-assignment right  | B.8.3.3 g) | Receiver, Sender, All   |                  |

#### C.11.2 Parameter values for CR TPDU (C1:: OR C2:: OR C3:: OR C4:: or C4L::)

If the additional options selection parameter is issued in a CR TPDU it is mandatory that

| Index |                                   | References |
|-------|-----------------------------------|------------|
| ICR1  | Bits 8 and 7 shall be set to zero | 13.3.4 g)  |

If the preferred class in the CR is 2, 3 or 4

| Index |  | References | Status     | Support          |
|-------|--|------------|------------|------------------|
| ICR2  | Is class 0 always offered as an alternative class? | 14.4       | O<br>CCT:M | Yes No<br>Yes No |

### C.11.3 Supported parameters for class 0 TPDU's (C0::)

The following parameters are optional if a CR TPDU is issued with preferred class 0

| Index | Supported parameters        | References | Status | Support |
|-------|-----------------------------|------------|--------|---------|
| I0CR6 | Called TSAP-ID              | 13.3.4 a)  | O      | Yes No  |
| I0CR7 | Calling TSAP-ID             | 13.3.4 a)  | O      | Yes No  |
| I0CR8 | TPDU size                   | 13.3.4 b)  | O      | Yes No  |
| I0CR9 | Preferred maximum TPDU size | 13.3.4 c)  | O      | Yes No  |

The following parameters are optional if a CC TPDU is issued in class 0

| Index | Supported parameters        | References | Status | Support |
|-------|-----------------------------|------------|--------|---------|
| I0CC6 | Called TSAP-ID              | 13.4.4     | O      | Yes No  |
| I0CC7 | Calling TSAP-ID             | 13.4.4     | O      | Yes No  |
| I0CC8 | TPDU size                   | 13.4.4     | O      | Yes No  |
| I0CC9 | Preferred maximum TPDU size | 13.4.4     | O      | Yes No  |

The following parameter is optional if a DR TPDU is issued in class 0

| Index | Supported parameter    | References | Status | Support |
|-------|------------------------|------------|--------|---------|
| I0DR4 | Additional information | 13.5.4 a)  | O      | Yes No  |

### C.11.4 Supported parameters for class 1 TPDU's (C1::)

The following parameters are optional if a CR TPDU is issued with preferred class 1

| Index  | Supported parameters        | References | Status | Support |
|--------|-----------------------------|------------|--------|---------|
| I1CR6  | Called TSAP-ID              | 13.3.4 a)  | O      | Yes No  |
| I1CR7  | Calling TSAP-ID             | 13.3.4 a)  | O      | Yes No  |
| I1CR8  | TPDU size                   | 13.3.4 b)  | O      | Yes No  |
| I1CR9  | Version number              | 13.3.4 d)  | O      | Yes No  |
| I1CR10 | Protection parameters       | 13.3.4 e)  | O      | Yes No  |
| I1CR11 | Additional option selection | 13.3.4 g)  | O      | Yes No  |
| I1CR12 | Alternative protocol class  | 13.3.4 h)  | O      | Yes No  |
| I1CR13 | Throughput                  | 13.3.4 j)  | O      | Yes No  |
| I1CR14 | Residual error rate         | 13.3.4 k)  | O      | Yes No  |
| I1CR15 | Priority                    | 13.3.4 l)  | O      | Yes No  |
| I1CR16 | Transit delay               | 13.3.4 m)  | O      | Yes No  |
| I1CR17 | Reassignment time           | 13.3.4 n)  | O      | Yes No  |
| I1CR18 | Preferred maximum TPDU size | 13.3.4 c)  | O      | Yes No  |

The following parameters are optional if a CC TPDU is issued in class 1

| Index  | Supported parameters        | References | Status | Support |
|--------|-----------------------------|------------|--------|---------|
| I1CC6  | Called TSAP-ID              | 13.4.4     | O      | Yes No  |
| I1CC7  | Calling TSAP-ID             | 13.4.4     | O      | Yes No  |
| I1CC8  | TPDU size                   | 13.4.4     | O      | Yes No  |
| I1CC9  | Protection parameters       | 13.4.4     | O      | Yes No  |
| I1CC10 | Additional option selection | 13.4.4     | O      | Yes No  |
| I1CC11 | Throughput                  | 13.4.4     | O      | Yes No  |
| I1CC12 | Residual error rate         | 13.4.4     | O      | Yes No  |
| I1CC13 | Priority                    | 13.4.4     | O      | Yes No  |
| I1CC14 | Transit delay               | 13.4.4     | O      | Yes No  |
| I1CC15 | Preferred maximum TPDU size | 13.4.4     | O      | Yes No  |

The following parameter is optional if a DR TPDU is issued in class 1

| Index | Supported parameter    | References | Status | Support |
|-------|------------------------|------------|--------|---------|
| I1DR4 | Additional information | 13.5.4 a)  | O      | Yes No  |

The following parameter is optional if an ER TPDU is issued in class 1

| Index | Supported parameter | References | Status | Support |
|-------|---------------------|------------|--------|---------|
| I1ER3 | Invalid TPDU        | 13.12.4 a) | O      | Yes No  |

The following parameter is mandatory in a DT TPDU if request of acknowledgement has been selected

| Index | Supported parameter | References | Status | Support |
|-------|---------------------|------------|--------|---------|
| I1DT4 | ROA                 | 13.7.3 a)  | M      | Yes No  |

### C.11.5 Supported parameters for class 2 TPDU's (C2::)

The following parameters are optional if a CR TPDU is issued with preferred class 2

| Index  | Supported parameters        | References | Status | Support |
|--------|-----------------------------|------------|--------|---------|
| I2CR6  | Called TSAP-ID              | 13.3.4 a)  | O      | Yes No  |
| I2CR7  | Calling TSAP-ID             | 13.3.4 a)  | O      | Yes No  |
| I2CR8  | TPDU size                   | 13.3.4 b)  | O      | Yes No  |
| I2CR9  | Version number              | 13.3.4 d)  | O      | Yes No  |
| I2CR10 | Protection parameters       | 13.3.4 e)  | O      | Yes No  |
| I2CR11 | Additional option selection | 13.3.4 g)  | O      | Yes No  |
| I2CR12 | Alternative protocol class  | 13.3.4 h)  | O      | Yes No  |
| I2CR13 | Throughput                  | 13.3.4 j)  | O      | Yes No  |
| I2CR14 | Residual error rate         | 13.3.4 k)  | O      | Yes No  |
| I2CR15 | Priority                    | 13.3.4 l)  | O      | Yes No  |
| I2CR16 | Transit delay               | 13.3.4 m)  | O      | Yes No  |
| I2CR17 | Preferred maximum TPDU size | 13.3.4 c)  | O      | Yes No  |

The following parameters are optional if a CC TPDU is issued in class 2

| Index  | Supported parameters        | References | Status | Support |
|--------|-----------------------------|------------|--------|---------|
| I2CC6  | Called TSAP-ID              | 13.4.4     | O      | Yes No  |
| I2CC7  | Calling TSAP-ID             | 13.4.4     | O      | Yes No  |
| I2CC8  | TPDU size                   | 13.4.4     | O      | Yes No  |
| I2CC9  | Protection parameters       | 13.4.4     | O      | Yes No  |
| I2CC10 | Additional option selection | 13.4.4     | O      | Yes No  |
| I2CC11 | Throughput                  | 13.4.4     | O      | Yes No  |
| I2CC12 | Residual error rate         | 13.4.4     | O      | Yes No  |
| I2CC13 | Priority                    | 13.4.4     | O      | Yes No  |
| I2CC14 | Transit delay               | 13.4.4     | O      | Yes No  |
| I2CC15 | Preferred maximum TPDU size | 13.4.4     | O      | Yes No  |

The following parameter is optional if a DR TPDU is issued in class 2

| Index | Supported parameter    | References | Status | Support |
|-------|------------------------|------------|--------|---------|
| I2DR4 | Additional information | 13.5.4 a)  | O      | Yes No  |

The following parameter is optional if an ER TPDU is issued in class 2

| Index | Supported parameter | References | Status | Support |
|-------|---------------------|------------|--------|---------|
| I2ER3 | Invalid TPDU        | 13.12.4 a) | O      | Yes No  |

### C.11.6 Supported parameters for class 3 TPDUs (C3::)

The following parameters are optional if a CR TPDU is issued with preferred class 3

| Index  | Supported parameters        | References | Status | Support |
|--------|-----------------------------|------------|--------|---------|
| I3CR6  | Called TSAP-ID              | 13.3.4 a)  | O      | Yes No  |
| I3CR7  | Calling TSAP-ID             | 13.3.4 a)  | O      | Yes No  |
| I3CR8  | TPDU size                   | 13.3.4 b)  | O      | Yes No  |
| I3CR9  | Version number              | 13.3.4 d)  | O      | Yes No  |
| I3CR10 | Protection parameters       | 13.3.4 e)  | O      | Yes No  |
| I3CR11 | Additional option selection | 13.3.4 g)  | O      | Yes No  |
| I3CR12 | Alternative protocol class  | 13.3.4 h)  | O      | Yes No  |
| I3CR13 | Throughput                  | 13.3.4 j)  | O      | Yes No  |
| I3CR14 | Residual error rate         | 13.3.4 k)  | O      | Yes No  |
| I3CR15 | Priority                    | 13.3.4 l)  | O      | Yes No  |
| I3CR16 | Transit delay               | 13.3.4 m)  | O      | Yes No  |
| I3CR17 | Reassignment time           | 13.3.4 n)  | O      | Yes No  |
| I3CR18 | Preferred maximum TPDU size | 13.3.4 c)  | O      | Yes No  |

The following parameters are optional if a CC TPDU is issued in class 3

| Index  | Supported parameters        | References | Status | Support |
|--------|-----------------------------|------------|--------|---------|
| I3CC6  | Called TSAP-ID              | 13.4.4     | O      | Yes No  |
| I3CC7  | Calling TSAP-ID             | 13.4.4     | O      | Yes No  |
| I3CC8  | TPDU size                   | 13.4.4     | O      | Yes No  |
| I3CC9  | Protection parameters       | 13.4.4     | O      | Yes No  |
| I3CC10 | Additional option selection | 13.4.4     | O      | Yes No  |
| I3CC11 | Throughput                  | 13.4.4     | O      | Yes No  |
| I3CC12 | Residual error rate         | 13.4.4     | O      | Yes No  |
| I3CC13 | Priority                    | 13.4.4     | O      | Yes No  |
| I3CC14 | Transit delay               | 13.4.4     | O      | Yes No  |
| I3CC15 | Preferred maximum TPDU size | 13.4.4     | O      | Yes No  |

The following parameter is optional if a DR TPDU is issued in class 3

| Index | Supported parameter    | References | Status | Support |
|-------|------------------------|------------|--------|---------|
| I3DR4 | Additional information | 13.5.4 a)  | O      | Yes No  |

The following parameter is optional if an ER TPDU is issued in class 3

| Index | Supported parameter | References | Status | Support |
|-------|---------------------|------------|--------|---------|
| I3ER3 | Invalid TPDU        | 13.12.4 a) | O      | Yes No  |

The following parameter is mandatory in a DT TPDU if request of acknowledgement has been selected

| Index | Supported parameter | References | Status | Support |
|-------|---------------------|------------|--------|---------|
| I3DT4 | ROA                 | 13.7.3 a)  | M      | Yes No  |

### C.11.7 Supported parameters for class 4 TPDU (C4 OR C4L::)

The following parameters are optional if a CR TPDU is issued with preferred class 4

| Index  | Supported parameters        | References | Status | Support |
|--------|-----------------------------|------------|--------|---------|
| I4CR7  | Called TSAP-ID              | 13.3.4 a)  | O      | Yes No  |
| I4CR8  | Calling TSAP-ID             | 13.3.4 a)  | O      | Yes No  |
| I4CR9  | TPDU size                   | 13.3.4 b)  | O      | Yes No  |
| I4CR10 | Version number              | 13.3.4 d)  | O      | Yes No  |
| I4CR11 | Protection parameters       | 13.3.4 e)  | O      | Yes No  |
| I4CR12 | Additional option selection | 13.3.4 g)  | O      | Yes No  |
| I4CR13 | Throughput                  | 13.3.4 j)  | O      | Yes No  |
| I4CR14 | Residual error rate         | 13.3.4 k)  | O      | Yes No  |
| I4CR15 | Priority                    | 13.3.4 l)  | O      | Yes No  |
| I4CR16 | Transit delay               | 13.3.4 m)  | O      | Yes No  |
| I4CR17 | Acknowledge time            | 13.3.4 i)  | O      | Yes No  |
| I4CR18 | Preferred maximum TPDU size | 13.3.4 c)  | O      | Yes No  |
| I4CR19 | Inactivity time             | 13.3.4 o)  | O      | Yes No  |

The following parameters are optional if a CR TPDU is issued with preferred class 4 over CONS

| Index  | Supported parameter        | References | Status | Support |
|--------|----------------------------|------------|--------|---------|
| I4CR20 | Alternative protocol class | 13.3.4 h)  | O      | Yes No  |



The following parameters are optional if a CC TPDU is issued in class 4

| Index  | Supported parameters        | References | Status | Support |
|--------|-----------------------------|------------|--------|---------|
| I4CC6  | Called TSAP-ID              | 13.4.4     | O      | Yes No  |
| I4CC7  | Calling TSAP-ID             | 13.4.4     | O      | Yes No  |
| I4CC8  | TPDU size                   | 13.4.4     | O      | Yes No  |
| I4CC9  | Protection parameters       | 13.4.4     | O      | Yes No  |
| I4CC10 | Additional option selection | 13.4.4     | O      | Yes No  |
| I4CC11 | Acknowledge time            | 13.4.4     | O      | Yes No  |
| I4CC12 | Throughput                  | 13.4.4     | O      | Yes No  |
| I4CC13 | Residual error rate         | 13.4.4     | O      | Yes No  |
| I4CC14 | Priority                    | 13.4.4     | O      | Yes No  |
| I4CC15 | Transit delay               | 13.4.4     | O      | Yes No  |
| I4CC16 | Preferred maximum TPDU size | 13.4.4     | O      | Yes No  |
| I4CC17 | Inactivity time             | 13.4.4     | O      | Yes No  |

The following parameter is optional if a DR TPDU is issued in class 4

| Index | Supported parameter    | References | Status | Support |
|-------|------------------------|------------|--------|---------|
| I4DR4 | Additional information | 13.5.4 a)  | O      | Yes No  |

The following parameter is mandatory in a DT TPDU if request of acknowledgement has been selected

| Index | Supported parameter | References | Status | Support |
|-------|---------------------|------------|--------|---------|
| I4DT4 | ROA                 | 13.7.3 a)  | M      | Yes No  |

The following parameter is mandatory in an AK TPDU if issued in class 4

| Index | Supported parameter       | References | Status | Support |
|-------|---------------------------|------------|--------|---------|
| I4AK4 | Flow control confirmation | 13.9.4 c)  | O      | Yes No  |

If the implementation can reduce credit and does so in the manner outlined in 12.2.3.8.2 then subsequence number in AK TPDUs is mandatory. Otherwise complete item I4AK5

| Index | Supported parameter | References | Status | Support |
|-------|---------------------|------------|--------|---------|
| I4AK5 | Subsequence number  | 13.9.4 b)  | O      | Yes No  |

The following parameter is optional in an AK TPDU if selective acknowledgement has been negotiated

| Index | Supported parameter                  | References | Status | Support |
|-------|--------------------------------------|------------|--------|---------|
| I4AK6 | Selective acknowledgement parameters | 13.9.4 d)  | O      | Yes No  |

The following parameter is optional if an ER TPDU is issued in class 4

| Index | Supported parameter | References | Status | Support |
|-------|---------------------|------------|--------|---------|
| I4ER3 | Invalid TPDU        | 13.12.4 a) | O      | Yes No  |

## C.12 Supported parameters for received TPDU

Implementors should be aware that implementations shall be capable of receiving and processing all possible parameters for all possible TPDU, dependent upon the class and optional functions implemented.

### C.12.1 Supported parameters for NCMS (A1::)

#### C.12.1.2 NCM TPDU (SN4::)

What are the allowed receive values of the following parameters for an NCM TPDU?

| Index | Supported parameters | References | Allowed values          | Supported values |
|-------|----------------------|------------|-------------------------|------------------|
| RN1   | NC-type              | B.8.3.3 c) | New, My, Yours          |                  |
| RN2   | NC-preference        | B.8.3.3 d) | Highest, Medium, Lowest |                  |
| RN3   | NC-collision         | B.8.3.3 e) | Resolution              |                  |
| RN4   | NC-recovery          | B.8.3.3 f) | Do not, Do              |                  |
| RN5   | NC-assignment right  | B.8.3.3 g) | Receiver, Sender, All   |                  |

### C.12.2 TPDU in class 4 (C4 OR C4L::)

If use of checksum has been selected then it is mandatory to process a checksum parameter in the following TPDU

| Index  | TPDU    | References | Status | Support |
|--------|---------|------------|--------|---------|
| R4CCch | CC TPDU | 13.4.4     | M      | Yes     |
| R4DRch | DR TPDU | 13.5.4 b)  | M      | Yes     |
| R4DCch | DC TPDU | 13.6.4     | M      | Yes     |
| R4DTch | DT TPDU | 13.7.4     | M      | Yes     |
| R4EDch | ED TPDU | 13.8.4     | M      | Yes     |
| R4AKch | AK TPDU | 13.9.4 a)  | M      | Yes     |
| R4EAch | EA TPDU | 13.10.4    | M      | Yes     |
| R4ERch | ER TPDU | 13.12.4 b) | M      | Yes     |

### C.13 User data in issued TPDU

A TS-user may issue data with a T-CONNECT request, T-CONNECT response or T-DISCONNECT request. Then it shall be possible to send user data as follows:

#### C.13.1 Class 1 (C1::)

| Index | User data   | References | Status | Support |
|-------|---|------------|--------|---------|
| D1ICR | User data of up to 32 octets in a CR with preferred class 1 | 13.3.5     | M      | Yes     |
| D1ICC | User data of up to 32 octets in a CC                        | 13.4.5     | M      | Yes     |
| D1IDR | User data of up to 64 octets in a DR                        | 13.5.5     | M      | Yes     |

#### C.13.2 Class 2 (C2::)

| Index | User data   | References | Status | Support |
|-------|---|------------|--------|---------|
| D2ICR | User data of up to 32 octets in a CR with preferred class 2 | 13.3.5     | M      | Yes     |
| D2ICC | User data of up to 32 octets in a CC                        | 13.4.5     | M      | Yes     |
| D2IDR | User data of up to 64 octets in a DR                        | 13.5.5     | M      | Yes     |

#### C.13.3 Class 3 (C3::)

| Index | User data   | References | Status | Support |
|-------|---|------------|--------|---------|
| D3ICR | User data of up to 32 octets in a CR with preferred class 3 | 13.3.5     | M      | Yes     |
| D3ICC | User data of up to 32 octets in a CC                        | 13.4.5     | M      | Yes     |
| D3IDR | User data of up to 64 octets in a DR                        | 13.5.5     | M      | Yes     |

#### C.13.4 Class 4 (C4 or C4L::)

| Index | User data   | References | Status | Support |
|-------|---|------------|--------|---------|
| D4ICR | User data of up to 32 octets in a CR with preferred class 4 | 13.3.5     | M      | Yes     |
| D4ICC | User data of up to 32 octets in a CC                        | 13.4.5     | M      | Yes     |
| D4IDR | User data of up to 64 octets in a DR                        | 13.5.5     | M      | Yes     |

### C.14 User data in received TPDU

For classes 1 to 4, if it is possible to initiate a CR TPDU then it shall be possible to receive the following

| Index | User data                           | References |
|-------|-------------------------------------|------------|
| DRCC  | 32 octets of user data in a CC TPDU | 13.4.5     |
| DRDR  | 64 octets of user data in a DR TPDU | 13.5.5     |

For classes 1 to 4, if it is possible to respond to a CR TPDU then it shall be possible to receive the following

| Index | User data                           | References |
|-------|-------------------------------------|------------|
| DRCR  | 32 octets of user data in a CR TPDU | 13.3.5     |

## C.15 Negotiation

### C.15.1 Class negotiation – initiator

If it is possible to initiate a CR TPDU in a particular class then the following holds

| Index |   | References |
|-------|---|------------|
| NC    | The preferred class in the CR TPDU may contain any of the classes supported by the implementation | 6.5.4 h)   |

What class(es) is (are) contained in the alternative class parameter if the preferred class is:

| Index | Preferred class   | References | Allowed values      | Supported values |
|-------|-------------------|------------|---------------------|------------------|
| NAC1  | Class 1           | 6.5.4 h)   | None, 0, 1          |                  |
| NAC2  | Class 2           | 6.5.4 h)   | None, 0, 2          |                  |
| NAC3  | Class 3           | 6.5.4 h)   | None, 0, 1, 2, 3    |                  |
| NAC4  | Class 4 over CONS | 6.5.4 h)   | None, 0, 1, 2, 3, 4 |                  |
| NAC5  | Class 4 over CLNS | 6.5.5 h)   | None                |                  |

### C.15.2 Class negotiation – responder

| Index | Preferred class   | References          | Allowed response   | Supported response |
|-------|---|---------------------|--|--------------------|
| RC0   | What classes can you respond with if CR proposes only class 0?  | 6.5.4 h)<br>Table 3 | 0 or connection refused depending on classes supported   |                    |
| RC1   | What classes can you respond with if CR proposes only class 1?  | 6.5.4 h)<br>Table 3 | 0, 1 or connection refused depending on classes supported  |                    |
| RC1a  | What classes can you respond with if CR proposes class 1 as preferred class and the alternative class parameter is present? | 6.5.4 h)<br>Table 3 | 0, 1 or connection refused depending on classes supported  |                    |
| RC2   | What classes can you respond with if CR proposes only class 2?  | 6.5.4 h)<br>Table 3 | 2 or connection refused depending on classes supported   |                    |
| RC2a  | What classes can you respond with if CR proposes class 2 as preferred class and the alternative class parameter is present? | 6.5.4 h)<br>Table 3 | 0, 2 or connection refused depending on classes supported and coding of alternative class          |                    |
| RC3   | What classes can you respond with if CR proposes only class 3?  | 6.5.4 h)<br>Table 3 | 2, 3 or connection refused depending on classes supported  |                    |
| RC3a  | What classes can you respond with if CR proposes class 3 as preferred class and the alternative class parameter is present? | 6.5.4 h)<br>Table 3 | 0, 1, 2, 3 or connection refused depending on classes supported and coding of alternative class    |                    |
| RC4   | What classes can you respond with if CR proposes only class 4?  | 6.5.4 h)<br>Table 3 | 2, 4 or connection refused depending on classes supported  |                    |
| RC4a  | What classes can you respond with if CR proposes class 4 as preferred class and the alternative class parameter is present? | 6.5.4 h)<br>Table 3 | 0, 1, 2, 3, 4 or connection refused depending on classes supported and coding of alternative class |                    |

### C.15.3 TPDU size negotiation

| Index |   | References | Status   | Support |
|-------|---|------------|----------|---------|
| TS1   | If maximum TPDU size is proposed in a CR TPDU then the initiator shall support all TPDU sizes from 128 octets to the maximum proposed.  | 14.5       | M        | Yes     |
| TS2   | If the preferred maximum TPDU size parameter is used in a CR TPDU then the initiator shall support all TPDU sizes, except 0, that are multiples of 128 octets up to the preferred maximum proposed. | 14.5 e)    | I4CR18:M | Yes No  |

| Index | TPDU size   | References | Allowed values  | Supported values |
|-------|---|------------|---|------------------|
| T0S1  | What is the largest value of the maximum TPDU size parameter in a CR TPDU with preferred class 0?                     | 14.5 e)    | NOT I0CR9: One of 128, 256, 512, 1024, 2048<br>I0CR9: One of $n \times 128$ with $n = 1, 2, 3, \dots$               |                  |
| T0S2  | What is the largest value of the maximum TPDU size parameter which may be sent in a CC TPDU when class 0 is selected? | 14.5 e)    | NOT I0CC9: One of 128, 256, 512, 1024, 2048<br>I0CC9: One of $n \times 128$ with $n = 1, 2, 3, \dots$               |                  |
| T1S1  | What is the largest value of the maximum TPDU size parameter in a CR TPDU with preferred class 1?                     | 14.5 e)    | NOT I1CR18: One of 128, 256, 512, 1024, 2048, 4096, 8192<br>I1CR18: One of $n \times 128$ with $n = 1, 2, 3, \dots$ |                  |
| T1S2  | What is the largest value of the maximum TPDU size parameter which may be sent in a CC TPDU when class 1 is selected? | 14.5 e)    | NOT I1CC15: One of 128, 256, 512, 1024, 2048, 4096, 8192<br>I1CC15: One of $n \times 128$ with $n = 1, 2, 3, \dots$ |                  |
| T2S1  | What is the largest value of the maximum TPDU size parameter in a CR TPDU with preferred class 2?                     | 14.5 e)    | NOT I2CR17: One of 128, 256, 512, 1024, 2048, 4096, 8192<br>I2CR17: One of $n \times 128$ with $n = 1, 2, 3, \dots$ |                  |
| T2S2  | What is the largest value of the maximum TPDU size parameter which may be sent in a CC TPDU when class 2 is selected? | 14.5 e)    | NOT I2CC15: One of 128, 256, 512, 1024, 2048, 4096, 8192<br>I2CC15: One of $n \times 128$ with $n = 1, 2, 3, \dots$ |                  |
| T3S1  | What is the largest value of the maximum TPDU size parameter in a CR TPDU with preferred class 3?                     | 14.5 e)    | NOT I3CR18: One of 128, 256, 512, 1024, 2048, 4096, 8192<br>I3CR18: One of $n \times 128$ with $n = 1, 2, 3, \dots$ |                  |
| T3S2  | What is the largest value of the maximum TPDU size parameter which may be sent in a CC TPDU when class 3 is selected? | 14.5 e)    | NOT I3CC15: One of 128, 256, 512, 1024, 2048, 4096, 8192<br>I3CC15: One of $n \times 128$ with $n = 1, 2, 3, \dots$ |                  |

| Index | TPDU size   | References | Allowed values   | Supported values |
|-------|---|------------|--|------------------|
| T4S1  | What is the largest value of the maximum TPDU size parameter in a CR TPDU with preferred class 4?                     | 14.5 e)    | NOT I4CR18: One of 128, 256, 512, 1024, 2048, 4096, 8192<br>I4CR18: One of n×128 with n = 1, 2, 3, ... |                  |
| T4S2  | What is the largest value of the maximum TPDU size parameter which may be sent in a CC TPDU when class 4 is selected? | 14.5 e)    | NOT I4CC16: One of 128, 256, 512, 1024, 2048, 4096, 8192<br>I4CC16: One of n×128 with n = 1, 2, 3, ... |                  |

#### C.15.4 Use of extended format

| Index | Extended format   | References           | Allowed values   | Supported values |
|-------|---|----------------------|------------------|------------------|
| NEF1  | What formats can you propose in the CR TPDU in class 2?                             | 6.5.4 l)             | normal, extended |                  |
| NEF2  | What formats can you propose in the CR TPDU in class 3?                             | 6.5.4 l)             | normal, extended |                  |
| NEF3  | What formats can you propose in the CR TPDU in class 4?                             | 6.5.4 l)<br>6.5.5 l) | normal, extended |                  |
| NEF4  | What formats can you select in CC when extended has been proposed in CR in class 2? | 6.5.4 l)             | normal, extended |                  |
| NEF5  | What formats can you select in CC when extended has been proposed in CR in class 3? | 6.5.4 l)             | normal, extended |                  |
| NEF6  | What formats can you select in CC when extended has been proposed in CR in class 4? | 6.5.4 l)<br>6.5.5 l) | normal, extended |                  |

#### C.15.5 Expedited data transport service

| Index |   | References           | Status | Support |
|-------|---|----------------------|--------|---------|
| TED1  | Expedited data indication in CR and CC TPDU | 6.5.4 g)<br>6.5.5 o) | M      | Yes     |

#### C.15.6 Non-use of checksum ((C4 OR C4L) AND T4F29::)

| Index | Non-use of checksum   | References           | Allowed values | Supported values |
|-------|---|----------------------|----------------|------------------|
| NUC1  | What proposals can you make in the CR?  | 6.5.4 m)<br>6.5.5 m) | non-use, use   |                  |
| NUC2  | What proposals can you make in CC when non-use of checksum has been proposed in CR? | 6.5.4 m)<br>6.5.5 m) | non-use, use   |                  |

**C.15.7 Explicit flow control (C2 and T2F19::)**

| Index | Explicit flow control  | References | Allowed values | Supported values |
|-------|--|------------|----------------|------------------|
| NUF1  | What proposals can you make in the CR?   | 6.5.4 o)   | non-use, use   |                  |
| NUF2  | What proposals can you make in CC when non-use of explicit flow control has been proposed in CR? | 6.5.4 o)   | non-use, use   |                  |

**C.15.8 Use of network receipt confirmation (C1 and T1F21::)**

| Index | Network receipt confirmation  | References | Allowed values | Supported values |
|-------|---|------------|----------------|------------------|
| UNRC1 | What proposals can you make in the CR?  | 6.5.4 p)   | non-use, use   |                  |
| UNRC2 | What proposals can you make in CC when use of network receipt confirmation has been proposed in CR? | 6.5.4 p)   | non-use, use   |                  |

**C.15.9 Use of network expedited data (C1 and T1F20::)**

| Index | Network expedited data  | References | Allowed values | Supported values |
|-------|---|------------|----------------|------------------|
| UNED1 | What proposals can you make in the CR?  | 6.5.4 p)   | non-use, use   |                  |
| UNED2 | What proposals can you make in CC when use of network expedited data has been proposed in CR? | 6.5.4 p)   | non-use, use   |                  |

**C.15.10 Use of selective acknowledgement**

| Index | Selective acknowledgement   | References           | Allowed values | Supported values |
|-------|---|----------------------|----------------|------------------|
| USA1  | Is use of selective acknowledgement proposed in CR TPDU's?                              | 6.5.4 r)<br>6.5.5 p) | Yes, No        |                  |
| USA2  | Is use of selective acknowledgement selected in a CC when it has been proposed in a CR? | 6.5.4 r)<br>6.5.5 p) | Yes, No        |                  |

**C.15.11 Use of request of acknowledgement**

| Index | Request of acknowledgement   | References           | Allowed values | Supported values |
|-------|--|----------------------|----------------|------------------|
| ROA1  | Is use of request of acknowledgement proposed in CR TPDU's?                              | 6.5.4 s)<br>6.5.5 g) | Yes, No        |                  |
| ROA2  | Is use of request of acknowledgement selected in a CC when it has been proposed in a CR? | 6.5.4 s)<br>6.5.5 g) | Yes, No        |                  |

## C.16 Error handling

### C.16.1 Action on receipt of a protocol error

| Index | Item              | References | Allowed values                        | Supported values |
|-------|-------------------|------------|---------------------------------------|------------------|
| PE0   | Class 0           | 6.22.1.3   | C0: ER, NDISreq, NRSTreq, Discard     |                  |
| PE1   | Class 1           | 6.22.1.3   | C1: ER, DR, NDISreq, NRSTreq, Discard |                  |
| PE2   | Class 2           | 6.22.1.3   | C2: ER, DR, NDISreq, NRSTreq, Discard |                  |
| PE3   | Class 3           | 6.22.1.3   | C3: ER, DR, NDISreq, NRSTreq, Discard |                  |
| PE4   | Class 4 over CONS | 6.22.1.3   | C4: ER, DR, NDISreq, NRSTreq, Discard |                  |
| PE4L  | Class 4 over CLNS | 6.22.2.3   | C4L: ER, DR, Discard                  |                  |

### C.16.2 Actions on receipt of an invalid or undefined parameter in a CR TPDU

| Index | Event   | References | Status | Support |
|-------|---|------------|--------|---------|
| RR1   | A parameter not defined in ITU-T Rec. X.224   ISO/IEC 8073 shall be ignored   | 13.2.3     | M      | Yes     |
| RR2   | An invalid value in the alternative protocol class parameter shall be treated as a protocol error   | 13.2.3     | M      | Yes     |
| RR3   | An invalid value in the class and option parameter shall be treated as a protocol error   | 13.2.3     | M      | Yes     |
| RR4   | On receipt of the additional option selection parameter bits 8 to 5, and bits 4 to 1 if not meaningful for the proposed class shall be ignored.           | 13.3.4     | M      | Yes     |
| RR5   | If non-use of explicit flow control is proposed and bit 1 of the additional option selection parameter equals 1, it shall be treated as a protocol error. | 13.2.3     | M      | Yes     |
| RR6   | On receipt of the class and option parameter bits 4 to 1 if not meaningful for the proposed class shall be ignored  | 13.3.3     | M      | Yes     |

What action is supported on receipt of the following?

| Index | Event   | References | Allowed actions        | Supported actions |
|-------|---|------------|------------------------|-------------------|
| RR7   | A parameter defined in ITU-T Rec. X.224   ISO/IEC 8073 (other than those covered above) and have an invalid value | 13.2.3     | Ignore, protocol error |                   |



### C.16.3 Actions on receipt of an invalid or undefined parameter in a TPDU other than a CR TPDU

The following actions are mandatory

| Index                 | Event   | References | Status | Support |
|-----------------------|---|------------|--------|---------|
| UI1                   | A parameter not defined in ITU-T Rec. X.224   ISO/IEC 8073 shall be treated as a protocol error                           | 13.2.3     | M      | Yes     |
| UI2                   | A parameter which has an invalid value as defined in ITU-T Rec. X.224   ISO/IEC 8073 shall be treated as a protocol error | 13.2.3     | M      | Yes     |
| UI3<br>(class 4 only) | A TPDU received with a checksum which does not satisfy the defined formula shall be discarded.                            | 6.17.3     | M      | Yes     |

### C.17 Timers and protocol parameters

The following are mandatory if class 4 is supported

| Index |       | References | Status | Support |
|-------|-------|------------|--------|---------|
| TA1   | $T_I$ | 12.2.1     | M      | Yes     |
| TA2   | $N$   | 12.2.1     | M      | Yes     |
| TA3   | $I_L$ | 12.2.1     | M      | Yes     |
| TA4   | $W$   | 12.2.1     | M      | Yes     |
| TA5   | $L$   | 12.2.1     | M      | Yes     |

| Index |  | References           | Status | Support |
|-------|--|----------------------|--------|---------|
| OT1   | Does IUT support optional timer $TS1$ when operating in class 0? | 6.5.4                | O      | Yes No  |
| OT2   | Does IUT support optional timer $TS1$ when operating in class 1? | 6.5.4                | O      | Yes No  |
| OT3   | Does IUT support optional timer $TS1$ when operating in class 2? | 6.5.4                | O      | Yes No  |
| OT4   | Does IUT support optional timer $TS1$ when operating in class 3? | 6.5.4                | O      | Yes No  |
| OT5   | Does IUT support optional timer $TS2$ when operating in class 0? | 6.7.1.5              | O      | Yes No  |
| OT6   | Does IUT support optional timer $TS2$ when operating in class 1? | 6.7.1.5              | O      | Yes No  |
| OT7   | Does IUT support optional timer $TS2$ when operating in class 2? | 6.7.1.5              | O      | Yes No  |
| OT8   | Does IUT support optional timer $TS2$ when operating in class 3? | 6.7.1.5              | O      | Yes No  |
| OT9   | Does IUT support optional timer $TS2$ when operating in class 4? | 6.22.1.3<br>6.22.2.3 | O      | Yes No  |

The following are mandatory if class 1 or 3 is supported

| Index |       | References       | Status | Support |
|-------|-------|------------------|--------|---------|
| TA6   | $TTR$ | 6.12.3<br>6.12.4 | M      | Yes     |
| TA7   | $TWR$ | 6.12.3<br>6.12.4 | M      | Yes     |

## Annex D (informative)

### Checksum Algorithms

(This annex forms an integral part of this Recommendation)

#### D.1 Symbols

The following symbols are used

$C0$ ,  $C1$  variables used in the algorithms

$i$  is the number (i.e. position) of an octet within the TPDU (see 12.1);

$n$  is the number (i.e. position) of the first octet of the checksum parameter;

$L$  is the length of the complete TPDU;

$X$  is the value of the first octet of the checksum parameter;

$Y$  is the value of the second octet of the checksum parameter.

#### D.2 Arithmetic conventions

Addition is performed in one of the two following modes:

- a) modulo 255 arithmetic;
- b) ones complement arithmetic in which if any of the variables has the value minus zero (i.e. 255) it shall be regarded as though it was plus zero (i.e. 0).

#### D.3 Algorithm for generating checksum parameters

**D.3.1** Set up the complete TPDU with the value of the checksum parameter field set to zero.

**D.3.2** Initialize  $C0$  and  $C1$  to zero.

**D.3.3** Process each octet sequentially from  $i = 1$  to  $L$  by

- a) adding the value of the octet to  $C0$ ; then
- b) adding the value of  $C0$  to  $C1$ .

**D.3.4** Calculate  $X$  and  $Y$  such that

$$X = -C1 + (L - n) C0$$

$$Y = C1 - (L - n + 1) C0$$

**D.3.5** Place the values  $X$  and  $Y$  in octets  $n$  and  $(n + 1)$  respectively.

NOTE – This algorithm calculates

$$C1 = \sum_{i=1}^L (L - i + 1) a_i$$

which is equal to zero, if the formulae in 6.17.3 are followed, since

$$\sum_{i=1}^L (L - i + 1) a_i = (L + 1) \sum_{i=1}^L a_i - \sum_{i=1}^L i a_i = 0$$

#### D.4 Algorithm for checking checksum parameters

**D.4.1** Initialize  $C0$  and  $C1$  to zero.

**D.4.2** Process each octet of the TPDU sequentially from  $i = 1$  to  $L$  by

- a) adding the value of the octet to  $C0$ ; then
- b) adding the value of  $C0$  to  $C1$ .

**D.4.3** If, when all the octets have been processed, one or both of the parameters  $C0$  and  $C1$  do not have the value zero, the checksum formulae in 6.17 have not been satisfied.

NOTE – The nature of the algorithm is such that it is not necessary to compare explicitly the stored checksum bytes.

## **Annex E** (informative)

### **State tables for operation of class 4 over connection-mode and connectionless-mode network services**

(This annex forms an integral part of this Recommendation)

#### **E.1 General**

This annex has been included as a guide for implementations which have been designed for operation over both the connection-mode network service and the connectionless-mode network service.

The introductory comments given in Annex A apply in this annex as well.

#### **E.2 Conventions**

Clause A.2 applies in this annex, except for item b) of A.2.3, in which the term "network connection" is to be extended to apply as well to the corresponding instance of communication over a connectionless-mode network service.

#### **E.3 Tables**

Clause A.3, including Tables A.1, A.2, A.3, apply in this annex.

#### **E.4 State tables for Class 4**

This clause incorporates all of clause A.6 with extensions to cover operation over connectionless-mode network services.

This clause provides a more precise description of a class 4 transport connection.

Tables E.1, E.2, E.3 give the predicates, actions and notes for class 4 respectively.

Table E.4 is the state table for a class 4 transport connection.

The following assumption and notations are used:

- \* Appropriate only for operation over connection-mode network service
- \*\* Appropriate only for operation over connectionless-mode network service
- a)\* the state of every network connection is known as being open or opening (i.e. an NCONreq has been issued and the NCONconf is awaited);
- b)\* for each transport connection the transport entity maintains the set of network connections to which the transport connection is assigned. A network connection in this set is either in the open or opening states;

- c)\* when an N-CONNECT confirmation, N-RESET indication or N-DISCONNECT indication is received this event is associated with the transport connection if the network connection belongs to the set;
  - d)\* when an N-DISCONNECT is received, the network connection ceases to exist and is therefore withdrawn from the set. When an NCONconf is received the state of the NC becomes “open”;
- NOTE – This is not shown by an explicit action in the state table. Conversely adding a network connection to a set and setting its state to “opening” is shown by an explicit action.
- e)\* when the state goes back to CLOSED or REFWAIT state, it is assumed that all timers are stopped (if running), the count is set to zero and the set becomes empty;
  - f)\* when a PDU is received the network connection on which it has been received is assumed to be known;
  - g)\* the variable “current-NC” is used to designate either the network connection on which a TPDU has been received or the network connection which has been chosen for a new assignment (either an existing one or a new one which is created);
  - h) we also assume the following variables:
    - local-ref: the reference (local) of the TC is chosen when sending the CR or when accepting a CR;
    - remote-ref: the reference of the remote entity is initially set to zero and initialised when processing the CC except if the CC is ignored.
    - SRC-REF: designates the corresponding field of the received TPDU.
    - DST-REF: designates the corresponding field of the received TPDU.
    - src-ref, dst-ref: designates the corresponding fields of the sent TPDU.
    - count: designates the numbers of times a TPDU has been sent (retransmissions);
  - i) the data transfer phase is not completely described in the state table but refers to the main text;
  - j)\* a spontaneous event called “new network connection assignment” has been introduced. It may occur at any time provided P1 or P2 are true (see predicate Table E.1) and the remote-ref is not zero (i.e. a CR TPDU has been received or a CC TPDU has been received and processed);
  - k)\* when an N-RESET indication is received, an N-RESET response is issued;
  - l)\*\* it is assumed that the connectionless-mode network service when being used is continuously available. The operations resulting from signalled inaccessibility of the network are a local matter.

TABLE E.1/X.224

**Predicates for class 4**

| Name   | Description  |
|--|--|
| P0   | T-CONNECT request is acceptable  |
| P1   | An assignment can be done to a suitable network connection (either open or opening)  |
| P2   | It is possible to open a new network connection  |
| P3   | Local choice   |
| P4   | A CR TPDU has never been sent  |
| P5   | The transport entity is the initiator and the set of network connections is now empty (i.e. a new assignment shall be done) or a new assignment is decided as a local choice |
| P6   | Local choice not to perform a new assignment if the set of network connections is empty (for closing state only)   |
| P7   | Count = maximum  |
| P8   | Acceptable CR TPDU   |
| P9   | Acceptable class 4 CC TPDU   |
| P10  | Unacceptable class 4 CC TPDU   |
| P11  | CC TPDU not specifying class 4   |
| P99  | Connection-mode network service being used   |
| NOTE – It is assumed that P99 = false implies only that the connectionless-mode network service is being used. |  |

TABLE E.2/X.224

**Specific actions for class 4**

| Name | Description  |
|------|--|
| [0]  | Set reference timer  |
| [1]  | Count = count + 1  |
| [2]  | Count = 0  |
| [3]  | Set retransmission timer   |
| [4]  | Stop retransmission timer if running   |
| [5]  | Set window timer   |
| [6]  | Stop window timer if running   |
| [7]  | Set inactivity timer   |
| [8]  | Stop inactivity timer if running   |
| [9]  | Set initial credit for sending according to the received CR/CC TPDU  |
| [10] | Set initial credit for controlling reception according to the sent CR/CC TPDU  |
| [11] | P99: Send the CR TPDU if there is a network connection in the open state in the set; not P99: send a CR TPDU   |
| [12] | P99: Add the current network connection to the set, if not already included; not P99: no action  |
| [13] | P99: The current network connection is now in the opening state; not P99: no action  |
| [14] | P99: Send the CC TPDU if a network connection in the open state is in the set; not P99: send CC TPDU   |
| [15] | P99: Send the DR TPDU if a network connection in the open state is in the set; not P99: send DR TPDU. In both cases, this DR TPDU is sent with src-ref = local-ref and dst-ref = remote-ref (may be zero)                      |
| [16] | P99: Send the DR TPDU if a network connection in the open state is in the set; not P99: send the DR TPDU. In both cases, the DR TPDU is sent with src-ref = 0 and dst-ref = remote-ref   |
| [17] | Send a TPDU according to data transfer procedure   |
| [18] | See state table of the class specified in the CC TPDU (refer to data transfer)   |
| [19] | P99: See state table of the class (refer to release procedure) send a DR TPDU if the class is not 0, otherwise issue a N-DISCONNECT request; not P99: send a DR TPDU   |
| [20] | Store request and exercise flow control to the user  |
| [21] | Send a DR TPDU with src-ref field set to zero  |
| [22] | Send a DC TPDU except if the SRC-REF field of the received DR TPDU is equal to zero  |
| [23] | P99: Send a DR TPDU with src-ref = local-ref and dst-ref = remote-ref (may be zero) if a network connection in the open state is in the set; not P99: send a DR TPDU with src-ref = local-ref and dst-ref = SRC-REF in CC TPDU |

TABLE E.3/X.224

**Specific notes for class 4**

| Name   | Description   |
|--|---|
| (1)*   | Not possible as no set of network connections is associated with this transport connection  |
| (2)*   | It is also possible to remain in the same state (T1 is still running) until: <ul style="list-style-type: none"> <li>• a CC TPDU is received which performs a new assignment,</li> <li>• a new assignment is tried (spontaneous event),</li> <li>• T1 runs out and the count is equal to the maximal value</li> </ul>  |
| (3)*   | No new assignment was possible: if the set is empty, the transport entity waits until a new assignment is received, or can be locally performed (spontaneous event)   |
| (4)*   | It is also possible to perform a new assignment. (This may be done in triggering the event "new network connection assignment")   |
| (5)  | Not a duplicated CR TPDU  |
| (6)*   | Since a new network connection is now assigned, it is recommended that the appropriate TPDU be sent on this network connection (if open) in order to make the remote entity aware of this assignment. It is also possible to allow the normal retransmission procedures to cause the TPDU to be sent; however the first TPDU available for sending should be sent on the new network connection |
| (7)  | As a local choice it is also possible to apply the following: [0], TDISind, REFWAIT   |
| (8)  | Association to this transport connection is done regardless of the SRC-REF field. If SRC-REF is not zero, a DC TPDU is set back   |
| (9)  | At least an AK TPDU shall be sent if the transport entity is the initiator in order to ensure that the responder will complete its three-way handshake  |
| (10)   | If association has been made, and DST-REF is zero, then the DC TPDU contains a src-ref field set to zero  |
| (11)   | If the CLOSING state has been entered, coming from WFCC state, the remote-ref is zero. The SRC-REF field of the CC TPDU is ignored (i.e. if the DR TPDU is retransmitted, it will be with the dst-ref field set to zero)  |
| (12)*  | If the CLOSING state has been entered, coming from WFCC state, the remote-ref (which is zero) shall be set with SRC-REF in order to comply with the release procedure of the negotiated class   |
| (13)   | The DR TPDU may be either repeated immediately or when T1 will run out  |
| (14)*  | If the set is empty, this event may be used as a criteria for triggering the event "new network connection assignment"  |
| (15)   | Previously stored T-DATA or T-EXPEDITED-DATA requests are ready for processing according to data transfer procedures  |
| (16)   | See data transfer procedures  |
| (17)*  | When an N-RESET indication is received, an N-RESET response has to be issued at once independent of the state automata  |
| (* appropriate only when operating over connection-mode network service) |   |

TABLE E.4/X.224

Class 4 connection/disconnection

|          |         | STATE   |  |      |                             |                         |                         |         |  |
|----------|---------|---|--|------|-----------------------------|-------------------------|-------------------------|---------|--|
| EVENT    | REFWAIT | CLOSED  | WFCC   | WBCL | OPEN                        | WFTRESP                 | AKWAIT                  | CLOSING |  |
| TCONreq  |         | Not P0:<br>TDISind<br>CLOSED;<br>P0 and ((P1 and P99) or not P99):<br>[12,1,3,10,11]<br>WFCC;<br>P99 and P0 and not P1 and P2:<br>13,12,1,3,10]<br>NCONreq<br>WFCC;<br>P99 and P0 and (not P1) and not P2:<br>TDISind<br>CLOSED |  |      |                             |                         |                         |         |  |
| TCONresp |         |   |  |      |                             | [3,2,1,10,14]<br>AKWAIT |                         |         |  |
| TDISreq  |         |   | P4:<br>CLOSED;<br>(not P4) and P3:<br>WBCL;<br>(not P4) and (not P3):<br>[4,3,2,1,15]<br>CLOSING |      | [6,8,4,3,2,1,15]<br>CLOSING | [16]<br>CLOSED          | [4,3,2,1,15]<br>CLOSING |         |  |



TABLE E.4/X.224 (continued)

**Class 4 connection/disconnection**

|   |         | STATE  |  |  |  |   |  |   |      |      |  |
|---|---------|--------|--|--|--|---|--|---|------|------|--|
| EVENT                                   | REFWAIT | CLOSED | WFCC   | WBCL   | OPEN   | WFTRESP   | AKWAIT   | CLOSING   |      |      |  |
| NDISind                                 | (1)     | (1)    | P99 and P1:<br>[12]<br>WFCC;<br>P99 and (not P1)<br>and P2:<br>[13,12]<br>NCONreq<br>WFCC;<br>P99 and (not P1)<br>and (not P2):<br>[0] [2]<br>TDISind<br>REFWAIT | P99 and P3:<br>[0]<br>REFWAIT;<br>P99 and (not P3)<br>and P1:<br>[12,11]<br>WBCL;<br>P99 and (not P3)<br>and (not P1) and<br>P2:<br>[13,12]<br>NCONreq<br>WBCL;<br>P99 and (not P3)<br>and (not P1) and<br>(not P2):<br>[0]<br>REFWAIT | P99 and P5 and<br>P1:<br>[12,17] (6)<br>OPEN;<br>P99 and (not P1)<br>and P5<br>and P2:<br>[13,12]<br>NCONreq<br>OPEN;<br>P99 and (not P1)<br>and P5 and (not<br>P2):<br>OPEN (3);<br>P99 and (not P5):<br>OPEN | P99:<br>WFTRESP<br>(4)  | P99 and P5 and<br>P1:<br>[12,14] (6)<br>AKWAIT;<br>P99 and (not P1)<br>and P5 and P2:<br>[13,12]<br>NCONreq<br>AKWAIT;<br>P99 and (not P1)<br>and P5 and (not<br>P2):<br>AKWAIT (3)<br>P99 and (not P5):<br>AKWAIT | P99 and P6: [0]<br>REFWAIT;<br>P99 and (not P6)<br>and P5 and P1:<br>[12,15]<br>CLOSING (6)<br>P99 and (not P6)<br>and P5 and (not<br>P1) and P2:<br>[13,12]<br>NCONreq<br>CLOSING;<br>P99 and (not P6)<br>and P5 and (not<br>P1) and (not P2):<br>CLOSING (3);<br>P99 and (not P6)<br>and (not P5):<br>CLOSING | (17) | (17) |  |
| NRSInd                                  |         |        | (17)   | (17)   | (17)   | (17)  | (17)   | (17)  |      |      |  |
| TDTrq<br>TEXreq                         |         |        |  |  | (16)<br>OPEN   |   | [20]<br>AKWAIT   |   |      |      |  |
| NCONconf                                | (1)     | (1)    | P99:<br>CR<br>WFCC (6)   | P99:<br>CR<br>WBCL (6)   | P99:<br>[17]<br>OPEN (6)   | P99:<br>WFTRESP   | P99:<br>CC<br>AKWAIT (6)   | P99:<br>[15]<br>CLOSING (6)   |      |      |  |
| New Network<br>Connection<br>Assignment |         |        |  |  | P99 and P1:<br>[12,17] OPEN<br>(6);<br>P99 and P2 and<br>(not P1):<br>[13,12] NCONreq<br>OPEN  | P99 and P1:<br>[12] WFTRESP;<br>P99 and P2 and<br>(not P1):<br>[13, 12]<br>NCONreq<br>WFTRESP | P99 and P1:<br>[12,14] (6)<br>AKWAIT;<br>P99 and P2 and<br>(not P1):<br>[13,12] NCONreq<br>AKWAIT  | P99 and P1:<br>[12,15] (6)<br>CLOSING;<br>P99 and P2 and<br>(not P1):<br>[13,12] NCONreq<br>CLOSING   |      |      |  |

TABLE E.4/X.224 (continued)

**Class 4 connection/disconnection**

| EVENT            | STATE      |  |   |   |  |              |   |  |  |  |
|------------------|------------|--|---|---|--|--------------|---|--|--|--|
|                  | REFWAIT    | CLOSED   | WFCC  | WBCL  | OPEN   | WFTRESP      | AKWAIT  | CLOSING  |  |  |
| Retrans-timer    |            |  | P7 and P3:<br>[0] TDISind<br>REFWAIT;<br>P7 and (not P3):<br>[3,2,1,15]<br>TDISind<br>CLOSING (14);<br>not P7:<br>[1,3,11] WFCC                                     | P7 and P3:<br>[0] REFWAIT;<br>P7 and (not P3):<br>[3,2,1,15]<br>CLOSING (14);<br>not P7:<br>[1,3,11] WBCL | P7:<br>[6,8,3,2,1,15]<br>TDISind<br>CLOSING (14);<br>not P7:<br>(16) (14) OPEN |              | P7:<br>[3,2,1,15]<br>TDISind (14)<br>CLOSING;<br>not P7:<br>[1,3,14] (14)<br>AKWAIT | P7:<br>[0] REFWAIT;<br>not P7:<br>[1,3,15] (14)<br>CLOSING   |  |  |
| Inactivity-Timer |            |  |   |   | [6,4,3,2,1,15]<br>TDISind<br>CLOSING (7)                                       |              |   |  |  |  |
| Reference-Time   | CLOSED     |  |   |   |  |              |   |  |  |  |
| CR               |            | not P8:<br>[21] CLOSED<br>(5);<br>P8:<br>[9,12] TCONind<br>WFTRESP (5) |   |   | [12,8,7] OPEN  | [12] WFTRESP | [12,14] AKWAIT  | [12] CLOSING<br>(13)   |  |  |
| CC               | DR REFWAIT | DR CLOSED  | P9:<br>[12,9,2,4,5,7,17]<br>TCONconf (9)<br>OPEN;<br>(not P9 and not<br>P99) or (P99 and<br>P10):<br>[12,4,3,2,1,23]<br>TDISind<br>CLOSING;<br>P99 and P11:<br>[18] | P99 and P11:<br>[19];<br>(not P99 and P9)<br>or (P99 and not<br>P11):<br>[12,2,4,3,1,15]<br>CLOSING       | [12,17,8,7] (9)<br>OPEN  |              |   | (P99 and P11):<br>[19] (12);<br>(not P99 and P9)<br>or (P99 and not<br>P11):<br>[12] CLOSING<br>(11) |  |  |
| ER               | REFWAIT    | CLOSED   | [0] TDISind<br>REFWAIT  | [0] REFWAIT   | [12,6,8,4,3,2,1,15]<br>] TDISind<br>CLOSING                                    |              | [12,4,3,2,1,15]<br>TDISind<br>CLOSING   | [0] REFWAIT  |  |  |

TABLE E.4/X.224 (end)

**Class 4 connection/disconnection**

|          |              | STATE       |                         |                 |                             |                        |                             |                   |  |  |  |
|----------|--------------|-------------|-------------------------|-----------------|-----------------------------|------------------------|-----------------------------|-------------------|--|--|--|
| EVENT    | REFWAIT      | CLOSED      | WFCC                    | WBCL            | OPEN                        | WFTRESP                | AKWAIT                      | CLOSING           |  |  |  |
| DR       | [22] REFWAIT | [22] CLOSED | (8) TDISind [0] REFWAIT | (8) [0] REFWAIT | DC (10) [0] TDISind REFWAIT | DC (10) TDISind CLOSED | DC (10) [0] TDISind REFWAIT | [0] REFWAIT       |  |  |  |
| DC       | REFWAIT      | CLOSED      |                         |                 |                             |                        |                             | [0] REFWAIT       |  |  |  |
| EA       | REFWAIT      | CLOSED      |                         |                 | [12,8,7] OPEN (16)          |                        |                             | [12] CLOSING (13) |  |  |  |
| DT/AK/ED | REFWAIT      | CLOSED      |                         |                 | [12,8,7] OPEN (16)          |                        | [12,7] OPEN (15) (16)       | [12] CLOSING (13) |  |  |  |

## **Appendix I**

### **Differences between Recommendation X.224 and ISO/IEC 8073:1992**

(This appendix does not form an integral part of this Recommendation)

Recommendation X.224 and ISO/IEC 8073:1992 are technically aligned except for the differences listed below.

#### **I.1 Assignment to network connections**

In 6.1.1.3 of Recommendation X.224, it is stated that the responder to the transport connection request becomes aware of the assignment when it receives particular TPDU. In clause 6.1.1.3 of ISO/IEC 8073, it is the non-owner of the network connection that becomes aware of the assignment upon receipt of these same TPDU.

#### **I.2 Conformance**

Recommendation X.224, in 14, requires all systems to implement class 0. ISO/IEC 8073, in clause 14, requires all systems to implement either class 0 or class 2.

#### **I.3 Class negotiation**

An additional restriction regarding the classes of transport protocol that are proposed in a CR TPDU is defined in item (a) in 14.4 of Recommendation X.224. No such restriction is contained in 14.5 of ISO/IEC 8073.

#### **I.4 Precedence**

ISO/IEC 8073, in its Annex A (State table), includes the statement: "In the event of a discrepancy between the description in these tables and that contained in the text, the text takes precedence." Recommendation X.224 does not contain this statement.

#### **I.5 Difference between Recommendation X.224 and ISO/IEC 8073**

The material in this appendix is not present in ISO/IEC 8073.



