IEC 61158-4-1

Edition 2.0 2014-08

INTERNATIONAL STANDARD

NORME INTERNATIONALE

**Industrial communication networks – Fieldbus specifications –
Part 4-1: Data-link layer protocol specification – Type 1 elements**

**Réseaux de communication industriels – Spécifications des bus de terrain –
Partie 4-1: Spécification du protocole de la couche liaison de données –
Éléments de type 1**

IEC 61158-4-1:2014-08(en-fr)

IEC Central Office
3, rue de Varembé
CH-1211 Geneva 20
Switzerland

Tel.: +41 22 919 02 11
Fax: +41 22 919 03 00
info@iec.ch
www.iec.ch

About the IEC
The International Electrotechnical Commission (IEC) is the leading global organization that prepares and publishes International Standards for all electrical, electronic and related technologies.

About IEC publications
The technical content of IEC publications is kept under constant review by the IEC. Please make sure that you have the latest edition, a corrigenda or an amendment might have been published.

IEC Catalogue - webstore.iec.ch/catalogue
The stand-alone application for consulting the entire bibliographical information on IEC International Standards, Technical Specifications, Technical Reports and other documents. Available for PC, Mac OS, Android Tablets and iPad.

IEC publications search - www.iec.ch/searchpub
The advanced search enables to find IEC publications by a variety of criteria (reference number, text, technical committee,…). It also gives information on projects, replaced and withdrawn publications.

IEC Just Published - webstore.iec.ch/justpublished
Stay up to date on all new IEC publications. Just Published details all new publications released. Available online and also once a month by email.

Electropedia - www.electropedia.org
The world's leading online dictionary of electronic and electrical terms containing more than 30 000 terms and definitions in English and French, with equivalent terms in 14 additional languages. Also known as the International Electrotechnical Vocabulary (IEV) online.

IEC Glossary - std.iec.ch/glossary
More than 55 000 electrotechnical terminology entries in English and French extracted from the Terms and Definitions clause of IEC publications issued since 2002. Some entries have been collected from earlier publications of IEC TC 37, 77, 86 and CISPR.

IEC Customer Service Centre - webstore.iec.ch/csc
If you wish to give us your feedback on this publication or need further assistance, please contact the Customer Service Centre: csc@iec.ch.

A propos de l'IEC
La Commission Electrotechnique Internationale (IEC) est la première organisation mondiale qui élabore et publie des Normes internationales pour tout ce qui a trait à l'électricité, à l'électronique et aux technologies apparentées.

A propos des publications IEC
Le contenu technique des publications IEC est constamment revu. Veuillez vous assurer que vous possédez l'édition la plus récente, un corrigendum ou amendement peut avoir été publié.

Catalogue IEC - webstore.iec.ch/catalogue
Application autonome pour consulter tous les renseignements bibliographiques sur les Normes internationales, Spécifications techniques, Rapports techniques et autres documents de l'IEC. Disponible pour PC, Mac OS, tablettes Android et iPad.

Recherche de publications IEC - www.iec.ch/searchpub
La recherche avancée permet de trouver des publications IEC en utilisant différents critères (numéro de référence, texte, comité d'études,…). Elle donne aussi des informations sur les projets et les publications remplacées ou retirées.

IEC Just Published - webstore.iec.ch/justpublished
Restez informé sur les nouvelles publications IEC. Just Published détaille les nouvelles publications parues. Disponible en ligne et aussi une fois par mois par email.

Electropedia - www.electropedia.org
Le premier dictionnaire en ligne de termes électroniques et électriques. Il contient plus de 30 000 termes et définitions en anglais et en français, ainsi que les termes équivalents dans 14 langues additionnelles. Egalement appelé Vocabulaire Electrotechnique International (IEV) en ligne.

Glossaire IEC - std.iec.ch/glossary
Plus de 55 000 entrées terminologiques électrotechniques, en anglais et en français, extraites des articles Termes et Définitions des publications IEC parues depuis 2002. Plus certaines entrées antérieures extraites des publications des CE 37, 77, 86 et CISPR de l'IEC.

Service Clients - webstore.iec.ch/csc
Si vous désirez nous donner des commentaires sur cette publication ou si vous avez des questions contactez-nous: csc@iec.ch.

IEC 61158-4-1

Edition 2.0 2014-08

INTERNATIONAL STANDARD

NORME INTERNATIONALE

Industrial communication networks – Fieldbus specifications – Part 4-1: Data-link layer protocol specification – Type 1 elements

Réseaux de communication industriels – Spécifications des bus de terrain – Partie 4-1: Spécification du protocole de la couche liaison de données – Éléments de type 1

INTERNATIONAL
ELECTROTECHNICAL
COMMISSION

COMMISSION
ELECTROTECHNIQUE
INTERNATIONALE

PRICE CODE
CODE PRIX XH

ICS 25.040.40; 35.100.20; 35.110

ISBN 978-2-8322-1719-1

Warning! Make sure that you obtained this publication from an authorized distributor.
Attention! Veuillez vous assurer que vous avez obtenu cette publication via un distributeur agréé.

CONTENTS

FOREWORD ... 9
0 INTRODUCTION ... 11
0.1 General ... 11
0.2 Nomenclature for references within this standard ... 11
1 Scope ... 12
1.1 General ... 12
1.2 Specifications ... 12
1.3 Procedures ... 12
1.4 Applicability ... 13
1.5 Conformance ... 13
2 Normative references ... 13
3 Terms, definitions, symbols and abbreviations ... 14
3.1 Reference model terms and definitions ... 14
3.2 Service convention terms and definitions ... 16
3.3 Terms and definitions ... 16
3.4 Symbols and abbreviations ... 25
4 Overview of the DL-protocol ... 29
4.1 Three-level model of the DLL ... 29
4.2 Service provided by the DLL ... 31
4.3 Structure and definition of DL-addresses ... 38
4.4 Service assumed from the PhL ... 50
4.5 Functions of the DLL ... 52
4.6 Functional classes ... 55
4.7 Local parameters, variables, counters, timers and queues ... 56
5 General structure and encoding of PhIDUs and DLPDUs, and related elements of procedure ... 70
5.1 PhIDU structure and encoding ... 70
5.2 Common DLPDU structure, encoding and elements of procedure ... 70
6 DLPDU-specific structure, encoding and elements of procedure ... 81
6.1 Establish connection (EC) DLPDU ... 83
6.2 Disconnect connection (DC) DLPDU ... 85
6.3 Reset connection (RC) DLPDU ... 88
6.4 Compel acknowledgement (CA) DLPDU ... 89
6.5 Compel data (CD) DLPDU ... 96
6.6 Exchange data (ED) DLPDU ... 103
6.7 Data (DT) DLPDU ... 111
6.8 Status response (SR) DLPDU ... 118
6.9 Compel time (CT) DLPDU ... 121
6.10 Time distribution (TD) DLPDU ... 123
6.11 Round-trip-delay query (RQ) DLPDU ... 125
6.12 Round-trip-delay reply (RR) DLPDU ... 127
6.13 Probe node DL-address (PN) DLPDU ... 129
6.14 Probe response (PR) DLPDU ... 131
6.15 Pass token (PT) DLPDU ... 133
6.16 Execute sequence (ES) DLPDU ... 141
6.17 Return token (RT) DLPDU ... 148

6.18 Request interval (RI) DLPDU 149
6.19 Claim LAS (CL) DLPDU 150
6.20 Transfer LAS (TL) DLPDU 152
6.21 Wakeup (WK) DLPDU 155
6.22 Idle (IDLE) DLPDU 157
6.23 Spare DLPDUs 158
6.24 Reserved (not to be used) DLPDUs 159
7 DLPDU-parameter structure and encoding 160
7.1 Structure and encoding of EC-PARAMETERS 160
7.2 Structure and encoding of DC-PARAMETERS 165
7.3 Structure and encoding of RC-PARAMETERS 166
7.4 Structure and encoding of SD-Parameters 168
7.5 Structure and encoding of SR-parameters 176
7.6 Structure and encoding of TD-parameters 176
7.7 Structure and encoding of RQ-parameters 179
7.8 Structure and encoding of RR-parameters 179
7.9 Structure and encoding of PN-parameters 180
7.10 Structure and encoding of DD-parameters 182
8 DL-service elements of procedure 182
8.1 Operation of the DL(SAP)-address, buffer and queue management services 183
8.2 Operation of the connection-mode services 186
8.3 Operation of the connectionless-mode services 226
8.4 Operation of the scheduling guidance services 237
9 DL-support subprotocol 246
9.1 General 246
9.2 Overview of LAS operation 247
9.3 DL-support subprotocol definition 247
9.4 Elements of Procedures for receiving SPDUs 280
10 Other DLE elements of procedure 282
10.1 DLE initialization 282
10.2 LAS behavior and operation 286
10.3 DL-support operation 293
10.4 DL-bridge elements of procedure and bridge sub-protocol 298
10.5 DL-management-information 328
10.6 Implementation profiles 332
11 PICS proforma 337
11.1 Introduction 338
11.2 General 338
11.3 Normative references 338
11.4 Definitions 338
11.5 Abbreviations 338
11.6 Conformance 339
11.7 Instructions 339
11.8 Identification 339
11.9 Implementation profile 340
11.10 Major low-level capabilities 344
11.11 Major high-level capabilities 357
Annex A (informative) Exemplary FCS implementation 366

Annex B (informative) Type 1: Formal protocol finite state machines ... 368
B.1 Basic reception and transmission FSMs ... 368
B.2 FSMs for DLCs ... 379
B.3 FSMs for scheduling ... 385
B.4 FSMs for bridges ... 385
Annex C (informative) Type 1: DLPDU and DL-addressing short-form summaries ... 387
C.1 Fields used in short-form summaries ... 387
C.2 DLPDU short-form summary grouped by function ... 388
C.3 DLPDU short-form summary in alphabetic order of DLPDU names ... 390
C.4 DLPDU short-form summary in alphabetic order of DLPDU acronyms ... 391
C.5 DLPDU FC code-point assignment matrix – overview and detail ... 392
C.6 SD-parameters (status and data-description parameters) of CA, CD, ED and DT DLPDUs ... 395
C.7 EC parameters of EC DLPDUs ... 398
C.8 Parameters of DC and RC DLPDUs ... 400
C.9 Parameters of TD, RQ and RR DLPDUs ... 401
C.10 Parameters of PN, PT, ES and RI DLPDUs ... 404
C.11 Addressing summary extracted from figures and tables of 4.3 ... 404
Bibliography ... 409

Figure 1 – Relationships of DLSAPs, DLSAP-addresses, DLCEPs, DLCEP-addresses, DLSEP-addresses and group DL-addresses ... 19
Figure 2 – Basic structure of a DL-address ... 38
Figure 3 – Basic structure of a sublink selector ... 39
Figure 4 – DL-address alternative structures ... 39
Figure 5 – Basic structure of MAC-addresses ... 49
Figure 6 – Representation of a DL-address as a MAC-address ... 49
Figure 7 – Linear relationships of sending and receiving DLCEP sequence-number variables ... 62
Figure 8 – DL-address alternative structures ... 73
Figure 9 – SHORT DL-address field – alternative implicit structures ... 74
Figure 10 – NODE DL-address field – implicit structure ... 74
Figure 11 – State transition diagram for a DLCEP ... 187
Figure 12 – Projection of the sending and receiving DLCEP sequence-number variables of Figure 7 onto the cyclic sequence-number parameters of CA, CD, DT, ED and RC DLPDUs, with consequent determination of required actions ... 203
Figure 13 – State transitions of a DLE ... 283
Figure 14 – Bridged network topology ... 299
Figure 15 – Spanning tree representation ... 300
Figure 16 – DLSDU transit delay, DLPDU lifetime and bridge forwarding delay ... 304
Figure 17 – Forwarding and delivering a received DLPDU ... 308
Figure 18 – Forwarding a locally-originated DLPDU ... 309
Figure 19 – Republishing a DLSDU received from another link ... 310
Figure 20 – Bridge architecture ... 311
Figure 21 – Replacement for [IL] Fig 3-2 Bridge ports ... 320
Figure 22 – Replacement for [IL] Fig 3-3 Bridge architecture ... 321
Figure A.1 – Example of FCS generation ... 366

Figure A.2 – Example of FCS syndrome checking on reception 366
Figure C.1 – Gross structure of FC code points 392
Figure C.2 [Figure 2] – Basic structure of a DL-address 405
Figure C.3 [Figure 3] – Basic structure of a sublink selector 405
Figure C.4 [Figure 4] – DL-address alternative structures 405
Figure C.5 [Figure 5] – Basic structure of MAC-addresses 405
Figure C.6 [Figure 6] – Representation of a DL-address as a MAC-address 405

Table 1 – Link || node || selector addressing 41
Table 2 – Link-local node || selector addressing 43
Table 3 – Link-local node designators 45
Table 4 – Node-local selector addressing 46
Table 5 – Predefined flat non-local DL-addresses 47
Table 6 – Predefined flat link-local DL-addresses 48
Table 7 – Predefined node-local DL-addresses 48
Table 8 – Correlation of DLPDUs with functional classes 54
Table 9 – FCS length, polynomial and expected residual 76
Table 10 – Summary structure of DLPDUs 82
Table 11 – DLPDU restrictions based on dominant token 83
Table 12 – Structure of EC DLPDUs 83
Table 13 – Structure of DC DLPDUs 86
Table 14 – Structure of RC DLPDUs 88
Table 15 – Structure of CA DLPDUs 90
Table 16 – Structure of CD DLPDUs 96
Table 17 – Structure of ED DLPDUs 103
Table 18 – Structure of DT DLPDUs 111
Table 19 – Structure of SR DLPDUs 119
Table 20 – Structure of CT DLPDUs 121
Table 21 – Structure of TD DLPDUs 123
Table 22 – Structure of RQ DLPDUs 125
Table 23 – Structure of RR DLPDUs 127
Table 24 – Structure of PN DLPDUs 129
Table 25 – Structure of PR DLPDUs 132
Table 26 – Structure of PT DLPDUs 133
Table 27 – Structure of ES DLPDUs 142
Table 28 – Structure of RT DLPDUs 148
Table 29 – Structure of RI DLPDUs 149
Table 30 – Structure of CL DLPDUs 150
Table 31 – Structure of TL DLPDUs 152
Table 32 – Structure of WK DLPDUs 155
Table 33 – Structure of IDLE DLPDUs 157
Table 34 – Assumed structure of undefined (spare) DLPDUs 158
Table 35 – Assumed structure of RESERVED (NOT TO BE USED) DLPDUs 160

Table 36 – Structure of an EC DLPDU's parameters ... 161
Table 37 – EC-parameters: 1st octet ... 161
Table 38 – EC-parameters: 2nd octet ... 161
Table 39 – EC-parameters: 3rd and 4th octets ... 162
Table 40 – EC-parameters: 5th and 6th octets ... 162
Table 41 – EC-parameters: 7th octet ... 163
Table 42 – EC-parameters: 8th octet ... 163
Table 43 – EC-parameters: 9th and 10th octets ... 164
Table 44 – EC-parameters: 11th octet ... 164
Table 45 – EC-parameters: 12th octet ... 165
Table 46 – EC-parameters: 13th and 14th octets ... 165
Table 47 – DC-parameters and RC-parameters: 1st octet ... 165
Table 48 – DC-parameters and RC-parameters: 2nd octet ... 166
Table 49 – Disconnect reasons ... 167
Table 50 – Reset reasons ... 168
Table 51 – RC-parameters: 3rd octet ... 168
Table 52 – RC-parameters: 4th octet ... 168
Table 53 – Structure of connectionless-mode CA, CD, DT and ED DLPDUs ... 169
Table 54 – Short format SD-parameters for connectionless transaction initiators ... 170
Table 55 – Short format SD-parameters for connectionless responders ... 170
Table 56 – Reply status for unitdata-acknowledgment and exchange-unitdata-reply DT DLPDUs ... 171
Table 57 – Structure of connection-oriented CA, CD, DT and ED DLPDUs ... 173
Table 58 – Short format SD-parameters for DLCEP state ... 174
Table 59 – Long format SD-parameters for DLCEP state: 1st octet ... 174
Table 60 – Long format SD-parameters for DLCEP state: 2nd octet ... 174
Table 61 – Long format SD-parameters for DLCEP state: 3rd octet ... 175
Table 62 – Reply status for SR DLPDUs ... 176
Table 63 – Short format SR-parameters ... 176
Table 64 – Structure of TD-parameters ... 177
Table 65 – Structure and encoding of the DL-time-quality measures ... 177
Table 66 – Approximate numeric significance of the bits of seven-octet DL-time ... 178
Table 67 – Approximate numeric significance of the bits of three-octet short time ... 179
Table 68 – Structure of RQ-parameters ... 179
Table 69 – Structure of RR-parameters ... 179
Table 70 – Structure and encoding of the RR-time-quality measures ... 180
Table 71 – Structure of PN-parameters ... 181
Table 72 – PN-parameters: 1st octet ... 181
Table 73 – PN-parameters: 2nd octet ... 181
Table 74 – PN-parameters: 3rd and 4th octets ... 181
Table 75 – PN-parameters: 5th octet ... 182
Table 76 – PN-parameters: 6th octet ... 182
Table 77 – Structure of DD-parameters ... 182

Table 78 – Components of returned DL-time 238
Table 79 – Time synchronization computation 240
Table 80 – SPDU 1st octet: SPDU class, and protocol version or subclass 248
Table 81 – Probe-response SPDU 249
Table 82 – DL-protocol versions supported 249
Table 83 – PR-SPDU: 3rd and 4th octets 249
Table 84 – Node-activation SPDU 251
Table 85 – Node-activation SPDU: 4th octet 251
Table 86 – LAS-data-base-status SPDU 252
Table 87 – LAS-data-base-status SPDU: 2nd octet 252
Table 88 – Live-list- change SPDU 252
Table 89 – DLE-status structure 253
Table 90 – Live-list-detail SPDU 254
Table 91 – DL-conformance-reply SPDU 255
Table 92 – DL-protocol versions supported 255
Table 93 – DL-conformance encoding (portion 1) 255
Table 94 – DL-conformance encoding (portion 2) 256
Table 95 – DL-conformance encoding (portion 3) 256
Table 96 – DL-conformance encoding (portion 4) 256
Table 97 – Link-basic-parameters-reply SPDU 257
Table 98 – Link-master-parameters-reply SPDU 258
Table 99 – Token-hold-time-request SPDU 259
Table 100 – Token-hold-time-array SPDU 259
Table 101 – Sequence element header encoding 261
Table 102 – SHORT DL-address and duration sequence element 261
Table 103 – LONG DL-address and duration sequence element 262
Table 104 – Wakeup request sequence element 263
Table 105 – Schedule-request SPDU 264
Table 106 – Sequence type, schedule type and priority encoding 264
Table 107 – Scheduling-completed SPDU 266
Table 108 – Status and reason codes 266
Table 109 – Cancel-schedule SPDU 266
Table 110 – Schedule-cancelled SPDU 267
Table 111 – Link-schedule 268
Table 112 – Schedule-summary SPDU 268
Table 113 – Subschedule-SPDU reference 269
Table 114 – Subschedule SPDU 270
Table 115 – Sequence Sub-SPDU 271
Table 116 – Element-description 271
Table 117 – Schedule-summary-request SPDU 272
Table 118 – Subschedule-request SPDU 273
Table 119 – Parameter-list element-header encoding 273
Table 120 – Begin/end-of-list element 274

Table 121 – Continuation-of-list element 274
Table 122 – SHORT DL-address list element 274
Table 123 – LONG DL-address element 275
Table 124 – DLSAP-address-characteristics element 275
Table 125 – DLCEP-characteristics element 276
Table 126 – Address-query SPDU 276
Table 127 – Address-report SPDU 277
Table 128 – Address-list-query SPDU 278
Table 129 – DL-address selection criteria 279
Table 130 – Address-list-reply SPDU 280
Table 131 – Topology change notification BPDU format 328
Table 132 – Configuration BPDU format 328
Table 133 – Maximum permitted phase-tracking error in a DLE's sense of DL-time at the minimum requireable Time Distribution period 336
Table C.1 – Generic assignment of FC code points 393
Table C.2 – Individual assignment of FC code points 394
Table C.3 – Reply status for SR DLPDUs 397
Table C.4 – Reply status for unitdata-acknowledgment and exchange-unitdata-reply DT DLPDUs 398
Table C.5 – Approximate numeric significance of the bits of seven-octet DL-time 403
Table C.6 – Approximate numeric significance of the bits of N(NT), A...A, and three-octet C(NT) 403
Table C.7 [Table 1] – Link || node || selector addressing 406
Table C.8 [Table 2] – Link-local node || selector addressing 406
Table C.9 [Table 5] – Predefined flat non-local DL-addresses 407
Table C.10 [Table 6] – Predefined flat link-local DL-addresses 407
Table C.11 [Table 3] – Link-local node designators 408
Table C.12 [Table 4] – Node-local selector addressing 408
Table C.13 [Table 7] – Predefined node-local DL-addresses 408

INTERNATIONAL ELECTROTECHNICAL COMMISSION

INDUSTRIAL COMMUNICATION NETWORKS – FIELDBUS SPECIFICATIONS –

Part 4-1: Data-link layer protocol specification – Type 1 elements

FOREWORD

1) The International Electrotechnical Commission (IEC) is a worldwide organization for standardization comprising all national electrotechnical committees (IEC National Committees). The object of IEC is to promote international co-operation on all questions concerning standardization in the electrical and electronic fields. To this end and in addition to other activities, IEC publishes International Standards, Technical Specifications, Technical Reports, Publicly Available Specifications (PAS) and Guides (hereafter referred to as "IEC Publication(s)"). Their preparation is entrusted to technical committees; any IEC National Committee interested in the subject dealt with may participate in this preparatory work. International, governmental and non-governmental organizations liaising with the IEC also participate in this preparation. IEC collaborates closely with the International Organization for Standardization (ISO) in accordance with conditions determined by agreement between the two organizations.
2) The formal decisions or agreements of IEC on technical matters express, as nearly as possible, an international consensus of opinion on the relevant subjects since each technical committee has representation from all interested IEC National Committees.
3) IEC Publications have the form of recommendations for international use and are accepted by IEC National Committees in that sense. While all reasonable efforts are made to ensure that the technical content of IEC Publications is accurate, IEC cannot be held responsible for the way in which they are used or for any misinterpretation by any end user.
4) In order to promote international uniformity, IEC National Committees undertake to apply IEC Publications transparently to the maximum extent possible in their national and regional publications. Any divergence between any IEC Publication and the corresponding national or regional publication shall be clearly indicated in the latter.
5) IEC itself does not provide any attestation of conformity. Independent certification bodies provide conformity assessment services and, in some areas, access to IEC marks of conformity. IEC is not responsible for any services carried out by independent certification bodies.
6) All users should ensure that they have the latest edition of this publication.
7) No liability shall attach to IEC or its directors, employees, servants or agents including individual experts and members of its technical committees and IEC National Committees for any personal injury, property damage or other damage of any nature whatsoever, whether direct or indirect, or for costs (including legal fees) and expenses arising out of the publication, use of, or reliance upon, this IEC Publication or any other IEC Publications.
8) Attention is drawn to the Normative references cited in this publication. Use of the referenced publications is indispensable for the correct application of this publication.
9) Attention is drawn to the possibility that some of the elements of this IEC Publication may be the subject of patent rights. IEC shall not be held responsible for identifying any or all such patent rights.

Attention is drawn to the fact that the use of the associated protocol type is restricted by its intellectual-property-right holders. In all cases, the commitment to limited release of intellectual-property-rights made by the holders of those rights permits a layer protocol type to be used with other layer protocols of the same type, or in other type combinations explicitly authorized by its intellectual-property-right holders.

NOTE Combinations of protocol types are specified in IEC 61784-1 and IEC 61784-2.

International Standard IEC 61158-4-1 has been prepared by subcommittee 65C: Industrial networks, of IEC technical committee 65: Industrial-process measurement, control and automation.

This second edition cancels and replaces the first edition published in 2007. This edition constitutes a technical revision.

The main change with respect to the previous edition is listed below:

- Improved terms

The text of this standard is based on the following documents:

FDIS	Report on voting
65C/762/FDIS	65C/772/RVD

Full information on the voting for the approval of this standard can be found in the report on voting indicated in the above table.

This publication has been drafted in accordance with the ISO/IEC Directives, Part 2.

NOTE 2 Slight variances from the directives have been allowed by the IEC Central Office to provide continuity of subclause numbering with prior editions.

A list of all the parts of the IEC 61158 series, under the general title *Industrial communication networks – Fieldbus specifications*, can be found on the IEC web site.

The committee has decided that the contents of this publication will remain unchanged until the stability date indicated on the IEC web site under "http://webstore.iec.ch" in the data related to the specific publication. At this date, the publication will be

- reconfirmed,
- withdrawn,
- replaced by a revised edition, or
- amended.

0 INTRODUCTION

0.1 General

This part of IEC 61158 is one of a series produced to facilitate the interconnection of automation system components. It is related to other standards in the set as defined by the "three-layer" fieldbus reference model described in IEC 61158-1.

The data-link protocol provides the data-link service by making use of the services available from the physical layer. The primary aim of this standard is to provide a set of rules for communication expressed in terms of the procedures to be carried out by peer data-link entities (DLEs) 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:

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 to the understanding of time-critical communications within OSI.

This standard is concerned, in particular, with the communication and interworking of sensors, effectors and other automation devices. By using this standard together with other standards positioned within the OSI or fieldbus reference models, otherwise incompatible systems may work together in any combination.

0.2 Nomenclature for references within this standard

Clauses, including annexes, can be referenced in their entirety, including any subordinate subclauses, as "Clause N" or "Annex N", where N is the number of the clause or letter of the annex.

Subclauses can be referenced in their entirety, including any subordinate subclauses, as "N.M" or "N.M.P" and so forth, depending on the level of the subclause, where N is the number of the subclause or letter of the annex, and M, P and so forth represent the successive levels of subclause up to and including the subclause of interest.

When a clause or subclause contains one or more subordinate subclauses, the text between the clause or subclause heading and its first subordinate subclause can be referenced in its entirety as "N.0" or "N.M.0" or "N.M.P.0" and so forth, where N, M and P are as above. Stated differently, a reference ending with ".0" designates the text and figures between a clause or subclause header and its first subordinate subclause.

NOTE This nomenclature provides a means of referencing text in hanging clauses. Such clauses existed in earlier editions of IEC 61784-3, Type 1 clauses. Those hanging clauses are maintained in this edition to minimize the disruption to existing national and multi-national standards and consortia documents which reference that prior subclause numbering.

INDUSTRIAL COMMUNICATION NETWORKS – FIELDBUS SPECIFICATIONS –

Part 4-1: Data-link layer protocol specification – Type 1 elements

1 Scope

1.1 General

The data-link layer provides basic time-critical messaging communications between devices in an automation environment.

This protocol provides the data-link service by making use of the services available from the physical layer. The relationship between the International Standards for fieldbus data-link service, fieldbus data-link protocol, fieldbus physical service and systems management is described in IEC 61158-1.

This protocol provides communication opportunities to all participating data-link entities

a) in a cyclic asynchronous manner, sequentially to each of those data-link entities, and
b) in a synchronous manner, either cyclically or acyclically, according to a pre-established schedule.

The specified protocol also provides means of changing the set of participating data-link entities and of modifying the set of scheduled communications opportunities. When the set of scheduled communications opportunities is null, the distribution of communication opportunities to the participating data-link entities is completely asynchronous.

Thus this protocol can be characterized as one which provides access asynchronously but with a synchronous overlay.

1.2 Specifications

This standard specifies

a) procedures for the timely transfer of data and control information from one data-link user entity to a peer user entity, and among the data-link entities forming the distributed data-link service provider;
b) the structure of the fieldbus DLPDUs used for the transfer of data and control information by the protocol of this standard, and their representation as physical interface data units.

NOTE In IEC 61158-4-1, gray boxes have been used in the tables to indicate that the specified field is not a conceptual part of the specific DLPDU.

1.3 Procedures

The procedures are defined in terms of

a) the interactions between peer DL-entities (DLEs) through the exchange of fieldbus DLPDUs;
b) the interactions between a DL-service (DLS) provider and a DLS-user in the same system through the exchange of DLS primitives;
c) the interactions between a DLS-provider and a Ph-service provider in the same system through the exchange of Ph-service primitives.

1.4 Applicability

These procedures are applicable to instances of communication between systems which support time-critical communications services within the data-link layer of the OSI or fieldbus reference models and which require the ability to interconnect in an open systems interconnection environment.

Profiles provide a simple multi-attribute means of summarizing an implementation's capabilities, and thus its applicability to various time-critical communications needs.

1.5 Conformance

This standard also specifies conformance requirements for systems implementing these procedures. This standard does not contain tests to demonstrate compliance with such requirements.

2 Normative references

The following documents, in whole or in part, are normatively referenced in this document and are indispensable for its application. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

NOTE All parts of the IEC 61158 series, as well as IEC 61784-1 and IEC 61784-2 are maintained simultaneously. Cross-references to these documents within the text therefore refer to the editions as dated in this list of normative references.

IEC 61158-1:2014, *Industrial communication networks – Fieldbus specifications – Part 1: Overview and guidance for the IEC 61158 and IEC 61784 series*

IEC 61158-2:2014, *Industrial communication networks – Fieldbus specifications – Part 2: Physical layer specification and service definition*

IEC 61158-3-1:2014, *Industrial communication networks – Fieldbus specifications – Part 3-1: Data link service definition – Type 1 elements*

ISO/IEC 7498-1*, Information technology – Open Systems Interconnection – Basic Reference Model: The Basic Model*

ISO/IEC 7498-3, *Information technology – Open Systems Interconnection – Basic Reference Model: Naming and addressing*

ISO/IEC 8886, *Information technology – Open Systems Interconnection – Data link service definition*

ISO/IEC 10038:1993, *Information technology – Telecommunications and information exchange between systems – Local area networks – Media access control (MAC) bridges*

NOTE This edition has been withdrawn and replaced by ISO/IEC 15802-3:1998. However, the detailed references in this standard are to the 1993 edition.

ISO/IEC 10731, *Information technology – Open Systems Interconnection – Basic Reference Model – Conventions for the definition of OSI services*

3 Terms, definitions, symbols and abbreviations

For the purposes of this document, the following terms, definitions, symbols and abbreviations apply.

3.1 Reference model terms and definitions

This standard is based in part on the concepts developed in ISO/IEC 7498-1 and ISO/IEC 7498-3, and makes use of the following terms defined therein.

3.1.1 called-DL-address [7498-3]

3.1.2 calling-DL-address [7498-3]

3.1.3 centralized multi-end-point-connection [7498-1]

3.1.4 correspondent (N)-entities [7498-1]
correspondent DL-entities (N=2)
correspondent Ph-entities (N=1)

3.1.5 demultiplexing [7498-1]

3.1.6 DL-address [7498-3]

3.1.7 DL-address-mapping [7498-1]

3.1.8 DL-connection [7498-1]

3.1.9 DL-connection-end-point [7498-1]

3.1.10 DL-connection-end-point-identifier [7498-1]

3.1.11 DL-connection-mode transmission [7498-1]

3.1.12 DL-connectionless-mode transmission [7498-1]

3.1.13 DL-data-sink [7498-1]

3.1.14 DL-data-source [7498-1]

3.1.15 DL-duplex-transmission [7498-1]

3.1.16 DL-facility [7498-1]

3.1.17 DL-local-view [7498-3]

3.1.18 DL-name [7498-3]

3.1.19 DL-protocol [7498-1]

3.1.20 DL-protocol-connection-identifier [7498-1]

3.1.21 DL-protocol-control-information [7498-1]

3.1.22 DL-protocol-data-unit [7498-1]

3.1.23 DL-protocol-version-identifier [7498-1]

3.1.24 DL-relay [7498-1]

3.1.25 DL-service-connection-identifier [7498-1]

3.1.26 DL-service-data-unit [7498-1]

3.1.27 DL-simplex-transmission [7498-1]

3.1.28 DL-subsystem [7498-1]

3.1.29 DL-user-data [7498-1]

3.1.30 flow control [7498-1]

3.1.31 layer-management [7498-1]

3.1.32 multiplexing [7498-3]

3.1.33 naming-(addressing)-authority [7498-3]

3.1.34 naming-(addressing)-domain [7498-3]

3.1.35 naming-(addressing)-subdomain [7498-3]

3.1.36 (N)-entity [7498-1]
DL-entity
Ph-entity

3.1.37 (N)-interface-data-unit [7498-1]
DL-service-data-unit (N=2)
Ph-interface-data-unit (N=1)

3.1.38 (N)-layer [7498-1]
DL-layer (N=2)
Ph-layer (N=1)

3.1.39 (N)-service [7498-1]
DL-service (N=2)
Ph-service (N=1)

3.1.40 (N)-service-access-point [7498-1]
DL-service-access-point (N=2)
Ph-service-access-point (N=1)

3.1.41 (N)-service-access-point-address [7498-1]
DL-service-access-point-address (N=2)
Ph-service-access-point-address (N=1)

3.1.42 peer-entities [7498-1]

3.1.43 Ph-interface-control-information [7498-1]

3.1.44 Ph-interface-data [7498-1]

3.1.45 primitive name [7498-3]

3.1.46 reassembling [7498-1]

3.1.47 recombining [7498-1]

3.1.48 reset [7498-1]

3.1.49 responding-DL-address [7498-3]

3.1.50 routing [7498-1]

3.1.51 segmenting [7498-1]

3.1.52 sequencing [7498-1]

3.1.53 splitting [7498-1]

3.1.54 synonymous name [7498-3]

3.1.55 systems-management [7498-1]

3.2 Service convention terms and definitions

This standard also makes use of the following terms defined in ISO/IEC 10731 as they apply to the data-link layer:

3.2.1 acceptor

3.2.2 asymmetrical service

3.2.3 confirm (primitive);
requestor.deliver (primitive)

3.2.4 deliver (primitive)

3.2.5 DL-confirmed-facility

3.2.6 DL-facility

3.2.7 DL-local-view

3.2.8 DL-mandatory-facility

3.2.9 DL-non-confirmed-facility

3.2.10 DL-provider-initiated-facility

3.2.11 DL-provider-optional-facility

3.2.12 DL-service-primitive;
primitive

3.2.13 DL-service-provider

3.2.14 DL-service-user

3.2.15 DL-user-optional-facility

3.2.16 indication (primitive)
acceptor.deliver (primitive)

3.2.17 multi-peer

3.2.18 request (primitive);
requestor.submit (primitive)

3.2.19 requestor

3.2.20 response (primitive);
acceptor.submit (primitive)

3.2.21 submit (primitive)

3.2.22 symmetrical service

3.3 Terms and definitions

For the purposes of this document, the following terms and definitions apply.

3.3.1
bridge
DL-router
DL-relay entity which performs selective store-and-forward and routing functions

a) to connect two or more separate DL-subnetworks (links) to form a unified DL-subnetwork (the extended link); and

b) to provide a means by which two end systems can communicate, when at least one of the end systems is periodically inattentive to the interconnecting DL-subnetwork,

and also provides time synchronization among the links to which it is forwarding

3.3.2
delocalization
DLE-internal process of converting synonymous DL-addresses to a canonical form for transmission or during reception (see 5.2.2)

3.3.3
DL-address user-request queue
multi-priority FIFO-within-priority queue, associated with a specific DL-address of a DLE, of in-process DLS-user-requests partitioned into three disjoint sections:

a) those requests which have already been transmitted but remain available for retransmission, confirmation, or both – that is, which have not reached their maximum confirm delay and which
   - are still within the DLCEP’s transmit window, or
   - are awaiting an acknowledging response DLPDU at the DLSAP;

b) those requests which are ready for transmission but have not been completely transmitted;

c) those requests which are either awaiting a DL-COMPEL-SERVICE request to release them for transmission, or are outside the DLCEP’s transmit window, or both

Note 1 to entry: Section 1 of the queue contains DLS-user requests which may have been successfully communicated to another DLS-user. Retransmission of all or part of the DLS-user data associated with these requests may be required, and can be attempted before the requests are purged from this partition (by reset or peer acknowledgment on a peer DLC; by reset or user-request timeout or the need for sequence-number reuse on a multi-peer DLC).

Note 2 to entry: Section 2 of the queue contains DLS-user requests which are ready for transmission, but could not have been completely communicated to another DLS-user.

Note 3 to entry: Section 3 of the queue can be non-empty only when the DL-scheduling-policy for the DL-address is EXPLICIT, or when the number of queued DLSDUs exceeds the DLCEP’s transmit window, or both.

Note 4 to entry: Members of a given priority are advanced from one section to another until they are removed from the queue. (In practice, the section partitions may be moved, rather than the members.) The FIFO ordering within each priority is strictly maintained.

Note 5 to entry: One such queue is associated with each DLSAP-address, each peer or publisher DLCEP-address, and each subscriber’s DLCEP (which can be considered to be associated with an implicit DLCEP-address) of the DLE, and with the DLE’s node DL-address (see 3.3.1 and 5.2.2.3).

3.3.4
DL-segment, link
local link
single DL-subnetwork in which any of the connected DLEs may communicate directly, without any intervening DL-relaying, whenever all of those DLEs that are participating in an instance of communication are simultaneously attentive to the DL-subnetwork during the period(s) of attempted communication

3.3.5
DLE unscheduled-service queue
multi-priority FIFO-within-priority queue of

a) references to DL-address user-request queues (see 3.3.3);
b) references to locally-scheduled active sequences (see Clause 11) resulting from DL-SCHEDULE-SEQUENCE requests (see 8.4.3.1);
c) DT DLPDUs containing DLS-user data which are delayed responses to received CD and ED DLPDUs, queued in support of the DL-UNITDATA-EXCHANGE service

Note 1 to entry: See 4.7.1.17 a) for a more elaborate definition.

Note 2 to entry: Since this is a multi-priority FIFO-within-priority queue, members are removed in priority order, and within a single priority, in FIFO order.

3.3.6
DLSAP
distinctive point at which DL-services are provided by a single DL-entity to a single higher-layer entity

Note 1 to entry: This definition, derived from ISO/IEC 7498-1, is repeated here to facilitate understanding of the critical distinction between DLSAPs and their DL-addresses. (See Figure 1.)

3.3.7
DL(SAP)-address
either an individual DLSAP-address, designating a single DLSAP of a single DLS-user, or a group DL-address potentially designating multiple DLSAPs, each of a single DLS-user

Note 1 to entry: This terminology is chosen because ISO/IEC 7498-3 does not permit the use of the term DLSAP-address to designate more than a single DLSAP at a single DLS-user

3.3.8
(individual) DLSAP-address
DL-address that designates only one DLSAP within the extended link

Note 1 to entry: A single DL-entity may have multiple DLSAP-addresses associated with a single DLSAP.

3.3.9
DLCEP-address
DL-address which designates either

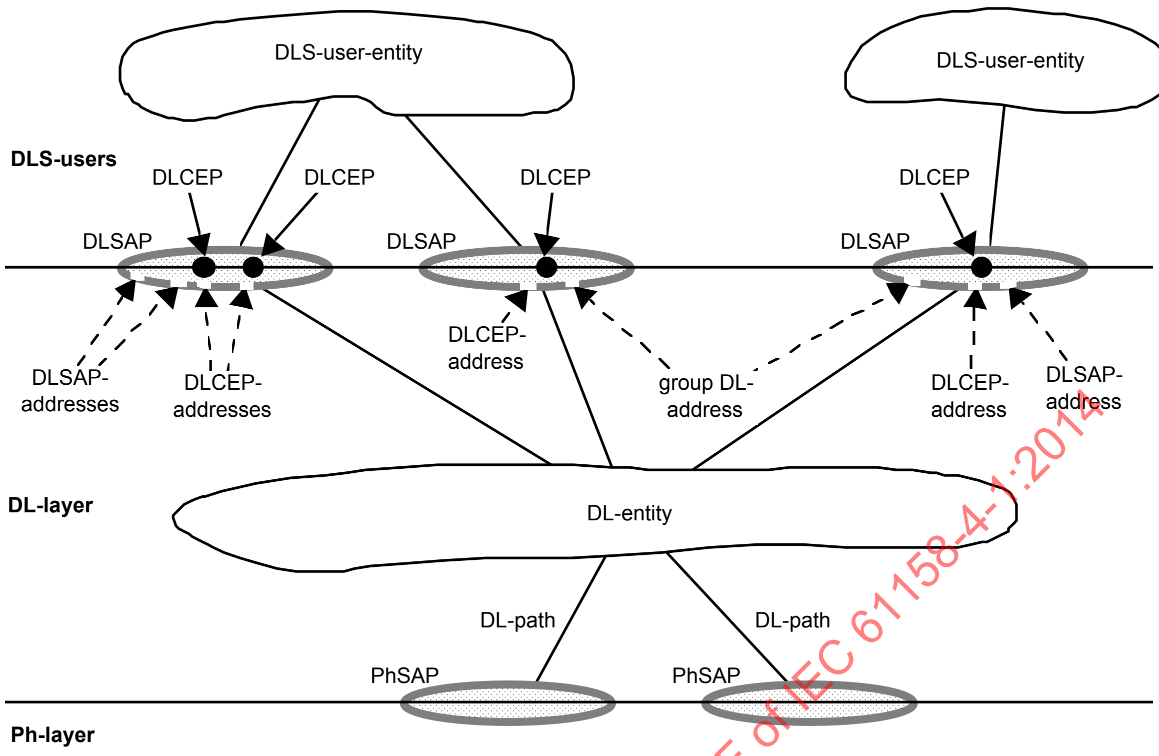
a) one peer DL-connection-end-point; or
b) one multi-peer publisher DL-connection-end-point, and implicitly the corresponding set of subscriber DL-connection-end-points

where each DL-connection-end-point exists within a distinct DLSAP and is associated with a corresponding distinct DLSAP-address

Note 1 to entry: This is an extension of the use of DL-addresses beyond that specified in ISO/IEC 7498-3. (See Figure 1.)

3.3.10
DLSEP-address
DL-address which designates a DL-scheduling-end-point within a DLE

Note 1 to entry: This is an extension of the use of DL-addresses beyond that specified in ISO/IEC 7498-3.



NOTE 1 DLSAPs and PhSAPs are depicted as ovals spanning the boundary between two adjacent layers.

NOTE 2 DL-addresses are depicted as designating small gaps (points of access) in the DLL portion of a DLSAP. A DLCEP-address also designates a specific point of information flow (its DLCEP) within the DLSAP.

NOTE 3 A single DL-entity may have multiple DLSAP-addresses and group DL-addresses associated with a single DLSAP.

NOTE 4 This figure also shows the relationships of DL-paths and PhSAPs.

Figure 1 – Relationships of DLSAPs, DLSAP-addresses, DLCEPs, DLCEP-addresses, DLSEP-addresses and group DL-addresses

3.3.11
dominant token
unique right to initiate the next transmission on the local link (see token)

Note 1 to entry: A reply token is the dominant token on the local link during the period after its creation and before its expiration or return, and the reply token holder (the responder) holds the unique right to transmit.

Note 2 to entry: Otherwise, when no reply token exists, a delegated token is the dominant token on the local link during the period after its creation and before its expiration or return, and its token holder (the initiator) holds the unique right to transmit.

Note 3 to entry: At all other times the scheduler token is the dominant token on the local link, and the LAS DLE (which functions as the initiator) holds the unique right to transmit

3.3.12
extended link
DL-subnetwork, consisting of the maximal set of links interconnected by DL-relays, sharing a single DL-name (DL-address) space, in which any of the connected DL-entities may communicate, one with another, either directly or with the assistance of one or more of those intervening DL-relay entities

Note 1 to entry: An extended link may be composed of just a single link.

3.3.13
fractional-duty-cycle
FDC
DLE which is not continuously attentive to received signaling, usually found in an end-system with an austere electrical power budget

Note 1 to entry: Such a DLE generally requires the assistance of a bridge (DL-relay) DLE to communicate with other DLEs on the same link, to provide for cases where the two DLEs are not concurrently active.

Three classes of FDC DLE have been identified:

class A – FDC DLEs which are totally inattentive to their local links during their “sleep” periods, and control the timing of their temporary “re-awakening” to link communications solely on the basis of internal DL-time or some other measurement;

class B – FDC DLEs which monitor the link during their periods of “sleep” for a particular wakeup DLPDU (see 6.21) addressed specifically to the DLE’s node DL-address, “re-awakening” upon receipt of such a DLPDU;

class C – FDC DLEs which monitor the link during their periods of “sleep” for any DLPDUs addressed specifically to the DLE’s node DL-address, “re-awakening” upon receipt of such a DLPDU.

Note 2 to entry: DLEs that “sleep” between receipt of DLPDUs, but that respond to all DLPDUs addressed to any of their DL-addresses, are not FDC DLEs.

3.3.14
frame
denigrated synonym for DLPDU

3.3.15
group DL-address
DL-address that potentially designates more than one DLSAP within the extended link

Note 1 to entry: A single DL-entity can have multiple group DL-addresses associated with a single DLSAP. A single DL-entity also can have a single group DL-address associated with more than one DLSAP

3.3.16
immediate reply
DLPDU sent by the responder in a transaction as a solicited reply to the immediately prior DLPDU

Note 1 to entry: The original DLPDU sent by the initiator of the transaction specifies the constraints on the replying DLPDU.

3.3.17
immediate-response-recovery-delay
measure, similar to maximum-response-delay, of the worst-case period of peer PhE-inactivity which can be observed

a) by a DLE holding a delegated token while awaiting an immediate response from another correctly functioning DLE on the local link;
b) by any other DLE on the local link, after receiving the initiating transaction of a two-DLPDU transaction, before detecting the link activity caused by transmission of that immediate response DLPDU;
c) by the LAS DLE while waiting for the initial link activity after
   – a reply token (via a PROBE NODE-ADDRESS (PN) DLPDU);
   – a delegated token (via a PASS TOKEN (PT) or EXECUTE SEQUENCE (ES) DLPDU);
   – the scheduler token (via a TRANSFER LAS (TL) DLPDU)

   to another correctly functioning DLE on the local link

Note 1 to entry: More formally, immediate-response-recovery-delay provides a bound for the worst-case delay between receipt of the Ph-Data confirm primitive for a Ph-Data request primitive whose PhIDU specified end-of-data-and-activity (see 4.4.3), and subsequent receipt of the next start-of-activity PhIDU (see 4.4.3), that can be observed by the sending DLE on the local link during

1) an initiator-responder interaction, in which a transaction-initiating DLE sends a DLPDU requiring an immediate reply to a responding DLE, and that responding DLE replies by sending a second DLPDU;

2) a token-passing interaction, in which the LAS DLE sends a delegated token DLPDU to the next intended token-holding DLE, or the scheduler token to a successor (as LAS) DLE, and that addressed DLE responds by sending a second DLPDU.

Note 2 to entry: The value of immediate-response-recovery-delay, in units of one slot-time, is (maximum-response-delay + 1).

3.3.18
initiator
DLE role in which a DLE sends a DLPDU to a peer responder DLE, which immediately sends a reply DLPDU back to the initiator DLE (and potentially to other DLEs) as part of the same transaction

Note 1 to entry: Some prior national standards have referred to this role as a "master" role.

3.3.19
Link Active Scheduler
LAS
special-purpose elective role of a DLE which schedules the local link and serves as the local source of DL-time for the link

Note 1 to entry: The LAS functions exist within each link master (LM) DLE, which contends for and then activates those LAS functions after detecting the absence on the link of a DLE whose LAS functions are active. The DLE serving as LAS, sometimes referred to as the LAS DLE, receives, and responds to, scheduling requests from all DLEs on the link, including itself. It also receives and responds to requests for the current DL-time.

3.3.20
link-id
two-octet primary identifier for the local link, within the extended link, whose values are constrained as specified in 4.3.2.1

3.3.21
Link Master
LM
DLE which can also provide the LAS functions for the link, including initializing and scheduling the link

3.3.20.1
maximum-inactivity-to-claim-LAS-delay
configuration parameter of each local link, its minimum value is the worst-case internal delays of any link master DLE and associated PhEs connected to that link

Note 1 to entry: This delay is computed as the sum of the following two components:

a) the internal delay between

   1) the moment of presentation of the last non-silent PhPDU of a transmission to the DLE's associated PhE at that PhE's point of connection to the local link;

   2) the resulting start of the DLE's internal timer which monitors link inactivity;

b) the internal delay between

   1) the expiration of that timer;

   2) the resulting moment of presentation of the first non-silent PhPDU of a transmission of the required Claim Las (CL) DLPDU by the DLE's associated PhE at that PhE's point of connection to the local link

The unit of measurement of this aggregate delay is one eighth of the transmission period of one octet (that is, one nominal "bit-period"). The range of values of this parameter is 1 to 4 095 nominal "bit-periods".

3.3.22
maximum-response-delay
measure, in units of one slot-time, greater than the worst-case period of local PhE-inactivity which can be observed by a DLE

a) which has just received a delegated token, before initiating transmission of a DLPDU;
b) which has just received a reply token, before initiating transmission of the required immediate reply DLPDU.

Note 1 to entry: A DLE's minimum required value for maximum-response-delay is determined by measuring at the responding DLE the delay between concluding the reception of a requesting DLPDU and initiating the transmission of the immediately-subsequent responding DLPDU.

Note 2 to entry: Maximum-response-delay is a configuration parameter of each local link, and has a value of 1 to 11. When multiplied by the duration of one slot-time, the product must be at least as large as the largest of the maximum-response-delay values required by each of the current or anticipated DLEs on the local link.

3.3.23
multi-peer DLC
centralized multi-end-point DL-connection offering DL-duplex-transmission between a single distinguished DLS-user, known as the publisher or publishing DLS-user, and a set of peer but undistinguished DLS-users, known collectively as the subscribers or subscribing DLS-users, where the publishing DLS-user can send to the subscribing DLS-users as a group (but not individually), and the subscribing DLS-users can send to the publishing DLS-user (but not to each other)

Note 1 to entry: A multi-peer DLC always provides asymmetrical service. It may also be negotiated to provide only DL-simplex service, either from the publisher to the subscribers, or from the subscribers to the publisher. In this last case, the characterizations as publisher and subscriber are misnomers.

Note 2 to entry: The publishing DLS-user may need to employ control of its publishing rate, because a subscribing DLS-user cannot exert either flow or rate control on its publishing peer entity. Similar considerations apply to subscribing DLS-users with respect to their sending DLSDUs to the publishing DLS-user.

3.3.24
node
single DL-entity as it appears on one local link

3.3.25
NODE DL-address
DL-address which designates the (single) DL-entity associated with a single node (see 3.3.23) on a specific local link

3.3.26
node-id
one-octet primary identifier for the DLE on the local link, whose values are constrained as specified in 4.3.2.2

Note 1 to entry: A value of zero is also permitted, which inhibits all transmission by this DLE.

3.3.27
node-timer
frequency-adjustable counter, maintained by the local DLE, which is used to provide the DLS-user with a local multi-partite sense of DL-time (see 4.5.4 and 8.4.1.1) such that one component of that DL-time forms a monotonically-increasing sense of local time available for use within the attached end system

Note 1 to entry: Conceptually, the node-timer counts in nominal units of 2-13 ms and has a period of over 100 years. Therefore, any actual counter shall be a binary counter whose least significant bit has a nominal weight of 2±N ms, with a rollover period greater than the expected maximum interval between resets of the DLE

When no information about that maximum interval is available, an interval of five years may be assumed.

Note 2 to entry: The node-timer is also used within the DL-protocol to provide a shared sense of DL-time which is used both to synchronize DLE scheduling actions, where appropriate, and to synchronize the rate of drift of all of the node-timers on the extended link. This latter is achieved by adjusting the frequency of each DLE's node-timer such that the DLE's monotonic sense of local time maintains an approximately constant phase relationship with that of the DLE serving as the time-reference DLE.

This adjustment is the reason why the weight of the node-timer unit of counting is only nominally $2^{\pm N}$ ms.

3.3.28
peer DLC
point-to-point DL-connection offering DL-duplex-transmission between two peer DLS-users where each can be a sending DLS-user, and each as a receiving DLS-user may be able to exert flow control on its sending peer

Note 1 to entry: A peer DLC is negotiated to provide either symmetrical service or asymmetrical service. A peer DLC may also be negotiated to provide only DL-simplex service.

3.3.29
receiving DLS-user
DL-service user that acts as a recipient of DL-user-data

Note 1 to entry: A DL-service user can be concurrently both a sending and receiving DLS-user.

3.3.30
responder
DLE role in which a DLE sends a DLPDU as an immediate reply to a DLPDU received from a peer initiator DLE, all as part of a single transaction

Note 1 to entry: Some prior national standards have referred to this role as a "slave" role.

3.3.31
sending DLS-user
DL-service user that acts as a source of DL-user-data

3.3.32
slot-time
configuration parameter of each local link, measured as an integral multiple of the transmission period of one complete octet, with an integral value of between 1 and 4 095 octets:

Note 1 to entry: Slot-time is a fundamental link parameter with multiple uses:

– Slot-time is used by each link master DLE connected to the link in determining the amount of time for which that DLE monitors the link for inactivity before sending a CLAIM LAS (CL) DLPDU. Slot-time is defined such that the nominal link-inactivity monitoring periods of two DLEs which have consecutive NODE DL-addresses and which do not hold any token differ by exactly one slot-time.

– Slot-time is used by each DLE on the local link to compute the durations of all other periods of link inactivity which the DLE must monitor. All such durations are specified as standardized or configured multiples of one slot-time.

– Slot-time is a configured minimum upper bound on the maximum two-way asynchronism in immediate communications among interacting DLEs on the local link when trying to (re)initialize the link, maximized across all pairs of DLEs on that local link. Given this viewpoint, slot-time is an aggregate measure of the worst-case implementation delays within the intervening media, the PhL, and the PhL/DLL interfaces, all of which limit the rapidity of two-way DLE interaction on the local link.

Slot-time is computed as the sum of

a) the worst-case two-way propagation delays through the intervening media and intervening PhEs, such as repeaters, between any two PhEs associated with their respective DLEs on the local link, measured between the point of connection to the local link of each of those PHEs, including worst-case internal logic and analog circuit delays in any intermediate repeaters;

b) the maximum-inactivity-to-claim-LAS-delay as defined in 3.3.32;

c) a safety factor, which is used to account for

    1) the difference in two relevant internal delays consisting of

        i) the delay between

            A) the presentation of the first non-silent PhPDU to the DLE's associated PhE at that PhE's point of attachment,

B) the indication of start of activity from that PhE to that DLE;

ii) the delay between

A) the presentation of the last non-silent PhPDU to the DLE's associated PhE at that PhE's point of attachment,

B) the indication of end of activity from that PhE to that DLE;

2) the rate differences in the internal timer clocks among the DLEs on the local link;

3) the limited resolution of the measurements and potential measurement errors;

4) any extra delay needed to make the sum of a), b) and c) round to an integral multiple of the transmission period of one octet.

These delays are defined in a manner that permits their measurement.

Finer resolution than one octet is not possible without knowledge of the specific associated PhL, because Ph-timing comes from the PhL and the formal PhL-DLL interface provides only octet timing (see 4.4).

The definitions of 3.3.31 to 3.3.34 are not recursive, because 3.3.31 is based on a different measurement than 3.3.21 to 3.3.34. The definitions of 3.3.21 to 3.3.34 are in units of one slot-time to reduce the complexities of implementations of this protocol. As a result, a single 8-bit hardware timer, prescaled by the slot-time, enabled on bus inactivity, with usage-specific stopping and retriggering conditions, can provide the required functionality for all of the slot-time-based timers of this protocol.

3.3.33
timeliness
DL-timeliness
attribute of a datum which provides an assessment of the temporal currency of that datum

Note 1 to entry: This attribute is of particular importance in sampled-data systems, which may need to make decisions based on the timeliness, or lack of timeliness, of current data samples.

Note 2 to entry: As a general rule, timeliness is a user attribute which can be affected negatively by the various layers of the data transport system. That is, a datum which was timely when the requesting user presented it to a data communications subsystem for transmission may become untimely due to delays in the communications subsystem.

Note 3 to entry: DL-timeliness is an attribute of a DLS-user datum relating the timing of a DLS-user/DLE interaction which writes or reads that datum to one or more other DLS-user/DLE interactions.

Note 4 to entry: These concepts also support migration from previous national standards.

3.3.34
token
right to transmit on the local link

Note 1 to entry: This right is assumed by a DLE when it activates its LAS functions. This right may be delegated to individual DLEs, subject to specified constraints on its usage. In all cases, this right ultimately reverts to the DLE which has activated its LAS functions (the LAS DLE). Each token is implicitly qualified by the method of its transmission or assumption:

a) A *scheduler token* is assumed and held by the LAS DLE, and can be sent to another LM DLE on the local link to transfer the activation of LAS functions to that receiving DLE.

b) A *delegated token* is created by the LAS DLE and sent to a DLE on the local link, and is returned upon completion of its use, or assumed by the LAS DLE at its expiration.

c) A *reply token* is created by the current delegated token holder (or scheduler token holder if there is no delegated token holder) and sent to a DLE on the local link, requesting an immediate reply; it is returned with that immediate reply, or assumed by the current delegated token holder at the expiration of the reply period

Note 2 to entry: It is possible, though not absolutely required, for a DLE to function simultaneously as an LAS, an initiator, and a responder, delegating a token and receiving that token, requesting an immediate reply and receiving that request, and then replying to itself as requestor and returning the token to itself as LAS. This would require the transmission of at least three DLPDUs – one from each role.

3.3.35
token-recovery-delay
measure, similar to maximum-response-delay, greater than the worst-case period of peer PhE-inactivity which can be observed by the LAS DLE while another correctly-functioning DLE is using a token

Note 1 to entry: More formally, token-recovery-delay provides a bound greater than the worst-case delay between receipt of an end-of-activity or end-of-data-and-activity PhIDU (see 4.4.4) and subsequent receipt of the next start-of-activity PhIDU, that can be observed by any DLE on the local link during normal link operation. This delay can be exceeded only when a DLE fails while holding a delegated or scheduler token.

Note 2 to entry: The value of token-recovery-delay, in units of one slot-time, may be any value between (maximum-response-delay + 3) and 14.

3.3.36
transaction
single DLPDU, or a sequence of two immediately consecutive related DLPDUs, resulting from a single DLS-user request

Note 1 to entry: The DLE sending the first DLPDU of the transaction is known as the initiator; the DLE which sends the second DLPDU of the transaction, if any, is known as the responder.

Note 2 to entry: A DL-entity can be both an initiator and a responder in the same transaction.

3.4 Symbols and abbreviations

3.4.1 Data units

3.4.1.1	**BPDU**	bridge protocol data unit, used in the ISO/IEC 10038 inter-bridge protocol
3.4.1.2	**pDLSDU**	partial DL-service data unit – one segment of a multi-segment DLSDU
3.4.1.3	**SPDU**	support protocol data unit, used to support the full DL-protocol

3.4.2 Local variables, timers, counters and queues

3.4.2.1	**V(ST)**	slot time	See 4.7.1.1
3.4.2.2	**V(PhLO)**	per DLPDU PhL overhead	See 4.7.1.2
3.4.2.3	**V(MRD)**	maximum response delay	See 4.7.1.3
3.4.2.4	**V(IRRD)**	immediate response recovery delay	See 4.7.1.4
3.4.2.5	**V(MRC)**	maximum retry count	See 4.7.1.5
3.4.2.6	**V(NRC)**	network retry count	See 4.7.1.6
3.4.2.7	**V(NDL)**	network DLPDU lifetime	See 4.7.1.7
3.4.2.8	**V(TN)**	this node	See 4.7.1.8
3.4.2.9	**V(TL)**	this link	See 4.7.1.9
3.4.2.10	**V(MEP)**	DL MAC address embedding prefix	See 4.7.1.10
3.4.2.11	**C(RD)**	remaining duration down-counter	See 4.7.1.11
3.4.2.12	**V(MID)**	minimum inter-DLPDU delay	See 4.7.1.12
3.4.2.13	**T(IRRD)**	immediate response recovery delay monitor	See 4.7.1.13
3.4.2.14	**V(RA)**	reply address	See 4.7.1.14
3.4.2.15	**V(OTA)**	outstanding transaction array	See 4.7.1.15
3.4.2.16	**V(LTI)**	last transaction index	See 4.7.1.16
3.4.2.17	**Q(US)**	unscheduled service queue	See 4.7.1.17

3.4.2.18	**V(RID)**	random identifier	See 4.7.1.18
3.4.2.19	**C(NT)**	node time up-counter	See 4.7.1.19
3.4.2.20	**V(LSTO)**	local link scheduling time offset	See 4.7.1.20
3.4.2.21	**V(DLTO)**	DL-time offset	See 4.7.1.21
3.4.2.22	**V(TQ)**	time quality	See 4.7.1.22
3.4.2.23	**V(MD)**	measured delay	See 4.7.1.23
3.4.2.24	**V(LN)**	LAS node	See 4.7.1.24
3.4.2.25	**V(TSC)**	time synchronization class	See 4.7.1.25
3.4.2.26	**T(TDP)**	time distribution period monitor	See 4.7.1.26
3.4.2.27	**V(TSL)**	time source link	See 4.7.1.27
3.4.2.28	**P_U(SDUL)**	DLSDU length request parameter	See 4.7.2.1
3.4.2.29	**P_U(SDU)**	DLSDU request parameter	See 4.7.2.2
3.4.2.30	**P_U(MCD)**	maximum confirm delay parameter	See 4.7.2.3
3.4.2.31	**T_U(MCD)**	maximum confirm delay monitor	See 4.7.2.4
3.4.2.32	**Q_A(UR)**	user request queue	See 4.7.3.1
3.4.2.33	**V_C(ST)**	DLCEP state	See 4.7.4.1
3.4.2.34	**V_C(NP)**	negotiated DLCEP parameters	See 4.7.4.2
3.4.2.35	**V_C(N)**	next sequence number to assign to a DLSDU	See 4.7.4.3
3.4.2.36	**V_C(R)**	maximum non-transmittable DLSDU sequence number	See 4.7.4.4
3.4.2.37	**V_C(A)**	maximum acknowledged DLSDU sequence number	See 4.7.4.5
3.4.2.38	**V_C(M)**	minimum untransmitted DLSDU sequence number	See 4.7.4.6
3.4.2.39	**V_C(MS)**	minimum untransmitted segment number	See 4.7.4.7
3.4.2.40	**$V_{C,K}$(SS)**	to-be-sent segments of a DLSDU	See 4.7.4.8
3.4.2.41	**$T_{C,K}$(SS)**	sent segments monitor for a DLSDU	See 4.7.4.9
3.4.2.42	**T_C(SS)**	simplified sent segments monitor for a DLSDU	See 4.7.4.9.1
3.4.2.43	**V_C(L)**	last reported DLSDU sequence number	See 4.7.4.10
3.4.2.44	**V_C(H)**	highest detected DLSDU sequence number	See 4.7.4.11
3.4.2.45	**V_C(HS)**	highest detected segment number of the highest detected DLSDU sequence number	See 4.7.4.12
3.4.2.46	**$V_{C,K}$(MRS)**	missing received segments of a DLSDU	See 4.7.4.13

3.4.2.47	**$V_{C,K}$(RRS)**	retransmission request required segments of a DLSDU	See 4.7.4.14
3.4.2.48	**$T_{C,K}$(RRS)**	retransmission request monitor for a DLSDU	See 4.7.4.15
3.4.2.49	**T_C(RAS)**	residual activity stimulus	See 4.7.4.16
3.4.2.50	**T_C(RAM)**	residual activity monitor	See 4.7.4.17
3.4.2.51	**V_C(TNA)**	DL-time of last network access	See 4.7.4.18
3.4.2.52	**V_B(TW)**	DL-time of last buffer write	See 4.7.4.19
3.4.2.53	**V_B(TP)**	DL-time of production	See 4.7.4.20
3.4.2.54	**V_B(TS)**	timeliness status of buffer write	See 4.7.4.21
3.4.2.55	**V(DTA)**	delegation address	See 4.7.5.1
3.4.2.56	**V(LL)**	local link live list	See 4.7.5.2
3.4.2.57	**V(TCL)**	token circulation list	See 4.7.5.3
3.4.2.58	**V(ENRL)**	expected non-response list	See 4.7.5.4
3.4.2.59	**V(MST)**	maximum scheduled traffic	See 4.7.5.5
3.4.2.60	**V(MSO)**	maximum scheduling overhead	See 4.7.5.6
3.4.2.61	**V(DMDT)**	default minimum token delegation time	See 4.7.5.7
3.4.2.62	**V(DTHT)**	default token holding time	See 4.7.5.8
3.4.2.63	**V(LTHT)**	link maintenance token holding time	See 4.7.5.9
3.4.2.64	**V(MTHA)**	maximum token holding time array	See 4.7.5.10
3.4.2.65	**V(TTRT)**	target token rotation time	See 4.7.5.11
3.4.2.66	**V(ATRT)**	actual token rotation time	See 4.7.5.12
3.4.2.67	**V(RTHA)**	remaining token holding time array	See 4.7.5.13
3.4.2.68	**V(NTHN)**	next token holding node	See 4.7.5.14
3.4.2.69	**V(FUN)**	first unpolled node id	See 4.7.5.15
3.4.2.70	**V(NUN)**	number of consecutive unpolled node ids	See 4.7.5.16
3.4.2.71	**P(TRD)**	token recovery delay	See 4.7.5.17
3.4.2.72	**V(TDP)**	time distribution period	See 4.7.5.18
3.4.2.73	**V(MICD)**	maximum inactivity to claim LAS delay	See 4.7.5.19
3.4.2.74	**V(LDDP)**	LAS data base distribution period	See 4.7.5.20
3.4.2.75	**V(ML)**	maximum link	See 4.7.6.1

3.4.3 DLPDU classes

3.4.3.1	**CA**	compel acknowledgment	See 6.4
3.4.3.2	**CD**	compel data	See 6.5
3.4.3.3	**CL**	claim LAS	See 6.19
3.4.3.4	**CT**	compel time	See 6.9
3.4.3.5	**DC**	disconnect connection	See 6.2
3.4.3.6	**DT**	data	See 6.7
3.4.3.7	**EC**	establish connection	See 6.1
3.4.3.8	**ED**	exchange data	See 6.6
3.4.3.9	**ES**	execute sequence	See 6.16
3.4.3.10	**IDLE**	idle	See 6.22
3.4.3.11	**PN**	probe node address	See 6.13
3.4.3.12	**PR**	probe return	See 6.14
3.4.3.13	**PT**	pass token	See 6.15
3.4.3.14	**RC**	reset connection	See 6.3
3.4.3.15	**RI**	request interval	See 6.18
3.4.3.16	**RQ**	round trip delay query	See 6.11
3.4.3.17	**RR**	round trip delay reply	See 6.12
3.4.3.18	**RT**	return token	See 6.17
3.4.3.19	**SR**	status response	See 6.8
3.4.3.20	**TD**	time distribution	See 6.10
3.4.3.21	**TL**	transfer LAS	See 6.20
3.4.3.22	**WK**	wakeup	See 6.21

3.4.4 Miscellaneous

3.4.4.1	**BME**	bridge management entity
3.4.4.2	**DLC**	DL-connection
3.4.4.3	**DLP-**	DL-protocol (as a prefix)
3.4.4.4	**DLPCI**	data link protocol control information
3.4.4.5	**DLSEP**	DL-schedule endpoint
3.4.4.6	**FC**	frame control (first octet of each DLPDU)
3.4.4.7	**FDC**	fractional duty cycle (type of DLE)
3.4.4.8	**LAS**	link active scheduler
3.4.4.9	**LM**	link master

3.4.4.10 PhICI physical interface control information

3.4.4.11 PhID physical interface data

3.4.4.12 SLAE systems load application entity

3.4.4.13 SMAE systems management application entity

4 Overview of the DL-protocol

4.1 Three-level model of the DLL

The DLL is modeled as

a) a low-level of path-access-and-scheduling functions, which supports,
b) an intermediate-level of bridge-operation functions, which in turn supports,
c) a higher-level of connection-mode and connectionless data transfer, bridge-coordination, and DL-service functions.

Interoperating with all three levels are DL-management functions, including bridge and redundant-path management functions where relevant.

NOTE 1 The term “sublayer” is not appropriate for describing these levels, since ISO/IEC 7498 requires that when multiple sublayers are defined, all but one of them must be optional.

NOTE 2 This three-level partitioning closely resembles the partitioning into lower-level “MAC”, intermediate-level “bridge-operation”, and higher-level “LLC” functions found in the ISO/IEC LAN standards (see ISO/IEC TR 8802-1 for the MAC service definition), with the following two significant differences.

a) This protocol’s low-level functionality is contained entirely within the Data Link layer as specified by ISO/IEC 7498. In contrast, the “MAC’ functionality of the ISO/IEC LAN protocols spans the lower part of the OSI Data Link Layer and the upper part of the OSI Physical Layer.
b) This protocol employs a single level of DL-addressing within the Data Link layer. In contrast, the ISO/IEC LAN protocols employ two levels of DL-addressing, one within the “MAC” and “bridge-operation” functionality and a second within the “LLC” functionality.

4.1.1 Path access and scheduling level

The path-access-and-scheduling level provides basic communication from one DLE to another. The timing of such communication is regulated

a) to provide an equitable distribution of opportunities for arbitrary communications to all DLEs on the local link, generally in a cyclic but asynchronous manner;
b) to provide a specific distribution of opportunities for designated communications to DLEs on the local link in accord with a pre-established schedule for the link.

It is the latter requirement for scheduled communications, in support of time-critical communications, which causes this protocol to rely more heavily on centralized real-time management of the communications opportunities than prior protocols such as ISO/IEC 8802-3, ISO/IEC 8802-4 and ISO/IEC 8802-5. The centralized manager is elected from the set of operating full-function DLEs on the local link by a procedure similar to that used in ISO/IEC 8802-4 and ISO/IEC 8802-5. Should that manager fail, a replacement is similarly elected.

NOTE Many traditional control algorithms used in the domain of applicability of this protocol require tight constraints on the cycle-to-cycle timing of information acquisition. In distributed systems such constraints frequently translate to limits on the permissible cycle-to-cycle periodicity of the messaging which conveys that repetitively acquired information.

c) The path-access-and-scheduling level forms each DLPDU from DL-protocol control information and DLS-user data; computes and appends an appropriate frame check sequence (FCS); and passes the whole as a sequence of PhIDUs (see 4.4) to the PhE for transmission to peer PhEs for reporting to peer DLEs.

In some cases it also appends the low-order three octets either

– of the current value of the local node-timer, C(NT); or
– of a computed value minus the current value of the local node-timer, C(NT),

during DLPDU formation, immediately preceding the appended FCS.

d) The path-access-and-scheduling level receives a sequence of PhIDUs from the PhE, concatenates those PhIDUs into a received DLPDU, computes a frame check sequence over the entire sequence of received data, and checks for the proper residual value. The first octet of the received sequence is examined to determine the type of the DLPDU, and an attempt is made to parse that DLPDU into its DL-protocol control information and DLS-user data components. If the FCS residual was correct and the parse was successful, then the appropriate low-level actions are performed, possibly including reporting the parsed DLPDU to a higher level.

   In some cases the value of the low-order three octets of the local node-timer, C(NT), at the time of receipt, is appended to this parsed DLPDU.

e) The path-access-and-scheduling level provides the basic functions of both responder and initiator. As a responder, it provides the sequencing functions necessary
   1) for receiving a DLPDU, possibly conveying a reply token;
   2) in that latter case (of the received reply token), for sending a DLPDU as an immediate reply to the just-received DLPDU.

   As an initiator, it provides the sequencing functions necessary for
   3) receiving a delegated token;
   4) sending one or more DLPDUs, including those requiring an immediate reply;
   5) receiving such an immediate reply, or inferring its absence;
   6) returning a delegated token.

f) The path-access-and-scheduling level provides the low-level scheduling functionality required for scheduling DLPDU transmissions on a specific path, including any interactions with the local link's link active scheduler (LAS) to coordinate the schedule with other DLEs or to request needed path transmission capacity.

The actions of c) and d) are augmented within a bridge relay DLE) to permit the retransmission of a received sequence of data octets, including the received FCS, with possible constrained alterations of the first octet and compensating alterations (see 5.2.5.3) of the received FCS, prior to retransmission.

The actions of e) and f) are based in part on two request-management and scheduling queues:

- a DL-address-specific request queue see 3.3.3), associated with the sending address, which is used to manage DLS-user requests originated from that DL-address;
- the common DLE unscheduled-service queue see 4.7.1.17), which is used to manage the servicing of unscheduled requests upon receipt of the “circulated” token.

Some of the more complex scheduling functionality of this level requires, and uses, the services of the upper level, see 4.1.3) by acting as a DLE-internal quasi-DLS-user.

4.1.2 Bridge-operation level

The bridge-operation level provides the intermediate-level functionality of

a) logically inter-connecting multiple local links into a single extended-link by physically interconnecting multiple paths;
b) serving as a possibly-distributed “time-space-time” switch:

1) providing DLPDU store-and-forward functions to permit communication between DLEs on the extended link which could not otherwise communicate;

NOTE This includes the coordination with fractional-duty-cycle FDC) DLEs (see 3.3.32) necessary to permit alternating periods of “FDC-DLE-awake’ and “FDC-DLE-asleep” operation.

2) providing surrogate functions to permit delayed-reply interactions between DLEs on the extended link as an extrapolation of “immediate-reply” transactions on a local link;

c) providing a shared sense of DL-time throughout the extended link;

d) coordinating local-link scheduling among two or more local links to provide any necessary multi-link coordination of scheduling within the extended link.

Much of the bridge-operation level of functionality is active only in “bridge” DLEs (see 4.6).

4.1.3 Connection-mode and connectionless data transfer, bridge-coordination, and DL-service level

The connection-mode and connectionless data transfer, bridge-coordination, and DL-service level provides the higher-level functionality of

a) managing all DLE interactions with the DLS-user, converting all DLS-user request and response primitives into the necessary sequence of DLE operations, and generating DLS-user indication and confirm primitives where appropriate;

b) managing the sequencing of each active DLCEP, including

   1) determination of the type and sequence of DLPDUs to be transmitted from the DLCEP;

   2) QoS negotiation;

   3) determination of the DLPCI to be included in each DLPDU;

   4) segmentation and reassembly of large DLSDUs;

c) managing the sequencing of unitdata transactions which require a response from a peer DLE, including

   1) determination of the DLPCI to be included in each DLPDU;

   2) correlation of a non-immediate reply DLPDU with the requesting transaction;

d) processing all inter-DLE state-information-distribution DLPDUs, such as TIME DISTRIBUTION (TD) (see 6.10 and 8.4.1) and bridge configuration (see 10.4) DLPDUs and LAS-backup SPDUs (see Clause 9);

e) managing all query-DLPDU/reply-DLPDU interactions with other DLEs, other than those which occur as an “immediate reply” and which are made on a reactive basis by the lower-level functions of 4.1.1.

These query-DLPDU / reply-DLPDU interactions also include computation of round-trip-delays (see 6.11, 6.12, and 8.4.1), support of remotely-initiated DL-SUBSCRIBER-QUERY (see 8.2.3) and DL-LISTENER-QUERY (see 8.3.4) requests, and inter-bridge exchanges of bridge state information (see 10.4).

4.2 Service provided by the DLL

The DLL provides connectionless data transfer services for limited-size DLSDUs, connection-mode data transfer services for limited-size DLSDUs, an internally synchronized time service, scheduling services to control the time allocation of the underlying shared PhL service, and a DL(SAP)-address, queue and buffer management service.

NOTE IEC 61158-3-1 shows many of the relationships among DLC QoS attributes.

Some relevant QoS attributes are as follows:

4.2.1 QoS – DLCEP class

Each DLCEP establishment request specifies the class of the DLCEP. The three choices for DLCEP-class are

a) **PEER** – the DLS-user can exchange DLSDUs with one other peer DLS-user;

b) **PUBLISHER** – the DLS-user can send DLSDUs to a set of zero or more associated subscribing DLS-users, and receive DLSDUs from any of those subscribing DLS-users;

c) **SUBSCRIBER** – the DLS-user can receive, and request, DLSDUs from the associated publishing DLS-user, and can send DLSDUs to that publishing DLS-user.

4.2.2 QoS – DLCEP data delivery features

Both members of a peer DLC, or the publishing DLS-user of a multi-peer DLC, specify the data delivery features of the DLC's DLCEP(s). The five choices for DLCEP data delivery features, and their effects, are

a) **CLASSICAL** – the DLS-user can send data which will be delivered without loss, duplication or misordering. All relevant DLS-users will be notified of any loss of synchronization on the DLC.

b) **DISORDERED** – the DLS-user can send data which will be delivered immediately upon receipt, without duplication but potentially in a different order than that of the sending DLS-user. All relevant DLS-users will be notified of any unrecoverable loss of DLS-user data or loss of synchronization on the DLC.

c) **ORDERED** – the DLS-user can send data which will be delivered immediately upon receipt, without duplication or misordering, but with potential loss of some DLS-user data. Loss of DLS-user data will not be reported, and recovery of DLS-user data lost prior to the last-reported DLS-user data will not be attempted.

d) **UNORDERED** – the DLS-user can send data which will be delivered immediately upon receipt. Loss, duplication and misordering of DLS-user data will not be detected or reported. No attempt will be made by the DLS-provider to recover from such events.

e) **NONE** – the DLS-user cannot send data in this direction of data transfer.

On a peer DLC, the QoS value for the sending DLCEP data delivery features may be chosen independently for each direction of data transfer. On a multi-peer DLC, the QoS value for the DLCEP data delivery features for the subscribers-to-publisher direction of data transfer is restricted to UNORDERED and NONE. The default QoS value for the DLCEP data delivery features in the publisher-to-subscribers direction is UNORDERED.

4.2.3 QoS – DLL priority

All DLCEP establishment requests and responses, all connectionless data transfer requests, and many DL-scheduling requests, specify an associated DLL priority used in scheduling DLL data transfer services. This DLL priority also determines the maximum amount of DLS-user data that can be conveyed in a single DLPDU. The three DLL priorities with their corresponding ranges of conveyable DLS-user data (per DLPDU) are, from highest priority to lowest priority:

a) **URGENT** – ≤ 64 DLS-user octets per DLPDU;

b) **NORMAL** – ≤ 128 DLS-user octets per DLPDU;

c) **TIME-AVAILABLE** – ≤ 256 DLS-user octets per DLPDU.

NOTE 1 URGENT and NORMAL are considered **time-critical** priority levels; TIME-AVAILABLE is considered a **non-time-critical** priority level.

NOTE 2 DLC establishment and DLC release DLPDUs which are sent at TIME-AVAILABLE priority are restricted to convey no more DLS-user data than would be permitted at NORMAL priority – 128 octets.

4.2.4 QoS – DLPDU authentication

Each DLCEP establishment negotiation, and each connectionless data transfer, uses this attribute to determine

a) a lower bound on the amount of DL-addressing information used in the DLPDUs that provide the associated DLL data transfer services;

   NOTE This has a slight impact on the residual rate of DLPDU misdelivery; more addressing information reduces the potential for misdelivery.

b) whether the current state of a sending peer or publisher DLCEP should be sent at low-frequency to the DLC's peer or subscriber DLCEP(s) even when there are no unconfirmed DLS-user requests outstanding at the sending DLCEP;

   NOTE This continuing background transmission is known as **residual activity**.

c) whether all related scheduling actions should be executed locally.

NOTE These last two aspects are of particular importance in safety systems.

The three levels specifiable, with their amounts of DL-addressing information, are

1) **ORDINARY** – each DLPDU shall include the minimum amount of addressing information necessary;
2) **SOURCE –** each DLPDU shall include a source DL-address where possible;
3) **MAXIMAL** – each DL-address shall include the maximal amount of addressing information possible. Also, all related scheduling actions should be executed locally; and each sending peer or publisher DLCEP of the DLC should maintain a low-frequency report of state information when there is no DLS-user activity.

4.2.5 QoS – DLL maximum confirm delay

Each DLCEP establishment request, and each response, specifies upper bounds on the maximum time duration permitted for the completion

a) of a DL-CONNECT, DL-RESET or DL-SUBSCRIBER-QUERY primitive, and, separately;

b) of a DL-DATA primitive.

Each connectionless service request specifies an upper bound on the maximum time duration permitted for the completion

c) of a locally-confirmed DL-UNITDATA primitive, and, separately;

d) of a remotely-confirmed DL-UNITDATA primitive, a DL-LISTENER-QUERY primitive, or an instance of the DL-UNITDATA-EXCHANGE service.

Each parameter either has the value UNLIMITED or specifies an interval, in units of 1 ms, from 1 ms to 60 s, inclusive. The value UNLIMITED provides compatibility with prior OSI protocols, and provides a means for DL-CONNECT requests to remain in a "listening" or "half-open" state. The completion status of "timeout" cannot occur on a DLS-user request which specifies UNLIMITED.

The parameters for the DL-DATA and locally-confirmed DL-UNITDATA primitives specify intervals less than or equal to that for the DL-CONNECT, DL-RESET, DL-SUBSCRIBER-QUERY, remotely-confirmed DL-UNITDATA, and DL-LISTENER-QUERY primitives.

The intervals specified are the maximum permissible delays

- between the issuing of the specified request primitives and the issuing of the corresponding confirm primitives;
- between the initiation and completion of a single instance of the specified publishing or unitdata-exchange service.

NOTE For DLEs that do not support a time resolution of 1 ms, the requested time interval may be rounded up to the next-greatest multiple of that resolution that the DLE does support, or to approximately 60 s if the DLE has no sense of time.

Failure to complete a DL-CONNECT or DL-RESET request within the specified interval shall result in a DLS-provider initiated release of the DLCEP, and possibly of the DLC.

4.2.6 QoS – DL-scheduling-policy

For each DLSAP-address, and each DLCEP, the DLS-user can override the normal (implicitly-scheduled) DLL policy of providing the requested DL-service as soon as possible, and instead can defer any inter-DLS-user communication required by a DL-DATA or DL-UNITDATA request DLS-primitive until that deferral is released by an involved DLS-user. Each such release, by execution of a DL-COMPEL-SERVICE request, specifying the DLSAP-address or DLCEP, permits the completion of just a single deferred request DLS-primitive. Only DL-services that provide DLS-user intercommunication are affected by this attribute.

The two choices are

a) **IMPLICIT** – any required communications with peer DLS-user(s) from this DLSAP-address, or from this DLCEP, will occur as soon as possible;
b) **EXPLICIT** – any required data or unitdata communications with peer DLS-user(s) from this DLSAP-address, or from this DLCEP, will occur only when the deferral is explicitly released by an involved DLS-user.

NOTE Scheduling of DLPDU transmission to support the DL-CONNECT, DL-RESET and DL-DISCONNECT services, and to support responder-deferred replies in the DL-UNITDATA-EXCHANGE and remotely-confirmed DL-UNITDATA services, is always IMPLICIT. Scheduling of DLPDU transmission to initiate the DL-UNITDATA-EXCHANGE service is always EXPLICIT.

4.2.7 QoS – maximum DLSDU sizes

Each DLC / DLCEP establishment request, and each response, specifies an upper bound on the size (in octets) of DLSDUs which will be offered for transmission, and an upper bound on the size of DLSDUs which are acceptable for reception.

For peer DLCs, the negotiated maximum DLSDU size shall be determined independently for each direction of data transfer as the smallest of that offered by the sender, that permitted by the sender's local DL-management, that permitted by the receiver's local DL-management, and that permitted by the receiver.

The sender's offered size may be any value between zero and 16 times the maximum number of DLS-user octets per DLPDU, as specified in 4.2.3. The receiving DLE and all DL-management agents shall choose their maximum permitted sizes from the following list of sizes:
0, 64, 128, 256 $\times$ N where $1 \leq N \leq 16$.

NOTE 1 The maximum size DLSDU supported by this protocol is 16 times the maximum number of octets of DLS-user data conveyable in a single DLPDU, as determined by the DLC's DLL Priority (see 4.2.3).

NOTE 2 The set of maximum permitted DLSDU sizes is limited to the above small list of values to promote interoperation. The sender's maximum specified size is permitted to take any value within the range permitted by the DLC's DLL priority to facilitate optimization of the shared communications capacity of the DLL.

NOTE 3 A value of zero (0) corresponds to the choice of simplex service, as specified by the DLS-user by the choice NONE as described in 4.2.2e).

For multi-peer DLCs, the negotiated maximum DLSDU size shall be the smaller of that offered by the publisher and that permitted by the publisher's local DL-management. For subscribers of multi-peer DLCs, the DLC shall be refused by the DLS-provider (the subscriber's DLE) if the maximum DLSDU size established by the publisher is larger than the smaller of that permitted by the subscriber and that permitted by the subscriber's local DL-management.

The publisher's offered size in the publisher-to-subscribers direction may be any value between zero and 16 times the maximum number of DLS-user octets per DLPDU, as specified in 4.2.3. The publisher's offered size in the subscriber-to-publisher direction may be any value between zero and the maximum number of DLS-user octets per DLPDU, as specified in 4.2.3.

The subscribers and all DL-management agents shall choose their maximum permitted sizes from the following list of sizes: 0, 64, 128, 256 × *N* where $1 \le N \le 16$.

NOTE 4 The maximum size DLSDU supported by this protocol in the publisher-to-subscribers direction is 16 times the maximum number of octets of DLS-user data conveyable in a single DLPDU, as determined by the DLC's DLL Priority (see 4.2.3).

NOTE 5 The maximum size DLSDU supported by this protocol in the subscriber-to-publisher direction is the maximum number of octets of DLS-user data conveyable in a single DLPDU, as determined by the DLC's DLL Priority (see 4.2.3).

NOTE 6 The set of maximum permitted DLSDU sizes is limited to the above small list of values to promote interoperation. The publisher's maximum specified DLSDU size is permitted to take any value within the range permitted by the DLC's DLL priority to facilitate optimization of the shared communications capacity of the DLL.

NOTE 7 A value of zero (0) corresponds to the choice of simplex service, as specified by the DLS-user by the choice NONE as described in 4.2.2e).

The default value for both the sender's and receiver's maximum DLSDU size is the maximum number of DLS-user octets which can be carried by a single DLPDU of the specified DLL priority. The DLS-provider shall always support this DLSDU size.

4.2.8 QoS – DLCEP and DLSAP-address buffer-and-queue bindings

Each DLCEP establishment request, and each response, can bind one or two local retentive buffers or specified-depth FIFO queues, created by DL-CREATE buffer and queue management primitives (or by DL-management), to the DLCEP:

> NOTE When these bindings are made for a DLS-user of a peer DLC, or a publishing DLS-user of a multi-peer DLC, they determine the maximum transmit window (that is, number of transmitted but unacknowledged DLSDUs) for that direction of DLC data transfer. Since the size of the transmit window can also be limited by DL-management, or by an implementation, the queue depth only imposes an upper limit on the window size.

a) One queue or retentive buffer can be bound to a DLCEP to convey DLSDUs from the DLS-user to the DLS-provider.

b) One queue or retentive buffer can be bound to a DLCEP to convey DLSDUs from the DLS-provider to the DLS-user.

c) It is also possible to bind a queue or retentive buffer to be written at one DLCEP and to source data at another DLCEP. Such an intermediate buffer or queue can serve to cross scheduling boundaries or redistribute received DLS-user data to a second set of DLS-users.

Such a binding is made by specifying, for the appropriate parameter, a buffer-or-queue DL-identifier which resulted from a prior DL-CREATE request (or by DL-management), and which has not yet been deleted.

When the sending DLCEP data delivery features specify UNORDERED or ORDERED, both the sender and receiver(s) may specify a queuing policy of BUFFER-R or QUEUE. When the DLCEP's sending data delivery features specify DISORDERED or CLASSICAL, both the sender and receiver(s) may specify a queuing policy of QUEUE; a queuing policy of BUFFER-R is not permitted. A queuing policy of BUFFER-NR is never permitted.

Each DLSAP-address bind request can bind up to six local retentive buffers or non-retentive buffers or specified-depth FIFO queues, created by DL-CREATE buffer and queue management primitives (or by DL-management), to the DLSAP-address:

d) One buffer or queue can be bound to the sending direction of a DLSAP-address at each priority to convey DLSDUs from the DLS-user to the DLS-provider. Buffers can be bound only to DLSAP-addresses whose DLSAP-role is INITIATOR or CONSTRAINED RESPONDER or UNCONSTRAINED RESPONDER. Queues can be bound only to DLSAP-addresses whose DLSAP-role is BASIC.

e) One buffer or queue can be bound to the receiving direction of a DLSAP-address at each priority to convey DLSDUs from the DLS-provider to the DLS-user. Buffers can be bound only to DLSAP-addresses whose DLSAP-role is INITIATOR or CONSTRAINED RESPONDER or UNCONSTRAINED RESPONDER. Queues can be bound to all DLSAP-addresses.

4.2.8.1 Binding to a buffer

When a sending buffer is bound to a DLCEP, or to a DLSAP-address and DLL priority, by a DLS-user,

a) a DL-PUT request primitive overwrites the buffer with a DLSDU, or may set the buffer empty;

   NOTE After creation, the buffer is empty.

b) a DL-COMPEL request primitive, specifying either

   1) a DLCEP, or

   2) unitdata-exchange at a DLSAP-address

   causes the transmission, at the first opportunity, of the then most recent contents of the buffer; the primitive does not itself specify a DLSDU;

c) a DL-BUFFER-SENT indication primitive notifies the DLS-user of the specific DLCEP on which the retentive buffer was transmitted, and to which the buffer is bound, that the buffer was just transmitted;

d) a DL-UNITDATA-EXCHANGE indication primitive notifies the single DLS-user attached to the specific DLSAP-address from which the buffer was transmitted (and to which the buffer is bound) that the buffer was just transmitted on that DLSAP-address, and emptied if it is non-retentive (BUFFER-NR).

When a receiving buffer is bound to a DLCEP, or to a DLSAP-address and DLL priority, by a DLS-user,

e) a DL-BUFFER-RECEIVED or DL-UNITDATA-EXCHANGE indication primitive notifies the DLS-user of the overwriting of the buffer by a newly-received DLSDU; the primitive does not itself specify a DLSDU;

f) a DL-GET request primitive copies the DLSDU from the buffer, and empties the buffer if it is non-retentive (BUFFER-NR).

Multiple concurrent output bindings to a retentive buffer are permitted as an implementation and conformance (see 10.6.1) option, but are not required by this protocol.

4.2.8.2 Binding to a FIFO queue

When a sending FIFO queue of maximum depth *K* is bound to a DLCEP, or to a DLSAP-address and DLL priority, by a DLS-user,

a) a DL-PUT request primitive is not permitted;

b) a DL-DATA or DL-UNITDATA request primitive attempts to append a DLSDU to the queue, but fails if the queue already contains *K* DLSDUs. If the append operation was successful, then the DLSDU will be transmitted at the first opportunity, after all preceding DLSDUs in the queue.

When a receiving FIFO queue of maximum depth *K* is bound to a DLCEP, or to a DLSAP-address and DLL priority, by a DLS-user,

c) a DL-GET request primitive attempts to remove a DLSDU from the queue, but fails if the queue is empty.

d) a DL-DATA or DL-UNITDATA or DL-UNITDATA-EXCHANGE indication primitive notifies the receiving DLS-user of the result of appending a newly received DLSDU to the receive queue; the primitive does not itself specify a DLSDU.

Multiple concurrent input bindings to a FIFO queue are permitted as an implementation option, but are not required by this protocol.

4.2.8.3 Default bindings

When these binding options are not specified, or for DLS-primitives for which explicit binding is not applicable, the conventional implicitly-queued sending and direct receiving interfaces between DLS-user and DLS-provider are employed. In this case each DL-DATA and DL-UNITDATA primitive conveys a DLSDU, and each DL-CONNECT, DL-DISCONNECT and DL-RESET primitive may convey a DLSDU:

a) DL-PUT and DL-GET request primitives are not permitted;
b) A DL-DATA or DL-UNITDATA or DL-DISCONNECT request primitive, or a DL-CONNECT or DL-RESET request or response primitive, issued by the sending DLS-user attempts to append a DLSDU to the implicit queue, but fails if the queue is full. If the append operation was successful, then the DLSDU will be transmitted at the first opportunity, after all preceding DLSDUs in the queue.
c) A DL-DATA or DL-UNITDATA or DL-DISCONNECT indication primitive, or a DL-CONNECT or DL-RESET indication or confirm primitive, notifies a receiving DLS-user of a newly received DLSDU, and conveys that DLSDU to the DLS-user. No apparent queuing is provided within the DLL.

4.2.9 DL-timeliness

This attribute applies only to DL-buffers, to DLCEPs at which DL-buffers are bound, and to those DLS-primitives which transfer DLS-user data to or from DL-buffers at such DLCEPs.

Each DLCEP establishment request, and each response, can specify DL-timeliness criteria which are to apply to information sent from, or received into, buffers at that DLCEP. Four types of DL-timeliness can be supported: RESIDENCE timeliness, UPDATE timeliness, SYNCHRONIZED timeliness, and TRANSPARENT timeliness. All four types of timeliness, and the case where there is no timeliness, are shown in IEC 61158-3-1, Figure 6.

a) *RESIDENCE* timeliness is an assessment based upon the length of time that a DLS-user datum has been resident in a buffer, which is the time interval between
   1) the moment when the buffer is written (by a DL-PUT request primitive, or by reception into the buffer at a DLCEP);
   2) the moment when the buffer is read (by a DL-GET request primitive, or by transmission from the buffer at a DLCEP);

   $$\text{DL-timeliness} \equiv 0 \leq (R_T - W_T) < \Delta T \qquad (1)$$

   NOTE This type of timeliness is also known as "*Asynchronous*".

b) *UPDATE* timeliness is an assessment based upon the time interval between
   1) the moment of occurrence of a multi-DLE synchronizing event (a DL-BUFFER-RECEIVED indication or DL-BUFFER-SENT indication);
   2) the moment when the buffer is written (by a DL-PUT request primitive, or by reception into the buffer at a DLCEP);

   $$\text{DL-timeliness} \equiv 0 \leq (WT - ST) < \Delta T \qquad (2)$$

   NOTE A type of timeliness closely related to this one is also known as "*Punctual*".

c) *SYNCHRONIZED* timeliness is an assessment based upon the time intervals and timing relationships between

1) the moment of occurrence of a multi-DLE synchronizing event (a DL-BUFFER-RECEIVED indication or DL-BUFFER-SENT indication);
2) the moment when the buffer is written (by a DL-PUT request primitive, or by reception into the buffer at a DLCEP);
3) the moment when the buffer is read (by a DL-GET request primitive, or by transmission from the buffer at a DLCEP);

$$\text{DL-timeliness} \equiv 0 \leq (WT - ST) \leq (RT - ST) < \Delta T \quad (3)$$

NOTE This type of timeliness is also known as "*Synchronous*".

d) *TRANSPARENT* timeliness occurs when timeliness is selected on a DLCEP but none of the above assessments are performed. In such a case the DLC preserves any prior buffer timeliness, but does not itself invalidate that timeliness. When no prior buffer timeliness exists, the default timeliness value shall be TRUE.

e) *NO* timeliness occurs when timeliness is not selected on a DLCEP. In such a case the DL-timeliness attribute of DLS-user data always shall be FALSE.

Where a buffer read or write occurs over a significant time interval, and not just as a momentary event, then the overall timeliness of the read or write operation shall be computed as the timeliness at the beginning of the read or write operation, logically ANDed with the timeliness at the end of the read or write operation, all using the same timeliness criteria.

4.2.10 Remote-DLE-confirmed

Each unitdata transfer request can specify whether confirmation of receipt of the associated DLSDU by the (implicitly addressed) remote DLE is required. Its permissible values are **TRUE** and **FALSE**.

NOTE Selection of the value TRUE inevitably uses more link capacity than does selection of the value FALSE.

4.3 Structure and definition of DL-addresses

DL-addresses are used as DLSAP-addresses and group DL-addresses, as DLCEP-addresses, and as DLSEP-addresses. DL-addresses conform to the following structure and coding limitations.

Clause 4.3 defines the form of DL-addresses and the usage of various ranges of DL-address components. It includes specific definitions for some standard DL-addresses.

This DL-protocol uses individual (non-group) DL-addresses for other purposes than simply as DLSAP-addresses. The same terminology and the following considerations apply to those non-DLSAP DL-addresses as well.

NOTE 1 This usage extends the definition of DL-addresses beyond that specified in ISO/IEC 7498-3.

NOTE 2 DL-addresses are also addressed to some extent in 5.1.

4.3.1 Form of DL-addresses

A standard DL-address may be considered to consist of two parts: link-designator and sublink selector. The link designator is an unsigned integer, two octets in length. The sublink selector is also two octets in length.

Link designator	Sublink selector
2 octets	2 octets

Figure 2 – Basic structure of a DL-address

Most values of the link designator specify an individual link. In these cases, the sublink selector specifies a unique DL-address within the designated local link, as further described in 4.3.2.2 and 4.3.2.3. When lacking address-specific forwarding information, bridges (relay DLEs) shall base their forwarding decisions only on the link designator portion of such addresses.

However, many values of the link designator specify that the link designator portion of the address is concatenated with the sublink selector portion to form a non-hierarchical (that is, flat) address. When lacking address-specific information, bridges shall not forward such addresses.

A standard sublink selector itself consists of two subparts: node-designator and subnode selector. Each of these subparts is a small unsigned integer, one octet in length.

Implied link designator (0000)	**Node designator**	**Sub-node selector**
	1 octet	1 octet

Figure 3 – Basic structure of a sublink selector

Most values of the node designator specify an individual node. In these cases, the subnode selector specifies a unique DL-address within the designated local node.

However, some values of the node designator specify that the node designator portion of the address is concatenated with the subnode selector portion to form a non-hierarchical (that is, flat) subaddress.

This almost-hierarchical structure of link designators, node designators, and subnode selectors was chosen primarily to facilitate the administration of DL-addresses, and secondarily to reduce the number of entries in bridge address-forwarding tables.

NOTE Protocol implementations may find that this partially hierarchical structure is of use during address recognition.

The resulting structural possibilities for DL-addresses are (where "||" indicates concatenation of the nominal link, node and selector designator subfields):

Link	**Node**	**Selector**
2 octets	1 octet	1 octet

or

Link	**Node || Selector**
2 octets	2 octets

or

Link || Node || Selector
4 octets

Figure 4 – DL-address alternative structures

A DL-address in which the link designator has a value of zero is known as a *local* address because its context for interpretation is limited to the local link. In contrast, other DL-addresses are known as *extended* addresses, because their interpretation context is the entire extended link.

There are four distinct types of DLS-user-visible DL-address:

a) DLSAP-addresses, which are associated with only one DLSAP in the entire address interpretation context;

b) Group DL-addresses, which are potentially associated with many DLSAPs in one or more DLEs, or with many DLEs;

c) DLCEP-addresses, which are associated

   1) with only one DLCEP and associated DLSAP of a peer-to-peer DLC, or

   2) directly with the publishing DLCEP and associated DLSAP of a multi-peer DLC, and indirectly with all the subscribing DLCEPs and associated DLSAPs of a multi-peer DLC;

d) DLSEP-addresses, which are associated with one sequence-end-point of a single DLE.

DLSAP-addresses may be used with all of the primitives of the DLS which specify a DLSAP-address. Group DL-addresses shall not be

   1) bound to a queue as a source DLSAP-address, or

   2) used as a source DLSAP-address in any connection-oriented or connectionless DLS primitive.

Individual non-”DLSAP” DL-addresses may be used as appropriate within the DL-protocol, and where permitted as specific parameters within the DL-services.

4.3.2 Predefined values and ranges for DL-address components

Some values of the link-designator or node-designator component of DL-addresses designate a group DL-address. All other combinations of values of the link-designator and node-designator components designate individual DL-addresses. This is summarized in Table 1 through Table 3.

NOTE All link, node and selector values are specified in hexadecimal (radix 16).

4.3.2.1 Link designators

The link designator component of a DL-address usually designates a specific single local link. Values for the link designator component are predefined, or their usage is constrained, as follows:

0000 The local link. The zero (0000) value is provided as a convenience for the unification of service primitives, to enable implementations to refer to the local link’s address by a constant zero (0000), and to enable isolated-link operation when no bridge is present to identify the link component of an extended link. When the DLPDU-authentication QoS does not specify *MAXIMAL* and the DL-addresses themselves permit it, the link-designator components of all DL-addresses shall be omitted before transmission. Otherwise, the nominal-link-id of the sourcing local link, if known, shall be substituted for the zero (0000) link value before transmission.

0001 … 003F Non-local group DL-addresses. These link designator values, concatenated with the associated node designator values and selector values – *link* || *node* || *selector* – form an address space of 4 128 768 identifiers usable as group DL-addresses. The resulting DL-addresses do not implicitly designate any specific link.

Non-local group DL-addresses shall be allocated individually or in contiguous blocks. When considered as unsigned 32-bit integers, the first (lowest-valued) eight such addresses are reserved for use by the DL-protocol; the next 56 such addresses are reserved for standardized group DL-addresses. All of the other addresses are available for use as non-standardized group DL-addresses.

0040 … Non-local globally-administered group DL-addresses. These link designator values, concatenated with the associated node designator values and selector values – *link* ||

007F *node* || *selector* – form an address space of 4 194 304 identifiers for globally-administered group DL-addresses. These DL-addresses are formed by prefixing the 10-bit string “0000 0000 01” to the lower 22 bits of the Organizationally Unique Identifiers specified in the ISO/IEC 8802 family of Local Area Network standards. These DL-addresses do not implicitly designate any specific link.

Table 1 – Link || node || selector addressing

Link	N | S	Assigned use for specified DL-address range
0000	0000 … FFFF	Local link DL-addresses as specified in Table 2
0001	0000 … 0007	Flat non-local group DL-addresses reserved for DLL use
0001	0008 … 003F	Flat non-local group DL-addresses reserved for standard AEs
0001 … 003F	0040 … FFFF	Flat non-local group DL-addresses available for vendor, user and dynamic assignment
0040 … 007F	0000 … FFFF	Flat non-local globally-administered group DL-addresses – potentially one per vendor (See Note 1)
0080	0000 … 0007	Flat non-local individual “DLSAP”-addresses reserved for DLL use
0080	0008 … 003F	Flat non-local individual DLSAP-addresses reserved for standard AEs
0080 … 00FF	0040 …	Flat non-local individual DLSAP-addresses available for vendor, user and dynamic assignment
0100 … 0FFF	0000 … FFBF	Flat non-local individual DL-addresses available for vendor, user and dynamic assignment as DLCEP-addresses and DLSEP-addresses
0FFF	FFC0 … FFF7	Flat non-local individual DL-addresses reserved for standard AE DLCEP-addresses and DLSEP-addresses
0FFF	FFF8 … FFFF	Flat non-local individual DL-addresses reserved for DLL use as DLCEP-addresses and DLSEP-addresses
1000 … FEFF	0000 … FFFF	Individual links of DL-addresses as specified in Table 2
NOTE These group DL-addresses are based on the Organizationally Unique Identifiers specified in the ISO/IEC 8802 family of Local area network standards.		

0080 … 00FF	Non-local individual DLSAP-addresses. These link designator values, concatenated with the associated node designator values and selector values – *link* || *node* || *selector* – form an address space of 8 388 608 identifiers usable as individual DLSAP-addresses. The resulting DLSAP-addresses do not implicitly designate any specific link or node. Non-local individual DLSAP-addresses shall be allocated individually or in contiguous blocks. When considered as unsigned 32-bit integers, the first (lowest-valued) eight such addresses are reserved for use by the DL-protocol as “DLSAP”-addresses; the next 56 such addresses are reserved for standardized individual DLSAP addresses for standard AEs. All of the other addresses are available for use as non-standardized individual DLSAP addresses.
0100 … 0FFF	Non-local individual non-DLSAP DL-addresses usable as DLCEP-addresses and DLSEP-addresses. These link designator values, concatenated with the associated node designator values and selector values – *link* || *node* || *selector* – form an address space of 251 658 240 identifiers usable as individual DLCEP- and DLSEP-addresses. The resulting DL-addresses do not implicitly designate any specific link or node. Non-local individual non-DLSAP DL-addresses shall be allocated individually or in contiguous blocks. When considered as unsigned 32-bit integers, the last (highest-valued) eight such addresses are reserved for use by the DL-protocol; the next last 56 such addresses are reserved for standardized individual DLCEP-addresses or DLSEP-addresses for standard AEs. All of the other addresses are available for use as non-standardized individual DLCEP addresses and DLSEP-addresses.
1000 … FEFF	Designators of individual links. Link designators shall be allocated, individually or in contiguous blocks, in order of increasing address values when considered as unsigned 16-bit integers. The value of the DL-management configuration parameter V(ML) shall be greater than or equal to the largest of the link-designator values assigned to links of the extended link. NOTE This parameter is used by bridges to assist in the provision of default forwarding actions for each link.

NOTE 1 The potential link-designator values FF00 – FFFF are reserved to support the implementation of identifier-references to DL-addresses as noted in IEC 61158-3-1.

In summary, each DLE capable of recognizing LONG DL-addresses (see 5.2.2) recognizes link designator values 0001 – 0FFF as designating the extended group and individual address spaces – *link* || *node* || *selector* – common to the extended link.

Each DLE capable of recognizing LONG DL-addresses recognizes link designator value 0000 as designating the nominal-link, which is a single local link designator value $\geq$ 1 000. The resulting identifier values – *nominal-link-id* || *node* || *selector* – form an address space of up to 65 280 identifiers usable as individual and group DL-addresses as described in 4.3.2.2.

NOTE 2 The *link* || *node* || *selector* components of these address identifiers take the hexadecimal form WWWWXXYY, where 1000 $\leq$ WWWW, 01 $\leq$ XX $\leq$ FF and 00 $\leq$ YY $\leq$ FF. When 01 $\leq$ XX $\leq$ 03, such addresses are group DL-addresses; all other such addresses are individual DL-addresses. Addresses where 1 000 $\leq$ WWWW and XX = 00 refer to the local node, and are valid only within the software of that node.

When the address space needs of a given link exceed that provided by a single nominal-link-id, secondary link-ids may be allocated to that link. For each such secondary link-id, the resulting values – *secondary-link-id* || *node* || *selector* – form an additional address space of

57 600 identifiers usable as additional individual and group DL-addresses for the link as described in 4.3.2.2.

NOTE 3 The *link* || *node* || *selector* components of these additional address identifiers take the hexadecimal form WWWWXXYY, where 1 000 ≤ WWWW, and either 0140 ≤ XXYY ≤ 03FF, or 0440 ≤ XXYY ≤ 0FBF, or 10 ≤ XX ≤ FF and 08 ≤ YY ≤ F7. When 01 ≤ XX ≤ 03, such address are group DL-addresses; all other such addresses are individual DL-addresses.

DLEs which are not capable of recognizing LONG DL-addresses recognize only SHORT and VERY-SHORT addresses (see 5.2.2), which always designate the nominal-link. Such DLEs are not capable of directly communicating or being directly managed through a bridge.

4.3.2.2 Node designators

The node designator component of a DL-address usually designates a single DLE. When the link designator component does not have the value 0001 – 0FFF, then the values for the node designator component are predefined, or their usage is constrained, as follows:

Table 2 – Link-local node || selector addressing

N	S	Assigned use for specified DL-address range
00	00 … FF	Local node DL-addresses as specified in Table 4
01	00 … 07	Flat link-local group DL-addresses reserved for DLL use
01	08 … 3F	Flat link-local group DL-addresses reserved for standard AEs
01 … 03	40 … FF	Flat link-local group DL-addresses available for vendor, user and dynamic assignment
04	00 … 07	Flat link-local individual “DLSAP”-addresses reserved for DLL use
04	08 … 3F	Flat link-local individual DLSAP-addresses reserved for standard AEs
04	40 … FF	Flat link-local individual DLSAP-addresses available for vendor, user and dynamic assignment
05 … 0F	00 … BF	Flat link-local individual DL-addresses available for vendor, user and dynamic assignment as DLCEP-addresses and DLSEP-addresses
0F	C0 … F7	Flat link-local individual DL-addresses reserved for standard AE DLCEP-addresses and DLSEP-addresses
0F	F8 … FF	Flat link-local individual DL-addresses reserved for DLL use as DLCEP-addresses and DLSEP-addresses
10 … FF	00 … FF	Individual nodes of DL-addresses as specified in Table 4

00 The local DLE. Equivalent to a DLE-internal address. This designator is provided as a convenience for the unification of service primitives, and to enable implementations to refer to the local DLE address by a constant zero (00). When transmission is appropriate, an implementation shall substitute the DLE's own nominal-node-id for this zero (00) address.

NOTE 1 As a consequence, DLPDUs with a node designator component of zero (00) in a DL-address (other than a flat DL-address) are never transmitted, and are discarded upon reception.

This node designator shall not be used with secondary-link-ids.

01 … 03 Link-local group DL-addresses. These node designator values, concatenated with the nominal or a secondary link-id of the local link and the associated selector values — *link-id* || *node* || *selector* – form address spaces of 768 identifiers each, usable as group DL-addresses.

Link-local group DL-addresses shall be allocated, individually or in contiguous blocks, in order of increasing address values when considered as unsigned 16-bit integers. The first (lowest-valued) such addresses are reserved for use by the DL-protocol; the next 56 such addresses are reserved for standardized group DL-addresses. All of the other addresses are available for use as non-standardized group DL-addresses

04 Link-local individual DLSAP-addresses. These node designator values, concatenated with the nominal or a secondary link-id of the local link and the associated selector values – *link-id* || *node* || *selector* – form address spaces of 256 identifiers each, usable as individual DLSAP-addresses. The resulting DLSAP-addresses do not implicitly designate any specific node.

Link-local individual DLSAP-addresses shall be allocated, individually or in contiguous blocks, in order of increasing address values when considered as unsigned 16-bit integers. The first (lowest-valued) such addresses are reserved for use by the DL-protocol as "DLSAP"-addresses; the next 56 such addresses are reserved for standardized individual DLSAP addresses for standard AEs.

05 … 0F Link-local individual non-DLSAP DL-addresses usable as DLCEP-addresses and DLSEP-addresses. These node designator values, concatenated with the nominal or a secondary link-id of the local link and the associated selector values – *link-id* || *node* || *selector* – form address spaces of 2 816 identifiers usable as individual DLCEP- and DLSEP-addresses. The resulting DL-addresses do not implicitly designate any specific node.

Link-local individual non-DLSAP DL-addresses usable with connected DL-services shall be allocated, individually or in contiguous blocks, in order of decreasing address values when considered as unsigned 16-bit integers. The first (highest-valued) eight such addresses are reserved for use by the DL-protocol; the next 56 such addresses are reserved for standardized individual DLCEP-addresses or DLSEP-addresses for standard AEs. All of the other addresses are available for use as non-standardized individual DLCEP addresses and DLSEP-addresses.

10 … FF Designators of individual nodes. These node designator values, concatenated with the nominal link-id of the local link, provide the equivalent of a "unique hardware address" for each individual physical DLE. Node designators shall be allocated, individually or in contiguous blocks, as shown in Table 3.

Table 3 – Link-local node designators

N	Assigned use for specified node designator range
00	local DLE
01 … 0F	unusable
10 … 13	Bridge-class DLEs
14	Link-master-class DLEs
15 … F6	unused
F7	Basic-class DLEs
F8 … FB	Non-”visitor” DLEs awaiting proper node designator assignment
FC … FF	“Visitor” DLEs

00 The local DLE. Equivalent to a DLE-internal address. This designator is provided as a convenience for the unification of service primitives, and to enable implementations to refer to the local DLE address by a constant zero (00). When transmission is appropriate, an implementation shall substitute the DLE’s own nominal-node-id for this zero (00) address.

NOTE 2 As a consequence, DLPDUs with a node designator component of zero (00) in a DL-address (other than a flat DL-address) are never transmitted, and are discarded upon reception.

01 … 0F Reserved for unrelated use within the DL-protocol - not available for DLE NODE DL-addresses.

10 … 13 Reserved for allocation to DLEs which bridge (interconnect) the local link (whose link-id is the nominal link-id) to other links, and in particular to higher-capacity links.

NOTE 3 Such bridges are expected to serve as link-local managers for the DL-protocol.

14 Reserved for allocation to other DLEs which are capable of initializing and managing the local link, and which are intended to be permanently attached to the local link (not, for example, maintenance DLEs such as “hand-held communicators”).

These node designators shall be allocated, individually or in contiguous blocks, in order of increasing address values when considered as unsigned 8-bit integers.

15 … F6 Reserved for allocation to DLEs which require more local DL-addresses than are provided hierarchically by the DLE’s primary node designator.

F7 Reserved for allocation to DLEs which are not capable (or are configured to be incapable) of initializing the local link.

These node designators shall be allocated, individually or in contiguous blocks, in order of decreasing address values when considered as unsigned 8-bit integers.

F8 … FB Reserved for temporary use by other DLEs which are intended to be permanently attached to the local link while they determine their appropriate final node designator.

FC … FF Reserved for allocation to DLEs which are not intended to be permanently attached to the local link (for example, maintenance DLEs such as “hand-held communicators).

NOTE 4 Such “visitor” DLEs normally are capable of initializing and managing the local link, but are the least desirable such DLEs for this purpose since their long-term presence on the link is not expected.

When not used to designate a single DLE, node designator values between those used for DLEs capable of initializing the local link, and those which are incapable (or are configured to be incapable) of initializing the local link, can be concatenated with the nominal or a secondary link-id of the local link, and with selector components whose value is ≥ 08 and ≤ F7, to form additional identifier spaces of 240 individual DL-addresses each – *link-id* || *node* || *selector*. Each such additional address space may be assigned to a single node, or may be shared among the nodes of the link.

NOTE 5 The *node* || *selector* components of these additional address identifiers take the hexadecimal form XXYY, where 10 < XX < FC and 08 ≤ YY ≤ F7.

In summary, when the link designator value designates the local link, then each DLE recognizes node designator values 01 – 0F as designating link-local group and individual address spaces – *link-id* || *node* || *selector* – common to all nodes on the local link.

Each DLE recognizes node designator value 00 as designating a single local node designator value ≥ 10 (hexadecimal), the nominal-node. The resulting identifier values – *nominal-link-id* || *nominal-node-id* || *selector* – form a space of identifiers usable as individual DL-addresses as described in 4.3.2.3.

When the address space needs of a given node exceed that provided by a single nominal-node-id, secondary node-ids may be allocated to that node. For each such secondary node id, the resulting values – *link-id* || *secondary-node-id* || *selector* – with selector component values ≥ 08 and ≤ F7, form an additional space of 240 identifiers usable as additional individual DL-addresses for the node as described in 4.3.2.3. The same method may be used to extend the set of non-node specific individual addresses for the link.

4.3.2.3 Selectors

The selector component of a DL-address usually designates a single protocol entity within the DLE.

When the link designator component does not have the value 0001 – 0FFF, and when the node designator component does not have the value 01 – 0F, then the values for the selector component are predefined, and their usage is constrained, as specified in Table 4.

Table 4 – Node-local selector addressing

Selector	Assigned use for specified DL-address range
00 … 01	Node-local individual “DLSAP”-addresses reserved for DLL use
02 … 07	Node-local individual DLSAP-addresses reserved for standard AEs
08 … 1F	Node-local individual DLSAP-addresses available for vendor, user and dynamic assignment
20 … F7	Node-local individual DL-addresses available for vendor, user and dynamic assignment as DLCEP-addresses and DLSEP-addresses
F8 … FF	Node-local individual DL-addresses reserved for use as standard AE DLCEP-addresses and DLSEP-addresses

00 … 07	Reserved for standard node-related DLSAP-addresses, and for intra-DLE "DLSAP"-addresses.
08 … 1F	Non-standardized individual DLSAP-addresses.
20 … F7	Non-standardized DLCEP-addresses and DLSEP-addresses.
F8 … FF	Reserved for standard node-related DLCEP-addresses and DLSEP-addresses.

4.3.3 Predefined DL-addresses

As indicated in Table 1 through Table 3, some specific DL-addresses are defined in Clause 4.3.3. These DL-addresses are defined to facilitate provision of DL-services and initialization of OSI communications within a physically distributed real system.

4.3.3.1 Predefined flat non-local DL-addresses

A number of flat non-local DL-addresses are defined within this subclause, as specified in Table 5.

NOTE SLAE is the System Load Application Entity; SMAE is the System Management Application Entity.

Table 5 – Predefined flat non-local DL-addresses

Link ‖	N ‖ S	Assigned use for specified DL-address
0001	0000	The DL-support functions of "all" (see NOTE) DLEs on the extended link
0001	0001	The DL-support functions of "all" (see NOTE) LM DLEs on the extended link
0001	0002	The DL-support functions of "all" (see NOTE) bridge DLEs on the extended link
0001	0003	The DL-bridge functions of "all" (see NOTE) bridge DLEs on the extended link
0001	0008	The SMAEs of "all" (see NOTE) unconfigured DLEs on the extended link
0001	0009	The SMAEs of "all" (see NOTE) DLEs on the extended link
0001	000A	The SMAEs of "all" (see NOTE) LM DLEs on the extended link
0001	000B	The SMAEs of "all" (see NOTE) bridge DLEs on the extended link
0001	000C	The SLAEs of "all" (see NOTE) LoadServers on the extended link
0001	000D	The SLAEs of "all" (see NOTE) LoadableDevices on the extended link
0080	0000	Reserved for DLL use for a DL-support "DLSAP"-address (See NOTE)
0080	0004	The "DLSAP"-address for the DL-bridge functions of the bridge DLE on the extended link which is the root bridge of the spanning tree
0FFF	FFFE	The DLCEP-address for the DLC from the bridge functions of the bridge DLE on the extended link which is the root bridge of the spanning tree to the bridge functions in all other bridge DLEs on the extended link
0FFF	FFFF	Reserved for DLL use for a DL-support DLCEP-address or DLSEP-address (See NOTE)
NOTE DLEs which do not recognize LONG DL-addresses are necessarily excluded from these sets.		

4.3.3.2 Predefined flat local DL-addresses

A number of flat local DL-addresses are defined within this subclause, as specified in Table 6. These correspond one-for-one with the predefined flat non-local DL-addresses specified in Table 5. The extent of the addresses in Table 6 is the local link; the extent of the addresses in Table 5 is the entire extended link.

The DL-addresses in Table 6 have a one-for-one correspondence with analogous DL-addresses in Table 5; while the addresses in Table 6 refer to functions local to each link, the analogous addresses of Table 5 refer to similar functions global to the entire extended link. This addressing correspondence is intentional, and may serve to ease implementation complexity. Thus this correspondence should be preserved in future address assignments.

Table 6 – Predefined flat link-local DL-addresses

Node || selector		Assigned use for specified DL-address
01	00	The DL-support functions of all DLEs on the link
01	01	The DL-support functions of all LM DLEs on the link
01	02	The DL-support functions of all bridge DLEs on the link
01	03	The DL-bridge functions of all bridge DLEs on the link
01	08	The SMAEs of all unconfigured DLEs on the link
01	09	The SMAEs of all DLEs on the link
01	0A	The SMAEs of all LM DLEs on the link
01	0B	The SMAEs of all bridge DLEs on the link
01	0C	The SLAEs of all LoadServers on the link
01	0D	The SLAEs of all LoadableDevices on the link
04	00	The “DLSAP”-address for the DL-support functions of the DLE on the link which is serving as LAS
04	04	The “DLSAP”-address for the DL-bridge functions of the bridge DLE on the link which is dominant (closest to the root) in the bridge spanning tree
0F	FE	The DLCEP-address for the DLC from the bridge functions of the bridge DLE on the link which is dominant (closest to the root) in the bridge spanning tree to the bridge functions in all other bridge DLEs on the link
0F	FF	The DLCEP-address for the DLC from the DL-support functions of the DLE on the link which is serving as LAS to the DL-support functions of all of the other LM DLEs on the link

4.3.3.3 Predefined node-local DL-addresses

A number of node-local DL-addresses are defined within this subclause, as specified in Table 7.

Table 7 – Predefined node-local DL-addresses

Selector	Assigned use for specified DL-address
00	The “DLSAP”-address for the DL-support functions of the node’s DLE
01	The “DLSAP”-address for the DL-bridge functions of the node’s DLE
02	The DLSAP-address for the node’s same
03	The DLSAP-address for the node’s SLAE

4.3.4 Representation of DL-Addresses as locally-administered 48-bit MAC-addresses

The ISO/IEC Local Area Network protocols (ISO/IEC 8802) provide a foundation for portions of the IEC fieldbus protocols, such as the Ph-redundancy state machine, the inter-DL-relay-

(bridge)-protocol, and the system management system-load-protocol. The latter two protocols, which specify six-octet encodings for each ISO/IEC MAC-address, are being adapted for fieldbus use with the minimum of necessary changes; thus there is a need to specify the representation of fieldbus four-octet DL-addresses for use with the fieldbus adaptations of these protocols.

A six-octet ISO/IEC MAC-address, as specified in ISO/IEC TR 8802-1, contains, in order of transmission

a) an initial bit, I/G, indicating whether the MAC-address is an individual MAC-SAP-address (I/G=0) designating a single MAC-sublayer-entity, or a group MAC-address (I/G=1) designating a group of zero or more MAC-sublayer-entities;
b) a second bit, U/L, indicating whether the MAC-address contains a globally administered component (U/L=0), or is completely locally administered (U/L=1);
c) a 22-bit globally administered component and a 24-bit vendor-administered component when U/L=0, or a 46-bit user-administered component when U/L=1 (see Figure 5).

Globally administered MAC-addresses			
I/G	U/L	Globally-administered	Vendor-administered
1 b	1 b	22 bits	24 bits

Locally administered MAC-addresses		
I/G	U/L	User-administered
1 b	1 b	46 bits

Figure 5 – Basic structure of MAC-addresses

Since DL-addresses are not administered so as to be globally unique they shall be represented as six-octet locally administered MAC-addresses as shown in Figure 6.

A DL-address as a locally administered MAC-address			
I/G	U/L	Set by DL-management	Extended DL-address
1 b	1 b	14 bits	32 bits

Figure 6 – Representation of a DL-address as a MAC-address

The 14-bit MAC-address component set by DL-management shall be set equal to the value of the extended-link's *DL-MAC-address-embedding-prefix* parameter, V(MEP), which shall be distributed as part of the extended-link's configuration information. The default value for this parameter shall be hexadecimal 3FFF (all ones), but it shall be changeable by DL-management to permit more than one extended link to be directly addressable from inter-working ISO/IEC local area networks.

Each 32-bit DLSAP-, DLCEP- and DLSEP-address shall be represented as a six-octet MAC-address by prefixing the 32-bit non-group DL-address with the binary string "01" followed by the extended-link's 14-bit DL-MAC-address-embedding-prefix, V(MEP).

Each 32-bit group DL-address shall be represented as a six-octet MAC-address by prefixing the 32-bit group DL-address with the binary string "11" followed by the extended-link's 14-bit DL-MAC-address-embedding-prefix, V(MEP).

In addition, should there be a need to specify a six-octet MAC-address for the group of multi-peer subscriber DLCEPs associated with a specified multi-peer publisher DLCEP, then the implicit address of that group of subscriber DLCEPs shall be represented as a six-octet MAC-address by prefixing the publisher's 32-bit individual DLCEP-address with the binary string "11" followed by the extended-link's 14-bit DL-MAC-address-embedding-prefix, V(MEP).

4.4 Service assumed from the PhL

This subclause defines the assumed Physical Service (PhS) primitives and their constraints on use by the DLE.

Proper layering requires that an (N+1)-layer entity not be concerned with, and that an (N)-service interface not overly constrain, the means by which an (N)-layer provides its (N)-services. Thus the Ph-service interface does not require DLEs to be aware of internal details of the PhE (for example, preamble, postamble and DLPDU delimiter signal patterns, number of bits per baud), and should not prevent the PhE from using appropriate evolving technologies.

4.4.1 Assumed primitives of the PhS

The granularity of transmission in the fieldbus protocol is one octet. This is the granularity of PhS-user data and timing information exchanged at the PhL – DLL interface.

4.4.1.1 PhS characteristics reporting service

The PhS is assumed to provide the following service primitive to report essential PhS characteristics used in DLL transmission, reception, and scheduling activities:

Ph-CHARACTERISTICS indication (minimum-data-rate, framing-overhead)

where

minimum-data-rate – specifies the effective minimum rate of data conveyance in bits/second, including any timing tolerances, of any PhE on the local link.

NOTE 1 A PhE with a nominal data rate of 1 Mbit/s ± 0,01 % would specify a minimum data rate of 0,9999 Mbit/s.

framing-overhead – specifies the maximum number of bit periods, where period 1/(data rate), used in any transmission for PhPDUs which do not directly convey data (for example, PhPDUs conveying preamble, DLPDU delimiters, postamble, inter-DLPDU "silence", and so on).

NOTE 2 If the framing overhead is F and two DL message lengths are L_1 and L_2, then the time to send two immediately consecutive messages of lengths L_1 and L_2 will be at least as great as the time required to send one message of length $L_1 + F + L_2$.

If this framing-overhead is more than the DLE's configured per-DLPDU-PhL-overhead, V(PhLO), then the DLE shall report this discrepancy to DL-management and shall not issue Ph-DATA requests while the discrepancy exists.

NOTE 3 This restriction prohibits DLE transmission while this discrepancy exists. The DLE's local station management may remedy this discrepancy by reconfiguring V(PhLO) to a greater value.

4.4.1.2 PhS transmission and reception services

The PhS is assumed to provide the following service primitives for transmission and reception:

Ph-DATA request (class, data);

Ph-DATA indication (class, data);

Ph-DATA confirm (status)

where

class – specifies the Ph-interface-control-information (PhICI) component of the Ph-interface-data-unit (PhIDU). For a Ph-DATA request, its possible values are

START-OF-ACTIVITY – transmission of the PhPDUs which precede Ph-user data should commence;

DATA – the single-octet value of the associated data parameter should be transmitted as part of a continuous correctly-formed transmission;

END-OF-DATA-AND-ACTIVITY – the PhPDUs which terminate Ph-user data should be transmitted after the last preceding octet of Ph-user data, culminating in the cessation of active transmission;

For a Ph-DATA indication, its possible values are:

START-OF-ACTIVITY – reception of an apparent transmission from one or more PhEs has commenced;

DATA – the associated data parameter was received as part of a continuous correctly-formed reception;

END-OF-DATA – the ongoing continuous correctly formed reception of Ph-user data has concluded with correct reception of PhPDUs implying END-OF-DATA;

END-OF-ACTIVITY – the ongoing reception (of an apparent transmission from one or more PhEs) has concluded, with no further evidence of PhE transmission;

END-OF-DATA-AND-ACTIVITY – simultaneous occurrence of END-OF-DATA and END-OF-ACTIVITY;

data – specifies the Ph-interface-data (PhID) component of the PhIDU. It consists of one octet of Ph-user data to be transmitted (Ph-DATA request) or which was received successfully (Ph-DATA indication).

status – specifies either success or the locally detected reason for inferring failure.

The Ph-DATA confirm primitive provides the critical physical timing feedback necessary to inhibit the DLE from starting a second transmission before the first is complete. The final Ph-DATA confirm of a transmission shall not be issued until the PhE has completed the current transmission.

4.4.2 Notification of PhS characteristics

The PhE has the responsibility for notifying the DLE of those characteristics of the PhS which are relevant to DLE operation. This notification is accomplished by the PhE by issuing a single Ph-CHARACTERISTICS indication primitive at each of the PhE's PhSAPs at PhE startup.

4.4.3 Transmission of Ph-user data

The PhE determines the timing of all transmissions. When a DLE has a DLPDU to transmit, and the DL-protocol gives that DLE the right to transmit, then the DLE shall send the DLPDU, including a concatenated FCS, by making a well-formed sequence of Ph-DATA requests, consisting of a single request specifying START-OF-ACTIVITY; followed by three to 300 consecutive requests, inclusive, specifying DATA; and concluded by a single request specifying END-OF-DATA-AND-ACTIVITY.

The PhE signals its completion of each Ph-DATA request, and its readiness to accept a new Ph-DATA request, with a Ph-DATA confirm primitive; the status parameter of the Ph-DATA confirm primitive conveys the success or failure of the associated Ph-DATA request. A second Ph-DATA request should not be issued until after the Ph-DATA confirm corresponding to the first request has been received from the PhE.

4.4.4 Reception of Ph-user data

The PhE reports a received transmission with a well-formed sequence of Ph-DATA indications, which shall consist of either

a) a single indication specifying START-OF-ACTIVITY; followed by consecutive indications specifying DATA; followed by a single indication specifying END-OF-DATA; and concluded by a single indication specifying END-OF-ACTIVITY, or
b) a single indication specifying START-OF-ACTIVITY; followed by consecutive indications specifying DATA; followed by a single indication specifying END-OF-DATA-AND-ACTIVITY, or
c) a single indication specifying START-OF-ACTIVITY; optionally followed by one or more consecutive indications specifying DATA; and concluded by a single indication specifying END-OF-ACTIVITY.

NOTE This last sequence is indicative of an incomplete or incorrect reception. Detection of an error in the sequence of received PhPDUs, or in the PhE's reception process, disables further Ph-DATA indications with a class parameter specifying DATA, END-OF-DATA, or END-OF-DATA-AND-ACTIVITY until after both the end of the current period of activity and the start of a subsequent period of activity have been reported by Ph-DATA indications specifying END-OF-ACTIVITY and START-OF-ACTIVITY, respectively.

In the first two cases, the DLE concatenates the received data and then attempts to parse it into a DLPDU followed by a concatenated FCS. In the last case the DLE discards all reported data and reports the event to DL-management.

4.5 Functions of the DLL

4.5.1 Overview of functions

The functions of the DLL are those necessary to bridge the gap between the services available from the PhL and those offered to DLS-users. When used in a normal OSI environment, the necessary functions of the DLL are those specified in ISO/IEC 8886. When used in a Time Critical OSI environment, the necessary functions are a superset of those specified in ISO/IEC 8886; the enhancements are primarily in

a) the availability of a confirm primitive and a service confirmation deadline for each connection-oriented and connectionless data-transfer DL-service;
b) the addition of connectionless receiving-DLE-acknowledgment and two-way data-exchange services to the basic OSI data-transfer DL-service;
c) the ability to defer and schedule data-transfer DL-services;
d) the efficient distribution of DLS-user data from a publishing DLS-user to a set of subscribing DLS-users;
e) the efficient convergence of independent DLS-user data from a set of subscribing DLS-users, delivered to a single publishing DLS-user;
f) the provision of a synchronized sense of internal time among the DLEs and available to the DLS-users of the extended link;
g) the standardized availability of local DL(SAP)-address, buffer and queue management capabilities.

4.5.2 Connectionless data transfer functions

The purpose of the connectionless data transfer functions is to transport DLSDUs of limited size between one DLS-user and one or more other DLS-users on the same link, without the requirement for establishing or maintaining a DLC with each of those other DLS-users. This purpose is achieved by means of transmission of DLPDUs providing transfer of a limited amount of user data to one or many DLS-users, with limited protection against loss of the DLSDU, or misordering of successively-transmitted DLPDUs. Both DLSDU transfer and DLSDU exchange services are provided.

4.5.3 Connection-oriented functions

Connection-oriented functions provide for the establishment, use, resynchronization and abrupt termination of a connection between DLS-users on an extended link. The type of the connection may be selected to support user data flow either

a) bidirectionally between two peer DLS-users,

b) unidirectionally from one peer DLS-user to another,

c) bidirectionally between one publishing DLS-user and the set of zero or more subscribing DLS-users,

d) unidirectionally from one publishing DLS-user to the set of zero or more subscribing DLS-users, or

e) unidirectionally from the set of zero or more subscribing DLS-users to the publishing DLS-user.

The features of the connection may be selected to support transfer of DLSDUs of a negotiated maximum size, with either

1) reliable in-sequence non-duplicated delivery with reset on DLSDU loss;

2) reduced delay, potentially out-of-sequence but non-duplicated delivery, and reset on DLSDU loss;

3) minimal delay with in-sequence non-duplicated delivery, but with potential DLSDU loss;

4) minimal delay, unsequenced delivery, potential duplication, potential but improbable DLSDU misordering, and no notification of DLSDU loss; or

5) no transfer in one specified direction.

Only cases 4) and 5) are permitted for the subscriber-to-publisher direction of a multi-peer DLC.

4.5.3.1 Connection establishment phase

The purpose of the connection establishment phase is

a) to establish a DLC between two DLS-users,

   NOTE The establishment of a publishing DLC is best modeled as the concurrent independent pair-wise establishment of the DLC between the common publisher and each separate subscriber.

b) to determine QoS attributes of the DLC;

c) to distinguish between DLCs.

4.5.3.2 Data transfer phase

The purpose of the data transfer phase is

a) to transport DLSDUs between two DLS-users connected by a DLC. This purpose is achieved by transmission of DATA (DT) and related DLPDUs (see 6.7, 6.6 and 6.4), which may involve segmenting of DLSDUs for conveyance in multiple DLPDUs and reassembly by the destination DLE;

b) to resynchronize the flow of DLSDUs between the DLS-users, and notify those DLS-users of information loss after an unrecovered error.

These concepts are defined in IEC 61158-3-1.

4.5.3.3 Connection termination phase

The purpose of the connection termination phase is to terminate abruptly a connection between two or more DLS-users and convey the reason for the termination.

4.5.4 Time synchronization function

The purpose of the time synchronization function is to provide a shared approximately-synchronized internal time reference for all DLS-users, consisting of two components:

a) a component which increases monotonically with time, with a value of zero at the startup of the local end system;

b) a second component which, when added to the first, causes the sum to be approximately equal to the corresponding sums of other correctly-functioning DLEs on the extended link.

Table 8 – Correlation of DLPDUs with functional classes

Functional class			Code	DLPDU name	Subclause
Basic	**Link master**	**Bridge**			
R[a] S[a]	R[a] S[a]	R[a] S[a]	EC	Establish Connection	6.1
R[a] S[a]	R[a] S[a]	R[a] S[a]	DC	Disconnect Connection	6.2
R[a] S[a]	R[a] S[a]	R[a] S[a]	RC	Reset Connection	6.3
R S	R S	R S	CA	Compel Acknowledgment	6.4
R S	R S	R S	CD	Compel Data	6.5
R S	R S	R S	ED	Exchange Data	6.6
R S	R S	R S	DT	Data	6.7
R —	R S	R S	SR	Status Response	6.8
– s[b]	R s[b]	R s[b]	CT	Compel Time	6.9
r[b] —	R[b] S	R[b] S	TD	Time Distribution	6.10
R s[b]	R s[b]	R s[b]	RQ	Round-trip-delay Query	6.11
r[b] S	r[b] S	r[b] S	RR	Round-trip-delay Reply	6.12
R —	R S	R S	PN	Probe Node DL-address	6.13
– S	R[c] S	R[c] S	PR	Probe Response	6.14
R —	R S	R S	PT	Pass Token	6.15
R —	R S	R S	ES	Execute Sequence	6.16
– S	R[c] S	R[c] S	RT	Return Token	6.17
– S	R[c] S	R[c] S	RI	Request Interval	6.18
– —	R S	R S	CL	Claim LAS	6.19
– —	R S	R S	TL	Transfer LAS	6.20
r —	r S	– S	WK	Wakeup	6.21
– s[d]	– s[d]	– s[d]	IDLE	Idle	6.22
– —	– —	– —	spare	Spare	6.23

where

r The specified class permits the ability to receive and act upon the specified type of DLPDU.

R The specified class always provides the ability to receive and act upon the specified type of DLPDU.

s The specified class permits the ability to send the specified type of DLPDU.

S The specified class always provides the ability to send the specified type of DLPDU.

– The specified class does not permit the specified type of DLPDU.

[a] Mandated support requires only minimal elements of procedure.

[b] All shall be supported when the DLE has a time-synchronism class (see 10.6.3 other than NONE.

[c] Shall respond to DLSDU contents when acting as LAS.

[d] No class is required to receive the IDLE DLPDU; its intended receiver is a specialized link monitor or analyzer, beyond the scope of this standard.

4.6 Functional classes

In this standard, a DLE's functional class determines its capabilities for autonomous DLL activity, and thus the minimum complexity of conforming implementations. Each class includes all lower-numbered classes. The three functional classes, in order of increasing complexity, are

a) Basic;

b) Link Master (LM);

c) Bridge.

All functional classes support all DLS-user services and are completely interoperable. The DLPDUs associated with elements of procedure for each functional class are specified in Table 8.

4.6.1 Basic class

The basic class includes the basic protocol elements of procedure necessary

a) to provide interoperability when responding to DLPDUs sent by a DLS-peer or a bridge DLE;

b) to initiate, reset and terminate DLCs with a peer DLE, to support the orderly conveyance of DLSDUs;

c) to send and receive connectionless and connection-oriented DLSDUs, and to reply to received DLSDUs where required;

d) to request services from the LAS;

e) to execute an uninterrupted sequence of link operations;

f) to optimize local use of the link.

This class is the minimum necessary for fieldbus interoperability.

4.6.2 Link master class

The link-master class includes all of the functions of the basic class. It also includes the protocol elements of procedure necessary

a) to cooperate with similar DLEs in establishing and sharing mastership of the link;

b) to detect the absence of an LAS on the link and activate the LAS functions within its own node,

and when providing the functions of the LAS,

c) to maintain an ordered access to the shared link communications resource, responding to requests from other DLEs for use of that shared resource;

d) to serve as the source of internal time for the other DLEs on the link.

This class is necessary for autonomous operation on the link. At least one DLE on the link shall operate in this class.

4.6.3 Bridge (DL-relay) class

The bridge class includes all of the functions of the link-master class. It also includes the protocol elements of procedure necessary

a) to enable communications between DLEs on a single link which are themselves periodically incapable of communicating directly on the link (that is, fractional duty cycle (FDC) DLEs), and in some cases serving as a surrogate for a DLE on the local link;

b) to interconnect two or more local links, by bridging them into an extended link, and in some cases serving as a surrogate for a remote DLE on one of the bridge's local links;

c) to provide a common sense of DL-internal time coordinated across the extended link.

This class is necessary when interconnecting two or more local links to form a multi-link extended link, or when one or more DLEs on the local link are fractional duty cycle (FDC) DLEs. When a multi-link extended link exists, the individual local links shall be interconnected only by DLEs which operate in this class.

4.7 Local parameters, variables, counters, timers and queues

This standard uses DLS-user request parameters P(...) and local variables V(...) as a means of clarifying the effect of certain actions and the conditions under which those actions are valid, local timers T(...) as a means of monitoring actions of the distributed DLS-provider and of ensuring a local DLE response to the absence of those actions, and local counters C(...) for performing rate measurement functions. It also uses local queues Q(...) as a means of ordering certain activities, of clarifying the effects of certain actions, and of clarifying the conditions under which those activities are valid.

Unless otherwise specified, at the moment of their creation or of DLE activation,

– all variables shall be initialized to their default value, or to their minimum permitted value if no default is specified;
– all counters shall be initialized to zero;
– all timers shall be initialized to inactive;
– all queues shall be initialized to empty.

DL-management may change the values of configuration variables.

4.7.1 Parameters, variables, counters, timers and queues to support the basic class

The parameter, variables, timer, counter and queue defined in 4.7.1.1 to 4.7.1.18 are required in all DLEs. The variables, counter and timer defined in 4.7.1.19 to 4.7.1.27 are required in all DLEs except those with a time synchronism class (see 10.6.3) of NONE.

4.7.1.1 V(ST) slot-time

V(ST) is used by the DLE to record the link's slot-time (see 3.3.31), which is a configuration parameter of the local link. Its range is 1 to 4 095, and its unit is the transmission duration of one octet.

4.7.1.2 V(PhLO) per-DLPDU-PhL-overhead

V(PhLO) is used by the DLE to account for the PhL-induced delay between the end of the last octet of one DLPDU as it appears on the link, and the beginning of the first octet of any other DLPDU as it appears on the link, measured in units of one octet-duration. Its range is 2 to 63.

4.7.1.3 V(MRD) maximum-response-delay

V(MRD) is used by the DLE to record the link's maximum-response-delay (see 3.3.21), which is a configuration parameter of the local link. The default value for this variable is 3. Its range is 1 to 11, and its unit is one slot-time.

NOTE This unit is chosen so that all measurements of an inactive link are for multiples of one slot-time.

4.7.1.4 V(IRRD) immediate-response-recovery-delay

V(IRRD) is used by the DLE to record the link's immediate-response-recovery-delay (see 3.3.16), which is a computed parameter of the local link. The default value for this variable is V(MRD) + 1. Its range is 2 to 12, and its unit is one slot-time.

4.7.1.5 V(MRC) maximum-retry-count

V(MRC) is used by the DLE to record the link's maximum-retry-count, which limits the number of immediate retries of a transaction which are permitted. V(MRC) is a configuration parameter of the local link. Its default value is 0, meaning immediate retries are not permitted. Its range is 0 to 7.

4.7.1.6 V(NRC) network-repeat-count

V(NRC) is used by the DLE to record the extended link's network-repeat-count, which specifies the maximum number of deferred retries of a multi-link communication which should be attempted. V(NRC) is a configuration parameter of the extended link, based to some extent on the values of V(MRC) for the various local links of the extended link. Its default value is 1, meaning a single deferred retry should be attempted when other considerations do not intervene. Its range is 0 to 7.

4.7.1.7 V(NDL) network-DLPDU-lifetime

V(NDL) is used by the DLE to record the extended link's network-DLPDU-lifetime, which specifies the maximum period that any DLPDU can remain in transit within the extended link. Its range is 1 ms to 60 s, in units of 1 ms, and its default value is 30 s.

4.7.1.8 V(TN) this-node

V(TN) is used by the DLE to record the local node-id (see 3.3.25), which is a configuration parameter of the local DLE. Its default uninitialized value is zero, which does not permit the DLE to transmit. Its range is 0, and 10_{16} to FF_{16} (see 4.3.2.2).

4.7.1.9 V(TL) this-link

V(TL) is used by the DLE to record the local link-id (see 3.3.19), which is a configuration parameter of the local link. Its default uninitialized value is zero, which does not permit a bridge to forward to another link. Its range is 0, and 1000_{16} to $FEFF_{16}$ (see 4.3.2.1).

4.7.1.10 V(MEP) DL-MAC-address-embedding-prefix

V(MEP) is used by the DLE (see 4.3.4) to embed the extended-link's 32-bit address space within the 48-bit address space used by the ISO/IEC LAN (local area network) standards (ISO/IEC TR 8802-1). The default value for this variable is zero. Its range is 0 to $3FFF_{16}$.

4.7.1.11 C(RD) remaining-duration counter

C(RD) is used by the DLE to record the remaining duration of token delegation when using a token delegated by a PASS TOKEN (PT) or EXECUTE SEQUENCE (ES) DLPDU, or the remaining link capacity until the next scheduled activity when using a scheduler token during link maintenance activity. C(RD) is initialized on receipt of such a DLPDU from the DLPDU's DD-parameter and decrements at the rate of at least one count per octet of elapsed transmission capacity until it reaches the value zero or until the token is returned.

NOTE The phrase "at least" permits implementations to approximate C(RD) provided that they are either accurate or pessimistic in estimating the actual time required for decision-making and for preparing and sending a transmission. This permission may reduce the number of counters required in an actual implementation.

4.7.1.12 V(MID) minimum-inter-DLPDU-delay

V(MID) is used to specify the minimum duration for an interval of non-transmission which a transmitting DLE shall provide after receiving, or transmitting, a DLPDU. This interval is measured in units of the transmission duration of one octet; the measurement interval begins with either

a) the receipt of a PH-DATA confirm primitive by the DLE which confirms a PH-DATA request primitive which specified END-OF-DATA-AND-ACTIVITY, OR
b) the receipt of a PH-DATA indication primitive by the DLE specifying END-OF-DATA, END-OF-DATA-AND-ACTIVITY, or END-OF-ACTIVITY;

whichever occurs first, and ends with the submission of a PH-DATA request primitive by the DLE specifying start-of-activity. The default value for this variable is 0, and its range is 0 to the smaller of 120 or (V(MRD)-1) × V(ST).

NOTE The PhL may specify a minimum inter-transmission delay of its own. Such a constraint is independent of, and may apply concurrently with, the constraint specified above.

4.7.1.13 T(IRRD) immediate-response-recovery-delay monitor

T(IRRD) is used by the DLE which initiates a two-DLPDU transaction to monitor the local link for an anticipated immediate reply DLPDU, and to reassert its own token as the active token on the link if no reply is detected.

4.7.1.14 V(RA) reply-address

V(RA) is used by a DLE to record the DL-address to which a reply token was last passed by a COMPEL ACKNOWLEDGMENT (CA), COMPEL DATA (CD), or EXCHANGE DATA (ED) DLPDU. It also can be used by the LAS DLE to record the DL-address to which a reply token was last passed by a PROBE NODE-ADDRESS (PN) DLPDU.

4.7.1.15 V(OTA) outstanding-transaction-array

V(OTA) is used by the DLE, in transactions requiring an immediate reply, to correlate an immediate or delayed reply DLPDU with the initiating DLS-user request. It is also used by the DLE to determine, for any given transaction index, whether that index is currently assigned by the DLE to an incomplete outstanding transaction. V(OTA) is an array with an index range of $0..F_{16}$.

4.7.1.16 V(LTI) last-transaction-index

V(LTI) is used by the DLE, in transactions requiring an immediate reply, to ensure that a new transaction does not reuse the index of the immediately previous transaction initiated by the same DLE. Its range is $0..\ F_{16}$, and its initial value is F_{16}.

4.7.1.17 Q(US) unscheduled-service queue

Q(US) is used by the DLE to manage

a) references to the DLE's DL-address user-request queues (Q_A(UR) for all *A* within the DLE (see 4.7.3.1), including
   1) references which signal the need to compel a DLSDU transmission from the correspondent peer or publisher DLCEP, when *A* is a peer or subscriber DLCEP-address;
   2) references which signal the need to compel an instance of the unitdata-exchange service with the specified DLSAP-address, when *A* is a DLSAP-address whose DL(SAP)-role is INITIATOR;
   3) references which signal the need to transmit the contents of a sending buffer bound to a local DLCEP, when *A* is a local DLCEP-address;
   4) references which signal the need to send a DLSDU from the sending queue bound to local DLCEP or DLSAP-address, when *A* is a local DLCEP-address or DLSAP-address;
b) references to locally-scheduled active sequences (see Clause 10) resulting from DL-SCHEDULE-SEQUENCE requests (see 8.4.3.1);
c) DT DLPDUs containing DLS-user data which are delayed responses to received CD and ED DLPDUs, queued in support of the DL-UNITDATA-EXCHANGE service.

This queue is used upon receipt of a PASS TOKEN (PT) DLPDU addressed to the DLE (see 3.3.24 and 5.2.2.3). The structure of this queue is described in 3.3.5.

4.7.1.18 V(RID) random identifier

V(RID) is an identifier, chosen from an approximately uniform random distribution, used by the DLE in responding to a Probe NODE-address (PN) DLPDU addressed to the DLE, to introduce a random element into the response, which in turn is used to enable the DLE to validate a subsequently received node-activation SPDU addressed to the DLE.

Each time it is used in a Probe Reply (PR) DLPDU, before its value is copied into the PR DLPDU, the variable shall be set to a new random value chosen uniformly from the range 0 to FF_{16}. The actual random choice shall be statistically independent of similar choices made by other DLEs.

NOTE This requirement for statistical independence minimizes the probability of repeatedly identical choices by identically-constructed real devices, and thus provides the basis for eventual discrimination among multiple DLEs which happen to respond to the same NODE DL-address, either through fault or misconfiguration.

4.7.1.19 C(NT) node-time counter

C(NT) is used by the DLE to record the monotonically increasing component of the DLE's local node-time (see 3.3.35). In the absence of any corrective frequency adjustment (see 8.4.1.3), its nominal long-term rate of counting shall result in incrementing 8 192 000 units/s (= $2^{13} \times 10^3$ units/s). Its initial value shall be zero. C(NT) shall be sized not to roll over before reinitialization.

NOTE When other data is lacking, implementors may assume that the maximum period between reinitializations in industrial environments is five years. The specified rate of counting does not imply the granularity of the counting, which depends upon the implementation and the time synchronization capability of the DLE.

4.7.1.20 V(LSTO) local-link-scheduling-time-offset

V(LSTO) is used by the DLE to record the signed offset (difference) between the sum N(NT) + N(LSTO) of the local LAS's local-link-scheduling-time, as received in a TIME DISTRIBUTION (TD) DLPDU, and the receiving DLE's C(NT), so that V(LSTO) = N_{LAS}(LSTO) + N_{LAS}(NT) – C(NT). The initial value of V(LSTO) shall be zero.

4.7.1.21 V(DLTO) DL-time-offset

V(DLTO) is used by the DLE to record the signed offset (difference) between the DL-time and the local-link-scheduling-time, such that DL-time = C(NT) + V(LSTO) + V(DLTO). This variable is also received in a TIME DISTRIBUTION (TD) DLPDU. Its initial value shall be zero.

4.7.1.22 V(TQ) time-quality

V(TQ) is used by the DLE to record the multi-partite quality of both the DLE and the source and path of time distribution on the extended link. Its initial value shall indicate a local time source.

4.7.1.23 V(MD) measured-delay

V(MD) is used by the DLE to record the filtered measured delay in two-way communications between the DLE and the current LAS, as measured with a series of ROUND-TRIP-DELAY QUERY / ROUND-TRIP-DELAY REPLY DLPDUs. Its initial value, and its value when the DLE's own LAS functions are active, shall be zero. Its value is invalidated under the conditions specified in 8.4.1.3.

4.7.1.24 V(LN) LAS-node

V(LN) is used by the DLE to record the node-id (see 3.3.19) of the local link's LAS, as received in a TD (TIME DISTRIBUTION) DLPDU (see 8.4.1.3), or zero if the local link's LAS' node-id is unknown. Its initial value shall be zero. V(LN) is used to determine the validity and relevancy of the computed value of V(MD). Its range is the same as that of V(TN) (see 4.7.1.8).

4.7.1.25 V(TSC) time-synchronization-class

V(TSC) is used by the DLE to record the link's time-synchronization-class and to determine the DLE's own requirement for the minimum time distribution period which the DLE requires to maintain the specified level of time synchronization. V(TSC) shall specify one of the time-synchronization-classes defined in 10.6.3. Its default value shall be the 10 ms time-synchronization class (see 10.6.3).

4.7.1.26 T(TDP) time-distribution-period monitor

T(TDP) is used by the DLE to measure the time elapsed since the last sent or received TIME DISTRIBUTION (TD) DLPDU.

4.7.1.27 V(TSL) time-source-link

V(TSL) is used by the DLE to record the link-id of the source of time distribution on the extended link. Its range is the same as that of V(TL). Its initial value shall be zero.

4.7.2 Parameters and timers to support a DLS-user's request

Each specific instance of a DLS-user service request or response may have associated parameters and timers, depending on the type of request or response.

4.7.2.1 P_U(SDUL) DLSDU-length request parameter

P_U(SDUL) designates the length of the DLSDU associated with the specified DLS-user request or response.

4.7.2.2 P_U(SDU) DLSDU request parameter

P_U(SDU) designates the DLSDU associated with the specified DLS-user request or response. P_U(SDU) is conceptualized as an array of P_U(SDUL) octets, indexed from 1 to P_U(SDUL).

4.7.2.3 P_U(MCD) maximum-confirm-delay parameter

P_U(MCD) designates a DLS-user-established maximum confirm delay for a specific request, where the class of delay is always determined by associated text within this standard.

4.7.2.4 T_U(MCD) maximum-confirm-delay monitor

T_U(MCD) is used by the DLE to monitor the completion of the actions associated with the specified DLS-user request, to ensure that a corresponding DLS confirm is given to the DLS-user within the interval of time allotted by the DLS-user, as specified in the DLS-user-established maximum confirm delay for the specific request.

4.7.3 Queues to support DL-address-based DL-scheduling

Each specific instance of a DLSAP-address or DLCEP-address or subscriber DLCEP within the DLE shall have an associated user-request queue (see 3.3.3 and 4.7.3.1).

4.7.3.1 Q_A(UR) user-request queue

Each specific instance of Q_A(UR) is used by the DLE to manage DLS-user requests and responses which require DLPDU transmissions originating at the associated DL-address or subscriber DLCEP. The structure of this queue is described in 3.3.3.

4.7.4 Variables and timers to support a DLCEP

The state of each DLCEP is maintained in variables and timers specific to that DLCEP. The specific set of variables and timers required for a given DLCEP is dependent on the DLCEP's class and data delivery features.

As shown in Figure 7, the following relationships exist among the variables L (see 4.7.4.10) and H (see 4.7.4.11) and the negotiated receiving window size W_R at a receiving DLCEP, and the variables N (see 4.7.4.3), R (see 4.7.4.4), A (see 4.7.4.5), M (see 4.7.4.6), and the negotiated sending window size W_S (see 4.7.4.2) at the corresponding sending DLCEP:

If "SC"-subscripts designate a sending DLE of a DLC, and "RC"-subscripts designate a receiving DLE of the same DLC, then the relationships among the actual DLCEP variables, $V_C(\ldots)$, and negotiated parameters, $P_C(\ldots)$, from the point of view of a single direction of data transfer, are

$$P_{RC}(W_{RC}) = P_{SC}(W_{SC}) = W \tag{4}$$

$$V_{RC}(H)–W \le V_{SC}(M)–W \le V_{SC}(A) \le V_{RC}(L) \le V_{RC}(H) \le V_{SC}(M) \le V_{SC}(A)+W \le V_{RC}(L)+W \tag{5}$$

$$V_{SC}(M)–W \le V_{SC}(R) \le V_{SC}(M) < V_{SC}(N) \tag{6}$$

$$V_{RC,K}(RRS) \le V_{RC,K}(MRS) \tag{7}$$

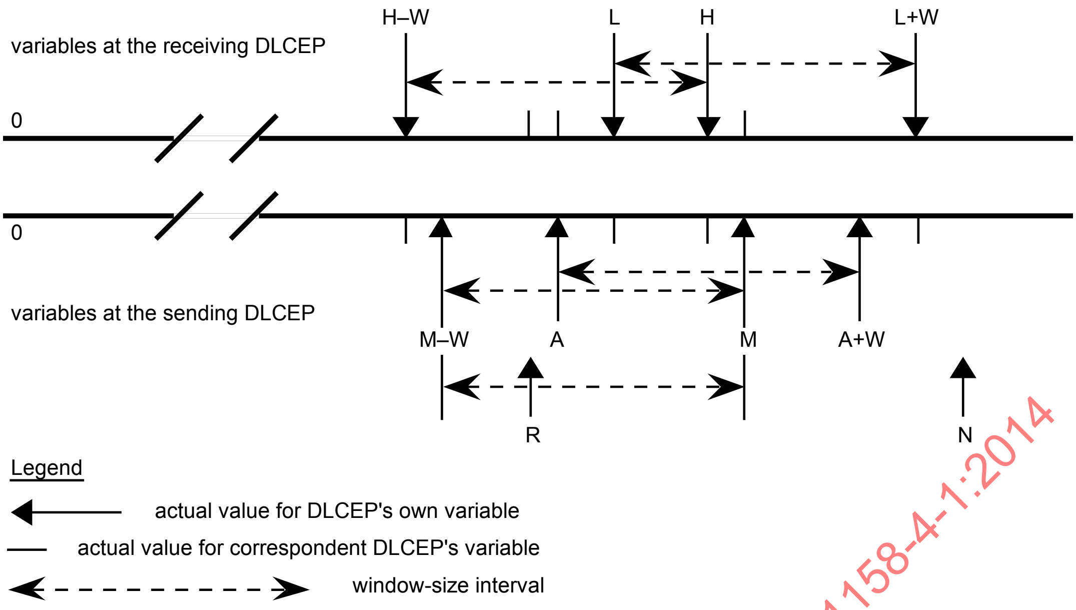


Figure 7 – Linear relationships of sending and receiving DLCEP sequence-number variables

4.7.4.1 V_C(ST) DLCEP state

V_C(ST) is used by a DLE to maintain the current user state of the DLCEP. These states are the same as those shown in Figure 11.

$$W_R = W_S = W \quad (8)$$

$$H–W \leq M–W \leq A \leq L \leq H \leq M \leq A+W \leq L+W \quad (9)$$

$$M–W \leq R \leq M < N \quad (10)$$

4.7.4.2 V_C(NP) negotiated DLCEP parameters

V_C(NP) is used by a DLE to maintain the current DLCEP parameters as a single structured variable, both during and after the DLCEP-establishment process. These parameters are defined by the EC-parameters (see 7.1) of the ESTABLISH CONNECTION (EC) DLPDU.

Within this standard, references to specific parameters within the EC parameters are denoted by the syntax "V.F", where V is a V_C(NP) variable and F is a field of the EC parameters as labeled in 7.1.

4.7.4.3 V_C(N) next sequence number to assign to a DLSDU

V_C(N) is used by a DLE to maintain the ordinal number one greater than the number of DLSDUs whose transmission on the DLCEP has been requested by the DLS-user. Its value at DLCEP establishment is one, and it increases monotonically each time the transmission of a new DLSDU is requested by the DLS-user. It does not increase when a DLS-user's request for transmission from a buffer (which must be on a DLCEP whose sending data delivery features are UNORDERED or ORDERED) results in the retransmission of a previously transmitted DLSDU.

NOTE This last requirement may be realized by a two-phase incrementation process, in which writing to the buffer arms the incrementation process for all DLCEPs which send from the buffer, and incrementation for a given DLCEP occurs during DLPDU formation immediately prior to transmission (but not retransmission) of the first segment of the buffer on that DLCEP.

4.7.4.4 $V_C(R)$ maximum non-transmittable DLSDU sequence number

$V_C(R)$ is used by a DLE for a DISORDERED or CLASSICAL PUBLISHER DLCEP, and optionally for an ORDERED PEER or PUBLISHER DLCEP, to maintain the ordinal number of the highest-numbered DLSDU which is no longer available for DLE transmission or retransmission, or zero when there is no such DLSDU. Its value at DLCEP establishment is zero, and it is non-decreasing. It increments each time a DLSDU, for which transmission was requested, becomes unavailable to the DLE for retransmission, either because the DLSDU has been queued within the DLL for longer than its specified maximum confirm delay, or because the DLSDU's sequence number differs from the highest sequence number transmitted by the sending window size $V_C(W_S)$, or due to DLC reset.

NOTE Incrementing $V_C(R)$ may cause a corresponding number of requests to move from the third to the second partition of the relevant $Q_A(UR)$.

4.7.4.5 $V_C(A)$ maximum acknowledged DLSDU sequence number

$V_C(A)$ is used by a DLE for a DISORDERED or CLASSICAL PEER DLCEP to maintain the ordinal number of the highest-numbered DLSDU transmitted from this DLCEP which has been acknowledged by the correspondent DLE. Its value at DLCEP establishment is zero, and it is non-decreasing. A newly-received CA, CD, DT, ED or RC DLPDU may cause it to be increased.

NOTE Incrementing $V_C(A)$ may cause a corresponding number of requests to move from the third to the second partition of the relevant $Q_A(UR)$.

4.7.4.6 $V_C(M)$ maximum transmitted DLSDU sequence number

$V_C(M)$ is used by a DLE for a sending DLCEP to maintain the ordinal number of the highest-numbered DLSDU which has at least one segment which is being, or has been, transmitted from this DLCEP. Its value at DLCEP establishment is zero, and it increases monotonically during the initial transmission of the first segment of each DLSDU.

4.7.4.7 $V_C(MS)$ maximum transmitted DLSDU segment number

$V_C(MS)$ is used by a DLE for an ORDERED, DISORDERED or CLASSICAL sending DLCEP to maintain the zero-based ordinal number of the highest-numbered segment of the $V_C(M)$'th DLSDU which is being, or has been, transmitted from this DLCEP. Its initial value at DLCEP establishment, before any DLSDU has been transmitted, is zero.

4.7.4.8 $V_{C,K}$ (SS) segments to send

$V_{C,K}$ (SS) is used by a DLE for a sending DLCEP to maintain the set of segments of the *K*'th DLSDU which still need to be transmitted. For each new (*N*'th) DLSDU presented by the DLS-user for transmission, a new list $V_{C,N}$ (SS) is created.

For A<*K*≤M and R<*K*<N, additions are made to the list upon receipt of a retransmission request for a segment of the *K*'th DLSDU, or upon timeout of the corresponding timer $T_{C,K}(SS)$ (or upon timeout of the timer $T_C(SS)$ when it applies to the *K*th DLSDU, when the permission of 4.7.4.9.1 is employed).

NOTE The lowest-numbered segment of the lowest-numbered DLSDU remaining is usually the next to be transmitted; the DLE deletes this segment from the list upon transmission. For ORDERED or PUBLISHER sending DLCEPs, the first untransmitted segment of the first not-completely-transmitted DLSDU may be sent instead, to maintain an assured rate of delivery of new DLSDUs on periodic DLCs.

4.7.4.9 $T_{C,K}(SS)$ sent-segments monitor

$T_{C,K}(SS)$ is used by a DLE for a DISORDERED or CLASSICAL PEER DLCEP to monitor timely peer acknowledgment of the *K*'th transmitted DLSDU, where A<*K*≤M. It is started when the DLSDU has been transmitted in its entirety, and is cancelled when the DLSDU is acknowledged. Upon its timeout, the last segment of the *K*'th DLSDU is again added to $V_{C,K}(SS)$, causing its retransmission and potentially triggering the receiving DLE to request retransmission of any lower-numbered segments which also have been missed.

4.7.4.9.1 $T_C(SS)$ simplified sent-segments monitor

It is permitted to replace the set of per-DLSDU timers { $T_{C,K}(SS)$ } with a single per-DLCEP-timer, $T_C(SS)$. Such a replacement will lead to equivalent DL-service when the number of outstanding DLSDUs is limited by other constraints, such as a small negotiated the window size, or when the residual error rate of the communications path is very low. When such constraints do not apply, then this replacement may lead to inferior DL-service.

If such a replacement is made, then the following definition applies:

$T_C(SS)$ is used by a DLE for a DISORDERED or CLASSICAL PEER DLCEP to monitor timely peer acknowledgment of transmitted DLSDUs. It is started, but not restarted, when a DLSDU has been transmitted in its entirety, and is cancelled when a DLSDU is acknowledged. It is then restarted if there are still unacknowledged DLSDUs.

Upon its timeout, the last segment of the unacknowledged $DLSDU_K$ with the lowest sequence number is again added to the $V_{C,K}(SS)$ corresponding to that DLSDU, causing its retransmission and potentially triggering the receiving DLE to request retransmission of any lower-numbered segments which also have been missed.

4.7.4.10 $V_C(L)$ last-reported DLSDU sequence number

$V_C(L)$ is used by a DLE for an ORDERED, DISORDERED or CLASSICAL receiving DLCEP to maintain the ordinal number of the last (highest-numbered) DLSDU received at the DLCEP and reported to the remote DLCEP. Its value at DLCEP establishment is zero, and it increases monotonically with the reporting of each completely- received DLSDU to the DLS-user.

4.7.4.11 $V_C(H)$ highest-detected DLSDU sequence number

$V_C(H)$ is used by a DLE for a PEER or SUBSCRIBER DLCEP to maintain the ordinal number of the highest-numbered DLSDU received, or detected as missing, at the DLCEP. Its value at DLCEP establishment is zero, and it increases with the reception of the first-received segment of each sequentially-higher-numbered newly-received DLSDU, or with the detection of a sequentially-higher-numbered missing DLSDU.

4.7.4.12 $V_C(HS)$ highest-detected segment number of the highest-detected DLSDU sequence number

$V_C(HS)$ is used by a DLE for a PEER or SUBSCRIBER DLCEP to maintain the zero-based ordinal number of the highest-numbered segment of the $V_C(H)$'th DLSDU which has been received, or detected as missing, at the DLCEP. Its value at DLCEP establishment is zero:

a) It is set to the zero-origin index of each newly-received segment, $N_R(ASN)$ (see 7.4.2.1C)6)), received in a DLPDU provided that
   1) the received DLSDU number is greater than the prior value of $V_C(H)$, or
   2) the received DLSDU number is the same as the prior value of $V_C(H)$, and the received segment number is greater than the prior value of $V_C(HS)$.
b) It is set to zero when $V_C(H)$ is advanced upon detection of a missing DLSDU, and the number of segments in that DLSDU is unknown.

4.7.4.13 $V_{C,K}(MRS)$ missing received segments

$V_{C,K}(MRS)$ is used by a DLE for an ORDERED, DISORDERED or CLASSICAL PEER or SUBSCRIBER DLCEP to maintain the set of missing segments for the *K*'th DLSDU, for L<*K*≤H. Additions are made to this list, or one or more new lists created, upon receipt of any segment of a new DLSDU. Deletions are made as segments included on the list are received.

NOTE 1 When at least one segment of the *K*'th DLSDU has been received, then the total number of segments in the DLSDU is known and the missing segments list is exact. Otherwise, it may be assumed that the missing DLSDU consists of only a single segment whose retransmission needs to be requested; upon receipt of any segment of the DLSDU, the list of segments still missing can be made exact.

NOTE 2 Subscribers of a common publisher DLCEP are permitted, but are not required, to eavesdrop on each other's transmissions to the publisher. Such eavesdropping may provide early detection of message loss. Thus one or more new lists may also be created upon receipt of a request for retransmission of a segment of the *K*'th DLSDU, where *K*>H.

4.7.4.14 $V_{C,K}$(RRS) retransmission-request required segments

$V_{C,K}$(RRS) is used by a DLE for an ORDERED, DISORDERED or CLASSICAL, PEER or SUBSCRIBER DLCEP to maintain the set of missing segments for the *K*'th DLSDU for which retransmission requests are required, for L<*K*≤H. Additions are made to this list when a received DLPDU implies that one or more DLSDU segments or complete DLSDUs have been missed, and (for SUBSCRIBER DLCEPs) upon timeout of the timer $T_{C,K}$(RRS) used by the DLE to monitor for a successful response to a previous retransmission request for one or more segments of the same DLSDU.

Deletions are made upon reception (including reception during transmission) of a DLPDU which either contains, or requests the retransmission of, a missing segment.

NOTE Such deletions upon reception of a request for retransmission for another subscriber on the same multi-peer DLC are optional (see Note 2 in 4.7.4.13).

4.7.4.15 $T_{C,K}$(RRS) retransmission request monitor

$T_{C,K}$(RRS) is used by a DLE for a DISORDERED or CLASSICAL SUBSCRIBER DLCEP, and optionally for an ORDERED PEER or SUBSCRIBER DLCEP, to monitor for a successful response to a previous retransmission request for one or more segments of the *K*'th DLSDU, where L<*K*≤H. The timer is started when not running, but not restarted while running, when a retransmission request is sent for one of the segments of the incomplete DLSDU. It is cancelled when all segments of the DLSDU have been received. Upon its timeout, $V_{C,K}$(RRS) is set equal to the then-current value of $V_{C,K}$(MRS).

NOTE The timer may also be started, but not restarted, when a retransmission request from another subscriber of the same multi-peer DLC, for one of the segments of the incomplete DLSDU, is overheard (see Note 2 in 4.7.4.13).

4.7.4.16 T_C(RAS) residual activity stimulus

T_C(RAS) shall be used by a DLE at an ORDERED, DISORDERED or CLASSICAL, PUBLISHER or sending PEER DLCEP, when DLC establishment (see 8.2.1.1) requested residual activity on the DLC in this (sender-to-receiver) direction of data transfer, to ensure activity from that DLCEP during periods when all DLSDUs presented for transmission at that DLCEP since DLC establishment, or since the most recent DLC reset, have been transmitted

a) at a sending DISORDERED or CLASSICAL PEER DLCEP, when all of those DLSDUs have been acknowledged (that is, V_C(A) = V_C(M));
b) at a PUBLISHER DLCEP, or sending ORDERED PEER DLCEP, when none of those DLSDUs need to be (wholly or partially) retransmitted.

This timer restarts at every transmission from the DLCEP. At timeout, it causes the current state of the DLCEP to be sent to the correspondent DLCEP(s).

4.7.4.17 T_C(RAM) residual activity monitor

T_C(RAM) is used by a DLE for an ORDERED, DISORDERED or CLASSICAL, SUBSCRIBER or receiving PEER DLCEP to detect inactivity on the DLC. It is only required when DLC establishment (see 8.2.1.1) requested residual activity on the DLC in this (receiver-from-sender) direction of data transfer. It runs continuously, and restarts upon reception of any DLPDU on the DLCEP. At timeout, it causes a DLCEP (and possibly DLC) reset, which may in turn lead to a DLCEP (and possibly DLC) disconnect.

4.7.4.18 V_C(TNA) DL-time of last network access

V_C(TNA) is used by the DLE to record the DL-time at which a DL-BUFFER-RECEIVED indication or DL-BUFFER-SENT indication was last generated at the associated DLCEP. It is used only when the DLCEP has been specified as a synchronizing DLCEP during the establishment of one or more other local DLCEPs.

4.7.4.19 V_B(TW) DL-time of last buffer write

V_B(TW) is used by the DLE to record the DL-time at which all of a DLSDU was last written to the associated buffer (by a DL-PUT request primitive, or by the completion of reception into the buffer at a DLCEP). It is used only when the DLCEP establishment specified a DL-timeliness class other than NONE.

4.7.4.20 V_B(TP) DL-time of production

V_B(TP) is used by the DLE to record the DL-time at which a DLS-user transferred the associated DLSDU to the DLE. This is the DL-time at which the sending DLS-user issued the DL-PUT request which placed the associated DLSDU in the sending DL-buffer. It is used only when the DLCEP establishment specified a DL-timeliness class other than NONE.

4.7.4.21 V_B(TS) Timeliness-status of buffer-writing

V_B(TS) is used by the DLE to record the timeliness-status associated with the DLSDU stored in a buffer. If the buffer is written as a result of a DL-PUT request, then the value of this variable shall be equal to the DLS-user data-timeliness. If the buffer is written as a result of reception of a DLPDU, then the value of this variable shall be updated as part of the reception procedure.

4.7.5 Variables and timers to support the link-master class

The variables defined in 4.7.5 are required in all link master and bridge DLEs.

4.7.5.1 V(DTA) delegation-address

V(DTA) is used by the LAS DLE to record the DL-address to which the token was last passed by a PASS TOKEN (PT), EXECUTE SEQUENCE (ES), or TRANSFER LAS (TL) DLPDU.

4.7.5.2 V(LL) local-link live-list

V(LL) is used by the LAS DLE to record the set of NODE DL-addresses (see 3.3.24 and 5.2.2.3) on the local link which appear to be in use by peer DLEs. It is used and updated by the LAS DLE during the LAS's address-probing and first-opportunity activities, as described in 10.2.

NOTE Some previously-existing national standards refer to such a set as a "live list", because it indicates the current set of interacting DLEs on the local link.

4.7.5.3 V(TCL) token-circulation list

V(TCL) is used by the LAS DLE to record the set of NODE DL-addresses (see 3.3.24 and 5.2.2.3) on the local link to which a PT DLPDU should be circulated to provide an opportunity for unscheduled transmissions, including requests to the LAS DLE for scheduling communications. This token circulation emulates the distributed token passing of prior national and international standards.

NOTE This variable is used to restrict the circulation of an unrestricted token to only the “master” (that is, self-sufficient) nodes of the local link. This functionality assists in migration from some previously-existing national standards.

V(TCL) is used and updated by the LAS DLE during the LAS’s address-probing and first-opportunity activities, as described in 10.2. The DLEs represented in V(TCL) shall always be a subset of the DLEs represented in V(LL).

4.7.5.4 V(ENRL) expected-non-response list

V(ENRL) is used by the LAS DLE to record the set of NODE DL-addresses (see 3.3.24 and 5.2.2.3) on the local link of fractional-duty-cycle (FDC) DLEs which are expected to be non-responsive (that is, “asleep”). This is a subset of the set of FDC DLEs on the link, and changes dynamically as FDC DLEs inform the LAS DLE of either

a) their attentiveness (“wakefulness”), or
b) their intent to become non-responsive (to “go to sleep”).

Failure of a DLE listed in V(ENRL) to respond to a PT DLPDU addressed to the DLE, or an ES DLPDU addressed to a hierarchically-structured (see 5.2.2.1 format 1 or 5.2.2.2 format 1) DLSEP of the DLE,

1) shall not be treated as an error;
2) the transmission of the PT or ES DLPDU shall not be retried.

NOTE This last requirement is equivalent to using a value of zero (0) for V(MRC) when sending a PT DLPDU (see 6.15.3) to such a DLE.

4.7.5.5 V(MST) maximum-scheduled-traffic

V(MST) is used by the LAS DLE, during any dynamic schedule construction, to determine the maximum fraction of the theoretical interval V(TTRT) (see 4.7.5.11) which may be dedicated to explicitly scheduled DL-activity.

V(MST) is a configuration parameter of the local link which takes the form of a one-octet binary fraction. The range for this parameter is $0{,}00_{16}$ to $0{,}BF_{16}$; its default value is $0{,}40_{16}$.

4.7.5.6 V(MSO) maximum-scheduling-overhead

V(MSO) specifies the maximum scheduling overhead permitted an LAS DLE by the existing link schedule. Its range is 0 to $3F_{16}$; its default value is $3F_{16}$, and its unit is the transmission duration of one octet. This overhead is included in the time allocated for each scheduled activity, and so is used only during schedule construction and determination of whether a DLE can serve as LAS for an existing schedule.

4.7.5.7 V(DMDT) default-minimum-token-delegation-time

V(DMDT) is used by the LAS DLE to determine the default minimum amount of local link capacity which the LAS shall allocate to a DLE in a single PT DLPDU sent to the DLE.

V(DMDT) is a configuration parameter of the local link. The range for this variable is 20_{16} to $7FFF_{16}$, its default value is 54_{16} + V(PhLO), which permits the sending of one URGENT DLPDU, and its unit is the transmission duration of one octet.

4.7.5.8 V(DTHT) default-token-holding-time

V(DTHT) is used by the LAS DLE to specify the default initial amount of local link capacity which should be allocated to each DLE, in one cycle of “circulating the token”, by the LAS when the LAS sends one or more PT DLPDUs to the DLE.

V(DTHT) is a configuration parameter of the local link. The range for this variable is 114_{16} to 65 000, its default value is 114_{16} + V(PhLO), which permits the sending of one TIME-AVAILABLE DLPDU, and its unit is the transmission duration of one octet.

4.7.5.9 V(LTHT) link-maintenance-token-holding-time

V(LTHT) is used by the LAS DLE to specify the initial amount of local link capacity which should be allocated to LAS-related link maintenance activities in one cycle of “circulating the token”. These activities include

– probing a single unused NODE DL-address via a PN DLPDU for the appearance of a DLE not currently active on the link, and the consequent sending of a DT DLPDU containing a node-activation SPDU to any such DLE;
– measuring the round-trip communications delay with another DLE on the local link via an RQ DLPDU;
– conveying updated schedule information to other link master DLEs on the local link to provide for schedule continuation after transfer of the LAS role to such a DLE;
– coordinating a multi-link schedule with other LAS DLEs of the extended link.

V(LTHT) is a configuration parameter of the local link. The range for this variable is 124_{16} to 65 000, and its unit is the transmission duration of one octet. Its default value is

124_{16} + (3 × V(PhLO)) + (V(IRRD) × V(ST)), which permits

the probing of one NODE DL-address and the sending of one TIME-AVAILABLE DT DLPDU.

4.7.5.10 V(MTHA) maximum-token-holding-time-array

V(MTHA) is used by the LAS to specify, separately for each DLE listed in V(TCL) (see 4.7.5.3), the initial amount of local link capacity which should be allocated to that DLE, in one cycle of “circulating the token”, by the LAS when the LAS sends one or more PT DLPDUs to that DLE. The range and units of this parameter are the same as V(DTHT) (see 4.7.5.8). Its default value is also V(DTHT).

4.7.5.11 V(TTRT) target-token-rotation-time

V(TTRT) is used by the LAS DLE to specify the desired upper bound on the time required for one cycle of “circulating the token” to all of the DLEs on the local link. One cycle of “circulating the token” is measured as the interval between successive occurrences of the LAS DLE sending a PT DLPDU, with a token-use subfield specifying RESTART (see 6.15.2), to the lowest-numbered NODE DL-address represented in V(LL).

V(TTRT) is a configuration parameter of the local link. The range for this variable is 1 to 60 000, its default value is 60 000, and its unit 1 ms.

4.7.5.12 V(ATRT) actual-token-rotation-time

V(ATRT) is used by the LAS DLE to determine the actual time used for each cycle of “circulating the token” to all of the DLEs on the local link, as measured by the interval between successive occurrences of the LAS DLE’s sending the PT DLPDU, with a token-use subfield specifying RESTART (see 6.15.2), to the lowest-numbered NODE DL-address represented in V(LL).

4.7.5.13 V(RTHA) remaining-token-holding-time-array

V(RTHA) is used by the LAS to specify, separately for each DLE listed in V(TCL) (see 4.7.5.3), the remaining amount of local link capacity which should be allocated to that DLE, in the current cycle of “circulating the token”, by the LAS when the LAS sends one or more PT

DLPDUs to that DLE. The range and units of this array are the same as V(MTHA), and it is reinitialized from V(MTHA) at the beginning of each cycle of "circulating the token".

4.7.5.14 V(NTHN) next-token-holding-node

V(NTHN) is used to record the NODE DL-address of the next DLE to which "the token should be circulated", which is the next NODE DL-address to which a PT DLPDU will be sent. Its range is 10_{16} to FF_{16} (see 4.3.2.2), and it advances cyclically through the set of DLEs specified by V(TCL) (see 4.7.5.3).

4.7.5.15 V(FUN) first-unpolled-node

V(FUN) specifies the first NODE DL-address of a series of consecutive NODE DL-addresses which are to be omitted from the orderly probe of NODE DL-addresses for DLE's not specified by V(LL). Its range is 14_{16} to $F7_{16}$ (see 4.3.2.2).

4.7.5.16 V(NUN) number-of-consecutive-unpolled-nodes

V(NUN) specifies the number of consecutive NODE DL-addresses which are to be omitted from the orderly probe of NODE DL-addresses for DLE's not specified by V(LL). Its range is 00_{16} to $E4_{16}$, and its default value is 0.

4.7.5.17 P(TRD) token-recovery-delay

P(TRD) is used by the LAS DLE to record the DLE's token-recovery-delay (see 3.3.34), which is determined solely by the DLE itself. The default value for this variable is 14. Its range is V(MRD)+3 to 14, and its unit is one slot-time.

NOTE This unit is chosen so that all measurements of an inactive link are for multiples of one slot-time.

4.7.5.18 V(TDP) time-distribution-period

V(TDP) is used to determine the minimum frequency of time distribution on the local link. Its initial value shall be the minimum value required for the link's time-synchronization-class, V(TSC) as specified in 10.6.3. Its range is 5 ms to 55 s, and its unit is 1 ms.

4.7.5.19 V(MICD) maximum-inactivity-to-claim-LAS-delay

V(MICD) is used by the LAS DLE to record the link's maximum-inactivity-to-claim-LAS-delay (see 3.3.32), which is a configuration parameter of the local link. Its range is 1 to 4 095, and its unit is one-eighth of the transmission period of one octet.

4.7.5.20 V(LDDP) LAS-data-base-distribution-period

V(LDDP) is used by the LAS to determine the time between two successive distributions of the LAS' data-base by means of LAS-data-base-status SPDUs sent on the local link. The range of V(LDDP) is 100 ms to 55 s, and its unit is 1 ms. The default value of this variable is 5 s.

4.7.6 Variables and timers to support the bridge class.

The variables and timers defined in 4.7.6 and in 10.4 are required in all bridge DLEs.

4.7.6.1 V(ML) maximum-link

V(ML) may be used by a bridge DLE to limit the number of entries in the bridge's table of default forwarding information, which can have one entry for each link of the extended link. Its range is 1000_{16} to $FEFF_{16}$ (see 4.3.2.1), which is the maximum link-id defined for the extended network.

Additional variables and timers are defined in 10.4, mostly by reference to ISO/IEC 10038.

5 General structure and encoding of PhIDUs and DLPDUs, and related elements of procedure

Within Clause 5, any reference to bit *K* of an octet is a reference to the bit whose weight in a one-octet unsigned integer is 2^K.

NOTE This is sometimes referred to as "little endian" bit numbering.

5.1 PhIDU structure and encoding

Each PhIDU consists of Ph-interface-control-information (PhICI) and in some cases one octet of Ph-interface-data (see 4.4). When the DLE transmits a DLPDU, it computes a frame check sequence for the DLPDU as specified in 5.2.5.1, concatenates the DLPDU and frame check sequence, and transmits the concatenated pair as a sequence of PhIDUs as follows:

a) The DLE issues a single Ph-DATA request primitive with PhICI specifying START-OF-ACTIVITY, and awaits the consequent Ph-DATA confirm primitive.

b) The DLE issues a sequence of Ph-DATA request primitives with PhICI specifying DATA, each accompanied by one octet of the DLPDU as Ph-interface-data, from first to last octet of the DLPDU, and after each Ph-DATA request primitive awaits the consequent Ph-DATA confirm primitive.

c) The DLE issues a sequence of two Ph-DATA request primitives with PhICI specifying DATA, each accompanied by one octet of the FCS as Ph-interface-data, from first to last octet of the FCS, and after each Ph-DATA request primitive awaits the consequent Ph-DATA confirm primitive.

d) The DLE issues a single Ph-DATA request primitive with PhICI specifying END-OF-DATA-AND-ACTIVITY, and awaits the consequent Ph-DATA confirm primitive.

The DLE forms a received DLPDU by concatenating the sequence of octets received as Ph-interface-control-information of consecutive Ph-DATA indications, computing a frame check sequence for those received octets as specified in 5.2.5.2, and checks the syndrome (residual) of the computed FCS for correctness as follows:

e) The DLE receives a single Ph-DATA indication primitive with PhICI specifying START-OF-ACTIVITY, and initializes its computation of an FCS for the received DLPDU.

f) The DLE receives a sequence of Ph-DATA indication primitives with PhICI specifying DATA, each accompanied by one octet of the received DLPDU as Ph-interface-data, incrementally computes an FCS on the received octet, and concatenates all but the last two of those received octets to form the received DLPDU.

g) The DLE receives a single Ph-DATA indication primitive with PhICI specifying either END-OF-DATA, END-OF-DATA-AND-ACTIVITY or END-OF-ACTIVITY, and checks the syndrome of the computed FCS for correctness:

   1) If the PhICI specified END-OF-DATA or END-OF-DATA-AND-ACTIVITY, and the computed FCS syndrome was correct, then the DLE reports the reconstructed DLPDU and the two octets of received FCS as a correctly-received DLPDU suitable for further analysis.

   2) Otherwise the DLE increments its management statistics to reflect the erroneously-received DLPDU.

5.2 Common DLPDU structure, encoding and elements of procedure

Each DLPDU consists of a frame control field which specifies the type of DLPDU and conveys small size (fractional octet) parameters of the DLPDU; zero to three explicit address fields, each containing a DL-address, all of the same length; additional parameters of the DLPDU; and for most DLPDUs, a user data field conveying all or part of a DLSDU. To this is appended before transmission, and removed after reception, an FCS field (see 5.2.5) used to check the integrity of the received DLPDU.

5.2.1 Frame control (FC) field

The frame control (FC) field consists of one octet. It specifies the type of the DLPDU. For many types of DLPDU it also conveys a number of fractional-octet parameters, known as frame-control subfields, specific to the DLPDU type.

Some types of DLPDU require an immediate reply. Such DLPDUs may only be sent while holding a scheduler or delegated token.

When a token is delegated, then the priority specified in the token DLPDU is the minimum priority required of all DLPDUs sent during the following period of token use. DLPDUs of lower priority shall not be sent during that period.

5.2.1.1 Address size subfield

An address-size subfield is used to specify the number of octets in each address field of the DLPDU.

In DLPDUs which support multiple address sizes, this subfield occupies bit 3 of the frame control field. Its encoding is

0: **SHORT** – the DLPDU's address fields are each two octets;

1: **LONG** – the DLPDU's address fields are each four octets, and their link designator subfields are non-zero if possible.

The DT DLPDU also has a special form with only implicit DL-addresses:

– **VERY-SHORT** – the DLPDU logically contains a single DL-address field, which is null (zero octets in length).

  i) If the single null DL-address field is a source address, then the source DL-address of the current DLPDU is implicitly the destination DL-address of the immediately prior CA, CD or ED DLPDU on the link (which must have been a DLCEP-address).

  ii) If the single null DL-address field is a destination address, then the destination DL-address of the current DLPDU is implicitly the source DL-address of the immediately prior CA, CD or ED DLPDU on the link (which itself may be an implicit DLSAP- or DLCEP-address).

NOTE This VERY SHORT form is indicated by other components of the DLPDU's frame control octet, and not by its address-size subfield.

5.2.1.2 Final token use subfield

A final-token-use designator is used to optimize the return of a delegated token at the end of the last transaction of an instance of delegated token usage. This subfield is present in most DLPDUs, including all DLPDUs which can be sent by a DLE which holds a delegated token or reply token. When present, this subfield occupies bit 2 of the frame control field. Its encoding and semantics depends on the type of DLPDU as follows.

a) The encoding and semantics for CA, CD, CT, ED and RQ DLPDUs is

   0: **NOT-FINAL** – the delegated token is not returned to the LAS at the end of the current transaction;

   1: **FINAL** – the delegated token is returned to the LAS at the end of the current transaction, after transmission of the DLPDU and its requested immediate reply DLPDU, whether detected or not, and no additional use of the token being returned is needed at this time.

b) The encoding and semantics for DC, DT, EC, RC, RR, SR and TD DLPDUs is:

0: **NOT-FINAL** – the delegated token is not returned to the LAS at the end of the current transaction;

1: **FINAL** – the delegated token is returned to the LAS at the end of the current transaction, after transmission of the DLPDU, and no additional use of the token being returned is needed at this time.

c) The encoding and semantics for PT and ES DLPDUs is:

0: **RESTART** – this is the initial token delegation within the current cycle of “circulating the token” or of scheduled sequence execution, and so the indicated sequence and any repetitively scheduled transactions should be restarted;

1: **CONTINUE** – this is a subsequent (that is, secondary) token delegation within the current cycle of “circulating the token” or of scheduled sequence execution, and so the previously initiated sequence of queued or scheduled transactions should be continued.

When a bridge receives for forwarding a DLPDU which contains a final-token-use subfield whose value is different than required by the bridge, then the bridge shall complement the received DLPDU’s final-token-use subfield before forwarding and shall make a compensating modification to the received DLPDU’s FCS field (see 5.2.5.3) to preserve the FCS integrity protection provided by the DLPDU’s originator.

NOTE Only CA, CD, DC, DT, EC, ED and RC DLPDUs (see 6.1 to 6.7) are forwarded through bridges.

5.2.1.3 Priority subfield

A Priority designator is used to specify the DLPDU’s or transaction’s priority, to limit the size of the user data field of the DLPDU, and to constrain the minimum (lowest) priority of any DLPDU sent as an immediate reply requested by, or token usage delegated by, this DLPDU. This subfield is present in all CA, CD, DT, ED, PT and RC DLPDUs which contain explicit DL-addresses. When present, this subfield occupies bits 1 and 0 of the frame control field. Its encoding is

01: **URGENT** (high) priority;

10: **NORMAL** (medium) priority;

11: **TIME-AVAILABLE** (low) priority.

5.2.2 DL-address fields

The structure of DL-addresses is specified in 4.3, which also specifies standard pre-assigned addresses and address ranges.

5.2.2.1 LONG address field

A **LONG** address field usually consists of a fixed sequence of three parts as specified in 4.3. The parts are

a) an explicit link designator component;

b) an explicit node designator component;

c) an explicit selector component.

Separately, the three parts reflect an hierarchical address structure. This hierarchy may be partially or totally flattened. These alternatives are shown in Figure 8.

format 1

Link	Node	Selector
2 octets	1 octet	1 octet

or

format 2

Link	Node || Selector
2 octets	2 octets

or

format 3

Link || Node || Selector
4 octets

Figure 8 – DL-address alternative structures

When present, the Link field shall be delocalized on transmission and reception, as specified in 5.2.2.4. Similarly, when the Link field is present and specifies the local link, and when the Node field is present, then the Node field shall be delocalized on transmission.

5.2.2.2 SHORT address field

A **SHORT** address field usually consists of a fixed sequence of one implicit and two explicit parts, as specified in 4.3 and Figure 9. The parts are

a) an implicit link designator component, specifying the local link, which is always present;

b) an explicit node designator component;

c) an explicit selector component.

Separately, the parts reflect an hierarchical address structure. This hierarchy may be partially flattened. The alternatives are shown in Figure 9.

format 1	0000	**Node**	**Selector**
		1 octet	1 octet

or

format 2	0000	**Node || Selector**
		2 octets

Figure 9 – SHORT DL-address field – alternative implicit structures

When present, the Node field shall always be delocalized on transmission, as specified in 5.2.2.4.

5.2.2.3 NODE DL-address field

A **NODE** DL-address is a one-octet version of a SHORT address field which designates some DLE's DL-support functions. It consists of a fixed sequence of two implicit and one explicit parts, as specified in 4.3 and Figure 10. The parts are

a) an implicit link designator component, specifying the local link;

b) an explicit node designator component, specifying the DLE;

c) an implicit selector component of zero, specifying the DLE's DL-support functions.

format 1	0000	**Node**	00
		1 octet	

Figure 10 – NODE DL-address field – implicit structure

The Node field shall always be delocalized on transmission, as specified in 5.2.2.4.

5.2.2.4 Delocalization

The following transmission and reception processes are collectively referred to as delocalization (see 3.3.2), which is the DLE-internal process of converting synonymous DL-addresses to a canonical form for transmission, or for DL-address recognition during reception.

Two values of the Link field – the value zero and the value equal to V(TL) – are equivalent in designating the local link:

a) On transmission, when the value of an explicit Link field would otherwise be zero, then the value of that Link field shall always be set to V(TL).

NOTE 1 The variable V(TL) can have the value zero, in which case this substitution effects no change.

NOTE 2 This equivalence makes it possible for a DLE to transmit on the local link without knowing the link's correct value for V(TL), or during periods when that value is in transition (in which case it is being administratively changed from zero to non-zero).

NOTE 3 This equivalence provision is an aid to DLS-users, and potentially in implementations of this DL-protocol; it permits references to the local link DL-address component to be represented uniformly by the value zero.

b) On reception, with respect to the Link field, the value V(TL) shall be considered equivalent to the value zero, except that a DLPDU containing a DL-address with a Link field actually equal to zero shall not be forwarded by a bridge onto a different link.

Within a DLE, and when addressing the local link, two values of the Node field – the value zero and the value equal to V(TN) – are equivalent in designating the local DLE on the local link.

c) On transmission, when the value of the Link field is equivalent to V(TL), and the value of a Node field would otherwise be zero, then the value of that Node field shall always be set to V(TN).

NOTE This equivalence provision is an aid to DLS-users, and potentially in implementations of this DL-protocol; it permits references to the local node DL-address component to be represented uniformly by the value zero.

5.2.3 Parameter field

Each DLPDU class may have a DLPDU-class-specific parameter field; these are all described in Clause 7.

When it is desirable to distinguish between the values of a DLE variable, V(xx), or counter, C(xx), which is copied into a parameter field, and the current value of the same variable or counter, then the value within the parameter field is referred to as N(xx), because it no longer tracks changes in the value of the source variable or counter. The need for this separation of nomenclature is particularly apparent in the case of counters such as C(NT), which never stop counting, and in the various time-related DLPDUs (TD, RQ and RR), which may contain multiple fields based on various samplings of C(NT).

5.2.4 User data field

RC, CA, DT and ED DLPDUs, associated with the connectionless and connection-oriented data transfer services, contain a user data field which is used to convey either a partial or complete DLSDU from one DLS-user to another. The size of this user data field is constrained to be no larger than that permitted by the priority of the conveying DLPDU:

01: **URGENT** (high) priority: ≤ 64 octets;

10: **NORMAL** (medium) priority: ≤ 128 octets;

11: **TIME-AVAILABLE** (low) priority: ≤ 256 octets.

PR DLPDUs contain a user data field which is used to convey a probe-response SPDU (see 9.3.2.1) from the token-holding DLE to the current LAS DLE; its size is constrained to be no larger than permitted at URGENT priority – 64 octets.

TL DLPDUs contain a user data field which is used to convey a LAS-data-base-status SPDU (see 9.3.2.3) from the LAS DLE to the addressed DLE; its size is constrained to be no larger than permitted at URGENT priority – 64 octets.

DC and EC DLPDUs contain a user data field. Their implicit priority is TIME-AVAILABLE; however, the size of their data fields is constrained to be no larger than permitted at NORMAL priority – 128 octets.

5.2.5 Frame check sequence field

Within this subclause, any reference to bit K of an octet is a reference to the bit whose weight in a one-octet unsigned integer is 2^K.

NOTE 1 This is sometimes referred to as “little endian” bit numbering.

For the protocol Type of this standard, as in other International Standards (see Note 2), DLPDU-level error detection is provided by calculating and appending a multi-bit frame check sequence (FCS) to the other DLPDU fields during transmission to form a "systematic code word"[1] of length n consisting of k DLPDU message bits followed by $n - k$ (equal to 16)

[1] W. W. Peterson and E. J. Weldon, Jr., *Error Correcting Codes* (2nd edition), MIT Press, Cambridge, 1972.

redundant bits, and by calculating during reception that the message and concatenated FCS form a legal (n,k) code word. The mechanism for this checking is as follows:

NOTE 2 For example, ISO/IEC 8802 and ISO/IEC 9314-2.

The generic form of the generator polynomial for this FCS construction is specified in equation (6) and the polynomial for the receiver's expected residue is specified in equation (11). The specific polynomials for this standard are specified in Table 9. An exemplary implementation is discussed in Annex A.

Table 9 – FCS length, polynomial and expected residual

Item	Value
n-k	16
G(x)	$X^{16} + X^{12} + X^{11} + X^{10} + X^8 + X^7 + X^6 + X^3 + X^2 + X + 1$ (Notes 1, 2, 3)
R(x)	$X^{15} + X^{14} + X^{13} + X^9 + X^8 + X^7 + X^4 + X^2$ (Note 4)

NOTE 1 Code words D(X) constructed from this G(X) polynomial have Hamming distance 4 for lengths ≤ 344 octets and Hamming distance 5 for lengths ≤ 15 octets.

NOTE 2 This G(X) polynomial is relatively prime to all, and is thus not compromised by any, of the polynomials commonly used in DCEs (modems): the differential encoding polynomial $1 + X^{-1}$ and all primitive scrambling polynomials of the form $1 + X^{-j} + X^{-k}$.

NOTE 3 This G(X) polynomial is the optimal 16-bit polynomial for burst error detection over DLPDUs of 300 octets or less when the statistics of the error burst have a Poisson distribution (as is the usual case).

NOTE 4 The remainder R(x) should be 1110 0011 1001 0100 (X^{15} to X^0, respectively) in the absence of errors.

5.2.5.1 At the sending DLE

The original message (that is, the DLPDU without an FCS), the FCS, and the composite message code word (the concatenated DLPDU and FCS) shall be regarded as vectors M(X), F(X), and D(X), of dimension k, $n - k$, and n, respectively, in an extension field over GF(2). If the message bits are $m_1 \ldots m_k$ and the FCS bits are $f_{n-k-1} \ldots f_0$, where

$m_1 \ldots m_8$ form the first octet sent,

$m_{8N-7} \ldots m_{8N}$ form the Nth octet sent,

$f_7 \ldots f_0$ form the last octet sent, and

m_1 is sent by the first PhL symbol(s) of the message and f_0 is sent by the last PhL symbol(s) of the message (not counting PhL framing information),

NOTE This "as transmitted" ordering is critical to the error detection properties of the FCS.

then the message vector M(X) shall be regarded to be

$$M(X) = m_1X^{k-1} + m_2X^{k-2} + \ldots + m_{k-1}X^1 + m_k \quad (11)$$

and the FCS vector F(X) shall be regarded to be

$$F(X) = f_{n-k-1}X^{n-k-1} + \ldots + f_0 \quad (12)$$
$$= f_{15}X^{15} + \ldots + f_0$$

The composite vector D(X), for the complete DLPDU, shall be constructed as the concatenation of the message and FCS vectors

$$D(X) = M(X)\,X^{n-k} + F(X) \quad (13)$$
$$= m_1X^{n-1} + m_2X^{n-2} + \ldots + m_kX^{n-k} + f_{n-k-1}X^{n-k-1} + \ldots + f_0$$
$$= m_1X^{n-1} + m_2X^{n-2} + \ldots + m_kX^{16} + f_{15}X^{15} + \ldots + f_0 \quad \text{(for the case of } k = 15\text{)}$$

The DLPDU presented to the PhL shall consist of an octet sequence in the specified order.

The redundant check bits f_{n-k-1} … f_0 of the FCS shall be the coefficients of the remainder F(X), after division by G(X), of $L(X)\,(X^k + 1) + M(X)\,X^{n-k}$

where G(X) is the degree ***n-k*** generator polynomial for the code words

$$G(X) = X^{n-k} + g_{n-k-1}X^{n-k-1} + \dots + 1 \qquad (14)$$

and L(X) is the maximal weight (all ones) polynomial of degree ***n-k****-1*

$$L(X) = \frac{X^{n-k}+1}{X+1} = X^{n-k-1} + X^{n-k-2} + \dots + X + 1 \qquad (15)$$

$$= X^{15} + X^{14} + X^{13} + X^{12} + \dots + X^2 + X + 1 \qquad \text{(for the case of k = 15)}$$

That is,

$$F(X) = L(X)\,(X^k + 1) + M(X)\,X^{n-k} \;\; (\text{modulo } G(X)) \qquad (16)$$

NOTE 1 The L(X) terms are included in the computation to detect initial or terminal message truncation or extension by adding a length-dependent factor to the FCS.

NOTE 2 As a typical implementation when n-k = 16, the initial remainder of the division is preset to all ones. The transmitted message bit stream is multiplied by X^{n-k} and divided (modulo 2) by the generator polynomial G(X), specified in equation (7). The ones complement of the resulting remainder is transmitted as the (n-k)-bit FCS, with the coefficient of X^{n-k-1} transmitted first.

5.2.5.2 At the receiving DLE

The octet sequence indicated by the PhE shall be concatenated into the received DLPDU and FCS, and regarded as a vector V(X) of dimension *u*

$$V(X) = v_1X^{u-1} + v_2X^{u-2} + \dots + v_{u-1}X + v_u \qquad (17)$$

NOTE 1 Because of errors u can differ from n, the dimension of the transmitted code vector.

A remainder R(X) shall be computed for V(X), the received DLPDU and FCS, by a method similar to that used by the sending DLE (see 5.2.5.1) in computing F(X)

$$R(X) = L(X)\,X^u + V(X)\,X^{n-k} \;\; (\text{modulo } G(X)) \qquad (18)$$

$$= r_{n-k-1}X^{n-k-1} + \dots + r_0$$

Define E(X) to be the error code vector of the additive (modulo-2) differences between the transmitted code vector D(X) and the received vector V(X) resulting from errors encountered (in the PhS provider and in bridges) between sending and receiving DLEs.

$$E(X) = D(X) + V(X) \qquad (19)$$

If no error has occurred, so that E(X) = 0, then R(X) will equal a non-zero constant remainder polynomial

$$R_{ok}(X) = L(X)\,X^{n-k} \;\; (\text{modulo } G(X)) \qquad (20)$$

whose value is independent of D(X). Unfortunately R(X) will also equal $R_{ok}(X)$ in those cases where E(X) is an exact non-zero multiple of G(X), in which case there are “undetectable” errors. In all other cases, R(X) will not equal $R_{ok}(X)$; such DLPDUs are erroneous and shall be discarded without further analysis.

NOTE 2 As a typical implementation, the initial remainder of the division is preset to all ones. The received bit stream is multiplied by X^{n-k} and divided (modulo 2) by the generator polynomial G(X), specified in equation (7).

5.2.5.3 Modification within bridges

When forwarding a Type 1 DLPDU, it is sometimes necessary for a bridge to alter one or more subfields of a DLPDU’s frame control field. When making these modifications, the bridge can be required to modify the received FCS to compensate for changes in the frame control octet; in this case the bridge does not discard the received FCS and recompute a new FCS after the DLPDU's frame control field has been altered[2]).

When the received DLPDU's length, plus that of its FCS field, is N octets, then the bridge can compensate for a change in bit K in the first octet by computing the residual of the polynomial

$$X^{8N+K-8} \ (\text{modulo} \ \ G(X)) \qquad (21)$$

and then updating the DLPDU's FCS field by exclusive-ORing the computed residual into that field.

NOTE When the bridge initializes, it can precompute the residuals for all permissible DLPDU lengths and bit positions potentially needing alteration, that is, for all values of N between 3 and 272 and of K equal to 2. Then for any DLPDU the bridge need only apply to the DLPDU's FCS that residual which corresponds to the change actually made in the DLPDU's frame control octet.

5.2.6 Elements of procedure for minimum-inter-DLPDU delay

The DLE which holds the dominant token shall start transmission only after providing a delay of at least minimum-inter-DLPDU delay, V(MID), octet durations, where the delay is measured as specified in 4.7.1.12.

5.2.7 Elements of procedure for dropping of token by dominant token holder

If the DLE which holds the dominant token has just completed either

a) the transmission of a DLPDU, as indicated by receipt of a Ph-DATA confirm primitive corresponding to the most-recent Ph-DATA request primitive, which shall have specified END-OF-DATA-AND-ACTIVITY, or

b) the reception of a DLPDU, as indicated by receipt of a Ph-Data indication primitive specifying either END-OF-DATA-AND-ACTIVITY or END-OF-DATA or END-OF-ACTIVITY

but has not yet started its next transmission, either

c) because it is waiting for the required interval of minimum-inter-DLPDU delay, or

d) because it is not ready to transmit due to delay in preparing the next transmission;

then if that DLE is able to receive during this interval and if a Ph-Data indication (see 4.4.4) reporting DATA is received, then the DLE shall drop the token.

5.2.8 Common elements of procedure for monitoring link activity

A number of DLPDU types (CA, CD, ED, CT, RQ, PT, ES, TL) sent by a token holder request an immediate response. Common procedures apply for those DLEs which

a) need to monitor for that immediate response;

b) need to determine
   – when to stop monitoring for that response;
   – whether or not that response occurred.

5.2.8.1 Monitoring for an immediate response by the initiating DLE

After sending a CA, CD, ED, CT or RQ, DLPDU which requests an immediate response,

a) where the sending DLE is not also the responding DLE, then the sending DLE shall

2) D. R. Irvin, *Preserving the integrity of cyclic-redundancy checks when protected text is intentionally altered*, IBM Journal of Research and Development, Vol. 33, No. 6, November 1989, pp. 618-626.

1) monitor the local link for a period of immediate-response-recovery-delay slot-times, V(IRRD) $\times$ V(ST) octet-durations, waiting for a reply;

2) then take appropriate action based on the result of that monitoring;

b) where the sending DLE is also the responding DLE, it shall take action as if the monitoring had occurred and had successfully detected the requested immediate response DLPDU.

The monitoring procedure is:

1) If a Ph-Data indication (see 4.4.4) reporting DATA is received, then the DLE shall

   i) stop further monitoring;

   ii) then wait for receipt of the Ph-Data indication reporting END-OF-DATA-AND-ACTIVITY or END-OF-ACTIVITY;

2) If 1) does not apply, and the monitoring period expires, then

   i) if the bus is not active at that moment (that is, the last-received Ph-Data indication reported END-OF-ACTIVITY), then the DLE shall stop further monitoring.

   ii) if i) does not apply, implying that the link is active at that moment (that is, the last-received Ph-Data indication reported START-OF-ACTIVITY), then

      α) For implementations that were not demonstrable on or before 31 December 1995, the DLE shall monitor the local link for a period of one additional slot-time, V(ST) octet-durations, waiting for a Ph-Data indication:

         A) If a Ph-Data indication reporting DATA is received, then the DLE shall proceed as in 1).

         B) If a Ph-Data indication reporting END-OF-ACTIVITY is received, and A) does not apply, then the DLE shall stop further monitoring.

         C) If neither A) nor B) applies

         D) and the monitoring period expires before a Ph-Data indication is received, then the DLE shall wait for receipt of the Ph-Data indication reporting END-OF-DATA-AND-ACTIVITY or END-OF-ACTIVITY.

      β) Implementations that were demonstrable on or before 31 December 1995 alternatively may just proceed as in α).C) when neither α).A) nor α).B) applies.

      NOTE After this standard achieved initial ACDV status, implementors were encouraged to develop chips to assist in evaluating this complex protocol. As a consequence, it was found desirable to improve the noise rejection characteristics of the token passing process, and the text 2)ii)α), which would require changes in those existing chips, was the result. β) grandfathers the noise rejection approach of the first ACDV for those early implementations, and only for those early implementations.

At the end of monitoring, the sending DLE shall act based on the result of that monitoring:

3) If 2)i) or 2)ii)α)B) applied, then the sending DLE shall

   – report the failure to detect a DLPDU to local DL-management;

   – if the final-token-use subfield of the originating CA, CD, ED, CT or RQ DLPDU specified NOT-FINAL, then the DLE shall start the next transmission within immediate-response-recovery-delay plus one slot-times, (V(IRRD) + 1) $\times$ V(ST) octet-durations, of the end of transmission of that CA, CD, ED, CT or RQ DLPDU.

4) If 1) or 2)ii)α)A) or 2)ii)α)C) applied and link activity did not result in a DLPDU, then the sending DLE shall

   – report the failure to detect a valid reply to local DL-management;

   – if the final-token-use subfield of the originating CA, CD, ED, CT or RQ DLPDU specified NOT-FINAL, then the DLE shall start the next transmission within maximum-reply-delay slot-times, V(MRD) $\times$ V(ST) octet-durations, of the beginning of the current period of link non-activity;

5) If

   – the link activity resulted in a DLPDU;

– if the received DLPDU was not a permissible reply DLPDU,

then any held token shall be dropped and local DL-management shall be notified of the event.

NOTE These DL-management reports may take the form of incrementing a DL-management error counter.

6) If

– the link activity resulted in a DLPDU;

– the received DLPDU was a permissible reply DLPDU;

– the final-token-use subfield of the originating CA, CD, ED, CT or RQ DLPDU specified NOT-FINAL,

then the DLE shall start the next transmission within maximum-reply-delay slot-times, V(MRD) × V(ST) octet-durations, of the beginning of the current period of link non-activity.

5.2.8.2 Monitoring for an immediate response by the LAS DLE

If the final-token-use subfield of the received CA, CD, ED or RQ DLPDU has the value FINAL, then

a) if the LAS DLE is not the addressed immediate responder DLE, then the LAS DLE shall apply the monitoring procedures of 5.2.8.1 1) and 5.2.8.1 2);

b) if the LAS DLE is also the responding DLE, then it shall take action as if the monitoring had occurred and had successfully detected the immediate response DLPDU.

After completing the monitoring a), or after sending the requested immediate response b), the LAS DLE shall

1) assume that the current use of the delegated token has terminated and that the scheduler token is again dominant on the local link; and

2) treat that termination as if the token had been returned by an RT (see 6.17) DLPDU;

3) resume active operation as the LAS.

5.2.8.3 Monitoring for an immediate response as a subscriber to a DLC

A DLE which

– receives a CA, CD or ED DLPDU requesting an immediate response;

– is an intended recipient of that requested immediate response DLPDU;

– is not itself the CA, CD or ED DLPDU's addressed responder,

shall initiate a timer with a duration of immediate-response-recovery-delay slot-times, V(IRRD) × V(ST) octet-durations:

a) If a Ph-Data indication (see 4.4.4) reporting DATA is received, then the DLE shall stop further monitoring and wait for receipt of the Ph-Data indication reporting END-OF-DATA-AND-ACTIVITY or END-OF-ACTIVITY and proceed as in e).

b) If the monitoring period expires, and the bus is not active at the end of this monitoring period, then the DLE shall stop further monitoring and proceed as in d).

c) If the monitoring period expires, and the bus is active at the end of this monitoring period, then the DLE shall wait for receipt of the Ph-Data indication reporting END-OF-DATA-AND-ACTIVITY or END-OF-ACTIVITY, and then proceed as in d) or e) as appropriate.

d) If the monitoring did not result in a DLPDU as in b) or possibly in c), then the DLE shall invalidate V(RA).

e) If the link activity resulted in a DLPDU as in a) or possibly in c), and if the received DLPDU was a DT DLPDU whose source DL-address is implicitly or explicitly the destination DLCEP-address specified by the CA, CD or ED DLPDU, then the recorded address V(RA) shall be assumed to be the implied source address of that CA, CD or ED DLPDU. Otherwise the recorded address V(RA) shall be invalidated.

5.2.9 Two-way alternate (half-duplex) operation

There is no requirement for a sending DLE to receive PH-DATA indications while it is transmitting a DLPDU; but when such indications are not receivable (that is, when the DLE is operating in a two-way alternate or half-duplex mode), then a sending DLE shall treat each transmitted DLPDU as if it had been received concurrently by that DLE.

6 DLPDU-specific structure, encoding and elements of procedure

NOTE 1 The elements of procedure in this clause correspond generally to the lowest DLL sublevel, as specified in 4.1.1.

Clause 6 defines the structure, contents and encoding of each DLPDU type and format, and specifies elements of procedure for that DLPDU type and format. (Table 10 summarizes their structure.)

Within each subclause, the DLPDU's structure, contents and encoding are described first. Then those aspects relating to the sending and receiving DLS-users and their DLEs are addressed, followed by those aspects, if any, that are unique to the store-and-forwarding functions of bridges (relay DLEs).

Throughout Clause 6, whenever a conditional action is specified and the specified enabling condition does not occur, then the corresponding action also does not occur. The net effect is that events which do not meet any of the enabling conditions specified in a service procedure will have no consequential actions with respect to that specific service procedure.

A DLE which is not ONLINE to the local link, but which is attempting to change its DL-state to ONLINE, may send a DLPDU only as a response to a PN DLPDU addressed to the DLE's DL-support functions, as specified in 6.13.4, 6.14.3 and 10.2.1. During this interval the DL-protocol specifies the response of the DLE to received PN, PT and DT DLPDUs addressed to the DLE's DL-support functions. The DLE's behavior after it changes its state to ONLINE shall not be dependent in any way on the receipt before it went ONLINE of other than the above-specified (PN, PT, DT) DLPDUs.

NOTE 2 Receipt of a DLPDU without observable side-effects is a purely local matter, and is thus outside the scope of standardization.

A DLE which is ONLINE to the local link may send a DLPDU only when it holds the dominant token on the local link. The set of DLPDUs which may be sent while holding the dominant token is dependent on the class of the dominant token, as summarized in Table 11:

a) If the dominant token is a scheduler token, then the permitted DLPDU classes are CA, CD, DC, DT, EC, ED, ES, PN, PT, RC, RQ, TD, TL, WK and IDLE.
b) If the dominant token is a delegated token, then the permitted DLPDU classes are CA, CD, CT, DC, DT, EC, ED, RC, RI, RQ, RT, WK and IDLE.
c) If the dominant token is a reply token, then the permitted DLPDU classes are
   1) DT and SR when replying to a CA, CD or ED DLPDU;
   2) TD when replying to a CT DLPDU;
   3) PR when replying to a PN DLPDU;
   4) RR when replying to an RQ DLPDU.
d) Tokens are created only by the following DLPDUs:
   1) A scheduler token is created only by a CL DLPDU;
   2) A delegated token is created only by PT and ES DLPDUs;

3) A reply token is created only by CA, CD, CT, ED, PN, RQ and TL DLPDUs.

Table 10 – Summary structure of DLPDUs

DLPDU class	Frame control	DL-addresses			Parameters	User data
		Destination	Source	2nd source		
EC 1	1111 LF00	[HL.]N.S	[HL.]N.S	[HL.]N.S	EC-p	o-DLSDU
EC 2	1110 LF00		[HL.]N.S	[HL.]N.S	EC-p	o-DLSDU
DC 1	0111 LF00	[HL.]N.S	[HL.]N.S		DC-p	o-DLSDU
DC 2	0110 LF00		[HL.]N.S		DC-p	o-DLSDU
RC 1	0111 LFPP	[HL.]N.S	[HL.]N.S		RC-p	o-DLSDU
RC 2	0110 LFPP		[HL.]N.S		RC-p	o-DLSDU
CA 1	1110 LFPP	[HL.]N.S	[HL.]N.S		SD-p	o-pDLSDU
CA 2	1010 LFPP	[HL.]N.S	—		SD-p	o-pDLSDU
CD 1	1111 LFPP	[HL.]N.S	[HL.]N.S		o-SD-p	—
CD 2	1011 LFPP	[HL.]N.S	—		o-SD-p	—
ED 1	1100 LFPP	[HL.]N.S	[HL.]N.S		SD-p	pDLSDU
ED 2	1000 LFPP	[HL.]N.S	—		SD-p	pDLSDU
DT 1	1101 LFPP	[HL.]N.S	[HL.]N.S		SD-p	o-pDLSDU
DT 2	1001 LFPP	[HL.]N.S	—		SD-p	o-pDLSDU
DT 3	0101 LFPP		[HL.]N.S		SD-p	o-pDLSDU
DT 4	1001 0F00	[PSA]	—		SD-p	o-pDLSDU
DT 5	0101 0F00		[PDA]		SD-p	o-pDLSDU
SR	0001 0F11	[PSA]	N		o-SR-p	—
CT	0001 0F00	—	—		—	—
TD	0001 0F01	—	N		TD-p	—
RQ	1100 0F00	N.0	N.0		RQ-p	—
RR	1101 0F00	N.0	N.0		RR-p	—
PN	0010 0110	N	—		PN-p	—
PR	0010 0111	—	—		—	SPDU
PT	0011 0FPP	N	—		DD-p	—
ES	1000 LF00	[HL.]N.S	—		DD-p	—
RT	0011 0100	—	[DTH]		—	—
RI	0010 0000	—	[DTH]		DD-p	—
CL	0000 0001	—	N		—	—
TL	0000 0110	N	—		—	SPDU
WK	0000 0000	N	—		—	—
IDLE	0001 0F10	—	—		—	o-DLSDU

where

L indicates the length of the associated DL-addresses (0 = SHORT, 1 = LONG)
F indicates final use of a token, or that a sequence should be finished rather than restarted
PP specifies the priority of the DLPDU and any passed token
shading indicates a logically non-existent field
— indicates a logically existent field whose contents are required to be null
[HL.]N.S is a four-octet LONG DL-address (HLNS) when L = 1
or a two-octet SHORT DL-address (NS) with HL = 00 implied when L = 0
N is a one-octet NODE DL-address
N.0 is the two-octet SHORT DL-address form of a one-octet NODE DL-address
[PDA] is the implied DL-address equal to the explicit destination DL-address of the immediately prior DLPDU on the link, which must have been a CA, CD or ED DLPDU. This is a logically existent field whose actual contents are required to be null
[PSA] is the implied DL-address equal to the implied or explicit source DL-address of the immediately prior DLPDU on the link. This is a logically existent field whose actual contents are required to be null
[DTH] is the implied DL-address equal to the explicit destination DL-address of the most recent PT or ES DLPDU on the link – the DLPDU which delegated the token returned by the RT or RI DLPDU. This is a logically existent field whose actual contents are required to be null
o- indicates optional field contents
xx-p indicates xx-class DLPDU parameters
DLSDU is a DL Service Data Unit
pDLSDU is a complete or partial DLSDU
SPDU is a Support Protocol Data Unit.

Table 11 – DLPDU restrictions based on dominant token

DLPDU class	Type of token created	Can be sent while using Scheduler token	Can be sent while using Delegated token	Can be sent in Reply to
EC	—	Y	Y	—
DC	—	Y	Y	—
RC	—	Y	Y	—
CA	Reply	Y	Y	—
CD	Reply	Y	Y	—
ED	Reply	Y	Y	—
DT	—	Y	Y	CA, CD, ED
SR	—	N	N	CA, CD, ED, TL
CT	Reply	N	Y	—
TD	—	Y	N	CT
RQ	Reply	Y	Y	—
RR	—	N	N	RQ
PN	Reply	Y	N	—
PR	—	N	N	PN
PT	Delegated	Y	N	—
ES	Delegated	Y	N	—
RT	—	N	Y	—
RI	—	N	Y	—
CL	—	N	N	—
TL	Scheduler	Y	N	—
WK	—	Y	Y	—
IDLE	—	Y	Y	—
where – means that no token is created.				

6.1 Establish connection (EC) DLPDU

An ESTABLISH CONNECTION (EC) DLPDU is used to establish a peer DLC between two DLS-users, or a multi-peer DLC between a publishing DLS-user and subscribing DLS-users.

6.1.1 Structure of the EC DLPDU

Table 12 – Structure of EC DLPDUs

format	Frame control	Destination address	Source address	2nd source address	Parameters	User data
1L	1111 1F00	HL.N.S	HL.N.S	HL.N.S	EC-p	o-DLSDU
1S	1111 0F00	N.S	N.S	N.S	EC-p	o-DLSDU
2L	1110 1F00		HL.N.S	HL.N.S	EC-p	o-DLSDU
2S	1110 0F00		N.S	N.S	EC-p	o-DLSDU

6.1.1.1 Frame control field

The frame control field shall specify

a) the DLPDU's function;

b) the DLPDU's implicit priority, which is TIME-AVAILABLE;

c) the length, number and type of DLPDU addresses;

d) whether or not transmission of the current DLPDU terminates use of a delegated token.

6.1.1.2 Address field

The address field shall consist of either

a) an explicit destination DL-address and two explicit source DL-addresses, in that order, for formats 1L and 1S, or

b) only two explicit source DL-addresses, for formats 2L and 2S.

For formats 1L and 2L, all addresses shall be LONG; for formats 1S and 2S, all addresses shall be SHORT.

6.1.1.3 Parameters field

The establish-connection parameters (EC-parameters) field specifies the proposed or selected QoS attributes of the DLC, including the version of the DL protocol in use. This field shall be structured and encoded as described in 7.1.

6.1.1.4 User data field

The user data field shall consist of a single optional DLSDU whose maximum size is limited to 128 octets.

6.1.2 Content of the EC DLPDU

The frame control field shall be encoded as specified in Table 12.

For formats 1L and 1S,

a) the first DL-address shall be a DL(SAP)-address or a DLCEP-address;

b) the second DL-address shall be a DLCEP-address or a DLSAP-address;

c) the third DL-address shall be a DLSAP-address of the DLSAP associated with that second DL-address.

For formats 2L and 2S, the first DL-address shall be a publisher DLCEP-address, and the second DL-address shall be a DLSAP-address of the DLSAP associated with that first DL-address.

The EC-parameters shall specify the proposed or selected QoS attributes of the DLC, including the version of the DL-protocol in use. The contents of this field shall be as described in 7.1.

6.1.3 Sending the EC DLPDU

An EC DLPDU may be sent on the link when the sending DLE holds a scheduler token or delegated token which is the dominant token on the local link, and when the remaining allocated duration of token usage, C(RD), permits completion of the EC DLPDU's transmission prior to expiration of the token.

Each DL-address in the DLPDU shall be delocalized (see 5.2.2) before transmission.

If the DLE holds a delegated token, and no additional use of that token after sending this DLPDU is needed at that time, then the DLE may set the final-token-use subfield of the EC DLPDU to the value FINAL; else that subfield shall have the value NOT-FINAL.

6.1.4 Receiving the EC DLPDU

Each DL-address in the DLPDU shall be delocalized (see 5.2.2) upon reception.

A received EC DLPDU shall be treated as follows by the receiving DLE.

6.1.4.1 Actions required of all DLEs

If the first DL-address specified by the DLPDU designates an active DL(SAP)-address or an active DLCEP of the DLE, then the received DLPDU shall be forwarded to the DLE's upper-level functions (see 8.3) for further processing.

6.1.4.2 Additional actions required of a link-master class DLE

None.

6.1.4.3 Additional actions required of a bridge class DLE

The following considerations apply:

a) Since every bridge class DLE has link master capability, any actions specified in 6.1.4.2 also apply to a bridge class DLE.
b) If the first DL-address specified in the DLPDU is one which the bridge should forward, and the bridge was able to buffer the DLPDU without error, then the received DLPDU shall be forwarded with modification of the frame-control field in the forwarded DLPDU as appropriate (see 5.2.5.3).
c) The bridge shall attempt to update its routing table entry for the source DLCEP-address and source DLSAP-address specified in the DLPDU to reflect the bridge port from which the DLPDU was received.
d) If the DLPDU's Establish-Connection parameters indicate that the DLPDU's addressee(s) will be the subscriber(s) of a multi-peer DLC, then the bridge shall attempt to update its routing table entry for the source DLCEP-address specified in the DLPDU by adding the bridge port(s) to which the DLPDU is being forwarded to the set of sink DLCEP ports associated with that source DLCEP-address.

NOTE This last procedure is only meaningful for formats 1L and 1S of the EC DLPDU.

6.1.4.4 Additional actions required of the current LAS DLE

If the final-token-use subfield of the received DLPDU has the value FINAL, then the LAS DLE shall

a) assume that the current use of the delegated token is terminated as if the token had been returned by an RT (see 6.17) DLPDU;
b) assume that the scheduler token is again dominant on the local link and resume active operation as the LAS.

6.2 Disconnect connection (DC) DLPDU

A DISCONNECT CONNECTION (DC) DLPDU is used to disconnect an existing or proposed DLC.

6.2.1 Structure of the DC DLPDU

Table 13 – Structure of DC DLPDUs

format	Frame control	Destination address	Source address	Parameters	User data
1L	0111 1F00	HL.N.S	HL.N.S	DC-p	o-DLSDU
1S	0111 0F00	N.S	N.S	DC-p	o-DLSDU
2L	0110 1F00		HL.N.S	DC-p	o-DLSDU
2S	0110 0F00		N.S	DC-p	o-DLSDU

6.2.1.1 Frame control field

The frame control field shall specify

a) the DLPDU's function;

b) the DLPDU's implicit priority, which is TIME-AVAILABLE;

c) the length, number and type of DLPDU addresses;

d) whether or not transmission of the current DLPDU terminates use of a delegated token.

6.2.1.2 Address field

The address field shall consist of either

a) an explicit destination DL-address and an explicit source DL-address, in that order, for formats 1L and 1S, or

b) only an explicit source DL-address, for format 2L and 2S.

For formats 1L and 2L, all addresses shall be LONG; for formats 1S and 2S, all addresses shall be SHORT.

6.2.1.3 Parameters field

The disconnect-connection parameters (DC-parameters) field shall specify the version of the DL protocol in use, the desired action and the reason for that action. This field shall be structured and encoded as described in 7.27.2.

6.2.1.4 User data field

The user data field shall consist of a single optional DLSDU whose maximum size is limited to 128 octets.

6.2.2 Content of the DC DLPDU

The frame control field shall be encoded as specified in Table 13.

For formats 1L and 1S, either

a) both DL-addresses shall be peer DLCEP-addresses, or

b) the first DL-address shall be a DL(SAP)-address, and the second DL-address shall be a DLCEP-address, or

c) the first DL-address shall be a DLCEP-address, and the second DL-address shall be a DLSAP-address.

For formats 2L and 2S, the sole DL-address shall be a publisher DLCEP-address.

The DC-parameters shall specify the version of the DL-protocol in use, the desired action and reason, and other information. The contents of this field shall be encoded as described in 7.27.2.

6.2.3 Sending the DC DLPDU

A DC DLPDU may be sent on the link when the sending DLE holds a scheduler token or delegated token which is the dominant token on the local link, and when the remaining allocated duration of token usage, C(RD), permits completion of the DLPDU's transmission prior to expiration of the token.

Each DL-address in the DLPDU shall be delocalized (see 5.2.2) before transmission.

If the DLE holds a delegated token, and no additional use of that token after sending this DLPDU is needed at that time, then the DLE may set the final-token-use subfield of the DC DLPDU to the value FINAL; else that subfield shall have the value NOT-FINAL.

6.2.4 Receiving the DC DLPDU

Each DL-address in the DLPDU shall be delocalized (see 5.2.2) upon reception.

A received DC DLPDU shall be treated as follows by the receiving DLE.

6.2.4.1 Actions required of all DLEs

If the first DL-address specified by the DLPDU designates an active DLSAP-address or an active DLCEP of the receiving DLE, then the received DLPDU shall be forwarded to the DLE's upper-level functions (see 8.2.1.7) for further processing.

6.2.4.2 Additional actions required of a link-master class DLE

None.

6.2.4.3 Additional actions required of a bridge class DLE

The following considerations apply:

a) Since every bridge class DLE has link master capability, any actions specified in 6.2.4.2 also apply to a bridge class DLE.
b) If the first DL-address specified in the DLPDU is one which the bridge should forward, and the bridge was able to buffer the DLPDU without error, then the received DLPDU shall be forwarded with modification of the frame-control field in the forwarded DLPDU as appropriate (see 5.2.5.3).
c) If the DLPDU's Disconnect-Connection parameters indicate that a reply is not requested, then the bridge may attempt to remove the routing table entries for any explicit DLCEP addresses specified in the DLPDU.

6.2.4.4 Additional actions required of the current LAS DLE

If the final-token-use subfield of the received DLPDU has the value FINAL, then the LAS DLE shall

a) assume that the current use of the delegated token is terminated as if the token had been returned by an RT (see 6.17) DLPDU;
b) assume that the scheduler token is again dominant on the local link and resume active operation as the LAS.

6.3 Reset connection (RC) DLPDU

A RESET CONNECTION (RC) DLPDU is used to reset an existing or proposed DLC.

6.3.1 Structure of the RC DLPDU

Table 14 – Structure of RC DLPDUs

format	Frame control	Destination address	Source address	Parameters	User data
1L	0111 1FPP	HL.N.S	HL.N.S	RC-p	o-DLSDU
1S	0111 0FPP	N.S	N.S	RC-p	o-DLSDU
2L	0110 1FPP		HL.N.S	RC-p	o-DLSDU
2S	0110 0FPP		N.S	RC-p	o-DLSDU

6.3.1.1 Frame control field

The frame control field shall specify

a) the DLPDU's function;

b) the DLPDU's priority;

c) the length, number and type of DLPDU addresses;

d) whether or not transmission of the current DLPDU terminates use of a delegated token.

6.3.1.2 Address field

The address field shall consist of either

a) an explicit destination DL-address and an explicit source DL-address, in that order, for formats 1L and 1S, or

b) only an explicit source DL-address, for format 2L and 2S.

For formats 1L and 2L, all addresses shall be LONG; for formats 1S and 2S, all addresses shall be SHORT.

6.3.1.3 Parameters field

The reset-connection parameters (RC-parameters) field shall specify the version of the DL protocol in use, the desired action and reason, and other information. This field shall be structured and encoded as described in 7.3.

6.3.1.4 User data field

The user data field shall consist of a single optional DLSDU whose maximum size is limited to the maximum DLS-user data size permitted for a DLPDU of the priority specified in 6.3.1.1b).

6.3.2 Content of the RC DLPDU

The frame control field shall be encoded as specified in Table 14.

For formats 1L and 1S, both DL-addresses shall be peer DLCEP-addresses. For formats 2L and 2S, the sole DL-address shall be a publisher DLCEP-address.

The RC-parameters shall specify the version of the DL protocol in use, the desired action and reason, and other information. The contents of this field shall be encoded as described in 7.3.

6.3.3 Sending the RC DLPDU

An RC DLPDU may be sent on the link when the sending DLE holds a scheduler token or delegated token which is the dominant token on the local link, and when the remaining allocated duration of token usage, C(RD), permits completion of the DLPDU's transmission prior to expiration of the token.

Each DL-address in the DLPDU shall be delocalized (see 5.2.2) before transmission.

If the DLE holds a delegated token, and no additional use of that token after sending this DLPDU is needed at that time, then the DLE may set the final-token-use subfield of the RC DLPDU to the value FINAL; else that subfield shall have the value NOT-FINAL.

6.3.4 Receiving the RC DLPDU

Each DL-address in the DLPDU shall be delocalized (see 5.2.2) upon reception.

A received RC DLPDU shall be treated as follows by the receiving DLE.

6.3.4.1 Actions required of all DLEs

If the first DL-address specified by the DLPDU designates an active DLCEP of the receiving DLE, then the received DLPDU shall be forwarded to the DLE's upper-level functions (see 8.2) for further processing.

6.3.4.2 Additional actions required of a link-master class DLE

None.

6.3.4.3 Additional actions required of a bridge class DLE

The following considerations apply:

a) Since every bridge class DLE has link master capability, any actions specified in 6.3.4.2 also apply to a bridge class DLE.
b) If the receiving DLE is a bridge, and the first DL-address specified in the DLPDU is one which the bridge should forward, and the bridge was able to buffer the DLPDU without error, then the received DLPDU shall be forwarded with modification of the frame-control field in the forwarded DLPDU as appropriate (see 5.2.5.3).

6.3.4.4 Additional actions required of the current LAS DLE

If the final-token-use subfield of the received DLPDU has the value FINAL, then the LAS DLE shall

a) assume that the current use of the delegated token is terminated as if the token had been returned by an RT (see 6.17) DLPDU;
b) assume that the scheduler token is again dominant on the local link and resume active operation as the LAS.

6.4 Compel acknowledgement (CA) DLPDU

A COMPEL ACKNOWLEDGMENT (CA) DLPDU is used

a) to transfer (or retransfer) a limited amount of transparent user data from one requesting DLS-user to another DLS-user;
b) to request that the transfer be acknowledged as soon as possible,

without requiring that the transaction (request and acknowledgment) occur within the context of a DLC. It is also used for similar purposes within the context of a DLC, and to assist in the synchronization of DLCEPs and of their DLS-users.

A CA DLPDU creates and passes a reply token to the addressed receiving DLE, which upon reception becomes the dominant token on its link. A CA DLPDU requires an immediate reply of either a DATA (DT) DLPDU with no DLS-user data or a STATUS RESPONSE (SR) DLPDU. If no DLPDU is received in reply, then the transaction is repeated a maximum of V(MRC) times.

6.4.1 Structure of the CA DLPDUs

Table 15 – Structure of CA DLPDUs

format	Frame control	Destination address	Source address	Parameters	User data
1L	1110 1FPP	HL.N.S	HL.N.S	SD-p	o-pDLSDU
1S	1110 0FPP	N.S	N.S	SD-p	o-pDLSDU
2L	1010 1FPP	HL.N.S	—	SD-p	o-pDLSDU
2S	1010 0FPP	N.S	—	SD-p	o-pDLSDU

6.4.1.1 Frame control field

The frame control field shall specify

a) the DLPDU's function, including that an immediate reply is being requested;

b) the transaction's and DLPDU's priority;

c) the length, number and type of DLPDU addresses;

d) whether or not transmission of the current DLPDU, and either its expected immediate reply or the appropriate link-idle timeout, terminates use of a delegated token.

NOTE This field necessarily has the value NOT-FINAL when an immediate retry of the current transaction is possible. Only a transaction which is guaranteed to not need an immediate retry may have the value FINAL specified in its associated DLPDU.

6.4.1.2 Address field

The address field shall consist of either

a) an explicit destination DL-address and an explicit source DL-address, in that order, for formats 1L and 1S, or

b) only an explicit destination DL-address, for formats 2L and 2S.

For formats 1L and 2L, all addresses shall be LONG; for formats 1S and 2S, all addresses shall be SHORT.

6.4.1.3 Parameters field

The status-data-parameters (SD-parameters) field specifies information appropriate to the associated destination DL-address:

a) If that DL-address is a DLSAP-address, then the SD-parameters specify

   1) a transaction-id used by the originating DLE to correlate a delayed returned reply with the originating request;

   2) a DLSDU-priority used to convey the actual priority of the accompanying DLSDU to the responding DLE, or that there is no accompanying DLSDU.

   This field shall be structured and encoded as described in 7.4.1.

b) If that DL-address is a DLCEP-address, then the SD-parameters specify state information for the addressed DLCEP. This field shall be structured and encoded as described in 7.4.2.

NOTE The size and structure of this field is dependent on the QoS attributes associated with the DLCEP addressed by the destination DL-address specified in this DLPDU, and is determined during DLCEP establishment.

6.4.1.4 User data field

The user data field size and content are limited by the associated destination DL-address:

a) If that DL-address is a DLSAP-address, then the user data field shall consist of DLS-user data whose maximum size is limited to the smaller of the maximum DLS-user data sizes permitted for a DLPDU of the priorities specified in 6.4.1.1b) and 6.4.1.3a)2), and shall not be null.
b) If that DL-address is a DLCEP-address, then the user data field shall consist of DLS-user data whose maximum size is limited to the smaller of
   1) the maximum DLS-user data size permitted for a DLPDU of the priority specified in 6.4.1.1b);
   2) the maximum DLSDU size negotiated on the DLC for data transmission to that DLCEP,

   and may be null.

6.4.2 Content of the CA DLPDU

The frame control field shall be encoded as specified in Table 15.

Either the DL-addresses shall be

a) two explicit DLSAP-addresses, or
b) one explicit DLCEP-address, and a second explicit or implicit DLCEP-address, or
c) one explicit DLCEP-address, followed by one explicit or implicit DLSAP-address.

6.4.2.1 Content of the CA DLPDU when specifying a destination DLSAP-address

When the first DL-address is a DLSAP-address as in 6.4.2a), then

a) if the DLPDU format is format 1L or 1S, then
   1) this DLPDU is being used to implement the unitdata transfer service with remote DLE confirmation;
   2) the DL(SAP)-role for the destination DLSAP-address shall be BASIC;
   3) the second address shall be present, shall be a DLSAP-address, and the DL(SAP)-role for that DLSAP-address shall be BASIC;
   4) the SD-parameters field shall specify a DLSDU-priority and a transaction-id used by the originating DLE to correlate a delayed returned reply with the originating request, where
      i) the contents of this field shall be as described in 7.4;
      ii) the DLSDU-priority shall be the priority of the accompanying user data and shall be the same as the DLPDU-priority specified in 6.4.1.1b);
   5) the user data shall be a single DLSDU whose size is limited to the maximum size for the priority specified in 6.4.1.4a), and shall not be null;
b) no other DLPDU format may be used.

6.4.2.2 Content of the CA DLPDU when specifying a destination DLCEP-address

When the first address is a DLCEP-address, as in 6.4.2b) and 6.4.2c), then

a) this DLPDU can convey a single or partial DLSDU
   – from one peer DLCEP to its corresponding peer DLCEP, or
   – from a subscriber DLCEP to its corresponding publisher DLCEP,

   shall request state information from the addressed DLCEP, and shall not permit DLS-user data to be included in the reply DLPDU;

b) the second address, if present,
   – shall be the peer DLCEP-address of the same DLC as the destination peer DLCEP-address, or
   – shall be a subscriber DLCEP's calling-DLSAP-address of the same DLC as the destination publisher DLCEP-address;

c) the SD-parameters field shall specify state information for the addressed DLCEP, and the contents of this field shall be as described in 7.4;

   NOTE The size and structure of this field is dependent on the QoS attributes associated with the DLCEP addressed by the destination DL-address specified in this DLPDU, and is determined during DLCEP establishment.

d) the user data shall specify those octets of a DLSDU consistent with the negotiated DLSDU size and the segmentation information specified in the accompanying SD-parameters, and may be null.

   NOTE A CA DLPDU with null user data can be used by a DLE with a subscriber DLCEP to solicit current DLC state information from the corresponding publisher DLCEP.

NOTE Formats 1L and 1S are used for peer-to-peer (see 6.4.2b)) and subscriber-to-publisher (see 6.4.2c)) communications when the DLPDU-authentication attribute is SOURCE or MAXIMAL. Formats 2L and 2S are used for peer-to-peer and subscriber-to-publisher communications when the DLPDU-authentication attribute is ORDINARY.

It is a protocol error if the above conditions are not met.

6.4.3 Sending the CA DLPDU

A CA DLPDU may be selected for transmission on the link when

a) the sending DLE holds a scheduler token or delegated token which is the dominant token on the local link;

b) the remaining allocated duration of token usage, C(RD), permits completion of V(MRC)+1 implied transactions prior to expiration of the token, where each transaction consists of sending the CA DLPDU that requires an immediate reply, and awaiting a worst-case SR DLPDU or worst-case permitted DT reply DLPDU not containing DLS-user data;

c) if the CA DLPDU will be addressed to a DLSAP-address, then the outstanding-transaction-array, V(OTA) (see 4.7.1.15), searched circularly from the last-transaction-index, V(LTI) (see 4.7.1.16), has an unassigned entry whose index is not V(LTI).

Once selected, if the CA DLPDU will be addressed to a DLSAP-address, then

– V(LTI) shall be set to the index of that unassigned entry in V(OTA);
– that entry in V(OTA) shall be assigned to the selected CA DLPDU and shall record information which permits an expected response DT DLPDU to be correlated with the specific invocation of the unitdata-transfer service which gave rise to the CA DLPDU.

Once selected, transmission of the CA DLPDU shall be retried until either

1) a permissible immediate reply DLPDU is received, or
2) an impermissible DLPDU is received when an immediate reply was expected, or
3) the original transmission and the permitted maximum number of transmission retries, V(MRC) (see 4.7.1.5), have all failed to elicit one of the permissible reply DLPDUs.

If the DLE holds a delegated token, and no additional use of that token after sending this DLPDU and awaiting its immediate reply is needed at that time, then the DLE may set the final-token-use subfield of the CA DLPDU to the value FINAL; else that subfield shall have the value NOT-FINAL.

NOTE The FINAL transaction is necessarily the (V(MRC)+1)'th in a series; otherwise, if a reply is not received, then another CA DLPDU would have to be sent before the current cycle of token use was completed, making the current transaction NOT-FINAL.

Each explicit DL-address in the CA DLPDU shall be delocalized (see 5.2.2) before transmission.

After sending a CA DLPDU, the sending DLE shall monitor the local link for a reply as specified in 5.2.8.1. The permissible reply DLPDU is either

– a DT DLPDU whose destination DL-address is implicitly or explicitly the originating DLSAP-address or DLCEP specified by the CA DLPDU, or
– a DT DLPDU without a destination DL-address, or
– an SR DLPDU.

If V(LTI) was assigned to the transaction, as a result of the search described in c), and if a permissible reply DLPDU was not received, then the V(LTI)'th entry shall be deassigned.

6.4.4 Receiving the CA DLPDU

Each DL-address in the DLPDU shall be delocalized (see 5.2.2) upon reception.

A received CA DLPDU shall be treated as follows by the receiving DLE.

6.4.4.1 Actions required of all DLEs

NOTE The next alternative attempts to detect the reception of a duplicated CA DLPDU resulting from an immediate retry by the current token-holding DLE, which itself was probably caused by an error detected during receipt of the earlier reply DT DLPDU. In such a case the response DT DLPDU is required to be identical to the first in those fields and subfields which convey DLS-user data and related information.

The following considerations apply:

a) If

   1) the destination DL-address specified by the DLPDU designates an active DLSAP-address of the receiving DLE, or an active DLCEP-address of a DLC for which the receiving DLE is peer or publisher;
   2) the immediately prior DLPDU was transmitted as an immediate-response to a received CA DLPDU whose destination DL-address was the same DLSAP-address or DLCEP-address;
   3) if a source DL-address is present in the just-received CA DLPDU, then it was also present and identical in that prior received CA DLPDU;
   4) starting with the SD-parameters field, the first three octets of the just-received CA DLPDU, or the remainder of the DLPDU if fewer than three octets, are identical to the corresponding octets of that prior received CA DLPDU;
   5) a period of link inactivity of (immediate-response-recovery-delay + 1) slot-times, (V(IRRD) + 1) × V(ST) octet-durations, has not occurred since receipt of that prior received CA DLPDU,

   then the receiving DLE

      i) shall discard the received DLPDU and not forward it to the DLE's upper-level functions for further processing;

ii) shall retransmit the prior-transmitted immediate-reply DT DLPDU, within a period of maximum-response-delay slot-times, V(MRD) × V(ST) octet-durations, of receipt of the CA DLPDU, unchanged from the prior transmission except for those subfields of

– the final-token-use subfield of the frame control octet which conveys information to the listening LAS;

– those subfields of any present DLCEP SD-parameters which acknowledge receipt of, or request retransmission of, received DLSDUs.

NOTE This requirement is meant to ensure that the DL-priority, content and identity of any DLS-user data conveyed in the immediate reply DT DLPDU is identical to that in the immediately prior DT DLPDU sent from this same DL-address, including the originator's transaction-id if one is present in the received and transmitted SD-parameters.

b) If a) does not apply, and the destination DL-address specified by the DLPDU designates a DLSAP-address of the receiving DLE, then the received DLPDU shall be forwarded to the DLE's upper-level functions (see 8.3) for further processing.

NOTE Group DL-addresses are not covered here; such CA DLPDUs are erroneous and are unrecognized upon receipt.

The receiving DLE shall initiate a reply within a period of maximum-response-delay slot-times, V(MRD) × V(ST) octet-durations, of receipt of the CA DLPDU. The reply DLPDU shall be a DT DLPDU with any explicit destination DL-address in the DT DLPDU having the same value as the source DL-address of the received CA DLPDU, and length and format as follows:

– DT (see 6.7.1 format 2L) in response to CA (see 6.4.1 format 1L), or

– DT (see 6.7.1 format 4) in response to CA (see 6.4.1 format 1S),

and shall include SD-parameters specifying

– the originator's transaction-id as received in the stimulating CA DLPDU, as specified in 6.4.2.1a)4);

– reception status for the DLSDU conveyed by that CA DLPDU.

The reply DT DLPDU shall not contain any DLS-user data.

c) If a) does not apply, and the destination DL-address specified by the DLPDU designates an active DLCEP-address of a DLC for which the receiving DLE

– is a peer or publisher, and the DLL priority of the DLCEP is not equal to the priority specified in the received DLPDU, or

– is a peer, and the length and number of DL-addresses is not as expected or the DLPDU specifies an explicit source address which is not equal to the remote peer DLCEP's DLCEP-address,

then

1) the received DLPDU shall be forwarded to the DLE's upper-level functions (see 8.2.2.9) for further processing;

2) the receiving DLE shall initiate a reply within a period of maximum-response-delay slot-times, V(MRD) × V(ST) octet-durations, of receipt of the CA DLPDU. The reply DLPDU shall be a DT DLPDU in the format negotiated for the DLC for the selected direction of transmission, and shall contain SD-parameters appropriate to the sending DLCEP, but shall not contain DLS-user data.

d) If neither a) nor c) applies, and the destination DL-address specified by the DLPDU designates an active DLCEP-address of a DLC for which the receiving DLE is a peer or publisher, then

1) the received DLPDU shall be forwarded to the DLE's upper-level functions (see 8.2.2.9) for further processing;

2) the receiving DLE shall initiate a reply within a period of maximum-response-delay slot-times, V(MRD) × V(ST) octet-durations, of receipt of the CA DLPDU. The reply DLPDU shall be a DT DLPDU in the format negotiated for the DLC for the selected direction of transmission, shall contain SD-parameters appropriate to the sending DLCEP, but shall not contain DLS-user data;

3) the requesting CA DLPDU may contain information about the state of the requesting DLCEP. The reply DT DLPDU is permitted, but is not required, to reflect that state information in its reply; immediate processing of that state information before sending the immediate reply shall be permitted but shall not be required.

e) If a) does not apply, and the destination DL-address specified by the DLPDU designates an active DLCEP-address of a DLC for which the receiving DLE is a subscriber, then

1) the non-DLS-user data portion of the received DLPDU shall be forwarded to the DLE's upper-level functions (see 8.2.2.9) for further processing;

2) the receiving DLE shall record the destination DL-address from the received CA DLPDU in V(RA) for subsequent association with the expected immediate reply DT DLPDU, which should be the next DLPDU received;

3) the receiving DLE shall monitor the local link for a reply and then act based on the result of that monitoring, all as specified in 5.2.8.3.

6.4.4.2 Additional actions required of a link-master class DLE

None.

6.4.4.3 Additional actions required of a bridge class DLE

The following considerations apply:

a) Since every bridge class DLE has link master capability, any actions specified in 6.1.4.2 also apply to a bridge class DLE.

b) If all of the following conditions hold

1) the destination DL-address specified in the DLPDU is one which the bridge should forward but to which the bridge DLE itself would not otherwise generate an immediate reply DLPDU;

2) the immediately prior DLPDU was transmitted as an immediate-response to a received CA DLPDU with the same destination DL-address;

3) if a source DL-address is present in the just-received CA DLPDU, then it was also present and identical in that prior received CA DLPDU;

4) the basic-DLC-parameters portion of the SD-parameters of the just-received CA DLPDU is identical to that of that prior received CA DLPDU;

5) a period of link inactivity of immediate-response-recovery-delay plus one slot-time, (V(IRRD)+1) × V(ST) octet-durations, has not occurred since receipt of that prior received CA DLPDU,

then the bridge

i) shall discard the received DLPDU and not forward it to the bridge's other functions for further processing;

ii) shall initiate retransmission of the prior-transmitted immediate-reply SR DLPDU, unchanged from the prior transmission.

NOTE This requirement is meant to ensure that any status conveyed in the immediate reply SR DLPDU is identical to that in the immediately prior SR DLPDU.

Otherwise,

A) If the destination DL-address specified in the DLPDU is one which the bridge should forward but which the bridge DLE itself would not otherwise receive, then the bridge shall form and send an SR DLPDU

– within a period of maximum-response-delay slot-times, V(MRD) × V(ST) octet-durations, of receipt of the CA DLPDU;

– with status indicating whether or not the bridge was able to buffer the received DLPDU;

B) If the destination DL-address specified in the DLPDU is one which the bridge should forward, and the bridge was able to receive and buffer the DLPDU without error, then the received DLPDU shall be forwarded with modification of the frame-control field in the forwarded DLPDU as appropriate (see 5.2.5.3);

C) If the DLPDU contains an explicit source DL-address, then the bridge shall attempt to update its routing table entry for the source DL-address specified in the DLPDU to reflect the bridge port from which the DLPDU was received.

c) It is a protocol error for a bridge DLE which will forward the received CA DLPDU to not send an SR reply DLPDU when a reply is required.

NOTE At most one bridge DLE on the local link should be forwarding the received CA DLPDU.

6.4.4.4 Additional actions required of the current LAS DLE

The LAS DLE shall act as specified in 5.2.8.2.

6.5 Compel data (CD) DLPDU

A COMPEL DATA (CD) DLPDU is used to request the transfer (or retransfer) of a limited amount of transparent user data from another DLS-user to the requesting DLS-user without requiring that the transaction (request and acknowledgment) occur within the context of a DLC. It is also used for similar purposes within the context of a DLC, and to assist in the synchronization of DLCEPs and of their DLS-users. When the CD DLPDU is addressed to a publishing DLCEP, then the DLS-user data whose transfer is requested will be distributed to all of the subscribers of the DLC.

A CD DLPDU creates and passes a reply token to the addressed receiving DLE, which upon reception becomes the dominant token on the link. A CD DLPDU requires an immediate reply of either a DATA (DT) DLPDU or a STATUS RESPONSE (SR) DLPDU. If no DLPDU is received in reply, then the transaction is repeated a maximum of V(MRC) times.

6.5.1 Structure of the CD DLPDUs

Table 16 – Structure of CD DLPDUs

format	Frame control	Destination address	Source address	Parameters
1L	1111 1FPP	HL.N.S	HL.N.S	o-SD-p
1S	1111 0FPP	N.S	N.S	o-SD-p
2L	1011 1FPP	HL.N.S	—	o-SD-p
2S	1011 0FPP	N.S	—	o-SD-p

6.5.1.1 Frame control field

The frame control field shall specify

a) the DLPDU's function, including that an immediate reply is being requested;

b) the transaction's priority and DLPDU's implied priority;

c) the length, number and type of DLPDU addresses;

d) whether or not transmission of the current DLPDU, and either its expected immediate reply or the appropriate link-idle timeout, terminates use of a delegated token.

NOTE This field necessarily has the value NOT-FINAL when an immediate retry of the current transaction is possible. Only a transaction which is guaranteed not to need an immediate retry may have the value FINAL specified in its associated DLPDU.

6.5.1.2 Address field

The address field shall consist of either

a) an explicit destination DL-address and an explicit source DL-address, in that order, for formats 1L and 1S, or
b) only an explicit destination DL-address, for formats 2L and 2S.

For formats 1L and 2L, all addresses shall be LONG; for formats 1S and 2S, all addresses shall be SHORT.

6.5.1.3 Parameters field

The status-data-parameters (SD-parameters) field specifies information appropriate to the associated destination DL-address:

a) If that DL-address is a DLSAP-address bound in a responder DL(SAP)-role, then the SD-parameters specify a transaction-id used by the originating DLE to correlate a delayed returned reply with the originating request. This field shall be structured and encoded as described in 7.4.1.
b) If that DL-address is a DLCEP-address, then the SD-parameters specify state information for the addressed DLCEP. This field may be null. When non-null, this field shall be structured and encoded as described in 7.4.2.

NOTE The size and structure of this field is dependent on the QoS attributes associated with the DLCEP addressed by the destination DL-address specified in this DLPDU, and is determined during DLCEP establishment.

When the CD DLPDU is sent by the LAS DLE, which occurs while the dominant token is a scheduler token, then this field shall be null.

6.5.1.4 User data field

The user data field shall be null.

6.5.2 Content of the CD DLPDU

The frame control field shall be encoded as specified in Table 16.

Either the DL-addresses shall be

a) two explicit DLSAP-addresses, or
b) one explicit DLCEP-address, and a second explicit or implicit or not-present DLCEP-address, or
c) one explicit DLCEP-address, followed by one explicit or implicit DLSAP-address.

6.5.2.1 Content of the CD DLPDU when specifying a destination DLSAP-address

When the first DL-address is a DLSAP-address as in 6.5.2a), then

a) if the DLPDU format is format 1L or 1S, then
   1) this DLPDU is being used to implement the unitdata exchange service;
   2) the DL(SAP)-role for the destination DLSAP-address shall be CONSTRAINED RESPONDER or UNCONSTRAINED RESPONDER;
   3) the second address shall be present, shall be a DLSAP-address, and the DL(SAP)-role for that DLSAP-address shall be INITIATOR;

4) the SD-parameters field shall specify a transaction-id used by the originating DLE to correlate a delayed returned reply with the originating request, where the contents of this field shall be as described in 7.4;

5) the user data shall be null.

b) no other DLPDU format may be used.

6.5.2.2 Content of the CD DLPDU when specifying a destination DLCEP-address

When the first address is a DLCEP-address, as in 6.5.2b) and 6.5.2c), then

a) this DLPDU shall request state information from the addressed DLCEP, and shall request that DLS-user data be included in the reply DLPDU;

b) the second address, if present,
   - shall be the peer DLCEP-address of the same DLC as the destination peer DLCEP-address, or
   - shall be a subscriber DLCEP's calling-DLSAP-address of the same DLC as the destination publisher DLCEP-address;

c) the SD-parameters field, if present, shall specify state information for the addressed DLCEP, and the contents of this field shall be as described in 7.4;

   NOTE The size and structure of this field is dependent on the QoS attributes associated with the DLCEP addressed by the destination DL-address specified in this DLPDU, and is determined during DLCEP establishment.

d) the user data shall be null.

   NOTE If the user data field is not null, then an ED DLPDU (see 6.6) should be used instead.

NOTE Formats 1L and 1S are used for peer-to-peer (see 6.5.2b)) and subscriber-to-publisher (see 6.5.2c)) communications when the DLPDU-authentication attribute is SOURCE or MAXIMAL. Formats 2L and 2S are used for peer-to-peer and subscriber-to-publisher communications when the DLPDU-authentication attribute is ORDINARY.

6.5.3 Sending the CD DLPDU

A CD DLPDU may be selected for transmission on the link when

a) the sending DLE holds a scheduler token or delegated token which is the dominant token on the local link;

b) the remaining allocated duration of token usage, C(RD), permits completion of V(MRC)+1 implied transactions prior to expiration of the token, where each transaction consists of sending the CD DLPDU that requires an immediate reply, and awaiting a worst-case SR DLPDU or worst-case permitted DT reply DLPDU containing DLS-user data;

c) if the CD DLPDU will be addressed to a DLSAP-address, then the outstanding-transaction-array, V(OTA) (see 4.7.1.15), searched circularly from the last-transaction-index, V(LTI) (see 4.7.1.16), has an unassigned entry whose index is not V(LTI).

Once selected, if the CD DLPDU will be addressed to a DLSAP-address, then

- V(LTI) shall be set to the index of that unassigned entry in V(OTA);
- that entry in V(OTA) shall be assigned to the selected CD DLPDU and shall record information which permits an expected response DT DLPDU to be correlated with the specific invocation of the unitdata-exchange service which gave rise to the CD DLPDU.

Once selected, transmission of the CD DLPDU shall be retried until either

1) a permissible immediate reply DLPDU is received, or
2) an impermissible DLPDU is received when an immediate reply was expected, or
3) the original transmission and the permitted maximum number of transmission retries, V(MRC) (see 4.7.1.5), have all failed to elicit one of the permissible reply DLPDUs.

If the DLE holds a delegated token, and no additional use of that token after sending this DLPDU and awaiting its immediate reply is needed at that time, then the DLE may set the final-token-use subfield of the CD DLPDU to the value FINAL; else that subfield shall have the value NOT-FINAL.

NOTE The FINAL transaction is necessarily the (V(MRC)+1)'th in a series; otherwise, if a reply is not received, then another CD DLPDU would have to be sent before the current cycle of token use was completed, making the current transaction NOT-FINAL.

Each explicit DL-address in the CD DLPDU shall be delocalized (see 5.2.2) before transmission.

After sending a CD DLPDU, the sending DLE shall monitor the local link for a reply as specified in 5.2.8.1. The permissible reply DLPDU is either

– a DT DLPDU whose destination DL-address is implicitly or explicitly the originating DLSAP-address or DLCEP specified by the CD DLPDU, or
– a DT DLPDU without a destination DL-address, or
– an SR DLPDU.

If V(LTI) was assigned to the transaction, as a result of the search described in c), and if a permissible reply DLPDU was not received, then the V(LTI)'th entry shall be deassigned.

6.5.4 Receiving the CD DLPDU

Each DL-address in the DLPDU shall be delocalized (see 5.2.2) upon reception.

A received CD DLPDU shall be treated as follows by the receiving DLE.

6.5.4.1 Actions required of all DLEs

NOTE The next alternative attempts to detect the reception of a duplicated CD DLPDU resulting from an immediate retry by the current token-holding DLE, which itself was probably caused by an error detected during receipt of the earlier reply DT DLPDU. In such a case the response DT DLPDU is required to be identical to the first in those fields and subfields which convey DLS-user data and related information.

The following considerations apply:

a) If
   1) the destination DL-address specified by the DLPDU designates an active DLSAP-address of the receiving DLE, or an active DLCEP-address of a DLC for which the receiving DLE is peer or publisher;
   2) the immediately prior DLPDU was transmitted as an immediate-response to a received CD DLPDU whose destination DL-address was the same DLSAP-address or DLCEP-address;
   3) if a source DL-address is present in the just-received CD DLPDU, then it was also present and identical in that prior received CD DLPDU;
   4) starting with the SD-parameters field, the first three octets of the just-received CD DLPDU, or the remainder of the DLPDU if fewer than three octets, are identical to the corresponding octets of that prior received CD DLPDU;
   5) a period of link inactivity of (immediate-response-recovery-delay + 1) slot-times, (V(IRRD) + 1) × V(ST) octet-durations, has not occurred since receipt of that prior received CD DLPDU,

   then the receiving DLE

      i) shall discard the received DLPDU and not forward it to the DLE's upper-level functions for further processing;

ii) shall retransmit the prior-transmitted immediate-reply DT DLPDU, within a period of maximum-response-delay slot-times, V(MRD) × V(ST) octet-durations, of receipt of the CD DLPDU, unchanged from the prior transmission except for those subfields of

– the final-token-use subfield of the frame control octet which conveys information to the listening LAS;

– those subfields of any present DLCEP SD-parameters which acknowledge receipt of, or request retransmission of, received DLSDUs.

NOTE This requirement is meant to ensure that the DL-priority, content and identity of any DLS-user data conveyed in the immediate reply DT DLPDU is identical to that in the immediately prior DT DLPDU sent from this same DL-address, including the originator's transaction-id if one is present in the received and transmitted SD-parameters.

b) If a) does not apply, and the destination DL-address specified by the DLPDU designates a DLSAP-address of the receiving DLE, then the processing of the received DLPDU shall be based upon the DL(SAP)-role specified for that DLSAP-address.

NOTE Group DL-addresses are not covered here; such CD DLPDUs are erroneous and are unrecognized upon receipt.

The receiving DLE shall initiate a reply within a period of maximum-response-delay slot-times, V(MRD) × V(ST) octet-durations, of receipt of the CD DLPDU:

1) If the DLE can prepare it in time, then the reply DLPDU shall be a DT DLPDU with any explicit destination DL-address in the DT DLPDU having the same value as the source DL-address of the received CD DLPDU, and length and format as follows:

– DT (see 6.7.1 format 2L) in response to CD (see 6.5.1 format 1L);

– DT (see 6.7.1 format 2S) in response to CD (see 6.5.1 format 1S) if the DT DLPDU contains DLS-user data, or

– DT (see 6.7.1 format 4) in response to CD (see 6.5.1 format 1S) if the DT DLPDU does not contain DLS-user data,

and shall include SD-parameters specifying

– the originator's transaction-id as received in the stimulating CD DLPDU, as specified in 6.5.2.1a)4);

– status for the DLSDU requested by that CD DLPDU.

The reply DT DLPDU may include a DLSDU which was already buffered at that responding DLE at the time of CD reception.

NOTE This restriction prohibits the reply DLPDU from reflecting any higher-layer (than DLL) processing of the received DLSDU. This restriction is necessary to enable migration from prior national standards.

If the DL(SAP)-role of the destination DLSAP-address specified

i) BASIC or INITIATOR, then the DLE shall reject the received CD DLPDU based on this DL(SAP)-role, and an appropriate error status without an accompanying DLSDU shall be included in the reply DT DLPDU;

ii) UNCONSTRAINED RESPONDER, then the SD-parameter status shall indicate the highest DLL priority sending buffer with a non-null DLSDU which was available at the addressed DLSAP at the time of reception of the CD DLPDU;

A) if that DLL priority is greater than or equal to the priority specified in 6.5.1.1b), then that DLSDU shall be included in the reply DT DLPDU and its buffer set to empty if so configured, and the DLL priority of that reply DT DLPDU shall be the DLL priority of the conveyed DLSDU; or

B) if that DLL priority is less than the priority specified in 6.5.1.1b), then no DLSDU shall be included in the reply DT DLPDU, and the DLL priority of that reply DT DLPDU shall be the DLL priority specified in 6.5.1.1b); or

C) if there is no such sending buffer with a non-null DLSDU, then the reply DT DLPDU shall specify an appropriate error status, the DLL priority of that reply DT DLPDU shall be the DLL priority specified in 6.5.1.1b), and that reply DT DLPDU shall not contain DLS-user data.

iii) constrained responder, then

A) if the source DLSAP-address of the CD DLPDU is equal to the remote DLSAP address which was specified in the prior DL-BIND request primitive for the receiving DLSAP-address (or its DL-management equivalent), both after delocalization, then the procedure specified in ii) shall be followed;

B) otherwise, the received DLSDU shall be discarded, and an appropriate error status without an accompanying DLSDU shall be included in the reply DT DLPDU.

2) If the DLE cannot prepare the required reply DT DLPDU in time, then the DLE shall send a DT DLPDU with any explicit destination DL-address in the DT DLPDU having the same value as the source DL-address of the received CD DLPDU, and length and format as follows:

– DT (see 6.7.1 format 2L) in response to CD (see 6.5.1 format 1L), or

– DT (see 6.7.1 format 4) in response to CD (see 6.5.1 format 1S),

and shall include SD-parameters

– with a transaction-id identical to the transaction-id from the received CD DLPDU, as specified in 6.5.2.1a)4);

– with a status (DR – "delayed reply") indicating that the DLE requires additional time to prepare the required response;

– with a null user data field,

and the DLE

– shall prepare that DT DLPDU as specified in 1) as soon as possible;

– shall include an explicit destination address in the reply DT DLPDU;

– shall append that reply DT DLPDU to the DLE's Q(US) to be transmitted at the first opportunity.

c) If a) does not apply, and the destination DL-address specified by the DLPDU designates an active DLCEP-address of a DLC for which the receiving DLE

– is a peer or publisher, and the DLL priority of the DLCEP is not equal to the priority specified in the received DLPDU, or

– is a peer, and the length and number of DL-addresses is not as expected or the DLPDU specifies an explicit source address which is not equal to the remote peer DLCEP's DLCEP-address,

then

1) the received DLPDU shall be forwarded to the DLE's upper-level functions (see 8.2.2.9) for further processing;

2) the receiving DLE shall initiate a reply within a period of maximum-response-delay slot-times, $V(MRD) \times V(ST)$ octet-durations, of receipt of the CD DLPDU. The reply DLPDU shall be a DT DLPDU in the format negotiated for the DLC for the selected direction of transmission, and shall contain SD-parameters appropriate to the sending DLCEP, but shall not contain DLS-user data.

d) If neither a) nor c) applies, and the destination DL-address specified by the DLPDU designates an active DLCEP-address of a DLC for which the receiving DLE is a peer or publisher, then

1) the received DLPDU shall be forwarded to the DLE's upper-level functions (see 8.2.2.9) for further processing;

2) the receiving DLE shall initiate a reply within a period of maximum-response-delay slot-times, V(MRD) × V(ST) octet-durations, of receipt of the CD DLPDU. The reply DLPDU shall be a DT DLPDU in the format negotiated for the DLC for the selected direction of transmission, shall contain SD-parameters appropriate to the sending DLCEP, and shall contain DLS-user data if any was available and waiting for transmission or retransmission from the DLCEP;

NOTE Migration from prior national standards requires that this data always be included when available.

3) The requesting CD DLPDU may contain information about the state of the requesting DLCEP. The reply DT DLPDU is permitted, but is not required, to reflect that state information in its reply; immediate processing of that state information before sending the immediate reply shall be permitted but shall not be required.

e) If a) does not apply, and the destination DL-address specified by the DLPDU designates an active DLCEP-address of a DLC for which the receiving DLE is a subscriber, then

1) the non-DLS-user data portion of the received DLPDU shall be forwarded to the DLE's upper-level functions (see 8.2.2.9) for further processing;

2) the receiving DLE shall record the destination DL-address from the received CD DLPDU in V(RA) for subsequent association with the expected immediate reply DT DLPDU, which should be the next DLPDU received;

3) the receiving DLE shall monitor the local link for a reply and then act based on the result of that monitoring, all as specified in 5.2.8.3.

6.5.4.2 Additional actions required of a link-master class DLE

None.

6.5.4.3 Additional actions required of a bridge class DLE

The following considerations apply:

a) Since every bridge class DLE has link master capability, any actions specified in 6.5.4.2 also apply to a bridge class DLE.

b) If all of the following conditions hold

A) the destination DL-address specified in the DLPDU is one which the bridge should forward but to which the bridge DLE itself would not otherwise generate an immediate reply DLPDU;

B) the immediately prior DLPDU was transmitted as an immediate-response to a received CD DLPDU with the same destination DL-address;

C) if a source DL-address is present in the just-received CD DLPDU, then it was also present and identical in that prior received CD DLPDU;

D) the basic-DLC-parameters portion of the SD-parameters of the just-received CD DLPDU is identical to that of that prior received CD DLPDU;

E) a period of link inactivity of immediate-response-recovery-delay plus one slot-times, (V(IRRD)+1) × V(ST) octet-durations, has not occurred since receipt of that prior received CD DLPDU,

then the bridge

1) shall discard the received DLPDU and not forward it to the bridge's other functions for further processing;

2) shall initiate retransmission of the prior-transmitted immediate-reply SR DLPDU, unchanged from the prior transmission.

NOTE This requirement is meant to ensure that any status conveyed in the immediate reply SR DLPDU is identical to that in the immediately prior SR DLPDU.

Otherwise

i) If the destination DL-address specified in the DLPDU is one which the bridge should forward but which the bridge DLE itself would not otherwise receive, then the bridge shall form and send an SR DLPDU,

   – within a period of maximum-response-delay slot-times, V(MRD) × V(ST) octet-durations, of receipt of the CD DLPDU;

   – with status indicating whether or not the bridge was able to buffer the received DLPDU.

ii) If the destination DL-address specified in the DLPDU is one which the bridge should forward, and the bridge was able to receive and buffer the DLPDU without error, then the received DLPDU shall be forwarded with modification of the frame-control field in the forwarded DLPDU as appropriate (see 5.2.5.3).

iii) If the DLPDU contains an explicit source DL-address, then the bridge shall attempt to update its routing table entry for the source DL-address specified in the DLPDU to reflect the bridge port from which the DLPDU was received.

c) It is a protocol error for a bridge DLE which will forward the received CD DLPDU to not send an SR reply DLPDU when a reply is required.

NOTE At most one bridge DLE on the local link should be forwarding the received CD DLPDU.

6.5.4.4 Additional actions required of the current LAS DLE

The LAS DLE shall act as specified in 5.2.8.2.

6.6 Exchange data (ED) DLPDU

An EXCHANGE DATA (ED) DLPDU is used

a) to transfer (or retransfer) a limited amount of transparent user data from the requesting DLS-user to another DLS-user;

b) to request the transfer (or retransfer) of a limited amount of transparent user data from that other DLS-user to the requesting DLS-user,

without requiring that the transaction (request and acknowledgment) occur within the context of a DLC. It is also used for similar purposes within the context of a DLC, and to assist in the synchronization of DLCEPs and of their DLS-users. When the ED DLPDU is addressed to a publishing DLCEP, then the DLS-user data whose transfer is requested in b) will be distributed to all of the subscribers of the DLC.

An ED DLPDU creates and passes a reply token to the addressed receiving DLE, which upon reception becomes the dominant token on the link. An ED DLPDU requires an immediate reply of either a DATA (DT) DLPDU or a STATUS RESPONSE (SR) DLPDU. If no DLPDU is received in reply, then the transaction is repeated a maximum of V(MRC) times.

6.6.1 Structure of the ED DLPDUs

Table 17 – Structure of ED DLPDUs

format	Frame control	Destination address	Source address	Parameters	User data
1L	`1100 1FPP`	HL.N.S	HL.N.S	SD-p	pDLSDU
1S	`1100 0FPP`	N.S	N.S	SD-p	pDLSDU
2L	`1000 1FPP`	HL.N.S	—	SD-p	pDLSDU
2S	`1000 0FPP`	N.S	—	SD-p	pDLSDU

6.6.1.1 Frame control field

The frame control field shall specify

a) the DLPDU's function, including that an immediate reply is being requested;

b) the transaction's priority, and the DLPDU's implied priority unless explicitly overridden by 6.6.1.3a)2);

c) the length, number and type of DLPDU addresses;

d) whether or not transmission of the current DLPDU, and either its expected immediate reply or the appropriate link-idle timeout, terminates use of a delegated token.

NOTE This field necessarily has the value NOT-FINAL when an immediate retry of the current transaction is possible. Only a transaction which is guaranteed to not need an immediate retry may have the value FINAL specified in its associated DLPDU.

6.6.1.2 Address field

The address field shall consist of either

a) an explicit destination DL-address and an explicit source DL-address, in that order, for formats 1L and 1S, or

b) only an explicit destination DL-address, for formats 2L and 2S.

For formats 1L and 2L, all addresses shall be LONG; for formats 1S and 2S, all addresses shall be SHORT.

6.6.1.3 Parameters field

The status-data-parameters (SD-parameters) field specifies information appropriate to the associated destination DL-address:

a) If that DL-address is a DLSAP-address bound in a responder DL(SAP)-role, then the SD-parameters specify

   1) a transaction-id used by the originating DLE to correlate a delayed returned reply with the originating request;

   2) a DLSDU-priority used to convey the actual priority of the accompanying DLSDU to the responding DLE.

   This field shall be structured and encoded as described in 7.4.1.

b) If that DL-address is a DLCEP-address, then the SD-parameters specify state information for the addressed DLCEP. This field shall be structured and encoded as described in 7.4.2.

   NOTE The size and structure of this field is dependent on the QoS attributes associated with the DLCEP addressed by the destination DL-address specified in this DLPDU, and is determined during DLCEP establishment.

6.6.1.4 User data field

The user data field size and content are limited by the associated destination DL-address:

a) If that DL-address is a DLSAP-address, then the user data field shall consist of DLS-user data whose maximum size is limited to the smaller of the maximum DLS-user data sizes permitted for a DLPDU of the priority specified in 6.6.1.1b) and 6.6.1.3a)2), and shall not be null.

b) If that DL-address is a DLCEP-address, then the negotiated DLC attributes for the intended direction of transmission shall permit DLS-user data to be carried in an ED DLPDU (see 7.1c)6) and 7.1d)6)), and the user data field shall consist of DLS-user data whose maximum size is limited to the smaller of

   1) the maximum DLS-user data size permitted for a DLPDU of the priority specified in 6.6.1.1b);

   2) the maximum DLSDU size negotiated on the DLC for data transmission to that DLCEP,

and shall not be null.

6.6.2 Content of the ED DLPDU

The frame control field shall be encoded as specified in Table 17.

Either the DL-addresses shall be

a) two explicit DLSAP-addresses, or
b) one explicit DLCEP-address, and a second explicit or implicit DLCEP-address, or
c) one explicit DLCEP-address, followed by one explicit or implicit DLSAP-address.

6.6.2.1 Content of the ED DLPDU when specifying a destination DLSAP-address

When the first DL-address is a DLSAP-address as in 6.6.2a), then

a) if the DLPDU format is format 1L or 1S, then
   1) this DLPDU is being used to implement the unitdata exchange service;
   2) the DL(SAP)-role for the destination DLSAP-address shall be CONSTRAINED RESPONDER or UNCONSTRAINED RESPONDER;
   3) the second address shall be present, shall be a DLSAP-address, and the DL(SAP)-role for that DLSAP-address shall be INITIATOR;
   4) the SD-parameters field shall specify a DLSDU-priority and a transaction-id used by the originating DLE to correlate a delayed returned reply with the originating request, where the contents of this field shall be as described in 7.4, and the DLSDU-priority shall be the priority of the accompanying user data and shall be the same as specified in 6.6.1.3a)2) and shall be as least as high as the DLPDU priority specified in 6.6.1.1b);
   5) the user data shall be a single DLSDU whose size is limited to the maximum size for the priority specified in 6.6.1.4a), and shall not be null.
b) no other DLPDU format may be used.

6.6.2.2 Content of the ED DLPDU when specifying a destination DLCEP-address

When the first address is a DLCEP-address, as in 6.6.2b) and 6.6.2c), then

a) this DLPDU can convey a single or partial DLSDU
   – from one peer DLCEP to its corresponding peer DLCEP, or
   – from a subscriber DLCEP to its corresponding publisher DLCEP,

   shall request state information from the addressed DLCEP, and shall request that DLS-user data be included in the reply DLPDU, and

b) the second address, if present,
   – shall be the peer DLCEP-address of the same DLC as the destination peer DLCEP-address, or
   – shall be a subscriber DLCEP's calling-DLSAP-address of the same DLC as the destination publisher DLCEP-address;
c) the SD-parameters field shall specify state information for the addressed DLCEP, and the contents of this field shall be as described in 7.4;

   NOTE The size and structure of this field is dependent on the QoS attributes associated with the DLCEP addressed by the destination DL-address specified in this DLPDU, and is determined during DLCEP establishment.
d) the user data shall specify those octets of a DLSDU consistent with the negotiated DLSDU size and the segmentation information specified in the accompanying SD-parameters, and shall not be null.

NOTE If the user data field is null, then a CD DLPDU (see 6.5) should be used instead.

NOTE Formats 1L and 1S are used for peer-to-peer (see 6.6.2b) and subscriber-to-publisher (see 6.6.2c)) communications when the DLPDU-authentication attribute is SOURCE or MAXIMAL. Formats 2L and 2S are used for peer-to-peer and subscriber-to-publisher communications when the DLPDU-authentication attribute is ORDINARY.

6.6.3 Sending the ED DLPDU

An ED DLPDU may be selected for transmission on the link when

a) the sending DLE holds a scheduler token or delegated token which is the dominant token on the local link;
b) the remaining allocated duration of token usage, C(RD), permits completion of V(MRC)+1 implied transactions prior to expiration of the token, where each transaction consists of sending the ED DLPDU that requires an immediate reply, and awaiting a worst-case SR DLPDU or worst-case permitted DT reply DLPDU not containing DLS-user data;
c) if the ED DLPDU will be addressed to a DLSAP-address, then the outstanding-transaction-array, V(OTA) (see 4.7.1.15), searched circularly from the last-transaction-index, V(LTI) (see 4.7.1.16), has an unassigned entry whose index is not V(LTI).

Once selected, if the ED DLPDU will be addressed to a DLSAP-address, then

– V(LTI) shall be set to the index of that unassigned entry in V(OTA);
– that entry in V(OTA) shall be assigned to the selected ED DLPDU and shall record information which permits an expected response DT DLPDU to be correlated with the specific invocation of the unitdata-exchange service which gave rise to the ED DLPDU.

Once selected, transmission of the ED DLPDU shall be retried until either

1) a permissible immediate reply DLPDU is received, or
2) an impermissible DLPDU is received when an immediate reply was expected, or
3) the original transmission and the permitted maximum number of transmission retries, V(MRC) (see 4.7.1.5), have all failed to elicit one of the permissible reply DLPDUs.

If the DLE holds a delegated token, and no additional use of that token after sending this DLPDU and awaiting its immediate reply is needed at that time, then the DLE may set the final-token-use subfield of the ED DLPDU to the value FINAL; else that subfield shall have the value NOT-FINAL.

NOTE The FINAL transaction is necessarily the (V(MRC)+1)'th in a series; otherwise, if a reply is not received, then another ED DLPDU would have to be sent before the current cycle of token use was completed, making the current transaction NOT-FINAL.

Each explicit DL-address in the ED DLPDU shall be delocalized (see 5.2.2) before transmission.

After sending an ED DLPDU, the sending DLE shall monitor the local link for a reply as specified in 5.2.8.1. The permissible reply DLPDU is either

– a DT DLPDU whose destination DL-address is implicitly or explicitly the originating DLSAP-address or DLCEP specified by the ED DLPDU, or
– a DT DLPDU without a destination DL-address, or
– an SR DLPDU.

If V(LTI) was assigned to the transaction, as a result of the search described in c), and if a permissible reply DLPDU was not received, then the V(LTI)'th entry shall be deassigned.

6.6.4 Receiving the ED DLPDU

Each DL-address in the DLPDU shall be delocalized (see 5.2.2) upon reception.

A received ED DLPDU shall be treated as follows by the receiving DLE.

6.6.4.1 Actions required of all DLEs

NOTE The next alternative attempts to detect the reception of a duplicated ED DLPDU resulting from an immediate retry by the current token-holding DLE, which itself was probably caused by an error detected during receipt of the earlier reply DT DLPDU. In such a case the response DT DLPDU is required to be identical to the first in those fields and subfields which convey DLS-user data and related information.

The following considerations apply:

a) If

1) the destination DL-address specified by the DLPDU designates an active DLSAP-address of the receiving DLE, or an active DLCEP-address of a DLC for which the receiving DLE is peer or publisher;

2) the immediately prior DLPDU was transmitted as an immediate-response to a received ED DLPDU whose destination DL-address was the same DLSAP-address or DLCEP-address;

3) if a source DL-address is present in the just-received ED DLPDU, then it was also present and identical in that prior received ED DLPDU;

4) starting with the SD-parameters field, the first three octets of the just-received ED DLPDU, or the remainder of the DLPDU if fewer than three octets, are identical to the corresponding octets of that prior received ED DLPDU;

5) a period of link inactivity of (immediate-response-recovery-delay + 1) slot-times, $(V(IRRD) + 1) \times V(ST)$ octet-durations, has not occurred since receipt of that prior received ED DLPDU,

then the receiving DLE

i) shall discard the received DLPDU and not forward it to the DLE’s upper-level functions for further processing;

ii) shall retransmit the prior-transmitted immediate-reply DT DLPDU, unchanged from the prior transmission except for

– the final-token-use subfield of the frame control octet which conveys information to the listening LAS;

– those subfields of any present DLCEP SD-parameters which acknowledge receipt of, or request retransmission of, received DLSDUs.

NOTE This requirement is meant to ensure that the DL-priority, content and identity of any DLS-user data conveyed in the immediate reply DT DLPDU is identical to that in the immediately prior DT DLPDU sent from this same DL-address, including the originator’s transaction-id if one is present in the received and transmitted SD-parameters.

b) If a) does not apply, and the destination DL-address specified by the DLPDU designates a DLSAP-address of the receiving DLE, then the processing of the received DLPDU shall be based upon the DL(SAP)-role specified for that DLSAP-address.

NOTE 1 Group DL-addresses are not covered here; such ED DLPDUs are erroneous and are unrecognized upon receipt.

The receiving DLE shall initiate a reply within a period of maximum-response-delay slot-times, $V(MRD) \times V(ST)$ octet-durations, of receipt of the ED DLPDU:

1) If the DLE can prepare it in time, then the reply DLPDU shall be a DT DLPDU

– with any explicit destination DL-address in the DT DLPDU having the same value as the source DL-address of the received ED DLPDU, and length and format as follows:

- DT (see 6.7.1 format 2L) in response to ED (see 6.6.1 format 1L), or
- DT (see 6.7.1 format 2S) in response to ED (see 6.6.1 format 1S) if the DT DLPDU contains DLS-user data, or

- DT (see 6.7.1 format 4) in response to ED (see 6.6.1 format 1S) if the DT DLPDU does not contain DLS-user data,

and shall include SD-parameters specifying

- the originator's transaction-id as received in the stimulating ED DLPDU, as specified in 6.6.2.1a)4);
- reception error status for that ED-conveyed DLSDU if the received DLSDU could not be buffered or queued; otherwise status for the DLSDU requested by that ED DLPDU.

The received DLSDU shall be forwarded to the DLE's upper-level functions (see 8.3) for further processing.

If the received DLSDU was buffered or queued successfully, then the reply DLPDU may include a DLSDU which was already buffered at that responding DLE at the time of ED reception.

NOTE 2 This restriction prohibits the reply DLPDU from reflecting any higher-layer (than DLL) processing of the received DLSDU. This restriction is necessary to enable migration from prior national standards.

If the DL(SAP)-role of the destination DLSAP-address specified

i) BASIC or INITIATOR, then the DLE shall reject the received ED DLPDU based on this DL(SAP)-role, the received DLSDU shall be discarded, and an appropriate error status without an accompanying DLSDU shall be included in the reply DT DLPDU;

ii) UNCONSTRAINED RESPONDER, then

A) if no receiving buffer or queue is explicitly bound to the destination DLSAP-address at the DLL priority of the received DLSDU, or if a receiving queue is explicitly bound but is full, then the received DLSDU shall be discarded, and an appropriate error status without an accompanying DLSDU shall be included in the reply DT DLPDU; or

B) otherwise the received DLSDU shall be put into that buffer or appended to that not-full queue and the SD-parameter status shall indicate the highest DLL priority sending buffer with a non-null DLSDU which was available at the addressed DLSAP at the time of reception of the ED DLPDU, and

I) if that DLL priority is greater than or equal to the priority specified in 6.6.1.1b), then that DLSDU shall be included in the reply DT DLPDU and its buffer set to empty if so configured, and the DLL priority of that reply DT DLPDU shall be the DLL priority of the conveyed DLSDU;

II) if that DLL priority is less than the priority specified in 6.6.1.1b), then no DLSDU shall be included in the reply DT DLPDU, and the DLL priority of that reply DT DLPDU shall be the DLL priority specified in 6.6.1.1b); or

III) if there is no such sending buffer with a non-null DLSDU, then the reply DT DLPDU shall specify an appropriate error status, the DLL priority of that reply DT DLPDU shall be the DLL priority specified in 6.6.1.1b), and that reply DT DLPDU shall not contain DLS-user data.

iii) constrained responder, then

A) if the source DLSAP-address of the ED DLPDU is equal to the remote DLSAP address which was specified in the prior DL-BIND request primitive for the receiving DLSAP-address (or its DL-management equivalent), both after delocalization, then the procedure specified in ii) shall be followed; or

B) otherwise, the received DLSDU shall be discarded, and an appropriate error status without an accompanying DLSDU shall be included in the reply DT DLPDU.

2) If the DLE cannot prepare the required reply DT DLPDU in time, then the DLE shall send a DT DLPDU with any explicit destination DL-address in the DT DLPDU having the same value as the source DL-address of the received ED DLPDU, and length and format as follows:

– DT (see 6.7.1 format 2L) in response to ED (see 6.6.1 format 1L), or

– DT (see 6.7.1 format 4) in response to ED (see 6.6.1 format 1S),

and shall include SD-parameters

– with a transaction-id identical to the transaction-id from the received ED DLPDU, as specified in 6.6.2.1a)4);

– with a status (DR – "delayed reply") indicating that the DLE requires additional time to prepare the required response;

– with a null user data field,

and the DLE

– shall prepare that required DT DLPDU as soon as possible, as just described;

– shall include an explicit destination address in the reply DLPDU;

– shall append that reply DLPDU to the DLE's Q(US) to be transmitted at the first opportunity.

c) If a) does not apply, and the destination DL-address specified by the DLPDU designates an active DLCEP-address of a DLC for which the receiving DLE

– is a peer or publisher, and the DLL priority of the DLCEP is not equal to the priority specified in the received DLPDU; or

– is a peer, and the length and number of DL-addresses is not as expected or the DLPDU specifies an explicit source address which is not equal to the remote peer DLCEP's DLCEP-address,

then

1) the received DLPDU shall be forwarded to the DLE's upper-level functions (see 8.2.2.9) for further processing;

2) the receiving DLE shall initiate a reply within a period of maximum-response-delay slot-times, V(MRD) × V(ST) octet-durations, of receipt of the CD DLPDU. The reply DLPDU shall be a DT DLPDU in the format negotiated for the DLC for the selected direction of transmission, and shall contain SD-parameters appropriate to the sending DLCEP, but shall not contain DLS-user data.

d) If neither a) nor b) applies, and the destination DL-address specified by the DLPDU designates an active DLCEP-address of a DLC for which the receiving DLE is a peer or publisher, then

1) the received DLPDU shall be forwarded to the DLE's upper-level functions (see 8.2.2.9) for further processing;

2) the receiving DLE shall initiate a reply within a period of maximum-response-delay slot-times, V(MRD) × V(ST) octet-durations, of receipt of the ED DLPDU. The reply DLPDU shall be a DT DLPDU in the format negotiated for the DLC for the selected direction of transmission, shall contain SD-parameters appropriate to the sending DLCEP, and shall contain DLS-user data, if any

– was available and waiting for transmission or retransmission from the DLCEP;

– can be included in the DT DLPDU while still meeting the required maximum reply delay;

3) the requesting ED DLPDU may contain information about the state of the requesting DLCEP. The reply DT DLPDU is permitted, but is not required, to reflect that state information in its reply; immediate processing of that state information before sending the immediate reply shall be permitted but shall not be required.

The received DLPDU shall be forwarded to the DLE's upper-level functions (see 8.2.2.9) for further processing.

e) If a) does not apply, and the destination DL-address specified by the DLPDU designates an active DLCEP-address of a DLC for which the receiving DLE is a subscriber, then
   1) the non-DLS-user data portion of the received DLPDU shall be forwarded to the DLE's upper-level functions (see 8.2.2.9) for further processing;
   2) the receiving DLE shall record the destination DL-address from the received ED DLPDU in V(RA) for subsequent association with the expected immediate reply DT DLPDU, which should be the next DLPDU received;
   3) the receiving DLE shall monitor the local link for a reply and then act based on the result of that monitoring, all as specified in 5.2.8.3.

6.6.4.2 Additional actions required of a link-master class DLE

None.

6.6.4.3 Additional actions required of a bridge class DLE

The following considerations apply:

a) Since every bridge class DLE has link master capability, any actions specified in 6.6.4.2 also apply to a bridge class DLE.
b) If all of the following conditions hold
   A) the destination DL-address specified in the DLPDU is one which the bridge should forward but to which the bridge DLE itself would not otherwise generate an immediate reply DLPDU;
   B) the immediately prior DLPDU was transmitted as an immediate-response to a received ED DLPDU with the same destination DL-address;
   C) if a source DL-address is present in the just-received ED DLPDU, then it was also present and identical in that prior received ED DLPDU;
   D) the basic-DLC-parameters portion of the SD-parameters of the just-received ED DLPDU is not null, and is identical to that of that prior received ED DLPDU;
   E) a period of link inactivity of immediate-response-recovery-delay plus one slot-time, $(V(IRRD)+1) \times V(ST)$ octet-durations, has not occurred since receipt of that prior received ED DLPDU,

   then the bridge

   1) shall discard the received DLPDU and not forward it to the bridge's other functions for further processing;
   2) shall initiate retransmission of the prior-transmitted immediate-reply SR DLPDU, unchanged from the prior transmission.

   NOTE This requirement is meant to ensure that any status conveyed in the immediate reply SR DLPDU is identical to that in the immediately prior SR DLPDU.

   Otherwise,

   i) if the destination DL-address specified in the DLPDU is one which the bridge should forward but which the bridge DLE itself would not otherwise receive, then the bridge shall form and send an SR DLPDU

– within a period of maximum-response-delay slot-times, V(MRD) × V(ST) octet-durations, of receipt of the ED DLPDU;

– with status indicating whether or not the bridge was able to buffer the received DLPDU.

ii) if the destination DL-address specified in the DLPDU is one which the bridge should forward, and the bridge was able to receive and buffer the DLPDU without error, then the received DLPDU shall be forwarded with modification of the frame-control field in the forwarded DLPDU as appropriate (see 5.2.5.3).

iii) if the DLPDU contains an explicit source DL-address, then the bridge shall attempt to update its routing table entry for the source DL-address specified in the DLPDU to reflect the bridge port from which the DLPDU was received.

c) It is a protocol error for a bridge DLE which will forward the received ED DLPDU not to send an SR reply DLPDU when a reply is required.

NOTE At most one bridge DLE on the local link should be forwarding the received ED DLPDU.

6.6.4.4 Additional actions required of the current LAS DLE

The LAS DLE shall act as specified in 5.2.8.2.

6.7 Data (DT) DLPDU

A DATA (DT) DLPDU is used to transfer a limited amount of transparent user data from one DLS-user to one or more other DLS-users; to acknowledge the transfer of such data; and to assist in the synchronization of DLCEPs and of DLS-users.

It is also used by a responding DLE when replying to a received CA, CD or ED DLPDU, when the time permitted by V(ST) and V(MRD) (see 4.7.1.1 and 4.7.1.2) is inadequate for that DLE to generate the required response to the received DLPDU.

It is also used by a DLE to send an SPDU to one or more other DLEs.

6.7.1 Structure of the DT DLPDUs

Table 18 – Structure of DT DLPDUs

format	Frame control	Destination address	Source address	Parameters	User data
1L	1101 1FPP	HL.N.S	HL.N.S	SD-p	o-pDLSDU
1S	1101 0FPP	N.S	N.S	SD-p	o-pDLSDU
2L	1001 1FPP	HL.N.S	—	SD-p	o-pDLSDU
2S	1001 0FPP	N.S	—	SD-p	o-pDLSDU
3L	0101 1FPP		HL.N.S	SD-p	o-pDLSDU
3S	0101 0FPP		N.S	SD-p	o-pDLSDU
4	1001 0F00	[PSA]	—	SD-p	o-pDLSDU
5	0101 0F00		[PDA]	SD-p	o-pDLSDU

where

[PDA] is the destination DL-address from the immediately prior CA, CD or ED DLPDU,

[PSA] is the implied source DL-address from the immediately prior CA, CD or ED DLPDU.

6.7.1.1 Frame control field

The frame control field shall specify

a) the DLPDU's function;

b) the DLPDU's priority;

c) the length, number and type of DLPDU addresses;

d) whether or not transmission of the current DLPDU terminates use of a delegated token.

6.7.1.2 Address field

The address field shall consist of either

a) an explicit destination DL-address and an explicit source DL-address, in that order, for formats 1L and 1S,

b) only an explicit destination DL-address, for formats 2L and 2S,

c) only an explicit source DL-address, for formats 3L and 3S,

d) only an implied destination DL-address, for format 4, or

e) only an implied source DL-address, for format 5.

For formats 1L, 2L and 3L, all addresses shall be LONG; for formats 1S, 2S and 3S, all addresses shall be SHORT; for formats 4 and 5 all addresses shall be VERY-SHORT (see 5.2.1.1) (that is, implicit).

6.7.1.3 Parameters field

When the destination DL-address explicitly specifies a group DL-address, the status-data-parameters (SD-parameters) field is null (formats 1L and 1S), as described in 7.4.1.

When the destination DL-address explicitly or implicitly specifies a DLSAP-address, the SD-parameters field specifies a transaction-id and status (formats 2L, 2S, 4), or is null (formats 1L and 1S), as described in 7.4.2.

When the first DL-address explicitly or implicitly specifies a DLCEP-address, the SD-parameters field specifies state information for the DLCEP addressed by the destination DL-address (formats 1L, 1S, 2L, 2S, 4), or source DL-address (formats 3L, 3S, 5), as described in 7.4. The size and structure of this field is dependent on the QoS attributes associated with the addressed DLCEP, and is determined during DLCEP establishment.

6.7.1.4 User data field

The user data field shall consist of a single or partial optional DLSDU whose maximum size is limited to the smaller of

a) the maximum DLS-user data size permitted for a DLPDU of the priority specified in 6.7.1.1b);

b) when the DLPDU's explicit or implied destination (or source) address is a DLCEP-address, the maximum DLSDU size negotiated on the DLC for this direction of data transmission to (or from) that DLCEP.

6.7.2 Content of the DT DLPDU

The frame control field shall be encoded as specified in Table 18.

Either the DL-addresses shall be

a) the first a group DL-address and the second a DLSAP-address,

b) all DLSAP-addresses,

c) all DLCEP-addresses, or

d) the first a DLCEP-address and the second a DLSAP-address.

6.7.2.1 Content of the DT DLPDU when specifying a destination DL(SAP)-address

When the first address is a group DL-address as in 6.7.2a), then

a) If the DLPDU format is format 1L or 1S, then
   1) the DLPDU is being used to implement the unitdata transfer service;
   2) the DL(SAP)-role for the destination DLSAP-address shall be GROUP;
   3) the DL(SAP)-role for the source DLSAP-address shall be BASIC;
   4) the SD-parameters field shall be null;
   5) the user data shall be a single DLSDU whose size is limited to the maximum size for the priority specified in 6.7.1.1b), and shall not be null.
b) No other DLPDU format may be used.

When the addresses are DLSAP-addresses as in 6.7.2b), then

c) If the DLPDU format is format 1L or 1S, then
   1) this DLPDU is being used to implement the unitdata transfer service;
   2) the DL(SAP)-role for the destination DLSAP-address shall be BASIC;
   3) the DL(SAP)-role for the source DLSAP-address shall be BASIC;
   4) the SD-parameters field shall be null;
   5) the user data shall be a single DLSDU whose size is limited to the maximum size for the priority specified in 6.7.1.1b), and shall not be null.
d) If the DLPDU format is format 2L, 2S or 4, then
   1) this DLPDU conveys an acknowledgment or reply as part of the unitdata transfer (with remote-DLE-confirmation) service, or the unitdata exchange service;
   2) the DL(SAP)-role for the destination DLSAP-address shall be BASIC or INITIATOR;
   3) the DL(SAP)-role for the implicit source DLSAP-address is BASIC or CONSTRAINED RESPONDER or UNCONSTRAINED RESPONDER;
   4) the SD-parameters field shall specify a transaction-id and reply status for the addressed DLSAP, and the contents of this field shall be as described in 7.4;
   5) the user data either shall be a single DLSDU whose size is limited to the maximum size for the priority specified in 6.7.1.1b), or shall be null, as follows:
      – formats 2L may be null or non-null;
      – format 2S shall always be non-null;
      – format 4 shall always be null.
e) No other DLPDU format may be used.

6.7.2.2 Content of the DT DLPDU when specifying a destination or source DLCEP-address

When the first address is a DLCEP-address, as in 6.7.2c) and 6.7.2d), then

a) this DLPDU can convey a single or partial DLSDU
   – from one peer DLCEP to its corresponding peer DLCEP,
   – from a subscriber DLCEP to its corresponding publisher DLCEP, or
   – from a publisher DLCEP to its corresponding subscriber DLCEPs

and

b) the second address, if present,
   – shall be the peer DLCEP-address of the same DLC as the destination DLCEP-address, or

   – shall be a subscriber DLCEP's calling-DLSAP-address;

c) the SD-parameters field shall specify state information for the addressed DLCEP, and the contents of this field shall be as described in 7.4.2;

   NOTE The size and structure of this field is dependent on the QoS attributes associated with the DLCEP addressed by the destination DL-address specified in this DLPDU, and is determined during DLCEP establishment.

d) the user data shall specify those octets of a DLSDU consistent with the negotiated DLSDU size and the segmentation information specified in the accompanying SD-parameters, and may be null.

NOTE 1 Formats 1L, 2L, 1S and 2S are used for peer-to-peer and subscriber-to-publisher communications; 1L is used when the DLPDU-authentication attribute is SOURCE or MAXIMAL, 1S is used when the DLPDU-authentication attribute is SOURCE, 2L and 2S are used when the DLPDU-authentication attribute is ORDINARY.

Format 3L is used for publisher-to-subscriber communications when the DLPDU-authentication attribute is MAXIMAL. Formats 3L and 3S are used for publisher-to-subscriber communications when the DLPDU-authentication attribute is ORDINARY or SOURCE.

The specific format to be used (of formats 1L to 3S) is determined as part of DLCEP establishment.

NOTE 2 Formats 4 and 5 can be used instead of formats 2S and 3S, respectively, only when the sending DLE holds a reply token and when the DLPDU-authentication attribute is ORDINARY.

6.7.3 Sending the DT DLPDU

A DT DLPDU may be selected for transmission on the link when the sending DLE

a) has just received a reply token in a CA, CD or ED DLPDU, permitting a single transmission of a DT or SR DLPDU; or

b) holds a scheduler token or delegated token which is the dominant token on the local link, and when the remaining allocated duration of token usage, C(RD), permits completion of the DT DLPDU's transmission prior to expiration of the token.

Each explicit DL-address in the DLPDU shall be delocalized (see 5.2.2) before transmission.

6.7.3.1 Transmission when the reply token is dominant

The following considerations apply:

a) A DT DLPDU may be sent on the link when the sending DLE has received a CD or ED DLPDU addressed

   – to one of its active DLSAP-addresses, or

   – to one of its active DLCEP-addresses for which it has a peer or publisher DLCEP,

   and the sending DLE is replying as specified by 6.5.4.1 or 6.6.4.1, by forming as an immediate reply a DT DLPDU which may include a DLSDU which was already buffered or queued at that responding DLE at the time of the CD or ED DLPDU's reception.

   NOTE This restriction prohibits the reply DLPDU from reflecting any higher-layer processing of the received DLSDU. This restriction is necessary to enable migration from prior national standards.

b) A DT DLPDU may be sent on the link when the sending DLE has received a CA DLPDU addressed

   – to one of its active DLSAP-addresses, or

   – to one of its active DLCEP-addresses for which it has a peer or publisher DLCEP,

   and the sending DLE is replying as specified by 6.4.4.1, by forming as an immediate reply a DT DLPDU which does not include a DLSDU.

When an immediate reply to a CA, CD or ED DLPDU is required, as specified in a) or b), then the replying DLE shall send a reply DT DLPDU within a period of maximum-response-delay slot-times, V(MRD) × V(ST) octet-durations, of receipt of the requesting CA, CD or ED DLPDU.

If the CA, CD or ED DLPDU which required the reply was addressed to a DLCEP, then that CA, CD or ED DLPDU may have contained SD-parameters which conveyed information about the state of the sending DLCEP to the receiving DLCEP. The receiving DLE is permitted, but is not required, to reflect that state information in its reply DT DLPDU; immediate processing of that state information before sending the immediate reply shall be permitted but shall not be required.

The final-token-use subfield of the reply DT DLPDU shall have the same value as that in the requesting CA, CD or ED DLPDU.

Each explicit DL-address in the reply DT DLPDU shall be delocalized (see 5.2.2) before transmission.

It is a protocol error for the addressed DLE not to send a DT reply DLPDU when a reply is required.

NOTE At most one DLE on the local link should be sending a reply to the received CA, CD or ED DLPDU. That reply may be either a DT or SR DLPDU.

6.7.3.2 Transmission when the delegated token is dominant

A DT DLPDU may be sent on the link when the sending DLE holds a delegated token which is the dominant token on the local link, and when the remaining allocated duration of token usage, C(RD), permits completion of the DLPDU's transmission prior to expiration of the token.

If no additional use of that token after sending this DLPDU is needed at that time, then the DLE may set the final-token-use subfield of the DT DLPDU to the value FINAL; else that subfield shall have the value NOT-FINAL.

Each explicit DL-address in the DLPDU shall be delocalized (see 5.2.2) before transmission.

6.7.3.3 Transmission when the scheduler token is dominant

The LAS DLE may send a DT DLPDU on the link when both the remaining allocated duration of token usage before the next scheduled activity, and the remaining allocated duration of scheduler token usage for link maintenance, which is V(LTHT) octet-durations minus the amount of link capacity used for link maintenance during the current cycle of "circulating the token", permits completion of the DLPDU's transmission prior to expiration of the token.

Each explicit DL-address in the DLPDU shall be delocalized (see 5.2.2) before transmission.

6.7.4 Receiving the DT DLPDU

Each DL-address in the DLPDU shall be delocalized (see 5.2.2) upon reception.

A received DT DLPDU shall be treated as follows by the receiving DLE.

6.7.4.1 Actions required of all DLEs

6.7.4.1.1 Actions required when the reply token was not dominant at start-of-reception

a) If the received DT DLPDU has format 1L or 1S, and its destination DL-address designates a DL(SAP)-address of the receiving DLE, then the received DLPDU shall be forwarded to the DLE's upper-level functions (see 8.3.1.3) for further processing.

b) If the received DT DLPDU has format 2L or 2S, and its destination DL-address designates a DLSAP-address of the receiving DLE, and the transaction-id reported by the SD-parameters indexes an assigned transaction in the outstanding-transaction-array, V(OTA) (see 4.7.1.15), then

   – the DLE shall consider the prior transmission to have been error-free;

   – the information in the indexed entry in V(OTA) shall be associated with the received DT DLPDU;

   – the received DT DLPDU and that associated information shall be forwarded to the DLE's upper-level functions (see 8.3) for further processing;

   – the indexed entry in V(OTA) shall be unassigned.

c) If the received DT DLPDU

   1) has format 1L, 1S, 2L or 2S, and its destination DL-address designates a DLCEP-address designating a peer or publisher DLCEP of the receiving DLE; or

   2) has format 3L or 3S, and its source DL-address designates a DLCEP-address designating a subscriber DLCEP of the receiving DLE,

   then the received DLPDU shall be forwarded to the DLE's upper-level functions (see 8.2) for further processing.

d) If the received DT DLPDU has format 1L or 1S, and its destination DL-address designates NODE.0 DL-address, V(TN).0, of the receiving DLE's DL-support functions, then the received DLPDU shall be forwarded to the DLE's upper-level functions (see 10.3) for further processing.

e) If none of a), b), c) or d) applies, then the DT DLPDU shall be reported to local DL-management as an unexpected response, and shall be discarded.

   NOTE This report may take the form of incrementing a DL-management error counter.

6.7.4.1.2 Actions required when the reply token was dominant at start-of-reception and the receiving DLE sent the CA, CD or ED DLPDU which created the reply token

a) If the received DT DLPDU

   – has format 2L or 2S, and its destination DL-address designates a DLSAP-address of the receiving DLE; or

   – has format 4, and the source DL-address from the immediately prior CA, CD or ED DLPDU was a DLSAP-address,

   then

   1) If the status reported by the SD-parameters is "DR – delayed reply" (that is, has a value of F_{16}), then the DLE

      – shall consider the prior transmission to have been error-free;

      – shall discard the DLPDU.

      NOTE The DLE should expect to receive another DT DLPDU at a later time, with the same destination DL-address and format, as a delayed reply.

   2) If 1) does not apply, then

      – the DLE shall consider the prior transmission to have been error-free;

      – the information in the V(LTI)'th entry in V(OTA) (see 4.7.1.16 and 4.7.1.15) shall be associated with the received DT DLPDU;

      – the received DT DLPDU and that associated information shall be forwarded to the DLE's upper-level functions (see 8.3) for further processing;

      – the V(LTI)'th entry in V(OTA) shall be deassigned.

b) If the received DT DLPDU has format 1L, 1S, 2L or 2S, and its destination DL-address designates a DLCEP-address of a peer DLCEP of the receiving DLE, then

– the DLE shall consider the prior transmission to have been error-free;

– the received DT DLPDU shall be forwarded to the DLE's upper-level functions (see 8.2) for further processing.

c) If the received DT DLPDU has format 4, and the explicit or implicit source DL-address from the immediately prior CA, CD or ED DLPDU was a DLCEP-address of a peer DLCEP of the receiving DLE, then

– the DLE shall consider the prior transmission to have been error-free;

– the received DT DLPDU shall be forwarded to the DLE's upper-level functions (see 8.2) for further processing, with its implied destination DLCEP-address assumed to be the explicit or implicit source DL-address from that immediately prior CA, CD or ED DLPDU.

d) If the received DT DLPDU has format 3L or 3S, and its source DL-address designates the publisher's DLCEP-address of a subscriber DLCEP of the receiving DLE, and this source DL-address is equal to the destination DL-address from the immediately prior CA, CD or ED DLPDU, then

– the DLE shall consider the prior transmission to have been error-free;

– the received DT DLPDU shall be forwarded to the DLE's upper-level functions (see 8.2) for further processing.

e) If the received DT DLPDU has format 5, and the explicit destination DL-address from the immediately prior CA, CD or ED DLPDU was a DLCEP-address of a subscriber DLCEP of the receiving DLE, then

– the DLE shall consider the prior transmission to have been error-free;

– the received DT DLPDU shall be forwarded to the DLE's upper-level functions (see 8.2) for further processing, with its implied source DLCEP-address assumed to be the explicit destination DL-address from that immediately prior CA, CD or ED DLPDU.

f) If none of a) to e) applies, then the DT DLPDU shall be reported to local DL-management as an unexpected response, and shall be discarded.

NOTE This report may take the form of incrementing a DL-management error counter.

6.7.4.1.3 Actions required when the reply token was dominant at start-of-reception and the receiving DLE did not send the CA, CD or ED DLPDU which created the reply token

a) If the received DT DLPDU has format 3L or 3S, and its source DL-address designates a publisher's DLCEP-address of a subscriber DLCEP of the receiving DLE, and this source DL-address is equal to the destination DL-address, V(RA), from the immediately prior CA, CD or ED DLPDU, then

– the DLE shall consider the prior transmission to have been error-free;

– the received DT DLPDU shall be forwarded to the DLE's upper-level functions (see 8.2) for further processing.

b) If the received DT DLPDU has format 5, and the explicit destination DL-address from the immediately prior CA, CD or ED DLPDU was the publisher's DLCEP-address of a subscriber DLCEP of the receiving DLE, then

– the DLE shall consider the prior transmission to have been error-free;

– the received DT DLPDU shall be forwarded to the DLE's upper-level functions (see 8.2) for further processing, with its implied source DLCEP-address assumed to be the explicit destination DL-address from that immediately prior CA, CD or ED DLPDU.

6.7.4.2 Additional actions required of a link-master class DLE

None.

6.7.4.3 Additional actions required of a bridge class DLE

a) Since every bridge class DLE has link master capability, any actions specified in 6.7.4.2 also apply to a bridge class DLE.

b) If the first DL-address specified in the DLPDU is an explicit DL-address to which the bridge should forward the DLSDU and the bridge was able to buffer the DLPDU without error, then the DLPDU shall be forwarded with modification of the frame-control field in the forwarded DLPDU as appropriate (see 5.2.5.3).

c) If the DLPDU contains an explicit source DL-address, then the bridge shall attempt to update its routing table entry for the source DL-address specified in the DLPDU to reflect the bridge port from which the DLPDU was received.

d) Otherwise, if neither b) nor c) applies, then the DLE shall not forward the DLPDU.

6.7.4.3.1 Actions required when the reply token was dominant at start-of-reception and the receiving bridge DLE forwarded, but did not originate, the CA, CD or ED DLPDU which created the reply token

a) The DLE shall consider the prior transmission to have been error-free.

b) If

   – the received DT DLPDU has format 4;

   – the destination DL-address from the immediately prior CA, CD or ED DLPDU was a DLSAP-address;

   – the status reported by the SD-parameters is other than “DR – delayed reply” (that is, has a value other than F_{16});

   – the user data field is null,

   then the DLE shall form a DT DLPDU

   1) with format 2L when the previous CA, CD or ED DLPDU was format 1L, or format 2S when the previous CA, CD or ED DLPDU was format 1S;

   2) with the explicit destination DL-address equal to the source DL-address in that immediately prior CA, CD or ED DLPDU;

   3) with an SD-parameters field equal to the SD-parameters field of the received DT DLPDU;

   4) with a null user data field;

   and shall forward the just-formed DT DLPDU as if it had just been originated by the upper-level functions of the bridge DLE.

   NOTE This means that the bridge DLE originates the FCS for the newly-formed DT DLPDU as in 5.2.5.1, rather than just modifying a previously received FCS during normal forwarding as in 5.2.5.3.

6.7.4.4 Additional actions required of the current LAS DLE

If the final-token-use subfield of the received DLPDU has the value FINAL, then the LAS DLE shall

a) assume that the current use of the delegated token is terminated as if the token had been returned by an RT (see 6.17) DLPDU;

b) assume that the scheduler token is again dominant on the local link and resume active operation as the LAS.

6.8 Status response (SR) DLPDU

A STATUS RESPONSE (SR) DLPDU is sent only while holding a reply token; it is used

a) to indicate the receipt of the immediately prior CA, CD or ED DLPDU by the bridge which would normally forward that DLPDU toward the addressed DLE, to indicate to the sending DLE that no error occurred, or that the indicated error occurred;

b) to reject an attempted transfer of the LAS role from the current LAS DLE to another link-master DLE.

6.8.1 Structure of the SR DLPDU

Table 19 – Structure of SR DLPDUs

Frame control	Destination address	Source NODE-address	Parameters
0001 0F11	[PSA]	N	o-SR-p

where

[PSA] is the implied DL-address equal to the implied or explicit source DL-address of the immediately prior DLPDU on the link

6.8.1.1 Frame control field

The frame control field shall specify

a) the DLPDU's function;
b) the DLPDU's implicit priority, which is that of the immediately prior DLPDU on the local link (to which the SR DLPDU is an immediate response);
c) the length, number and type of DLPDU addresses;
d) whether or not transmission of the current DLPDU terminates use of a delegated token.

6.8.1.2 Address field

The address field shall consist of an explicit source NODE DL-address.

6.8.1.3 Parameters field

The delayed-reply-parameters (SR-parameters) field specifies an optional transaction id and status. When non-null, this field shall be structured and encoded as described in 7.5.

6.8.1.4 User data field

The user data field shall be null.

6.8.2 Content of the SR DLPDU

The frame control field shall be encoded as specified in Table 19:

– When sent as an immediate reply to a TL DLPDU, the final-token-use subfield of the SR DLPDU shall specify FINAL.
– When sent as an immediate reply to a CA, CD or ED DLPDU, the final-token-use subfield of the SR DLPDU shall have the same value as that in the requesting CA, CD or ED DLPDU.

The address field shall be the replying DLE's NODE DL-address, whose value is the sender's node-id, V(TN).

NOTE The primary purposes for the inclusion of the NODE DL-address is to identify the DLE which has assumed the reply token to any observing link analyzer.

When sent as an immediate reply to a CA, CD or ED DLPDU,

a) the SR-parameters shall be null when the DLE is a bridge DLE which has been able to accept the transaction-initiating DLPDU for forwarding;
b) otherwise, when a) does not apply, the SR-parameters shall be encoded as described in 7.5.

When sent as an immediate reply to a TL DLPDU, the SR-parameters shall be encoded as described in 7.5.

6.8.3 Sending the SR DLPDU

a) An SR DLPDU may be sent on the link when the sending DLE is a bridge DLE which has just received a reply token in a CA, CD or ED DLPDU, permitting that DLE to transmit a single DT or SR DLPDU, and the DLE is replying as specified by 6.4.4.3, 6.5.4.3 or 6.6.4.3.

   It is a protocol error for a bridge DLE which will forward the received CA, CD or ED DLPDU to not send an SR reply DLPDU when a reply is required.

   NOTE At most one bridge DLE on the local link should be forwarding the received CA, CD or ED DLPDU.

b) An SR DLPDU may be sent on the link when the sending DLE is a link master or bridge DLE which has just received a reply token in a TL DLPDU, and the receiving DLE needs to reject the transfer of the LAS role (see 6.20.4.2).

The final-token-use subfield of the reply SR DLPDU shall have the same value as that in the requesting CA, CD or ED DLPDU, or shall have the value FINAL for a reply to a requesting TL DLPDU.

When an immediate reply to a CA, CD, ED or TL DLPDU is required, then the DLE shall reply within a period of maximum-response-delay slot-times, V(MRD) × V(ST) octet-durations, of receipt of the requesting CA, CD, ED or TL DLPDU.

6.8.4 Receiving the SR DLPDU

A received SR DLPDU shall be treated as follows by the receiving DLE.

6.8.4.1 Actions required of all DLEs

The following considerations apply:

a) A received SR DLPDU, received as a reply to an immediately prior CA, CD or ED DLPDU which was originated by the receiving DLE and which was addressed to a DLSAP-address, shall cause the receiving DLE
   1) to consider the prior transmission to have been error-free;
   2) to forward that received DLPDU to the DLE's upper-level functions (see 8.3 for further processing.
b) A received SR DLPDU, received as a reply to an immediately prior TL DLPDU which was originated by the receiving (LAS) DLE shall cause the receiving DLE
   1) to consider the prior transmission to have been error-free;
   2) as specified in 6.20.3
      – to re-assume the scheduler token;
      – to inform local DL-management of the event;
      – to resume active operation as the LAS, and commence transmission on the link.

6.8.4.2 Additional actions required of a link-master class DLE

None.

6.8.4.3 Additional actions required of a bridge class DLE

The following considerations apply:

a) Since every bridge class DLE has link master capability, any actions specified in 6.8.4.2 also apply to a bridge class DLE.

b) A received SR DLPDU
   1) which reports a status other than "BOK – bridge OK" (that is, has a value other than E_{16});
   2) which was received as a reply to an immediately prior CA, CD or ED DLPDU which was forwarded (and therefore not originated) by the receiving DLE and which was addressed to a DLSAP-address

   shall cause the receiving DLE to form a DT DLPDU

   i) with format 2L when the previous CA, CD or ED DLPDU was format 1L, or format 2S when the previous CA, CD or ED DLPDU was format 1S;
   ii) with the explicit destination DL-address equal to the source DL-address in that immediately prior CA, CD or ED DLPDU;
   iii) with an SD-parameters field equal to the SR-parameters field of the received SR DLPDU;
   iv) with a null user data field;

   and shall forward the just-formed DT DLPDU as if it had just been originated by the upper-level functions of the bridge DLE.

   NOTE This means that the bridge DLE originates the FCS for the newly-formed DT DLPDU as in 5.2.5.1, rather than just modifying a previously received FCS during normal forwarding as in 5.2.5.3.

c) Otherwise, if b) does not apply, then the DLE shall not forward the DLPDU.

6.8.4.4 Additional actions required of the current LAS DLE

The following considerations apply:

a) If the final-token-use subfield of the received DLPDU has the value FINAL, then the LAS DLE shall
   1) assume that the current use of the delegated token is terminated as if the token had been returned by an RT (see 6.17) DLPDU;
   2) assume that the scheduler token is again dominant on the local link and resume active operation as the LAS.
b) A received SR DLPDU, received as a reply to an immediately prior TL DLPDU which was originated by the LAS DLE, shall cause the LAS DLE
   1) to consider the prior transmission to have been error-free;
   2) to terminate the attempted transfer of the LAS role;
   3) to inform DL-management of the event.

6.9 Compel time (CT) DLPDU

A COMPEL TIME (CT) DLPDU is used by the current holder of a delegated token to request the LAS DLE to transmit as an immediate response a TIME DISTRIBUTION (TD) DLPDU, thus enabling all other DLEs on the local link

– to update their sense of DL-time;
– to synchronize the rates of advance of their senses of time.

6.9.1 Structure of the CT DLPDU

Table 20 – Structure of CT DLPDUs

Frame control
0001 0F00

6.9.1.1 Frame control field

The frame control field shall specify

a) the DLPDU's function;
b) the DLPDU's implicit priority, which is URGENT;
c) the length, number and type of DLPDU addresses;
d) whether or not transmission of the current DLPDU, and either its expected immediate reply or the appropriate link-idle timeout, terminates use of a delegated token.

6.9.1.2 Address field

The address field shall be null.

6.9.1.3 Parameters field

The parameters field shall be null.

6.9.1.4 User data field

The user data field shall be null.

6.9.2 Content of the CT DLPDU

The frame control field shall be encoded as specified in Table 20.

6.9.3 Sending the CT DLPDU

A DLE's need for the current DL-time, or to synchronize with the LAS DLE and that DLE's rate of advance of the current DL-time, may result in transmission of a CT DLPDU. A CT DLPDU requests that the LAS DLE send a TD DLPDU in immediate reply.

A CT DLPDU may be sent on the link when the sending DLE holds a delegated token whose remaining allocated duration of token usage, C(RD), permits completion of the implied transaction – sending the CT DLPDU, which requires an immediate reply, and awaiting the TD reply DLPDU – prior to return of the token.

If the DLE holds a delegated token, and no additional use of that token after sending this DLPDU and awaiting its immediate reply is needed at that time, then the DLE may set the final-token-use subfield of the CT DLPDU to the value FINAL; else that subfield shall have the value NOT-FINAL.

After sending a CT DLPDU, the sending DLE shall monitor the local link for a reply as specified in 5.2.8.1. The permissible reply DLPDU is a TD DLPDU.

6.9.4 Receiving the CT DLPDU

A received CT DLPDU shall be treated as follows by the receiving DLE.

6.9.4.1 Actions required of all DLEs

None.

6.9.4.2 Additional actions required of a link-master class DLE

None.

6.9.4.3 Additional actions required of a bridge class DLE

a) Since every bridge class DLE has link master capability, any actions specified in 6.9.4.2 also apply to a bridge class DLE.
b) The DLE shall not forward the DLPDU.

6.9.4.4 Additional actions required of the current LAS DLE

The DLE shall reply immediately with a TD DLPDU.

If the final-token-use subfield of the received CT DLPDU has the value FINAL, then the LAS DLE shall

a) assume that the current use of the delegated token has terminated and that the scheduler token is again dominant on the local link;
b) treat that termination as if the token had been returned by an RT (see 6.17) DLPDU;
c) resume active operation as the LAS.

6.10 Time distribution (TD) DLPDU

A TIME DISTRIBUTION (TD) DLPDU is transmitted by the LAS DLE to enable the DLEs on the local link to coordinate and to synchronize the rates of advance of their senses of DL-time.

6.10.1 Structure of the TD DLPDU

Table 21 – Structure of TD DLPDUs

Frame control	Source NODE-address	Parameters
0001 0F01	N	TD-p

6.10.1.1 Frame control field

The frame control field shall specify

a) the DLPDU's function;
b) the DLPDU's implicit priority, which is URGENT;
c) the length, number and type of DLPDU addresses;
d) whether or not transmission of the current DLPDU terminates use of a delegated token.

6.10.1.2 Address field

The address field shall consist of an explicit source NODE DL-address.

6.10.1.3 Parameters field

The parameter field shall be structured and encoded as specified in 7.6.

6.10.1.4 User data field

The user data field shall be null.

6.10.2 Content of the TD DLPDU

The frame control field shall be encoded as specified in Table 21.

The address field shall specify the node designator of the LM DLE which is co-resident with the LAS DLE.

The TD-parameters field shall be encoded as specified in 7.6, and shall reflect the current values of the DLE's time-related variables and node-time counter at the moment of DLPDU formation, which shall be less than 1 s before DLPDU transmission.

The user data field shall be null.

6.10.3 Sending the TD DLPDU

A TD DLPDU may be sent on the link when the sending DLE (the LAS)

a) has just received a reply token in a CT DLPDU, permitting a single transmission of a TD DLPDU; or

b) holds a scheduler token which is the dominant token on the local link, and is required to send a TD DLPDU as specified in 8.4.1.6, and the remaining allocated duration of token usage before the next scheduled activity permits completion of the DLPDU's transmission prior to expiration of the token.

If the TD DLPDU is sent as a reply to a received CT DLPDU, then the final-token-use subfield of the reply TD DLPDU shall have the same value as that in the requesting CT DLPDU.

If the time-synchronization class of the sending DLE is not NONE, then after transmitting the TD DLPDU, the LAS DLE shall schedule another transmission of a TD DLPDU as specified in 8.4.1.2.

Reception of a CT DLPDU by the LAS DLE shall cause the LAS DLE to reply immediately with a TD DLPDU, sent to all other DLEs on the local link.

6.10.4 Receiving the TD DLPDU

A received TD DLPDU shall be treated as follows by the receiving DLE.

6.10.4.1 Actions required of all DLEs

The following considerations apply:

a) Each DLE that supports the variables, timers and counters defined in 4.7.1.19 to 4.7.1.26, other than the sending DLE, shall append to the received DLPDU the low-order 24 bits of the local node-time C(NT), known thereafter as N_R(NT), at which the TD DLPDU reception completed (that is, receipt of the END-OF-DATA or END-OF-DATA-AND-ACTIVITY PhIDU).

b) Each DLE other than the sending DLE shall process the received DLPDU as specified in 8.4.1.3.

6.10.4.2 Additional actions required of a link-master class DLE

None.

6.10.4.3 Additional actions required of a bridge class DLE

The following considerations apply:

a) Since every bridge class DLE has link master capability, any actions specified in 6.10.4.2 also apply to a bridge class DLE.

b) The DLE shall not forward the DLPDU. However, if the TD DLPDU was received at the port which is the bridge DLE's root port (that is, the port toward the root of the bridge spanning tree), then the DLE shall forward the DLPDU reception event to the bridge operation level (see 4.1.2).

6.10.4.4 Additional actions required of the current LAS DLE

If the DLE did not just send the DLPDU, then the DLE shall drop the scheduler token and deactivate its role as LAS; and shall inform local DL-management of the event.

If the final-token-use subfield of the received DLPDU has the value FINAL, then the LAS DLE shall

a) assume that the current use of the delegated token is terminated as if the token had been returned by an RT (see 6.17) DLPDU;
b) assume that the scheduler token is again dominant on the local link and resume active operation as the LAS.

6.11 Round-trip-delay query (RQ) DLPDU

A ROUND-TRIP-DELAY QUERY (RQ) DLPDU is sent from one DLE to another on the local link to initiate the measurement and computation of the round-trip delay intrinsic to their inter-communication. Its receipt results in the return of a complementary ROUND-TRIP-DELAY REPLY (RR) DLPDU completing the measurement.

6.11.1 Structure of the RQ DLPDU

Table 22 – Structure of RQ DLPDUs

Frame control	Destination address	Source address	Parameters
1100 0F00	N.0	N.0	RQ-p

6.11.1.1 Frame control field

The frame control field shall specify

a) the DLPDU's function;
b) the DLPDU's implicit priority, which is NORMAL;
c) the length, number and type of DLPDU addresses;
d) whether or not transmission of the current DLPDU, and either its expected immediate reply DLPDU or the appropriate link-idle timeout, terminates use of a delegated token.

6.11.1.2 Address field

The address field shall consist of two explicit SHORT DL-addresses, destination and source, in that order.

6.11.1.3 Parameters field

The parameters field shall be structured and encoded as specified in 7.7.

6.11.1.4 User data field

The user data field shall be null.

6.11.2 Content of the RQ DLPDU

The frame control field shall be encoded as specified in Table 22.

The address field shall specify two NODE DL-addresses expressed in SHORT DL-address format, where the destination DL-address is

– the SHORT flat DL-address for the local LAS, 0400_{16} (see 4.3.3.2); or

– has the value formed by concatenating the intended receiver's node-id, V_R(TN), and an octet of zero, thus specifying that DLE's DL-support functions;

and the source DL-address has the value formed by concatenating the sender's node-id, V_S(TN), and an octet of zero, thus specifying the sending DLE's DL-support functions.

The RQ-parameters field shall be encoded as specified in 7.7, and shall reflect the value of the DLE's node-time counter, if maintained by the DLE, at a moment of DLPDU formation which is a constant time offset from the moment when the END-OF-DATA-AND-ACTIVITY PhIDU will be transmitted for the DLPDU. Otherwise, the DLE shall use any value for this field.

The user data field shall be null.

6.11.3 Sending the RQ DLPDU

An RQ DLPDU may be sent on the link when the sending DLE

a) holds a delegated token which is the dominant token on the local link, and when the remaining allocated duration of token usage, C(RD), permits completion of the implied transaction – sending the RQ DLPDU, which requires an immediate reply, and awaiting the RR reply DLPDU – prior to expiration of the token; or

b) holds a scheduler token which is the dominant token on the local link, and when both

   – the remaining allocated duration of token usage before the next scheduled activity;

   – the remaining allocated duration of scheduler token usage for link maintenance, which is V(LTHT) octet-durations minus the amount of link capacity used for link maintenance during the current cycle of "circulating the token",

   permit completion of the implied transaction – sending the RQ DLPDU, which requires an immediate reply, and awaiting the RR reply DLPDU – prior to expiration of the token.

The RQ DLPDU shall be sent at NORMAL priority.

If the DLE holds a delegated token, and no additional use of that token after sending this DLPDU and awaiting its immediate reply RR DLPDU is needed at that time, then the DLE may set the final-token-use subfield of the RQ DLPDU to the value FINAL; else that subfield shall have the value NOT-FINAL.

After sending an RQ DLPDU, the sending DLE shall monitor the local link for a reply as specified in 5.2.8.1. The permissible reply DLPDU is an RR DLPDU.

6.11.4 Receiving the RQ DLPDU

A received RQ DLPDU shall be treated as follows by the receiving DLE.

6.11.4.1 Actions required of all DLEs

If the destination NODE DL-address specified by the DLPDU designates the receiving DLE, then

a) when the receiving DLE's time-synchronization class (see 10.6.3) is NONE and the DLE does not maintain even an estimated C(NT), then the DLE shall append to the received DLPDU the 24-bit value zero; or

b) else if a) does not apply, then the DLE shall

   1) append to the received DLPDU the low-order 24 bits of the local node-time, C(NT), at which the RQ DLPDU reception completed (that is, receipt of the END-OF-DATA or END-OF-DATA-AND-ACTIVITY PhIDU);

2) adjust that 24-bit value to remove any systemic difference in the DLE-internal time delay between the DLE's transmit and receive paths, caused by known implementation considerations within the receiving real end system.

NOTE 1 This adjustment will probably vary inversely with the DLE's actual instantaneous rate of data transmission.

NOTE 2 The objective of this adjustment is to ensure that V(MD), as computed in 8.4.1.5, is as close as possible to double the sum of the one-way transmission, propagation, and reception delays incurred in transmission of a TD DLPDU between the addressed source node and the local DLE.

In either case the receiving DLE shall initiate a reply within a period of maximum-response-delay slot-times, V(MRD) × V(ST) octet-durations, of receipt of the RQ DLPDU. The reply DLPDU shall be an RR DLPDU with a destination DL-address equal to the source DL-address of the received RQ DLPDU, and with a source DL-address specifying the replying DLE's DL-support functions.

6.11.4.2 Additional actions required of a link-master class DLE

None.

6.11.4.3 Additional actions required of a bridge class DLE

The following considerations apply:

a) Since every bridge class DLE has link master capability, any actions specified in 6.11.4.2 also apply to a bridge class DLE.
b) The DLE shall not forward the DLPDU.

6.11.4.4 Additional actions required of the current LAS DLE

The LAS DLE shall act as specified in 5.2.8.2.

6.12 Round-trip-delay reply (RR) DLPDU

A ROUND-TRIP-DELAY REPLY (RR) DLPDU is sent from one DLE to another on the local link to permit completion of the measurement and computation of the round-trip delay intrinsic to their inter-communication. It is only sent as an immediate reply to a received RQ DLPDU.

6.12.1 Structure of the RR DLPDU

Table 23 – Structure of RR DLPDUs

Frame control	Destination address	Source address	Parameters
1101 0F00	N.0	N.0	RR-p

6.12.1.1 Frame control field

The frame control field shall specify

a) the DLPDU's function;
b) the DLPDU's implicit priority, which is NORMAL;
c) the length, number and type of DLPDU addresses;
d) whether or not transmission of the current DLPDU terminates use of a delegated token.

6.12.1.2 Address field

The address field shall consist of two explicit SHORT DL-addresses, destination and source, in that order.

6.12.1.3 Parameters field

The parameters field shall be structured and encoded as specified in 7.8.

6.12.1.4 User data field

The user data field shall be null.

6.12.2 Content of the RR DLPDU

The frame control field shall be encoded as specified in Table 23.

The address field shall specify two NODE DL-addresses expressed in SHORT DL-address format, where the destination DL-address is equal to the source DL-address received in the immediately prior RQ DLPDU, and the source DL-address has the value formed by concatenating the sender's node-id, $V_S(TN)$, and an octet of zero, thus specifying the sending DLE's DL-support functions.

The RR-parameters field shall be encoded as specified in 7.8 , and shall reflect the value of the DLE's node-time counter, if maintained by the DLE, at a moment of DLPDU formation which is a constant time offset from the moment when the END-OF-DATA-AND-ACTIVITY PhIDU will be transmitted for the DLPDU. Otherwise the DLE shall use the value zero for this field.

The user data field shall be null.

6.12.3 Sending the RR DLPDU

An RR DLPDU may be sent on the link when the sending DLE has just received a reply token in an RQ DLPDU, permitting a single transmission of an RR DLPDU. The final-token-use subfield of the reply RR DLPDU shall have the same value as that in the requesting RQ DLPDU.

6.12.4 Receiving the RR DLPDU

A received RR DLPDU shall be treated as follows by the receiving DLE.

6.12.4.1 Actions required of all DLEs

If the destination NODE DL-address specified by the DLPDU designates the receiving DLE, and if the immediately prior DLPDU was an RQ DLPDU sent by the receiving DLE, then the DLE shall consider the prior transmission to have been error-free, and if the DLE maintains C(NT), then the DLE shall

a) append to the received DLPDU the low-order 24 bits of the local node-time, C(NT), at which the RR DLPDU reception completed (that is, receipt of the END-OF-DATA or END-OF-DATA-AND-ACTIVITY PhIDU);
b) adjust that 24-bit value to remove any systemic difference in the DLE-internal time delay between the DLE's transmit and receive paths, caused by known implementation considerations within the receiving real end system;

   NOTE 1 This adjustment probably will vary inversely with the DLE's actual instantaneous rate of data transmission.

   NOTE 2 The objective of this adjustment is to ensure that V(MD), as computed in 8.4.1.5, is as close as possible to double the sum of the one-way transmission, propagation, and reception delays incurred in transmission of a TD DLPDU between the addressed destination node and the local DLE.

c) and the DLE shall process the received DLPDU as specified in 8.4.1.5.

6.12.4.2 Additional actions required of a link-master class DLE

None.

6.12.4.3 Additional actions required of a bridge class DLE

The following considerations apply:

a) Since every bridge class DLE has link master capability, any actions specified in 6.12.4.2 also apply to a bridge class DLE.
b) The DLE shall not forward the DLPDU.

6.12.4.4 Additional actions required of the current LAS DLE

If the final-token-use subfield of the received DLPDU has the value FINAL, then the LAS DLE shall

a) assume that the current use of the delegated token is terminated as if the token had been returned by an RT (see 6.17) DLPDU;
b) assume that the scheduler token is again dominant on the local link and resume active operation as the LAS.

6.13 Probe node DL-address (PN) DLPDU

A PROBE NODE DL-ADDRESS (PN) DLPDU is used by the LAS DLE to probe for the existence of previously unrecognized DLEs on the local link. A PN DLPDU creates and passes a reply token to any DLE(s) which may have their V(TN) (see 4.7.1.8) equal to the specified (configured but inactive) NODE DL-address. The only permissible reply is a PR DLPDU (see 6.14), which must be an immediate reply.

6.13.1 Structure of the PN DLPDU

Table 24 – Structure of PN DLPDUs

Frame control	Destination NODE-address	Parameters
0010 0110	N	PN-p

6.13.1.1 Frame control field

The frame control field shall specify

a) the DLPDU's function;
b) the length, number and type of DLPDU addresses.

6.13.1.2 Address field

The address field shall consist of an explicit destination NODE DL-address.

6.13.1.3 Parameters field

The parameters field shall be structured and encoded as specified in 7.9.

6.13.1.4 User data field

The user data field shall be null.

6.13.2 Content of the PN DLPDU

The frame control field shall be encoded as specified in Table 24.

The address field shall specify the NODE DL-address which is being probed.

The Probe NODE DL-ADDRESS parameters (PN-parameters) field shall convey, as specified in 7.9, the then-current values for the local link of those DLE and PhE parameters necessary for a receiving DLE to configure itself and its associated PhE so that they can reply to this or a subsequent PN DLPDU.

The user data fields shall be null.

6.13.3 Sending the PN DLPDU

The LAS on its own, at the beginning of each cycle of link maintenance-related transmissions, and possibly during what otherwise would be unoccupied periods of link capacity, probes configured but apparently-unused NODE DL-addresses by sending PN DLPDUs to those DL-addresses and awaiting a reply. Should such a reply occur, it is an indication that a new DLE has joined (or rejoined) the link, and the LAS then adds that DLE's NODE DL-address

– to its live list, V(LL), to be polled with PT DLPDUs;
– to its token-circulation-list, V(TCL), if so requested in the responding PR DLPDU.

PN DLPDUs are not otherwise sent to NODE DL-addresses.

The LAS DLE shall send a PN DLPDU when probing NODE DL-addresses which are not associated with any known active DLEs on the local link. A PN DLPDU may be sent on the link by the LAS DLE when

a) the LAS DLE is using the link capacity allocated for link maintenance as specified in Clause 9; or
b) the scheduler token is the dominant token on the local link; and the remaining allocated duration of token usage before the next scheduled activity permits completion of a PN - PR transaction as specified in Clause 9).

Before sending the PN DLPDU, the LAS DLE shall determine the next NODE DL-address to be probed by choosing, on alternate instances of sending the PN DLPDU, from the following two sets of NODE DL-addresses, as defined in 4.3.2.2, and cyclically in increasing numeric order within each set:

1) $\{ 10_{16}..F7_{16} \}$ – V(LL) – { V(FUN)..(V(FUN) + V(NUN) – 1) },
   which is the set of configured bridge-class, link-master-class and basic-class DLE addresses, minus the set of active DLE addresses, and minus the set of excluded unused DLE addresses
2) $\{ F8_{16}..FF_{16} \}$ – V(LL) ,
   which is the set of visitor DLE addresses, plus the set of DLE addresses for non-visitor DLEs which do not know their proper DLE addresses, minus the set of active DLE addresses.

After sending a PN DLPDU, the LAS DLE shall monitor local link activity for a period of immediate-response-recovery-delay slot-times, V(IRRD) × V(ST) octet-durations, waiting for a reply. After this interval, as soon as the link is inactive, the LAS DLE shall continue with its other activities. Even when a response has been received, the LAS shall wait until the end of the immediate-response-recovery-delay slot-times, V(IRRD) × V(ST) octet-durations, after the PN transmission, and then wait for the link to become inactive, before continuing with its next transaction, to provide for a potential second response from another DLE which has the same NODE DL-address.

NOTE This requirement imposes timer considerations which differ subtly from all other uses of slot-time-based timers in the DL-protocol. For this timer, the timing action does not stop when ACTIVITY or DATA is reported on the link, but rather continues even after such reports.

If the LAS receives a DLPDU, which is not a PR DLPDU, during this monitoring interval, then the LAS DLE shall drop its scheduler token and inform local DL-management of the event.

6.13.3.1 Additional considerations

From the LAS DLE's perspective, the maximum overhead link capacity required by any PN DLPDU and an upper bound on the maximum permitted responses may be pre-computed as

$$\begin{aligned} & 2 \times \text{framing-overhead} \\ + & 1 \times \text{PN-size} \\ + & 1 \times V(\text{IRRD}) \times V(\text{ST}) \\ + & 1 \times \text{maximum-PR-size} \\ + & 1 \times V(\text{MID}) \end{aligned}$$

6.13.4 Receiving the PN DLPDU

Receipt of a PN DLPDU by a newly attached or newly initialized DLE enables the DLE to learn those PhL and DLL configuration parameters which are essential for transmission on the local link.

Receipt of a PN DLPDU by the addressed DLE enables that DLE

– to indicate its presence and desired activity on the local link;
– to begin the process of being included in the link's other activities.

A received PN DLPDU shall be treated as follows by the receiving DLE.

6.13.4.1 Actions required of all DLEs

The DLE shall use the received PN DLPDU in the DLE-initialization procedures of 10.1.2.1 and 10.1.3. If those procedures require the receiving DLE to reply to the received PN DLPDU, then the DLE

a) shall assume a reply token;
b) shall initiate transmission of a PR DLPDU within a period of maximum-response-delay slot-times, V(MRD) × V(ST) octet-durations (as measured at the PN-receiving DLE);
c) shall include a probe-response SPDU, as described in Clause 9, in the user data field of that PR DLPDU, and shall ensure that any randomly chosen fields of that SPDU are chosen from approximately-uniform distributions in a manner that is statistically independent of similar choices made by other DLEs.

6.13.4.2 Additional actions required of a link-master class DLE

None.

6.13.4.3 Additional actions required of a bridge class DLE

The following considerations apply:

a) Since every bridge class DLE has link master capability, any actions specified in 6.13.4.2 also apply to a bridge class DLE.
b) The DLE shall not forward the DLPDU.

6.13.4.4 Additional actions required of the current LAS DLE

If the DLE did not just send the DLPDU, then the DLE shall drop the scheduler token and deactivate its role as LAS; and shall inform local DL-management of the event.

6.14 Probe response (PR) DLPDU

A PROBE RESPONSE (PR) DLPDU is sent to the LAS in response to the immediately prior PROBE NODE DL-ADDRESS (PN) DLPDU, to convey probe-response information to the LAS.

6.14.1 Structure of the PR DLPDU

Table 25 – Structure of PR DLPDUs

Frame control	User data
0010 0111	SPDU

6.14.1.1 Frame control field

The frame control field shall specify the DLPDU's function.

6.14.1.2 Address field

The address field shall be null.

6.14.1.3 Parameters field

The parameters field shall be null.

6.14.1.4 User data field

The user data field shall consist of a higher-level probe-response SPDU.

NOTE This small limit on the size of the probe-response SPDU facilitates probing of NODE DL-addresses during intervals within the link schedule which would otherwise be unusable.

6.14.2 Content of the PR DLPDU

The frame control field shall be encoded as specified in Table 25; the implicit value for the final-token-use subfield shall be FINAL.

The address field shall be null. The generic LAS NODE DL-address of 04 is implicitly addressed as the DLPDU's destination, and the destination address from the immediately prior PN DLPDU sent on the local link is implicitly the PR DLPDU's source address.

The parameters field shall be null.

The user data field shall convey a higher-level probe-response SPDU whose maximum size is limited as specified in 6.14.1.4.

6.14.3 Sending the PR DLPDU

A PR DLPDU shall result from receipt of a PN DLPDU. The PR DLPDU shall be sent as specified in Clause 9 and 6.13.4.

6.14.4 Receiving the PR DLPDU

A received PR DLPDU shall be treated as follows by the receiving DLE.

6.14.4.1 Actions required of all DLEs

None.

6.14.4.2 Additional actions required of a link-master class DLE

None.

6.14.4.3 Additional actions required of a bridge class DLE

The following considerations apply:

a) Since every bridge class DLE has link master capability, any actions specified in 6.14.4.2 also apply to a bridge class DLE.
b) The DLE shall not forward the DLPDU.

6.14.4.4 Additional actions required of the current LAS DLE

If the DLE had sent a PN DLPDU as the previous DLPDU on the link, then the DLE

a) shall report any higher-level SPDU conveyed by the received DLPDU, as if it had been received as DLS-user data contained in a DT DLPDU addressed to the LAS's NODE DL-address and sent from the NODE DL-address which was contained in the last-sent PN DLPDU (see 6.13.3), and shall process the SPDU as specified in Clause 9;
b) shall assume that the scheduler token is again dominant on the local link, resume active operation as the LAS, and initiate transmission
   1) within a period of maximum-response-delay slot-times, V(MRD) × V(ST) octet-durations, of the end of reception of the PR DLPDU, as measured at the receiving DLE,
   2) but not before the end of the observation interval following the PN DLPDU, as specified in 6.13.3.

If the DLE had sent a PN DLPDU as a previous DLPDU on the link, but another PR DLPDU was received after sending the PN DLPDU, and before reception of the current PR DLPDU, then the DLE shall ignore the current PR DLPDU and any contained SPDU.

If the DLE had not sent a PN DLPDU as the immediately prior non-PR DLPDU on the link, then the DLE shall drop the scheduler token and deactivate its role as LAS; and shall inform local DL-management of the event.

6.15 Pass token (PT) DLPDU

A PASS TOKEN (PT) DLPDU is used to pass a delegated token from the DLE functioning as an LAS to a DLE on the local link. By doing this repeatedly, the LAS DLE provides a delegated token which "circulates" successively, usually in NODE DL-address order, to all active DLEs on the local link which are included in the link's token-circulation list, V(TCL) (see 4.7.5.3).

NOTE This usage also supports migration from previously existing national standards.

This DLPDU provides the receiving DLE with the right to initiate DL-transactions for a period of time specified in the delegating DLPDU.

6.15.1 Structure of the PT DLPDU

Table 26 – Structure of PT DLPDUs

Frame control	Destination NODE-address	Parameters
0011 0FPP	N	DD-p

6.15.1.1 Frame control field

The frame control field shall specify

a) the DLPDU's function;
b) the minimum required priority of token use;

c) the phase (initial or continuation) of the token delegation;

d) the length, number and type of DLPDU addresses.

6.15.1.2 Address field

The address field shall consist of an explicit destination NODE DL-address.

6.15.1.3 Parameters field

The parameters field shall be structured and encoded as specified in 7.10.

6.15.1.4 User data field

The user data field shall be null.

6.15.2 Content of the PT DLPDU

The frame control field shall be encoded as specified in Table 26. The value of the priority subfield determines the minimum permitted priority of DLPDU which may be transmitted by the receiving DLE. The value of the final-use subfield shall be interpreted generally as specified in 5.2.1.2c), and specifically as:

> **RESTART** – this is the initial token delegation within the current cycle of “circulating the token”, and so any repetitively scheduled transactions should be restarted;
>
> **CONTINUE** – this is a subsequent (that is, secondary) token delegation within the current cycle of “circulating the token”, and so the previously initiated sequence of queued transactions should be continued.

The address field shall specify the NODE DL-address to which the token is being delegated.

The Delegation-Duration-parameter (DD-parameter) field shall convey, as specified in 7.10, the duration for which the token is being delegated to service Q(US), measured in octet-durations. Its permitted values are 0 to 65 000, measured in units of the transmission-duration of one octet (that is, in octet-durations).

NOTE The lower bound of zero is required to permit probing of node DL-addresses for DLEs which are included in the local link’s live-list, V(LL), but not in the local link’s token circulation list, V(TCL). The minimum practical value for use of this duration is about 16.

A duration of zero shall be used only when the addressed DLE’s NODE DL-address is in the live list, V(LL), but not in the list of DLEs, V(TCL), to be polled with PT DLPDUs. In this case the addressed DLE indicates its continued presence by responding with an RT or RI DLPDU.

The user data field shall be null.

6.15.3 Sending the PT DLPDU

The LAS DLE sends a PT DLPDU to “circulate the token”, usually in NODE DL-address order, through all of the active DLEs on the local link.

6.15.3.1 Determination of the PT DLPDU fields and related “token-rotation” parameters

The following considerations apply:

a) A PT DLPDU may be sent on the link

   – when the sending DLE (the LAS) holds a scheduler token which is the dominant token on the local link;

- when the number of octet-durations of link capacity remaining until the next scheduled activity permits sending the PT DLPDU, plus the minimal required use by the receiving DLE, plus return of the token in an RI DLPDU or recovery of the token if required;
- when the sending DLE does not need to send some other DLPDU.

b) The overhead in delegating the token, which is not included in the delegated link capacity specified in the DD-parameters of the PT DLPDU, but which is factored into the determination of whether to send the PT DLPDU, is computed as the sum of

1) the link capacity required to send the PT DLPDU;
2) V(IRRD) × V(ST) octet-durations for token acceptance and commencement of initial use;
3) the link capacity required for the delegated-token-holder to send an RI DLPDU terminating token usage;
4) the larger of
   i) the minimum inter-DLPDU delay, V(MID) octet-durations (see 4.7.1.12), which follows the RI DLPDU,
   ii) any implementation-dependent period, measured in units of link capacity, required to recover the token upon expiration of the link capacity actually delegated.

c) The maximum number of octet-durations of link capacity which can be delegated is computed as the smaller of

1) 65 000 octet-durations,
2) the number of octet-durations of link capacity remaining until the next scheduled activity minus the overhead in delegating the token specified in b), with the result reduced to account for possible differences in the two DLEs' data rates (that is, multiplied by the minimum possible ratio between the LAS DLE's and receiving DLE's Ph-data rates, as reported in 4.4.1.1).

d) The minimum number of octet-durations of link capacity which a DLE requires shall be computed as follows:

1) zero, when the DLE's NODE DL-address is not a member of the set V(TCL), indicating that the DLE does not require use of the token;
2) V(DMDT) (see 4.7.5.7), when 1) does not apply, and when
   - the PT DLPDU's final-use subfield specifies RESTART, or
   - the last token sent by a PT DLPDU to the same DL-address was not returned by an RI DLPDU; or
3) the value from the DD-parameters of the RI DLPDU which returned the last token sent by a PT DLPDU to the same DL-address, when 1) and 2) do not apply.

e) The maximum number of octet-durations of link capacity which may be delegated to a DLE shall be computed as follows:

1) zero, when the DLE's NODE DL-address is not a member of the set V(TCL), indicating that the DLE does not require use of the token;
2) the member of the array V(MTHA) corresponding to the DLE's NODE DL-address, when 1) does not apply, and when the PT DLPDU's final-use subfield specifies RESTART;
3) the member of the array V(RTHA) corresponding to the DLE's NODE DL-address, when 1) and 2) do not apply.

   NOTE The link capacity actually used by token-returning RI and RT DLPDUs is not included in this computation, since that link capacity is allocated to the overhead of the token-passing process.

f) The token shall be delegated in ascending order of NODE DL-addresses to those DLE NODE DL-addresses represented in V(LL), beginning with the lowest-numbered such NODE DL-address. When a DLE has just assumed the scheduler token (by sending a CL DLPDU or receiving a TL DLPDU) and become the LAS, it shall commence token-passing with this lowest-numbered NODE DL-address, and shall restart token-passing with the lowest-numbered DLE after all other DLE's represented in the set V(LL) have terminated their token usage

– by specifying a final use of FINAL, or
– by being unable to use the token without exceeding the maximum number of octet-durations of link capacity determined in e).

A DLE can, but need not, be temporarily bypassed in this ascending order while its requirements, as computed in d), exceed the link capacity momentarily available for token delegation, as computed in c). Any such bypassed DLE shall be delegated its remaining allocated link capacity before a PT DLPDU whose final-use field specifies RESTART is again delegated to that DLE.

NOTE This bypassing is an implementation choice, since its existence merely improves the momentary efficiency of the link and does not affect interoperability.

It is a protocol error to specify any DL-address other than a valid NODE DL-address, or to explicitly address a DLE which is known to be not active on the local link.

g) The actual period of token delegation shall be computed to be the lesser of the values computed in c) and e), and the delegation shall occur only if that period is at least as great as the value computed in d).

h) The initial PT DLPDU sent to each DLE on the link during one cycle of "circulating the token" shall specify RESTART in its final-use field; all subsequent PT DLPDUs sent to the same DLE during the same cycle, if any, shall specify CONTINUE in that final-use field. When RESTART is specified, the DLE also shall initialize the relevant member of V(RTHA) to the value of the corresponding member of V(MTHA).

j) The actual period of token rotation, V(ATRT), shall be measured by the LAS DLE as the interval between successive occurrences of the LAS DLE's sending the PT DLPDU, with a token-use subfield specifying RESTART (see 6.15.2), to the lowest-numbered NODE DL-address represented in V(LL).

k) The DLL priority specified in the PT DLPDU shall be determined before each cycle of "circulating the token" as follows:

   1) After the LAS DLE has just assumed the scheduler token (by sending a CL DLPDU or receiving a TL DLPDU), the priority shall be NORMAL.

   2) If 1) does not apply, and the just-computed V(ATRT) was greater than V(TTRT), then

      – DL-management shall be notified of the event;

      – the token priority shall be increased to the next higher level of urgency, if possible.

   3) If 1) and 2) do not apply, so that the just-computed V(ATRT) was less than or equal to V(TTRT), then the token priority shall be decreased to the next lower level of urgency, if possible.

   NOTE The inclusion of hysteresis or "learning" in this priority adjustment is an area of future study.

Once determined, the same DLL priority shall be included in all PT DLPDUs transmitted during that cycle of "circulating the token".

6.15.3.2 Sending the PT DLPDU and monitoring the DLE to which the token is delegated

When sending the PT DLPDU, the LAS DLE shall record the destination address from the DLPDU as the delegation-address, V(DTA) (see 4.7.5.1), for potential DL-management use in case the token is lost and not returned, and for potential association with any received RI or RT DLPDU.

After sending a PT DLPDU, the LAS DLE shall monitor the local link for a period of immediate-response-recovery-delay slot-times, V(IRRD) × V(ST) octet-durations, waiting for a response:

a) If a Ph-Data indication (see 4.4.4) reporting DATA is received, then the DLE shall proceed as in c).

b) If a) does not apply, and the monitoring period of immediate-response-recovery-delay slot-times, V(IRRD) × V(ST) octet-durations expires, then

   1) if the bus is not active at that moment (that is, the last-received Ph-Data indication reported END-OF-ACTIVITY), then the DLE shall proceed as in d);

   2) if 1) does not apply, implying that the link is still active at that moment (that is, the last-received Ph-Data indication reported START-OF-ACTIVITY), then

      A) for implementations that were not demonstrable on or before 31 December 1995, the DLE shall monitor the local link for a period of one additional slot-time, V(ST) octet-durations, waiting for a Ph-Data indication:

         i) if a Ph-Data indication reporting DATA is received, then the DLE shall proceed as in c).

         ii) if a Ph-Data indication reporting END-OF-ACTIVITY is received, and i) does not apply, then the DLE shall stop further monitoring and proceed as in d).

         NOTE Without provision ii), ill-timed low-level noise can cause the LAS DLE to not retry the token pass, resulting in omitting a DLE from the current cycle of circulating the token. Provision ii) enables the LAS DLE to distinguish probable noise events from actual transmissions.

         iii) if neither i) nor ii) applies, and the monitoring period expires before a Ph-Data indication is received, then the DLE shall proceed as in c).

      B) Implementations that were demonstrable on or before 31 December 1995 alternatively may just proceed as in c).

c) Until the delegated token is returned, the LAS DLE shall continually monitor local link activity, and if it observes

   1) a period of token-recovery-delay slot-times, P(TRD) × V(ST) octet-durations, of continuous non-activity; or

   2) for implementations that were not demonstrable on or before 31 December 1995, a period of token-recovery-delay slot-times, P(TRD) × V(ST) octet-durations, during which any continuous period of activity (that is, between a Ph-Data indication reporting START-OF-ACTIVITY and the next following Ph-Data indication reporting END-OF-ACTIVITY or END-OF-DATA-AND-ACTIVITY) both

      i) did not result in a Ph-Data indication (see 4.4.4) reporting DATA, and

      ii) had a duration of less than one to two V(ST) octet-durations

      NOTE The vagueness in the extent of permissible activity duration permits some economy of implementation of the measurement.

   then it shall

   – assume that the scheduler token is dominant on the local link;
   – inform local DL-management of the event;
   – commence transmission on the link.

d) If the monitoring period of immediate-response-recovery-delay slot-times, V(IRRD) × V(ST) octet-durations expires and a) does not apply; or, for implementations that were not demonstrable on or before 31 December 1995, the additional monitoring period of one slot-time, V(ST) octet-durations, expires and c) does not apply; then the LAS DLE shall retry sending the PT DLPDU and monitoring the local link, up to the number of times specified as the maximum-retry-count, V(MRC) (see 4.7.1.5), of the local link, each time using as the value for its duration parameter the smaller of

   1) the value originally sent; or

   2) the number of octet-durations of link capacity remaining until the next scheduled activity minus the overhead in delegating the token specified in 6.15.3.1b), with the result reduced to account for possible differences in the two DLEs' data rates as specified in 6.15.3.1c)2),

   provided that the retry and attempted re-delegation shall occur only if that period is at least as great as the value computed in 6.15.3.1d).

This retry shall commence within token-recovery-delay slot-times, P(TRD) × V(ST) octet-durations, of the beginning of the current period of link non-activity.

If all attempted retries are unsuccessful, then the LAS DLE shall

– inform local DL-management of the event;

NOTE DLEs with node DL-addresses in the set { $F8_{16}$..FF_{16} } are expected to terminate operation by dropping out of the token circulation process. Thus, DL-management should not treat such occurrences as evidence of DLE or local-link malfunction.

– start the next transmission within token-recovery-delay slot-times, P(TRD) × V(ST) octet-durations, of the beginning of the current period of link non-activity.

After three such event reports for any specific NODE DL-address within three cycles of “circulating the token”, DL-management shall remove that NODE DL-address from the LAS’s local-link token-circulation-list, V(TCL), and local-link live-list, V(LL).

6.15.4 Receiving the PT DLPDU

Receipt of a PT DLPDU by the addressed DLE enables the DLE to emulate a “circulated token” as found in previous token-passing bus standards, such as ISO/IEC 8802-4, by enabling transmission of DLPDUs at the specified and higher priority either

– until all pending transmissions have occurred, or

– until the portion of the local-link’s transmission capacity which was allocated by the DD-parameters of the received PT DLPDU requires return of the delegated token.

A received PT DLPDU shall be treated by the receiving DLE as follows:

6.15.4.1 Actions required of all DLEs

If the destination DL-address specified by the DLPDU designates the NODE DL-address of the receiving DLE, then if the receiving DLE, due to its construction, does not use the delegated token, then the receiving DLE shall initiate transmission of an RT DLPDU within a period of maximum-response-delay slot-times, V(MRD) × V(ST) octet-durations, of the end of reception of the PT DLPDU, as measured at the receiving DLE; else the receiving DLE shall

a) copy the value of the duration parameter from the DLPDU into its local remaining-duration down-counter, C(RD);
b) assume the delegated token;
c) initiate transmission within a period of maximum-response-delay slot-times, V(MRD) × V(ST) octet-durations, of the end of reception of the PT DLPDU, as measured at the receiving DLE;
d) repeatedly employ the following selection criteria until the delegated token is returned to the LAS, each time initiating transmission within a period of maximum-response-delay slot-times, V(MRD) × V(ST) octet-durations, of the end of the prior transmission (unless that was a CA, CD, CT, ED or RQ transmission), as measured at the PT-receiving DLE.

The counter C(RD) shall decrement and be managed as specified in 4.7.1.11.

Transactions of the priority specified in the delegating PT DLPDU, or of higher priority shall be initiated successively, beginning as follows:

– starting from the first transaction and including all REPETITIVE sequences if the token-use subfield of the received PT DLPDU specified RESTART, or

– continuing from the previously initiated transaction and thus excluding all REPETITIVE sequences which have been executed since the last PT DLPDU specifying RESTART was received if the token-use subfield of the received PT DLPDU specified CONTINUE;

subject to the constraint that the DLE shall limit its token use to a sequence of transactions which is guaranteed to complete within the duration specified in the DD-parameter of the received PT DLPDU.

6.15.4.1.1 Selection of the next transaction to be executed

If the DLE has a need to send a CT DLPDU, the DLE shall transmit a CT DLPDU and await its immediate reply. Otherwise, the receiving DLE repeatedly shall select the next member, if any, of the DLE's unscheduled-service queue, Q(US), whose priority is not less than the priority of the received PT, and shall apply all of the remainder of this subclause, as appropriate:

a) If that next member is a reference to a DL-address user-request queue, Q_A(UR), then

   1) if the member is a reference which signals the need to compel a DLSDU transmission from the remote correspondent of a local peer or subscriber DLCEP (see 4.7.1.17a)1)), then the DLE shall

      i) transmit a CD DLPDU to the remote DLC endpoint, with SD-parameters conveying the state of the local DLCEP to the remote DLCEP, as specified in 8.2.2.1 and 8.2.2.4;

      ii) await an immediate reply;

      NOTE The maximum DLPDU size of that immediate reply can be determined solely on the basis of the negotiated DLCEP parameters.

      iii) delete the referencing member from the DLE's unscheduled-service queue, Q(US).

   2) if the member is a reference which signals the need to compel an instance of unitdata exchange with a specified DLSAP-address at a specified service priority (see 4.7.1.17a)2)), then the DLE shall

      i) transmit an ED or CD DLPDU to that remote DLSAP-address, at the specified service priority, as appropriate, as specified in 8.2.2.1 and 8.2.2.4;

      ii) await an immediate reply;

      NOTE The maximum DLPDU size of that immediate reply can be determined solely on the basis of the negotiated DLCEP parameters.

      iii) delete the referencing member from the DLE's unscheduled-service queue, Q(US).

   3) if the member is a reference which signals the need to transmit the contents of a sending buffer which is bound to a local DLCEP (see 4.7.1.17a)3)), then the DLE shall

      i) transmit the DLPDU appropriate to the state of the DLCEP;

      ii) if the resultant transmitted DLPDU completes the transmission of a DLSDU (so that no segments of that DLSDU remain to be transmitted), then the DLE shall

         – notify the DLS-user of the request's completion, if appropriate;

         – delete the referencing member from the DLE's unscheduled-service queue, Q(US).

   4) if the member is a reference which signals the need to send a DLSDU from the sending queue which is bound to a local DLCEP or "DLSAP-address" (see 4.7.1.17a)4)), or to send a DLPDU from the DLE's NODE DL-address, then the DLE shall select a member of the referenced Q_A(UR), of the specified or higher priority, which has been released for transmission and which requires full or partial transmission or retransmission, as follows:

      – either the first member of the second partition of Q_A(UR), or

      – the first member of the first partition of Q_A(UR) which has one or more segments marked for retransmission.

      NOTE Due to potential concurrency in real implementations, the set of segments so marked may not reflect status information from DLPDUs recently received at the DLCEP.

      If such a member exists, then

i) the DLE shall transmit the DLPDU with a segment of the DLSDU appropriate to the transmission segment status, $V_{C,K}$ (SS) (see 4.7.4.8), if applicable, of that member of that Q_A(UR);

ii) if the resultant transmitted DLPDU completes the transmission of a DLSDU (so that no segments of that DLSDU remain to be transmitted), then the DLE shall

– advance that member of Q_A(UR) to the already sent partition, or remove it, as appropriate to the type of member;

– notify the DLS-user of the request's completion, if appropriate;

– delete the referencing member from the DLE's unscheduled-service queue, Q(US).

If no such member of the referenced DL-address request queue, Q_A(UR), exists, then

iii) if the DL-address is a DLCEP-address of a local peer or publisher DLCEP, then the DLE shall

A) transmit a DT or RC DLPDU, as appropriate for the state of the DLCEP, with a null user data field, to convey the state of the DLCEP to the correspondent peer or subscriber DLCEP(s);

B) delete the referencing member from the DLE's unscheduled-service queue, Q(US).

iv) otherwise, when iii) does not apply, then the DL-address is not a DLCEP-address and the DLE shall repeat the selection process.

b) If that next member is a reference to a REPETITIVE sequence (see 8.4.3.1d)), then

1) if there are one or more active members of the sequence (that is, not excluded by the most recent DL-SUBSET-SEQUENCE request, if any, specifying the sequence);

2) if one or more of those active members has not been executed since the DLE last received a PT DLPDU specifying RESTART,

then the DLE shall

i) insert a reference to the next active member of that sequence on Q(US) immediately before the reference to the sequence (with the result that the newly-inserted reference will be deleted at completion of the just-scheduled action);

ii) continue as in the preceding case a), as determined by the type of the just-inserted member of the sequence.

NOTE At completion of the just-scheduled action, the next member of Q(US) will be the reference to the REPETITIVE sequence, and the above procedure will continue until all active members of the REPETITIVE sequence have been processed.

3) otherwise, when neither 1) nor 2) applies, then the DLE shall repeat the selection process.

c) If that next member is a reference to a sequence (see 8.4.3.1a)) which is not a REPETITIVE sequence, and if the sequence consists of a single element, then the DLE shall

1) replace the reference to the sequence on Q(US) with the reference to the single element, so that the same member of Q(US) now is the element reference;

2) issue a DL-SCHEDULE-SEQUENCE confirm primitive if it has not already been confirmed;

3) delete the sequence;

4) continue as in the preceding case a), as determined by the type of the sole element of the selected and just-deleted sequence.

All other cases are erroneous and are not permitted.

6.15.4.1.2 Additional considerations

The following considerations also apply:

a) If there are no additional members, of the priority specified in the received PT DLPDU, or of any higher priority, in the DLE's unscheduled-service queue, Q(US), and if immediate retry of the currently-selected transaction is not possible, then the final-token-use subfield in that non-RT DLPDU can be set to the value FINAL before transmission, thereby returning the delegated token to the LAS DLE at the end of the current transaction.

b) If there are no members, of the priority specified in the received PT DLPDU, or of any higher priority, in the DLE's unscheduled-service queue, Q(US), and the delegated token was not returned by setting a final-token-use subfield to the value FINAL in the last DLPDU transmitted by the DLE, then the DLE shall return the delegated token by transmitting an RT DLPDU.

c) If at any time

   1) the remaining duration of token usage – as indicated by the remaining-duration down-counter, C(RD) – is inadequate to permit any further use by the receiving DLE;

   2) the DLE's unscheduled-service queue, Q(US), contains additional members whose priority is equal to or higher than the priority specified in the last-received PT DLPDU,

   then the receiving DLE shall return the delegated token to the LAS by sending an RI DLPDU specifying the minimum delegation interval for token use required when the token is next returned.

   NOTE An RI DLPDU, which is sent to terminate use of a token delegated by a PT DLPDU, indicates to the receiving LAS that additional service is needed and that another PT DLPDU, with a final-token-use subfield specifying CONTINUE, should be sent to the DLE during the current cycle of "circulating the token" if the total link capacity allocated to the DLE by the DLE'th member of V(MTHA) (see 4.7.5.10) so permits.

d) If none of a) to c) applies, then the DLE shall transmit the non-RT non-RI DLPDU with a final-token-use subfield value of NOT-FINAL.

6.15.4.2 Additional actions required of a link-master class DLE

The DLE may, but is not required to, record the destination address from the DLPDU as the delegation-address, V(DTA) (see 4.7.5.1), for potential use in restarting the circulated token should the current LAS fail and this LM DLE become the new LAS. The DLE may also use that recorded V(DTA) to detect additions to, but not deletions from, the set of DLEs receiving the circulated token, and update its copy of the token circulation list, V(TCL) (see 4.7.5.3) accordingly.

NOTE These actions optimize recovery of the token circulation process in the event of failure of the DLE currently functioning as LAS.

6.15.4.3 Additional actions required of a bridge class DLE

The following considerations apply:

a) Since every bridge class DLE has link master capability, any actions specified in 6.15.4.2 also apply to a bridge class DLE.

b) The DLE shall not forward the DLPDU.

6.15.4.4 Additional actions required of the current LAS DLE

If the DLE did not just send the DLPDU, then the DLE shall drop the scheduler token and deactivate its role as LAS; and shall inform local DL-management of the event.

6.16 Execute sequence (ES) DLPDU

An EXECUTE SEQUENCE (ES) DLPDU is used to pass a delegated token from the DLE functioning as an LAS to a DLE on the local link which has previously requested or scheduled local link capacity, or for which such a schedule has been configured by DL-management.

NOTE This usage supports migration from previously-existing national standards.

This DLPDU provides the receiving DLE with the right to initiate DL-transactions for a period of time, specified in the delegating DLPDU; the destination DL-address of that DLPDU is used to determine which set of transactions to initiate.

6.16.1 Structure of the ES DLPDU

Table 27 – Structure of ES DLPDUs

format	Frame control	Destination DLSEP-address	Parameters
1L	`1000 1F00`	HL.N.S	DD-p
1S	`1000 0F00`	N.S	DD-p

6.16.1.1 Frame control field

The frame control field shall specify

a) the DLPDU's function;

b) the phase (initial or continuation) of the sequence execution;

c) the length, number and type of DLPDU addresses.

6.16.1.2 Address field

The address field shall consist of an explicit destination DL-address.

6.16.1.3 Parameters field

The parameter field shall be structured and encoded as specified in 7.10.

6.16.1.4 User data field

The user data field shall be null.

6.16.2 Content of the ES DLPDU

The frame control field shall be encoded as specified in Table 27. The value of the final-use subfield shall be interpreted generally as specified in 5.2.1.2c), and specifically as:

> **RESTART** – this is the initial token delegation within the current cycle of scheduled sequence execution, and so the indicated sequence and any repetitively scheduled transactions should be restarted;
>
> **CONTINUE** – this is a subsequent (that is, secondary) token delegation within the current cycle of scheduled sequence execution, and so the previously-initiated sequence of scheduled transactions should be continued at the point where execution was last suspended and the ES DLPDU was re-requested (see 6.18.3).

The address field shall specify the DLSEP-address to which the token is being delegated. For format 1L, the address shall be LONG; for format 1S it shall be SHORT.

The Delegation-Duration-parameter (DD-parameter) field shall convey, as specified in 7.10 the duration for which the token is being delegated to execute the addressed scheduled sequence, measured in units of the transmission-period of one octet on the local link (that is, eight times the local link's nominal bit period). It has the range 0 to 65 000 octet-durations.

The user data field shall be null.

6.16.3 Sending the ES DLPDU

The LAS DLE sends an ES DLPDU when so instructed by the local link's current schedule, or in response to requests to extend the time allocated for an earlier scheduled (and executed) token delegation.

6.16.3.1 Sending the ES DLPDU as instructed by the local link's current schedule

An ES DLPDU may be sent on the link when the sending DLE (the LAS) holds a scheduler token which is the dominant token on the local link, and the link's schedule requires transmission of the ES DLPDU. The final-use subfield of such an ES DLPDU shall be dictated by the schedule.

6.16.3.2 Sending the ES DLPDU in response to earlier requests for extensions of scheduled token delegation

An ES DLPDU shall only be sent as a response to an RI DLPDU as specified in Clause 9. If the LAS DLE sends an ES DLPDU which is not dictated by the link's current schedule, then the following procedures shall apply:

a) An ES DLPDU may be sent on the link when the sending DLE (the LAS) holds a scheduler token which is the dominant token on the local link, and when the number of octet-durations of link capacity remaining until the next scheduled activity permits sending the ES DLPDU, plus the minimal required use by the receiving DLE, plus return of the token in an RI DLPDU or recovery of the token if required. The final-use subfield of such an ES DLPDU shall specify CONTINUE.
b) The overhead in delegating the token, which is not included in the delegated link capacity specified in the DD-parameters of the ES DLPDU, but which is factored into the determination of whether to send the ES DLPDU, is computed as the sum of
   1) the link capacity required to send the ES DLPDU;
   2) V(IRRD) × V(ST) octet-durations for token acceptance and commencement of initial use;
   3) the link capacity required for the delegated-token-holder to send an RI DLPDU terminating token usage;
   4) the larger of
      i) the minimum inter-DLPDU delay, V(MID) octet-durations (see 4.7.1.12), which follows the RI DLPDU; or
      ii) any implementation-dependent period, measured in units of link capacity, required to recover the token upon expiration of the link capacity actually delegated.
c) The maximum number of octet-durations of link capacity which can be delegated is computed as the smaller of
   1) 65 000 octet-durations; or
   2) the number of octet-durations of link capacity remaining until the next scheduled activity minus the overhead in delegating the token specified in b), with the result reduced to account for possible differences in the two DLEs' data rates (that is, multiplied by the minimum possible ratio between the LAS DLE's and receiving DLE's Ph-data rates, as reported in 4.4.1.1).
d) The actual period of token delegation shall be the value returned in the DD-parameters of the RI DLPDU which terminated the prior interval of delegated token usage at the same DL-address.

6.16.3.3 Monitoring the DLE to which the token is delegated and additional considerations

It is a protocol error to specify any DL-address other than a valid DLSEP-address, or to explicitly or implicitly address a DLE not active on the local link. However, an LAS which receives a DLSEP-address from another DLE in a scheduling-related SPDU may presume that such an address is valid.

When sending the ES DLPDU, the LAS DLE shall record the destination address from the DLPDU as the delegation-address, V(DTA) (see 4.7.5.1), for potential DL-management use in case the token is lost and not returned, and for potential association with any received RI or RT DLPDU.

After sending an ES DLPDU, the LAS DLE shall monitor the local link for a period of immediate-response-recovery-delay slot-times, V(IRRD) × V(ST) octet-durations, waiting for a response:

a) If a Ph-Data indication (see 4.4.4) reporting DATA is received, then the DLE shall proceed as in c).

b) If a) does not apply, and the monitoring period of immediate-response-recovery-delay slot-times, V(IRRD) × V(ST) octet-durations expires, then

   1) if the bus is not active at that moment (that is, the last-received Ph-Data indication reported END-OF-ACTIVITY), then the DLE shall proceed as in d).

   2) if 1) does not apply, implying that the link is still active at that moment (that is, the last-received Ph-Data indication reported START-OF-ACTIVITY), then

      i) for implementations that were not demonstrable on or before 31 December 1995, the DLE shall monitor the local link for a period of one additional slot-time, V(ST) octet-durations, waiting for a Ph-Data indication:

         A) if a Ph-Data indication reporting DATA is received, then the DLE shall proceed as in c).

         B) if a Ph-Data indication reporting END-OF-ACTIVITY is received, and i) does not apply, then the DLE shall stop further monitoring and proceed as in d);

         NOTE Without provision ii), ill-timed low-level noise can cause the LAS DLE not to retry the token pass, resulting in omitting a DLE from the current cycle of schedule execution. Provision ii) enables the LAS DLE to distinguish probable noise events from actual transmissions.

         C) if neither A) nor B) applies, and the monitoring period expires before a Ph-Data indication is received, then the DLE shall proceed as in c).

      ii) Implementations that were demonstrable on or before 31 December 1995 alternatively may just proceed as in c).

c) Until the delegated token is returned, the LAS DLE shall continually monitor local link activity, and if it observes

   1) a period of token-recovery-delay slot-times, P(TRD) × V(ST) octet-durations, of continuous non-activity; or

   2) for implementations that were not demonstrable on or before 31 December 1995, a period of token-recovery-delay slot-times, P(TRD) × V(ST) octet-durations, during which any continuous period of activity (that is, between a Ph-Data indication reporting START-OF-ACTIVITY and the next following Ph-Data indication reporting END-OF-ACTIVITY or END-OF-DATA-AND-ACTIVITY) both

      i) did not result in a Ph-Data indication (see 4.4.4) reporting data, and

      ii) had a duration of less than one to two V(ST) octet-durations,

      NOTE The vagueness in the extent of permissible activity duration permits some economy of implementation of the measurement.

   then it shall

   – assume that the scheduler token is dominant on the local link;
   – inform local DL-management of the event;
   – commence transmission on the link.

d) If the monitoring period of immediate-response-recovery-delay slot-times, V(IRRD) × V(ST) octet-durations expires and a) does not apply; or, for implementations that were not demonstrable on or before 31 December 1995, the additional monitoring period of one slot-time, V(ST) octet-durations, expires and c) does not apply; then the LAS

DLE shall retry sending the ES DLPDU and monitoring the local link, up to the number of times specified as the maximum-retry-count, V(MRC) (see 4.7.1.5), of the local link, each time using as the value for its duration parameter the value originally-sent, and the re-delegation shall occur only if that period is at least as great as the value specified in 6.16.3.1 or 6.16.3.2d).

This retry shall commence within token-recovery-delay slot-times, P(TRD) × V(ST) octet-durations, of the beginning of the current period of link non-activity.

If all attempted retries are unsuccessful, then the LAS DLE shall

– inform local DL-management of the event;
– start the next transmission within token-recovery-delay slot-times, P(TRD) × V(ST) octet-durations, of the beginning of the current period of link non-activity.

6.16.4 Receiving the ES DLPDU

Receipt of an ES DLPDU by the addressed DLE enables the DLE to execute all or part of a previously scheduled sequence whose initiation is synchronized by the LAS, either

– until the sequence has been completed, or
– until the portion of the local-link's transmission capacity which was allocated by the DD-parameters of the received ES DLPDU requires return of the delegated token.

NOTE This latter situation can occur

A) on sequences which permit interruption;

B) on sequences which permit dynamic appending of DL-COMPEL-SERVICE requests;

C) when retransmission of the initiating ES DLPDU, or possibly of another DLPDU within the defined sequence, caused link capacity beyond that allocated to be needed.

A received ES DLPDU shall be treated by the receiving DLE as follows.

6.16.4.1 Actions required of all DLEs

6.16.4.1.1 When addressed to a DL-address of the DLE other than a DLSEP-address

If the destination DL-address specified by the DLPDU designates a DL-address of the receiving DLE other than a DLSEP-address, then the DLE shall

a) inform local DL-management of the event;
b) initiate transmission within a period of maximum-response-delay slot-times, V(MRD) × V(ST) octet-durations, as measured at the receiving DLE, of an RT DLPDU.

NOTE The receiving DLE is rejecting the received ES DLPDU.

6.16.4.1.2 When addressed to a DLSEP-address of the DLE

If the destination DL-address specified by the DLPDU designates a DLSEP-address of the receiving DLE, then the receiving DLE shall

a) copy the value of the duration parameter from the DLPDU into its local remaining-duration down-counter, C(RD);
b) assume the delegated token;
c) initiate transmission within a period of maximum-response-delay slot-times, V(MRD) × V(ST) octet-durations, of the end of reception of the ES DLPDU, as measured at the receiving DLE;
d) repeatedly employ the following selection criteria until the delegated token is returned to the LAS, each time initiating transmission within a period of maximum-response-delay

slot-times, V(MRD) × V(ST) octet-durations, of the end of the prior transmission (unless that was a CA, CD, CT or ED transmission), as measured at the ES-receiving DLE.

The counter C(RD) shall decrement and be managed as specified in 4.7.1.11.

Transactions in the sequence associated with the designated DLSEP-address

1) shall be initiated successively;
2) beginning as follows:
   – starting with the initial transaction of the sequence if the token-use subfield of the received ES DLPDU specified RESTART, or
   – continuing from the point at which the sequence was previously suspended if the token-use subfield of the received ES DLPDU specified CONTINUE;
3) subject to the constraint that the worst-case link capacity required to complete the transaction be less than the then-current value of the remaining-duration down-counter, C(RD).

6.16.4.1.2.1 Selection of the next transaction to be executed

The receiving DLE shall execute the transactions in the sequence as follows.

If that next member of the sequence is a reference to a DL-address user-request queue, Q_A(UR), then

a) if the member is a reference to the remote correspondent of a local peer or subscriber DLCEP (see 8.4.2.1b)1)i)), then the DLE shall
   1) transmit a CD DLPDU to the remote DLC endpoint, with SD-parameters conveying the state of the local DLCEP to the remote DLCEP, as specified in 8.2.2.1 and 8.2.2.4;
   2) await an immediate reply;

      NOTE The maximum DLPDU size of that immediate reply can be determined solely on the basis of the negotiated DLCEP parameters.
   3) advance to the next element of the sequence which was addressed by the received ES DLPDU;
b) if the member is a reference which indicates the need to perform a unitdata exchange with a specified DLSAP-address at a specified service priority (see 8.4.2.1c)1)i)), then the DLE shall
   1) transmit an ED or CD DLPDU to that remote DLSAP-address, at the specified service priority, as appropriate, as specified in 8.2.2.1 and 8.2.2.4;
   2) await an immediate reply;

      NOTE The maximum DLPDU size of that immediate reply can be determined solely on the basis of the negotiated DLCEP parameters.
   3) advance to the next element of the sequence which was addressed by the received ES DLPDU.

   Otherwise, the DLE shall select the first member of the referenced Q_A(UR) which has been released for transmission and which requires full or partial transmission or retransmission.
c) If such a member exists, then
   1) the DLE shall transmit the DLPDU appropriate to the state of that member of that Q_A(UR);
   2) if the resultant transmitted DLPDU completes the transmission of a DLSDU (so that no segments of that DLSDU remain to be transmitted), then the DLE shall
      – advance that member of Q_A(UR) to the already-sent partition, or remove it, as appropriate to the type of member;

– notify the DLS-user of the request's completion, if appropriate;

– advance to the next element of the sequence which was addressed by the received ES DLPDU.

d) If no such member of the referenced DL-address request queue, Q_A(UR), exists, then

1) if the DL-address is

– a DLCEP-address of a local peer or publisher DLCEP, or

– a publisher DLCEP-address of a local subscriber DLCEP,

and the sending binding of the DLCEP is to a buffer which is not empty, then

i) the DLE shall transmit the DLPDU appropriate to the state of the DLCEP;

ii) if the resultant transmitted DLPDU completes the transmission of a DLSDU (so that no segments of that DLSDU remain to be transmitted), then the DLE shall notify the DLS-user of the request's completion, if appropriate.

2) Otherwise, when i) does not apply, and the DL-address is a DLCEP-address of a local peer or publisher DLCEP, then the DLE shall transmit a DT or RC DLPDU, as appropriate for the state of the DLCEP, with a null user data field, to convey the state of the DLCEP to the correspondent peer or subscriber DLCEP(s).

3) Whether or not a DLPDU was transmitted, the DLE shall advance to the next member of the sequence which was addressed by the received ES DLPDU.

All other cases are erroneous and are not permitted.

6.16.4.1.2.2 Additional considerations

The following considerations also apply:

a) If there are no additional members in the sequence, and if immediate retry of the currently-selected transaction is not possible, then the final-token-use subfield in that non-RT DLPDU can be set to the value FINAL before transmission, thereby returning the delegated token to the LAS DLE at the end of the current transaction.

b) If the sequence is completed, and the delegated token was not returned by setting a final-token-use subfield to the value FINAL in the last DLPDU transmitted by the DLE, then the DLE shall return the delegated token by transmitting an RT DLPDU.

c) If at any time

1) the remaining-duration down-counter, C(RD), is inadequate to permit any further use by the receiving DLE;

2) the sequence contains additional members,

then the receiving DLE shall return the delegated token to the LAS by sending an RI DLPDU specifying the minimum delegation interval for token use required when the token is next returned.

NOTE An RI DLPDU, which is sent to terminate use of a token delegated by an ES DLPDU, indicates to the receiving LAS that sequence execution was incomplete and that another ES DLPDU, with a final-token-use subfield specifying CONTINUE, should be sent to the same DLSEP-address.

6.16.4.2 Additional actions required of a link-master class DLE

None.

6.16.4.3 Additional actions required of a bridge class DLE

a) Since every bridge class DLE has link master capability, any actions specified in 6.16.4.2 also apply to a bridge class DLE.

b) The DLE shall not forward the DLPDU.

6.16.4.4 Additional actions required of the current LAS DLE

If the DLE did not just send the DLPDU, then the DLE shall drop the scheduler token and deactivate its role as LAS; and shall inform local DL-management of the event.

6.17 Return token (RT) DLPDU

A RETURN TOKEN (RT) DLPDU is used to return the token delegated by a PT or ES DLPDU to the LAS and to indicate that no additional token delegation, to the DL-address of the last PT or ES DLPDU, is required.

NOTE The delegated token also may be returned by setting the final-token-use designator in any DLPDU to the value FINAL, and then ceasing use of the delegated token after transmitting that DLPDU.

6.17.1 Structure of the RT DLPDU

Table 28 – Structure of RT DLPDUs

Frame control
0011 0100

6.17.1.1 Frame control field

The frame control field shall specify the DLPDU's function.

6.17.1.2 Address field

The address field shall be null.

6.17.1.3 Parameters field

The parameters field shall be null.

6.17.1.4 User data field

The user data field shall be null.

6.17.2 Content of the RT DLPDU

The frame control field shall be encoded as specified in Table 28.

The address, parameters and user data fields shall be null.

6.17.3 Sending the RT DLPDU

An RT DLPDU results only from receipt of a PT or ES DLPDU. The RT DLPDU shall be sent as specified in 6.15.4 and 6.16.4.

6.17.4 Receiving the RT DLPDU

A received RT DLPDU shall be treated as follows by the receiving DLE.

6.17.4.1 Actions required of all DLEs

None.

6.17.4.2 Additional actions required of a link-master class DLE

None.

6.17.4.3 Additional actions required of a bridge class DLE

The following considerations apply:

a) Since every bridge class DLE has link master capability, any actions specified in 6.17.4.2 also apply to a bridge class DLE.
b) The DLE shall not forward the DLPDU.

6.17.4.4 Additional actions required of the current LAS DLE

When the scheduler token is not dominant on the local link, then

a) if the token being returned was delegated by a PT DLPDU, then the DLE shall set the value of the V(DTA)'th member of the remaining-token-holding-time-array, V(RTHA) (see 4.7.5.13) to zero;
b) the DLE shall assume that the scheduler token is again dominant on the local link and resume active operation as the LAS.

6.18 Request interval (RI) DLPDU

A REQUEST INTERVAL (RI) DLPDU is used to return a token delegated by a PT or ES DLPDU to the LAS and to indicate the minimal useful delegation interval required for the next delegation.

NOTE The delegated token also may be returned by setting the final-token-use designator in any DLPDU to the value FINAL, and then ceasing use of the delegated token after transmitting that DLPDU, or by sending an RT DLPDU.

6.18.1 Structure of the RI DLPDU

Table 29 – Structure of RI DLPDUs

Frame control	Parameters
0010 0000	DD-p

6.18.1.1 Frame control field

The frame control field shall specify the DLPDU's function.

6.18.1.2 Address field

The address field shall be null.

6.18.1.3 Parameters field

The parameter field shall be structured and encoded as specified in 7.10.

6.18.1.4 User data field

The user data field shall be null.

6.18.2 Content of the RI DLPDU

The frame control field shall be encoded as specified in Table 29. The address field shall be null.

The parameters field shall specify the minimum useful duration for the next delegation of the current token. The user data field shall be null.

6.18.3 Sending the RI DLPDU

An RI DLPDU results only from receipt of an ES or PT DLPDU. The RI DLPDU shall be sent as specified in 6.15.4 and 6.16.4.

6.18.4 Receiving the RI DLPDU

A received RI DLPDU shall be treated as follows by the receiving DLE.

6.18.4.1 Actions required of all DLEs

None.

6.18.4.2 Additional actions required of a link-master class DLE

None.

6.18.4.3 Additional actions required of a bridge class DLE

The following considerations apply:

a) Since every bridge class DLE has link master capability, any actions specified in 6.18.4.2 also apply to a bridge class DLE.
b) The DLE shall not forward the DLPDU.

6.18.4.4 Additional actions required of the current LAS DLE

When the scheduler token is not dominant on the local link, then the DLE shall

a) if the token being returned was delegated by a PT DLPDU, then the DLE shall set the value of the V(DTA)'th member of the remaining-token-holding-time-array, V(RTHA) (see 4.7.5.13), to account for the link capacity just used by the DLE;
b) if the token being returned was delegated by an ES or PT DLPDU, then the DLE shall forward the DD-parameter field of this DLPDU, together with V(DTA), to the LAS DLE's upper-level functions (see Clause 9) for attempted rescheduling;
c) the DLE shall assume that the scheduler token is again dominant on the local link and resume active operation as the LAS.

6.19 Claim LAS (CL) DLPDU

A CLAIM LAS (CL) DLPDU is used by an LM DLE to initialize the local link, or to recover from a prolonged period of silence on the local link indicative of the failure of the prior LAS.

6.19.1 Structure of the CL DLPDU

Table 30 – Structure of CL DLPDUs

Frame control	Source NODE-address
0000 0001	N

6.19.1.1 Frame control field

The frame control field shall specify

a) the DLPDU's function;
b) the length, number and type of DLPDU addresses.

6.19.1.2 Address field

The address field shall consist of an explicit source NODE DL-address.

6.19.1.3 Parameters field

The parameters field shall be null.

6.19.1.4 User data field

The user data field shall be null.

6.19.2 Content of the CL DLPDU

The frame control field shall be encoded as specified in Table 30. The address field shall be the NODE DL-address of the DLE which has detected the prolonged link inactivity, whose value is the sender's node-id, V(TN). The parameters and user data fields shall be null.

NOTE The primary purposes of the NODE DL-address in the CL DLPDU are

a) to permit improved detection of collisions during the LAS claiming procedure;

b) to identify the DLE which detected inactivity to observing DLEs, which may have utility in real open systems for identifying DLEs with particularly obstructive classes of PhE or DLE faults, sometimes characterized as "faulty transmitter" or "deaf receiver".

6.19.3 Sending the CL DLPDU

A CL DLPDU shall result from the detection in an LM DLE of a period of node-id slot-times, V(TN) × V(ST) octet-durations, of continuous link inactivity, implying failure of the previous LAS DLE, if any.

Upon such detection, the DLE shall immediately send one CL DLPDU, and shall again monitor the medium for node-id slot-times, V(TN) × V(ST) octet-durations, of continuous link inactivity after sending the first CL DLPDU.

If no activity is again heard (other than potentially a CL DLPDU with the NODE DL-address that was just sent), then the sending LM DLE shall immediately send a second identical CL DLPDU, after which the DLE shall choose a uniformly distributed random integer in the range zero to three, and shall monitor the medium for that many slot-times, random(0..3) × V(ST) octet-durations. The actual random choice shall be statistically independent of similar choices made by other DLEs.

NOTE This requirement for statistical independence minimizes the probability of repeatedly identical choices by identically constructed real devices.

If no activity is again heard (other than potentially the CL DLPDU with the NODE DL-address that was just sent), then the sending LM DLE shall activate its LAS functions, assume the scheduler token and commence active operation as the LAS.

6.19.4 Receiving the CL DLPDU

A received CL DLPDU shall be treated as follows by the receiving DLE.

6.19.4.1 Actions required of all DLEs

The following considerations apply:

a) The DLPDU shall be reported to the DLE's local DL-management.

b) If the receiving DLE holds a token on the local link, then the receiving DLE shall drop the token at once and shall inform local DL-management of the event.

6.19.4.2 Additional actions required of a link-master class DLE

None.

6.19.4.3 Additional actions required of a bridge class DLE

The following considerations apply:

a) Since every bridge class DLE has link master capability, any actions specified in 6.1.4.2 also apply to a bridge class DLE.
b) The DLE shall not forward the DLPDU.

6.19.4.4 Additional actions required of the current LAS DLE

The DLE shall drop the scheduler token and deactivate its role as LAS; and shall inform local DL-management of the event.

6.20 Transfer LAS (TL) DLPDU

A TRANSFER LAS (TL) DLPDU is used by the current LAS DLE to transfer the scheduler token and the role of LAS to another LM DLE on the local link. The TL DLPDU is sent only after having been requested by the addressed LM DLE, and may be rejected if the addressed DLE determines that its own copy of the local link's schedule is not current.

6.20.1 Structure of the TL DLPDU

Table 31 – Structure of TL DLPDUs

Frame control	Destination NODE-address	User data
0000 0110	N	o-SPDU

6.20.1.1 Frame control field

The frame control field shall specify

a) the DLPDU's function;
b) the DLPDU's implicit priority, which is URGENT;
c) the length, number and type of DLPDU addresses.

6.20.1.2 Address field

The address field shall consist of an explicit destination NODE DL-address.

6.20.1.3 Parameters field

The parameters field shall be null.

6.20.1.4 User data field

The user data field shall consist of an optional higher-level LAS-database-status SPDU of 64 octets or less.

6.20.2 Content of the TL DLPDU

The frame control field shall be encoded as specified in Table 31. The address field shall be the NODE DL-address of the DLE to which the LAS role is being transferred. The parameters field shall be null.

The user data field shall convey a higher-level LAS-data-base-status SPDU (see 9.3.2.3) whose maximum size is limited as specified in 6.20.1.4, or shall be null. When present, the contents of this SPDU shall reflect the most recent schedule construction activities, if any, prior to sending the TL DLPDU.

6.20.3 Sending the TL DLPDU

The LAS DLE shall send a TL DLPDU in response to the receipt of an explicit scheduling request received by the LAS from another LM DLE on the local link, requesting transfer of the scheduling-token and the associated LAS role. It is a protocol error to send a TL DLPDU to a non-requesting DLE.

NOTE 1 Such a request can only be made through a higher-level SPDU sent to the current LAS' DL-support functions. DL-management is not permitted to command such a transfer, though it may be able to prompt the requesting LM DLE to request the transfer directly.

If so instructed by DL-management, the LAS DLE may compare the NODE DL-address of its associated LM DLE with that of the requesting DLE and reject the transfer request when the requesting DLE has a higher NODE DL-address.

NOTE 2 This ordering improves predictability of which LM DLE will assume the LAS role on local links which do not employ dynamic determination of that LM DLE which is most capable of assuming the LAS role.

A TL DLPDU may be sent on the link when

a) the sending DLE, the LAS, holds a scheduler token which is the dominant token on the local link;

b) the LAS DLE has completed the last cycle of "circulating the token", and would otherwise initiate the next cycle of "circulating the token" by explicitly sending a PT DLPDU to the lowest-numbered NODE DL-address represented in V(LL);

   NOTE This restriction on transferring the role of LAS eliminates the need to transfer (reliably) the current value of the array V(RTHA) just before sending the TL DLPDU.

c) the remaining allocated duration of token usage before the next scheduled activity permits the LAS DLE to repeatedly send a TL DLPDU and monitor the local link for a period of immediate-response-recovery-delay slot-times, V(IRRD) × V(ST) octet-durations, for any immediate response from the addressed DLE.

After sending a TL DLPDU, the LAS DLE shall monitor the local link for a period of immediate-response-recovery-delay slot-times, V(IRRD) × V(ST) octet-durations, waiting for a response:

1) If a Ph-Data indication (see 4.4.4) reporting DATA is received, then the DLE shall
   – stop further monitoring;
   – then wait for receipt of the Ph-Data indication reporting END-OF-DATA-AND-ACTIVITY or END-OF-ACTIVITY;
   – then proceed as in 3).
2) If 1) does not apply, and the monitoring period expires, then
   i) if the bus is not active at that moment (that is, the last-received Ph-Data indication reported END-OF-ACTIVITY), then the DLE shall proceed as in 4).
   ii) if i) does not apply, implying that the link is still active at that moment (that is, the last-received Ph-Data indication reported START-OF-ACTIVITY), then
      α) For implementations that were not demonstrable on or before 31 December 1995, the DLE shall monitor the local link for a period of one additional slot-time, V(ST) octet-durations, waiting for a Ph-Data indication:
         A) If a Ph-Data indication reporting DATA is received, then the DLE shall proceed as in 1).
         B) If a Ph-Data indication reporting END-OF-ACTIVITY is received, and A) does not apply, then the DLE shall stop further monitoring and proceed as in 4).

C) If neither A) nor B) applies, and the monitoring period expires before a Ph-Data indication is received, then the DLE shall

– wait for receipt of the Ph-Data indication reporting END-OF-DATA-AND-ACTIVITY or END-OF-ACTIVITY;

– then proceed as in 3).

β) Implementations that were demonstrable on or before 31 December 1995 alternatively may just proceed as in α).C) when neither α).A) nor α).B) applies.

3) If the link activity was sufficient to infer that DATA was received as in 1), 2)ii)α)A) or 2)ii)α)C), then

i) If the link activity did not result in a DLPDU, then the LAS DLE shall drop its scheduler token and shall continue or resume monitoring the local link until either 3)i)A) or 3)i)B) occurs.

NOTE This monitoring provides for recovery of the LAS role when the transfer is unsuccessful.

A) If the monitoring DLE observes a DLPDU (with correct FCS) then the monitoring DLE shall cease monitoring for LAS role recovery, and shall inform local DL-management of the event.

B) If the monitoring DLE observes a period of 15 slot-times, 15 × V(ST) octet-durations, of continuous non-activity on the local link, then the monitoring DLE shall

– re-assume the scheduler token;

– inform local DL-management of the event;

– resume active operation as the LAS, and commence transmission on the link.

ii) If the link activity resulted in a DLPDU (with correct FCS) then

A) if the received DLPDU is an SR DLPDU, specifying a reason of “failure – LAS transfer rejected”, then the monitoring DLE shall

– re-assume the scheduler token;

– inform local DL-management of the event;

– resume active operation as the LAS, and commence transmission on the link.

B) else if the received DLPDU is any other DLPDU, then the monitoring DLE shall cease monitoring for LAS role recovery, and shall inform local DL-management of the event.

4) If there was no link activity, or the link activity was not sufficient to infer that DATA was received as in 2)i) or 2)ii)α)B), then the LAS DLE may retry re-sending the TL DLPDU, up to the number of times specified as the maximum-retry-count, V(MRC) (see 4.7.1.5), of the local link.

If all attempted retries are unsuccessful, then the LAS DLE shall

– retain the scheduler token and continue active operation as the LAS;

– inform local DL-management of the event;

– start the next transmission within token-recovery-delay, P(TRD) × V(ST) octet-durations, of the beginning of the current period of link non-activity.

6.20.4 Receiving the TL DLPDU

A received TL DLPDU shall be treated as follows by the receiving DLE.

6.20.4.1 Actions required of all DLEs

The DLPDU shall be reported to the DLE’s local DL-management.

6.20.4.2 Additional actions required of a link-master class DLE

If the destination DL-address specified by the DLPDU designates the DLE's NODE DL-address, then

a) If

1) the receiving DLE is not awaiting receipt of the TL DLPDU;
2) the receiving DLE cannot execute the existing schedule, either due to schedule complexity or length, or because the receiving DLE cannot meet the schedule overhead constraint, V(MSO) (see 4.7.5.6), built into the existing schedule; or
3) if the schedule construction and live-list information conveyed in the LAS-database-status SPDU within the received TL DLPDU indicates that the receiving DLE does not have a current copy of either the schedule, or the live-list, or both,

then the receiving DLE shall reply with an SR DLPDU within a period of maximum-response-delay slot-times, V(MRD) × V(ST) octet-durations, as measured at the receiving DLE, with a status of "failure —LAS transfer rejected".

b) Otherwise, when a) does not apply, then the receiving DLE shall assume the scheduler token, activate its LAS functions, and re-commence operation as the LAS.

6.20.4.3 Additional actions required of a bridge class DLE

The following considerations apply:

a) Since every bridge class DLE has link master capability, any actions specified in 6.20.4.2 also apply to a bridge class DLE.
b) The DLE shall not forward the DLPDU.

6.20.4.4 Additional actions required of the current LAS DLE

If the DLE did not just send the DLPDU, then the DLE shall drop the scheduler token and deactivate its role as LAS; and shall inform local DL-management of the event.

6.21 Wakeup (WK) DLPDU

A WAKEUP (WK) DLPDU is used by the current LAS DLE to stimulate a class B or class C fractional duty cycle (FDC) DLE on the local link (see 3.3.32) to transition to a fully operational state, capable of direct communication.

Transmission of the WK DLPDU usually occurs as a result of previous explicit scheduling, either by the addressed FDC DLE or by DL-management. However, the LAS DLE, which must also be a bridge DLE to support FDC DLEs, may transmit the WK DLPDU on its own initiative, typically when its bridge-function's queue of DLPDUs waiting for transmission to the FDC DLE exceeds some internal threshold.

6.21.1 Structure of the WK DLPDU

Table 32 – Structure of WK DLPDUs

Frame control	Destination NODE-address
0000 0000	N

6.21.1.1 Frame control field

The frame control field shall specify

a) the DLPDU's function;

b) the DLPDU's implicit priority, which is URGENT;

c) the length, number and type of DLPDU addresses.

6.21.1.2 Address field

The address field shall consist of an explicit destination NODE DL-address.

6.21.1.3 Parameters field

The parameters field shall be null.

6.21.1.4 User data field

The user data field shall be null.

6.21.2 Content of the WK DLPDU

The frame control field shall be encoded as specified in Table 32. The address field shall be the NODE DL-address of the DLE which is to receive the WK DLPDU. The parameters and user data fields shall be null.

NOTE The specific encoding of the WK DLPDU minimizes the amount of active DLPDU-recognition circuitry required within a "sleeping" FDC DLE.

6.21.3 Sending the WK DLPDU

The LAS DLE shall send a WK DLPDU when requested by prior explicit scheduling. The link capacity used for sending the WK DLPDU shall be deducted (to the extent possible) from the remaining allocated duration of scheduler token usage for scheduled traffic, which is V(MST) × V(TTRT) octet-durations, within the current cycle of "circulating the token".

A bridge DLE may also send a WK DLPDU when so requested by the DLE's own bridge-forwarding functions when the DLE holds a scheduler token or delegated token which is the dominant token on the local link, and when the remaining allocated duration of token usage prior to the next scheduled activity permits completion of the DLPDU's transmission prior to expiration of the token.

6.21.4 Receiving the WK DLPDU

A received WK DLPDU shall be treated as follows by the receiving DLE.

6.21.4.1 Actions required of all DLEs

If the destination DL-address specified by the DLPDU designates the DLE's NODE DL-address, then the DLE shall activate the remainder of its DLE functions and prepare for active communication on the local link.

NOTE Receipt of this DLPDU has no impact on non-FDC DLEs.

6.21.4.2 Additional actions required of a link-master class DLE

None.

6.21.4.3 Additional actions required of a bridge class DLE

The following considerations apply:

a) Since every bridge class DLE has link master capability, any actions specified in 6.21.4.2 also apply to a bridge class DLE.

b) The DLE shall not forward the DLPDU.

6.21.4.4 Additional actions required of the current LAS DLE

None.

6.22 Idle (IDLE) DLPDU

An IDLE DLPDU is used by the current token holder to place a DLPDU on the local link which is ignored by all DLEs, other than for its impact on each DLE's sense of link activity, but which may convey information to, or trigger, a data link analysis device.

6.22.1 Structure of the IDLE DLPDU

Table 33 – Structure of IDLE DLPDUs

Frame control	User data
0001 0F10	o-DLSDU

6.22.1.1 Frame control field

The frame control field shall specify

a) the DLPDU's function;

b) whether or not transmission of the current DLPDU terminates use of a delegated token.

6.22.1.2 Address field

The address field shall be null.

6.22.1.3 Parameters field

The parameters field shall be null.

6.22.1.4 User data field

The data field shall consist of a single DLSDU whose maximum size is the maximum permitted for an URGENT priority DLPDU. This data field may be null.

6.22.2 Content of the IDLE DLPDU

The frame control field shall be encoded as specified in Table 33. The address and parameter fields shall be null.

The contents of the user data field, possibly null, is not specified by this standard.

6.22.3 Sending the IDLE DLPDU

An IDLE DLPDU may be sent on the link when the sending DLE holds a scheduler token or delegated token which is the dominant token on the local link, and when the remaining allocated duration of token usage, C(RD), permits completion of the IDLE DLPDU's transmission prior to expiration of the token.

NOTE The LAS DLE uses the IDLE DLPDU to fill otherwise unused link capacity until the time of the next scheduled link activity, thereby preventing other DLEs from inferring the LAS DLE's failure from the lack of activity on the link. To facilitate this use, the IDLE DLPDU has a minimum size of one octet.

A local management request may result in the transmission of an IDLE DLPDU.

If the DLE holds a delegated token, and this DLPDU is the final DLPDU which needs to be sent on this use of the delegated token, then the DLE may set the final-token-use subfield of the IDLE DLPDU to the value FINAL; else that subfield shall have the value NOT-FINAL.

6.22.4 Receiving the IDLE DLPDU

A received IDLE DLPDU shall be treated as follows by the receiving DLE.

6.22.4.1 Actions required of all DLEs

The DLE shall ignore the DLPDU on receipt, except for possible local management reporting or special instrumentation purposes.

6.22.4.2 Additional actions required of a link-master class DLE

None.

6.22.4.3 Additional actions required of a bridge class DLE

The following considerations apply:

a) Since every bridge class DLE has link master capability, any actions specified in 6.22.4.2 also apply to a bridge class DLE.
b) The DLE shall not forward the DLPDU.

6.22.4.4 Additional actions required of the current LAS DLE

If the final-token-use subfield of the received DLPDU has the value FINAL, then the LAS DLE shall

a) assume that the current use of the delegated token is terminated as if the token had been returned by an RT (see 6.17) DLPDU;
b) assume that the scheduler token is again dominant on the local link and resume active operation as the LAS.

6.23 Spare DLPDUs

A SPARE DLPDU is reserved for future standard use, and may not be sent by any DLE.

6.23.1 SPARE DLPDU code points

Table 34 – Assumed structure of undefined (spare) DLPDUs

format	**Frame control**
1	0000 0x11
2	0010 0x01
3	00xx 1xxx
4	0100 xxxx
5	101x xx00

A SPARE DLPDU shall consist of a frame control field. Each spare FC code point is reserved for future standard use.

The DLPDU format and contents of spare code points are subject to change if and when used in a revision of this standard. Until that time they shall not be used.

6.23.1.1 Frame control field

The frame control field shall specify the DLPDU's function – reserved spare. No DLPDU priority is implied.

6.23.1.2 Address field

The address field shall be null.

6.23.1.3 Parameters field

The parameters field shall be null.

6.23.1.4 User data field

The user data field shall be null.

6.23.2 Content of a SPARE DLPDU

The frame control field shall be encoded as specified in Table 34. The address, parameters and user data fields shall be null.

6.23.3 Sending a SPARE DLPDU

SPARE DLPDUs may not be sent or forwarded.

6.23.4 Receiving a SPARE DLPDU

A received SPARE DLPDU shall be treated as follows by the receiving DLE.

6.23.4.1 Actions required of all DLEs

The DLE shall ignore the DLPDU on receipt, except for possible local management reporting or special instrumentation purposes.

6.23.4.2 Additional actions required of a link-master class DLE

None.

6.23.4.3 Additional actions required of a bridge class DLE

The following considerations apply:

a) Since every bridge class DLE has link master capability, any actions specified in 6.23.4.2 also apply to a bridge class DLE.
b) The DLE shall not forward the DLPDU.

6.23.4.4 Additional actions required of the current LAS DLE

None.

6.24 Reserved (not to be used) DLPDUs

A RESERVED (NOT TO BE USED) DLPDU is reserved for future standard use, and may not be sent by any DLE.

6.24.1 RESERVED (NOT TO BE USED) DLPDU code points

Table 35 – Assumed structure of RESERVED (NOT TO BE USED) DLPDUs

format	**Frame control**
1	0000 0010
2	0000 010x
3	0010 001x
4	0010 0100
5	0011 0000
6	0101 1x00
7	1001 1x00
8	110x 1x00

The FC code point for a RESERVED (NOT TO BE USED) DLPDU shall not be used; these code points are reserved to simplify decoding of the FC octets of other defined DLPDUs. The set of such reserved code points is shown in Table 35.

The frame control field shall specify the DLPDUs function – reserved (not to be used). It shall not be used.

NOTE Since these FC code points are reserved to simplify DLE FC-code-point decoding, their recognition as being distinct from structurally related FC code points cannot be mandated.

6.24.2 Content of a RESERVED (NOT TO BE USED) DLPDU

No content is defined, since DLPDUs with these code points may not be originated.

6.24.3 Sending a RESERVED (NOT TO BE USED) DLPDU

DLPDUs specifying RESERVED (NOT TO BE USED) FC code points shall not be originated. However, they may be forwarded by a bridge if they were not discarded upon receipt.

6.24.4 Receiving a RESERVED (NOT TO BE USED) DLPDU

If the receiving DLE distinguishes between RESERVED (NOT TO BE USED) and other DLPDUs, then the DLE should ignore the RESERVED (NOT TO BE USED) DLPDU on receipt (except for possible local management reporting or special instrumentation purposes). However, the ability to make such a distinction and to perform such discarding is not required.

7 DLPDU-parameter structure and encoding

Boolean variables are all encoded with a common representation:

0: FALSE

1: TRUE

The 3-bit DL protocol version number (VVV), which is used in a number of the DLPDU parameters and SPDUs (see clause 11), shall have the value one (1).

7.1 Structure and encoding of EC-PARAMETERS

An EC DLPDU is used to establish a peer DLC between two DLS-users, or a multi-peer DLC between a publishing DLS-user and subscribing DLS-users.

Table 36 – Structure of an EC DLPDU's parameters

DLC basic attributes	DLC attributes when sending	DLC attributes when receiving

The connection parameters (EC-parameters) field, which partitions roughly as shown in Table 36, shall specify the parameters for the proposed DLC:

a) The first two octets, ordered as shown in Table 37 and Table 38, specify basic attributes of the DLC, beginning with the DL protocol version number:

Table 37 – EC-parameters: 1st octet

Reply request	Publisher-DLCEP-address reuse discriminator			Path diversity	DL-protocol version		
R	NNN			Q	VVV		
7	6	5	4	3	2	1	0

1) a 1-bit reply-request subfield (R), specifying whether a reply is requested (=1) or not (=0);

NOTE This field is provided to ensure proper action independent of any state of the receiving DLCEP.

2) a 3-bit publisher-DLCEP-address reuse-discriminator subfield (NNN), designating the current set of EC-parameters associated with a PUBLISHER DLCEP-address

i) encoded as zero, when the sender's DLCEP-class (CC) is not PUBLISHER;

ii) assigned by the DLE, when the sender's DLCEP-class (CC) is PUBLISHER

NOTE The DLE assigns a value to this subfield whenever it receives a request to initiate a new PUBLISHER DLCEP from either a local DLS-user in a DL-CONNECT request primitive, or from a remote DLS-user in an EC DLPDU. The choice of value can be random, or can be based on knowledge of recently used values which the DLE avoids.

This subfield is used to discriminate between two different DLCs using the same publisher-DLCEP-address. This could happen when the first DLC had been disconnected and the publishing DLE used the same DLCEP-address for establishing the second DLC, which is different from the first DLC. If a subscriber to the first DLC did not receive the DC DLPDU for that DLC, it could use this field of the just received EC DLPDU to determine that the received EC DLPDU is for a different DLC. Therefore, if the publisher-DLCEP should send an EC DLPDU for the same DLC, it does not change this subfield.

3) a one-bit DLL path-diversity subfield (Q), encoded as:

0: any-path;

1: this-path – use the path on which this EC DLPDU was received;

NOTE This subfield provides a means by which a DLE can restrict all communications at a DLCEP to a specific DL-path, thereby providing a means for testing the specific DL-path. The means by which the THIS-PATH value is selected, and by which the actual path employed is chosen, is a DLE-local issue.

4) a 3-bit DL protocol version number (VVV), whose value is specified at the beginning of Clause 7;

Table 38 – EC-parameters: 2nd octet

Sender's DLCEP class		DL-priority		DL-address size		DLPDU authentication	
CC		PP		SS		XX	
7	6	5	4	3	2	1	0

5) a two-bit sender's DLCEP-class subfield (CC), encoded as:

00: reserved for future standard use;

01: PEER DLCEP;

10: PUBLISHER DLCEP;

11: SUBSCRIBER DLCEP;

6) a two-bit DLL priority subfield (PP), encoding the DLL priority as (see 5.2.1.3):

00: reserved for future standard use;

01: URGENT (high) priority;

10: NORMAL (medium) priority;

11: TIME-AVAILABLE (low) priority;

7) a two-bit DLL address size subfield (SS), encoding the proposed DLL address size (see 5.2.1.1) as:

00: VERY-SHORT – addresses are omitted where possible;

01: SHORT;

10: LONG;

11: reserved for future standard use;

8) a two-bit DLPDU-authentication subfield (XX), encoding the required DLPDU authentication (see 4.2.4) as:

00: ORDINARY;

01: reserved for future standard use;

10: SOURCE;

11: MAXIMAL;

b) The next four octets, ordered as shown in Table 39 and Table 40, specify the maximum confirm delay, in units of 1 ms, for the sender's DL-CONNECT, DL-RESET and DL-SUBSCRIBER-QUERY request primitives, and DL-DATA request primitives, respectively. The DLS-user value UNLIMITED shall be encoded as the value $FFFF_{16}$ (all ones).

Table 39 – EC-parameters: 3rd and 4th octets

Sender's Maximum Confirm Delay for DL-CONNECT, DL-RESET and DL-SUBSCRIBER-QUERY requests															
MCD_CRS															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

and

Table 40 – EC-parameters: 5th and 6th octets

Sender's Maximum Confirm Delay for DL-DATA requests															
MCD_D															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

NOTE Each two-octet maximum confirm delay is sent most-significant-octet first.

c) The next four octets, ordered as shown in Table 41 through Table 43, specify proposed attributes for the sender's sending direction of data flow:

Table 41 – EC-parameters: 7th octet

Sender's sending DLC attributes							
DLCEP data delivery features		Residual activity	Queue / buffer	Maximum window size			
TT_S		A_S	B_S	$WWWW_S$			
7	6	5	4	3	2	1	0

1) a two-bit sending DLCEP data-delivery features subfield (TT_S), encoded as:

   00: UNORDERED DLC;

   01: ORDERED DLC;

   10: DISORDERED DLC;

   11: CLASSICAL DLC;

2) a one-bit residual activity subfield (A_S), indicating whether residual activity in the specified (sender-to-receiver) direction will be provided when there are no unacknowledged DLSDUs, encoded as a Boolean;

   NOTE Residual activity is required when MAXIMAL DLPDU-authentication is specified, or upon DLS-user or DL-management direction. Other methods of requesting residual activity are for future study.

3) a one-bit buffer/queue subfield (B_S), indicating whether the source in the specified (sender-to-receiver) direction is a buffer (=1) or a queue (=0);

4) a four-bit window-size subfield ($WWWW_S$), indicating the maximum number of previously sent DLSDUs which will be available for retransmission, and encoded as zero when transmission in that direction is not permitted on the DLC;

Table 42 – EC-parameters: 8th octet

Sender's sending DLC attributes (cont.)							
Basic-DLC-parameters format			2-way data exchange	reserved	Timeliness included	Time stamp format	
FFF_S			E_S	0	G_S	HH_S	
7	6	5	4	3	2	1	0

5) a three-bit format subfield (FFF_S), giving the desired format of the basic-DLC-parameters portion of the Status-Data-parameters (SD-parameters) in the specified direction, encoded as a three-bit integer:

   0: format A – [7.4.2.1A)];

   1: format B – [7.4.2.1B)];

   2: format C – [7.4.2.1C)];

   3: reserved for future standard use;

   4: format D – [7.4.2.1D)];

   5: format E – [7.4.2.1E)];

   6: format F – [7.4.2.1F)];

   7: format G – [7.4.2.1G)].

6) a one-bit 2-way data exchange subfield (E_S), indicating whether DLSDU data may be sent in the sender-to-receiver direction by ED DLPDUs, encoded as a Boolean;

7) a one-bit subfield reserved for future standards use, encoded as zero;

8) a one-bit timeliness-included subfield (G_S), indicating the presence of a timeliness parameter (see 7.4.2.1B)3) and 7.4.2.1C)5)) associated with transmitted DLSDUs in transmitted DLPDUs, encoded as a Boolean;

9) a two-bit time-stamp-format subfield (HH_S), indicating whether a DL-time-stamp (see 7.4.2.2) is included in SD-parameters which accompany transmitted DLSDUs, encoded as:

00: format J – [7.4.2.2J)], which is a null field with no time stamp;

01: format K – [7.4.2.2K)], which is a two-octet field with a time stamp period of over 2 s;

10: format L – [7.4.2.2L)], which is a three-octet field with a time stamp period of over 8 min;

11: format M – [7.4.2.2M)], which is a six-octet field with a time stamp period of over 100 years;

Table 43 – EC-parameters: 9th and 10th octets

Sender's sending DLC attributes (cont.)															
Maximum DLSDU size															
$M...M_S$															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

10) a two-octet subfield ($M...M_S$), specifying the maximum size DLSDU that can be sent on the DLC.

NOTE 1 The two-octet maximum size is transmitted most-significant-octet first.

NOTE 2 The range for each parameter is dependent on the DLC's priority, and can range between zero and 16 times the maximum data length specified in 5.2.4 for the corresponding priority, inclusively.

d) The next four octets, identical in form to those specified in c) and ordered as shown in Table 44 through Table 46, specify proposed attributes for the sender's receiving direction of data flow:

Table 44 – EC-parameters: 11th octet

Sender's receiving DLC attributes							
DLCEP data delivery features		**Residual activity**	**Queue / buffer**	**Maximum window size**			
TT_R		A_R	B_R	$WWWW_R$			
7	6	5	4	3	2	1	0

1) a two-bit receiving DLCEP data delivery features subfield (TT_R), encoded as specified in c)1);

2) a one-bit residual activity subfield (A_R), indicating whether residual activity in the specified (receiver-to-sender) direction is required, encoded as specified in c)2);

NOTE See Note after c)2).

3) a one-bit buffer/queue subfield (B_R), indicating whether the sink in the specified (receiver-to-sender) direction is a buffer or a queue, encoded as specified in c)3);

4) a four-bit window-size subfield ($WWWW_R$), indicating the maximum number of unacknowledged DLSDUs which can be usefully received;

Table 45 – EC-parameters: 12th octet

Sender's receiving DLC attributes (cont.)							
Basic-DLC-parameters format			**2-way data exchange**	**reserved**	**Timeliness included**	**Time stamp format**	
FFF_R			E_R	0	G_R	HH_R	
7	6	5	4	3	2	1	0

5) a three-bit format subfield (FFF_R), giving the desired format of basic-DLC-parameters portion of the SD-parameters in the specified direction, encoded as specified in c)5).

6) a one-bit 2-way data exchange subfield (E_R), indicating whether DLSDU data may be sent in the receiver-to-sender direction by ED DLPDUs, encoded as a Boolean;

7) a one-bit subfield reserved for future standards use, encoded as zero;

8) a one-bit timeliness-included subfield (G_R), indicating the presence in received DLPDUs of a timeliness parameter (see 7.4.2.1b)3) and 7.4.2.1c)5)) associated with received DLSDUs, encoded as a Boolean;

9) a two-bit time-stamp-format subfield (HH_R), indicating whether a DL-time-stamp is included in SD-parameters which accompany received DLSDUs, encoded as specified in c)9);

Table 46 – EC-parameters: 13th and 14th octets

Sender's receiving DLC attributes (cont.)															
Maximum DLSDU size															
$M...M_R$															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

10) a two-octet subfield ($M...M_R$), specifying the maximum size DLSDU that can be received on the DLC.

NOTE See Notes following c)10).

7.2 Structure and encoding of DC-PARAMETERS

A DISCONNECT CONNECTION (DC) DLPDU is used to disconnect an existing or proposed DLC.

The Disconnect-Connection-parameters (DC-parameters) field shall specify the desired DLC-support action and reason, together with any associated operational parameters. This parameter field is two octets, ordered as shown in Table 47 through Table 49.

Table 47 – DC-parameters and RC-parameters: 1st octet

Reply request	**reserved**				**DL-protocol version**		
R	0000				VVV		
7	6	5	4	3	2	1	0

a) a one-octet subfield, coded as in Table 47, consisting of

1) a one-bit subfield (R) specifying whether a reply is requested (=1) or not (=0);

NOTE This field is provided to ensure proper action independent of any state of the receiving DLCEP.

2) a four-bit subfield reserved for future standards use and encoded as zero;

3) a 3-bit DL protocol version number (VVV), whose value is specified at the beginning of Clause 7;

Table 48 – DC-parameters and RC-parameters: 2nd octet

Reason for DLPDU							
Reason							
7	6	5	4	3	2	1	0

b) a one-octet subfield, specifying the reason for the requested DLC support action, based on IEC 61158-3-1, 8.6.2, and coded in hexadecimal as specified in Table 49. All unused reason codes in Table 49 in the hexadecimal range 40 to 7F are reserved for future standards use; the other codes in the range 00 to 3F may be used as desired, and can be interpreted as reason unspecified (by this standard).

7.3 Structure and encoding of RC-PARAMETERS

A RESET CONNECTION (RC) DLPDU is used to reset or disconnect an existing or proposed DLC.

The Reset-Connection-parameters(RC-parameters) field shall specify the desired DLC-support action and reason, together with any associated operational parameters.

This parameter field is four octets, ordered as shown in Table 47 through Table 52:

a) A one-octet subfield, coded as specified in 7.2a).

Table 49 – Disconnect reasons

Code	Reason for disconnect	Reason class	Mapping to IEC 61158-3-1
00	user-originated disconnection – normal condition	user-originated	8.6.2.5b)1)
02	user-originated disconnection – abnormal condition	disconnection	8.6.2.5b)2)
1E	user-originated disconnection or connection rejection – reason unspecified	(00..1F)	8.6.2.5b)7)
20	user-originated connection rejection – connection not authorized, permanent condition	user-originated	8.6.2.5b)4)
21	user-originated connection rejection – unacceptable QoS, permanent condition	connection	8.6.2.5b)4)
22	user-originated connection rejection – non-QoS reason, permanent condition	rejection	8.6.2.5b)5)
24	user-originated connection rejection – transient condition	(20..3F)	8.6.2.5b)6)
40	provider-originated disconnection – incorrect DLCEP pairing, permanent condition		8.6.2.5a)1)
41	provider-originated disconnection – wrong publisher-DLCEP-address reuse-discriminator, permanent condition		8.6.2.5a)1)
42	provider-originated disconnection – other permanent condition	provider-originated	8.6.2.5a)1)
43	provider-originated disconnection – wrong DLPDU format or parameters, permanent condition	disconnection	8.6.2.5a)1)
44	provider-originated disconnection – wrong DLSDU size, permanent condition	(40..5F)	8.6.2.5a)1)
45	provider-originated disconnection – transient condition		8.6.2.5a)2)
46	provider-originated disconnection – timeout		8.6.2.5a)3)
5E	provider-originated disconnection or connection rejection – reason unspecified		8.6.2.5a)12)
60	provider-originated connection rejection – DL(SAP) address unknown		8.6.2.5a)5)
62	provider-originated connection rejection – DLSAP unreachable, permanent condition	provider-originated	8.6.2.5a)6)
64	provider-originated connection rejection – DLSAP unreachable, transient condition	connection	8.6.2.5a)7)
65	provider-originated connection rejection – inconsistent DLCEP state, permanent condition	rejection	8.6.2.5a)8)
66	provider-originated connection rejection – QoS unavailable, permanent condition	(60..7D)	8.6.2.5a)8)
68	provider-originated connection rejection – QoS unavailable, transient condition		8.6.2.5a)9)
7E	disconnection or connection rejection, unknown origin – reason unspecified	unknown origin or type (7E..7F)	8.6.2.5c)

b) A one-octet subfield, specifying the reason for the requested DLC support action, based on IEC 61158-3-1, 8.7.3.5 and IEC 61158-3-1, 8.7.3.6, and coded in hexadecimal as specified in Table 50. All unused reason codes in Table 50 in the hexadecimal range C0 to FF are reserved for future standards use; the other codes in the range 80 to BF may be used as desired, and can be interpreted as reason unspecified (by this standard).

Table 50 – Reset reasons

Code	Reason for reset	Reason class	Mapping to IEC 61158-3-1
80	user-originated reset – resynchronization after user timeout		8.7.3.6b)1)
82	user-originated reset – resynchronization after user-detected user-state inconsistencies	user-originated reset	8.7.3.6b)2)
9E	user-originated reset – reason unspecified	(80..BF)	8.7.3.6b)3)
C0	provider-originated reset – resynchronization after activation of a DL-management-established DLCEP		8.7.3.6a)1)
C2	provider-originated reset – resynchronization after timeout		8.7.3.6a)2)
C4	provider-originated reset – resynchronization after maximum number of retransmission requests or attempts	provider-originated reset	8.7.3.6a)3)
C6	provider-originated reset – resynchronization after detected sequence number error	(C0..FD)	8.7.3.6a)4)
C8	provider-originated reset – resynchronization after other detected DLCEP state inconsistencies		8.7.3.6a)5)
FC	provider-originated reset – reason unspecified		8.7.3.6a)7)
FE	reset, unknown origin – reason unspecified	unknown origin (FE..FF)	8.7.3.6a)1)

Table 51 – RC-parameters: 3rd octet

Modulus number preceding next DLSDU to be sent, if any							
any value			NDS mod 2^5				
7	6	5	4	3	2	1	0

c) A one-octet subfield, specifying, in its low-order bits, the five low-order bits of the sequence number NDS preceding the sequence number of the next DLSDU to be sent. The high-order three bits of this octet may have any value (for example, the next higher-order bits of NDS) and shall be ignored on reception (see Table 51).

Table 52 – RC-parameters: 4th octet

Modulus number of last complete DLSDU received, if any							
any value			NDR mod 2^5				
7	6	5	4	3	2	1	0

d) A one-octet subfield, specifying, in its low-order bits, the five low-order bits of the sequence number NDR of the last DLSDU received, if any. The high-order three bits of this octet may have any value (for example, the next higher-order bits of NDR) and shall be ignored on reception (see Table 52).

7.4 Structure and encoding of SD-Parameters

7.4.1 SD-Parameters in DLPDUs addressed to a DL(SAP)-address

A COMPEL ACKNOWLEDGMENT (CA) DLPDU addressed to a DL(SAP)-address is used as the first phase of the remotely confirmed unitdata transfer service to transfer (or retransfer) a limited amount of transparent user data from one DLS-user to another DLS-user and to request the status of the transfer from the receiving DLE.

A COMPEL DATA (CD) DLPDU addressed to a DL(SAP)-address is used as the first phase of the unitdata exchange service to request the immediate transfer (or retransfer) of a limited amount of transparent user data from one DLS-user to another DLS-user. A CD DLPDU always contains a non-null SD-parameters field.

An EXCHANGE DATA (ED) DLPDU addressed to a DL(SAP)-address is used as the first phase of the unitdata exchange service to transfer (or retransfer) a limited amount of transparent user data from one DLS-user to another DLS-user, and to request the immediate transfer (or retransfer) of a limited amount of transparent user data from that second DLS-user to the first DLS-user. An ED DLPDU always contains a non-null SD-parameters field.

A DATA (DT) DLPDU addressed to a DL(SAP)-address is used

1) to transfer a limited amount of transparent user data from one requesting DLS-user to one or more other DLS-users without establishing or later releasing a DLC; or
2) as the second phase of the unitdata exchange service or remotely-confirmed unitdata-transfer service to acknowledge the transfer of such data without establishing or later releasing a DLC.

The DLPDUs of 1) always contain a null SD-parameters field; the DLPDUs of 2) always contain a non-null SD-parameters field.

NOTE These DLPDUs are also used within the context of a DLC to provide DL-DATA services.

The various SD-parameter formats, and their potential applicability to the CA, CD, DT and ED DLPDUs addressed implicitly or explicitly to a DLSAP-address, shown in Table 53, are:

Table 53 – Structure of connectionless-mode CA, CD, DT and ED DLPDUs

Connectionless-mode DLPDUs					
format	**Frame control**	**d-DLSAP-address**	**s-DLSAP-address**	**Parameters**	**User data**
1	CA	[HL.]N.S	[HL.]N.S	format R	DLSDU
2	CD	[HL.]N.S	[HL.]N.S	format R	—
3	ED	[HL.]N.S	[HL.]N.S	format R	DLSDU
4	DT 1	[HL.]N.S	[HL.]N.S	format P	DLSDU
5	DT 2	[HL.]N.S	—	format U	o-DLSDU
6	DT 3	implied PSA	—	format U	—

format	**SD-parameters**		
	Non-DLC-parameters		
P	— (null) —		
R	ZZ	PP	N(LTI)
U	XXXX		N(LTI)

a) The **null** format, format P, of length zero octets, is used for the unitdata transfer service. It is always used in DT DLPDUs which have two explicit addresses (see 6.7.1 formats 1L and 1S).
b) The **initiator** format, format R (see Table 54), of length one octet, is used for the first phase of the two-phase transaction required for unitdata-exchange and remotely-confirmed unitdata-transfer services. It is always used in CA, CD and ED DLPDUs, all of which always have two explicit addresses (see 6.4.1, 6.5.1 and 6.6.1 formats 1L and 1S).

Table 54 – Short format SD-parameters for connectionless transaction initiators

reserved		DLSDU priority and presence		Initiator's transaction index			
ZZ		PP		N(LTI)			
7	6	5	4	3	2	1	0

The octet comprises

1) a two-bit reserved subfield, reserved for future standards use, which shall be zero;
2) a two-bit DLSDU-priority-and-presence subfield, specifying the presence and DLL priority of the accompanying DLSDU (see 5.2.1.3), encoded as

   00: no accompanying DLSDU;

   01: URGENT (high) priority accompanying DLSDU;

   10: NORMAL (medium) priority accompanying DLSDU;

   11: TIME-AVAILABLE (low) priority accompanying DLSDU;

3) a four-bit transaction-index subfield, specifying the transaction index assigned during CA, CD or ED formation and transmission as specified in 6.4.3, 6.5.3 and 6.6.3.

c) The **responder** format, format U (see Table 55), of length one octet, is used for the second phase of the two-phase transaction required for unitdata-exchange and remotely-confirmed unitdata-transfer services. It is always used in DT DLPDUs which do not have an explicit source address (see 6.7.1 formats 2L, 2S and 4):

Table 55 – Short format SD-parameters for connectionless responders

Status				Initiator's transaction index			
XXXX				N(LTI)			
7	6	5	4	3	2	1	0

1) a four-bit status subfield, encoded as specified in Table 56;
2) a four-bit transaction-index subfield, identical to b)3).

Table 56 – Reply status for unitdata-acknowledgment and exchange-unitdata-reply DT DLPDUs

Short name	Definition	Hexadecimal coding	Mapping to IEC 61158-3-1
OK	success – no reply data available at responder or service does not provide this information	0	9.5.1.2.5a) 9.5.2.2.5a)
OK_U	success – URGENT priority reply data available at responder	1	9.5.2.2.5a)
OK_N	success – NORMAL priority reply data available at responder	2	9.5.2.2.5a)
OK_TA	success – TIME-AVAILABLE priority reply data available at responder	3	9.5.2.2.5a)
RR	failure —resource limitation in responder – no reply data available at responder or service does not provide this information (note 1)	4	9.5.1.2.5j) 9.5.2.2.5m)
RR_U	failure —resource limitation in responder – URGENT priority reply data available at responder (note 1)	5	9.5.2.2.5c)
RR_N	failure —resource limitation in responder – NORMAL priority reply data available at responder (note 1)	6	9.5.2.2.5c)
RR_TA	failure —resource limitation in responder – TIME-AVAILABLE priority reply data available at responder (note 1)	7	9.5.2.2.5c)
RF	failure —fault in responder	8	9.5.1.2.5k) 9.5.2.2.5e)
RI	failure —responder DL(SAP)-role incompatible with this DLPDU (note 2)	9	9.5.1.2.5g) 9.5.2.2.5h)
RA	failure – response restricted to a different peer DLSAP-address (note 3)	A	9.5.2.2.5g)
—	reserved for compatibility with the SR DLPDU's reply status – not available for other use	B	
BF	failure – fault in intermediary bridge	C	9.5.2.2.5f)
BR	failure – resource limitation in intermediary bridge	D	9.5.1.2.5n) 9.5.2.2.5f)
BOK	reserved for interim success – intermediary bridge is forwarding transaction	E	
DR	interim success – delayed reply; end station needs more time to prepare response	F	

NOTE 1 This status can occur when an addressed responder cannot buffer the received DLSDU (possibly because of a queue-full condition). In such a case the responder is not permitted to send a DLSDU in reply.

NOTE 2 This status can be generated only when the DL(SAP)-role associated with the received destination DLSAP-address is BASIC or INITIATOR.

NOTE 3 This status can be generated only when the DL(SAP)-role associated with the received destination DLSAP-address is CONSTRAINED RESPONDER.

7.4.2 SD-Parameters in DLPDUs addressed to a DLCEP

A COMPEL ACKNOWLEDGMENT (CA) or EXCHANGE DATA (ED) DLPDU addressed to a DLCEP is used to transfer (or retransfer) a limited amount of transparent user data from one DLS-user to one or more other DLS-users and to request the status of the remote DLCEP.

A COMPEL DATA (CD) or EXCHANGE DATA (ED) DLPDU addressed to a DLCEP is used to request the immediate transfer (or retransfer) of a limited amount of transparent user data from one DLS-user to one or more other DLS-users. A CD DLPDU may contain an SD-parameters field; a CA or ED DLPDU always contains an SD-parameters field.

A DATA (DT) DLPDU addressed to a DLCEP is used to transfer a limited amount of transparent user data from one requesting DLS-user to one or more other DLS-users within the context of a DLC; or to acknowledge the transfer of such data within the context of a DLC. These DLPDUs always contain an SD-parameters field (which may be null).

All four types of DLPDU assist in the synchronization of DLCEPs and of their DLS-users:

NOTE These DLPDUs are also used outside the context of a DLC to provide DL-UNITDATA services.

a) When a CD DLPDU is being sent by an LAS DLE, or such a DLPDU is being forwarded by a bridge, then the DLPDU shall not contain an explicit source DLSAP-address and its parameter field (SD-parameters) shall be null.

b) When a CA, CD, DT or ED DLPDU is being sent from a DLCEP, or being forwarded by a bridge, and the DLPDU does not contain user data, then the negotiated parameter field shall be present, and shall have two subparts:

   1) the first shall have fixed structure and length [formats A – G] as determined during the DLC establishment process (see 7.1c)5) or 7.1d)5));

   2) the second either

      – shall have the fixed structure and length [formats J – M] determined during the DLC establishment process (see 7.1c)5) or 7.1d)5)), or

      – shall be the null format, format J.

c) In all other cases, the negotiated parameter field shall be present, and shall have two subparts each with fixed structure and length [formats A – G, concatenated with formats J – M] as determined during the DLC establishment process (see 7.1c)5) or 7.1d)5)).

d) All DLEs shall support formats A, C, F, G and J.

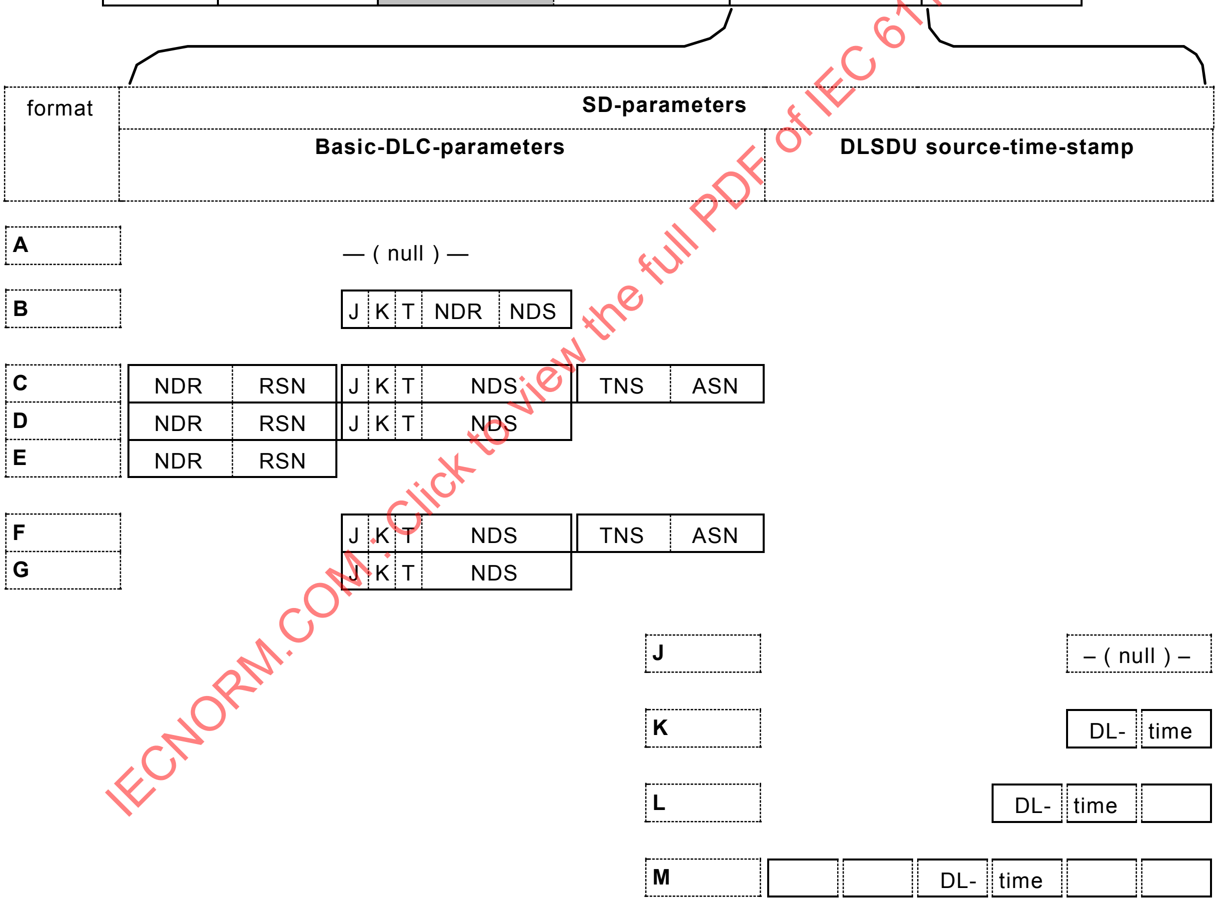
NOTE Support of formats A and G is required to facilitate migration from existing national standards.

e) DLEs which support both **timeliness** and a time-synchronism class (see 10.6.3) other than **none** shall support formats K, L and M.

The various SD-parameter formats, and their potential applicability to the CA, CD, DT and ED DLPDUs, are shown in Table 57.

Table 57 – Structure of connection-oriented CA, CD, DT and ED DLPDUs

Connection-oriented DLPDUs					
format	**Frame control**	**d-DLCEP-address**	**s-DLCEP-address**	**Parameters**	**User data**
	CA 1	[HL.]N.S	[HL.]N.S	parameters	o-pDLSDU
2	CA 2	[HL.]N.S	—	parameters	o-pDLSDU
3	CD 1	[HL.]N.S	[HL.]N.S	o-parameters	—
4	CD 2	[HL.]N.S	—	o-parameters	—
5	ED 1	[HL.]N.S	[HL.]N.S	parameters	pDLSDU
6	ED 2	[HL.]N.S	—	parameters	pDLSDU
7	DT 1	[HL.]N.S	[HL.]N.S	parameters	o-pDLSDU
8	DT 2	[HL.]N.S	—	parameters	o-pDLSDU
9	DT 3		[HL.]N.S	parameters	o-pDLSDU
10	DT 4	implied PSA	—	parameters	o-pDLSDU
11	DT 5		implied PDA	parameters	o-pDLSDU



7.4.2.1 Parameters conveying DLCEP state and DLSDU timeliness

The formats for the first subpart of the SD-parameters have been given names that reflect their expected usage. These formats are:

A) the **null** format, format A, of zero octets in length;

B) the **short** format, format B (see Table 58), of length one octet, is as follows:

Table 58 – Short format SD-parameters for DLCEP state

Retransmit	Acknowledge	Timeliness	Modulus number of post-acknowledged or requested DLSDU		Modulus number of associated DLSDU or of highest-numbered DLSDU sent		
J	K	T	NDR mod 2^2		NDS mod 2^3		
7	6	5	4	3	2	1	0

1) a one-bit retransmission-request (selective-reject) subfield, requesting (when J=1) retransmission of the NDR'th DLSDU;
2) a one-bit acknowledgment subfield, acknowledging (when K=1) that all DLSDUs before the NDR'th have been received and reported to the DLS-user;
3) a one-bit timeliness subfield, indicating (when T = 1) that the associated DLSDU originated in a buffer with associated timeliness criteria, and that those timeliness criteria were met;
4) a two-bit subfield, specifying the low-order two bits of the sequence number NDR of the DLSDU being requested, or of the DLSDU after the one being acknowledged, or both;
5) a three-bit subfield, specifying the low-order three bits of the sequence number NDS of the associated DLSDU (if this DLPDU contains user data) or of the highest-numbered DLSDU which has been sent (if this DLPDU does not contain user data);
6) the implied values for the fields of 7.4.2.1c) which are omitted from 7.4.2.1b) are zero.

NOTE DLSDUs sent from a DLCEP are assigned consecutive sequence numbers, starting with one, before transmission.

C) The **long** format, format C (see Table 59 through Table 61), of length three octets, is as follows:

Table 59 – Long format SD-parameters for DLCEP state: 1st octet

Modulus number of post-acknowledged or requested DLSDU				Zero-based Requested Segment Number			
NDR mod 2^4				RSN			
7	6	5	4	3	2	1	0

1) a four-bit subfield, specifying the low-order four bits of the sequence number NDR of the DLSDU being requested, or of the DLSDU after the one being acknowledged, or both;
2) a four-bit subfield, specifying the zero-based segment number of the DLSDU segment being requested, or zero when no segment is being requested;

Table 60 – Long format SD-parameters for DLCEP state: 2nd octet

Retransmit	Acknowledge	Timeliness	Modulus number of associated DLSDU or highest-numbered DLSDU Sent				
J	K	T	NDS mod 2^5				
7	6	5	4	3	2	1	0

3) a one-bit retransmission-request (selective-reject) subfield, requesting (when J=1) retransmission of the NDR'th DLSDU;
4) a one-bit acknowledgment subfield, acknowledging (when K=1) that all DLSDUs before the NDR'th have been received in their entirety and reported to the DLS-user;
5) a one-bit timeliness subfield, indicating (when T = 1) that the associated DLSDU originated in a buffer with associated timeliness criteria, and that those timeliness criteria were met;

6) a five-bit subfield, specifying the low-order five bits of the sequence number NDS of the associated DLSDU (if this DLPDU contains user data) or of the highest-numbered DLSDU which has been sent (if this DLPDU does not contain user data);

NOTE If this DLPDU contains no user data, then all previously-requested DLSDUs must have been sent in their entirety.

Table 61 – Long format SD-parameters for DLCEP state: 3rd octet

Zero-based Total Number of Segments in DLSDU				Zero-based Associated Segment Number			
TNS				ASN			
7	6	5	4	3	2	1	0

7) a four-bit subfield, specifying the zero-based total number of segments in the associated DLSDU, or zero when the DLPDU contains no user data;

8) a four-bit subfield, specifying the zero-based segment number of the associated DLSDU segment, or zero when the DLPDU contains no user data.

NOTE This representation has the following benefits:

a) The zero-based segment number, coupled with the fact that all segments except the last are the maximum size permitted by the DLPDU's priority, facilitates DLSDU reassembly during reception.

b) The total number of segments in the actual DLSDU can be used to allocate a buffer or queue record of appropriate size upon first receipt of any segment of the DLSDU.

D) The **unsegmented long** format, format D, of length two octets, consists of the first two octets of format C (see Table 59 and Table 60).

E) The **subscriber** format, format E, of length one octet, consists of the first octet of format C (see Table 59), with implied values for the fields J, K and T of the absent second octet. The implied values for the J, K and T fields of 7.4.2.1C), which are omitted from 7.4.2.1E), are one (1), zero (0) and zero (0), respectively.

F) The **publisher** format, format F, of length two octets, consists of the last two octets of format C (see Table 60 and Table 61). The values for the J and K fields of 7.4.2.1C) shall be zero (0).

G) The **unsegmented publisher** format, format G, of length one octet, consists of the second octet of format C (see Table 60). The values for the J and K fields of 7.4.2.1C) shall be zero (0).

Only formats A, C, F and G need be supported; support for formats B, D and E is optional.

7.4.2.2 Parameters conveying DLSDU time-of-production

The formats for the second subpart of the SD-parameters are

J) the **no-time** format, format J, of zero octets in length;

K) the **2-second** format, format K, of length two octets, consists of the third-lowest-order and second-lowest-order octets of the originating DLE's DL-time at the moment of DL-Put request execution, in that order. It is adequate to distinguish the timing of events less than 2 s apart, and provides a potential resolution of 2^{-5} ms;

L) the **8-minute** format, format L, of length three octets, consists of the fourth-lowest-order, third-lowest-order and second-lowest-order octets of the originating DLE's DL-time at the moment of DL-Put request execution, in that order. It is adequate to distinguish the timing of events less than 8 min apart, and provides a potential resolution of 2^{-5} ms;

M) the **full-time** format, format M, of length six octets, consists of the high-order six octets of the originating DLE's DL-time at the moment of DL-Put request execution, encoded most-significant-octet first. It is adequate to distinguish the timing of any two events, and provides a potential resolution of 2^{-5} ms.

7.5 Structure and encoding of SR-parameters

There are two SR-parameter formats:

a) The **null** format, format X, of length zero octets, is identical to SD-parameter format P and is used to indicate that the sending bridge was able to buffer for forwarding a received CA, CD or ED DLPDU, when the reply token was created by that received DLPDU. By using this format, the bridge implicitly returns the symbolic status "BOK" as specified in Table 62.

Table 62 – Reply status for SR DLPDUs

Short name	Definition	Hexadecimal coding
RR	failure – resource limitation in responder	4
RI	failure – responder DLE-class incompatible with this request	9
RTR	failure – LAS transfer rejected	B
BF	failure – fault in intermediary bridge	C
BR	temporary failure —resource limitation in intermediary bridge	D
BOK	interim success – intermediary bridge is forwarding transaction	E

NOTE This table is necessarily a subset of Table 56.

b) The **error** format, format Y (see Table 63), is based on SD-parameter format U (see Table 55), and so is of length one octet. Format Y is used
   - to indicate that a bridge which should have forwarded a received CA, CD or ED DLPDU was unable to buffer the received DLPDU, when the reply token was created by that received DLPDU (reasons BF or BR);
   - to reject a CA, CD or ED DLPDU received at a DLCEP under certain error conditions;
   - to reject an attempt to transfer the LAS role to the replying DLE, when the reply token was created with a TL DLPDU (reasons RR, RI and RNC).

   The subfields of the SR-parameters field's error format are.

   1) a four-bit status subfield, XXXX, encoded as specified in Table 62;

      NOTE The only two values possible for the returned status are "BF" and "BR".

   2) a four-bit subfield, ZZZZ, coded as zero and reserved for future standard use.

Table 63 – Short format SR-parameters

Status				**must be zero**			
XXXX				ZZZZ			
7	6	5	4	3	2	1	0

7.6 Structure and encoding of TD-parameters

The TD-parameters field shall be composed of four subfields (see Table 64):

Table 64 – Structure of TD-parameters

TD-parameters and the variable values transmitted					
Field type	**Link originating DL-time**	**DL-time quality**	**DL-time offset**	**DL-time prior to end-of-transmission**	**DL-time adjustment**
Value as sent	$V_S(TSL)$	$V_S(TQ)$	$V_S(DLTO)$	$V_S(DLTO) + V_S(LSTO) + C_S(NT)_1$	$P_S(TD) = C_S(NT)_2 - C_S(NT)_1$
Name at receipt	$N_S(TSL)$	$N_S(TQ)$	$N_S(DLTO)$	$N_S(DLT)$	$N_S(DLTA)$
Field size	two octets	one octet	seven octets	seven octets	three octets

a) The link designator of the DL-time-source DLE – the DLE which originates the sense of DL-time on the extended link – shall be expressed in two octets as specified in 4.3.

NOTE When a bridge interconnects two or more single links into an extended link, this is the link-id of the root link of the spanning tree formed by the bridges which interconnect the links of the extended link. It is the local link-id only for DLEs on that root link, or when no bridges interconnect the local link with other links; in this latter case, the link-id may have the value zero.

b) The quality of DL-time shall be expressed in one octet as follows (see Table 65):

Table 65 – Structure and encoding of the DL-time-quality measures

Limiting time-synchronism class			**Number of intervening links**			**Time-source type**	
TTT			LLL			SS	
7	6	5	4	3	2	1	0

1) a three-bit subfield, TTT, specifying the least-capable time-synchronism class (see 10.6.3) of all of the DLEs on the DL-time propagation path from the DL-time source to the DLE, inclusive, encoded as:

   000 NONE;

   001 1 s;

   010 100 ms;

   011 10 ms;

   100 1 ms;

   101 100 µs;

   110 10 µs;

   111 1 µs;

2) a three-bit subfield, LLL, specifying the number of intervening links on the DL-time propagation path from the DL-time source DLE to the sending DLE, expressed as 0 to 7, where

   0 indicates that the DL-time originates with the DLE itself;

   1 indicates that the DL-time originates with another DLE on the local link;

   2 indicates that the DL-time originates with a DLE on a link one bridge removed,

   and so forth;

3) a two-bit subfield, SS, expressing the method of synchronization of the DL-time source DLE with Universal Coordinated Time (UTC), the worldwide time standard, encoded as

   00 some DLE's node-time, locally generated and not received from an extra-DLS-provider source;

   01 local time (not UTC) received (directly or indirectly) from a human source;

   10 UTC received (directly or indirectly) from a human source or an unreliable

electronic source, or previously (but no longer) received (directly or indirectly) from a reliable electronic source;

11 UTC continuously received (directly or indirectly) from a reliable electronic source.

NOTE 1 Reliable electronic sources include radio receivers for national time broadcasts, atomic clocks, and similar sources which provide intrinsic worldwide time coordination.

NOTE 2 The mechanisms for synchronizing a DLE which originates the extended link's sense of DL-time with an external time source is necessarily implementation-specific and beyond the scope of standardization.

c) The DL-time-offset, V_S(DLTO), of the sending DLE during DLPDU formation shall be expressed as a signed seven-octet integer encoded most-significant-octet first, where the least significant bit represents a time granularity of approximately 2^{-13} ms as shown in Table 66.

d) The DL-time, $C_S(NT)_1 + V_S(LSTO) + V_S(DLTO)$, of the sending DLE at a moment during DLPDU formation, less than 1 s prior to transmission, shall be expressed as a seven-octet non-negative integer encoded most-significant-octet first (see Table 66), where the most significant bit is always zero and the least significant bit represents a time granularity of approximately 2^{-13} ms.

NOTE The granularity of encoded DL-time does not imply that a DLE increments DL-time at that granularity, but rather that each time the DLE increments its sense of DL-time, it does so by an amount that causes the average rate of incrementation to be approximately 2^{13} counts per millisecond (which is $2^{13} \times 10^3$ counts per second).

Table 66 – Approximate numeric significance of the bits of seven-octet DL-time

Octet of DL-time in transmission order	Symbolic contents	Approximate weight of the low-order bit of the octet
1	0YYYYYYY	1,09 year
2	DDDDDDDD	1,55 day
3	HHHHHMMM	8,74 min
4	MMMSSSSS	2,05 s
5	Smmmmmmm	8,00 ms
6	mmm•µµµµµ	31,25 µs
7	µµµµµnnn	122 ns

e) The time-adjustment, $P_S(TD) = C_S(NT)_2 \sim C_S(NT)_1$, is a small adjustment to account for systemic offsets in the accompanying DL-time subfield d), caused by known implementation considerations within the sending real end system when sending a TD DLPDU, such as early sampling of $C_S(NT)_1$ during TD DLPDU formation and transmission, such that the sum of the values specified by d) and e) will be the DL-time,

$V_S(DLTO) + V_S(LSTO) + C_S(NT)_2$,

at which the END-OF-DATA-AND-ACTIVITY PhIDU (see 5.1) will be sent to the associated PhE.

The time-adjustment shall be expressed as a three-octet unsigned integer encoded most-significant-octet first (see Table 67), where the least significant bit represents a time granularity of approximately 2^{-13} ms.

NOTE 1 This field is included to avoid the requirement that the DLE perform an equivalent real-time adjustment to the full seven octet DL-time specified by c).

NOTE 2 This adjustment will vary inversely with the DLE's actual instantaneous rate of data transmission.

Table 67 – Approximate numeric significance of the bits of three-octet short time

Octet of short time in transmission order	Symbolic contents	Approximate weight of the low-order bit of the octet
1	Smmmmmmm	8,00 ms
2	mmm•µµµµµ	31,25 µs
3	µµµµµnnn	122 ns

NOTE The objective of the computation of d) and e) is to ensure that sum of subfields d) and e) is larger than the sender's V_S(NTO) such that

A) were the sender to measure its round-trip-delay, V_S(MD) (see 8.4.1.5), when communicating with itself through the PhE and medium, then

B) if the sender were to receive its own transmissions of TD DLPDUs by receiving while transmitting, then

C) the sum of subfields d) and e) and, as computed by the sender from the received TD DLPDU, would be equal to the current value of
V_S(DLTO) + V_S(LSTO) + C_S(NT)
at the sender at the moment that the sender's PhE reports completion of reception of the TD DLPDU from the local medium.

7.7 Structure and encoding of RQ-parameters

The RQ-parameters field shall be composed of one subfield (see Table 68):

Table 68 – Structure of RQ-parameters

RQ-parameters
C_S(NT)
short-time$_1$
three octets

a) The parameter subfield shall equal the low-order 24 bits of the node-time, C_S(NT), of the sending DLE at the moment of RQ DLPDU formation just prior to (or during) transmission, representing the low-order three octets of the DLS-provider's current node-time in units of 2^{-13} ms.

NOTE The contents of this subfield reflect the relative time at which the RQ DLPDU is transmitted.

If the sending DLE's time-synchronization class (see 10.6.3) is NONE and the DLE does not maintain even an estimated C(NT), then it may use any value for this subfield.

Table 67 shows the transmission order and approximate numeric significance of the octets of this subfield. It is encoded as a three-octet unsigned integer sent most-significant octet first.

7.8 Structure and encoding of RR-parameters

The RR-parameters field shall be composed of four subfields (see Table 69):

Table 69 – Structure of RR-parameters

RR-parameters			
N_S(TQ)$_{7\text{-}5}$	N_1(NT)	N_2(NT)	C_S(NT)
measurement quality	short-time$_1$	short-time$_2$	short-time$_3$
one octet	three octets	three octets	three octets

a) The first parameter subfield shall be encoded as specified in Table 70, and shall specify
   1) the time-synchronism class of the replying DLE, TTT, encoded as in 7.6b)1);
   2) whether a DLE with a time-synchronism class of NONE maintains an estimated C(NT), E, encoded
      i) as a Boolean (0=FALSE, 1=TRUE) when TTT has the value 000;
      ii) as 1 (TRUE) when TTT has a value other than 000;
   3) a 4-bit subfield, encoded as zero.

Table 70 – Structure and encoding of the RR-time-quality measures

Limiting time-synchronism class			**DLE maintains estimated C(NT)**	**reserved**			
TTT			E	0000			
7	6	5	4	3	2	1	0

b) The second and third parameter subfields shall equal the received and locally appended parameter subfields, respectively, of the received RQ DLPDU.

c) The fourth parameter subfield shall equal the low-order 24 bits of the node-time, C_S(NT), of the replying DLE at the moment of RR DLPDU formation just prior to (or during) transmission, representing the low-order three octets of the DLS-provider's current node-time in units of 2^{-13} ms.

   NOTE The contents of this subfield reflect the relative time at which the RR DLPDU is transmitted. The resulting round-trip-delay computations are accurate, independent of any internal delays within the DLEs, provided that the sum of the propagation delays between the two DLEs is less than 2 s.

   If the sending DLE's time-synchronization class (see 10.6.3) is NONE and the DLE does not maintain even an estimated C(NT), then it shall encode this subfield as zero.

Table 67 shows the transmission order and approximate numeric significance of the octets of each of the last three subfields. Each of these subfields is encoded as a three-octet unsigned integer sent most-significant octet first.

7.9 Structure and encoding of PN-parameters

The PN-parameters field specifies the current values of those DLE and PhE parameters necessary for a receiving DLE to configure itself and its associated PhE so that they can reply to a subsequent PN DLPDU. Once set, the values of these parameters can not be changed while the receiving DLE remains ONLINE, other than by reception of another PN DLPDU as specified in 10.1.3.

NOTE Theoretically, the current LAS DLE could change the link's parameters by sending a PN DLPDU to each of the other DLEs on the local link, forcing each of them OFFLINE, after which they would learn the new link parameters in the process of again coming ONLINE. But this DL-protocol does not specify a means of requesting this LAS action.

The PN-parameters field shall be composed of seven subfields (see Table 71 through Table 76):

Table 71 – Structure of PN-parameters

PN-parameters and the variable values transmitted						
PhL maximum inter-channel signal skew	**Version**	**PhL post-transmission-gap extension units**	**PhL preamble extension units**	**Slot-time**	**Maximum-response-delay**	**Minimum-inter-DLPDU-delay**
V(PhlS)	0VVV	V(PhGE)	V(PhPE)	V(ST)	V(MRD)	V(MID)
one quartet	one quartet	one quartet	one quartet	two octets	one octet	one octet

The PN-parameters field, which partitions as shown in Table 71, shall specify the parameters needed to reply to the PN DLPDU:

a) The first two octets, ordered as shown in Table 72 and Table 73, specify the DL protocol version number and the PhL-parameters of the local link, defined in IEC 61158-2, Table 4 required for generating a reply to a received PN DLPDU:

Table 72 – PN-parameters: 1st octet

PhL maximum inter-channel signal-skew				zero	Version		
V(PhlS)				0	VVV		
7	6	5	4	3	2	1	0

1) a 4-bit subfield which specifies the required PhL maximum inter-channel signal skew, defined in IEC 61158-2, 6.2.2.2 for the PhEs of the local link;

2) a 1-bit subfield, encoded as zero;

3) a 3-bit DL protocol version number (VVV), whose value is specified at the beginning of Clause 7;

Table 73 – PN-parameters: 2nd octet

PhL post-transmission-gap extension units				PhL preamble extension units			
V(PhGE)				V(PhPE)			
7	6	5	4	3	2	1	0

4) a 4-bit subfield which specifies the required number of PhL post-transmission-gap extension units, defined in IEC 61158-2, 6.2.2.2 for the PhEs of the local link;

NOTE Although this measure is named "post-transmission-gap-extension" in IEC 61158-2, it is a measure of the amount of observable non-transmission required between any two transmissions on the local medium, as observed from any point on that medium, whether the two transmissions are from the same source PhE or from two different source PhEs. Thus it could just as correctly be named "pre-transmission-gap-extension", or even more correctly "inter-transmission-gap-extension".

5) a 4-bit subfield which specifies the required number of PhL preamble extension units, defined in 6.2.2.2 of IEC 61158-2, for the PhEs of the local link;

b) The next four octets, ordered as shown in Table 74 through Table 76, specify the DLL-parameters for the local link required for generating a reply to a received PN DLPDU:

Table 74 – PN-parameters: 3rd and 4th octets

Slot-time															
V(ST)															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Table 75 – PN-parameters: 5th octet

Maximum-response-delay							
V(MRD)							
7	6	5	4	3	2	1	0

Table 76 – PN-parameters: 6th octet

Minimum-inter-DLPDU-delay							
V(MID)							
7	6	5	4	3	2	1	0

1) The fifth parameter subfield shall convey, in two octets transmitted most significant octet first, the current value of V(ST), slot-time, defined in 4.7.1.1.
2) The sixth parameter subfield shall convey, in one octet, the current value of V(MRD), maximum-response-delay, defined in 4.7.1.3.
3) The seventh parameter subfield shall convey, in one octet, the current value of V(MID), minimum-inter-DLPDU-delay, defined in 4.7.1.12.

7.10 Structure and encoding of DD-parameters

The DD-parameters (Delegation-Duration parameters) field shall be composed of one subfield (see Table 77):

Table 77 – Structure of DD-parameters

DD-parameters
Delegated or requested duration
V(RD)
two octets

a) The subfield shall specify the duration for which the token is being delegated or for which delegation is being requested. It is measured in octet-durations, and is encoded as a two-octet unsigned integer sent most-significant octet first. Its range is 0 to 65 000.

NOTE The minimum practical value for using this duration for a transmission is about 16. The value 0 is used only during maintenance of the local link's live-list V(LL).

8 DL-service elements of procedure

NOTE 1 The elements of procedure in this clause correspond generally to the highest DLL sublevel, as specified in 4.1.3.

Throughout Clause 8, whenever a conditional action is specified and the specified enabling condition does not occur, then the corresponding action also does not occur. The net effect is that events which do not meet any of the enabling conditions specified in a service procedure will have no consequential actions with respect to that specific service procedure.

Throughout Clause 8, the value (V(NRC)+1) is used as a link-independent indication of the maximum number of times to perform a procedure – once initially, plus V(NRC) repetitions – before concluding that an unrecoverable error situation exists and that alternate more drastic action is required. When the DLE can ascertain, from the involved DL(SAP)-addresses or DLCEP-addresses, that all of the DLEs involved in a DLC are local to a single link, then the DLE may substitute in a consistent manner the link-dependent value (V(MRC)+1) for the link-independent value (V(NRC)+1).

NOTE 2 Throughout this clause, a DLE receives a transmit opportunity, either by receipt of an appropriate immediate-reply opportunity or by receipt of an appropriate token.

8.1 Operation of the DL(SAP)-address, buffer and queue management services

The DL(SAP)-address, buffer and queue management services are the create and delete buffer or queue services, the bind and unbind DL(SAP)-address services, the put buffer service, and the get buffer or queue service.

8.1.1 Receipt of a DL-CREATE request primitive

When the DLE receives a DL-CREATE request, it shall

a) allocate a buffer of the specified DLSDU size, or queue of the specified DLSDU size and maximum number of entries, and initialize it to empty;

   NOTE Statistical allocation techniques using a shared multi-use storage area are permissible.

b) assign a buffer-or-queue DL-identifier, and any provided DLS-user-identifier, to that buffer or queue;

c) return that identifier to the DLS-user, together with a status of “success”.

Alternatively, an appropriate failure status shall be returned to the DLS-user.

8.1.2 Receipt of a DL-DELETE request primitive

When the DLE receives a DL-DELETE request, if the specified buffer or queue

a) was created by the DLS-user and not by DL-management action;

b) is not currently bound to any DLCEP or DL(SAP)-address

then the DLE shall delete the specified buffer or queue and return a status of “success”.

Otherwise the DLE shall return an appropriate failure status.

8.1.3 Receipt of a DL-BIND request primitive

When the DLE receives a DL-BIND request, it shall check

a) that the specified DL(SAP)-address is a group DL-address not currently in use by the requesting DLS-user, or a DLSAP-address not currently in use by the DLE;

b) that the specified link and node components of the DL(SAP)-address are compatible with the one or more link address components and one or more node address components assigned to the DLE (see 4.3);

c) that any specified explicit bindings to DL-queues or DL-buffers are valid and do not conflict with other existing bindings to those queues or buffers;

d) that any specified static or dynamic QoS attributes are valid and permitted by local DL-management.

Failure shall be reported to the DLS-user with an appropriate status. Otherwise

1) the DL(SAP)-address shall be associated to the requesting DLS-user, and to any provided DLS-user-identifier;

2) all explicitly-specified queues and buffers shall be bound as specified to the DL(SAP)-address;

3) the static and dynamic QoS attributes shall be bound to the DLSAP-address;

4) the specified DL(SAP)-address shall be activated for reception;

5) a DL(SAP)-address identifier shall be assigned to the DL(SAP)-address and returned to the DLS-user, together with a status of “success”;

6) if either
   i) the DLE is a fractional-duty-cycle (FDC) DLE, or
   ii) the DL(SAP)-address has a link-designator component (see 4.3.2.1) whose value specifies a flat non-local DL-address (between 0001_{16} and $0FFF_{16}$, inclusive), and the variable V(TL) has a non-zero value,

   then the DLE shall send a DL-address report SPDU as specified in 9.3.6.3, with an appropriate reason for the report SPDU and with a request identifier of zero, to the DL-support functions of all bridges on the local link (see 4.3.3.2).

8.1.4 Receipt of a DL-UNBIND request primitive

When the DLE receives a DL-UNBIND request,

a) if the specified DL(SAP)-address currently is not bound to the requesting DLS-user, or if the specified DL(SAP)-address was bound by DL-management action, then the DLE shall ignore the request primitive;
b) otherwise the DLE shall
   – disconnect any DLCEPs associated with the DL(SAP)-address;
   – confirm with an appropriate error status any unconfirmed connection-oriented or connectionless-mode service requests outstanding at that DL(SAP)-address or its DLCEPs;
   – unbind all queues and buffers from the specified DL-address;
   – disassociate the specified DL-address from the requesting DLS-user.

   If the DLE has no other DLS-users associated with that DL-address (which is always the case for DLSAP-addresses) then the DLE shall deactivate reception of that DL-address, in which case if either

      1) the DLE is a fractional-duty-cycle (FDC) DLE, or
      2) the DL(SAP)-address has a link-designator component (see 4.3.2.1) whose value specifies a flat non-local DL-address (between 0001_{16} and $0FFF_{16}$, inclusive), and the variable V(TL) has a non-zero value,

   then the DLE shall send a DL-address report SPDU as specified in 9.3.6.3, with an appropriate reason for the report SPDU and with a request identifier of zero, to the DL-support functions of all bridges on the local link (see 4.3.3.2).

8.1.5 Receipt of a DL-PUT request primitive

When the DLE receives a DL-PUT request for a buffer

a) which is associated with the requesting DLS-user;
b) which is not bound as a receiving buffer
   1) to a peer or subscriber DLCEP, or
   2) to a DLSAP-address whose DL(SAP)-role is INITIATOR or CONSTRAINED RESPONDER or UNCONSTRAINED RESPONDER;

then

c) if a DLSDU is presented, and the size of the DLSDU is less than or equal to the buffer-size, then the DLE shall
   1) set the contents of the buffer equal to the presented DLSDU;
   2) indicate for each DLCEP which has a sending binding to the buffer that the buffer contains a new DLSDU;

3) set the timeliness-status of buffer-writing, V_B(TS) (see 4.7.4.21), to the DLS-user-specified timeliness, or FALSE if the user did not specify timeliness;
4) if the timeliness-status of buffer-writing is TRUE, then set the variable time-of-last-buffer-write, V_B(TW) (see 4.7.4.19), associated with the buffer to the current DL-time;
5) if the timeliness-status of buffer-writing is TRUE, then set the time-of-production associated with the buffer, V_B(TP) (see 4.7.4.20), to the current DL-time;
6) return a status of "success";

d) else if c) does not apply, and no DLSDU is being presented (that is, the buffer is being set empty), and the buffer is bound to a DLSAP-address, then the DLE shall set the buffer empty and shall return a status of "success";
e) else if neither c) nor d) applies, then the DLE shall return an appropriate failure status.

Any ongoing accesses to the contents of a buffer or to its associated timeliness information, which are incomplete at the time of a DL-PUT request, shall not be affected by the DL-PUT request.

NOTE This constraint ensures that each access to a buffer is logically atomic.

8.1.6 Receipt of a DL-GET request primitive

a) If the DLE receives a DL-GET request for a buffer which is associated with the requesting DLS-user, then
   1) if the buffer was written by a DLCEP which specified RESIDENCE timeliness or UPDATE timeliness or SYNCHRONIZED timeliness or TRANSPARENT timeliness, then the DLE shall evaluate the associated timeliness criteria as specified in 8.1.7, using the current DL-time as the time of buffer readout, and shall return the result as the local-DLE-timeliness attribute of the request;
   2) otherwise, if 1) does not apply, then the DLE shall return the value FALSE as the local-DLE-timeliness attribute of the request;
   3) the DLE shall return the timeliness-status, V_B(TS) (see 4.7.4.21), associated with the buffer's writing as the sender-and-remote-DLE-timeliness attribute (see 8.2.2.5.3) of the request.

      NOTE This sender-and-remote timeliness attribute always has the value FALSE when the buffer was written by an instance of the unitdata exchange service.

   4) If the sender-and-remote-DLE-timeliness attribute associated with the buffer is TRUE, and the buffer was written by a DLCEP which provides DL-time-of-production, then the DLE shall return the DL-time-of-production associated with the buffer, V_B(TP) (see 4.7.4.20).
   5) The DLE shall return the current contents of the buffer, with a status of "success" if the buffer is non-empty, and a status of "possible failure – buffer empty" if the buffer is empty.
   6) If the buffer is a non-retentive buffer (BUFFER-NR), then the buffer shall be set empty.
   7) Reading a buffer and its associated timeliness information shall be logically atomic with respect to writing the buffer.

      NOTE When an implementation provides access to a buffer for an extended period of time during buffer read or write, then in the worst case this atomicity restriction requires that separate copies of the buffer's contents and timeliness information be provided for each reader and for the one writer, in addition to the actual buffer with its contents and timeliness. Then each reader may be in the middle of an extended access to a different epoch of the buffer's contents, and the writer may be writing a tentative buffer during reception, which will become the current buffer only if a Ph-error or FCS error is not detected before the end of the reception process.

b) If the DLE receives a DL-GET request for a DLS-user specified queue which is
   – associated with the requesting DLS-user;
   – not bound as a sending queue either to a DLCEP or to a DLSAP-address;
   – non-empty

then the DLE shall

1) return the next DLSDU contained in the queue, together with the called DLC identifier associated with that DLSDU, or DLL priority and called and calling DL(SAP)-addresses or DL(SAP)-address-identifiers associated with that DLSDU;
2) return local- and remote-DLE-timeliness attributes of FALSE;
3) remove that DLSDU from the queue;
4) return a status of “success”.

Otherwise the DLE shall return an appropriate failure status.

8.1.7 Computation of DL-timeliness

The DLE shall compute the intrinsic DL-timeliness for the buffer based on the type of buffer access – writing or reading – and the corresponding type of DL-timeliness which was specified on the corresponding local DL-CONNECT request or response primitive, as follows, where $P_C(NP.\Delta T)$ is the appropriate time-window-size specified in that request or response primitive:

a) **RESIDENCE**

DL-timeliness ≡ TRUE when $0 \le$ (current-DL-time – $V_B(TW)$) and (current-DL-time – $V_B(TW)$) $\le P_C(NP.\Delta T)$;
DL-timeliness ≡ FALSE otherwise.

b) **UPDATE**

DL-timeliness ≡ TRUE when $0 \le (V_B(TW) - V_C(TNA))$ and $(V_B(TW) - V_C(TNA)) \le P_C(NP.\Delta T)$;
DL-timeliness ≡ FALSE otherwise.

c) **SYNCHRONIZED**

DL-timeliness ≡ TRUE when $0 \le (V_B(TW) - V_C(TNA))$ and $(V_B(TW) - V_C(TNA)) \le$ (current-DL-time – $V_C(TNA)$) and (current-DL-time – $V_C(TNA)$) $\le P_C(NP.\Delta T)$;
DL-timeliness ≡ FALSE otherwise.

d) **TRANSPARENT**

DL-timeliness ≡ TRUE.

e) **NONE**

DL-timeliness ≡ FALSE.

8.2 Operation of the connection-mode services

The connection-mode services are the DLCEP establishment and DLCEP release services, the DLC data transfer and DLCEP reset service, and the DLC subscriber query service.

IEC 61158-3-1, Figure 16, shows a conceptual state transition diagram for sequences of DLC service primitives at a DLCEP. A corresponding state transition diagram for this DL-protocol is shown in Figure 11 and in Annex B, where similar states have identical state numbers.

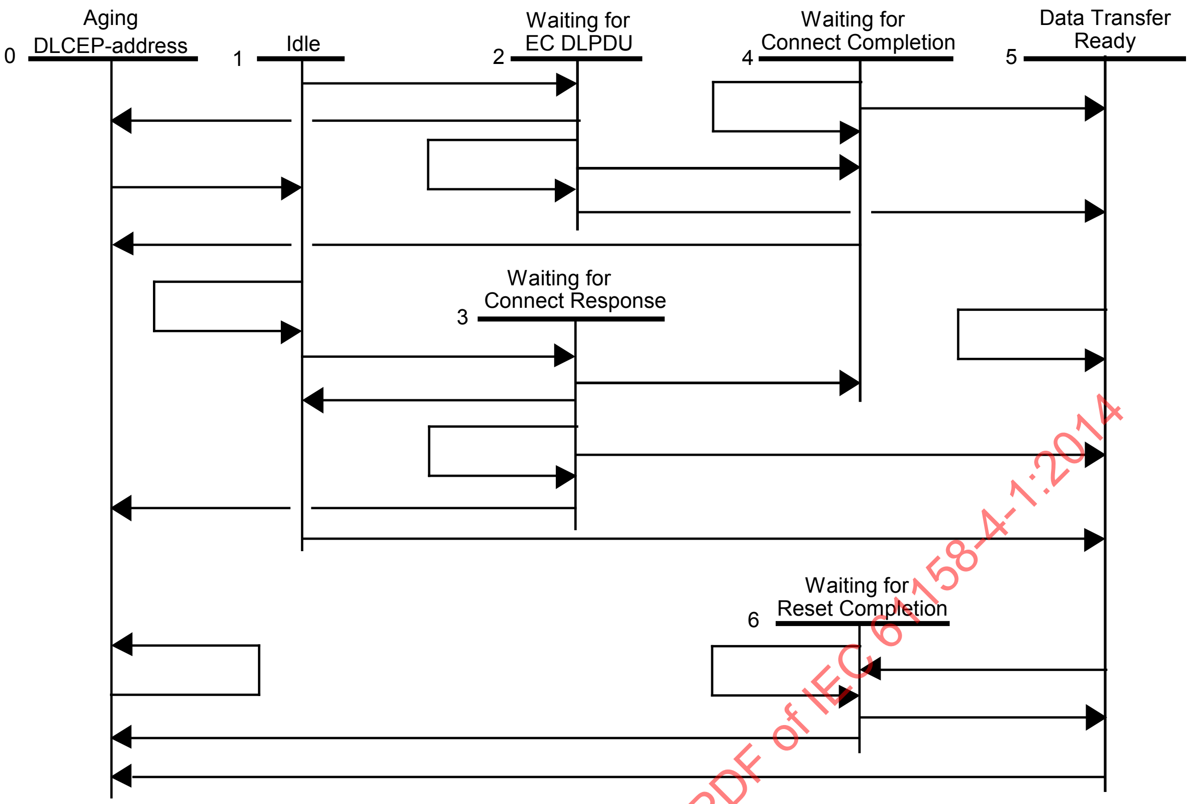


Figure 11 – State transition diagram for a DLCEP

8.2.1 Operation of the DLCEP establishment and DLCEP release services

The primitives of the DLCEP-establishment service are DL-CONNECT request, indication, response and confirm, and DL-CONNECTION-ESTABLISHED indication. The primitives of the DLCEP-release service are DL-DISCONNECT request and indication. The associated local DLS-user, or the DLCEP's remote peer or publisher DLS-user, or the DLS provider, may invoke the DLCEP-release service any time after DLCEP establishment has begun.

During DLCEP establishment and release, if the DLCEP is a peer or publisher DLCEP, and the DLCEP-address used by the DLCEP is not sent as a source DL-address in a DC or EC DLPDU, and if either

a) the DLE is a fractional-duty-cycle (FDC) DLE, or
b) the DLCEP-address has a link-designator component (see 4.3.2.1) whose value specifies a flat non-local DL-address (between 0001_{16} and $0FFF_{16}$, inclusive), and the variable V(TL) has a non-zero value,

then the DLE shall send a DL-address report SPDU as specified in 9.3.6.3, including the specified DLCEP-address, with an appropriate reason for the report SPDU and with a request identifier of zero, to the DL-support functions of all bridges on the local link (see 4.3.3.2).

8.2.1.1 DLC negotiation rules

The DLS-user-visible aspects of the DLC negotiation rules are specified in IEC 61158-3-1. Additional negotiation rules which do not impact DLS-user visible aspects of the DLC are permitted to be specified in 7.1 of this standard. In case of apparent conflict, the rules specified in this subclause take precedence over any which may be specified in 7.1, which in turn take precedence over those specified in IEC 61158-3-1.

NOTE Where a subscribing DLS-user which is attempting to join an existing DLC requests DLC attributes which are not provided by the attributes of that existing DLC, then the publishing DLS-user may either

A) indicate its inability to provide the requested attributes by replying with a DL-DISCONNECT request primitive to the proposed subscriber,

B) create a new DLC with the desired attributes and connect the subscriber to that DLC, or

C) modify the existing DLC to include the requested characteristics, when permitted by this standard.

a) If the publisher, or either peer, of a DLC specifies a DLPDU-authentication attribute of MAXIMAL, then

NOTE DLPDU-authentication of MAXIMAL is provided primarily for use in safety systems. For this reason it maximizes the amount of state information exchanged in each DLPDU sent on the DLC and prohibits two-way-user data exchange in a single transaction, centralized schedule execution and other activities in which multiple DLEs need to have consistent state information.

1) each DLPDU sent from each DLCEP of the DLC shall contain the maximum permitted number of explicit addresses;

2) the EC-parameters in each EC DLPDU shall be constrained as follows:

i) the address-size subfield (SS) shall specify LONG;

ii) the DLPDU-authentication subfield (XX) shall specify MAXIMAL;

iii) the residual-activity subfield (A) shall specify TRUE in the publisher-to-subscriber direction, or in all sending peer-to-peer directions, of data transfer;

NOTE Residual activity is not meaningful in the subscribers-to-publisher direction.

iv) both two-way data exchange subfields (E) shall specify FALSE;

v) SD-parameter format B (subfield FFF), and time-stamp formats K and L (subfield HH), shall not be requested or used in either direction on the DLC;

3) if a DL-SCHEDULE-SEQUENCE request specifies a sequence in which one or more sequence elements specify the local or remote DLCEP of this DLC, then that sequence may be either locally or centrally scheduled, but shall be locally executed (see see 8.4.3.1).

b) If a) does not apply then

i) if the publisher, or either peer, of a DLC specifies a DLPDU-authentication attribute of SOURCE, then the DLPDU-authentication subfield (XX) in the EC-parameters shall specify SOURCE, and each DLPDU sent from each DLCEP of the DLC shall contain the maximum permitted number of explicit addresses;

ii) if a subscriber of a DLC specifies a DLPDU-authentication attribute of MAXIMAL in a DL-CONNECT request primitive, then the DLPDU-authentication subfield (XX) in the resulting EC-parameters DLPDU shall specify MAXIMAL. If the requested DLC was already established, then

A) if that DLC was not established with DLPDU-authentication attribute of MAXIMAL then the publishing DLE shall reject the connection establishment request from that subscriber;

B) otherwise, when A) does not apply, then the publishing DLE shall attempt to add that subscriber to the existing DLC;

iii) if a subscriber of a DLC specifies a DLPDU-authentication attribute of SOURCE in a DL-CONNECT request primitive, then the DLPDU-authentication subfield (XX) in the resulting EC-parameters DLPDU shall specify SOURCE. If the requested DLC was already established, then

A) if that DLC was established with a DLPDU-authentication attribute of ORDINARY then the publisher's DLE shall change the DLPDU-authentication to SOURCE and each DLPDU sent from each DLCEP of the DLC shall thereafter contain the maximum permitted number of explicit addresses;

B) otherwise, when A) does not apply, then the publishing DLE shall attempt to add that subscriber to the existing DLC;

iv) else if none of i) to iii) applies, then the DLPDU-authentication subfield (XX) in the EC-parameters shall specify ORDINARY, and each DLPDU sent from each DLCEP of the DLC shall contain the minimum permitted number of explicit addresses;

c) The DLL path-diversity subfield (Q) of the EC-parameters shall specify ANY-PATH as a default. Negotiation of this subfield is from ANY-PATH to THIS-PATH.

NOTE See the note in 7.1a)3) for use of the value THIS-PATH and choice of the actual path in the THIS-PATH case.

d) The address-size subfield of the EC-parameters shall be determined as follows.

1) If required by a)2)i), or if any of the DL-addresses of the EC DLPDU have only a LONG representation, then the address-size subfield of the EC-parameters shall specify LONG.

NOTE A DL-address has a SHORT representation when there is an equivalent two-octet DL-address (see 5.2.1.1).

2) Else, when 1) does not apply, and either b)i) applies or any member of the DLC is a fractional-duty-cycle (FDC) DLE, then the address-size subfield of the EC-parameters shall specify SHORT.

3) Otherwise, when 1) and 2) do not apply, the address-size subfield of the EC-parameters shall specify VERY-SHORT.

NOTE 6 The address-size VERY-SHORT applies only to DT DLPDUs sent using a reply token (see 6.7.1 formats 4 and 5); in all other cases the address-size SHORT is actually used.

e) The DLCEP-data-delivery-features subfield (TT) of the EC-parameters shall specify, independently for each direction of the DLC, the provided data-delivery features, as specified in IEC 61158-3-1, except that the value NONE shall be replaced by UNORDERED with a maximum window size (WWWW) of zero and maximum-DLSDU-size subfield (M…M) of zero in the corresponding direction, indicating a simplex DLC.

f) The residual-activity subfield (A) of the EC-parameters shall specify TRUE in a publisher-to-subscriber or sending peer-to-peer direction of data transfer when so required by a)2)iii), or by DL-management, or by a publishing or peer DLS-user, and shall specify FALSE otherwise. Negotiation of this subfield is from FALSE to TRUE.

g) Window size negotiation occurs independently for each direction of the DLC. The actual maximum window size for a given direction of transmission shall be the smaller of the sender's maximum window size and the receiver's maximum window size in that direction, and the maximum-window-size subfield (WWWW) of the EC-parameters shall specify zero only when the maximum-DLSDU-size subfield (M…M) in the same direction is zero, indicating a simplex DLC.

h) The SD-parameter-format subfield (FFF) of the EC-parameters shall specify the negotiated format for each direction of data transmission. The initial formats for the two directions of data transmission shall be chosen to meet the following constraints.

NOTE 1 These constraints are imposed by the DLC protocol and selected service features.

1) If a) applies, then format B shall not be chosen for either direction.

2) If the DLCEP-data-delivery-features subfield (TT) in either direction specifies ORDERED, DISORDERED, or CLASSICAL, then

i) if the maximum-window-size subfield (WWWW) in that direction specifies a value of four (4) or more, then format B shall not be chosen for that direction;

ii) if the DLCEP-class subfield (CC) specifies PEER, then the format chosen for the other direction shall contain both J and K subfields;

iii) if the DLCEP-class subfield (CC) specifies PUBLISHER, then the format chosen for that direction shall be format F or G, and the format chosen for the other direction shall be format C or D or E;

iv) if the DLCEP-class subfield (CC) specifies SUBSCRIBER, then the format chosen for that direction shall be format C or D or E, and the format chosen for the other direction shall be format F or G.

NOTE Format C is included in iii) and iv) only because support of formats D and E are not mandatory.

3) If a DLCEP-data-delivery-features subfield (TT) specifies ORDERED, DISORDERED, or CLASSICAL, then the format chosen for that direction shall contain an NDS subfield, and the format chosen for the other direction shall contain an NDR subfield.

4) If a maximum-DLSDU-size subfield (M…M) specifies a value greater than the amount of DLS-user data that can be conveyed in a single DLPDU of the DLC's priority, as determined by the DLC's DLL priority subfield (PP), then the format chosen for that direction shall contain TNS and ASN subfields, and the format chosen for the other direction shall contain an RSN subfield.

5) If a timeliness-included subfield (G) specifies TRUE, then the format chosen for that direction shall contain a T subfield.

6) The format chosen shall be the shortest (fewest octets in length) possible of those which the DLE supports and which meets all of constraints 1) to 5).

NOTE 2 The expected set of formats in the sending direction are as follows.

a) Formats A, and potentially E, from a PEER DLCEP when the sending direction's DLCEP-data-delivery-features subfield (TT) specifies UNORDERED, depending on the corresponding DLCEP-data-delivery-features subfield (TT) in the other direction;

b) Formats B, C and D, and potentially F and G, from a PEER DLCEP when the sending direction's DLCEP-data-delivery-features subfield (TT) specifies ORDERED, DISORDERED, or CLASSICAL, depending on the corresponding DLCEP-data-delivery-features subfield (TT) in the other direction;

c) Formats A, F and G from a PUBLISHER DLCEP;

d) Formats A, C, D, E and G from a SUBSCRIBER DLCEP, where formats D and G are chosen when subscriber-to-publisher DLSDUs have associated timeliness, and format C is chosen when format D would otherwise be chosen but is unavailable because its support is not mandatory. For formats C, D and G, the publishing DLE will ignore the NDS and K subfields within the SD-parameters of received CA, CD, DT and ED DLPDUs.

i) The 2-way-data-exchange subfield (E) shall specify FALSE when the maximum-DLSDU-size subfields (M…M) is zero in one direction of data transmission, or when requested by a peer DLE during negotiation of a PEER DLC, or when required by a)2)iv) or by local DL-management or by DLE construction; and shall specify TRUE otherwise. Negotiation of this subfield is from TRUE to FALSE.

j) Timeliness attributes of the DLCEP are communicated but not negotiated:

1) The timeliness-included subfield (G) of the EC-parameters shall specify FALSE when the specified sender-timeliness is NONE, and shall specify TRUE otherwise.

2) The time-stamp-format subfield (HH) of the EC-parameters shall specify

i) format J when there is no sender timeliness, or the specified sender-timeliness is NONE, or time-of-production is not requested;

ii) format K when there is sender timeliness, and time-of-production is requested, and the timeliness-class is RESIDENCE or SYNCHRONIZED, and the associated time-window-size (ΔT) is 1 s or less;

iii) format L when there is sender timeliness, and time-of-production is requested, and the timeliness-class is RESIDENCE or SYNCHRONIZED, and the associated time-window-size (ΔT) is greater than 1 s;

iv) format M when there is sender timeliness, and time-of-production is requested, and the timeliness-class is UPDATE or TRANSPARENT.

k) If one direction of data communication is not required for the DLC, because the DLS-user-specified data delivery features for that direction specified NONE, then in that direction

1) the residual-activity subfield (A) shall be specified as FALSE;

2) the Queue/Buffer (B) subfield shall be specified as QUEUE;

3) the timeliness subfield (G) shall be specified as FALSE;

4) the time-stamp-format subfield (HH) shall be specified as FORMAT J.

8.2.1.2 Receipt of a DL-CONNECT request primitive

When the DLE receives a DL-CONNECT request primitive from a DLS-user, the DLE shall perform the following series of actions, and if any error is detected during the process, then the DLCEP shall be disconnected as specified in 8.2.1.8.

a) The DLE shall assign a new DLCEP-identifier, and the provided DLS-user-identifier, to the DLCEP which may result from the request, and provide that DLCEP-identifier to the DLS-user as the single output parameter of the request.

b) The DLE shall create and start a user-request timer T_U(MCD) with a duration based on the user-specified maximum confirm delay for DL-CONNECT, DL-RESET and DL-SUBSCRIBER-QUERY primitives. If the specified value was other than UNLIMITED, then the duration of this timer should be **Error!**; otherwise the duration should be **Error!**. DL-management may override these preferred durations.

   NOTE See the permission at the start of clause 8 with regard to use of V(NRC) versus V(MRC).

c) The DLE shall validate the calling-DLSAP-address or DLCEP-identifier provided by the DLS-user; if invalid, the DLE shall reject the DL-CONNECT request with a DL-DISCONNECT indication.

   NOTE The structure and value ranges of a DLSAP-address are specified in 4.3.

d) The DLE shall validate the self-consistency of the requested QoS parameter set, where all static and non-specified dynamic parameters assume the default values associated with the calling-DLSAP-address, and where the following automatic adjustments to that QoS occur.

   1) Where any parameter is in violation of a local DL-management-imposed limit, then that parameter shall be set equal to that limit, if permitted by the negotiation rules specified in 8.2.1.1, or the DLE shall reject the DL-CONNECT request with a DL-DISCONNECT indication.

   2) If any maximum DLSDU size equals zero or the sending DLCEP data delivery features specify NONE, then the corresponding sending DLCEP data delivery features shall be set to UNORDERED.

      NOTE This special case is not considered to be a violation of the negotiation rules of 8.2.1.1.

   3) if any maximum DLSDU size is greater than the maximum which can be conveyed in a single DLPDU of the specified DLL priority (see 4.2.3), and the sending DLCEP data delivery features in the corresponding direction are UNORDERED, then those sending DLCEP data delivery features shall be set to ORDERED.

e) if the calling address identifier is a DLCEP-identifier for an existing DLCEP, then

   1) If the existing DLCEP is a publisher DLCEP, then the DLE shall

      i) use the specified DLCEP's local DLSAP-address as the calling-DLSAP-address for this new DL-CONNECT request;

      ii) set each QoS parameter equal to the corresponding parameter of the specified DLCEP, if permitted by the negotiation rules of 8.2.1.1, or the proposed DL-CONNECT request shall be rejected with a DL-DISCONNECT indication specifying "connection rejection – QoS not available, permanent condition", terminating processing of the DL-CONNECT request;

      iii) use the specified DLCEP's publisher-DLCEP-address reuse-discriminator in the EC-parameter;

      iv) consider the new DL-CONNECT request to be a request for adding new subscribers to the existing DLC and continue with j).

   2) If the existing DLCEP's DLCEP-class is PEER or SUBSCRIBER, then the DLE shall reject the DL-CONNECT request with a DL-DISCONNECT indication.

f) Otherwise, if e) does not apply, then the DLE shall determine the maximum send and receive window sizes based on the respective buffer-and-queue bindings, as follows

   1) If the DLCEP-features are NONE, then the corresponding window size shall be zero (0).

2) Otherwise, if 1) does not apply, then

   i) if the binding was to a BUFFER, then the corresponding window size shall be one (1);

   ii) if the binding was to a QUEUE-*K*, then the corresponding window size shall be the smaller of *K* or 15;

   iii) if the default binding was used, then the corresponding window size shall be at least one (1);

   iv) in all cases, DL-management can further constrain this window size.

g) If the optional calling-DLCEP-address was specified in the request primitive, then the DLE shall assign that DLCEP-address to the DLCEP; if any conflicting assignment is detected, then the DLCEP shall be terminated as specified in 8.2.1.8.

h) Otherwise, if g) does not apply, and if the called address is not a DLCEP-address presumed to be for a publisher DLCEP, then a DLCEP-address allocated to the DLE and not currently in use shall be assigned to the DLCEP; where possible this shall be a SHORT DL-address of the local link.

   When the called address is a DLCEP-address presumed to be for a publisher DLCEP, then this assignment is optional.

i) The DLC shall initialize the DLCEP's V_C(NP), V_C(N), V_C(R), V_C(A), V_C(M), V_C(MS), V_C(H), V_C(HS) and V_C(L) variables as specified in 4.7.4.

j) The DLE shall encode an EC DLPDU as specified in 6.1 and 7.1:

   1) If the called-DL-address parameter specifies a DL(SAP)-address or DLCEP-address, then the DLE shall form an EC DLPDU with three addresses, whose values shall be, respectively,

      i) the called-DL(SAP)-address or DLCEP-address;

      ii) the DLCEP-address assigned to the DLCEP, or calling-DLSAP-address if no such assignment was done as in h);

      iii) the calling-DLSAP-address.

   2) If the called-DL(SAP)-address parameter specifies UNKNOWN, then

      i) if the DLS-user did not specify a calling-DLCEP-address, then the DLE shall reject the DL-CONNECT request with a DL-DISCONNECT indication;

      ii) else if i) does not apply, and the DLCEP-class is PUBLISHER, then the DLE shall form an EC DLPDU with two addresses, whose values shall be the DLCEP-address assigned to the DLCEP, and the calling-DLSAP-address, respectively;

      iii) else if i) does not apply, and the DLCEP-class is PEER or SUBSCRIBER, then the DLE shall not form and send an EC DLPDU, but shall await the receipt of a complementary EC DLPDU from the remote PEER or PUBLISHER.

k) If the DLCEP class of the DLE is to be either PEER or SUBSCRIBER, then

   1) if an EC DLPDU was formed, then

      i) the DLE shall set the reply-requested field in the EC-parameters in the DLPDU;

      ii) the DLE shall queue the DLPDU at TIME-AVAILABLE priority as specified in 8.4.5.

   2) the DLE shall activate recognition of the DLCEP's local DLCEP-address and change the DLCEP state, V_C(ST), to WAITING-FOR-EC-DLPDU.

l) If the sending DLCEP class of the DLE is to be PUBLISHER, then

   1) the DLE shall clear the reply-requested field in the EC-parameters in the DLPDU;

   2) if the source DLCEP-address is not that of an existing DLCEP, the DLE shall assign a new value to the publisher-DLCEP-address reuse-discriminator subfield (NNN) of the EC-parameters (see 7.1a)2)):

i) if the DLE is capable of recording the publisher-DLCEP-address reuse-discriminator between DLCEP incarnations, then it should maximize the interval between reuse of the same discriminator value;

ii) otherwise, when i) does not apply, the DLE shall choose the discriminator value randomly with an approximately uniform distribution;

3) the DLE shall queue the DLPDU at TIME-AVAILABLE priority as specified in 8.4.5;

4) the DLE shall issue the DL-CONNECT confirm for the DLCEP immediately after transmission of the EC DLPDU;

5) the DLE shall cancel the user-request timer T_U(MCD);

6) if e) did not apply, then the DLE shall activate recognition of the DLCEP's local DLCEP-address and change the DLCEP state, V_C(ST), to DATA-TRANSFER -READY.

8.2.1.3 Receipt of a DL-CONNECT response primitive

When the DLE receives a DL-CONNECT response primitive from a DLS-user, the DLE shall perform the following series of actions; if any error is detected during the process, then the DLCEP shall be disconnected as specified in 8.2.1.8.

a) The DLE shall validate the DLCEP-identifier, and the responding DLSAP-address or DLCEP-identifier, provided by the DLS-user, and shall associate the provided DLS-user-identifier with the DLCEP.

b) If the identified DLCEP is not in the WAITING-FOR-CONNECT-RESPONSE state, the DLCEP shall be disconnected.

c) The DLE shall validate the self-consistency of the response QoS parameter set, where all static and non-specified parameters assume their default values associated with the responding-DLSAP-address, and where the automatic adjustments to that QoS specified in 8.2.1.2d) occur. The DLE shall then validate the consistency of the resulting QoS parameter set with the corresponding parameters from the received EC DLPDU, and the adherence to the rules of parameter negotiation specified in 8.2.1.1.

d) If the responding address identifier in the DL-CONNECT response was for a DLCEP, then the DLE shall consider the new DL-CONNECT response as requesting a merger of the response's attempted DLC with an existing DLC, after which the DLCEP-identifier which was specified in the associated DL-CONNECT indication will no longer be valid and any DLS-provider state information (including publisher-DLCEP-address reuse-discriminator) related to the indication's attempted DLC shall be discarded:

1) If the existing DLC is a multi-peer DLC, then the DLE shall consider the DL-CONNECT response to be a request for adding a new subscriber to the existing DLC:

i) If the specified DLCEP is a publisher DLCEP, then the DLE shall

A) use the specified DLCEP's local DLCEP-address and DLSAP-address as the responding DLCEP-address and DLSAP-address for this DL-CONNECT response;

B) set each QoS parameter, and the publisher-DLCEP-address reuse-discriminator, equal to the corresponding parameter of the specified DLCEP, if permitted by the negotiation rules of 8.2.1.1;

C) if necessitated by the rule of 8.2.1.1d), then change the address size of the existing DLC from VERY-SHORT to SHORT or from SHORT to LONG.

ii) If no negotiation-rule violation is detected, then the DLE shall

A) encode an EC DLPDU not requesting a reply, with three addresses as specified in 6.1 and 7.1, which are, respectively,

– the first of the two source DL-addresses from the received EC DLPDU which resulted in the DL-CONNECT indication and its consequent DL-CONNECT response;

– the DLCEP-address of the existing DLC;

– the DLSAP-address associated with this responding DLCEP-address, respectively;

B) queue the DLPDU at TIME-AVAILABLE priority as specified in 8.4.5;

C) stop the timer which was started in 8.2.1.4.2b)4)iv).

iii) Otherwise, when i) does not apply because the specified DLCEP is a subscriber DLCEP, or when ii) does not apply because a negotiation rule violation was detected, then the DLE shall reject the received DLC-establishment request and terminate processing of the received EC DLPDU, as follows:

If the destination DL-address of the received EC DLPDU was not a group DL-address, then

A) The DLE shall encode a DC DLPDU as specified in 6.2 and 7.2, with a reason of “connection rejection – QoS not available, permanent condition”, and schedule the DLPDU for transmission at TIME-AVAILABLE priority as specified in 8.4.5.

B) The DC DLPDU shall have both destination and source addresses (see 6.2.1 formats 1L and 1S), the destination address shall be identical to the first source DLCEP-address of the received EC DLPDU, which resulted in the DL-CONNECT indication and its consequent DL-CONNECT response, and the source address shall be identical to the destination DL-address of that received EC DLPDU.

2) If the existing DLCEP’s DLCEP-class is PEER, then the DLE shall consider the DL-CONNECT response to be a request for resolving a DL-CONNECT request collision on a peer DLC which is in the WAITING-FOR-EC-DLPDU state. The DLE shall apply the negotiation rules of 8.2.1.1 jointly to the QoS parameters of the two DLCEPs and shall reflect the results in the QoS of the originally-requested DLCEP.

f a negotiation rule violation is detected, or if the DLCEP specified by the responding-address is not in the WAITING-FOR-EC-DLPDU state, then both DLCEPs shall be disconnected as specified in 8.2.1.8, but with a reason of "connection rejection – QoS not available, permanent condition”.

Otherwise, when no negotiation-rule violation was detected, then the DLE shall

i) use the source DLCEP-address and DLSAP-address, of the received EC DLPDU, which resulted in the DL-CONNECT indication and its consequent DL-CONNECT response, as the remote DLCEP-address and DLSAP-address of the DLCEP which was in the WAITING-FOR-EC-DLPDU state;

ii) consider the reply-requested field of the received EC DLPDU, which resulted in the DL-CONNECT indication and its consequent DL-CONNECT response, to have been set;

iii) cause the DLCEP specified as the responding-address to

– send a three-address EC DLPDU specifying that a reply is not requested;

– stop the timer which was started in 8.2.1.4.2b)4)iv);

– start a timer as in 8.2.1.2b), with a duration equal to the value for the maximum confirm delay on DL-CONNECT as specified in the DL-CONNECT response primitive;

– activate recognition of the DLCEP’s local DLCEP-address;

– change the specified DLCEP’s state, V_C(ST), to WAITING-FOR-CONNECT-COMPLETION.

e) If the responding address identifier in the DL-CONNECT response was a DLSAP-address, then

– that DLSAP-address shall be used as the local DLSAP-address;

– the DLE shall determine the local maximum send and receive window sizes based on the respective buffer-and-queue bindings, possibly further restricted by DL-management, as specified in 8.2.1.2.

The DLE shall then determine the actual maximum send window size as the smaller of the local send window size and the received EC DLPDU's receive window size, and the actual maximum receive window size as the smaller of the local receive window size and the received EC DLPDU's send window size, as specified in 8.2.1.1. The DLE also shall perform all other required negotiations, as specified in 8.2.1.1.

f) If the optional DLCEP-address was specified in the response primitive, then the DLE shall assign that DLCEP-address to the DLCEP; if any conflicting assignment is detected, then the DLCEP shall be disconnected as specified in 8.2.1.8 with a reason of "disconnection – incorrect DLCEP pairing, permanent condition", Otherwise a DLCEP-address not currently in use shall be assigned to the DLCEP; where possible this shall be a SHORT DL-address of the local link.

   When the DLE is serving only as a subscriber in the DLC, then no reply EC DLPDU is permitted, and so no assignment of a DLCEP-address is required.

   NOTE After DLCEP establishment is completed, a subscriber substitutes its calling-DLSAP-address where a sending DLCEP-address otherwise would be required in a CA, CD, ED or DT DLPDU.

g) If the responding DLCEP class is SUBSCRIBER, then the DLE shall
   – accept the next received sequence number of the DLC as the sequence number of the first DLSDU;
   – stop the timer which was started in 8.2.1.4.2b)4)iv);
   – issue a DL-CONNECTION-ESTABLISHED indication;
   – activate recognition of the DLCEP's remote (publisher) DLCEP-address;
   – change the DLCEP's state, $V_C(ST)$, to DATA-TRANSFER-READY.

h) Otherwise, if g) does not apply, then the DLE shall
   1) encode an EC DLPDU not requesting a reply, with three addresses as specified in 6.1 and 7.1, where its addresses are, respectively,
      – the first of the two source DL-addresses from the received EC DLPDU which resulted in the DL-CONNECT indication and its consequent DL-CONNECT response;
      – the DLCEP-address just assigned to the DLCEP;
      – the responding DLSAP-address, respectively;
   2) assign a publisher-DLCEP-address reuse-discriminator when the responding DLCEP-class is PUBLISHER;
   3) schedule the DLPDU for transmission at TIME-AVAILABLE priority as specified in 8.4.5.

      NOTE This procedure causes a publisher DLCEP to reply to an EC DLPDU from a proposed subscriber DLCEP by sending an EC DLPDU directly to that subscriber DLCEP's temporary calling DLCEP-address. Alternatively, the publishing DLE can create an independent publisher DLCEP (through use of a DL-CONNECT request) after receiving the DL-CONNECT indication from the proposed subscriber, and then merge the subscriber's requested publishing DLCEP into the one just established. This latter approach will cause an EC DLPDU to be sent to the publisher's Called DL(SAP)-address before the response EC DLPDU is sent to the requesting subscriber.

i) If the responding DLCEP class is PUBLISHER, then the DLE shall
   – stop the timer which was started in 8.2.1.4.2b)4)iv);
   – issue a DL-CONNECTION-ESTABLISHED indication;
   – activate recognition of the DLCEP's local DLCEP-address;
   – change the DLCEP's state, $V_C(ST)$, to DATA-TRANSFER-READY.

j) If the responding DLCEP class is PEER, then the DLE shall
   – stop the timer which was started in 8.2.1.4.2b)4)iv);
   – start a timer as in 8.2.1.2b), with a duration equal to the value for the maximum confirm delay on DL-CONNECT as specified in the DL-CONNECT response primitive;
   – activate recognition of the DLCEP's local DLCEP-address;
   – change the DLCEP's state, $V_C(ST)$, to WAITING-FOR-CONNECT-COMPLETION.

8.2.1.4 Receipt of an EC DLPDU

When the DLE receives an EC DLPDU, the DLE shall determine the version number of the DL-protocol in use, as specified in the received EC DLPDU, and shall interpret the other EC-parameters of the DLPDU accordingly.

8.2.1.4.1 Receipt of an EC DLPDU with two addresses

The DLE shall perform the following series of actions, and if any error is detected during the process, then the DLC shall be disconnected as specified in 8.2.1.8:

a) If the first (source) address of the received EC DLPDU is a PUBLISHER DLCEP-address for an existing SUBSCRIBER DLCEP, and if that SUBSCRIBER DLCEP is in the WAITING-FOR-EC-DLPDU state, then the DLE shall

   i) validate the received DLC parameters against those earlier requested for the SUBSCRIBER DLCEP (and possibly not sent in an EC DLPDU);

   ii) set the parameters of the existing DLCEP to equal the received DLC parameters, if permitted by the negotiation rules of 8.2.1.1.

   If an error is detected, then the DLE shall disconnect the SUBSCRIBER DLCEP as specified in 8.2.1.8, with a reason of "connection rejection – QoS not available, permanent condition".

   If no error is detected during this validation, then the DLE shall

   iii) set the publisher-DLCEP-address reuse-discriminator equal to that in the received EC DLPDU;

   iv) cancel the user-request timer T_U(MCD);

   v) accept the next received sequence number of the DLC as the sequence number of the first DLSDU;

   vi) issue a DL-CONNECT confirm to the DLS-user;

   vii) change the SUBSCRIBER DLCEP to the DATA-TRANSFER-READY state.

b) When the first (source) address of the received EC DLPDU is a PUBLISHER DLCEP-address for an existing SUBSCRIBER DLCEP, but a) does not apply because the SUBSCRIBER DLCEP is not in the WAITING-FOR-EC-DLPDU state, then the DLE shall compare the publisher-DLCEP-address reuse-discriminator of the existing DLCEP with that in the received EC DLPDU. If the two values are equal then the DLE shall set the address size of the existing DLCEP to that in the received EC DLPDU.

c) Otherwise, when a) and b) do not apply, then the existing SUBSCRIBER DLCEP shall be disconnected as specified in 8.2.1.8, with a reason of "disconnection – wrong publisher-DLCEP-address reuse-discriminator, permanent condition".

8.2.1.4.2 Receipt of an EC DLPDU with three addresses

The DLE shall perform the following series of actions, and if any error is detected during the process, then the DLC shall be disconnected as specified in 8.2.1.8:

a) If the first address in the received EC DLPDU is a group-DL-address associated with more than one of the DLE's DLS-users, then the DLE shall treat each of those DLS-users as if that user had individually received the EC DLPDU. However, no DC DLPDU shall be sent as a direct response to the received EC DLPDU.

b) If the first address of the received EC DLPDU is a DLSAP-address, then

   1) the DLE shall validate the self-consistency of the received EC DLPDU, where all static and non-specified dynamic parameters assume the default values associated with that called-DLSAP-address, and where any parameter in violation of a local DL-management-imposed limit shall be set equal to that limit, if permitted by the negotiation rules of 8.2.1.1, or the DLCEP shall be disconnected as specified in 8.2.1.8 with a reason of "connection rejection – QoS not available, permanent condition";

2) the DLE shall check whether a DLS-user associated with that DLSAP-address has an active DLCEP whose remote DLCEP-address equals the source DLCEP-address specified in the received EC DLPDU;

3) if such an active DLCEP exists, then

   i) if

      – the DLCEP is in the waiting-for-connect-completion state;

      – the received EC DLPDU requests a reply

   then a return EC DLPDU, addressed to the first source DL-address specified in the received EC DLPDU, with source addresses equal to the DLCEP's local DLCEP- and DLSAP-addresses, and specifying the parameters of the active DLCEP, and not requesting a reply, shall be encoded and shall be queued at TIME-AVAILABLE priority as specified in 8.4.5;

   ii) else when i) does not apply, then the DLE shall disconnect the DLCEP as specified in 8.2.1.8 with a reason of "connection rejection – inconsistent DLCEP state, permanent condition";

NOTE This reconfirmation of the existing DLC does not cause a change in the responding DLCEP's state.

4) if no such active DLCEP exists, then the DLE shall assign a new DLCEP identifier to the DLCEP, and shall apply the negotiation rules of 8.2.1.2d). If any violation of the negotiation rules occurs, then the DLE shall disconnect the proposed DLCEP as specified in 8.2.1.8 with a reason of "connection rejection – QoS not available, permanent condition". If no violation is detected, then for each DLS-user associated with the DLSAP-address or group DL-address which was the first address of the received EC DLPDU, the DLE shall

   i) create a DLCEP, initializing its $V_S(NP)$, $V_S(N)$, $V_S(R)$, $V_S(A)$, $V_S(M)$, $V_S(MS)$, $V_S(H)$, $V_S(HS)$ and $V_S(L)$ variables as specified in 4.7.44.7.4;

   ii) record the source DLCEP-address and source DLSAP-address from the received EC DLPDU as the DLCEP's remote DLCEP-address and remote DLSAP-address, respectively and when sender's DLCEP class is PUBLISHER, also record the publisher-DLCEP-address reuse-discriminator of the EC DLPDU as the DLCEP's local publisher-DLCEP-address reuse-discriminator;

   iii) report a DL-CONNECT indication to the DLS-user;

   iv) start a timer to monitor for the DLS-user's response to the DL-CONNECT indication, as specified in 8.2.1.2b);

   v) change the DLCEP state, $V_C(ST)$, to WAITING-FOR-CONNECT-RESPONSE.

c) Else if the first address of the received EC DLPDU is a DLCEP-address for an existing DLCEP, and if the addressed DLCEP is in the WAITING-FOR-EC-DLPDU state, then the DLE shall validate the received DLC parameters, and the received DLPDU's source DL-addresses when their expected values are known, against those sent in an earlier EC DLPDU, and if an error is detected, then

   1) if the called address of the DLCEP was a group DL-address, then the DLE shall reply with a DC DLPDU not requesting a reply, addressed to the first source DL-address specified in the received EC DLPDU, with a source address equal to the DLCEP's local DLCEP-address, and shall otherwise ignore the received EC DLPDU.

   2) otherwise, when the called address of the DLCEP was a DLSAP-address, then the DLE shall disconnect the DLCEP as specified in 8.2.1.8 with a reason of "connection rejection – QoS not available, permanent condition".

   If no error is detected during the validation of the received EC DLPDU, then

   3) If the receiving DLCEP's DLCEP-class is PEER and the received EC DLPDU requests a reply, then

      i) the two source DL-addresses of the received EC DLPDU shall be noted as the remote-DLCEP-address and remote-DLSAP-address of the DLCEP;

ii) a return EC DLPDU, addressed to the first source DL-address specified in the received EC DLPDU, with source addresses equal to the DLCEP's local DLCEP- and DLSAP-addresses, and specifying the parameters of the active DLCEP, and not requesting a reply, shall be encoded and shall be queued at TIME-AVAILABLE priority as specified in 8.4.5;

iii) the DLE shall start a timer as specified in 8.2.1.2b) with a duration equal to the value for the maximum confirm delay on DL-CONNECT as specified in the DL-CONNECT request primitive;

iv) the DLE shall change the state to WAITING-FOR-CONNECT-COMPLETION.

4) Else when the receiving DLCEP's DLCEP-class is SUBSCRIBER or the received EC DLPDU does not request a reply, then

i) if the receiving DLCEP's DLCEP-class is PEER or SUBSCRIBER , then the two source DL-addresses of the received EC DLPDU shall be noted as the remote-DLCEP-address and remote-DLSAP-address of the DLCEP;

ii) if the DLCEP-class of the receiving DLCEP is PEER, then

– a DT DLPDU not containing DLS-user data;

– with a destination address equal to the first source DL-address specified in the received EC DLPDU;

– when the DLCEP's attributes require the DLPDU to have a source address, with a source address equal to the DLCEP's local DLCEP-address

shall be encoded and shall be queued at the DLCEP's priority as specified in 8.4.5, to notify the peer DLE of the successful receipt of the confirming EC DLPDU

iii) if the receiving DLCEP's DLCEP-class is SUBSCRIBER then the DLE shall accept the next received sequence number of the DLC as the sequence number of the first DLSDU;

iv) the DLE shall issue a DL-CONNECT confirm primitive, conveying the negotiated DLCEP-attributes, to the requesting DLS-user;

v) the DLE shall cancel the user-request timer T_U(MCD) and change the DLCEP state to DATA-TRANSFER-READY.

d) Else, if the first address of the received EC DLPDU is a DLCEP-address for an existing DLCEP, and if the addressed DLCEP is in the WAITING-FOR-CONNECT-COMPLETION state, then the DLE shall validate the received DLC parameters against those sent in an earlier EC DLPDU, and if an error is detected, then the DLE shall disconnect the DLCEP as specified in 8.2.1.8 with a reason of "connection rejection – QoS not available, permanent condition".

If no error is detected during this validation, then

1) if the received EC DLPDU requests a reply, then

i) a return EC DLPDU, addressed to the first source DL-address specified in the received EC DLPDU, with source addresses equal to the DLCEP's local DLCEP- and DLSAP-addresses, and specifying the parameters of the active DLCEP, and not requesting a reply, shall be encoded and shall be queued at TIME-AVAILABLE priority as specified in 8.4.5;

ii) the DLE shall restart the T_U(MCD) timer with the same period as the previous time.

2) else, when 1) does not apply, then

i) a DT DLPDU not containing DLS-user data,

– with a destination address equal to the first source DL-address specified in the received EC DLPDU;

– if the DLCEP's attributes require the DLPDU to have a source address, with a source address equal to the DLCEP's local DLCEP-address

shall be encoded and shall be queued at the DLCEP's priority as specified in 8.4.5, to notify the peer DLE of the successful receipt of the confirming EC DLPDU;

ii) if the state before WAITING-FOR-CONNECT-COMPLETION was WAITING-FOR-EC-DLPDU, then the DLE shall issue a DL-CONNECT confirm primitive, conveying the negotiated DLCEP-attributes, to the requesting DLS-user;

iii) else, when ii) does not apply, then the state before WAITING-FOR-CONNECT-COMPLETION was WAITING-FOR-CONNECT-RESPONSE, and the DLE shall issue a DL-CONNECTION-ESTABLISHED indication primitive to the responding DLS-user;

iv) the DLE shall cancel the user-request timer T_U(MCD) and change the DLCEP state to DATA-TRANSFER-READY.

e) Else if the first address of the received EC DLPDU is a DLCEP-address for an existing DLCEP, and the received EC DLPDU requests a reply, and if the addressed DLCEP is in the DATA-TRANSFER-READY state, then

1) if the existing DLCEP is a publisher DLCEP, then the DLE shall

i) set each QoS parameter, and the publisher-DLCEP-address reuse-discriminator, equal to the corresponding parameter of the specified DLCEP, if permitted by the negotiation rules of 8.2.1.1;

ii) if necessitated by the rule of 8.2.1.1d), then change the address size of the existing DLC from VERY-SHORT to SHORT or from SHORT to LONG.

2) If no negotiation-rule violation is detected, then the DLE shall

i) encode an EC DLPDU not requesting a reply, with two addresses as specified in 6.1 and 7.1, where its addresses are, respectively,

– the DLCEP-address of the existing DLC;

– the DLSAP-address associated with this existing DLCEP-address, respectively;

ii) schedule the EC DLPDU for transmission at TIME-AVAILABLE priority as specified in 8.4.5.

3) When 2) does not apply, because a negotiation rule violation was detected, then the DLE shall reject the received DLC-establishment request and terminate processing of the received EC DLPDU, as follows:

i) The DLE shall encode a DC DLPDU as specified in 6.2 and 7.2, with its reply-requested field set to FALSE, with a reason of “provider-originated disconnection – QoS not available, permanent condition”, and schedule the DC DLPDU for transmission at TIME-AVAILABLE priority as specified in 8.4.5.

ii) The DC DLPDU shall have both destination and source addresses (see 6.2.1 formats 1L and 1S), the destination address shall be identical to the first source DL-address of the received EC DLPDU, and the source address shall be identical to the destination DL-address of that received EC DLPDU.

f) Otherwise, the DLE shall ignore the received EC DLPDU.

8.2.1.5 Expiration of the timer T_U(MCD)

If the timer T_U(MCD) expires, then if the DLCEP state, V_C(ST), is

a) waiting-for-EC-DLPDU, then

1) if this is the (V(NRC)+1)’th consecutive expiration, then

NOTE See the permission at the start of Clause 8 with regard to use of V(NRC) versus V(MRC).

i) the DLE shall terminate processing of the request;

ii) if the user-specified maximum confirm delay on the DL-CONNECT request primitive specified a value other than UNLIMITED, then

A) the DLE shall initiate a DL-DISCONNECT indication reporting “connection rejection – DLSAP unreachable, transient condition, local origin”;

B) if the called address was either a DLSAP-address or a DLCEP-address, and the DLCEP’s DLCEP-class is PEER, then the DLE

– shall encode a DC DLPDU requesting disconnect, with a reason of "reason unspecified", to the same DL-address as that to which the previous EC DLPDU had been sent;

– shall be queued at TIME-AVAILABLE priority as specified in 8.4.5;

iii) otherwise, when the user-specified maximum confirm delay on the DL-CONNECT request primitive specified a value of UNLIMITED, the DLE shall leave the request queued, and the DLCEP in its current state, but shall initiate no other action with respect to the request.

NOTE The DLE may still respond to receipt of an EC DLPDU which attempts to establish a connection with the above DLCEP, by either accepting or rejecting the proffered connection.

2) otherwise, if 1) does not apply, then the DLE shall

i) restart the T_U(MCD) timer with the same period as the previous time;

ii) requeue the same EC DLPDU for retransmission at TIME-AVAILABLE priority as specified in 8.4.5;

b) WAITING-FOR-CONNECT-RESPONSE, then the DLE shall disconnect the DLCEP as specified in 8.2.1.8, specifying a disconnect reason of "provider-originated disconnection – timeout";

c) waiting-for-connect-completion, then

1) if this is the (V(NRC)+1)'th consecutive expiration, then the DLE shall disconnect the DLCEP as specified in 8.2.1.8, specifying a disconnect reason of "provider-originated disconnection – timeout";

NOTE See the permission at the start of Clause 8 with regard to use of V(NRC) versus V(MRC).

2) otherwise, if 1) does not apply, then the DLE shall resend the three-address EC DLPDU which was sent when the WAITING-FOR-CONNECT-COMPLETION state was entered, and again restart the T_U(MCD) timer as specified in 8.2.1.2b).

d) DATA-TRANSFER-READY, then the DLE shall act as specified in 8.2.2.10;

e) WAITING-FOR-RESET-COMPLETION, then the DLE shall act as specified in 8.2.2.18.

8.2.1.6 Receipt of a DL-DISCONNECT request primitive

When the DLE receives at a DLCEP a DL-DISCONNECT request from a DLS-user, then the DLE

a) shall encode a DC DLPDU as specified in 6.2 and 7.2, requesting disconnect and specifying the DLS-user-given reason, and shall schedule the DLPDU for transmission at TIME-AVAILABLE priority as specified in 8.4.5, except when the DLCEP

1) is in the WAITING-FOR-CONNECT-RESPONSE state, and the destination DL-address of the EC DLPDU which activated the DLCEP is a group DL-address; or

2) is in the WAITING-FOR-EC-DLPDU state, and the called DL(SAP)-address is a group DL-address or is UNKNOWN; or

3) is a SUBSCRIBER DLCEP;

If a DC DLPDU is encoded, then

i) if the DLCEP being disconnected is a PEER DLCEP, then the DC DLPDU shall have both destination and source addresses (see 6.2.1 formats 1L and 1S), and the destination address shall be the remote DLCEP-address of the DLC, if known, or the called-DL(SAP)-address of the initiating EC DLPDU in all other cases. The reply-requested field shall be set to TRUE in the DC-parameters of the initiating DC DLPDU.

ii) if the DLCEP being disconnected is a PUBLISHER DLCEP, then the DC DLPDU shall have only a source address (see 6.2.1 formats 2L and 2S). The reply-requested field shall be set to FALSE in the DC-parameters of the initiating DC DLPDU.

iii) the source address of the DC DLPDU shall be the local DLCEP-address, if one exists; or the responding or calling local DLSAP-address, if one exists, or the called-DLSAP-address of the initiating EC DLPDU in all other cases.

b) shall terminate the DLCEP, including
   1) for each outstanding (that is, not-yet-confirmed) DL-DATA request:
      i) remove the request from the appropriate DLCEP user-request queue, Q_A(UR), and references to the request from all DLE queues;
      ii) initiate a DL-DATA confirm with the associated request identifier reporting "failure – reset or disconnection";
      iii) delete the timer T_U(MCD) associated with the request.
   2) if an DL-RESET request is outstanding, then initiate a DL-RESET confirm with the DLS-user's DLCEP-identifier reporting "failure – disconnection";
   3) delete all timers associated with the DLCEP.

After disconnection, if a local DLCEP-address associated with the DLCEP was included in any DLPDU sent on the link, then the DLE shall ensure that the DLCEP-address is not reused for a period of time exceeding the greater of

1) twice the maximum DLPDU lifetime in the network, V(NDL);
2) the DLCEP's D...D_{CR} delay, when that delay is not UNLIMITED.

In the state machine of Figure 11, this procedure is modeled as entry into an extra state, Aging-DLCEP-address, which is exited at the end of the above-specified period of time.

8.2.1.7 Receipt of a DC DLPDU

When the DLE receives a DC DLPDU, specifying that the DLCEP should be disconnected, then

a) the DLE shall determine the version number of the DL-protocol in use, as specified in the received DC DLPDU, and shall interpret the other DC-parameters of the DLPDU accordingly;
b) if the received DC DLPDU requests a reply, then a return DC DLPDU, addressed to the source DL-address specified in the received DC DLPDU, and specifying a disconnect reason of "disconnection or connection rejection, unknown origin – reason unspecified", and not requesting a reply, shall be encoded and shall be queued at TIME-AVAILABLE priority as specified in 8.4.5;
c) if the received DC DLPDU
   1) specifies only a source address (see 6.2.1 formats 2L and 2S) and the source address is a DLCEP-address of a multi-peer DLC to which the DLE is a subscriber; or
   2) specifies both destination and source addresses (see 6.2.1 formats 1L and 1S), where
      i) the destination address is a DL(SAP)-address, and the DLE has a DLCEP, at a DLSAP to which that DL(SAP)-address is bound, whose remote DLCEP-address has the same value as the received source DL-address; or
      ii) the destination address is a DLCEP-address, and the remote DLCEP-address of the identified DLCEP has the same value as the received source DL-address; or
      iii) the destination address is a DLCEP-address, and the called DLSAP-address of the identified DLCEP has the same value as the received source DL-address

   then if the DLCEP is known to the local DLS-user, then
   3) the DLE shall report a DL-DISCONNECT indication to the local DLS-user specifying both the non-local origin and the reason for the DL-DISCONNECT indication as received in the DC DLPDU;
   4) the DLE shall terminate the DLCEP as specified in 8.2.1.6b);
   5) after disconnection, the DLE shall ensure that any DLCEP-address which had been assigned to the DLCEP is not reused for a period of time exceeding the greater of
      i) twice the maximum DLPDU lifetime in the network, V(NDL);
      ii) the DLCEP's D...D_{CR} delay, when that delay is not UNLIMITED.

8.2.1.8 DLE-initiated disconnection

When the DLE determines on its own that it is necessary to disconnect the DLCEP, then

a) if the DLCEP is known to the local DLS-user, then the DLE shall report a DL-DISCONNECT indication to the local DLS-user, specifying both the reason for the DL-DISCONNECT indication and that its origin was local.

   NOTE The DLCEP will not be known to the local DLS-user if the disconnection occurs while processing a received EC DLPDU whose receipt had just triggered the DL to create the DLCEP.

b) If

   1) the DLCEP's DLCEP-class is PEER or PUBLISHER;

   2) the called DL(SAP)-address of the EC DLPDU which activated the DLCEP was not a group DL-address,

   then

   i) the DLE shall encode a DC DLPDU as specified in 6.2 and 7.2, and shall schedule the DLPDU for transmission at TIME-AVAILABLE priority as specified in 8.4.5;

   ii) if the DLCEP being disconnected is a PEER DLCEP, then the DC DLPDU shall have both destination and source addresses (see 6.2.1 formats 1L and 1S), and the destination address shall be the remote DLCEP-address of the DLC, if known, or the called-DL(SAP)-address of the initiating EC DLPDU in all other cases. The reply-requested field shall be set to TRUE in the DC-parameters of the initiating DC DLPDU;

   iii) if the DLCEP being disconnected is a PUBLISHER DLCEP, then the DC DLPDU shall have only a source address (see 6.2.1 formats 2L and 2S) and the reply-requested field shall be set to FALSE in the DC-parameters of the initiating DC DLPDU;

   iv) the source address of the DC DLPDU shall be the local DLCEP-address, if one exists; or the responding or calling local DLSAP-address, if one exists, or the called-DLSAP-address of the initiating EC DLPDU in all other cases.

c) the DLE shall terminate the DLCEP as specified in 8.2.1.6b).

After disconnection, the DLE shall ensure that any DLCEP-address which had been assigned to the DLCEP is not reused for a period of time exceeding the greater of

1) twice the maximum DLPDU lifetime in the network, V(NDL);
2) the DLCEP's D…D_{CR} delay, when that delay is not UNLIMITED.

8.2.2 Operation of the DLC data transfer and DLCEP reset services

The primitives of the DLC data transfer service are DL-DATA request, indication and confirm, DL-BUFFER-SENT indication, and DL-BUFFER-RECEIVED indication. The primitives of the DLCEP reset services are DL-RESET request, indication, response and confirm, and DL-RESET-COMPLETED indication. A DLCEP reset may be invoked, by any DLS-user or the DLS-provider, at any time after DLCEP establishment and before DLCEP release.

If the sending DLCEP data delivery features in either direction of the DLC are ORDERED, DISORDERED, or CLASSICAL, then the linear (conceptually unlimited) DLCEP variables specified in 4.7.4 and Figure 7 are projected onto a cyclic ring of 2^N sequence identifiers as shown in Figure 12. Each DLSDU is associated with one of these identifiers prior to transmission, and all DLPDUs which convey segments of the DLSDU also specify that associated identifier. The dynamic partitioning of these identifiers, and the resultant categorization of DLSDUs associated with these identifiers, is also shown in Figure 12.

NOTE The relationships among the linear (conceptually unlimited) DLCEP variables specified in 4.7.4 and Figure 7 are repeated here to assist in the understanding of Figure 12.

$H–W \le M–W \le A \le L \le H \le M \le A+W \le L+W$

M–W ≤ R ≤ M

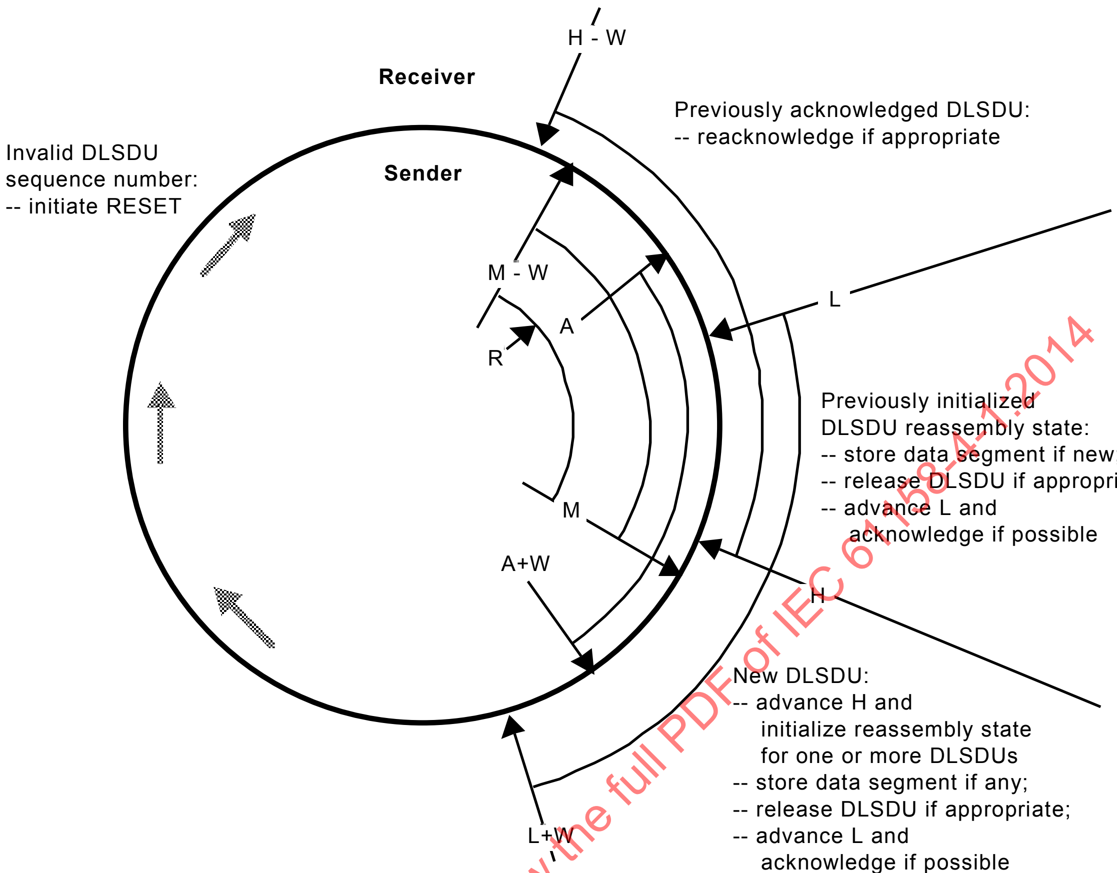


Figure 12 – Projection of the sending and receiving DLCEP sequence-number variables of Figure 7 onto the cyclic sequence-number parameters of CA, CD, DT, ED and RC DLPDUs, with consequent determination of required actions

Each DLPDU with non-null SD-parameters can convey to the peer DLE the low-order 2 to 5 bits of the sender's $V_C(L)$ and $V_C(M)$, as the (NDR mod 2^R) and (NDS mod 2^S) subfields of the SD-parameters.

8.2.2.1 Selection of the format of a CA, CD, DT and ED DLPDUs

The address format of all CA, CD, ED, and DT DLPDUs sent from a DLCEP shall be chosen as determined during the DLCEP-establishment process (see 8.2.1.1) and as specified in 6.4.3, 6.5.3 , 6.6.3 and 6.7.3, respectively. The SD-parameter format of all such CA, CD, ED and DT DLPDUs formed by the DLE shall be the same as that negotiated for the sending DLCEP (see 7.1c)5), 7.1d)5), and 8.2.1.1).

All CD DLPDUs sent from an LAS DLE as part of its schedule execution activities, and not from a DLCEP of the LAS DLE, shall specify an explicit destination address of the length negotiated in 8.2.1.1 and shall omit both source address and SD-parameters.

NOTE An address format of VERY-SHORT is always realized by use of SHORT addresses in any associated CA, CD, DC, ED and RC DLPDUs.

8.2.2.2 Receipt of a DL-DATA request primitive

If the request is accepted, as indicated by a returned status of "success" for the DL-DATA request, then upon completion of the request, either successfully or after failure, the DLE shall issue a DL-DATA confirm with the same request identifier as specified by the DLS-user in the corresponding DL-DATA request primitive, conveying the status of the request to the DLS-user.

The DLCEP source specified in the DL-DATA request should be bound to either an explicit (user-controlled) queue or to an implicit (DLE-controlled) queue. If the queue is full, or if the specified DLSDU length, $P_U(L)$, is invalid, or if the DLCEP-state $V_C(ST)$ is not DATA-TRANSFER-READY, then the DLE shall immediately return the corresponding DL-DATA confirm indicating the reason for failure.

Otherwise

a) the DLE shall create and start a user-request timer $T_U(MCD)$ with a duration based on the user-specified maximum confirm delay for DL-DATA primitives. If the specified value was other than UNLIMITED, then the duration of this timer shall be equal to that user-specified maximum confirm delay; otherwise the duration should be 60 s. DL-management may override these preferred durations;
b) the DLE shall assign the next unassigned sequence number $N = V_C(N)$ to the request and its associated DLSDU;
c) the DLE shall initialize the variable $V_{C,N}(SS)$ based on the length, $P_N(L)$, of the Nth DLSDU, to indicate that all segments of the Nth DLSDU, and no other segments of that DLSDU, need transmission;
d) the DLE shall append the request to the DLCEP-address's user-request queue, $Q_A(UR)$, as follows.
   1) If the DL-scheduling-policy attribute of the source DLCEP-address is EXPLICIT, then the request shall be placed in the third partition of $Q_A(UR)$, where it shall await a DL-COMPEL-SERVICE request to release it for transmission.

      NOTE Release of a deferred request applies to DLSDUs, not DLPDUs.

   2) Else if 1) does not apply, and
      i) if $N > V_C(A) + P_C(WS)$, and the sending DLCEP is a CLASSICAL or DISORDERED peer, or
      ii) if $N > V_C(R) + P_C(WS)$, and the sending DLCEP is ORDERED and $V_C(R)$ exists, or is a CLASSICAL or DISORDERED publisher or subscriber,

      then the request shall be placed in the third partition of $Q_A(UR)$.

   3) Else if 1) and 2) do not apply, then the third partition of $Q_A(UR)$ is empty, and so the request shall be placed in the second partition of $Q_A(UR)$, and the DLE shall append to the DLE's unscheduled-service queue, Q(US), a reference to $Q_A(UR)$ of the same priority as the just-appended request.

      NOTE Q(US) never needs to have more references to a $Q_A(UR)$ than the number of DLSDUs waiting for transmission or retransmission.

e) The DLE shall increment $V_C(N)$.

8.2.2.3 Transmission of a DT DLPDU from a DLCEP

In the following, let MUD be defined as the maximum number of user data octets which may be conveyed in a single DT DLPDU of the DLCEP's priority, as specified in 4.2.3.

Upon receipt of a transmission opportunity for a queued or buffered DLSDU, the DLE shall form and send a DT DLPDU of the specified priority; with DL-address field and SD-parameter field formats as specified in 6.7, 7.4 and 8.2.2.1; with the remote (destination) and local (source) DLCEP-addresses of the DLC, as appropriate; with user data-field length and contents, and SD-parameter-field contents, as follows:

NOTE With the exception of the final-token-use subfield of the DT DLPDU's first (frame control) octet; probably of the NDR, RSN, J, K and T subfields, if present, of the SD-parameter format negotiated for transmission from the sending DLCEP, the remainder of the data DT DLPDU may be formed when the request is queued, and need not be formed dynamically at the moment of transmission.

8.2.2.3.1 Formation of the user data field and related SD-parameter subfields

The T, NDS, TNS, ASN and truncated-DL-time subfields of the SD-parameters field, and the DLPDU's user data field, shall be formed as follows.

a) If

i) the sending DLCEP is a DISORDERED or CLASSICAL PUBLISHER DLCEP, or an ORDERED PEER or PUBLISHER DLCEP and $V_C(R)$ exists;

ii) $V_C(M) = V_C(R)+P_C(WS)$ and $V_C(M) +1 < V_C(N)$ and $V_{C,V_C(M)}(SS)$ is empty;

iii) a DL-DATA confirm primitive has been issued for the $(V_C(R)+1)$'th DL-DATA request

then the DLE shall

1) cancel the timer associated with the $(V_C(R)+1)$'th DL-DATA request;

2) increment $V_C(R)$.

b) If the sending DLCEP is bound to a sending queue, and is

i) a SUBSCRIBER DLCEP, or an UNORDERED PEER or PUBLISHER DLCEP, and there is a smallest K such that $V_C(M) < K \leq V_C(N)-1$ and $V_{C,K}(SS)$ is non-empty;

NOTE In this case $V_{C,K}(SS)$ can contain only a single element, since such DLSDUs cannot require segmentation.

ii) a DISORDERED or CLASSICAL PEER DLCEP, and there is a K such that $V_C(A) < K \leq \min(V_C(A)+P_C(WS), V_C(N)-1)$ and $V_{C,K}(SS)$ is non-empty;

iii) a DISORDERED or CLASSICAL PUBLISHER DLCEP, or an ORDERED PEER or PUBLISHER DLCEP and $V_C(R)$ exists; and there is a K such that $V_C(R) < K \leq \min(V_C(R)+P_C(WS), V_C(N)-1)$ and $V_{C,K}(SS)$ is non-empty; or

iv) an ORDERED PEER or PUBLISHER DLCEP and $V_C(R)$ does not exist; and there is a K such that $V_C(M) \leq K \leq \min(V_C(M)+P_C(WS), V_C(N)-1)$ and $V_{C,K}(SS)$ is non-empty; or

then the DLE shall form the remainder of the DLPDU as follows.

1) The NDS subfield, if present, shall convey the lowest-order three or five bits of the value K, as appropriate.

2) If S is the zero-origin index of the lowest-numbered member of the set $V_{C,K}(SS)$, which is the segment number of a segment of the K'th DLSDU requiring transmission, then the ASN subfield, if present, shall convey the value of S.

NOTE It is possible for an implementation of this DL-protocol to perform DLSDU segmentation at the time of DLPDU transmission. Such an implementation approach could minimize the DLE's processing of DLS-user data.

3) The TNS subfield, if present, shall convey the value $(P_K(SDUL) -1)/MUD$, rounded down, which is the zero-origin number of segments in the K'th DLSDU.

4) The user data field shall consist of octets
$S \times MUD +1$ to $\min((S+1) \times MUD, P_K(SDUL))$, inclusive, of $P_K(SDU)$.

5) The T subfield shall specify FALSE.

6) The truncated-DL-time subfield shall not be present.

the DLE shall remove member number S from $V_{C,K}(SS)$.

c) If the sending DLCEP is bound to a sending buffer, then

1) the DLE shall increment $V_C(N)$.if the buffer has been written since the last transmission from the buffer on this DLCEP;

2) the DLE shall let K equal $V_C(N)-1$;

3) if K is not equal to zero, then the DLE shall form the remainder of the DLPDU as specified in b)1) to b)4) and as follows.
   i) If the DLSDU has no timeliness attribute, then the T subfield shall specify FALSE.
   ii) If the DLCEP has a sender's-DL-timeliness class other than NONE, then the DLE shall
      A) compute the timeliness of the Sth segment of the K'th DLSDU as specified in 8.1.7;
      B) perform a logical AND of that computed timeliness status with the timeliness-status associated with writing the buffer, V_B(TS) (see 4.7.4.21);
      C) convey that result in the T subfield of the DLPDU.
   iii) If present, the truncated-DL-time subfield shall convey the appropriate octets of the time-of-production associated with the buffer.

   the DLE shall remove member number *S* from $V_{C,K}$(SS).

d) If there is no such *K* as in b), or if K is equal to zero in c), or if the DLE is required to send the DLPDU without user data as in 6.4.4.1c), 6.5.4.1c), and 6.6.4.1c); or 6.4.4.1d), 6.5.4.1d), and 6.6.4.1d), then
   1) the T, TNS, ASN and truncated-DL-time subfields of the SD-parameters, if present, shall be encoded as zero (0); NDS shall be encoded as the appropriate number of low-order bits of V_C(M);
   2) the truncated DL-time subfield may be null (that is, omitted and not transmitted);
   3) the user data field shall be null.

8.2.2.3.2 Formation of the other SD-parameter subfields

The J, K, NDR and RSN subfields of the SD-parameters shall be formed as follows:

a) If the sending DLCEP is a PUBLISHER DLCEP, then the J, K, NDR and RSN subfields of the SD-parameters shall be encoded as zero (0).
b) If the sending DLCEP is a PEER or SUBSCRIBER DLCEP, then
   1) if there is a smallest *K* such that $V_C(L) < K \le V_C(H)$ and $V_{C,K}$(RRS) is non-empty, then if *S* is the zero-origin index of the lowest-numbered member of the set $V_{C,K}$(RRS), which is the segment number of a missing segment of the *K*'th DLSDU, and either $K < V_C(H)$ or $S \le V_C(HS)$, then
      i) the J subfield of the SD-parameters, if present, shall be encoded as one (1);
      ii) the NDR subfield, if present, shall convey the lowest-order two or four bits of the value *K*;
      iii) the RSN subfield, if present, shall convey the value of *S*;
      iv) segment *S* shall be removed from $V_{C,K}$(RRS).
   2) Otherwise, if 1) does not apply, either the SD-parameter field shall be null, if permitted by 8.2.2.1; or
      i) the J and RSN subfields shall be encoded as zero (0), if present;
      ii) the NDR subfield, if present, shall convey the lowest-order two or four bits of (V_C(L)+1).
   3) If the K and NDR subfields of the SD-parameters are both present, and the value of the NDR subfield equals the value of the corresponding lowest-order bits of (V_C(L)+1), then the K subfield shall be encoded as one (1); otherwise the K subfield shall be encoded as zero (0).

8.2.2.3.3 Transmission completion

a) If the just-transmitted DLPDU contained DLS-user data, then the DLE shall update V_C(M) and V_C(MS) from the local variables *K* and *S* of 8.2.2.3.1 as follows.

– If K > V_C(M), or K = V_C(M) and S > V_C(MS), then V_C(M) shall be set equal to K, and V_C(MS) shall be set equal to S.

b) If the sending DLCEP

– is a peer DLCEP whose sending DLCEP features are UNORDERED, or is a publisher DLCEP whose sending DLCEP features are UNORDERED, or is a subscriber DLCEP;

– the DT DLPDU has a non-null user data field which contains the last, or only, segment of a DLSDU;

then

1) if the sending source is a buffer, then the DLE shall issue a DL-BUFFER-SENT indication primitive specifying the DLS-user-identifier if known, or the DL-identifier otherwise, for the DLCEP; or

2) if the sending source is a queue and there is an unconfirmed DL-DATA request for that DLSDU, then the DLE shall

i) issue a DL-DATA confirm primitive with the same request identifier as that DL-DATA request, reporting “success”;

ii) cancel and delete the associated timer T_U(MCD).

c) If the sending DLCEP

– is a publisher DLCEP whose sending DLCEP features are ORDERED, DISORDERED or CLASSICAL, or is a peer DLCEP whose sending DLCEP features are ORDERED;

– the DT DLPDU has a non-null user data field which contains the last, or only, segment of a DLSDU;

then

1) if the sending source is a buffer, then the DLE shall issue a DL-BUFFER-SENT indication primitive specifying the DLS-user-identifier, if known, or the DL-identifier otherwise, for the DLCEP; or

2) if the sending source is a queue and there is an unconfirmed DL-DATA request for that DLSDU, then

i) the DLE shall issue a DL-DATA confirm primitive with the same request identifier as that DL-DATA request, reporting “success”;

ii) if V_C(R) exists, and V_C(M) –P_C(WS) > V_C(R), then the DLE shall set V_C(R) equal to V_C(M) – P_C(WS);

iii) the DLE shall retain the DLSDU for potential retransmission to subscribers

– until the expiration of the associated timer T_U(MCD); or

– until the need to transmit another DLSDU with a resultant increase in V_C(M), when coupled with the negotiated window size P_C(WS), requires the discard of the retained DLSDU with a resultant increase in V_C(R).

d) If this DLCEP has been specified as a synchronizing DLCEP during the establishment of one or more other local DLCEPs, and if a DL-BUFFER-SENT indication primitive was issued in b)1) or c)1), then the DLE shall record the DL-time of network access, V_C(TNA), for use in the timeliness computations of those referencing DLCEP(s).

8.2.2.4 Transmission of a CA, CD or ED DLPDU from a DLCEP

8.2.2.4.1 Transmission of a CA DLPDU

Upon receipt of a transmission opportunity to compel transmission from a remote DLCEP, when

a) the DLC is

1) simplex, with DLS-user data transmission only from the local DLCEP to the remote DLCEP; or

2) duplex, with DLS-user data transmission from the local DLCEP to the remote DLCEP, and from that remote DLCEP to the local DLCEP (and possibly other DLCEPs), and the DLCEP attributes do not permit sending an ED DLPDU from the DLCEP to the remote DLCEP;

b) the sending DLE needs to determine the state of the remote DLCEP;

then the DLE shall form and send a CA DLPDU of the specified priority; with DL-address field and SD-parameter field formats as specified in 6.5, 7.4 and 8.2.2.2; with the remote (destination) and local (source) DLCEP-addresses of the DLC, as appropriate; and with SD-parameter-field contents and user data-field contents as specified in 8.2.2.3.

8.2.2.4.2 Transmission of a CD DLPDU

This subclause does not apply to the LAS DLE when it sends CD DLPDUs as part of its scheduled activity and not from a DLCEP of the LAS DLE; such DLPDUs are constrained as specified in 6.5 and 8.2.2.

Upon receipt of a transmission opportunity to compel transmission from a remote DLCEP, when

a) the DLC is
   1) simplex, with DLS-user data transmission only from the remote DLCEP to the local DLCEP (and possibly other DLCEPs); or
   2) duplex, with DLS-user data transmission from the local DLCEP to the remote DLCEP, and from that remote DLCEP to the local DLCEP (and possibly other DLCEPs), and either
      i) the DLCEP attributes do not permit sending an ED DLPDU from the local DLCEP to the remote DLCEP, or
      ii) there are no DLSDU segments awaiting transmission to the remote DLS-user as specified by the selection criteria of 8.4.2.1a);

b) the local execution of a DL-COMPEL-SERVICE request primitive, or of a scheduled compel-service action, compels transmission from a remote peer or publisher DLCEP;

then the DLE shall form and send a CD DLPDU of the specified priority; with DL-address field and SD-parameter field formats as specified in 6.5, 7.4 and 8.2.2.2; with the remote (destination) and local (source) DLCEP-addresses of the DLC, as appropriate; and with SD-parameter-field contents, as follows:

NOTE With the exception of the final-token-use subfield of the CD DLPDU's first (frame control) octet; and probably of the NDR, RSN, J and K subfields, if present, of the SD-parameter format negotiated for transmission from the sending DLCEP, the remainder of the CD DLPDU may be formed when the request is queued, and need not be formed dynamically at the moment of transmission.

1) The T, NDS, TNS, ASN and truncated-DL-time subfields of the SD-parameters, if present, shall be encoded as zero (0).
2) The truncated DL-time subfield may be null (that is, omitted and not transmitted).
3) The user data field shall be null.
4) the J, K, NDR and RSN subfields of the SD-parameters shall be formed as specified in 8.2.2.3.2.

8.2.2.4.3 Transmission of an ED DLPDU

Upon receipt of a transmission opportunity to compel transmission from a remote DLCEP, when

a) the DLC is duplex, with DLS-user data transmission from the local DLCEP to the remote DLCEP, and from that remote DLCEP to the local DLCEP (and possibly other DLCEPs);

b) the DLCEP attributes permit sending an ED DLPDU from the DLCEP to the remote DLCEP;

c) one or more DLSDU segments await transmission to the remote DLS-user as specified by the selection criteria of 8.4.2.1a);

d) the local execution of a DL-COMPEL-SERVICE request primitive, or of a scheduled compel-service action, compels transmission from a remote peer or publisher DLCEP;

then the DLE shall form and send an ED DLPDU of the specified priority; with DL-address field and SD-parameter field formats as specified in 6.5, 7.4 and 8.2.2.2; with the remote (destination) and local (source) DLCEP-addresses of the DLC, as appropriate; and with SD-parameter-field contents and user data-field contents as specified in 8.2.2.3.

8.2.2.5 Validation and processing of SD-parameters in a CA, CD, ED or DT DLPDU received at a DLCEP

If the DLCEP state, $V_C(ST)$, is

1) waiting-for-connect-completion, then
   i) if the state before WAITING-FOR-CONNECT-COMPLETION was WAITING-FOR-EC-DLPDU, then the DLE shall issue a DL-CONNECT confirm primitive, conveying the negotiated DLCEP-attributes, to the requesting DLS-user and cancel the associated user request timer $T_U(MCD)$;
   ii) else, when i) does not apply, then the state before WAITING-FOR-CONNECT-COMPLETION was WAITING-FOR-CONNECT-RESPONSE, and the DLE shall issue a DL-CONNECTION-ESTABLISHED indication primitive to the receiving DLS-user and cancel the associated user request timer $T_U(MCD)$;
   iii) the DLE shall change the DLCEP state, $V_C(ST)$, to DATA TRANSFER READY; and shall apply the remainder of this subclause.
2) waiting-for-reset-completion, then
   i) the DLE shall issue a DL-RESET-COMPLETED indication primitive to the receiving DLS-user specifying the DLS-user-identifier for the DLCEP if known, or the DL-identifier for the DLCEP otherwise;
   ii) the DLE shall cancel the associated user request timer $T_U(MCD)$;
   iii) the DLE shall change the DLCEP state, $V_C(ST)$, to DATA-TRANSFER-READY; and shall apply the remainder of this subclause.
3) not WAITING-FOR-CONNECT-COMPLETION, and not WAITING-FOR-RESET-COMPLETION, and not DATA-TRANSFER-READY, then the received DLPDU shall be ignored by the upper-level DLC functions.

Otherwise, the DLE shall validate and process the SD-parameters of the received DLPDU according to the SD-parameter format, $P_C(NP.FFF_R)$, negotiated for this (receiving) direction of DLC transmission. This validation and processing shall be as specified in the remainder of 8.2.2.5, with format-dependent considerations as follows, based on the SD-parameter format (A – G) and the truncated DL-time format (J – M). The format-dependent value of the sending modulus MOD_S shall also be used in the procedures of 8.2.2.6.

Format A) The sending and receiving SD-parameters of the DLPDU are implicit and thus always valid; the implied values of RSN, T, TNS, ASN and truncated DL-time are all zero; and any accompanying user data is a complete DLSDU. 8.2.2.5.2 does not apply.

Format B) The sending and receiving SD-parameters of the DLPDU are explicit; the sending modulus MOD_S equals 2^3; the receiving modulus MOD_R equals 2^2; the implied values of RSN, TNS and ASN are zero; and any accompanying user data is a complete DLSDU.

Format C) The sending and receiving SD-parameters of the DLPDU are explicit; the sending modulus MOD_S equals 2^5; the receiving modulus MOD_R equals 2^4; and any accompanying user data may be only a partial DLSDU.

Format D) The sending and receiving SD-parameters of the DLPDU are explicit; the sending modulus MOD_S equals 2^5; the receiving modulus MOD_R equals 2^4; the implied values of TNS and ASN are zero; and any accompanying user data is a complete DLSDU.

Format E) The sending SD-parameters of the DLPDU are non-existent; the receiving SD-parameters of the DLPDU are explicit; the receiving modulus MOD_R equals 2^4; the implied values of J, K and T are as specified in 7.4.2.1E); the implied value of NDS is $V_C(H)+1$; the implied values of TNS and ASN are zero; and there cannot be accompanying user data.

Format F) The sending SD-parameters of the DLPDU are explicit; the receiving SD-parameters of the DLPDU are non-existent; the sending modulus MOD_S equals 2^5; the implied value of NDR is $V_C(M)+1$; the implied value of RSN is zero; and any accompanying user data may be only a partial DLSDU.

Format G) The sending SD-parameters of the DLPDU are explicit; the receiving SD-parameters of the DLPDU are non-existent; the sending modulus MOD_S equals 2^5; the implied value of NDR is $V_C(M)+1$; the implied values of RSN, TNS and ASN are zero; and any accompanying user data is a complete DLSDU.

8.2.2.5.1 Validation of the NDS, TNS, ASN and truncated-DL-time subfields of the received SD-parameters

In the following, $P_C(NP.WWWW_R)$ is the negotiated receive window size and $P_C(NP.TT_R)$ is the negotiated receiving DLCEP data delivery features.

a) If $P_C(NP.TT_R)$ specifies UNORDERED, as is always the case with format A, then if the received DLPDU's user data field is non-null, then the receiving DLE
   i) shall increment $V_C(H)$, and shall let *K* equal the new value of $V_C(H)$;
   ii) shall create the variable $V_{C,K}(MRS)$ with a value indicating that segment number zero (0) of the *K*'th DLSDU is missing;
   iii) shall process the received user data as specified in 8.2.2.6.

b) Otherwise, when $P_C(NP.TT_R)$ specifies ORDERED, DISORDERED or CLASSICAL, and if the receiving DLCEP is a subscriber DLCEP, and this is the first DT DLPDU received after the DLCEP state was changed to DATA-TRANSFER-READY, then the DLE shall set the variables $V_C(L)$ and $V_C(H)$ to the value of the $N_R(NDS)$ subfield of the received DT DLPDU.

   The DLE shall compute

$$TEMP = (\ N_R(NDS)\ +\ P_C(NP.WWWW_R)\ -\ V_C(H)\ -1\ )\ \text{modulo}\ MOD_S \tag{22}$$

   1) If

$$TEMP\ >\ (\ V_C(L)\ +\ 2 \times P_C(NP.WWWW_R)\ -\ V_C(H)\ -1\ )\ \text{modulo}\ MOD_S$$

   then
      i) if $P_C(NP.TT_R)$ is ORDERED, then the DLE shall
         A) set $V_C(L)$ equal to
            $V_C(L) + ((\ N_R(NDS)\ -\ (P_C(NP.WWWW_R) + V_C(L)))\ \text{modulo}\ MOD_S)$;
         B) cancel all timers $T_{C,N}(RRS)$ which may exist, where *N* is less than or equal to $V_C(L)$. All DLE resources devoted to reception and reassembly of DLSDUs with sequence numbers less than or equal to $V_C(L)$ should be released;
      ii) if $P_C(NP.TT_R)$ is CLASSICAL or DISORDERED, then the received DLSDU sequence number is invalid; the procedures of 8.2.2.6 do not apply; and the DLE shall initiate a reset at the DLCEP (see 8.2.2.19).

   2) Else if 1)ii) does not apply, then if either

– *TEMP* > (P_C(NP.WWWW$_R$) –1), or

– *TEMP* = (P_C(NP.WWWW$_R$) –1) and N_R(ASN) > V_C(HS)

then

i) the DLE shall set *N* equal to *TEMP* – (P_C(NP.WWWW$_R$) –1);

ii) if *N* > 0, then the received DLSDU sequence number is for a new DLSDU, not previously received or inferred; the DLE shall repeat the following step A) *N* times, followed by step B) once.

A) The DLE shall increment V_C(H). Let *K* equal the just-incremented value of V_C(H). Then $V_{C,K}$(MRS) shall be created and shall indicate that all possible segments of the *K*'th DLSDU, based on the negotiated maximum-DLSDU-size, are missing; and $V_{C,K}$(RRS) shall be created and shall indicate that segment number zero (0) of the *K*'th DLSDU is missing.

NOTE 1 Alternatively, the DLE can simply set $V_{C,K}$(MRS) to indicate that all (16) segments are missing; later procedures will correct the number of missing segments to those of the actual DLSDU.

NOTE 2 This combination of values for $V_{C,K}$(MRS) and $V_{C,K}$(RRS) ensures that all segments of the *K*'th DLSDU will be received before the reassembled DLSDU is delivered to the DLS-user.

NOTE 3 The DLE repeats the above step *N* times.

B) if the received SD-parameters contain an explicit TNS field, then the DLE shall modify $V_{C,K}$(MRS) to indicate that all segments whose zero-origin number is greater than the value of TNS are not missing, where *K* equals the new value of V_C(H) after step A).

iii) if *N* = 0, then the DLE shall set *K* equal the value of V_C(H).

iv) for all values of *N*,

A) the DLE shall set the variable V_C(HS) equal to the value of the N_R(ASN) field. If there is any accompanying user data in the received DLPDU, then the DLE shall modify both $V_{C,K}$(MRS) and $V_{C,K}$(RRS) to indicate that the segment whose zero-origin number is equal to the value of N_R(ASN) field is not missing, and the procedures of 8.2.2.6 also shall be applied.

B) if there is any $V_{C,K}$(RRS), as created in b)2)ii)A), which is not empty and which therefore requires a retransmission request, and if the receiving DLCEP is a CLASSICAL or DISORDERED DLCEP, or optionally is an ORDERED DLCEP, then

– the DLE shall check for a reference to the DLCEP on the DLE's unscheduled-service queue, Q(US);

– if no such reference is found then the DLE shall add a reference to the DLCEP onto the DLE's unscheduled-service queue, Q(US), to ensure that another DLPDU requesting retransmission of the missing segment, is sent from the receiving DLCEP.

3) Else if 1) and 2) do not apply, and
TEMP < (V_C(L) + P_C(NP.WWWW$_R$) – V_C(H)) modulo MOD_S
then the received DLSDU sequence number is for a previously delivered, and on peer DLCs previously acknowledged, DLSDU. If there is any accompanying user data in the received DLPDU, and the DLCEP is a CLASSICAL or DISORDERED peer DLCEP, then the DLE shall check for a reference to the DLCEP on the DLE's unscheduled-service queue, Q(US), and if not found then add a reference to the DLCEP to the DLE's unscheduled-service queue, Q(US), to ensure that another DLPDU reacknowledging the just-referenced DLSDU is sent from the receiving DLCEP.

If the DLCEP's receive binding is explicitly or implicitly to a queue, then the procedures of 8.2.2.6 do not apply. If the DLCEP's receive binding is to a buffer, then any DLS-user data in the DLPDU shall be discarded and the receipt of the duplicate DLPDU shall be reported to the DLS-user with a DL-BUFFER-RECEIVED indication specifying that the reported DLSDU is a duplicate DLSDU.

4) Else if 1), 2) and 3) do not apply, then the received DLSDU sequence number is for a previously received or inferred, but not yet acknowledged, or delivered, or both, DLSDU.

Let $K = V_C(H) + TEMP + 1 – P_C(NP.WWWW_R)$.

If the received SD-parameters contain an explicit TNS subfield, then $V_{C,K}(MRS)$ and $V_{C,K}(RRS)$ both shall be modified to indicate that all segments whose zero-origin number is greater than TNS are not missing. If there is any accompanying user data in the received DLPDU, and $V_{C,K}(MRS)$ indicates that the user data has not previously been received, then the DLE shall modify both $V_{C,K}(MRS)$ and $V_{C,K}(RRS)$ to indicate that the segment whose zero-origin number is equal to the value of $N_R(ASN)$ field is not missing, and the procedures of 8.2.2.6 also shall be applied.

8.2.2.5.2 Validation of the NDR, RSN, J and K subfields of the received SD-parameters

In the following, $P_C(NP.WWWW_R)$ is the negotiated receive window size and $P_C(NP.TT_R)$ is the negotiated receiving DLCEP data delivery features.

a) If the DLCEP is a subscriber DLCEP, and the NDR, RSN, J and K subfields, if present, of the received SD-parameters are not all zero, then the DLE shall disconnect from the DLCEP as specified in 8.2.1.8 with a reason of "provider-originated disconnection – wrong DLPDU format or parameters, permanent condition".

b) If the DLCEP is a CLASSICAL or DISORDERED peer DLCEP, and the J and K subfields of the received SD-parameters are not both zero, then the DLE shall compute

$$TEMP = (N_R(NDR) – V_C(A)) \text{ modulo } MOD_R \tag{23}$$

$$N = TEMP + V_C(A) \tag{24}$$

The received DLPDU is acknowledging a previously unacknowledged transmitted DLSDU (K=1), or requesting retransmission of a segment of a previously transmitted DLSDU (J=1), or both.

If K=1, and the DLCEP is a CLASSICAL or DISORDERED peer DLCEP, and $V_C(A) < N \le V_C(M) + 1$, then the DLE shall

i) set $V_C(A)$ equal to *N-1*;

ii) issue, in the order originally requested, a DL-DATA confirm for each DL-DATA request which was acknowledged by the received NDR;

iii) cancel the set of associated user request timers { $T_U(MCD)$ } for the just-confirmed DL-DATA requests;

iv) cancel any retransmission timers $T_{C,K}(SS)$ associated with the just-confirmed DL-DATA requests, or the simplified timer $T_C(SS)$ associated with the DLCEP, and in this latter case (using $T_C(SS)$), if $V_C(A) < V_C(M)$, which implies that there are unacknowledged DLSDUs, then $T_C(SS)$ shall be restarted;

v) where possible and permitted, move DL-DATA requests from the third partition to the second partition of the corresponding user-request queue, $Q_A(UR)$, as specified in 8.2.2.2d);

if the $V_{C,K}(SS)$ associated with the just-confirmed DL-DATA requests were not empty, then the DLE may cancel such retransmission requests and set the corresponding $V_{C,K}(SS)$ to empty.

If J=1, and *N* is greater than $V_C(A)$, and either

- $N < V_C(M)$, or

- N = $V_C(M)$ and RSN $\le V_C(MS)$,

then the DLE shall add the RSN'th member to the set $V_{C,N}(SS)$; and if the set $V_{C,N}(SS)$ was previously empty, then the DLE shall

– cancel any retransmission timers $T_{C,N}(SS)$ associated with the N'th DLSDU, or $T_C(SS)$ associated with the DLCEP;

– add to the DLE's unscheduled-service queue, Q(US), a reference to $Q_A(UR)$ of the receiving DLCEP, to ensure that the requested DLPDU is sent from the receiving DLCEP.

NOTE Q(US) never needs to have more references to a $Q_A(UR)$ than the number of DLSDUs waiting for transmission or retransmission.

c) If the DLCEP is an ORDERED DLCEP or a publisher DLCEP, and $V_C(R)$ exists, and the J subfield of the received SD-parameters is not zero, then the DLE shall compute

$$TEMP = (N_R(NDR) - V_C(R)) \text{ modulo } MOD_R \quad (25)$$

1) If $TEMP > (V_C(M) - V_C(R))$, or
$TEMP = (V_C(M) - V_C(R))$ and RSN > $V_C(MS)$,
then the received sequence number residue $N_R(NDR)$ for an acknowledged or a re-requested DLSDU is obsolete or invalid and shall be ignored.

NOTE Misordering of transmitted DLPDUs, which can result in this condition, is possible when the communications path between the sending and receiving DLEs includes active or backup redundant bridges or DL-paths.

2) Else if 1) does not apply, then the received DLPDU is requesting retransmission of a segment of a previously-transmitted DLSDU (J=1). Let N equal $TEMP + V_C(R)$. If J=1 and N is greater than $V_C(R)$, then the DLE shall add the RSN'th member to the set $V_{C,N}(SS)$; and if the set $V_{C,N}(SS)$ was previously empty, then the DLE shall add to the DLE's unscheduled-service queue, Q(US), a reference to $Q_A(UR)$ of the receiving DLCEP, to ensure that the requested DLPDU is sent from the receiving DLCEP.

NOTE 3 Q(US) never needs to have more references to a $Q_A(UR)$ than the number of DLSDUs waiting for transmission or retransmission.

8.2.2.5.3 Processing of the T and truncated DL-time subfields of the received SD-parameters

If the DLCEP's receive binding is to a buffer, then

a) if the receiving DLCEP has a sender's DL-timeliness class of NONE, then the timeliness-status, $V_B(TS)$ (see 4.7.4.21), associated with writing the buffer shall be set to FALSE;

b) otherwise, when a) does not apply, then

   1) If the received DLPDU conveyed the first-received segment of the DLSDU, then

      i) the buffer's associated timeliness-status, $V_B(TS)$ (see 4.7.4.21), shall be set equal to the T subfield of the received DLPDU;

      ii) if that timeliness-status is TRUE, and if the SD-parameter included time-of-production, format K to M, then the time-of-production of the buffer, $V_B(TP)$ (see 4.7.4.20), shall be inferred as the most recent DL-time whose residue under the negotiated DLPDU format would give rise to the DL-time residue conveyed by the received DLPDU;

      NOTE This inference will cause the current DL-time to be inferred when the SD-parameters field of the received DLPDU does not convey octets of DL-time.

      iii) the DL-time of reception of the DLPDU shall be used as the time of writing the buffer, $V_B(TW)$ (see 4.7.4.19).

2) if 1) does not apply, so that the received DLPDU conveyed a not-previously-received segment of a multi-segment DLSDU for which at least one segment had been previously received, then

   i) the value of the T subfield from the newly received DLPDU shall be ANDed into the buffer's associated timeliness-status, $V_B(TS)$ (see 4.7.4.21);

   ii) any DL-time conveyed in the SD-parameter of the DLPDU shall be ignored;

   iii) the DL-time of reception of the DLPDU shall be used as the time of writing the buffer, $V_B(TW)$ (see 4.7.4.19).

8.2.2.6 Validation and processing of user data received in a CA, DT or ED DLPDU

In the following, MUD is defined to be the maximum number of user data octets which may be conveyed in a single CA, DT or ED DLPDU of the DLCEP's priority, as specified in 4.2.3.

If a received CA, DT or ED DLPDU has a non-null user data field following its SD-parameters field, then the contained data is the ASN'th segment, of TNS segments, of the *K*'th DLSDU, where ASN and TNS are both zero-origin, and *K* has the value last given it in 8.2.2.5.1a)i) or 8.2.2.5.1b)2)ii)A) or 8.2.2.5.1b)2)iii) or 8.2.2.5.1b)4).

The receiving DLE shall check whether (ASN × MUD plus the length of the received user data) is less than or equal to the permitted maximum DLSDU size, $P_C(NP.M...M_R)$, negotiated for this (receiving) direction of DLC transmission. If ASN is less than TNS, then the DLE shall also check whether the length of the received user data is equal to MUD. If either of these requirements is violated, then the DLE shall disconnect the DLCEP as specified in 8.2.1.8, with a reason of "provider-originated disconnection – wrong DLSDU size, permanent condition":

If member number ASN is a member of the set variable Vc,k(MRS), then

a) the receiving DLE shall determine whether a received-data record has already been allocated to the *K*'th DLSDU, and if not, shall allocate a record which may contain at least min ($P_C(NP.M...M_R)$, TNS × MUD) octets of DLS-user data and shall associate that record with the *K*'th received DLSDU;

b) the receiving DLE shall copy the user data field from the received DLPDU to that received-data record, starting with octet (ASN × MUD +1) of the received-data record;

c) if the set variable $V_{C,K}(MRS)$ is now empty, then the receiving DLE shall attempt to deliver the DLSDU as specified in 8.2.2.7.

NOTE It is possible for an implementation of this DL-protocol to perform DLSDU reassembly at the time of DLPDU receipt. Such an implementation approach could minimize the DLE's processing of DLS-user data.

8.2.2.7 Delivery of an entire DLSDU which has been completely received at a DLCEP

Let *K* be the sequence number associated with the completely received DLSDU, as determined by 8.2.2.5.1 at the time that DLSDU reception was completed.

If *K* is greater than $V_C(L)$, and the current state $V_C(ST)$ of the DLCEP is DATA TRANSFER READY, then the DLCEP's receiving data-delivery features and the type of receive buffer or queue binding determine the DLSDU delivery policy.

a) If the DLC is a CLASSICAL DLC, and *K* is greater than $V_C(L)$ + 1, then the DLSDU has been received out of order, and the DLE shall retain but not deliver the DLSDU at this time.

b) Otherwise,

   1) if a receiving buffer is bound to the DLCEP, then the DLE shall do as specified in 8.2.2.7.1;

   2) if an explicit receiving queue is bound to the DLCEP, then the DLE shall do as specified in 8.2.2.7.2; or

3) if no receiving queue is bound to the DLCEP, which is the OSI default situation, then the implementation shall consistently either
   i) do as specified in 8.2.2.7.3; or
   ii) treat this as a variant of case b), using a receiving queue which the implementation has assigned for this purpose.

   NOTE This queue may be unique to this DLCEP, or may be shared with other similar-priority DLCEPs at this DLSAP.

8.2.2.7.1 Delivery to a receive buffer

a) If the receiving DLCEP has a receiver timeliness attribute other than NONE, then the DLE shall set the variable V_B(TW) (see 4.7.4.19) associated with writing the buffer to the current DL-time.

   NOTE This is the local time of receipt, and not the remote time-of-production.

b) The DLE shall deliver the complete DLSDU as the new contents of the buffer, and shall associate any timeliness information received in the conveying DLPDU(s).
c) The DLE shall report a DL-BUFFER-RECEIVED indication to the DLS-user. If the receiving DLCEP-class was ORDERED, and the received DLSDU is a duplicate of a prior-received DLSDU, then the duplicated-DLSDU attribute of the DL-BUFFER-RECEIVED indication shall specify TRUE; in all other cases it shall specify FALSE.

   If this DLCEP has been specified as a synchronizing DLCEP during the establishment of one or more other local DLCEPs, then the DLE shall record the DL-time of network access, V_C(TNA), for use in the timeliness computations of those referencing DLCEP(s).

d) Any ongoing accesses to the contents of the buffer which are incomplete at the time of DLSDU delivery shall not be affected by the DLSDU delivery.

   NOTE This constraint ensures that each access to a buffer is logically atomic.

e) The DLE shall set V_C(L) equal to *K*; and shall cancel all timers $T_{C,N}$(RRS) which may exist, where *N* is less than or equal to *K*. All DLE resources devoted to reception and reassembly of DLSDUs with sequence numbers less than or equal to *K* should be released.

8.2.2.7.2 Delivery to a receive queue

The DLE shall attempt to append the complete DLSDU, together with identification of the receiving DLCEP, to the receiving queue.

If unsuccessful, the DLE shall inform local DL-management of this queue-full situation.

NOTE This DL-management notification may take the form of incrementing a counter of discarded DLSDUs.

If successful,

a) the DLE shall report a DL-DATA indication to the DLS-user;
b) the DLE shall cancel the timer $T_{C,K}$(RRS) if it exists;
c) if the DLC is an UNORDERED or ORDERED DLC, then
   1) the DLE shall set V_C(L) equal to *K*;
   2) the DLE shall cancel all timers $T_{C,N}$(RRS) which may exist, where *N* is less than *K*. All DLE resources devoted to reception and reassembly of DLSDUs with sequence numbers less than or equal to *K* should be released.
d) if the DLC is a DISORDERED DLC, and if *K* equals (V_C(L) +1), then
   1) the DLE shall set V_C(L) equal to *K;*
   2) if *K* is less than V_C(H), then the DLE shall increment *K*. If the set variable $V_{C,K}$(MRS) is empty, then the DLE shall set V_C(L) equal to *K* and shall repeat this step;

3) if the DLC is a PEER DLC, then if the DLE's DL-address unscheduled-service queue, Q(US), does not already contain a reference to the DLCEP, then the DLE shall append to that Q(US) a reference to the DLCEP, to ensure that an acknowledgment of DLSDU receipt is sent from the receiving DLCEP;

e) if the DLC is a CLASSICAL DLC, then
   1) the DLE shall set $V_C(L)$ equal to *K;*
   2) if *K* is less than $V_C(H)$, then the DLE shall increment *K*. If the set variable $V_{C,K}(MRS)$ is empty, then the DLE shall repeat the entire data delivery procedure (see beginning of 8.2.2.7.2) using the new value of *K*;
   3) if the DLC is a PEER DLC, then if the DLE's DL-address unscheduled-service queue, Q(US), does not already contain a reference to the DLCEP, then the DLE shall append to that Q(US) a reference to the DLCEP, to ensure that an acknowledgment of DLSDU receipt is sent from the receiving DLCEP.

8.2.2.7.3 OSI-default delivery

The DLE shall report a DL-DATA indication to the DLS-user, conveying the received DLSDU as a parameter; after which the DLE shall do as specified in 8.2.2.7.2b) to 8.2.2.7.2e).

8.2.2.8 Receipt of a DT DLPDU addressed to a DLCEP

In the following, MUD is defined to be the maximum number of user data octets which may be conveyed in a single DT DLPDU of the DLCEP's priority, as specified in 4.2.3.

When the DLE receives a DT DLPDU addressed to a DLCEP of the DLE, the DLE shall perform the following series of actions:

a) If the DLCEP state, $V_C(ST)$, is WAITING-FOR-RESET-completion and the DLE is waiting only for receipt of a DT DLPDU at the DLCEP, then the DLE shall change the DLCEP state to DATA-TRANSFER-READY.
b) The DLE shall validate that
   1) the priority of the received DT DLPDU is as expected;
   2) in a received DT DLPDU addressed to all subscribers of a PUBLISHER DLCEP, the length of the publisher's DL-address is greater than or equal to that expected;
   3) in a received DT DLPDU addressed to a PUBLISHER DLCEP, the number of DL-addresses is as expected;
   4) that in a received DT DLPDU addressed to a PEER DLCEP
      i) the length and number of the DL-address(es) is as expected (only LONG, or only SHORT, or either SHORT or VERY-SHORT at the sender's option);
      ii) when two addresses are expected, that the second DL-address of the DLPDU is the DLCEP-address of the remote peer of the DLCEP addressed by the DT DLPDU's first DL-address.

   If this validation fails, then

   – if the DLCEP is PEER DLCEP, the DLE shall disconnect the DLCEP from the DLC as specified in 8.2.1.8 with a reason of "provider-originated disconnection – wrong DLPDU format or parameters, permanent condition";
   – else the DLE shall discard the DT DLPDU.
c) If the DLCEP is a PEER or SUBSCRIBER DLCEP whose negotiated residual-activity attribute is TRUE, then the DLE shall restart the DLCEP's $T_C(RAM)$ as specified in 8.2.2.14.

   NOTE See the permission at the start of Clause 8 with regard to use of V(NRC) versus V(MRC).
d) If the remaining number of octets in the DLPDU is less than the number of octets in the negotiated SD-parameters format for the applicable sender-to-receiver direction of transmission, then

– if the DLCEP is PEER or SUBSCRIBER DLCEP, then the DLE shall disconnect the DLCEP from the DLC as specified in 8.2.1.8 with a reason of “provider-originated disconnection – wrong DLPDU format or parameters, permanent condition”;

– else the DLE shall discard the DT DLPDU.

Otherwise the DLE shall parse and process the applicable-format SD-parameters from those remaining octets as specified in 8.2.2.5; and if the remaining number of octets in the DLPDU, after the SD-parameters, is greater than zero, then the DLE shall process that user data as specified in 8.2.2.6 and possibly 8.2.2.7.

8.2.2.9 Receipt of a CA, CD or ED DLPDU

When the DLE receives a CA, CD or ED DLPDU addressed to a DLCEP of the DLE, the DLE shall perform the following series of actions:

a) If the DLCEP is a PEER DLCEP then the DLE shall validate that

   1) the length and number of DL-addresses is as expected;

   2) when present and the receiving DLCEP is a PEER DLCEP, that the second DL-address of the DLPDU is the DLCEP-address of the remote peer of the receiving DLCEP;

   3) the DLL priority of the DLCEP is equal to the priority specified in the received DLPDU.

   If this validation fails, then the DLE shall disconnect the DLCEP from the DLC as specified in 8.2.1.8 with a reason of “provider-originated disconnection – wrong DLPDU format or parameters, permanent condition”.

b) If the DLCEP is a PEER DLCEP whose negotiated residual-activity attribute is TRUE, then the DLE shall restart the DLCEP’s $T_C(RAM)$ with a duration based on the user-specified maximum confirm delay for DL-CONNECT, DL-RESET and DL-SUBSCRIBER-QUERY primitives. If the specified value was other than UNLIMITED, then the duration of this timer should be; otherwise the duration should be $\frac{60\text{ s}}{V(NRC)+1}$. DL-management may override these preferred durations.

   NOTE See the permission at the start of Clause 8 with regard to use of V(NRC) versus V(MRC).

c) If the received DLPDU is a CD DLPDU, then

   1) if the DLCEP is a PEER DLCEP, and if the remaining number of octets in the DLPDU is not zero, and is not equal to the number of octets in the negotiated SD-parameters format for the applicable sender-to-receiver direction of transmission, then the DLE shall disconnect the DLCEP from the DLC as specified in 8.2.1.8 with a reason of “provider-originated disconnection – wrong DLPDU format or parameters, permanent condition”;

   2) otherwise, when 1) does not apply, then

      i) if the remaining number of octets in the DLPDU is equal to the number of octets in the negotiated SD-parameters format for the applicable sender-to-receiver direction of transmission, then the DLE shall parse and process the applicable-format SD-parameters from those remaining octets as specified in 8.2.2.5;

      ii) when the receiving DLCEP’s features are ORDERED, and the DLCEP is bound to a sending buffer, and the DLSDU in the buffer was too large to transmit completely in the immediate response to the received CD DLPDU, then the DLE shall

         A) for the appropriate *K*, modify the variable $V_{C,K}(SS)$, based on the length of the *K*’th DLSDU, to indicate that all segments of the DLSDU except segment zero (0) require transmission;

         B) append to the DLE’s unscheduled-service queue, Q(US), a reference to the DLCEP’s $Q_A(UR)$. The reference signals the need to transmit the contents of the sending buffer which is bound to the DLCEP.

d) If the received DLPDU is a CA or ED DLPDU, then

1) if the remaining number of octets in the DLPDU is equal to or greater than the number of octets in the negotiated SD-parameters format for the applicable sender-to-receiver direction of transmission; then the DLE shall
   i) parse and process the applicable-format SD-parameters from those remaining octets as specified in 8.2.2.5;
   ii) then process any remaining octets of the DLSDU as received DLS-user data as specified in 8.2.2.6 and 8.2.2.7;
2) if 1) does not apply and the DLCEP is a PEER DLCEP, then the DLE shall disconnect the DLCEP from the DLC as specified in 8.2.1.8 with a reason of "provider-originated disconnection – wrong DLPDU format or parameters, permanent condition".

8.2.2.10 Starting, cancellation and expiration of the timer T_U(MCD) on a DL-DATA request

The timer T_U(MCD) shall be started when the DLS-user issues the corresponding DL-DATA request. It shall be cancelled

– at a subscriber DLCEP, or at a CLASSICAL or DISORDERED peer DLCEP, or at an ORDERED peer or publisher DLCEP when V_C(R) does not exist, when the DLE issues the corresponding DL-DATA confirm;
– at a CLASSICAL or DISORDERED publisher DLCEP, or at an ORDERED peer or publisher DLCEP when V_C(R) exists, when the DLE finds it necessary to increment V_C(R) and the resultant value of V_C(R) equals the sequence number assigned to the corresponding DLSDU.

If the timer T_U(MCD) expires on a DL-DATA request, then the DLE shall

a) increment V_C(R);

   NOTE After incrementation, V_C(R) should have a value equal to the sequence number of the DLSDU associated with the expired request.

b) remove the request from the sending DLCEP-address's user-request queue, Q_A(UR), and terminate processing of the request;
c) maintain any appropriate DL-management statistics;

   NOTE Determination of the minimum necessary statistics is for future study.

d) if a DL-DATA confirm primitive for the request has not yet been issued, then
   1) initiate a DL-DATA confirm reporting "provider-originated failure – request timeout";
   2) if the choice of sending DLCEP data delivery features is DISORDERED or CLASSICAL, then initiate a reset of the DLC as specified in 8.2.2.19.

8.2.2.11 Starting, cancellation and expiration of the timer $T_{C,K}$(SS)

NOTE This timer is used only by PEER DLCEPs whose sending data delivery features are DISORDERED or CLASSICAL.

The timer $T_{C,K}$(SS) shall be started whenever a DLPDU containing all or part of $DLSDU_K$ is transmitted and $V_{C,K}$(SS) is empty; it shall be cancelled whenever V_C(A) is greater than or equal to *K* or whenever $V_{C,K}$(SS) becomes non-empty (see 8.2.2.5.2b)).

The duration of this timer shall be based on the local user-specified maximum confirm delay for DL-DATA primitives. If the specified value was other than UNLIMITED, then the duration of this timer should be between 25 % and 50 % of $\frac{V_c(NP)\times MCD\_D}{V(NRC)+2}$; otherwise the duration should be between 25 % and 50 % of $\frac{60s}{V(NRC)+2}$. DL-management may override these preferred durations.

NOTE See the permission at the start of Clause 8 with regard to use of V(NRC) versus V(MRC).

If the timer $T_{C,K}(SS)$ expires, then the DLE shall

a) modify the variable $V_{C,K}(SS)$, based on the length of the *K*'th DLSDU, to indicate that the last segment of the *K*'th DLSDU needs retransmission;

b) append to the DLE's unscheduled-service queue, Q(US), a reference to the DLCEP's $Q_A(UR)$, to schedule a retransmission of the unacknowledged DLSDU;

c) maintain any appropriate DL-management statistics.

NOTE 3 Determination of the minimum necessary statistics is for future study.

8.2.2.11.1 Use of the simplified timer $T_C(SS)$

When the permission of 4.7.4.9.1 is employed, the following rules apply:

- The timer $T_C(SS)$ shall be started, but not restarted, whenever a DLPDU containing all or part of $DLSDU_K$ is transmitted and $V_{C,K}(SS)$ is empty. The timer shall be restarted whenever it is not running and $V_C(A)$ is less than $V_C(M)$; it shall be cancelled whenever $V_C(A)$ equals $V_C(M)$ or whenever $V_{C,K}(SS)$ becomes non-empty due to receipt of a request for retransmission (see 8.2.2.5.2b)).
- The duration of this timer shall be based on the local user-specified maximum confirm delay for DL-DATA primitives If the specified value was other than UNLIMITED, then the duration of this timer should be between 25 % and 50 % of $\frac{V_c(NP) \times MCD\_D}{V(NRC)+2}$; otherwise the duration should be between 25 % and 50 % of $\frac{60s}{V(NRC)+2}$ DL-management may override these preferred durations.

  NOTE See the permission at the start of Clause 8 with regard to use of V(NRC) versus V(MRC).
- If the timer $T_C(SS)$ expires, then the DLE shall
  a) modify the variable $V_{C,K}(SS)$, based on the length of the unacknowledged $DLSDU_K$ with the lowest sequence number, to indicate that the last segment(s) of that DLSDU need retransmission;
  b) append to the DLE's unscheduled-service queue, Q(US), a reference to the DLCEP's $Q_A(UR)$, to schedule a retransmission of the unacknowledged DLSDU;
  c) maintain any appropriate DL-management statistics.

8.2.2.12 Starting, cancellation and expiration of the timer $T_{C,K}(RRS)$

NOTE This timer is used by DISORDERED or CLASSICAL subscriber DLCEPs, and optionally by ORDERED peer or subscriber DLCEPs.

The timer $T_{C,K}(RRS)$ shall be started whenever it is not running and a DLPDU requesting retransmission of one or more segments of the *K*'th DLSDU is transmitted. It shall be cancelled whenever all segments of the *K*'th DLSDU are received.

The duration of this timer shall be based on the remote user-specified maximum confirm delay for DL-DATA primitives and conveyed in an EC DLPDU previously received from the sending DLCEP. If the specified value was other than UNLIMITED, then the duration of this timer should be between 25 % and 50 % of $\frac{V_c(NP) \times MCD\_D}{V(NRC)+2}$; otherwise the duration should be between 25 % and 50 % of $\frac{60s}{V(NRC)+2}$. DL-management may override these preferred durations.

NOTE See the permission at the start of Clause 8 with regard to use of V(NRC) versus V(MRC).

If the timer $T_{C,K}$(RRS) expires, then the DLE shall

a) set $V_{C,K}$(RRS) equal to the then-current value of $V_{C,K}$(MRS), to indicate that the specified segments of the *K*'th DLSDU need retransmission;

b) check for a reference to the DLCEP on the DLE's unscheduled-service queue, Q(US), and if not found then append a reference to the DLCEP to the DLE's unscheduled-service queue, Q(US), to schedule a retransmission of the missing DLSDU segments;

c) maintain any appropriate DL-management statistics.

NOTE Determination of the minimum necessary statistics is for future study.

8.2.2.13 Starting, cancellation and expiration of the timer T_C(RAS)

NOTE This timer is used only by PUBLISHER or sending PEER DLEs whose sending data delivery features are ORDERED, DISORDERED or CLASSICAL, and is only required when DLC establishment requested residual activity on the DLC in this (sender-to-receiver) direction of data transfer.

When applicable (see 4.7.4.16 for the conditions of the timer's use), the timer T_C(RAS) shall be started

a) at a sending DISORDERED or CLASSICAL PEER DLCEP, whenever it is not running and when V_C(A) equals (V_C(N) – 1);

b) at a PUBLISHER DLCEP, or sending ORDERED PEER DLCEP, whenever it is not running and when V_C(M) equals (V_C(N) – 1) and $V_{C,V_C(M)}$(SS) is empty.

The duration of this timer shall be based on the user-specified maximum confirm delay for DL-CONNECT request or response primitives. If the specified value was other than UNLIMITED, then the duration of this timer should be between 70 % and 95 % of $\frac{V_c(NP) \times MCD\_D}{V(NRC)+2}$; otherwise the duration should be between 70 % and 95 % of $\frac{60s}{V(NRC)+2}$. DL-management may override these preferred durations.

NOTE See the permission at the start of Clause 8 with regard to use of V(NRC) versus V(MRC).

It shall be cancelled whenever V_C(A) is not equal to (V_C(N) – 1). If the timer T_C(RAS) expires, then the DLE shall check for a reference to the DLCEP on the DLE's unscheduled-service queue, Q(US), and if not found then append a reference to the DLCEP to the DLE's unscheduled-service queue, Q(US), to schedule a transmission to the remote DLCEP(s).

NOTE The resulting transmission may consist of a DT DLPDU with a null user data field, or a CA DLPDU.

8.2.2.14 Starting, cancellation and expiration of the timer T_C(RAM)

NOTE This timer is used only by SUBSCRIBER or receiving PEER DLEs whose receiving data delivery features are ORDERED, DISORDERED or CLASSICAL, and is only required when DLC establishment requested residual activity on the DLC in this (receiver-from-sender) direction of data transfer.

When applicable (see 4.7.4.17 for the conditions of the timer's use), the timer T_C(RAM) shall run continuously. It shall be restarted whenever any DLPDU is received on the DLCEP.

The duration of this timer shall be based on the remote user-specified maximum confirm delay for DL-CONNECT request or response primitives and conveyed in an EC DLPDU previously-received from the sending DLCEP. If the specified value was UNLIMITED, then the duration of this timer should be 60 s. Otherwise, the duration should be V_C(NP).MCD_CRS. DL-management may override these preferred durations.

If the timer T_C(RAM) expires, then the DLE shall initiate a reset of the DLCEP as specified in 8.2.2.19.

8.2.2.15 Receipt of a DL-RESET request primitive

When the DLE receives a DL-RESET request from a DLS-user for a DLCEP in the DATA-TRANSFER-READY state, then

a) if the DLCEP class is not PEER, the DLE shall issue a confirm primitive with a status of "failure – reason unspecified" for each outstanding DL-DATA request primitive on the local DLCEP, and shall increment $V_C(R)$ for each such confirmed primitive;
b) the DLE shall
   1) set $V_C(M)$ equal to $V_C(N)$ -1; set $V_C(MS)$ and $V_C(HS)$ equal to zero;
   2) release all DLE resources devoted to reception and reassembly of DLSDUs;
   3) cancel all timers $T_{C,K}(SS)$ and $T_{C,K}(RRS)$.
c) if the DLCEP-class is PEER, then the DLE shall
   1) note that both
      i) a DL-RESET confirm primitive;
      ii) receipt of an RC DLPDU not requesting a reply RC DLPDU from the remote peer DLCEP

      will be required;
   2) encode an RC DLPDU, addressed to the remote peer DLCEP and requesting a reply RC DLPDU, with the reason specified by the DLS-user, using $V_C(L)$ +1 for the NDR sequence number and $V_C(M)$ for the NDS sequence number;
   3) schedule the DLPDU for transmission at the DLC's priority as specified in 8.4.5;
   4) start a timer as specified in 8.2.1.2b);
   5) change the DLCEP state, $V_C(ST)$, to WAITING-FOR-RESET-COMPLETION, while waiting for an RC DLPDU.
d) if the DLCEP-class is PUBLISHER, then the DLE shall
   1) encode an RC DLPDU, addressed to all of the DLC's subscriber's DLCEPs, not requesting a reply RC DLPDU, with the reason specified by the DLS-user, using zero (0) for the NDR sequence number and $V_C(M)$ for the NDS sequence number,
   2) set $V_C(A)$ or $V_C(R)$, as appropriate, equal to $V_C(N)$ –1;
   3) schedule the DLPDU for transmission at the DLC's priority as specified in 8.4.5;
   4) report a DL-RESET confirm to the local DLS-user with a status parameter of "success".
e) If the DLCEP-class is SUBSCRIBER, then the DLE shall report a DL-RESET confirm to the local DLS-user with a status parameter of "success".

8.2.2.16 Receipt of a DL-RESET response primitive

When the DLE receives a DL-RESET response from a DLS-user for an active DLCEP whose state is WAITING-FOR-RESET-COMPLETION then

a) if the DLCEP-class is PEER, and the set of events at the DLCEP for which the DLE is waiting is
   1) only a DL-Reset response, then the DLE shall
      i) encode a DT DLPDU (with or without data) from the DLCEP to notify its peer of the successful receipt of the confirming RC DLPDU;
      ii) schedule the DLPDU for transmission at the DLC's priority as specified in 8.4.5;
      iii) again flush the DLCEP's internal queues;
      iv) change the DLCEP state, $V_C(ST)$, to DATA-TRANSFER-READY;
      v) report a DL-RESET-COMPLETED indication to the local DLS-user;
   2) both
      i) a DL-Reset response;

ii) receipt of an RC DLPDU not requesting a reply from the peer DLCEP,

then the DLE shall wait for that RC DLPDU;

3) both

i) a DL-Reset response;

ii) subsequent receipt of a DT DLPDU from the peer DLCEP,

then the DLE shall

A) encode an RC DLPDU, addressed to the remote peer DLCEP and not requesting a reply RC DLPDU, with a reason of "reset, unknown origin – reason unspecified", using $V_C(L)$ +1 for the NDR sequence number and $V_C(M)$ for the NDS sequence number;

B) schedule the DLPDU for transmission at the DLC's priority as specified in 8.4.5;

C) start a timer as specified in 8.2.1.2b);

D) note that a DL-RESET-COMPLETED indication to the local DLS-user is still required;

E) wait for a DT DLPDU.

b) if the DLCEP-class is PUBLISHER or SUBSCRIBER, then the DLE shall

1) change the DLCEP state, $V_C(ST)$, to DATA-TRANSFER-READY;

2) again flush the DLCEP's internal queues;

3) report a DL-RESET-COMPLETED indication to the local DLS-user.

8.2.2.17 Receipt of an RC DLPDU

When the DLE receives an RC DLPDU, the DLE shall determine the version of the DL-protocol in use, as specified in the received RC DLPDU, and shall interpret the other RC-parameters of the DLPDU accordingly.

a) If the DLCEP's DLCEP-class is PEER or SUBSCRIBER, when the DLE receives an RC DLPDU specifying that the DLCEP should be reset, then the DLE shall take action depending on the DLCEP state, $V_C(ST)$:

1) waiting-for-EC-DLPDU or waiting-for-connect-response or waiting-for-connect-completion:

The DLE shall

i) disconnect the DLCEP as specified in 8.2.1.8 with a reason of "connection rejection – inconsistent DLCEP state, – permanent condition".

2) data-transfer-ready:

The DLE shall

i) report a DL-RESET indication to the local DLS-user with a reason parameter equal to that in the received RC DLPDU;

ii) if the DLCEP class is PEER and $V_C(A)$ is less than the sequence number implied by the NDR subfield of the RC parameters of the received RC DLPDU (which is the smallest sequence number ≥ $V_C(A)$ whose modulus would be NDR), then the DLE shall process outstanding DL-DATA requests on the local DLCEP as specified in 8.2.2.5.2, with NDR as received in the RC DLPDU, J=0 and K=1;

iii) if the DLCEP's DLCEP-class is PEER, the DLE shall issue a confirm primitive with a status of "failure – reset or disconnection" for each outstanding DL-DATA request primitive on the local DLCEP, and shall increment $V_C(R)$ for each such confirmed primitive; and set $V_C(A)$ or $V_C(R)$, as appropriate, equal to $V_C(N)$ –1;

iv) set $V_C(L)$ and $V_C(H)$ equal to the sequence number implied by the NDS sub-field of the RC-parameters of the received RC DLPDU (which is the smallest sequence number ≥ $V_C(H)$ whose modulus would be NDS);

v) set $V_C(M)$ equal to $V_C(N)$ -1; set $V_C(MS)$ and $V_C(HS)$ equal to zero;

vi) release all DLE resources devoted to reception and reassembly of DLSDUs;

vii) cancel all timers $T_{C,K}$(SS) and $T_{C,K}$(RRS);

viii) change the DLCEP state, V_C(ST), to WAITING-FOR-RESET-COMPLETION;

ix) wait for a DL-RESET response plus, if the DLCEP class is PEER, subsequent receipt of a DT DLPDU indicating peer reset completion;

3) waiting-for-reset-completion:

i) If the DLCEP-class is PEER and V_C(A) is less than the sequence number implied by the NDR subfield of the RC-parameters of the received RC DLPDU (which is the smallest sequence number ≥ V_C(A) whose modulus would be NDR), then the DLE shall process outstanding DL-DATA requests on the local DLCEP as specified in 8.2.2.5.2, with NDR as received in the RC DLPDU, J=0 and K=1.

ii) If the DLCEP's DLCEP-class is PEER, the DLE shall issue a confirm primitive with a status of "failure – reset or disconnection" for each outstanding DL-DATA request primitive on the local DLCEP, and shall increment V_C(R) for each such confirmed primitive; and set V_C(A) or V_C(R), as appropriate, equal to V_C(N) –1.

iii) The DLE shall set V_C(L) and V_C(H) equal to the sequence number implied by the NDS subfield of the RC-parameters of the received RC DLPDU (which is the smallest sequence number ≥ V_C(H) whose modulus would be NDS).

iv) If the DLCEP-class is SUBSCRIBER, then the DLE shall discard the received RC DLPDU.

v) If the DLCEP-class is PEER, and the DLE is waiting for a DL-RESET response primitive at the DLCEP, then

– if the received RC DLPDU specified that a reply RC DLPDU is required, then the DLE shall note that an RC DLPDU is required, else the DLE shall note that a DT DLPDU is required.

vi) If the DLCEP-class is PEER, and the DLE is not waiting for a DL-RESET response primitive at the DLCEP, and the received RC DLPDU requested a reply, then the DLE shall

A) encode an RC DLPDU, addressed to the remote peer DLCEP and not requesting a reply RC DLPDU, with a reason of "reset, unknown origin – reason unspecified", using V_C(L) +1 for the NDR sequence number and V_C(M) for the NDS sequence number;

B) schedule the DLPDU for transmission at the DLC's priority as specified in 8.4.5;

C) start a timer as specified in 8.2.1.2b);

D) if the reset-completion primitive expected by the DLS-user at the DLCEP is DL-RESET confirm, then the DLE shall note that an RC DLPDU is required, else the DLE shall note that a DT DLPDU is required.

vii) If the DLCEP-class is PEER, and the DLE is not waiting for a DL-RESET response primitive at the DLCEP, and the received RC DLPDU did not request a reply, then the DLE shall

A) encode a DT DLPDU (with or without data) from the DLCEP to notify its peer of the successful receipt of the confirming RC DLPDU;

B) schedule the DLPDU for transmission at the DLC's priority as specified in 8.4.5;

C) again flush the DLCEP's internal queues;

D) change the DLCEP state, V_C(ST), to DATA-TRANSFER-READY;

E) if the reset-completion primitive expected by the DLS-user at the DLCEP is DL-RESET confirm, then the DLE shall report a DL-RESET confirm to the local DLS-user, else the DLE shall report a DL-RESET-COMPLETED indication to the local DLS-user.

viii) When none of ii) to v) applies, then

A) the DLE shall encode a DC DLPDU as specified in 6.2 and 7.2, with a reason of "provider-originated disconnection or connection rejection – reason unspecified", and schedule the DC DLPDU for transmission at TIME-AVAILABLE priority as specified in 8.4.5;

B) the DC DLPDU shall have both destination and source addresses (see 6.2.1 formats 1L and 1S), the destination address shall be identical to the first source DLCEP-address of the received RC DLPDU, and the source address shall be identical to the destination DL-address of that received RC DLPDU.

4) In all other cases the DLE shall act as in a)3)v).

b) If the DLCEP's DLCEP-class is PUBLISHER, then the DLE shall act as in a)3)v).

8.2.2.18 Expiration of the timer T_U(MCD) on a DL-RESET request or indication

If the timer T_U(MCD) expires on a DL-RESET request or DLE-initiated reset, then

a) if this is the (V(NRC)+1)'th consecutive expiration without returning to the DATA-TRANSFER-READY state, then the DLE shall terminate DLCEP-reset processing and shall disconnect the DLCEP from the DLC as specified in 8.2.1.8 with a reason of "provider-originated disconnection – timeout, permanent condition";

NOTE See the permission at the start of Clause 8 with regard to use of V(NRC) versus V(MRC).

b) otherwise, if a) does not apply, then the DLE shall

1) restart the timer with the same period as the previous time;

2) reschedule the RC DLPDU (which was first sent just after exiting the DATA-TRANSFER-READY state) for retransmission at the DLC's priority as specified in 8.4.5.

8.2.2.19 DLE-initiated reset

When the DLE initiates a reset, then

a) the DLE shall change the DLCEP state, V_C(ST), to WAITING-FOR-RESET-COMPLETION;

b) the DLE shall note the need for

1) receipt of a DL-RESET response from the DLS-user;

2) if the DLCEP class is PEER, subsequent receipt of an RC DLPDU indicating peer reset completion;

c) if the DLCEP's DLCEP-class is PEER, then the DLE shall start a user-request timer T_U(MCD) with a duration based on the user-specified maximum confirm delay for DL-CONNECT, DL-RESET and DL-SUBSCRIBER-QUERY primitives. If the specified value was other than UNLIMITED, then the duration of this timer should be $\frac{V_c(NP) \times MCD\_CRS}{V(NRC)+1}$; otherwise the duration should be $\frac{60s}{V(NRC)+1}$. DL-management may override these preferred durations;

NOTE See the permission at the start of Clause 8 with regard to use of V(NRC) versus V(MRC).

d) if the DLCEP's DLCEP-class is not PEER, the DLE shall issue a confirm primitive with a status of "failure – reset or disconnection" for each outstanding DL-DATA request primitive on the local DLCEP, shall increment V_C(R) for each such confirmed primitive, and shall set V_C(A) or V_C(R), as appropriate, equal to V_C(N) –1;

e) the DLE shall

1) set V_C(M) equal to V_C(N) -1; set V_C(MS) and V_C(HS) equal to zero;

2) release all DLE resources devoted to reception and reassembly of DLSDUs;

3) cancel all timers $T_{C,K}$(SS) and $T_{C,K}$(RRS);

f) the DLE shall issue a DL-RESET indication primitive to the local DLS-user, specifying the reason for the reset;

g) if the DLCEP class is PEER, then the DLE shall

1) encode an RC DLPDU, with the same reason as the reason given to the local DLS-user and requesting a reply RC DLPDU, using $V_C(L)$ +1 for the NDR sequence number and $V_C(M)$ for the NDS sequence number;

2) queue that RC DLPDU for transmission at the DLC's priority as specified in 8.4.5;

h) if the DLCEP-class is PUBLISHER, then the DLE shall

1) encode an RC DLPDU, addressed to all of the DLC's subscriber's DLCEPs, not requesting a reply RC DLPDU, with the same reason as the reason given to the local DLS- user, using zero (0) for the NDR sequence number and $V_C(M)$ for the NDS sequence number, and

2) schedule the DLPDU for transmission at the DLC's priority as specified in 8.4.5.

8.2.3 Operation of the DLC subscriber query service

The primitives of the DLC subscriber query service are DL-SUBSCRIBER-QUERY request and confirm.

8.2.3.1 Receipt of a DL-SUBSCRIBER-QUERY request primitive

If the request is accepted, as indicated by a returned status of "request accepted" for the DL-SUBSCRIBER-QUERY request, then upon completion of the request, either successfully or after failure, the DLE shall issue a DL-SUBSCRIBER-QUERY confirm at the DLCEP, conveying the status of the request to the DLS-user.

The DL-SUBSCRIBER-QUERY request shall be queued on the DLE's NODE DL-address user-request queue, $Q_N(UR)$, which is an implicit (DLE-controlled) queue.

a) If that queue is full, then the DLE shall immediately return the corresponding DL-LISTENER-QUERY confirm indicating the reason for failure – "provider-originated failure – queue full".

b) If a) does not apply, then the DLE shall start a user-request timer $T_U(MCD)$ with a duration based on the user-specified maximum confirm delay for DL-CONNECT, DL-RESET and DL-SUBSCRIBER-QUERY primitives. If the specified value was other than UNLIMITED, then the duration of this timer should be $\dfrac{V_c(NP) \times MCD\_CRS}{V(NRC)+1}$; otherwise, the duration should be $\dfrac{60s}{V(NRC)+1}$. DL-management may override these preferred durations.

NOTE See the permission at the start of Clause 8 with regard to use of V(NRC) versus V(MRC).

c) If a) does not apply, then the DLE shall

– append the request to the DLE's NODE user-request queue, $Q_N(UR)$, at TIME-AVAILABLE priority, where the request shall be placed in the second partition;

NOTE The third partition in the DLE's $Q_N(UR)$ is always empty.

– append a reference to $Q_N(UR)$ to the DLE's unscheduled-service queue, Q(US).

8.2.3.2 Transmission of a DL-address query SPDU

Upon receipt of a transmission opportunity for the queued DL-SUBSCRIBER-QUERY request, the DLE shall

a) form a DL-address query SPDU (see 9.3.6.2) specifying the requested DL-address;

b) include that SPDU as the DLSDU of a connectionless format-1S DT DLPDU of TIME-AVAILABLE priority with a SHORT destination DL-address for the DL-support functions of all

DLE's on the local link, 0100_{16} (see 4.3.3.2), and source DL-address equal to the NODE.0 DL-address, V(TN).0, of the sending DLE's DL-support functions.

8.2.3.3 Receipt of a DL-address query SPDU specifying a subscriber DLCEP-address

If a DLE receives a DL-address query SPDU/DLPDU for a DL-address and DL-address-class which is active within the DLE, then the DLE shall respond at a TIME-AVAILABLE priority with a corresponding DL-address reply SPDU/DLPDU, confirming the received DL-address query.

8.2.3.4 Receipt of a DL-address reply SPDU

If the DLE receives a DL-address reply SPDU/DLPDU in response to the DL-address query SPDU/DLPDU which was transmitted as a result of the DL-SUBSCRIBER-QUERY request, then the DLE shall

a) cancel the timer T_U(MCD);

b) initiate a DL-SUBSCRIBER-QUERY confirm reporting "success – a subscriber exists";

c) release the request identifier, making it available for subsequent re-use.

8.2.3.5 Expiration of the timer T_U(MCD) on a DL-SUBSCRIBER-QUERY request

If the timer T_U(MCD) expires on a DL-SUBSCRIBER-QUERY request,

a) if this is the (V(NRC)+1)'th consecutive expiration, then the DLE shall

   NOTE See the permission at the start of Clause 8 with regard to use of V(NRC) versus V(MRC).

   1) terminate processing of the request;
   2) initiate a DL-SUBSCRIBER-QUERY confirm reporting "provider-originated failure – request timeout";

b) otherwise, if a) does not apply, then the DLE shall

   1) restart the timer with the same period as the previous time;
   2) requeue the SPDU/DLPDU for retransmission as in 8.2.3.2b).

8.3 Operation of the connectionless-mode services

The connectionless-mode services are connectionless data transfer with local-DLE-confirmation, connectionless data transfer with remote-DLE-confirmation, connectionless data exchange, and listener query.

8.3.1 Operation of the connectionless data transfer with local-DLE-confirmation service

The primitives of the connectionless data transfer with local-DLE-confirmation service are DL-UNITDATA request, indication and confirm.

8.3.1.1 Receipt of a DL-UNITDATA request primitive not specifying remote-DLE-confirmation

When the DLE receives a DL-UNITDATA request primitive not specifying remote-DLE-confirmation, it shall associate the DLS-user-specified request identifier with the request. If the request is rejected, then the DLE shall issue a DL-UNITDATA confirm with the same request identifier, conveying the status of the request to the DLS-user. If the request is accepted, then upon completion of the requested transmission, either successfully or after failure, the DLE shall issue a DL-UNITDATA confirm with the same request identifier, conveying the status of the request to the DLS-user.

a) If the DL(SAP)-role of that source DL(SAP)-address is BASIC, then the calling-DL(SAP)-address specified in the DL-UNITDATA request is bound as a source at the specified priority to either an explicit (user-controlled) or implicit (DLE-controlled) queue.

b) If the DL(SAP)-role of that source DL(SAP)-address is not BASIC, or if that queue is full, or if the specified DLSDU has an invalid length, then the DLE shall immediately return the reason for failure (in a DL-UNITDATA confirm primitive) as the status of the DL-UNITDATA request primitive.

c) If the called-DL(SAP)-address specified in the DL-UNITDATA request is not a DL(SAP)-address whose DL(SAP)-role is BASIC or GROUP, then the request is erroneous. The DLE is permitted, but is not required, to detect such an error (for example, by detecting that the called address is not a DL(SAP)-address). If such an error is detected, then the DLE shall immediately return the reason for failure (in a DL-UNITDATA confirm primitive) as the status of the DL-UNITDATA request primitive.

d) Otherwise, if neither b) nor c) applies, then

   1) the DLE shall create and start a user-request timer T_U(MCD) with a duration based on the user-specified maximum confirm delay for the DL-UNITDATA request primitive. If the specified value was other than UNLIMITED, then the duration of this timer should be equal to that user-specified maximum confirm delay, P_U(MCD); otherwise the duration should be 60 s. DL-management may override these preferred durations.

   2) the DLE shall append the request to the calling DLSAP-address's user-request queue, Q_A(UR), as follows:

      i) If the DL-scheduling-policy attribute of the calling DLSAP-address is EXPLICIT, then the request shall await a DL-COMPEL-SERVICE request to release it for transmission and so shall be placed in the third partition of the user-request queue, Q_A(UR).

      ii) Otherwise, if i) does not apply, then the request shall be placed in the second partition and the DLE shall append to the DLE's unscheduled-service queue, Q(US), a reference to Q_A(UR) of the same priority as the just-appended request.

8.3.1.2 Transmission of a unitdata DT DLSDU

Upon receipt of a transmission opportunity for the queued DLSDU, the DLE shall

a) remove the request from the appropriate DLSAP-address user-request queue, Q_A(UR);

b) form and send a DT DLPDU of the specified priority, with the specified called and calling DL(SAP)-addresses, with a null SD-parameters field (format P (see 7.4.1a))), and with a user data field whose length and contents equal the specified DLSDU;

   NOTE With the exception of the final-token-use subfield of the DT DLPDU's first octet, the DT DLPDU may be formed when the request is queued, and need not be formed dynamically at the moment of transmission.

c) issue a DL-UNITDATA confirm with the associated request identifier reporting "success";

d) cancel and delete the associated timer T_U(MCD).

8.3.1.3 Receipt of a DT DLPDU, with an explicit source address, addressed to a DL(SAP)-address

When the DLE receives a DT DLPDU with an explicit source address addressed to a DL(SAP)-address bound to one or more of the DLE's DLSAPs, then if the DL(SAP)-role for that destination DL(SAP)-address is

a) INITIATOR or CONSTRAINED RESPONDER or UNCONSTRAINED RESPONDER, then the received DLPDU is erroneous and the DLE

   1) shall inform local DL-management of the event, possibly including the erroneous DLPDU's addresses;

   2) shall discard the DLPDU.

b) BASIC or GROUP, then for each of those bindings

   1) the DLE shall attempt to append the received user data as a DLSDU, together with the called and calling DL(SAP)-addresses and DLL priority of the received DLPDU, to

      – the receiving queue which was bound at the received priority level, or

– to the implicit OSI queue if no receiving buffer or queue was bound at that received priority level;

2) if 1) is successful, then the DLE shall initiate a DL-UNITDATA indication primitive at that DLSAP;

3) otherwise, if 1) is not successful, then the DLE shall inform local DL-management of the queue-full situation.

NOTE This DL-management notification may take the form of incrementing a counter of discarded DLSDUs.

8.3.1.4 Expiration of the timer T_U(MCD) on a DL-UNITDATA request not specifying remote-DLE-confirmation

If the timer T_U(MCD) expires on a DL-UNITDATA request not specifying remote-DLE-confirmation, then the DLE shall

a) remove the request from the appropriate DLSAP-address user-request queue, Q_A(UR), and the reference to the request from the DLE's unscheduled-service queue, Q(US);

b) initiate a DL-UNITDATA confirm with the associated request identifier reporting "failure – timeout before transmission".

8.3.2 Operation of the connectionless data transfer service with remote-DLE-confirmation

The primitives of the connectionless data transfer with remote-DLE-confirmation service are DL-UNITDATA request, indication and confirm.

8.3.2.1 Receipt of a DL-UNITDATA request primitive specifying remote-DLE-confirmation

When the DLE receives a DL-UNITDATA request primitive specifying remote-DLE-confirmation, it shall associate the DLS-user-specified request identifier with the request. Upon completion of the request, either successfully or after failure, the DLE shall issue a DL-UNITDATA confirm with the same request identifier, conveying the status of the request to the DLS-user.

The calling DL(SAP)-address specified in the DL-UNITDATA request is bound as a source at the specified priority to either an explicit (user-controlled) or implicit (DLE-controlled) queue. If the DL(SAP)-role of that source DL(SAP)-address is not BASIC, or if that queue is full, or if the specified DLSDU has an invalid length, then the DLE shall immediately return the reason for failure (in a DL-UNITDATA confirm primitive) as the status of the DL-UNITDATA request primitive.

If the called-DL(SAP)-address specified in the DL-UNITDATA request is not a DLSAP-address whose DL(SAP)-role is BASIC, then the request is erroneous. The DLE is permitted, but not required, to detect such an error (for example, by detecting that the called address is not a DLSAP-address). If such an error is detected during request processing, then the DLE shall return an appropriate error status in a DL-UNITDATA confirm primitive, indicating the reason for failure, and shall terminate processing of the request.

Otherwise

a) The DLE shall create and start a user-request timer T_U(MCD) with a duration based on the user-specified maximum confirm delay for the DL-UNITDATA request primitive. If the specified value was other than UNLIMITED, then the duration of this timer should be equal to that user-specified maximum confirm delay, P_U(MCD); otherwise the duration should be 60 s. DL-management may override these preferred durations.

b) The DLE shall append the request to the calling DLSAP-address's user-request queue, Q_A(UR), as follows:

1) If the DL-scheduling-policy attribute of the calling DLSAP-address is EXPLICIT, then the request shall await a DL-COMPEL-SERVICE request to release it for transmission and so shall be placed in the third partition of the user-request queue, Q_A(UR).
2) Otherwise the request shall be placed in the second partition of the user-request queue, Q_A(UR), and the DLE shall append to the DLE's unscheduled-service queue, Q(US), a reference to Q_A(UR) of the same priority as the just-appended request.

8.3.2.2 Transmission of a unitdata CA DLPDU

Upon receipt of a transmission opportunity for the queued DLSDU, the DLE shall form and send a CA DLPDU of the specified priority,

a) with the specified called and calling DLSAP-addresses;
b) with an SD-parameters field of format R (see 7.4.1b));
c) with a user data field whose length and contents equal the specified DLSDU.

NOTE With the exception of the final-token-use subfield of the CA DLPDU's first octet, and possibly of the initiator's transaction index subfield, N(LTI), of the CA DLPDU's SD-parameters (see 7.4.1b)3)), the CA DLPDU may be formed when the request is queued, and need not be formed dynamically at the moment of transmission.

8.3.2.3 Receipt of a CA DLPDU, with an explicit source address, addressed to a DL(SAP)-address

When the DLE receives a CA DLPDU with an explicit source address addressed to a DL(SAP)-address bound to one or more of the DLE's DLSAPs, then if the DL(SAP)-role for that destination DL(SAP)-address is

a) GROUP, then the received DLPDU is erroneous, and
   1) the DLE shall inform local DL-management of the event, possibly including the erroneous DLPDU's addresses;
   2) the DLE shall discard the received CA DLPDU.
b) INITIATOR or CONSTRAINED RESPONDER or UNCONSTRAINED RESPONDER, then the received DLPDU is erroneous and
   1) the DLE shall inform local DL-management of the event, possibly including the erroneous DLPDU's addresses;
   2) the DLE shall form and send, as an immediate reply, a reply DT DLPDU, as specified in 6.4.4.1 and 6.7, whose reported status indicates "failure – responder DL(SAP)-role incompatible with this DLPDU";
   3) the DLE shall discard the received CA DLPDU.
c) BASIC, then
   1) the DLE shall attempt to append the received user data as a DLSDU, together with the called and calling DLSAP-addresses and DLL priority of the received DLPDU, to the receiving queue which was explicitly or implicitly bound at the received priority level;
   2) if 1) is successful, then the DLE shall initiate a DL-UNITDATA indication primitive at that DLSAP and shall indicate its success in 4);
   3) otherwise, if 1) is not successful, then the DLE shall indicate the reason for its failure in 4), and shall notify local DL-management of that failure;

      NOTE This DL-management notification may take the form of incrementing a counter of discarded DLSDUs.
   4) the DLE shall form and send, as an immediate reply, a reply DT DLPDU, as specified in 6.4.4.1 and 6.7, whose reported status indicates the success 2) or failure 3).

8.3.2.4 Receipt of a DT DLPDU, with an implicit source address, addressed to a DL(SAP)-address

When the DLE receives a DT DLPDU, with an implicit source address, addressed to a DL(SAP)-address bound to one or more of the DLE's DLSAPs, then if the DL(SAP)-role for that destination DL(SAP)-address is

a) CONSTRAINED RESPONDER or UNCONSTRAINED RESPONDER or GROUP, then the received DLPDU is erroneous;
   1) the DLE shall inform local DL-management of the event, possibly including the erroneous DLPDU's addresses;
   2) the DLE shall discard the received DT DLPDU.
b) INITIATOR, then the DLE shall act as specified in 8.3.3.3.
c) BASIC, then
   1) If the transaction index subfield, N(LTI), of the SD-parameters of the received DLPDU specifies a transaction index for an outstanding incomplete request, then the DLE
      i) shall release that transaction index, making it available for subsequent reuse;
      ii) shall remove the request from the appropriate DLSAP-address user-request queue, Q_A(UR);
      iii) shall cancel and delete the timer T_U(MCD) associated with the request;
      iv) shall issue a DL-UNITDATA confirm to the requesting DLS-user, indicating the request's DLS-user-identifier and reporting the status conveyed by the received DT DLPDU.
   2) Otherwise, if 1) does not apply, then the DLE shall notify local DL-management of the receipt of an inappropriate DLPDU.

NOTE This DL-management notification may take the form of incrementing a counter of discarded DLSDUs.

8.3.2.5 Receipt of an SR DLPDU

When the DLE receives an SR DLPDU,

– which reports a status other than "BOK – bridge OK" (that is, has a value other than E_{16});
– which is received as a reply to an immediately prior CA, CD or ED DLPDU which was originated by the receiving DLE and which was addressed to a DLSAP-address;
– where the DL(SAP)-role for the DLSAP-address from which the transaction initiated (that is, the source DLSAP-address of the initiating CA, CD or ED DLPDU) is BASIC,

then the receiving DLE shall

a) release the transaction index, N(LTI), used in the immediately prior CA, CD or ED DLPDU, making it available for subsequent reuse;
b) remove the request from the appropriate DLSAP-address user-request queue, Q_A(UR);
c) cancel and delete the timer T_U(MCD) associated with the request;
d) issue a DL-UNITDATA confirm to the requesting DLS-user, indicating the request's DLS-user-identifier and reporting the status conveyed by the received SR DLPDU.

8.3.2.6 Expiration of the timer T_U(MCD) on a DL-UNITDATA request specifying remote-DLE-confirmation

If the timer T_U(MCD) expires on a DL-UNITDATA request specifying remote-DLE-confirmation, then the DLE

a) shall release the corresponding transaction index and remove it from active use, making it available for subsequent reuse only after a period of twice the maximum network DLPDU lifetime, V(NDL), of the extended link;

b) shall remove the request from the appropriate DLSAP-address user-request queue, Q_A(UR);

c) shall delete the timer T_U(MCD);

d) shall initiate a DL-UNITDATA confirm to the requesting DLS-user, indicating the request's DLS-user-identifier and reporting a status of "failure – timeout before transmission".

8.3.3 Operation of the connectionless data exchange service

8.3.3.1 Transmission of a unitdata CD or ED DLSDU

Upon receipt of a transmission opportunity for a reference to a DL-address user-request queue, Q_A(UR), for a DLSAP-address whose DL(SAP)-role is INITIATOR, the DLE shall examine the set of sending buffer bindings for that DLSAP-address in priority order as follows:

a) If

- there is a sending binding at URGENT priority to a buffer, and
- the buffer is non-empty,

then the DLS-user data contained in the URGENT buffer becomes the DLSDU selected for transmission, and the priority of that DLSDU is URGENT; or

b) If a) does not apply, and

- the transaction-priority specified in the invoking reference is normal or time-available;
- there is a sending binding at normal priority to a buffer;
- the buffer is non-empty,

then the DLS-user data contained in the NORMAL buffer becomes the DLSDU selected for transmission, and the priority of that DLSDU is NORMAL; or

c) If neither a) nor b) applies, and

- the transaction-priority specified in the invoking reference is TIME-AVAILABLE;
- there is a sending binding at TIME-AVAILABLE priority to a buffer;
- the buffer is non-empty,

then the DLS-user data contained in the TIME-AVAILABLE buffer becomes the DLSDU selected for transmission, and the priority of that DLSDU is TIME-AVAILABLE; or

d) If none of a) to c) applies, then no DLSDU is selected for transmission, then the DLE shall form and send an appropriate DLPDU:

e) If any of a) to c) applies, then a DLSDU was selected by this procedure, and

1) the DLE shall form and send an ED DLPDU as specified in 6.6.3
   i) with the destination address equal to the remote DLSAP-address specified in the invoking reference;
   ii) with the source address equal to the DLSAP-address associated with the user-request queue, Q_A(UR), specified by that reference;
   iii) with a priority equal to the transaction-priority specified in the invoking reference;
   iv) with an SD-parameters field of format R (see 7.4.1b)), specifying the actual priority of the selected DLSDU;
   v) with a user data field whose length and contents equal the selected DLSDU;
2) the DLE shall await an immediate response. If
   i) the DLSDU which was sent to the responder in the ED DLPDU was obtained from a non-retentive buffer (BUFFER-NR);
   ii) an appropriate DT or SR DLPDU is received as an immediate reply, as specified in 6.5.3, 6.5.4, 6.6.3, 6.6.4, 6.7.3 and 6.8.3;

iii) the explicit or implicit status conveyed by that reply is either “success” or “interim success”,

then the buffer shall be set empty.

f) If d) applies, then no DLSDU was selected by this procedure, and

1) the DLE shall form and send a CD DLPDU as specified in 6.5.3

i) with the destination address equal to the remote DLSAP-address specified in the invoking reference;

ii) with the source address equal to the DLSAP-address associated with the user-request queue, Q_A(UR), specified by that reference;

iii) with a priority equal to the transaction-priority specified in the invoking reference;

iv) with an SD-parameters field of format R (see 7.4.1b)), specifying that there is no accompanying DLSDU;

2) the DLE shall await an immediate response.

8.3.3.2 Receipt of a CD or ED DLPDU, with an explicit source address, addressed to a DL(SAP)-address

When the DLE receives a CD or ED DLPDU with an explicit source address addressed to a DL(SAP)-address bound to one or more of the DLE’s DLSAPs, then if the DL(SAP)-role for that destination DL(SAP)-address is

a) GROUP, then the received DLPDU is erroneous, and

1) the DLE shall inform local DL-management of the event, possibly including the erroneous DLPDU’s addresses;

2) the DLE shall discard the received CD or ED DLPDU.

b) BASIC or INITIATOR, then the received DLPDU is erroneous, and

1) the DLE shall inform local DL-management of the event, possibly including the erroneous DLPDU’s addresses;

2) the DLE shall form and send, as an immediate reply, a reply DT DLPDU, as specified in 6.5.4.1, 6.6.4.1 and 6.7, whose reported status indicates “failure – responder DL(SAP)-role incompatible with this DLPDU”;

3) the DLE shall discard the received CD or ED DLPDU.

c) constrained responder, then

1) if the source address of the received CD or ED DLPDU equals the value of the remote-DLSAP-address parameter which was specified in the most recent DL-BIND request primitive for the responding DLSAP-address, then the DLE shall act as in d);

2) otherwise, when 1) does not apply, the received DLPDU is erroneous and the DLE shall act as in b) except that the status returned in b)2) shall be “failure – response restricted to a different peer DLSAP-address”.

d) unconstrained responder, then

1) if the received DLPDU is an ED DLPDU and thus conveys a non-null DLSDU, then

i) if

– the responding DLSAP-address has an explicit binding as receiver at the priority of the received DLPDU;

– the binding is to a buffer;

– the size of the buffer is at least as great as the size of the DLSDU,

then the DLE shall overwrite the buffer with the just-received DLSDU.

ii) If

– the responding DLSAP-address has an explicit binding as receiver at the priority of the received DLPDU;

– the binding is to a queue;
– the size of each queue record is at least as great as the size of the DLSDU;
– the queue is not full,

then the DLE shall append the just-received DLSDU to the queue.

iii) if either i) or ii) applies, then the DLE shall examine the set of sending buffer bindings for that DLSAP-address in priority order as follows:

A) if
– there is a sending binding at URGENT priority to a buffer;
– the buffer is non-empty,

then the DLS-user data contained in the URGENT buffer becomes the DLSDU selected for transmission, and the priority of that DLSDU is URGENT; or

B) if A) does not apply, and
– the transaction-priority specified in the received DLPDU is NORMAL or TIME-AVAILABLE;
– there is a sending binding at NORMAL priority to a buffer;
– the buffer is non-empty,

then the DLS-user data contained in the NORMAL buffer becomes the DLSDU selected for transmission, and the priority of that DLSDU is NORMAL; or

C) if neither A) nor B) applies, and
– the transaction-priority specified in the received DLPDU is TIME-AVAILABLE;
– there is a sending binding at TIME-AVAILABLE priority to a buffer;
– the buffer is non-empty,

then the DLS-user data contained in the TIME-AVAILABLE buffer becomes the DLSDU selected for transmission, and the priority of that DLSDU is TIME-AVAILABLE; or

D) if none of A) to C) applies, then no DLSDU is selected for transmission.

iv) If either i) or ii) applies, then iii) applies, and

A) the DLE shall form and send as an immediate reply an appropriate DT DLPDU as specified in 6.7;

B) if a DLSDU was sent to the initiator in a replying DT DLPDU, and that DLSDU was obtained from a non-retentive buffer (BUFFER-NR), then the buffer shall be set empty;

C) the DLE shall issue a DL-UNITDATA-EXCHANGE indication to the responding DLS-user, reporting
– the responding DLSAP-address's DLS-user-/DL-identifier;
– the priority of the DLS-user data conveyed by the received ED DLPDU, or that no such DLS-user data was received (in a CD DLPDU);
– the priority of the DLS-user data conveyed in the reply DT DLPDU, or that no such DLS-user data was sent;
– a status of "success".

v) If neither i) nor ii) applies, then the received DLSDU was discarded, and

A) the DLE shall form and send as an immediate reply an appropriate DT DLPDU as specified in 6.7;

B) the DLE shall issue a DL-UNITDATA-EXCHANGE indication to the responding DLS-user, reporting
– the responding DLSAP-address's DLS-user-/DL-identifier;
– the priority of the DLS-user data conveyed by the received ED DLPDU which was discarded;

– a status of “failure”.

2) If the received DLPDU is a CD DLPDU and thus did not convey a DLSDU, then

i) the DLE shall examine the set of sending buffer bindings for that DLSAP-address in priority order as in d)1)iii);

ii) the DLE shall act as in d)1)iv)A) and d)1)iv)B);

iii) if

– a DLSDU was selected in ii), or

– the indicate-null-UNITDATA-EXCHANGE-transactions parameter which was specified in the most recent DL-BIND request primitive for the responding DLSAP-address had the value TRUE,

then the DLE shall act as in d)1)iv)C) except that the reported status shall be “failure – no DLS-user data exchanged”.

8.3.3.3 Receipt of a DT DLPDU, with an implicit source address, addressed to a DL(SAP)-address

When the DLE receives a DT DLPDU, with an implicit source address, addressed to a DL(SAP)-address bound to one or more of the DLE’s DLSAPs, then if the DL(SAP)-role for that destination DL(SAP)-address is

a) BASIC or CONSTRAINED RESPONDER or UNCONSTRAINED RESPONDER or GROUP, then the DLE shall act as specified in 8.3.2.4.

b) INITIATOR, and if the transaction index subfield, N(LTI), of the SD-parameters of the received DLPDU specifies a transaction index for an outstanding incomplete request, then

1) the DLE shall release that transaction index, making it available for subsequent reuse;

2) the DLE shall cancel and delete the timer T_U(MCD) associated with the now-complete unitdata-exchange transaction;

3) if the invoking reference was in the DLE’s unscheduled-service queue, Q(US), or had been dynamically appended to a specified scheduled sequence, then the DLE shall remove that reference from that queue or sequence appendage;

4) if the received DT DLPDU contains DLS-user data and thus conveys a non-null DLSDU, then

i) if

– the initiating DLSAP-address has an explicit binding as receiver at the priority of the received DLPDU;

– the binding is to a buffer;

– the size of the buffer is at least as great as the size of the DLSDU,

the DLE shall overwrite the buffer with the just-received DLSDU;

ii) if

– the initiating DLSAP-address has an explicit binding as receiver at the priority of the received DLPDU;

– the binding is to a queue;

– the size of each queue record is at least as great as the size of the DLSDU;

– the queue is not full,

the DLE shall append the just-received DLSDU to the queue;

iii) if either i) or ii) applies, then

A) if a DLSDU which was sent to the responder in an ED DLPDU, and that DLSDU was obtained from a non-retentive buffer (BUFFER-NR), then the buffer shall be set empty.

B) the DLE shall issue a DL-UNITDATA-EXCHANGE indication to the initiating DLS-user, reporting

– the initiating DLSAP-address's DLS-user-/DL-identifier;

– the priority of the DLS-user data conveyed by the previously sent ED DLPDU, or that no such DLS-user data was sent (in a CD DLPDU);

– the priority of the DLS-user data conveyed by the received DLPDU;

– a status of "success";

iv) if neither i) nor ii) applies, then the DLE shall issue a DL-UNITDATA-EXCHANGE indication to the initiating DLS-user, reporting

– the initiating DLSAP-address's DLS-user-/DL-identifier;

– the priority of the DLS-user data conveyed by the previously-sent ED DLPDU, or that no such DLS-user data was sent (in a CD DLPDU);

– the priority of the DLS-user data conveyed by the received DLPDU;

– a status of "failure – resource limitation in initiator";

5) if 4) does not apply, and the DLPDU sent to the responding DLE was an ED DLPDU, thus conveying a non-null DLSDU, then

i) If the DLSDU which was sent to the responder in the ED DLPDU was obtained from a non-retentive buffer (BUFFER-NR), then the buffer shall be set empty;

ii) the DLE shall issue a DL-UNITDATA-EXCHANGE indication to the initiating DLS-user, reporting

– the initiating DLSAP-address's DLS-user/DL-identifier;

– the priority of the DLS-user data conveyed by the previously-sent ED DLPDU, or that no such DLS-user data was sent (in a CD DLPDU);

– the status conveyed by the received DT DLPDU;

6) if neither 4) nor 5) applies, and the DLPDU sent to the responding DLE was a CD DLPDU, then no DLSDUs were exchanged. In this case, if the indicate-null-UNITDATA-EXCHANGE-transactions parameter which was specified in the most recent DL-BIND request primitive for the initiating DLSAP-address had the value TRUE, then the DLE shall issue a DL-UNITDATA-EXCHANGE indication to the initiating DLS-user, reporting the initiating DLSAP-address's DLS-user-/DL-identifier and a status of "failure – no DLS-user data exchanged";

c) INITIATOR, and if the transaction index subfield, N(LTI), of the SD-parameters of the received DLPDU does not specify a transaction index for an outstanding incomplete request, then the DLE shall notify local DL-management of the receipt of an inappropriate DLPDU.

NOTE This DL-management notification may take the form of incrementing a counter of discarded DLSDUs.

8.3.3.4 Receipt of an SR DLPDU

When the DLE receives an SR DLPDU,

– which reports a status other than "BOK – bridge OK" (that is, has a value other than E_{16});

– which was received as a reply to an immediately prior CA, CD or ED DLPDU which was originated by the receiving DLE and which was addressed to a DLSAP-address;

– where the DL(SAP)-role for the DLSAP-address from which the transaction initiated (that is, the source DLSAP-address of the initiating CA, CD or ED DLPDU) is INITIATOR,

then

a) the receiving DLE shall release the transaction index, N(LTI), used in the immediately prior CA, CD or ED DLPDU, making it available for subsequent reuse;

b) the receiving DLE shall cancel and delete the timer T_U(MCD) associated with the request;

c) if the invoking reference was in the DLE's unscheduled-service queue, Q(US), or had been dynamically appended to a specified scheduled sequence, then the DLE shall remove that reference from that queue or sequence appendage;

d) if the indicate-null-UNITDATA-EXCHANGE-transactions parameter which was specified in the most recent DL-BIND request primitive for the initiating DLSAP-address had the value TRUE, then the DLE shall issue a DL-UNITDATA-EXCHANGE indication to the initiating DLS-user, reporting the initiating DLSAP-address's DLS-user-/DL-identifier and a status of "failure – no DLS-user data exchanged".

8.3.3.5 Expiration of the timer T_U(MCD) on an incomplete instance of the unitdata exchange service

If the timer T_U(MCD) expires on an instance of the unitdata-exchange service, then the DLE

a) shall release the corresponding transaction index and remove it from active use, making it available for subsequent re-use only after a period of twice the maximum DLPDU lifetime, V(MDL), of the extended link;
b) shall delete the timer T_U(MCD);
c) shall issue a DL-UNITDATA-EXCHANGE indication to the initiating DLS-user, specifying the initiating DLSAP-address's DLS-user-/DL-identifier and reporting a status of "provider-originated failure – request timeout".

8.3.4 Operation of the listener query service

The primitives of the listener query service are DL-LISTENER-QUERY request and confirm.

8.3.4.1 Receipt of a DL-LISTENER-QUERY request primitive

When the DLE receives a DL-LISTENER-QUERY request, it shall associate the DLS-user-specified request identifier with the request. If the request is accepted, as indicated by a returned status of "request accepted" for the DL-LISTENER-QUERY request, then upon completion of the request, either successfully or after failure, the DLE shall issue a DL-LISTENER-QUERY confirm with the same request identifier, conveying the status of the request to the DLS-user.

The DL-LISTENER-QUERY request shall be queued on the DLE's NODE DL-address user-request queue, Q_N(UR), which is an implicit (DLE-controlled) queue.

a) If that queue is full, then the DLE shall immediately return the corresponding DL-LISTENER-QUERY confirm indicating the reason for failure – "provider-originated failure – queue full".
b) If a) does not apply, then the DLE shall start a user-request timer T_U(MCD) with a duration based on the user-specified maximum confirm delay for the DL-LISTENER-QUERY request primitive. If the specified value was other than UNLIMITED, then the duration of this timer should be $\frac{R_U(\text{MCD})}{V(\text{NRC})+1}$; otherwise the duration should be $\frac{60\text{s}}{V(\text{NRC})+1}$. DL-management may override these preferred durations.

   NOTE See the permission at the start of Clause 8 with regard to use of V(NRC) versus V(MRC).

c) If a) does not apply, then the DLE shall
   – append the request to the DLE's NODE user-request queue, Q_N(UR), at TIME-AVAILABLE priority, where the request shall be placed in the second partition;

     NOTE The third partition in the DLE's Q_N(UR) is always empty.

   – append a reference to Q_N(UR) to the DLE's unscheduled-service queue, Q(US).

8.3.4.2 Transmission of a DL-address query SPDU

Upon receipt of a transmission opportunity for the queued DL-LISTENER-QUERY request, the DLE shall form and send a DL-address query SPDU as specified in 8.2.3.2.

8.3.4.3 Receipt of a DL-address query SPDU specifying a group DL-address

If a DLE receives a DL-address query SPDU (conveyed by a DT DLPDU) for a DL-address and DL-address -class which is active within the DLE, then the DLE shall

a) form a replying DT DLPDU containing a corresponding DL-address reply SPDU, confirming the received DL-address query;
b) append the DT DLPDU to the DLE's NODE user-request queue, $Q_N(UR)$, at the relevant priority;
c) append a reference to $Q_N(UR)$ to the DLE's unscheduled-service queue, Q(US).

8.3.4.4 Receipt of a DL-address reply SPDU

If the DLE receives a DL-address reply SPDU (conveyed by a DT DLPDU) as an apparent response to the DL-address query SPDU/DLPDU which was transmitted as a result of the DL-LISTENER-QUERY request, then the DLE shall

a) cancel the timer $T_U(MCD)$;
b) initiate a DL-LISTENER-QUERY confirm with the corresponding request identifier reporting "success – a listener exists".

8.3.4.5 Expiration of the timer $T_U(MCD)$ on a DL-LISTENER-QUERY request

If the timer $T_U(MCD)$ expires on a DL-LISTENER-QUERY request, then

a) If this is the (V(NRC)+1)'th consecutive expiration, then the DLE shall

   NOTE See the permission at the start of Clause 8 with regard to use of V(NRC) versus V(MRC).

   1) terminate processing of the request;
   2) initiate a DL-LISTENER-QUERY confirm with the corresponding request identifier reporting "provider-originated failure – request timeout".

b) Otherwise, if a) does not apply, then the DLE shall
   1) restart the timer with the same period as the previous time;
   2) requeue the same DT DLPDU for retransmission as specified in 8.3.4.3b).

8.4 Operation of the scheduling guidance services

The scheduling guidance services are the DL-time service, the compel-service service, the sequence scheduling service, and the sequence subsetting service.

8.4.1 Operation of the DL-time service

The only primitive of the DL-time service is DL-TIME request.

8.4.1.1 Receipt of a DL-TIME request primitive

When the DLE receives a DL-TIME request, it shall respond with a five-component record as shown in Table 78, whose values are

a) the current value of the time source's link-id, V(TSL);
b) the quality of the current DL-time, V(TQ), as defined in 7.6b);
c) the current value of the local DL-time-offset, V(DLTO);
d) the current value of the local-link-scheduling-time-offset, V(LSTO);
e) the current value of the local node-time, C(NT).

The DLS-user may then sum the last three values to determine the node's current DL-time, or use the values separately, as appropriate.

Table 78 – Components of returned DL-time

Octet index	Contents of field
1 2	V(TSL)
3	V(TQ)
4 5 6 7 8 9 10	V(DLTO)
11 12 13 14 15 16 17	V(LSTO)
18 19 20 21 22 23 24	C(NT)

If the DLE's time-synchronization class (see 10.6.3) is NONE and the DLE does not maintain even an estimated C(NT), then it may respond with the values 0…0 for V(TQ), V(DLTO), V(LSTO) and C(NT), and $FFFF_{16}$ for V(TSL); this latter, which is an invalid link-id, indicates to the DLS-user the DLE's incapability.

8.4.1.2 Transmission of a TD DLPDU

After sending a TD DLPDU, the LAS DLE shall restart its local timer T(TDP), to expire at a time no greater than 0,95 × V(TDP) in the future (see 4.7.1.26 and 4.7.5.18).

8.4.1.3 Receipt of a TD DLPDU

If the DLE's time-synchronization class (see 10.6.3) is NONE, then

a) If the DLE which is not the LAS DLE supports the variables, timers and counters defined in 4.7.1.19 to 4.7.1.27, then the DLE shall set V(TSL) equal to N_S(TSL), set V(DLTO) equal to N_S(DLTO), and set V(LSTO) equal to

$$\left(N_S(DLT) + N_S(DLTA) - N_S(DLTO) - N_R(NT) + \frac{V(MD)}{2} \right) \text{modulo } 2^{56} . \qquad (26)$$

b) Otherwise, when a) does not apply, then the DLE shall ignore receipt of the TD DLPDU.

When a DLE other than the LAS DLE, with a time-synchronization class other than NONE, receives a TD DLPDU, then

c) the DLE shall restart its local timer T(TDP), to expire at a time such that if it does not receive a TD DLPDU until after the expiration of this timer, it can still maintain time synchronization within the limits implied by the local link's time-synchronization-class (see 4.7.1.25). The value of this timer shall be larger than the minimum specified for the time-synchronization-class in 10.6.3, but the exact value may be determined locally by the DLE.

d) the DLE shall note that the DLE does not have to send a CT.

NOTE Variance in the expiration periods of the timers T(TDP) among DLEs, and cancellation upon receipt of a TD DLPDU of such requests for transmission of a CT DLPDU, both serve to reduce the likelihood that a single link-wide noise event, inhibiting correct reception of a TD DLPDU, will result in multiple transmissions of CT DLPDUs.

e) the DLE shall update

– its locally-maintained sense of the link-id of that link, within the extended-link, which is the time source, V(TSL);

– its locally-maintained sense of the local LAS node, V(LN);

– its locally-maintained sense of the quality of DL-time, V(TQ);

– its DL-time-offset, V(DLTO);

– its local-link-scheduling-time-offset, V(LSTO);

– its node-timer frequency,

by comparing the local variables V(TQ), V(TSL) and V(LN) with the corresponding fields of the received TD DLPDU – N_S(TQ) and N_S(TSL) and the DLPDU's source NODE DL-address – to detect changes in either the reference source for the time or in the time distribution path.

1) If the TD DLPDU's source NODE DL-address differs from the receiving DLE's V(LN), then

   i) the DLE shall set V(LN) equal to the source NODE DL-address from the received TD DLPDU;

   ii) the DLE shall inform local DL-management that a change of time source on the local link has occurred;

   iii) if the DLE requires a measurement of round-trip-delay between itself and the local LAS, to maintain the required accuracy for its time-synchronization class, then the DLE shall invalidate V(MD), the prior round-trip-delay-measurement with the current LAS.

2) If the DLE requires a measurement of round-trip-delay between itself and the local LAS, and if the current value for that round-trip delay, V(MD), is known to be invalid, then the DLE shall form an RQ DLPDU addressed to V(LN).0, and append the DLPDU as specified in 8.4.5, at NORMAL priority to the DLE's NODE DL-address user-request queue, Q_N(UR).

   NOTE This is not the queue normally used with the procedures of 8.4.5.

3) If the measurement of round-trip-delay to the current LAS, V(MD), is valid, then the DLE shall set its V(TQ) from the node-time-quality value, N_S(TQ), in the received DLPDU, after adjustment of the number-of-intervening-links subfield for the one extra intervening link between the sending and receiving DLEs (by incrementing N_S(TQ).LLL when forming V(TQ) to account for the receiving DLE).

   NOTE Therefore a DLE with its own time source, which is not also serving as the source of DL-time to the local link, must ignore its own time source and synchronize with the other DLEs on the local link.

4) If the time-source-link value, N_S(TSL), in the received DLPDU differs from the receiving DLE's V(TSL), and the round-trip-delay-measurement with the current LAS, V(MD), is valid, then the DLE shall

   i) set V(TSL) equal to N_S(TSL);

   ii) set all other local references to the time-source-link, such as the link-id of the source of time distribution of periodic scheduled activities in 10.3.7), equal to N_S(TSL);

   iii) inform local DL-management that a change of time source on the extended link has occurred.

5) If the DL-time-offset, N_S(DLTO), in the received DLPDU differs from the receiving DLE's V(DLTO), then the DLE shall

   i) set V(DLTO) equal to N_S(DLTO);

ii) adjust all other local references based on DL-time, such as the periodic schedule DL-time base (T0) in 10.3.7), by the amount that V(DLTO) just changed, to reflect the new DL-time base;

iii) inform local DL-management that a change in V(DLTO) has occurred.

6) If the measurement of round-trip-delay to the current LAS, V(MD), is valid, then the DLE shall compute

$$TEMP = (N_S(\text{DLT}) + N_S(\text{DLTA}) - N_S(\text{DLTO}) - V(\text{LSTO}) - N_R(\text{NT}))\ \text{modulo}\ 2^{56} \qquad (27)$$

NOTE The above equation is equivalent in form to the computations in a).

In the following computations, the DLE shall use the value of *K* specified in Table 79.

Table 79 – Time synchronization computation

Time-synchronization class	Value of *K* for use in e)4)
1 µs	4
10 µs	41
100 µs	410
1 ms	4 096
10 ms	40 960
100 ms	409 600
1 s	4 096 000

i) If abs(*TEMP*), the absolute value of *TEMP*, is greater than *K*, and this has not been the case for the prior two received TD DLPDUs, then the DLE shall

A) set V(LSTO) equal to (*TEMP* + V(LSTO)) modulo 2^{56};

B) inform local DL-management that a discontinuous change in V(LSTO) has occurred.

ii) If abs(*TEMP*), the absolute value of *TEMP*, is less than or equal to *K*, or if it was greater than *K* for the prior two received TD DLPDUs, then the DLE shall use the computed value, *TEMP*, together with prior computed values as appropriate, to adjust the frequency of incrementation of the DLE's C(NT)

A) to reduce the long-term precession of V(LSTO), relative to $N_S(\text{DLT}) + N_S(\text{DLTA}) - N_S(\text{DLTO})$, to zero;

B) to keep the long-term counting rate of C(NT) to increment by approximately 8 192 000 ($2^{13} \times 10^3$) counts/s.

This frequency adjustment shall be maintained until a newer adjustment is computed.

NOTE The details of the filtering algorithm which determines this frequency adjustment are intentionally left to the implementor.

8.4.1.3.1 Additional actions required of a bridge

If the TD DLPDU was received at an active port in the forwarding state (see Clause 9) which is the bridge DLE's root port (that is, the active port toward the root of the bridge spanning tree), then the DLE shall forward the DLPDU reception event to the bridge operation level (see 4.1.2). The bridge operation level shall share the received sense of time with

– those other active ports of the bridge which are currently serving as the LAS on their own local links;

– all bridge ports which are not in the forwarding or listening states;

– any local DLS-user.

The following also apply.

a) If the bridge has any port other than its root port which is acting as LAS, then the bridge shall
   1) update the DL-time-offset, V(DLTO), for each of those other ports, such that the sum of C(NT), V(LSTO) and V(DLTO) for that non-root port equals the sum of C(NT), V(LSTO) and V(DLTO) for the root port;

      NOTE The DL-time-offset, V(DLTO), and local-link-scheduling-time-offset, V(LSTO), are port-specific (as are most of the variables of 4.7.5, and thus are maintained independently for each port.

   2) if 1) caused a change in the DL-time-offset, V(DLTO), then adjust the periodic schedule DL-time base, T0, in 10.3.7) by the same amount that V(DLTO) just changed, to reflect the new DL-time base;
   3) if there was a change in the time-source-link, V(TSL), then set the link-id of the source of time distribution of periodic scheduled activities in 10.3.7), equal to the new value of V(TSL).
b) The bridge operation level may schedule the sending of additional TD DLPDUs on those other ports which are acting as LAS.
c) The contents of the time subfields of those later DLPDUs shall be based on the then-current values for the sending port's internally-maintained node-time variables, C(NT), V(LSTO) and V(DLTO), rather than the value of the DL-time parameters from the just-received TD DLPDU.
d) The contents of the time-source and time-quality subfields – V(TSL) and N(TQ) – of those later DLPDUs shall be based on the corresponding fields from the just-received TD DLPDU, except that the sent N(TQ) shall reflect both
   – any impact of the DLE's timeliness class on the resolution of the forwarded time;
   – the fact that there is one additional bridge in the forwarding path.

8.4.1.4 Receipt of an RQ DLPDU

When the DLE receives an RQ DLPDU, it shall form and send as an immediate reply a ROUND-TRIP-DELAY REPLY (RR) DLPDU

a) whose destination DL-address equals the source address of the received RQ DLPDU;
b) whose source address is formed by concatenating the sender's node-id, $V_S(TN)$, and an octet of zero;
c) whose first two parameter subfields equal the parameter subfields, respectively, of the received and appended-to RQ DLPDU.

8.4.1.5 Receipt of an RR DLPDU

When the DLE receives an RR DLPDU, if the DLE maintains C(NT), then it shall, within some reasonable period not exceeding 60 s, compute the round-trip delay, RTD, as a function of the four parameter subfields of the received and appended-to RR DLPDU, as:

$$\text{RTD} = (\text{short-time}_4 - \text{short-time}_3 + \text{short-time}_2 - \text{short-time}_1)\ \text{modulo}\ 2^{24} \tag{28}$$

where the fields are numbered in their order of transmission (or appending) within the DLPDU. If the remote DLE in the measurement was the current LAS DLE, V(LN), then the computed delay shall be saved as the measured-delay

$$\text{V(MD)} = \text{RTD} \tag{29}$$

which thereby becomes valid for use in C(NT) frequency-adjustment computations, as specified in equation (15) (see 8.4.1.3).

The results of several round-trip measurements with the same remote node may be combined to reduce the mean error in V(MD).

8.4.1.6 Expiration of the timer T(TDP)

If the timer T(TDP) expires, then

a) if the DLE is the local link's LAS, then the DLE shall note the requirement to transmit a TD DLPDU at the first opportunity;

b) otherwise the DLE shall note the requirement to transmit a CT DLPDU at the first opportunity.

8.4.2 Operation of the compel-service service

The only primitive of the compel-service service is DL-COMPEL-SERVICE request.

8.4.2.1 Receipt of a DL-COMPEL-SERVICE request primitive

When the DLE receives a DL-COMPEL-SERVICE request, it shall classify the request and take the appropriate corresponding action. If the request is for

a) a local (to the DLE) DLSAP-address whose DL(SAP)-role is BASIC, or a peer or publisher or subscriber DLCEP, for which the DL-scheduling-policy is EXPLICIT, then

   1) if the DL-address is bound to a sending queue, and the number of DLSDUs in the third partition of that DL-address's user-request queue, Q_A(UR), which are waiting for a DL-COMPEL-SERVICE request primitive is non-zero, then the DLE shall

      i) adjust the local DL-address's user-request queue, Q_A(UR), by reducing that number (of DLSDUs in the third partition) by one;

      ii) reassign the highest-priority request, from that third partition to the second partition, unless such movement would cause the sending window size of the addressed DLCEP to be exceeded;

      NOTE 1 The restriction in ii) above applies only to peer and publisher DLCEPs, and never to DLSAP-addresses.

      NOTE 2 Implementations are permitted to defer the actual reassignment until the occurrence of a transmit opportunity for the DLSDU being reassigned. Such deferral may permit a subsequently-requested higher-priority DLSDU to be reassigned in lieu of the DLSDU that would have been reassigned at the moment when the DL-COMPEL-SERVICE request was made. However, in no case may an implementation accumulate a "quota of DL-COMPELled requests" which extends beyond the next time that the third partition of the queue Q_A(UR) becomes empty.

      iii) form a reference to that DL-address's Q_A(UR) at the priority specified in the DL-COMPEL-SERVICE request, where the reference indicates the need to send a DLSDU from the sending queue identified in 1), and append the reference to either

         A) the specified scheduled sequence, if the DL-COMPEL-SERVICE request specified a sequence-identifier, or

         B) the DLE's unscheduled-service queue, Q(US), if no Sequence-identifier was specified in the request;

      iv) return an immediate status of "success".

   Otherwise, when no reassignment occurred, the DLE shall return an immediate status of "user failure – no DLSDU to release".

   2) If the DL-address is bound to a sending buffer, then the DLE shall

      i) modify the variable $V_{C,K}$(SS), for the appropriate K corresponding to the DLSDU currently associated with the buffer, to indicate that all segments of the DLSDU require transmission;

ii) form a reference to the Q_A(UR) of the specified local peer or publisher DLCEP, at the DLCEP's priority, where the reference indicates the need to send a DLSDU from the sending buffer identified in 2), and append the reference to either

A) the specified scheduled sequence, if the DL-COMPEL-SERVICE request specified a sequence-identifier, or

B) the DLE's unscheduled-service queue, Q(US), if no Sequence-identifier was specified in the request;

iv) return an immediate status of "success".

b) the remote peer or publisher DLCEP of a local peer or subscriber DLCEP, respectively, then the DLE shall

1) form a reference to the Q_A(UR) of the specified local peer or subscriber DLCEP, at the DLCEP's priority, where the reference indicates the need to compel the transmission of a DLSDU from the remote correspondent peer or publisher DLCEP identified in b), and append the reference to either

i) the specified scheduled sequence, if the DL-COMPEL-SERVICE request specified a sequence-identifier, or

ii) the DLE's unscheduled-service queue, Q(US), if no sequence-identifier was specified in the request;

NOTE The above reference is distinguishable from references which result in local release of DLSDUs.

2) return an immediate status of "success";

c) a local (to the DLE) DLSAP-address whose DL(SAP)-role is INITIATOR, then the DLE shall

1) form a reference to that DL-address's Q_A(UR) at the priority specified in the DL-COMPEL-SERVICE request, where the reference indicates the need to compel an instance of the unitdata-exchange service with the specified remote DLSAP-address, and append the reference to either

i) the specified scheduled sequence, if the DL-COMPEL-SERVICE request specified a sequence-identifier, or

ii) the DLE's unscheduled-service queue, Q(US), if no sequence-identifier was specified in the request;

2) return an immediate status of "success".

d) Some other DL-address, then the DLE shall return an immediate status of "user failure – invalid DL-address".

8.4.3 Operation of the sequence scheduling service

The primitives of the sequence scheduling service are DL-SCHEDULE-SEQUENCE request and confirm, and DL-CANCEL-SCHEDULE request and confirm.

The ability of a DLE to respond positively to a DL-SCHEDULE-SEQUENCE request is determined by both the DLE's own abilities to service a dynamic (that is, not pre-configured by DL-management) request, and the current LAS DLE's ability to support that scheduling process. The conformance claimed in Clause 9 gives some information on the DLE's capabilities.

8.4.3.1 Receipt of a DL-SCHEDULE-SEQUENCE request primitive

When the DLE receives a DL-SCHEDULE-SEQUENCE request, it shall assign a schedule identifier to the request and return that identifier as part of the DL-SCHEDULE-SEQUENCE request primitive.

Upon completion of the scheduling request, either successfully or after failure, the DLE shall issue a DL-SCHEDULE-SEQUENCE confirm with the same schedule identifier, conveying the status of the scheduling request to the DLS-user. If the scheduling request failed, then the DLE shall release the schedule identifier, making it available for subsequent reuse.

NOTE 1 L-SCHEDULE-SEQUENCE confirm indicates completion of the *scheduling* of the associated sequence, either locally or in conjunction with the local LAS; it does not indicate completion of the *execution* of that sequence. An implementation which defers issuing DL-SCHEDULE-SEQUENCE confirm primitives until after the completion of their scheduled sequences is erroneous.

The sequence itself is defined in terms of primitive sequence components specified in Clause 9 and in IEC 61158-3-1, 10.5.3.2.2.

NOTE 2 The specific representations of these primitive sequence components within an end-system is a local-view issue, outside the scope of open system standardization. However, Clause 9 does include a standardized representation for some of these components when they are communicated between end-systems and an LAS DLE, and implementations may choose to use that same representation locally. Formal DL-programming-interface specifications, which would include the details of such an encoding, could standardize this or a similar local representation.

The DLE shall determine the class of scheduling operation requested, based on the sequence definition and the type of schedule requested.

If the Schedule Type parameter has the value ONE-TIME or PERIODIC, then

a) if the sequence consists of a single element, and the Schedule Type parameter has the value ONE-TIME, and the Desired Starting Time parameter specifies IMMEDIATE, and the sequence-priority parameter specifies a priority less than or equal to the intrinsic priority of the single element, then the DLE shall append a reference to the requested sequence at the Sequence Priority to the DLE's unscheduled-service queue, Q(US);
b) else if a) does not apply and the sequence consists of one or more elements, all of which are references to DLCEPs remote from the DLE, none of which are for DLCs specifying EXTRA DLPDU-authentication, then the DLE shall
   1) form an SPDU equivalent to the sequence, of the same priority as the sequence;
   2) include that SPDU as the DLSDU of a connectionless DT DLPDU of TIME-AVAILABLE priority with a SHORT destination DL-address of the local LAS, 0400_{16} (see 4.3.3.2), and source DL-address equal to the NODE.0 DL-address, V(TN).0, of the sending DLE's DL-support functions;
   3) append the DT DLPDU to the DLE's NODE user-request queue, Q_N(UR), at the relevant priority;
   4) append a reference to Q_N(UR) to the DLE's unscheduled-service queue, Q(US);
c) else if a) and b) do not apply then the DLE shall
   1) use the DLS-user-provided DLSEP-address, or allocate a DLSEP-address if one was not provided;
   2) associate the requested sequence to the DLSEP-address, so that receipt of an ES DLPDU specifying that DLSEP-address will cause execution of the associated sequence;
   3) compute the maximum uninterrupted duration of the request;
   4) form a scheduling-request SPDU with priority equal to the Sequence-Priority of the DLS-user request; where the SPDU requests that the token be delegated to the sequence at the specified DLSEP-address, for the computed duration;
   5) include that SPDU in a DT DLPDU and queue that DLPDU for transmission as specified in b)2) to b)4).

If the Schedule Type parameter has the value REPETITIVE, then

d) the DLE shall append a reference to the requested sequence at the Sequence Priority to the DLE's unscheduled-service queue, Q(US).

It is permitted, but not desirable, for a sequence that meets the requirements of b) to be executed locally as in a), c) or d). It is also permitted, but not desirable, for a sequence that meets the requirements of a) or c) to be sent as an SPDU as in b) to the LAS for remote execution; this permission requires that the SPDU syntax (see Clause 11) be capable of conveying the necessary information.

NOTE Application of a DLS-request confirm delay to a DL-SCHEDULE-SEQUENCE request primitive is for future study.

8.4.3.2 Receipt of a DL-CANCEL-SCHEDULE request primitive

When the DLE receives a DL-CANCEL-SCHEDULE request, it shall determine the class of scheduled sequence being cancelled:

a) If the sequence is locally scheduled (see 8.4.3.1a) or d)), then the DLE shall
   1) remove the reference to the schedule from the unscheduled-service queue, Q(US);
   2) issue a DL-CANCEL-SCHEDULE confirm with the identifier of the cancelled schedule;
   3) release the schedule identifier, making it available for subsequent reuse;
b) If the sequence is centrally scheduled but locally executed (see 8.4.3.1c)), then the DLE shall
   1) remove the reference to the schedule from the DLSEP-address associated with the schedule;
   2) form a cancel-schedule SPDU (see 9.3.4.4);
   3) include that SPDU as the DLSDU of a connectionless DT DLPDU of TIME-AVAILABLE priority with a SHORT destination DL-address of the local LAS, 0400_{16} (see 4.3.3.2), and source DL-address equal to the NODE.0 DL-address, V(TN).0, of the sending DLE's DL-support functions;
   4) append the DT DLPDU to the DLE's NODE user-request queue, Q_N(UR), at the relevant priority;
   5) append a reference to Q_N(UR) to the DLE's unscheduled-service queue, Q(US);
c) If the sequence is centrally scheduled and centrally executed (see 8.4.3.1b)), then the DLE shall form and send a cancel-schedule SPDU as specified in b)2) to b)5).

8.4.3.3 Receipt of a schedule-cancelled SPDU

If the DLE receives a schedule-cancelled SPDU in response to a cancel-schedule SPDU, then the DLE shall

a) issue a DL-CANCEL-SCHEDULE confirm with the identifier of the cancelled schedule;
b) release the schedule identifier and the associated DLSEP-address, if any, making it (them) available for subsequent reuse.

8.4.3.4 LAS-initiated schedule cancellation

If the DLE receives a schedule-cancelled SPDU from the LAS, then the DLE shall determine the class of scheduled sequence being cancelled:

a) If the sequence is centrally scheduled but locally executed (see 8.4.3.1c)), then the DLE shall
   1) remove the reference to the schedule from the DLSEP-address associated with the schedule;
   2) issue a DL-CANCEL-SCHEDULE confirm with the identifier of the cancelled schedule;
   3) release the schedule identifier and the associated DLSEP-address, making them available for subsequent reuse.
b) If the sequence is centrally scheduled and centrally executed (see 8.4.3.1b)), then the DLE shall
   1) issue a DL-CANCEL-SCHEDULE confirm with the identifier of the cancelled schedule;
   2) release the schedule identifier, making it available for subsequent reuse.

8.4.4 Operation of the sequence subsetting service

The primitives of the sequence subsetting service are DL-SUBSET-SEQUENCE request and confirm.

8.4.4.1 Receipt of a DL-SUBSET-SEQUENCE request primitive

When the DLE receives a DL-SUBSET-SEQUENCE request sequence,

a) if the specified sequence is not defined or not subsettable, then the DLE shall return an appropriate error status and reject the request;

b) otherwise, when a) does not apply, then the DLE shall subset the sequence as requested and issue a DL-SUBSET-SEQUENCE confirm primitive with the request identifier provided by the DL-SUBSET-SEQUENCE request.

8.4.5 Implicit scheduling of DLS-user requests

When the DLE receives a DLS-user request or response primitive, and the applicable DL-scheduling-policy (see 4.2.6) is IMPLICIT, then the DLE shall execute that request or response as soon as possible, subject to the implicit, or explicit, priority of the request or response. If the request or response requires transmission of one or more DLPDUs, then the DLE shall

a) select the appropriate queue, Q_A(UR), to which the request should be appended:
   1) for connectionless-mode procedures (see 8.3), the associated sending DLSAP-address;
   2) for connection-oriented procedures (see 8.2),
      i) the associated local DLCEP-address for DT DLPDUs when the sending DLCEP's DLCEP-class is PEER or PUBLISHER, or
      ii) the associated sending (calling or responding) DLSAP-address for EC, DC and RC DLPDUs, or
      iii) either i) or ii) for DT DLPDUs when the sending DLCEP's DLCEP-class is SUBSCRIBER;
   3) for other procedures (see 8.4), the queue designated by the procedure;

b) append the request, at the appropriate priority, to that Q_A(UR), and adjust that queue, if necessary, so as not to increase the total number of queue members which are in the third partition of the queue, if any;

c) append a reference to that Q_A(UR), at the appropriate priority, to the DLE's scheduled-service queue, Q(US).

9 DL-support subprotocol

9.1 General

This subclause

a) defines the encodings for the higher-level (see 4.1.3) support PDUs (SPDUs) required to support the LAS operation and scheduling subprotocol;

b) defines the components of a defined sequence (see IEC 61158-3-1, 10.5.3.2.2);

c) defines the encodings for the higher-level (see 4.1.3) support PDUs (SPDUs) required to support various configuration and capability queries used by
   – the DL-SUBSCRIBER-QUERY request and DL-LISTENER-QUERY request elements of procedure;
   – bridges for forwarding-database maintenance;
   – specialized DLL monitoring, analysis and management tools;

d) specifies elements of procedure for processing the received SPDUs.

9.2 Overview of LAS operation

An LAS is required to perform the following functions:

a) Link maintenance – the LAS
   1) detects the presence of a newly added DLE on the local link, after which it activates that DLE;
   2) detects that a previously participating DLE is no longer active on the local link;
   3) maintains a list of all DLEs operational on the local link;
   4) distributes the list of operational DLEs to all other link-master DLEs on the local link.
b) Time distribution – the LAS acts as the time master for the local link and sends time distribution DLPDUs to all other DLEs on the local link to provide a common sense of DL-time for the local link.
c) Token circulation – the LAS sends a PT DLPDU to each active DLE on the local link. By doing this repeatedly, the LAS DLE provides a delegated token which "circulates" successively, usually in NODE DL-address order, to all active DLEs on the local link.
d) Schedule execution – the LAS sends DLPDUs on the local link as dictated by a schedule which specifies the initial DL-time and repetition period for each of those transmissions. An initial schedule (possibly null) is provided by DL-management, and may be subsequently modified by the active LAS.
e) LAS transfer – the LAS transfers its role to another link master DLE, if requested by that link master or by DL-management.

An LAS may also perform the following functions:

f) Schedule construction – the LAS may process scheduling requests from other DLEs on the local link and modify the existing schedule, if any, to satisfy those requests. The schedule construction is required to take into account the maximum-scheduled-traffic, V(MST), (see 4.7.5.5) and maximum-scheduling-overhead, V(MSO), (see 4.7.5.6).
g) Schedule transfer – the LAS may send the current state of the link's schedule to another DLE on the local link, either as a backup measure or preceding an intentional transfer of the LAS role. This subclause specifies the SPDUs required to transfer the pre-constructed schedule from the LAS DLE to another LM DLE on the local link. The format of these SPDUs does not support the updating of such a transferred schedule. The procedures for schedule transfer are specified in Clause 9.

The procedures for LAS operation and DL-support operation are specified in 10.2; procedures for schedule construction requests are specified in 8.4.3; procedures for time distribution are specified in 8.4.1; procedures for token circulation are specified in 6.15; procedures for local schedule construction requests are specified in 8.4.3. The SPDUs for transferring the schedule construction requests to the LAS DLE are specified in this subclause. The procedures for schedule construction depend upon the implementation and thus are not specified.

This subclause also specifies SPDUs to support bridges, protocol-analyzers and other diagnostic tools, (see 9.3.2.7, 9.3.2.8, 9.3.2.9, 9.3.2.10, 9.3.6.4 and 9.3.6.5). The procedures for sending and receiving these SPDUs depend upon the implementation and are thus not specified.

9.3 DL-support subprotocol definition

The DL-support subprotocol defines Support Protocol Data Unit (SPDU) encodings to support the needs of LAS operation, including scheduling and other DLE functions. Other DLE support functions presently defined include mechanisms to share address and limited configuration information in support of

– DL-SUBSCRIBER-QUERY (see 8.2.3) and DL-LISTENER-QUERY (see 8.3.4) requests;
– filtering-database-maintenance within bridges (see Clause 9));
– network analysis (link analyzer) support tools.

Any DLPDU sent to, or by, the DL-support functions within a DLE, including any DLPDU addressed to a NODE DL-address, which has a non-null user data field, shall contain as "user data" a single SPDU whose encoding and interpretation is as described in this subclause. This requirement includes any DLPDU addressed to a DLSAP-address designating LAS functionality, such as link-local DL-address 0400_{16}. It also includes any PR or TL DLPDU, both of which always have a user data field.

9.3.1 Common definitions

9.3.1.1 SPDU header

The first octet of each SPDU shall specify a header common to all SPDUs, as depicted in Table 80. It shall contain:

a) an SPDU classification;
b) for some SPDUs, the protocol version, whose value is specified in Clause 7.

Table 80 – SPDU 1st octet: SPDU class, and protocol version or subclass

SPDU class					Protocol version or subclass		
CCCCC ≤ 11011					VVV		
CCCCC ≥ 11100					SSS		
7	6	5	4	3	2	1	0

When the SPDU class is in the range 00 to $1B_{16}$, then the protocol version, VVV, whose value is specified at the beginning of Clause 7, shall be present and shall be encoded as in 7.1a)4).

NOTE 1 The resulting range of values for the octet is 00 to DF_{16}.

When the SPDU class is in the range $1C_{16}$ to $1F_{16}$, then the subclass field, SSS, shall be present.

NOTE 2 The resulting range of values for the octet is $E0_{16}$ to FF_{16}.

Some of the SPDUs contain only the header octet. The SPDUs fall into five classes:

a) SPDUs supporting normal link initialization, DLE activation, and other link maintenance functions,
b) SPDUs supporting LAS transfer,
c) SPDUs supporting schedule construction,
d) SPDUs supporting schedule transfer,
e) SPDUs supporting non-LAS features of the DL-protocol, such as DL-SUBSCRIBER-QUERY and DL-LISTENER-QUERY requests, bridges, remote DL-management, and special-purpose DL-protocol analyzers.

9.3.1.2 Booleans

All Boolean elements share a common encoding:

0 FALSE;

1 TRUE.

9.3.2 Link-maintenance SPDUs

9.3.2.1 Probe-response SPDU

The probe-response SPDU is sent in a Probe Response (PR) DLPDU (see 6.14.1). The probe-response SPDU shall be formatted as specified in Table 81. Multi-octet values shall be encoded with the most significant octet of the value encoded in the lowest-index octet of the multi-octet field:

Table 81 – Probe-response SPDU

Octet index	Contents of field							
1	0000 0VVV							
2	DL-protocol versions supported							
3	N	F	CC		V(MICD)			
4	V(MICD)							
5 **6**	V(MRD) × V(ST)							
7	V(MID)							
8	V(RID)							
	7	6	5	4	3	2	1	0

a) The SPDU header octet, octet 1, shall be encoded as 0000 0VVV, which identifies this as a DL-probe-response SPDU.

Table 82 – DL-protocol versions supported

DL-protocol versions supported							
V7	V6	V5	V4	V3	V2	V1	V0
7	6	5	4	3	2	1	0

b) Octet 2 shall specify the version(s) of the DL-protocol supported, as depicted in Table 82, where each V_K shall be encoded as a Boolean indicating whether protocol version K is supported.

Table 83 – PR-SPDU: 3rd and 4th octets

Token circulation not needed	**Fractional duty cycle operation**	**Expected DLE functional class**		**Maximum-inactivity-to-claim-LAS-delay** (most significant bits)			
N	F	CC		V(MICD) / 256			
15	14	13	12	11	10	9	8

Maximum-inactivity-to-claim-LAS-delay (least significant bits)							
V(MICD) modulo 256							
7	6	5	4	3	2	1	0

c) Octets 3 and 4 shall specify, as depicted in Table 83:

1) the DLE’s lack of need for token circulation without an explicit request, encoded as a Boolean, N: 0 (no, token circulation is needed) or 1 (yes, token circulation is not needed);

2) whether the DLE will function as an FDC DLE which can be expected to be non-responsive to some live-list link-maintenance queries, and whether that DLE should be

included in the expected-non-response list, V(ENRL) (see 4.7.5.4), encoded as a Boolean, F;

3) the expected functional class of DLE operation: basic, link master or bridge (see 4.6), to enable the current LAS to determine whether it should query the new DLE's capabilities further and potentially initiate a transfer of the LAS role to the new DLE, encoded as:

   00 not reported;

   NOTE This alternative exists solely for backward compatibility with the earlier editions of IEC 61158-4.

   01 Basic;

   10 Link Master;

   11 Bridge;

NOTE The latter three values are the same coding as specified in 9.3.2.8f)1).

4) if the responding DLE is a link master then that DLE's required maximum-inactivity-to-claim-LAS-delay, V(MICD) defined in 4.7.5.19, else zero.

NOTE If this value is larger than the link's configured value of V(MICD), then the responding DLE is not capable of operating as LAS on the link as specified in Clause 9.

d) Octets 5 and 6 shall specify the responding DLE's required value of maximum-response-delay-in-octets, V(MRD) × V(ST), defined in 4.7.1.3 and 4.7.1.1. If this value, which reflects the device's capability, is larger than the link's configured value of maximum-response-delay-in-octets, V(MRD) × V(ST), then the responding node is not capable of operating on the local link.

e) Octet 7 shall specify the responding DLE's required value of minimum-inter-DLPDU-delay, V(MID) defined in 4.7.1.12. If this value, which reflects the device's capability, is larger than the link's configured value of V(MID), then the responding DLE is not capable of operating on the local link.

f) Octet 8 shall specify the value for the sending DLE's random identifier variable, V(RID) (see 4.7.1.18), which is re-randomized upon receipt of a PN DLPDU addressed to the DLE.

9.3.2.2 Node-activation SPDU

The node-activation SPDU specifies the DL-configuration parameters for the receiving DLE's link. Its primary purpose is to permit the early initiation of DL-services and thus permit higher-layer protocol stacks, including any OSI management protocol stack, to become fully operational.

The node-activation SPDU shall be formatted as specified in Table 84. Multi-octet values shall be encoded with the most significant octet of the value encoded in the lowest-index octet of the multi-octet field:

Table 84 – Node-activation SPDU

<table>
<tr><th>Octet index</th><th colspan="8">Contents of field</th></tr>
<tr><td>1</td><td colspan="8">0000 1VVV</td></tr>
<tr><td>2</td><td colspan="4">value for V(MRC)</td><td colspan="4">value for V(NRC)</td></tr>
<tr><td>3</td><td colspan="8">value for V(PhLO)</td></tr>
<tr><td>4</td><td colspan="5">reserved</td><td colspan="3">value for V(TSC)</td></tr>
<tr><td>5
6</td><td colspan="8">value for V(MEP)</td></tr>
<tr><td>7
8</td><td colspan="8">value for V(TL)</td></tr>
<tr><td>9
10</td><td colspan="8">value for V(NDL)</td></tr>
<tr><td>11</td><td colspan="8">V(RID)</td></tr>
<tr><td></td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
</table>

a) The SPDU header octet, octet 1, shall be encoded as 0000 1VVV, which identifies this as a node-activation SPDU.

b) Octet 2 shall specify in two quartets the value for the receiving DLE's maximum-retry-count variable, V(MRC), as specified in 4.7.1.5, and the value for the receiving DLE's network-retry-count variable, V(NRC), as specified in 4.7.1.6. Both values are properties of the local link.

c) Octet 3 shall specify the value for the per-DLPDU-PhL-overhead, as specified in 4.7.1.2. When the value for this variable is not configured, a value corresponding to the LAS DLE's own associated PhE, as then configured, shall be used.

Table 85 – Node-activation SPDU: 4th octet

<table>
<tr><th colspan="5">reserved</th><th colspan="3">V(TSC)</th></tr>
<tr><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
</table>

d) Octet 4 shall specify, as depicted in Table 85:
   1) reserved for future use, encoded as 00000;
   2) value to be used by the receiving DLE as the local link's time-synchronization-class, V(TSC), as specified in 4.7.1.25, using the coding specified in 7.6 b)1).

e) Octets 5 and 6 shall specify the value for the receiving DLE's DL-MAC-address-embedding-prefix variable, V(MEP), as specified in 4.7.1.10.

f) Octets 7 and 8 shall specify the value for the receiving DLE's this-link variable, V(TL), as specified in 4.7.1.9. When the appropriate value for this variable is unknown, the LAS DLE shall use a value of zero.

g) Octets 9 and 10 shall specify the value for the receiving DLE's network-DLPDU-lifetime variable, V(NDL), as specified in 4.7.1.7.

h) Octet 11 shall specify the value of the addressed DLE's random identifier variable, V(RID) (see 4.7.1.18), as recorded by the LAS from a prior-received probe-response SPDU.

9.3.2.3 LAS-data-base-status SPDU

The LAS-data-base-status SPDU is sent by the LAS to all LM DLEs on the local link to publish the latest version number for the active schedule and the latest revision number for the local-link live-list, V(LL) (see 4.7.5.2). It is also sent in each TL DLPDU. The LAS-data-base-status SPDU shall be encoded as specified in Table 86. Multi-octet values shall be encoded with the most significant octet of the value encoded in the lowest-index octet of the multi-octet field:

Table 86 – LAS-data-base-status SPDU

Octet index	Contents of field
1	1111 0000
2	schedule-type
3 **4**	schedule version-number
5 **6**	live-list revision-number

a) The SPDU header octet, octet 1, shall be encoded as 1111 0000, specifying an LAS-data-base-status SPDU.

Table 87 – LAS-data-base-status SPDU: 2nd octet

Schedule-type							
T	reserved					D	S
7	6	5	4	3	2	1	0

b) Octet 2 shall specify, as depicted in Table 87:

1) the LAS's capability to transfer its schedule, T, encoded as a Boolean: 0 (no, LAS is not capable) or 1 (yes, LAS is capable of transferring its schedule to other LM DLEs);
2) reserved for future use, encoded as 00000;
3) whether all or part of the active schedule has been dynamically constructed by the LAS, D, encoded as a Boolean;
4) whether all or part of the active schedule has been statically constructed by DL-management, S, encoded as a Boolean;

NOTE If the LAS has an active schedule, the two subfields D and S cannot both be encoded as 0.

c) Octets 3 and 4 shall specify the version-number of the currently active schedule. If there is no active schedule, then its value shall be zero; otherwise, it shall have a non-zero value.

d) Octets 5 and 6 shall specify the current live-list revision-number. It is always a non-zero value, which is incremented (modulo 2^{16}) every time the LAS detects a change in the live-list.

9.3.2.4 Live-list-change SPDU

The Live-list-change SPDU is sent by the LAS to all LM DLEs on the local link, whenever the LAS has detected a change in the status of one or more DLEs. This SPDU shall be encoded as specified in Table 88. Multi-octet values shall be encoded with the most significant octet of the value encoded in the lowest-index octet of the multi-octet field.

Table 88 – Live-list- change SPDU

Octet index	Contents of field
1	1111 0001
2 **3**	live-list revision-number
4 **5**	DLE-status
...	DLE-status

a) The SPDU header octet, octet 1, shall be encoded as 1111 0001, specifying a Live-list-change SPDU.

b) Octets 2 and 3 shall specify the live-list revision-number. It is incremented every time the LAS detects a change in the live-list and prepares the live-list-SPDU, but it is not incremented if the change is added to an SPDU which has not yet been published by the LAS. The value in this SPDU is the revision-number which applies to the live-list immediately after the changes specified in this SPDU. When this number reaches its maximum value, the next time it is incremented, it is incremented to 1 (thus this number is always non-zero).

c) The remainder of the SPDU is an array of two octet members specifying DLE-status and structured as shown in Table 89:

   1) The first octet of each member shall specify the NODE DL-address of the DLE which is affected by the change.

   2) The second octet of each member shall specify the status of the DLE, encoded as.

      i) N, the DLE's non-need for token circulation, as returned by that DLE in a previously sent PR SPDU (see 9.3.2.1c)1));

      ii) F, whether the DLE is an FDC DLE, as returned by that DLE in a previously sent PR SPDU (see 9.3.2.1c)2));

      iii) reserved for future use, encoded as 0000;

      iv) SS, the last-observed status of that DLE, encoded as

         01: not present,

         10: present but presumed now asleep, which is only possible for FDC DLEs, or

         11: present and awake.

Table 89 – DLE-status structure

Octet sub-index	Contents of subfield							
1	NODE DL-address of the described DLE							
2	N	F	reserved				SS	
	7	6	5	4	3	2	1	0

The length of this SPDU may approach 128 octets.

9.3.2.5 Live-list-request SPDU

This SPDU requests that the current LAS DLE publish the entire live-list. The Live-list-request SPDU shall be encoded with the format specified in Table 80, as follows:

a) The SPDU header octet, octet 1, shall be encoded as 1111 0010, specifying a Live-list-request SPDU.

9.3.2.6 Live-list-detail SPDU

The Live-list-detail SPDU is sent by the LAS to all LM DLEs on the local link. This SPDU shall be encoded as specified in Table 90. Multi-octet values shall be encoded with the most significant octet of the value encoded in the lowest-index octet of the multi-octet field

Table 90 – Live-list-detail SPDU

Octet index	Contents of field
1	1111 0011
2	NODE DL-address of LAS's LM DLE
3 **4**	live-list revision-number
5	V(FUN)
6	V(NUN)
7 ...	Liveness-status
... ...	DLE-type

a) The SPDU header octet, octet 1, shall be encoded as 1111 0011, specifying a Live-list-detail SPDU.

b) Octet 2 shall specify the NODE DL-address of the link-master DLE which is currently providing the LAS functionality.

c) Octets 3 and 4 shall specify the current live-list revision-number.

d) Octet 5 shall specify the current value of first-unpolled-node, V(FUN) (see 4.7.5.15).

e) Octet 6 shall specify the current value of number-of-consecutive-unpolled-nodes, V(NUN), (see 4.7.5.16).

f) Octet 7 and following shall specify the liveness-status of up to 240 DLEs, one bit per DLE, in order of descending NODE DL-address, encoded as an array of Booleans: 0 (DLE is not communicating and presumed "dead"), 1 (DLE is communicating and thus "alive").

   The DLEs not expected to be connected to the link, as specified by the variables V(FUN) and V(NUN), are not included in this status. The array of Booleans shall be ordered so that bit 7 of octet 7 specifies the status of any DLE associated with the highest address (FF_{16}), bit 6 shall specify the status of any DLE associated with the next highest address (FE_{16}), and so on. The unused bits of the last octet shall be set to 0.

g) This field specifies the FDC status for up to 240 DLEs, one bit per DLE, in order of descending NODE address, encoded as an array of Booleans: 0 (DLE is not an FDC DLE) or 1 (DLE is an FDC DLE). The DLEs not expected to be connected to the link, as specified by the variables V(FUN) and V(NUN), are not included in this status. The array of Booleans shall be ordered so that bit 7 of first octet of this field specifies the status of any DLE associated with the highest address (FF_{16}), bit 6 shall specify the status of any DLE associated with the next highest address (FE_{16}), and so on. The DLE-type for any DLE with a liveness-status of 0, as indicated by the corresponding Boolean encoded in f), shall be encoded as 0.

   Trailing octets with the value zero shall be omitted from this field and from the SPDU. Therefore, when the local link has no FDC DLEs, then this entire field shall be omitted from the SPDU.

9.3.2.7 DL-conformance-query SPDU

The DL-conformance-query SPDU requests the receiving DLE to send a DL-conformance SPDU specifying the DL-conformance classes of the replying DLE to the requesting DLE. Its purposes are

1) to permit a new LAS DLE's DL-management to assess the impact of DLEs which are already-connected to the local link, facilitating recovery of the LAS role after failure of the previous LAS DLE, and to facilitate potential transfer of the LAS role to a more-appropriate DLE;

2) to permit remote DL-management to interrogate DLEs on the extended link, by providing a query (and reply) which can be forwarded through a bridge;
3) to facilitate DL-protocol analyzer initialization after attachment to an operating local link.

The DL-conformance-query SPDU shall be encoded with the format specified in Table 80, as follows:

a) The SPDU header octet, octet 1, shall be encoded as 1110 1100, specifying a DL-conformance-query SPDU.

9.3.2.8 DL-conformance-reply SPDU

The DL-conformance-reply SPDU shall be formatted as specified in Table 91. Multi-octet values shall be encoded with the most significant octet of the value encoded in the lowest-index octet of the multi-octet field.

Table 91 – DL-conformance-reply SPDU

Octet index	Contents of field
1	0010 0VVV
2	DL-protocol versions supported
3 **…** **6**	DL-conformance encoding

a) The SPDU header octet, octet 1, shall be encoded as 0010 0VVV, which identifies this as a DL-conformance-reply SPDU.

Table 92 – DL-protocol versions supported

DL-protocol versions supported							
V_7	V_6	V_5	V_4	V_3	V_2	V_1	V_0
7	6	5	4	3	2	1	0

b) Octet 2 shall specify the version(s) of the DL-protocol supported, as depicted in Table 92, where each V_K shall be encoded as a Boolean indicating whether protocol version *K* is supported.

Table 93 – DL-conformance encoding (portion 1)

Supported connectionless extended data delivery features		**Supported DL-priorities**	**Timeliness time-stamp support**	**Supported types of timeliness**			
remote-DLE-confirmed unitdata transfer	unitdata exchange	multi-priority	DL-time-stamped DLSDU support	TRANSPARENT	RESIDENCE	UPDATE	SYNCHRONIZED
C	X	M	D	T	R	U	S
7	6	5	4	3	2	1	0

c) Octet 3 shall specify, as depicted in Table 93.
   1) the DLE's support for connectionless-mode extended data-delivery features, encoded as a set of two Booleans;
   2) the DLE's support for multiple DL-priorities, encoded as a Boolean;
   3) the DLE's support for buffer time-stamps on DLCs, encoded as a Boolean;

4) the DLE's supported types of DLC timeliness computation, encoded as a set of four Booleans.

Table 94 – DL-conformance encoding (portion 2)

Subscribers-to-publisher support	**Supported DLCEP data-delivery features**		**Supported maximum DLSDU : DLPDU data ratio**				
X	FF		RRRRR				
7	6	5	4	3	2	1	0

d) Octet 4 shall specify, as depicted in Table 94.

1) the DLE's support for the subscribers-to-publisher direction of multi-peer DLSDU transfer, X, as specified in Clause 9 and 11.9.2.3, which shall be encoded as a Boolean;

2) the DLE's supported DLCEP data-delivery features, FF, as specified in Clause 9 and 11.9.2.2, which shall be encoded as one less than the relevant item number of 11.9.2.2, with a value of 0 to 3;

3) the DLE's maximum supported DLSDU : DLPDU data ratio for ORDERED DLCEPs, RRRRR, as specified in Clause 9 and 11.9.2.4, which shall be encoded as the relevant supported value from 11.9.2.4, with a value of 0 to 16.

Table 95 – DL-conformance encoding (portion 3)

reserved				**Supported time-based scheduling services**	**Supported DL-time synchronism class**		
000				S	TTT		
7	6	5	4	3	2	1	0

e) Octet 5 shall specify, as depicted in Table 95.

1) a four-bit field, coded as zero, reserved for future standards use;

2) the DLE's support for time-based scheduling services, S, as specified in Clause 9 and 11.9.3.2, which shall be encoded as a Boolean;

3) the DLE's DL-time synchronism class, TTT, as specified in 10.6.3 and 11.9.3.1, which shall be encoded as specified in 7.6b)1).

All other values for these fields are invalid and shall not be used.

Table 96 – DL-conformance encoding (portion 4)

Supported DLE class		**Support for LONG DL-addresses**	**Fractional duty cycle operation**	**Token circulation desired**	**Supported LAS time-based scheduling activity class**		
CC		A	F	D	SSS		
7	6	5	4	3	2	1	0

f) Octet 6 shall specify, as depicted in Table 96.

1) the DLE's current class of operation, CC, as specified in 4.6 and 11.10.1, which shall be encoded as the relevant item number of 11.10.1, with a value of 1 to 3;

2) the DLE's support for LONG DL-addresses, encoded as a Boolean;

3) whether the DLE is an FDC DLE, F, which can be expected to be non-responsive to some live-list link-maintenance queries, encoded as 0 (not an FDC DLE) or 1 (an FDC DLE);

4) the DLE's need for token circulation without an explicit request, D, encoded as 0 (token circulation desired) or 1 (token circulation not desired);

5) the DLE's support for LAS time-based scheduling activities, SSS, as specified in Clause 9 and 11.9.3.3, which shall be encoded as the relevant item number of 11.9.3.3, with a value of 1 to 5;

All other values for these fields are invalid and shall not be used.

9.3.2.9 Link-basic-parameters-request SPDU

The link-basic-parameters-request SPDU requests the link-basic-parameters SPDU from the addressed LAS. The link-basic-parameters-request SPDU shall be encoded as specified in Table 80:

– The SPDU header octet, octet 1, shall be encoded as 1110 1101, specifying a link-basic-parameters-request SPDU.

9.3.2.10 Link-basic-parameters-reply SPDU

The link-basic-parameters-reply SPDU specifies the DL-configuration parameters for basic-class DLEs on the receiving DLE's link. The link-basic-parameters-reply SPDU shall be formatted as specified in Table 97. Multi-octet values shall be encoded with the most significant octet of the value encoded in the lowest-index octet of the multi-octet field.

Table 97 – Link-basic-parameters-reply SPDU

Octet index	Contents of field
1	0010 1VVV
2 **…** **10**	same as octets 2…10 of node-activation SPDU
11 **12**	value for V(ST)
13	value for V(MRD)
14	value for V(MID)

a) The SPDU header octet, octet 1, shall be encoded as 0010 1VVV, which identifies this as a link-basic-parameters-reply SPDU.

b) Octets 2 to 10 shall be identical to the octets 2 to 10 of the node-activation SPDU (see 9.3.2.2).

c) Octets 11 and 12 shall specify the value for the receiving DLE's slot-time variable, V(ST), as specified in 4.7.1.1.

d) Octet 13 shall specify the value for the receiving DLE's maximum-reply-delay variable, V(MRD), as specified in 4.7.1.3.

e) Octet 14 shall specify the value for the receiving DLE's minimum-inter-DLPDU-delay variable, V(MID), as specified in 4.7.1.12. When the appropriate value for this variable is unknown, a value of zero shall be used.

9.3.2.11 Link-master-parameters-request SPDU

The link-master-parameters-request SPDU requests the link-parameters SPDU from the addressed LAS. The link-master-parameters-request SPDU shall be encoded as specified in Table 80.

The SPDU header octet, octet 1, shall be encoded as 1110 1110, specifying a link-master-parameters-request SPDU.

9.3.2.12 Link-master-parameters-reply SPDU

The link-master-parameters-reply SPDU specifies the DL-configuration parameters for link-master-class DLEs on the receiving DLE's link. Its primary purpose is to permit link-master DLEs to prepare to assume the role of LAS on the local link. The link-master-parameters-reply SPDU shall be formatted as specified in Table 98. Multi-octet values shall be encoded with the most significant octet of the value encoded in the lowest-index octet of the multi-octet field:

Table 98 – Link-master-parameters-reply SPDU

Octet index	Contents of field
1	0011 0VVV
2	value for V(MST)
3 4	value for V(DMDT)
5 6	value for V(DTHT)
7 8	value for V(LTHT)
9 10	value for V(TTRT)
11	value for V(FUN)
12	value for V(NUN)
13 14	value for V(TDP)
15 16	value for V(MICD)
17 18	value for V(LDDP)
19 20	value for V(ML)

a) The SPDU header octet, octet 1, shall be encoded as 0011 0VVV, which identifies this as a link-master-parameters-reply SPDU.

b) Octet 2 shall specify the value for the receiving DLE's local link's maximum-scheduled-traffic variable, V(MST), as specified in 4.7.5.5.

c) Octets 3 and 4 shall specify the value for the receiving DLE's local link's default-minimum-token-delegation-time variable, V(DMDT), as specified in 4.7.5.7.

d) Octets 5 and 6 shall specify the value for the receiving DLE's local link's default-token-holding-time variable, V(DTHT), as specified in 4.7.5.8.

e) Octets 7 and 8 shall specify the value for the receiving DLE's local link's link-maintenance-token-holding-time variable, V(LTHT), as specified in 4.7.5.9.

f) Octets 9 and 10 shall specify the value for the receiving DLE's local link's target-token-rotation-time variable, V(TTRT), as specified in 4.7.5.11.

g) Octet 11 shall specify the value for the receiving DLE's local link's first-unpolled-node id variable, V(FUN), as specified in 4.7.5.15.

h) Octet 12 shall specify the value for the receiving DLE's local link's number-of-consecutive-unpolled-nodes ids variable, V(NUN), as specified in 4.7.5.16.

j) Octets 13 and 14 shall specify the value for the receiving DLE's local link's time-distribution-period variable, V(TDP), as specified in 4.7.1.18. When the appropriate value for this variable is unknown, the minimum required value for the LAS's time-synchronism class (see 10.6.3) shall be used.

k) Octets 15 and 16 shall specify the value for the receiving DLE's local link's maximum-inactivity-to-claim-LAS-delay variable, V(MICD), as specified in 4.7.5.19.

l) Octets 17 and 18 shall specify the value for the receiving DLE's local link's LAS-data-base-distribution-period variable, V(LDDP), as specified in 4.7.1.20.

m) Octets 19 and 20 shall specify the value for a receiving bridge DLE's maximum-link variable, V(ML), as specified in 4.7.6.1. Non-bridge DLEs which are functioning as an LAS DLE may specify a value of zero for this field of the SPDU when they do not maintain the proper value of V(ML) for the local link.

9.3.2.13 Token-hold-time-request SPDU

The token-hold-time-request SPDU requests the token-hold-time-array SPDU from the addressed LAS. The token-hold-time-request SPDU shall be encoded as specified in Table 99:

Table 99 – Token-hold-time-request SPDU

Octet index	Contents of field
1	1110 1111
2	starting NODE DL-address

a) The SPDU header octet, octet 1, shall be encoded as 1110 1111, specifying a token-hold-time-request SPDU.

b) Octet 2 shall specify the NODE DL-addresses for which maximum-token-hold-time is requested. The lower quartet of this octet shall always be 0. If maximum-token-hold-time for all DLEs is requested, then the value of this field shall be 00. If the value of this field is $N0_{16}$, where N = 1 to F_{16}, then the request is for DLEs with NODE DL-addresses from $N0_{16}$ to NF_{16}.

9.3.2.14 Token-hold-time-array SPDU

The token-hold-time-array SPDU specifies the maximum-token-hold-time-array parameters (see 4.7.5.10) for DLEs on the receiving DLE's link. Its primary purpose is to permit link-master DLEs to prepare to assume the role of LAS on the local link. This SPDU is sent by the LAS either to all link master DLEs to update their data base, or to a specific link master DLE as a response to token-hold-time-request from that link master DLE. The token-hold-time-array SPDU shall be formatted as specified in Table 100. Multi-octet values shall be encoded with the most significant octet of the value encoded in the lowest-index octet of the multi-octet field:

Table 100 – Token-hold-time-array SPDU

Octet index	Contents of field
1	0011 1VVV
2	starting NODE DL-address
3 4	maximum-token-hold-time
33 34	maximum-token-hold-time

a) The SPDU header octet, octet 1, shall be encoded as 0011 1VVV, which identifies this as a token-hold-time-array SPDU.

b) Octet 2 shall specify the NODE DL-addresses for which maximum-token-hold-time is included in this SPDU. The lower quartet of this octet shall always be 0. If the value of this field is $N0_{16}$, where N = 1 to F_{16}, then this SPDU includes the maximum-token-hold-time for DLEs with NODE DL-addresses from $N0_{16}$ to NF_{16}. If the value of this field is 00, then this SPDU does not include the maximum-token-hold-time for any DLE, and it is used to indicate that the LAS has sent all of token-hold-time-array SPDUs, so that the receiving DLE can do error recovery, if it had not received all of such SPDUs.

c) Octets 3 to 34 shall specify the value of maximum-token-hold-time parameter for DLEs as an array of two octet values in the ascending sequence of NODE DL-addresses starting with $N0_{16}$ and ending with NF_{16}. If any DLE in this array is constructed or configured so that it does not need to receive a token delegated by a PT DLPDU, so that the value of the N subfield in 9.3.2.1c)1) was 1, then the maximum-token-hold-time for that DLE shall be set to zero.

9.3.2.15 FDC-DLE-has-“awakened” SPDU

The FDC-DLE-has-“awakened” SPDU notifies the local LAS DLE that the specified DLE is now online and ready to receive any DLPDUs which the LAS DLE has stored, awaiting the FDC DLE’s return to active participation on the local link.

The FDC-DLE-has-“awakened” SPDU shall be encoded with the format specified in Table 80, as follows:

– The SPDU header octet, octet 1, shall be encoded as 1111 1000, specifying an FDC-DLE-has-“awakened” SPDU.

9.3.2.16 FDC-DLE-may-“go-to-sleep”-notification SPDU

The FDC-DLE-may-“go-to-sleep”-notification SPDU notifies the local LAS DLE that the specified DLE is now ready to go temporarily offline, and that the LAS DLE should act as a bridge with respect to any non-WK DLPDUs addressed to the FDC DLE until the FDC DLE next sends an FDC-DLE-has-“awakened” SPDU.

The FDC-DLE-may-“go-to-sleep”-notification SPDU shall be encoded with the format specified in Table 80, as follows:

– The SPDU header octet, octet 1, shall be encoded as 1111 1001, specifying an FDC-DLE-may-“go-to-sleep”-notification SPDU.

9.3.2.17 FDC-DLE-may-“go-to-sleep”-acknowledge SPDU

The FDC-DLE-may-“go-to-sleep”-acknowledge SPDU notifies the requesting DLE that the local LAS DLE is prepared to act as a bridge with respect to any non-WK DLPDUs addressed to the FDC DLE until the FDC DLE next sends an FDC-DLE-has-“awakened” SPDU.

The FDC-DLE-may-“go-to-sleep”-acknowledge SPDU shall be encoded with the format specified in Table 80, as follows:

– The SPDU header octet, octet 1, shall be encoded as 1111 1010, specifying an FDC-DLE-may-“go-to-sleep”-acknowledge SPDU.

9.3.3 LAS-transfer SPDUs

LAS-transfer SPDUs are used to coordinate link schedule information on a local link, and to transfer the LAS role from one DLE to another DLE on the same link.

9.3.3.1 Relinquish-LAS-role-request SPDU

This SPDU requests that the current LAS DLE transfer its role as LAS to the requesting DLE. If the current LAS DLE accepts the request, it shall transfer the LAS role by sending a TL DLPDU addressed to that requesting DLE as specified in 6.20.3. The relinquish-LAS-role-request SPDU shall be encoded with the format specified in Table 80, as follows:

– The SPDU header octet, octet 1, shall be encoded as 1111 0100, specifying a relinquish-LAS-role-request SPDU.

9.3.3.2 Accept-LAS-role-request SPDU

This SPDU requests that the receiving DLE prepare to accept the role as LAS from the requesting LAS DLE. The accept-LAS-role-request SPDU shall be encoded with the format specified in Table 80, as follows:

– The SPDU header octet, octet 1, shall be encoded as 1111 0101, specifying an accept-LAS-role-request SPDU.

9.3.3.3 Accept-LAS-role-reply SPDU

This SPDU is sent in response to an accept-LAS-role-request SPDU (see 9.3.3.2). By sending the response, the sending DLE accepts the request to become the LAS in the near future and agrees to take whatever actions are necessary (such as LAS-database synchronization) to prepare for the transfer of the LAS role. When the sending DLE is ready to accept the LAS role, it shall initiate the transfer by sending a relinquish-LAS-role-request SPDU (see 9.3.3.1) to the LAS DLE. The accept-LAS-role-reply SPDU shall be encoded with the format specified in Table 80, as follows:

– The SPDU header octet, octet 1, shall be encoded as 1111 0110, specifying an accept-LAS-role-reply SPDU.

9.3.4 Schedule-construction SPDUs

Schedule-construction SPDUs convey requests from a non-LAS DLE to the LAS DLE on the same link, and replies from that LAS DLE to the requesting DLE. They can also be used to cancel previously requested scheduling actions.

9.3.4.1 Sequence description encoding

Sequence descriptions are used in schedule construction (see 9.3.4.2) SPDUs to define desired elements of LAS activity. Each sequence description consists of a series of encoded sequence elements, with the order of sequence elements the same as that in the defining DLS-user-specified sequence.

Each encoded sequence element consists of a header, as specified in Table 101, followed by two or more octets of associated information.

Table 101 – Sequence element header encoding

Item type and format						Priority	
TTTTTT						PP	
7	6	5	4	3	2	1	0

The sequence-element header shall encode the type and format, and priority if relevant, of the sequence element. When present, the priority field shall be encoded as specified in 5.2.1.3.

9.3.4.1.1 SHORT DL-address and duration sequence element

Table 102 – SHORT DL-address and duration sequence element

Octet index	Contents of field
1	TTTT TTPP
2 **3**	SHORT DL-address
4 **5**	duration

A SHORT DL-address and duration sequence element shall be encoded as specified in Table 102:

a) The sequence element header octet, octet 1, shall be encoded as TTTT TTPP, in accordance with Table 101, as follows:

   01_{16} – CD-request, SHORT DL-address and associated duration, URGENT priority;

   02_{16} – CD-request, SHORT DL-address and associated duration, NORMAL priority;

   03_{16} – CD-request, SHORT DL-address and associated duration, TIME-AVAILABLE priority;

   04_{16} – ES-request, SHORT DL-address and associated duration;

b) Octets 2 and 3 of the sequence element shall specify a SHORT DLCEP-address (CD-request) or SHORT DLSEP-address (ES-request).

c) Octets 4 and 5 shall specify a two-octet maximum duration of reply-token (CD-request) or delegated-token (ES-request) usage, measured in octets of link transmission capacity.

   This is the maximum duration of a single instance of usage of the specified token by the addressed DLE, and does not include the link capacity required to send CD or ES DLPDU, immediate-response-recovery-delay slot-times, V(IRRD) × V(ST) octet-durations and the retry considerations of the LAS DLE in sending the token through transmission of the CD or ES DLPDU.

9.3.4.1.2 LONG DL-address and duration sequence element

Table 103 – LONG DL-address and duration sequence element

Octet index	Contents of field
1	TTTT TTPP
2 3 4 5	LONG DL-address
6 7	duration

A LONG DL-address and duration sequence element shall be encoded as specified in Table 103.

a) The sequence element header octet, octet 1, shall be encoded as TTTT TTPP, in accordance with Table 101, as follows:

   09_{16} – CD-request, LONG DL-address and associated duration, URGENT priority;

   $0A_{16}$ – CD-request, LONG DL-address and associated duration, NORMAL priority;

   $0B_{16}$ – CD-request, LONG DL-address and associated duration, TIME-AVAILABLE priority;

   $0C_{16}$ – ES-request, LONG DL-address and associated duration.

b) Octets 2 to 5 of the sequence element shall specify a delocalized LONG DLCEP-address (CD-request) or LONG DLSEP-address (ES-request).

c) Octets 6 and 7 shall specify a two-octet maximum duration of reply-token (CD-request) or delegated-token (ES-request) usage, measured in octets of link transmission capacity.

   This is the maximum duration of a single instance of usage of the specified token by the addressed DLE, and does not include the link capacity required to send CD or ES DLPDU, immediate-response-recovery-delay slot-times, V(IRRD) × V(ST) octet-durations and the retry considerations of the LAS DLE in sending the token through transmission of the CD or ES DLPDU.

9.3.4.1.3 Wakeup request sequence element

Table 104 – Wakeup request sequence element

Octet index	Contents of field
1	TTTT TT00
2	NODE DL-address

A wakeup request sequence element shall be encoded as specified in Table 104:

a) The sequence element header octet, octet 1, shall be encoded as TTTT TT00, in accordance with Table 101, as follows:

00_{16} – WK-request, NODE DL-address;

b) Octet 2 of the sequence element shall specify a NODE DL-address.

NOTE While a Wakeup sequence element cannot be included in a DLS-user-originated schedule request, it can be included in a scheduling request made by a fractional-duty-cycle (FDC) DLE wishing to schedule its own "reawakening".

9.3.4.2 Schedule-request SPDU

Scheduling requests consist of requests for either one-time or periodic (that is, cyclic with a fixed period) execution of a sequence, subject to specified scheduling constraints. These requests differ only in the parameters specifying the required period and permissible jitter of cyclic execution of the sequence, which are not relevant to one-time execution of the sequence.

Table 105 – Schedule-request SPDU

Octet index	Contents of field
1	0110 0VVV
2	Sequence type, schedule type, and schedule priority
3 4	Schedule identifier
5 6	Value for V_S(TSL)
7 8 9 10 11 12	Desired starting time
13 14 15 16	Earliest starting time, relative to desired starting time
17 18 19 20	Latest ending time, relative to desired starting time
21 22 23 24	Period of cyclic sequence
25 26 27 28	Maximum permissible jitter
29 ...	Sequence description

Scheduling-request SPDUs shall be formatted as specified in Table 105.

a) The SPDU header octet, octet 1, shall be encoded as 0110 0VVV.

Table 106 – Sequence type, schedule type and priority encoding

Sequence type			Schedule type			Priority	
SSS			TTT			PP	
7	6	5	4	3	2	1	0

b) Octet 2 shall identify the sequence type, the schedule type, and the priority of the scheduling request. It shall be encoded as specified in Table 106.

1) The sequence type SSS shall specify whether the entire sequence is to be executed contiguously or whether discontiguous (interrupted) execution is permitted, and in the latter case whether the sequence as a whole is to be considered ordered or unordered:

– A **contiguous** sequence is one in which the DLS-provider must schedule enough consecutive segments of local-link capacity to execute the entire sequence.

– A **discontiguous** sequence is one in which the DLS-provider may intersperse the execution of the sequence with other link activity.

– An **ordered** sequence is one in which the DLS-provider must execute the sequence elements in the order specified.

– An **unordered** sequence is one in which the DLS-provider may rearrange the order of execution of the elements of the sequence, facilitating DLL-internal optimizations.

NOTE Contiguous sequences are always treated as ordered, since unordered permits no additional optimization.

This subfield shall be encoded in three bits as.

000) unordered discontiguous;

001) ordered discontiguous;

011) (ordered) contiguous.

All other values are reserved for future standards use.

2) The schedule type TTT shall be encoded in three bits as.

001) one-time;

100) periodic.

All other values are reserved for future standards use.

3) The priority PP shall be encoded in two bits as specified in 5.2.1.3.

c) Octets 3 and 4 shall encode a non-zero schedule-identifier provided by the requesting DLE.

NOTE This identifier enables the DLE which requests scheduling to associate a reply with its corresponding scheduling request.

d) Octets 5 and 6 shall encode the current value of the DL-time source link-id variable, V_S(TSL), of the sending DLE.

e) Octets 7 to 12 shall encode the desired starting DL-time for the first instance of execution of the scheduled sequence, in units of 2^{-5} ms, where that time is referenced to the DL-time source link-id specified in d).

f) Octets 13 to 16 shall encode the offset of the earliest acceptable starting DL-time for the first instance of execution of the scheduled sequence, in units of 2^{-5} ms, where that time is referenced to the DL-time specified in octets 7 to 12.

g) Octets 17 to 20 shall encode the offset of the latest acceptable ending DL-time for the first instance of execution of the scheduled sequence, in units of 2^{-5} ms, where that time is referenced to the DL-time specified in octets 7 to 12.

h) In periodic-scheduling SPDUs, octets 21 to 24 shall encode the repetition period for execution of a cyclic scheduled sequence, in units of 2^{-5} ms. In one-time-scheduling SPDUs, octets 21 to 24 shall be encoded as zero.

i) In periodic-scheduling SPDUs, octets 25 to 28 shall encode the maximum permissible cycle-to-nominal jitter, in units of 2^{-5} ms, in the repetitive starting times of a cyclic scheduled sequence. In one-time-scheduling SPDUs, octets 25 to 28 shall be encoded as zero.

j) The remaining octets, through the last octet of the SPDU, shall encode a description of the sequence to be scheduled, as specified in 9.3.4.1.

The length of this SPDU may approach 256 octets.

9.3.4.3 Scheduling-completed SPDU

The LAS DLE shall acknowledge receipt and completion of a schedule-request SPDU by replying with a scheduling-completed SPDU, which shall be encoded as specified in Table 107:

a) The SPDU header octet, octet 1, shall be encoded as 0110 1VVV, specifying a scheduling-completed SPDU.

Table 107 – Scheduling-completed SPDU

Octet index	Contents of field
1	0110 1VVV
2 3	Schedule identifier
4	Status or reason
5 6	value for V_S(TSL)
7 8 9 10 11 12	Scheduled starting time

b) Octets 2 and 3 shall encode the schedule-identifier provided by the requesting DLE in the schedule-request SPDU to which this SPDU is a reply.

NOTE This identifier enables the DLE which requested the scheduling to associate the reply with its corresponding scheduling request.

c) Octet 4 shall encode the status of the corresponding request – either success or the reason for failure, as specified in Table 108.

Table 108 – Status and reason codes

Status	Code
SUCCESS	00_{16}
SERVICE-UNAVAILABLE – TEMPORARY	02_{16}
RESOURCE-UNAVAILABLE – TEMPORARY	03_{16}
TIME-BASE MISMATCH – TEMPORARY	04_{16}
SERVICE-UNAVAILABLE – PERMANENT	82_{16}
RESOURCE-UNAVAILABLE – PERMANENT	83_{16}

d) If octet 4 encodes success (00), then octets 5 and 6 shall encode the current value of the DL-time source link-id variable, V_S(TSL), of the LAS DLE.

e) If octet 4 encodes success (00), then octets 7 to 12 shall encode the scheduled starting DL-time for execution of the sequence, in units of 2^{-5} ms, where that time is referenced to the DL-time source link-id specified in d).

f) If octet 4 encodes failure (any value other than 00), then this SPDU shall have only 4 octets and the next two fields shall be omitted.

9.3.4.4 Cancel-schedule SPDU

Table 109 – Cancel-schedule SPDU

Octet index	Contents of field
1	1111 1100
2 3	Schedule identifier

The DLE which requested a scheduling action of the LAS DLE may also request the cancellation of that previous scheduling action. The SPDU by which this request is made shall be encoded as specified in Table 109.

a) The SPDU header octet, octet 1, shall be encoded as 1111 1100, specifying a cancel-schedule SPDU.

b) Octets 2 and 3 shall encode the schedule-identifier provided by the requesting DLE in the schedule-request SPDU which this SPDU is canceling.

NOTE This identifier enables the LAS DLE to associate the cancellation request with the corresponding scheduling request.

9.3.4.5 Schedule-cancelled SPDU

Table 110 – Schedule-cancelled SPDU

Octet index	Contents of field
1	1111 1101
2 3	Schedule identifier
4	Status or reason

The LAS DLE shall acknowledge cancellation of a scheduled sequence by sending a schedule-cancelled SPDU to the DLE which originally made the schedule request. The schedule-cancelled SPDU shall be encoded as specified in Table 110:

a) The SPDU header octet, octet 1, shall be encoded as 1111 1101, specifying a schedule-cancelled SPDU.

b) Octets 2 and 3 shall encode the schedule-identifier provided by the requesting DLE in the schedule-request SPDU which this SPDU is canceling.

NOTE This identifier enables the DLE which requested the scheduling to associate the reply with its corresponding scheduling request.

c) Octet 4 shall encode the reason for cancellation, as specified in Table 108.

9.3.5 Schedule-transfer SPDUs

Schedule-transfer SPDUs convey a link schedule from an LAS DLE to a non-LAS DLE on the same link. This transfer allows another link master DLE to assume the LAS role and continue the execution of the link schedule. The link schedule transferred via these SPDUs could have been constructed by the current LAS or could have been originally transferred to the current LAS from another previous LAS DLE or by DL-management. The format of these SPDUs does not save the original schedule-request SPDU. Therefore, an LAS DLE which receives such a schedule from another LAS DLE or from DL-management cannot modify this schedule.

The link schedule consists of one or more periodic subschedule SPDUs and one schedule-summary SPDU as shown in Table 111. Each subschedule contains one or more sequences, all of the same execution period. An LAS may have a limit on the number of different subschedules it can execute simultaneously. The schedule can be transferred one SPDU at a time, or all SPDUs can be included in one DLPDU. If the schedule is transferred in one DLPDU, then the SPDUs should be in one contiguous sequence as shown in the Table 111. The format of each SPDU is such that the end of each SPDU can be determined without any ambiguity.

The 3-bit version number, NNN, of the schedule transfer encoding which is used in schedule transfer SPDUs shall be encoded as 000.

Table 111 – Link-schedule

Contents
Schedule-summary SPDU
Subschedule SPDU
Subschedule SPDU

9.3.5.1 Schedule-summary SPDU

The schedule-summary SPDU is a record of attributes which are common to all subschedules. This SPDU shall be encoded as specified in Table 112. Multi-octet values shall be encoded with the most significant octet of the value encoded in the lowest-index octet of the multi-octet field. If this SPDU is for a null schedule, then it shall consist of only the first four octets, and the remaining octets shown shall be omitted from the SPDU.

a) The SPDU header octet, octet 1, shall be encoded as 1111 1111, specifying a schedule transfer SPDU.

b) Octet 2 shall specify the type of schedule transfer SPDU – schedule-summary, and shall be encoded as 0000 0NNN, where NNN is the version number of the schedule transfer encoding.

Table 112 – Schedule-summary SPDU

Octet index	Contents of field
1	1111 1111
2	0000 0NNN
3 **4**	Schedule version-number
5 **6**	Schedule-builder identifier (0100...0FFF)
7	Number of subschedules
8	V(MSO)
9 **10**	Storage required for total schedule
11 **12**	Timing resolution required to meet schedule-jitter commitments
13 **14**	V(MRD) × V(ST)
15 **16**	V(TSL)
17 **18** **19** **20** **21** **22**	Periodic schedule DL-time base (T0)
23 **24** **25** **26**	Macro-cycle duration
27 **28**	Subschedule-SPDU reference
...	Subschedule-SPDU reference
...	Subschedule-SPDU reference
...	End-of-schedule-summary

c) Octets 3 and 4 shall specify the non-zero value of the schedule version-number as a two-octet integer, or shall specify the value zero. If the value of this field is zero, which is used to indicate a null schedule, then this field shall be the last field actually present in the SPDU.

d) Octets 5 and 6 shall specify the source of the schedule as an identifier with a value in the range of 0100_{16} … $0FFF_{16}$.

NOTE This field is only for reference and is not used by the LAS.

e) Octet 7 shall specify the number of subschedules in the schedule.

f) Octet 8 shall specify the value of V(MSO), maximum-scheduling-overhead (see 4.7.5.6), which was used in the construction of the schedule. Its unit of measurement and range of values is specified in 4.7.5.6.

g) Octets 9 and 10 shall specify the total storage space required for the schedule-summary and all of the subschedules in number of octets. This is the sum of number of octets in each SPDU of the schedule.

h) Octets 11 and 12 shall specify the minimum resolution of the time clock required to execute this schedule and still meet the jitter specifications to which the schedule was constructed. Its units are 2^{-5} ms.

i) Octets 13 and 14 shall specify the value of maximum-response-delay slot-times, V(MRD) × V(ST) octet-durations, which was used in the construction of the schedule. The unit of measurement and range of values are specified in 4.7.1.1 and 4.7.1.3.

j) Octets 15 and 16 shall specify the link-id of the source of time distribution, which was used to specify the DL-time used in this schedule.

k) Octets 17 to 22 shall specify the starting DL-time (T0), referenced to the DL-time source link-id specified in j), of all periodic subschedules in this schedule. The start time of all sequences in all subschedules is shown as offset with respect to this periodic schedule DL-time base. Its units are 2^{-5} ms.

l) Octets 23 to 26 shall specify the duration of macrocycle, which is equal to the least common multiple of periods of all subschedules. Its units are 2^{-5} ms.

m) Octets 27 and up to the end of the SPDU shall reference the subschedule-SPDUs by its identifier, as shown in Table 113. Each identifier shall be 2 octets long and shall have a non-zero value.

NOTE This could be a null (zero-length) sequence of identifiers.

n) The end-of-schedule-summary shall be encoded as a multiple of two octets whose values are all zero.

NOTE The schedule builder can fill the end of this SPDU with more than the minimum number of octets in the anticipation of future addition of subschedule-SPDU references.

Table 113 – Subschedule-SPDU reference

Octet index	Contents of subfield
1 2	Subschedule identifier

The length of this SPDU may approach 256 octets.

9.3.5.2 Subschedule SPDU

The subschedule SPDU is a record of attributes which are common to one subschedule and specification of all sequences for that subschedule. This SPDU shall be encoded as specified in Table 114. Multi-octet values shall be encoded with the most significant octet of the value encoded in the lowest-index octet of the multi-octet field:

Table 114 – Subschedule SPDU

Octet index	Contents of subfield
1	1111 1111
2	0000 1NNN
3 **4**	Schedule version-number
5 **6**	Subschedule identifier
7 **8** **9** **10**	Subschedule period
11 **...**	Sequence sub-SPDU
...	Sequence sub-SPDU
...	Sequence sub-SPDU
...	End-of-subschedule

a) The SPDU header octet, octet 1, shall be encoded as 1111 1111, specifying a schedule transfer SPDU.

b) Octet 2 shall specify the type of schedule transfer SPDU - subschedule, and shall be encoded as 0000 1NNN, where NNN is the version number of the schedule transfer encoding.

c) Octets 3 and 4 shall specify the two octet integer value of the version-number of the schedule which includes this subschedule; all values except zero are permitted. The value of this field shall be identical to the same field of schedule-summary SPDU.

d) Octets 5 and 6 shall either specify the non-zero value of a subschedule identifier as referenced in the schedule-summary SPDU (see 9.3.5.1d)) or it shall be zero. If the value of this field is zero, then this shall be the last octet of this SPDU, and it is used to indicate that the LAS has sent all of subschedule SPDUs, so that the receiving DLE can do error recovery, if it had not received all of such SPDUs. If the value of this octet is non-zero, then this octet shall be followed by the other fields of this SPDU as specified in e), f) and g).

e) Octets 7 to 10 shall specify a non-zero execution period of all sequences in the subschedule. Its units are 2^{-5} ms.

f) Octets 11 and up to the end of the SPDU shall specify one or more sequences as specified in 9.3.5.2.1.

g) The end-of-subschedule shall be encoded as a multiple of four octets whose values are all zero.

NOTE The schedule builder can fill the end of this SPDU with more than the minimum number of octets in the anticipation of future addition of sequence sub-SPDUs.

The length of this SPDU may approach 256 octets.

9.3.5.2.1 Sequence sub-SPDU

The sequence sub-SPDU is a sub-record of attributes which are common to one sequence and specification of all elements of that sequence. This sub-SPDU shall be encoded as specified in Table 115. Multi-octet values shall be encoded with the most significant octet of the value encoded in the lowest-index octet of the multi-octet field:

Table 115 – Sequence Sub-SPDU

Octet index	Contents of subfield
1 2 3 4	Scheduled-starting-time-offset
5 6	Maximum computed duration of sequence execution
7 ...	Element-description
...	Element-description
...	Element-description
...	End-of-sequence

a) Octets 1 to 4 shall specify a non-zero starting time of the sequence as offset from periodic schedule DL-time base (T0). Its units are 2^{-5} ms. The LAS is required to execute the entire sequence continuously starting from the first to the last element. The starting time of the sequence is

$$T0 + N \times (\text{subschedule period}) + \text{scheduled-starting-time-offset},$$

where N is a non-negative integer.

NOTE Whenever there is a change in the DL-time-offset, V(DLTO), the periodic schedule time-base, (T0), is adjusted by the same amount. Therefore, once the execution of a schedule has begun, the start time of each sequence is given by

$$V(LSTO) + C(NT) =$$
$$V_{start}(LSTO) + C_{start}(NT) + N \times (\text{subschedule-period}) + \text{scheduled-starting-time-offset},$$

where the sum $V_{start}(LSTO) + C_{start}(NT)$ is the value of V(LSTO) + C(NT) at the start of the schedule execution.

b) Octets 5 and 6 shall specify the worst-case duration of executing the sequence. This field is only for reference and LAS does not use this to execute the sequence.

c) Octet 7 and following shall specify one or more elements as specified in 9.3.5.2.2.

d) The end-of-sequence shall be encoded as two octets whose values are both zero.

9.3.5.2.2 Element

The element is the lowest level component of the schedule, and it represents a transaction (see 3.3.32). The element shall be encoded as specified in Table 116. Multi-octet values shall be encoded with the most significant octet of the value encoded in the lowest-index octet of the multi-octet field.

Table 116 – Element-description

Octet index	Contents of subfield
1	Element-type
2 ...	Element-parameter

a) Octets 1 shall specify the type of transaction and shall be encoded as:

1011 00PP – CD-request, SHORT DL-address, PP = priority (see 5.2.1.3);

1011 10PP – CD-request, LONG DL-address, PP = priority (see 5.2.1.3);

1000 0000 – ES-request, SHORT DL-address, first and last ES of one sequence;

1000 0010 – ES-request, SHORT DL-address, first ES of one sequence;

1000 0110 – ES-request, SHORT DL-address, subsequent, but not last ES of one sequence;

1000 0111 – ES-request, SHORT DL-address, last ES of one sequence;

1000 1000 – ES-request, LONG DL-address, first and last ES of one sequence;

1000 1010 – ES-request, LONG DL-address, first ES of one sequence;

1000 1110 – ES-request, LONG DL-address, subsequent, but not last ES of one sequence;

1000 1111 – ES-request, LONG DL-address, last ES of one sequence;

0000 0000 – WK-request, NODE DL-address;

b) Octet 2 and up shall specify the parameters for the element. The length and encoding depends upon the element-type.
   1) If element type is CD-request, SHORT DL-address, then the element parameter shall have two octets and these two octets, octets 2 and 3, shall specify a SHORT DLCEP-address.
   2) If element type is CD-request, LONG DL-address, then the element parameter shall have four octets and these four octets, octets 2 to 5, shall specify a LONG DLCEP-address.
   3) If the element type is ES-request, SHORT DL-address, then the element parameter shall have four octets. The first two octets of these four octets, octets 2 and 3, shall specify a SHORT DLSEP-address. The next two octets of these four octets, octets 4 and 5, shall specify duration measured in octets of link transmission capacity, DD-parameter of ES DLPDU.
   4) If the element type is ES-request, NODE DL-address, then the element parameter shall have six octets. The first four octets of these four octets, octet 2 to 5 shall specify a NODE DLSEP-address. The next two octets of these six octets, octet 6 and 7 shall specify duration measured in octets of link transmission capacity, DD-parameter of ES DLPDU.
   5) If the element type is WK-request, NODE DL-address, then the element parameter shall have one octet and this octet, octet 2 shall specify a non-zero NODE DL-address.

9.3.5.3 Schedule-summary-request SPDU

The schedule-summary-request SPDU requests the schedule-summary SPDU for the currently active schedule, from the addressed LAS. The schedule-summary-request SPDU shall be encoded as specified in Table 117.

Table 117 – Schedule-summary-request SPDU

Octet index	Contents of field
1	1111 1110
2	0000 0000

a) The SPDU header octet, octet 1, shall be encoded as 1111 1110, specifying a schedule transfer request SPDU.

b) Octet 2 shall be encoded as 0000 0000, specifying a schedule-summary-request SPDU.

9.3.5.4 Subschedule-request SPDU

The subschedule-request SPDU requests the subschedule SPDU from the addressed LAS. The subschedule-request SPDU shall be encoded as specified in Table 118:

Table 118 – Subschedule-request SPDU

Octet index	Contents of field
1	1111 1110
2	0000 1000
3 **4**	Schedule version-number
5 **6**	Subschedule identifier

a) The SPDU header octet, octet 1, shall be encoded as 1111 1110, specifying a schedule transfer request SPDU.

b) Octet 2 shall be encoded as 0000 1000, specifying a subschedule-request SPDU.

c) Octets 3 and 4 shall specify the two octet integer value of the version-number of the schedule; all values except zero are permitted.

d) Octets 5 and 6 shall either specify the non-zero value of a subschedule identifier as referenced in the schedule-summary SPDU (see 9.3.5.1m)) or it shall be zero. If the value of this field is zero, then this is a request to the LAS to send all subschedule SPDUs for the specified schedule, otherwise it is a request to the LAS to send the specified subschedule SPDU.

9.3.6 Non-LAS SPDUs

Non-LAS SPDUs are used to support DL-services other than scheduling, such as DL-SUBSCRIBER-QUERY requests and DL-LISTENER-QUERY requests, to support bridge filtering database maintenance operations, and to support protocol-analyzers. These SPDUs may be sent between any two DLEs on the extended link, and may be broadcast to all DLEs on a local link where appropriate.

9.3.6.1 Parameter lists in SPDUs

Non-LAS SPDUs are used to query or transfer DLS-provider database information from one DLE to another. These SPDUs use a common parameter-list structure to encode the transferred database items, and to specify the beginning, continuation, or ending of such a list.

Each encoded element of parameter-list consists of a header, as specified in Table 119, followed by zero or more octets of associated information.

Table 119 – Parameter-list element-header encoding

List element type				List element subtype			
TTTT				SSSS			
7	6	5	4	3	2	1	0

The list element-header shall encode the type and subtype, if relevant, of the list element. The format and primary function of the list element shall be determined by the list element type, TTTT, whose encodings shall be.

0 begin/end-of-list element;

1 continuation-of-list element;

2 SHORT DL-address element;

3 LONG DL-address element;

4 DLSAP-address-characteristics element;

5 DLCEP-characteristics element.

9.3.6.1.1 Begin/end-of-list element

Table 120 – Begin/end-of-list element

Octet index	Contents of field
1	0000 0000

A begin/end-of-list element shall be encoded as specified in Table 120:

– The list element header octet, octet 1, shall be encoded as 0000 0000, in accordance with Table 119.

9.3.6.1.2 Continuation-of-list element

Table 121 – Continuation-of-list element

Octet index	Contents of field
1	0001 0000
2 3 4 5	Continuation identifier

A continuation-of-list element shall be encoded as specified in Table 121:

a) The list element header octet, octet 1, shall be encoded as 0001 0000, in accordance with Table 119.

b) Octets 2 to 5 shall encode a continuation-identifier provided by the DLE whose DLS-provider database is being read. There is no constraint on its value or interpretation.

NOTE This identifier enables the DLE whose DLS-provider database is being read to pass a *handle* to its peer DLE, for later return in another database-accessing request, such that the two DLEs can continue the database access at later time using only the connectionless lower-level DL-protocol.

9.3.6.1.3 SHORT DL-address element

Table 122 – SHORT DL-address list element

Octet index	Contents of field
1	0010 SSSS
2 3	SHORT DL-address

Each SHORT DL-address list element shall be encoded as specified in Table 122:

a) The list element header octet, octet 1, shall be encoded as 0010 SSSS, in accordance with Table 119, and the sub-type field, SSSS, shall specify the DL-address type of the accompanying DL-address parameter, as follows:

20_{16} – SHORT group DL-address;

21_{16} – SHORT DLSAP-address whose DL(SAP)-role is BASIC;

24_{16} – SHORT DLSAP-address whose DL(SAP)-role is INITIATOR;

25_{16} – SHORT DLSAP-address whose DL(SAP)-role is UNCONSTRAINED RESPONDER;

26_{16} – SHORT DLSAP-address whose DL(SAP)-role is CONSTRAINED RESPONDER;

28_{16} – SHORT DLCEP-address of a peer DLCEP;

29_{16} – SHORT DLCEP-address of a publisher DLCEP;

$2A_{16}$ – SHORT DLCEP-address of a subscriber DLCEP's publisher;

$2B_{16}$ – SHORT DLCEP-address of a subscriber DLCEP (which exists during DLCEP-establishment);

$2C_{16}$ – SHORT DLSEP-address;

b) Octets 2 and 3 of the list element shall specify a SHORT DL-address.

9.3.6.1.4 LONG DL-address element

Table 123 – LONG DL-address element

Octet index	Contents of field
1	0011 SSSS
2 3 4 5	LONG DL-address

Each LONG DL-address list element shall be encoded as specified in Table 123:

a) The list element header octet, octet 1, shall be encoded as 0011 SSSS, in accordance with Table 119, and the subtype field, SSSS, shall specify the DL-address type of the accompanying DL-address parameter, as follows:

30_{16} – LONG group DL-address;

31_{16} – LONG DLSAP-address whose DL(SAP)-role is BASIC;

34_{16} – LONG DLSAP-address whose DL(SAP)-role is INITIATOR;

35_{16} – LONG DLSAP-address whose DL(SAP)-role is UNCONSTRAINED RESPONDER;

36_{16} – LONG DLSAP-address whose DL(SAP)-role is CONSTRAINED RESPONDER;

38_{16} – LONG DLCEP-address of a peer DLCEP;

39_{16} – LONG DLCEP-address of a publisher DLCEP;

$3A_{16}$ – LONG DLCEP-address of a subscriber DLCEP's publisher;

$3B_{16}$ – LONG DLCEP-address of a subscriber DLCEP (which exists during DLCEP-establishment);

$3C_{16}$ – LONG DLSEP-address;

b) Octets 2 to 5 of the list element shall specify a delocalized LONG DL-address.

9.3.6.1.5 DLSAP-address-characteristics element

Table 124 – DLSAP-address-characteristics element

Octet index	Contents of field
1	0100 0000
2 ... 5	Constrained-responder's remote-DLSAP-address as a LONG DL-address

Each DLSAP-address-characteristics list element shall be encoded as specified in Table 124:

a) The list element header octet, octet 1, shall be encoded as 0100 0000, in accordance with Table 119.

b) Octets 2 to 5 shall encode the remote DLSAP-address parameter of a DLSAP-address whose DL(SAP)-role binding is CONSTRAINED RESPONDER. This encoding shall be in the form of a delocalized LONG DL-address.

9.3.6.1.6 DLCEP-characteristics element

Table 125 – DLCEP-characteristics element

Octet index	Contents of field
1	0101 0000
2 3	Basic DLCEP parameters
4 … 7	Sender DLCEP parameters
8 … 11	Receiver DLCEP parameters

Each DLCEP-characteristics list element shall be encoded as specified in Table 125:

a) The list element header octet, octet 1, shall be encoded as 0101 0000, in accordance with Table 119.

b) Octets 2 and 3 shall encode the basic parameters of the DLCEP, as specified for the first and second octets of the parameters of an EC DLPDU in Table 37 and Table 38 and in 7.1a)2) to a)8), except that reply-request subfield (R) (see 7.1a)1)) shall be coded as 0.

c) Octets 4 to 7 shall encode the sender parameters of the DLCEP, as specified for the seventh to tenth octets, respectively, of an EC DLPDU in Table 41 through Table 43 and in 7.1c)1) to c)10).

d) Octets 8 and 11 shall encode the receiver parameters of the DLCEP, as specified for the eleventh to fourteenth octets, respectively, of an EC DLPDU in Table 44 through Table 46 and in 7.1d)1) to d)10).

9.3.6.2 Address-query SPDU

Table 126 – Address-query SPDU

Octet index	Contents of field
1	0100 0VVV
2	Query modifiers
3 4 5 6	Request identifier
7 …	DL-address parameter-list element

The address-query SPDU requests the receiving DLE to search its forwarding database (of DL-addresses recognized for either local reception or forwarding) for the DL-address specified by the associated single DL-address parameter list element, and to compose and send an address-report SPDU if that address is found. Its primary purpose is to support DL-SUBSCRIBER-QUERY and DL-LISTENER-QUERY requests and link protocol analyzers.

The address-query SPDU shall be formatted as specified in Table 126:

a) The SPDU header octet, octet 1, shall be encoded as 0100 0VVV, specifying an address-query SPDU.

b) Octet 2 specifies whether additional information on the DLSAP-address-binding or DLCEP should be contained within a responder’s address-report SPDU. Its encoding shall be:

00_{16} – No accompanying peer DL-address or DLCEP information is requested.

01_{16} – Responders should include list elements for peer DL-address, and, where appropriate, for DLCEP-characteristics, in the replying address-report SPDU when known.

Other values are reserved.

c) Octets 3 to 6 shall encode a non-zero request-identifier provided by the requesting DLE.

NOTE This identifier enables the requesting DLE to associate a reply with its corresponding request.

d) Octets 7 and following shall encode either one SHORT DL-address element or one LONG DL-address element, as specified in 9.3.6.1.3 and 9.3.6.1.4. The encoded DL-address shall specify the DL-address which is the subject of the query.

9.3.6.3 Address-report SPDU

Table 127 – Address-report SPDU

Octet index	Contents of field
1	0100 1VVV
2	Reason for report
3 **4** **5** **6**	Request identifier
7 **...**	Parameter-list elements

The address-report SPDU is sent in response to an address-query SPDU when a receiving DLE determines that the queried DL-address is contained within the receiving DLE's forwarding database, for either local reception or for forwarding to another DLE.

It is also sent to all local bridges to report the activation or deactivation of a DL-address

1) by a non-FDC DLE when the DL-address is non-local (that is, has a link component with a value other than zero or V(TL));
2) by an FDC DLE for all of its addresses.

NOTE DL-address activation occurs upon executing a DL-BIND or DL-CONNECT request; DL-address deactivation occurs upon executing a DL-UNBIND or DL-DISCONNECT request.

Its primary purposes are to support DL-SUBSCRIBER-QUERY and DL-LISTENER-QUERY requests, and to facilitate bridges and link protocol analyzers.

The address-report SPDU may be addressed to a DLE on another link. It shall be formatted as specified in Table 127:

a) The SPDU header octet, octet 1, shall be encoded as 0100 1VVV, specifying an address-report SPDU.
b) Octet 2 specifies the reason for the address-report SPDU; it shall be encoded as:

00_{16} – The SPDU is a reply to a prior address-query SPDU;

10_{16} – A non-FDC DLE has activated the specified DL-address;

11_{16} – An FDC DLE has activated the specified DL-address;

12_{16} – A non-FDC DLE has deactivated the specified DL-address;

13_{16} – An FDC DLE has deactivated the specified DL-address.

All other values are reserved for future standard use.

c) Octets 3 to 6 shall encode
   1) the non-zero request-identifier provided by the requesting DLE in the address-query SPDU to which this SPDU is a reply; or

   NOTE This identifier enables the querying DLE to associate the reply with its corresponding query.

   2) a value of zero indicating that this is a spontaneous report of the activation or deactivation of a DL-address, and not a reply to a prior SPDU.
d) Octets 7 and following shall encode either one SHORT DL-address element or one LONG DL-address element, as specified in 9.3.6.1.3 and 9.3.6.1.4.
e) If this SPDU is a reply to an address-query SPDU, and that associated query SPDU specified that responders should include peer DL-address and DLCEP-characteristics elements in the reply SPDU when known.
   1) the responding DLE shall append a DLSAP-address-characteristics or DLCEP-address-characteristics element specifying the characteristics of the queried DL-address to the reply SPDU, as specified in 9.3.6.1.5 and 9.3.6.1.6; and then
   2) if the queried DL-address specified a peer DLCEP-address, then the responding DLE shall append a DL-address element specifying the remote peer DL-address of the queried DL-address to the reply DLPDU, as specified in 9.3.6.1.3 and 9.3.6.1.4.

   NOTE These last two elements provide necessary functionality to support link protocol analyzers.

9.3.6.4 Address-list-query SPDU

The address-list-query SPDU requests the receiving DLE to search its forwarding database (of DL-addresses recognized for either local reception or forwarding) for DL-addresses which meet the query criteria, starting at the beginning-of-list or continuation-point specified in the query, and to include as many of those DL-addresses which meet the criteria as possible in an address-list-reply SPDU, concluding the SPDU with an end-of-list or continuation-point list element.

Table 128 – Address-list-query SPDU

Octet index	Contents of field
1	0101 0VVV
2 3	Selection criteria
4 5 6 7	Request identifier
8 …	Parameter-list element

The primary purpose of this SPDU is to provide bridges with a means of building and maintaining their forwarding databases.

The address-list-query SPDU shall be formatted as specified in Table 128:

a) The SPDU header octet, octet 1, shall be encoded as 0101 0VVV, specifying a address-list-query SPDU.

Table 129 – DL-address selection criteria

Include standard addresses	reserved			DLSEP-address	Subscriber DLCEP	Publisher DLCEP	Peer DLCEP
A	B (=0)	C (=0)	D (=0)	E	F	G	H
15	14	13	12	11	10	9	8

reserved			Constrained responder DLSAP-address	Unconstrained responder DLSAP-address	Initiator DLSAP-address	Basic DLSAP-address	Group DL-address
J (=0)	K (=0)	L (=0)	M	N	P	Q	R
7	6	5	4	3	2	1	0

b) Octets 2 and 3 specify the selection criteria for DL-addresses to be included within a responder's address-list-reply SPDU. They shall be encoded as two octets of individual selection criteria, as specified in Table 129, with octet 2 consisting of fields A to H, and octet 3 of fields J to R.

1) If field A=1, then the standard DL-addresses specified in Table 4 through Table 6 which meet the other selection criteria shall be included in the reply list; otherwise (field A=0) they shall be excluded from the reply list.

2) Fields B, C, D, J, K and L are reserved for future standards use and shall be encoded as zero.

3) If field E=1, then active DLSEP-addresses of the receiving DLE shall be included in the reply list; otherwise (field E=0) they shall be excluded from the reply list.

4) If field F=1, then (publisher) DLCEP-addresses of subscriber DLCEPs within the receiving DLE, or forwarded by the receiving DLE, shall be included in the reply list; otherwise (field F=0) they shall be excluded from the reply list.

5) If field G=1, then DLCEP-addresses of publisher DLCEPs within the receiving DLE, or forwarded by the receiving DLE, shall be included in the reply list; otherwise (field G=0) they shall be excluded from the reply list.

6) If field H=1, then DLCEP-addresses of peer DLCEPs within the receiving DLE, or forwarded by the receiving DLE, shall be included in the reply list; otherwise (field H=0) they shall be excluded from the reply list.

7) If field M=1, then DLSAP-addresses of DLSAPs within the receiving DLE, or forwarded by the receiving DLE, whose DL(SAP)-role is CONSTRAINED-RESPONDER, shall be included in the reply list; otherwise (field M=0) they shall be excluded from the reply list.

8) If field N=1, then DLSAP-addresses of DLSAPs within the receiving DLE, or forwarded by the receiving DLE, whose DL(SAP)-role is UNCONSTRAINED-RESPONDER, shall be included in the reply list; otherwise (field N=0) they shall be excluded from the reply list.

9) If field P=1, then DLSAP-addresses of DLSAPs within the receiving DLE, or forwarded by the receiving DLE, whose DL(SAP)-role is INITIATOR, shall be included in the reply list; otherwise (field P=0) they shall be excluded from the reply list.

10) If field Q=1, then DLSAP-addresses of DLSAPs within the receiving DLE, or forwarded by the receiving DLE, whose DL(SAP)-role is BASIC, shall be included in the reply list; otherwise (field Q=0) they shall be excluded from the reply list.

11) If field R=1, then group DL-addresses of DLSAPs within the receiving DLE, or forwarded by the receiving DLE, shall be included in the reply list; otherwise (field R=0) they shall be excluded from the reply list.

c) Octets 4 to 7 shall encode a non-zero request-identifier provided by the requesting DLE.

NOTE This identifier enables the requesting DLE to associate a reply with its corresponding request.

d) Octets 8 and following shall encode either one begin/end-of-list element or one continuation-of-list element, as specified in 9.3.6.1.1 and 9.3.6.1.2. The encoded element shall specify to the receiving DLE where, within its forwarding database, the search should begin:
   1) If a begin/end-of-list element is specified, it indicates that the query is interrogating the entire forwarding database of the receiving DLE.
   2) If a continuation-of-list element is specified, it indicates that the query is interrogating the forwarding database of the receiving DLE continuing from the specified point. For this reason, the continuation-of-list element shall be identical to that returned by a prior address-list-reply SPDU received from the same correspondent DLE.

9.3.6.5 Address-list-reply SPDU

Table 130 – Address-list-reply SPDU

Octet index	Contents of field
1	0101 1VVV
2 3 4 5	Request identifier
6 ...	Parameter-list elements

The address-list-reply SPDU is sent in response to an address-list-query SPDU. Its primary purpose is to provide bridges with a means of building and maintaining their forwarding databases.

The address-list-reply SPDU may be addressed to a DLE on another link. It shall be formatted as specified in Table 130.

a) The SPDU header octet, octet 1, shall be encoded as 0101 1VVV, specifying an address-list-reply SPDU.
b) Octets 2 to 5 shall encode the non-zero request-identifier provided by the requesting DLE in the address-list-query SPDU to which this SPDU is a reply.

NOTE This identifier enables the querying DLE to associate the reply with its corresponding query.

c) Octets 6 and following shall encode zero or more SHORT DL-address and LONG DL-address elements, as specified in 9.3.6.1.3 and 9.3.6.1.4, specifying DL-addresses which meet the selection criteria specified in the corresponding address-list-query SPDU to which this is a reply.

   The last element in the parameter-list shall encode either one begin/end-of-list element or one continuation-of-list element, as specified in 9.3.6.1.1 and 9.3.6.1.2. If the encoded element is a begin/end-of-list element, it shall specify to the querying DLE that the requested search of the replying DLE's forwarding database is complete; if it is a continuation-of-list element, then the querying DLE can request that the search be continued at the point indicated in that continuation-of-list element.

The length of this SPDU may approach 256 octets.

9.4 Elements of Procedures for receiving SPDUs

9.4.1 Procedures for SPDUs received by the DLE acting as LAS

9.4.1.1 Receipt of a Link-basic-parameters-request SPDU

The LAS DLE shall

a) form a Link-basic-parameters-reply SPDU (9.3.2.10);
b) schedule the transmission of a connectionless DT DLPDU with
   – format 1S;
   – NORMAL priority;
   – a SHORT destination DL-address equal to the source DL-address of the DLPDU which conveyed the Link-basic-parameters-request SPDU;
   – a SHORT source DL-address of the local LAS, 0400_{16} (see 4.3.3.2);
   – a null SD-parameter field;
   – the required Link-basic-parameters-reply SPDU as the conveyed DLSDU.

9.4.1.2 Receipt of an FDC-DLE-has-“awakened” SPDU

The LAS DLE shall forward all those DLPDUs addressed to the notifying DLE which were stored by the LAS DLE while the notifying DLE had “gone-to-sleep”, and the LAS DLE shall stop storing any new received DLPDUs addressed to the notifying DLE.

9.4.1.3 Receipt of a FDC-DLE-may-“go-to-sleep”-notification SPDU

The LAS DLE shall send an FDC-DLE-may-“go-to-sleep”-acknowledge SPDU to the notifying DLE, and shall start storing any new received DLPDUs addressed to the notifying DLE for subsequent forwarding when that notifying DLE next notifies the LAS DLE that it has “awakened”.

9.4.1.4 Receipt of an Accept-LAS-role-reply SPDU

The LAS DLE shall note that the Accept-LAS-role-request SPDU has been received by the replying DLE, and shall wait for the Relinquish-LAS-role-request SPDU from that DLE.

9.4.1.5 Receipt of a Schedule-request SPDU

The LAS DLE shall attempt to update the schedule to satisfy this request. After completing the processing of this request, the LAS DLE shall send the Scheduling-completed SPDU as specified in 9.3.4.3.

9.4.1.6 Receipt of a Cancel-schedule SPDU

The LAS DLE shall attempt to update the schedule to satisfy this request. After completing the processing of this request, the LAS DLE shall send the Schedule-cancelled SPDU as specified in 9.3.4.5.

9.4.2 Procedures for SPDUs received by an LM DLE

9.4.2.1 Receipt of an Accept-LAS-role-request SPDU

If the addressed DLE accepts the request, it shall respond with an accept-LAS-role-reply SPDU, and shall take whatever actions are necessary to prepare for the transfer of the LAS role. When the addressed DLE is ready to accept the LAS role, it shall initiate the transfer by sending a relinquish-LAS-role-request SPDU (9.3.3.1) to the LAS DLE.

9.4.3 Procedures for SPDUs received by all DLEs

9.4.3.1 Receipt of a DL-conformance-query SPDU

The addressed DLE shall schedule the transmission of a connectionless DT DLPDU of NORMAL priority with the required DL-conformance-reply SPDU (9.3.2.8) as the conveyed DLSDU. The destination DL-address of this DLPDU shall be equal to the source DL-address of the DLPDU which conveyed the DL-conformance-query SPDU.

9.4.3.2 Receipt of a FDC-DLE-may-“go-to-sleep”-acknowledge SPDU

If the receiving DLE is an FDC DLE then it may become inattentive to all non-WK DLPDUs received on the local link.

10 Other DLE elements of procedure

Throughout Clause 10, whenever a conditional action is specified and the specified enabling condition does not occur, then the corresponding action also does not occur. The net effect is that events which do not meet any of the enabling conditions specified in a service procedure will have no consequential actions with respect to that specific service procedure.

10.1 DLE initialization

10.1.1 Hardware or host-system initialization

Upon power-up or after being reset, a DLE shall enter the OFFLINE DL-state, in which it is incapable of transmission on the local link, and shall issue a DLM-EVENT indication notifying DL-management of the event. The DLE shall remain in the OFFLINE state until instructed by DL-management to commence normal DL-operation.

DL-reception in the OFFLINE state is permitted but is not required.

The PhE reports its characteristics to the DLL during its own initialization, as specified in 4.4.1.1. If the reported framing-overhead is more than the DLE’s configured per-DLPDU-PhL-overhead, V(PhLO), then the DLE shall remain in the OFFLINE state and shall notify local DL-management of the problem.

10.1.2 Receipt of a DLM-ACTION request primitive

When the DLE receives a DLM-ACTION request, it shall associate the DLMS-user-specified request identifier with the request. If the request is accepted, as indicated by a returned status of “request accepted” for the DLM-ACTION request, then upon completion of the request, either successfully or after failure, the DLE shall issue a DLM-ACTION confirm with the same request identifier, conveying the status of the request to the DLMS-user.

10.1.2.1 Receipt of a DLM-ACTION request primitive requesting a change in DLE state

Figure 13 shows the state transitions between a DLE’s offline and online states.

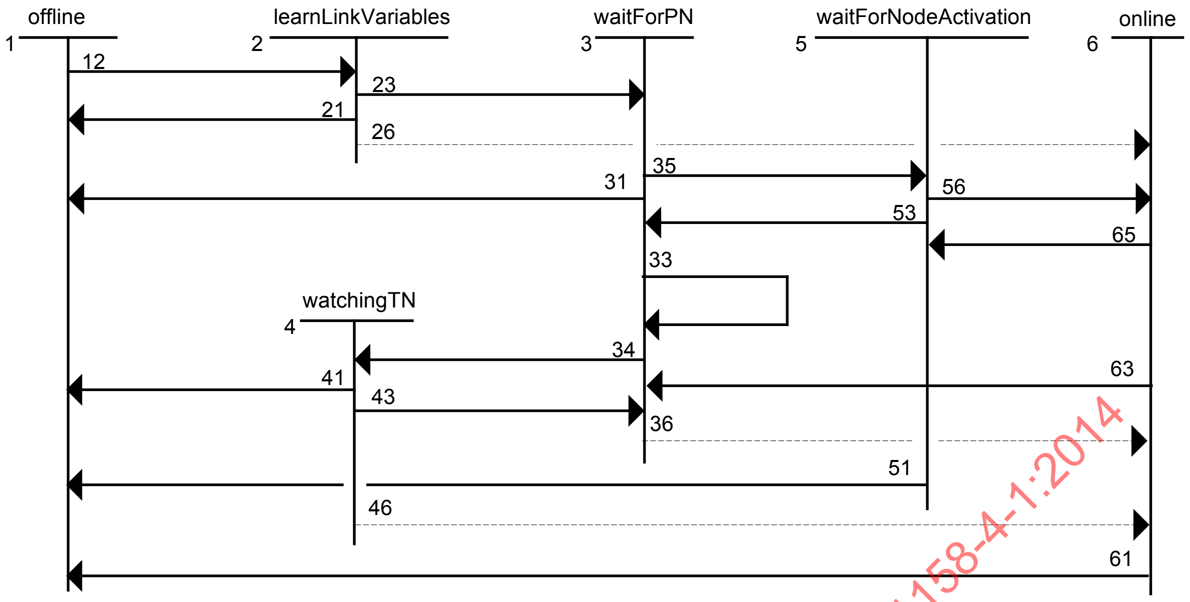


Figure 13 – State transitions of a DLE

If the DLM-ACTION request specifies that the DLE should change its DL-state to ONLINE, then the DLE shall initiate this process by changing its state to learnLinkVariables.

a) If the DLE's currently-assigned NODE DL-address is in the range $F8_{16}$..FF_{16} (see 4.3.2.2), then the DLE shall randomly choose, with an approximately uniform distribution, the low-order two bits of that NODE DL-address, adopt that modified address as its currently assigned address, and use the resulting address until a new NODE DL-address is assigned to (or by) the DLE. The actual random choice shall be statistically independent of similar choices made by other DLEs.

   NOTE This requirement for statistical independence minimizes the probability of repeatedly identical choices by identically constructed real devices, and thus provides the basis for eventual discrimination among multiple DLEs which happen to choose or respond to the same NODE DL-address, either through fault or misconfiguration.

b) If the DLE is capable of, and enabled for, link-master-class or bridge-class operation, and the DLE has knowledge of

   – the slot-time, V(ST);
   – the maximum-inactivity-to-claim-LAS-delay, V(MICD);
   – the Ph-parameters required for forming the PN DLPDU (see 6.13.2);
   – the DL-parameters of the local link enumerated in the node-activation SPDU (see 9.3.2.2],

   then the DLE shall enable the link-activity-monitoring and CL DLPDU transmission activities of 6.19.3, and if successful in claiming the scheduler token, shall

   – change its DL-state to ONLINE;
   – issue a DLM-EVENT indication notifying DL-management of the event.

c) The DLE shall enable the NODE DL-address monitoring activities of 6.13.4 and 6.15.4 and PR DLPDU transmission activities of 6.14.3:

   1) The DLE shall learn the DLL protocol version and some of the PhL and DLL configuration parameters of the local link from the PN-parameters of any received PN DLPDU and enter the waitForPN state.
   2) If the DLE receives a PT DLPDU addressed to its currently-assigned NODE DL-address, then the DLE shall monitor the apparent usage of its currently-assigned NODE DL-address (by another DLE on the local link) as follows:

i) If the currently-assigned NODE DL-address is in the range 10_{16}..$F7_{16}$ (see 4.3.2.2), then the DLE shall enter the watchingTN state, and if the DLE detects that its currently-assigned NODE DL-address is in use by another DLE (by detecting a subsequent immediate-response DLPDU, using the same procedure that the LAS uses to monitor the response to a transmitted PT DLPDU as specified in 6.15.3.2), then the DLE shall first change its NODE DL-address to $F8_{16}$ and then restart this procedure with step a);

ii) If the currently-assigned NODE DL-address is in the range $F8_{16}$..FF_{16} (see 4.3.2.2), then the DLE shall restart this procedure with step a);

NOTE Restarting this procedure with step a) does not cause a state change.

3) If the DLE receives a PN DLPDU addressed to its currently assigned NODE DL-address, and

– the DLE is not in the waitForNodeActivation state;

– the DLE supports the protocol version specified in that PN DLPDU;

– the DLE and its associated PhE are configured as specified in that PN DLPDU;

– the DLE is able to respond within the maximum-reply-delay, V(MRD), specified by the received PN DLPDU,

NOTE The DLE may have learned the link parameters from an earlier PN DLPDU, but it is necessary to check the parameters of the just received PN DLPDU, and reconfigure if necessary, because the parameters may have changed.

then

i) the DLE shall respond as in 6.13.4 and 6.14.3;

ii) if

– the DLE's required minimum-inter-DLPDU-delay is less than or equal to the value of V(MID) specified in the received PN DLPDU;

– the DLE's required maximum-response-delay is less than or equal to the value of V(MRD) specified in the received PN DLPDU,

NOTE The above two comparisons are made between the local link's configured value of V(MID) and V(MRD) as received in a PN DLPDU and the DLE's minimum requirements (that is, its device capabilities) for these two parameters. The DLE sends its minimum required values in a PR SPDU. Any DLPDU sent by any DLE, including any PR DLPDU, is required to use the link's configured value of V(MID) for delaying its transmission, as specified in 6)2)5).

then the DLE shall change its state to waitForNodeActivation;
iii) if ii) does not apply, then the DLE shall restart this procedure with step a).

4) If the DLE is in the waitForNodeActivation state, and

i) if a PN or PT DLPDU is received then the DLE shall restart this procedure with step a);

ii) if a DT DLPDU is received, and the DT DLPDU contains a node-activation SPDU (see 9.3.2.2), and the N(RID) field of that SPDU equals the value of V(RID) (see 4.7.1.18) which was used in the probe-response SPDU conveyed in the PR DLPDU last sent by this DLE, then the DLE shall

– change its configuration variables to reflect the contents of that received node-activation SPDU;

– change its DL-state to ONLINE;

– issue a DLM-EVENT indication notifying DL-management of the event.

iii) if

A) a DT DLPDU with two addresses is received, where the destination address of the DT DLPDU is the NODE DL-address currently assigned to the DLE;

B) the source address of the DT DLPDU is the default address for the local-link's LAS;

C) either the DT DLPDU does not contain a node-activation SPDU, or the N(RID) field of the received node-activation SPDU is not equal to the value of V(RID) (see 4.7.1.18) which was used in the probe-response SPDU conveyed in the PR DLPDU last sent by this DLE,

then the DLE shall infer that its currently assigned NODE DL-address is in use by another DLE, at which point it shall first change its NODE DL-address to $F8_{16}$ and then restart this procedure with step a).

If the DLM-ACTION request specifies that the DLE should change its DL-state to OFFLINE, then the DLE shall

– change its DL-state to OFFLINE;
– issue a DLM-EVENT indication notifying DL-management of the event.

10.1.2.2 Receipt of a DLM-ACTION request primitive requesting a change in link configuration parameters

If a DLE which is acting as the link's LAS receives a DLM-ACTION request which would change any of the link parameters conveyed by a Node-activation SPDU, a Link-basic-parameters-reply SPDU or a Link-master-parameters-reply SPDU, then the LAS DLE shall convey the changed parameter(s) to the other affected DLEs on the local link by sending the appropriate SPDU(s) to those other affected DLEs at the appropriate group DL-address(es).

If a DLE which is not acting as the link's LAS receives a DLM-ACTION request which would change any of the link parameters conveyed by a Node-activation SPDU, a Link-basic-parameters-reply SPDU or a Link-master-parameters-reply SPDU, then that non-LAS DLE shall not change its operational values for those variables, but shall update the values which it would use should it initialize the link after detecting the non-existence of any other DLE acting as LAS.

10.1.2.3 Receipt of a DLM-ACTION request primitive requesting assumption of the LAS role

If an LM DLE which is not acting as the link's LAS receives a request from local DL-management to assume the LAS role, then

– if the DLE does not have the necessary parameters required to operate as the LAS, then it shall follow the procedure of Clause 9 to obtain those parameters.

After receiving the necessary parameters to operate as LAS, the LM DLE shall

a) form a Relinquish-LAS-role-request SPDU;
b) schedule the transmission of a connectionless DT DLPDU with
   – format 1S;
   – NORMAL priority;
   – a SHORT destination DL-address for the local LAS, 0400_{16} (see 4.3.3.2);
   – a SHORT source DL-address equal to the NODE.0 DL-address, V(TN).0, of the sending DLE's DL-support functions;
   – a null SD-parameter field;
   – a Relinquish-LAS-role-request SPDU as the conveyed DLSDU;
c) wait for reception of a TL DLPDU;
d) if a TL DLPDU is not received within 15 receptions of a PT DLPDU with a token-use-subfield equal to RESTART, then the LM DLE shall retry this process up to a maximum of two times commencing with step a);

e) if a TL DLPDU is received, but is rejected as specified in 6.20.4.2, then the LM DLE shall assume that it does not have the necessary parameters required to operate as the LAS and shall repeat the procedures of Clause 9 up to a maximum of two times;

f) if a TL DLPDU is still not received after all retry attempts (as in d) or e)), then the LM DLE shall inform local DL-management of this failure via the corresponding DLM-ACTION confirm.

10.1.3 Receipt of a PN DLPDU while in the ONLINE state

If the DLE is not in any of the initialization states of Clause 9, and the DLE receives a PN DLPDU addressed to its currently assigned NODE DL-address, and the DLE supports the protocol version specified in that PN DLPDU, then

– the DLE and its associated PhE shall be configured as specified in that PN DLPDU;

– if the DLE is able to respond within the maximum-reply-delay, V(MRD), specified by the received PN DLPDU, then

  i) The DLE shall respond as in 6.13.4 and 6.14.3.

  ii) if

    – the DLE's required minimum-inter-DLPDU-delay is less than the value of V(MID) specified in the received PN DLPDU;

    – the DLE's maximum-response-delay is less than the value of V(MRD) specified in the received PN DLPDU,

  the DLE shall change its state to waitForNodeActivation and behave as specified in Clause 9.

  NOTE The DLE should not disconnect any DLCEPs or do any other initialization and continue to operate normally after it changes its state from waitForNodeActivation to ONLINE.

  iii) If

    – the DLE does not support the protocol version specified in the received PN DLPDU,

    – the DLE is not able to respond within the maximum-reply-delay, V(MRD), specified by the received PN DLPDU, or

    – ii) does not apply,

  then the DLE shall

    – issue a DL-Disconnect indication to all non-idle DLCEPs;

    – initialize the state of all DLCEPs to Idle;

    – reset all queues to empty;

    – issue a confirm primitive with an appropriate status for any outstanding DLS-user request;

    – perform any other initialization necessary to transition the DLE to the OFFLINE state;

    – start with the procedure of 10.1.2.1b).

10.2 LAS behavior and operation

10.2.1 LAS operation when holding a scheduler token

The LAS DLE shall initiate each scheduled transaction at its scheduled time within the jitter limits permitted by the schedule for that transaction.

The LAS DLE shall repetitively use the following procedure to determine the next (unscheduled) transaction, subject to the constraint that the transaction can be completed before the next scheduled activity.

a) If the LAS DLE is required to send a TD DLPDU, then it shall send a TD DLPDU.

b) If the LAS DLE has not completed the current cycle of “circulating the token”, as indicated by the fact that

– the next DLE to receive the PT DLPDU is not the lowest-numbered NODE DL-address represented in V(LL), or

– the value of the final-use subfield of that next PT DLPDU will be CONTINUE,

then

1) The LAS DLE may use the available bus capacity for link maintenance as specified in Clause 9.

2) If the LAS DLE is required to send an ES DLPDU as specified in Clause 9, then it may use the available bus capacity for sending that ES DLPDU as specified in 6.16.3.2, provided this use is limited to sending one ES DLPDU to the DL-address specified in the ES DLPDU per cycle of “circulating the token”.

3) If the available link capacity permits, then the LAS DLE shall send a PT DLPDU as specified in 6.15.3.2;

4) Otherwise, when 3) does not apply and the available link capacity is not sufficient to send a PT DLPDU, the LAS DLE shall repetitively use the following criteria to determine the next DLPDU to be transmitted:

i) If a node-activation SPDU is scheduled for transmission, and there is sufficient available link capacity to send the SPDU in a DT DLPDU as specified in 6.7, then the LAS DLE shall transmit the SPDU within the envelope of a DT DLPDU addressed the intended recipient.

ii) If i) does not apply, and there is sufficient available link capacity to probe a new node address as specified in 6.13, then the LAS DLE shall perform the procedure of 6.13.3.

iii) If neither i) nor ii) applies, then the LAS DLE may transmit any other DLPDU

– from the generic source address for the local LAS, 0400_{16} (see 4.3.3.2);

– which is already queued for transmission,

provided that there is sufficient available link capacity to complete the transaction.

iv) If none of i) to iii) applies, and the duration until the next scheduled transaction exceeds token-recovery-delay slot-times, P(TRD) × V(ST) octet-durations, then the LAS DLE shall send one or more IDLE DLPDUs of appropriate length to keep the local link from experiencing an excessive period of inactivity.

Any such link capacity which is not adequate for the procedure of 6.15.3 need not be attributed to the duration of token usage allocated for the LAS DLE’s link-maintenance activities.

c) Otherwise, when b) does not apply, so that the LAS DLE has just completed a cycle of “circulating the token”, then

1) If the LAS DLE has a need to transfer its role to another DLE, then

i) the LAS DLE shall not start the next cycle of “circulating the token”; but instead

ii) the LAS DLE shall send all DLPDUs which it has already scheduled for transmission from the generic source address for the local LAS, 0400_{16} (see 4.3.3.2), after which

iii) the LAS DLE shall send a TL DLPDU as specified in 6.20.3.

2) Otherwise, when 1) does not apply, then the LAS DLE shall send all link-maintenance DLPDUs (see 4.7.5.9) which are queued for transmission, if necessary over an interval of one or more cycles of “circulating the token”, so that the link capacity used for link maintenance during one cycle of “circulating the token” does not exceed the configured link-maintenance-token-holding-time, V(LTHT).

The LAS DLE shall repetitively use the following criteria to determine the next transaction.

i) If there is a need to send a node-activation SPDU, then the DLE shall send such an SPDU (in a DT DLPDU).

ii) If i) does not apply, and the LAS DLE has not sent any PN DLPDUs during the just completed cycle of "circulating the token", then the LAS DLE shall send a PN DLPDU to the next NODE DL-address to be probed, as specified in 6.13.3.

iii) If neither i) nor ii) applies, then the LAS DLE may send any DLPDU, other than a TD, PN, PT or ES DLPDU, which it needs to send in its role as LAS.

After completing any such permitted transactions, the LAS DLE shall start a new cycle of "circulating the token".

NOTE The LAS DLE will perform as many transactions as possible, as specified by a), b) and c), provided that the transactions can be completed before the next scheduled activity.

10.2.2 Return of a delegated token; assumption of a scheduler token

When a delegated token which was delegated by a PT DLPDU is returned by an RI DLPDU, then the LAS DLE shall associate the value of the DD-parameter of that received RI DLPDU with the DLE returning the token, and use it as the minimum token delegation time for sending the next PT DLPDU to that DLE as required in 6.15.3.1d)3).

When a delegated token which was delegated by an ES DLPDU is returned by an RI DLPDU, then the LAS DLE shall check whether the delegating ES DLPDU was the last delegation of a scheduled sequence, as indicated by the encoding specified in 9.3.5.2.2a):

a) If it was not the last delegation within the current cycle of that scheduled sequence execution, then the LAS DLE shall ignore the request for an additional delegation which is requested by the just-received RI DLPDU;
b) Otherwise, when it was the last delegation of a scheduled sequence, then the DLE may, but need not,
   1) associate the value of the DD-parameter of that received RI DLPDU with the destination DL-address of that ES DLPDU;
   2) note the need to send another ES DLPDU with a final token use subfield equal to CONTINUE to the requesting DL-address, during an interval of otherwise-unscheduled bus capacity.

When the LAS DLE assumes a scheduler token due to link inactivity, on expiration of a link monitoring interval, then the LAS DLE shall initiate transmission of its next DLPDU such that there is no more than 14 slot-times, 14 × V(ST) octet-durations, of inactivity on the local link.

The LAS DLE may measure the duration for which the token has been delegated, and if that duration expires before the token is returned, then the LAS DLE may assume the scheduler token immediately upon that expiration.

If the delegated token has not been returned, and it is time for the next scheduled activity, then the LAS DLE may assume the scheduler token immediately.

10.2.3 Receipt of a probe-response (PR) SPDU

If one or more PR DLPDUs are received in response to a PN DLPDU sent by the LAS DLE, then the LAS DLE shall process the first received PR DLPDU (see 9.3.2.1) as specified in 10.2, and shall discard all of the other PR DLPDUs received during that response window.

The LAS DLE shall compare the values specified in the minimum-inter-DLPDU-delay, V(MID), and maximum-response-delay-in-octets, V(MRD) × V(ST), fields of a received PR SPDU with the same parameters as configured for the local link.

If

– the responding DLE's V(MID) is less than or equal to the link's configured value for V(MID);

– the responding DLE's V(MRD) is less than or equal to the link's configured value for V(MRD),

then

a) the LAS DLE shall form a node-activation SPDU with its V(RID) field equal to the V(RID) field of the received PR SPDU;

b) the LAS DLE shall note the need to send a node-activation SPDU to the responding DLE, where that node-activation SPDU is conveyed in a connectionless DT DLPDU with

   – format 1S;

   – NORMAL priority;

   – a SHORT destination DL-address equal to the NODE.0 DL-address of the responding DLE's DL-support functions;

   – a SHORT source DL-address of the local LAS, 0400_{16} (see 4.3.3.2);

   – a null SD-parameter field;

   – the required node-activation SPDU as the conveyed DLSDU.

c) after sending this DT DLPDU, the LAS DLE shall include the NODE DL-address of the responding DLE in its local-link live-list, V(LL).

d) If the token-circulation-not-needed field, N, of the responding PR SPDU was set to 0 (no, token circulation is needed), then the LAS DLE shall include the NODE DL-address of the responding DLE in its local-link-token-circulation-list, V(TCL).

e) The LAS DLE shall publish the live-list change in one of the following two ways:

   1) the LAS DLE shall

      i) increment the live-list revision-number;

      ii) form a live-list-change SPDU (see 9.3.2.4) indicating the NODE DL-address of the affected DLE and its status;

      iii) schedule the transmission of a connectionless DT DLPDU with

         – format 1S,

         – NORMAL priority,

         – a SHORT destination DL-address equal to the group DL-address for the DL-support functions of all LM DLE's on the local link, 0101_{16} (see 4.3.3.2),

         – a SHORT source DL-address of the local LAS, 0400_{16} (see 4.3.3.2),

         – a null SD-parameter field,

         – the required live-list-change SPDU as the conveyed DLSDU; or

   2) the LAS DLE shall append the NODE DL-address of the affected DLE and its status to an existing DT DLPDU conveying a live-list-change SPDU, formed as in 1) and 2), which has previously been scheduled for transmission but has not yet been sent.

f) If the PR DLPDU specified a DLE class of link master or bridge for the responding DLE, the LAS may consider that information in determining whether to request the new DLE to assume the LAS role after synchronizing with the existing schedule (for example, because the current LAS is a link master DLE and the responding device is a bridge DLE).

Otherwise, when either

– the responding DLE's V(MID) is greater than the link's configured value for V(MID), or

– the responding DLE's V(MRD) is greater than the link's configured value for V(MRD),

then the DLE shall notify its local DL-management.

10.2.4 Lack of response to a PT DLPDU

If a DLE does not send any DLPDU in response to a PT DLPDU addressed to the DLE, as specified in 6.15.4.1, then the LAS DLE shall note the event. After three such non-responses for any specific NODE DL-address within three consecutive cycles of “circulating the token”, the LAS DLE shall

a) remove that NODE DL-address from the LAS DLE’s local-link token-circulation-list, V(TCL), and local-link live-list, V(LL);
b) publish the live-list change as specified in Clause 9.

10.2.5 Receipt of a live-list-request SPDU

The LAS shall

a) form a live-list-detail SPDU (see Clause 9);
b) schedule the transmission of a connectionless DT DLPDU with
   – format 1S;
   – NORMAL priority;
   – a SHORT destination DL-address equal to the source DL-address of the DLPDU which conveyed the live-list-request SPDU, or equal to the group DL-address for the DL-support functions of all LM DLE’s on the local link, 0101_{16} (see 4.3.3.2);
   – a SHORT source DL-address of the local LAS, 0400_{16} (see 4.3.3.2);
   – a null SD-parameter field;
   – the required live-list-detail SPDU as the conveyed DLSDU.

10.2.6 Receipt of a relinquish-LAS-role-request SPDU

If the LAS had already noted the need to transfer the LAS role, then the LAS shall ignore the received SPDU, else the LAS shall note the need to transfer the LAS role.

10.2.7 Other link-maintenance requirements

Whenever there is a change in the LAS DLE’s schedule, or whenever an LM DLE other than the most recent LAS DLE assumes the scheduler role, or periodically such that the time between two successive distributions of the LAS database by means of LAS-data-base-status SPDUs is equal to or more than the LAS-data-base-distribution-period, V(LDDP) (see 4.7.5.20) and less than 1,5 × V(LDDP), then the LAS shall

a) form an LAS-database-status SPDU (see Clause 9);
b) schedule the transmission of a connectionless DT DLPDU with
   – format 1S;
   – NORMAL priority;
   – a SHORT destination DL-address equal to the group DL-address for the DL-support functions of all LM DLE’s on the local link, 0101_{16} (see 4.3.3.2);
   – a SHORT source DL-address of the local LAS, 0400_{16} (see 4.3.3.2);
   – a null SD-parameter field;
   – the required LAS-database-status SPDU as the conveyed DLSDU.

10.2.8 Receipt of a link-master-parameters-request SPDU

The LAS shall

a) form a link-master-parameters-reply SPDU (see 9.3.2.12);
b) schedule the transmission of a connectionless DT DLPDU with

– format 1S;

– NORMAL priority;

– a SHORT destination DL-address equal to the source DL-address of the DLPDU which conveyed the link-master-parameters-request SPDU, or equal to the group DL-address for the DL-support functions of all LM DLE's on the local link, 0101_{16} (see 4.3.3.2);

– a SHORT source DL-address of the local LAS, 0400_{16} (see 4.3.3.2);

– a null SD-parameter field;

– the required link-master-parameters-reply SPDU as the conveyed DLSDU.

10.2.9 Receipt of a token-hold-time-request SPDU

If the starting NODE DL-address field of the received SPDU is 00, then the LAS DLE shall

a) form as many token-hold-time-array SPDUs (see 9.3.2.14) as necessary to send the maximum-token-hold-time for all DLEs represented in the local-link-live-list, V(LL);

b) form one token-hold-time-array SPDU with a starting NODE DL-address field with a value of zero;

c) for each token-hold-time-array SPDU, schedule the transmission of a connectionless DT DLPDU with

   – format 1S;

   – NORMAL priority;

   – a SHORT destination DL-address equal to the source DL-address of the DLPDU which conveyed the token-hold-time-request SPDU, or equal to the group DL-address for the DL-support functions of all LM DLE's on the local link, 0101_{16} (see 4.3.3.2);

   – a SHORT source DL-address of the local LAS, 0400_{16} (see 4.3.3.2);

   – a null SD-parameter field;

   – the required token-hold-time-array SPDU as the conveyed DLSDU;

   such that the token-hold-time-array SPDU with a starting NODE DL-address field with a value of zero is sent after all of the other token-hold-time-array SPDUs have been sent.

Otherwise, when the starting NODE DL-address field of the received SPDU is $N0_{16}$, where N = 1 to F_{16}, then the LAS DLE shall

d) form one token-hold-time-array SPDU with a starting NODE DL-address field equal to the starting NODE DL-address field of the received SPDU;

e) schedule the transmission of a connectionless DT DLPDU with

   – format 1S;

   – NORMAL priority;

   – a SHORT destination DL-address equal to the source DL-address of the DLPDU which conveyed the token-hold-time-request SPDU, or equal to the group DL-address for the DL-support functions of all LM DLE's on the local link, 0101_{16} (see 4.3.3.2);

   – a SHORT source DL-address of the local LAS, 0400_{16} (see 4.3.3.2);

   – a null SD-parameter field;

   – the required token-hold-time-array SPDU as the conveyed DLSDU.

Else, the received SPDU is invalid and shall be ignored.

10.2.10 Receipt of a schedule-summary-request SPDU

The LAS shall

a) form a schedule-summary SPDU (see 9.3.5.1);

b) schedule the transmission of a connectionless DT DLPDU with

- format 1S;
- TIME-AVAILABLE priority;
- a SHORT destination DL-address equal to the source DL-address of the DLPDU which conveyed the schedule-summary-request SPDU, or equal to the group DL-address for the DL-support functions of all LM DLE's on the local link, 0101_{16} (see 4.3.3.2);
- a SHORT source DL-address of the local LAS, 0400_{16} (see 4.3.3.2);
- a null SD-parameter field;
- the required schedule-summary SPDU as the conveyed DLSDU.

10.2.11 Receipt of a subschedule-request SPDU

If the schedule version-number field of the received SPDU is equal to the schedule version-number field of the currently active schedule, then

a) If the subschedule identifier field of the received SPDU is 00, then the LAS DLE shall
   1) form as many subschedule SPDUs (see 9.3.5.2) as necessary to send all of the subschedules of the currently active schedule;
   2) form one subschedule SPDU with a subschedule identifier field with a value of zero;
   3) for each subschedule SPDU, schedule the transmission of a connectionless DT DLPDU with
      - format 1S;
      - TIME-AVAILABLE priority;
      - a SHORT destination DL-address equal to source DL-address of the DLPDU which conveyed the subschedule-request SPDU, or equal to the group DL-address for the DL-support functions of all LM DLE's on the local link, 0101_{16} (see 4.3.3.2);
      - a SHORT source DL-address of the local LAS, 0400_{16} (see 4.3.3.2);
      - a null SD-parameter field;
      - the required subschedule SPDU as the conveyed DLSDU;

   such that the subschedule SPDU with subschedule identifier field equal zero is sent after all other subschedule SPDUs are sent.

b) Otherwise, when the subschedule identifier field of the received SPDU is non-zero and equal to the subschedule identifier of one of the subschedules of the currently active schedule, then the LAS DLE shall
   1) form one subschedule SPDU with a subschedule identifier field whose value is equal to the value of the subschedule identifier field of the received SPDU;
   2) schedule the transmission of a connectionless DT DLPDU with
      - format 1S;
      - TIME-AVAILABLE priority;
      - a SHORT destination DL-address equal to source DL-address of the DLPDU which conveyed the subschedule-request SPDU, or equal to the group DL-address for the DL-support functions of all LM DLE's on the local link, 0101_{16} (see 4.3.3.2);
      - a SHORT source DL-address of the local LAS, 0400_{16} (see 4.3.3.2);
      - a null SD-parameter field;
      - the required subschedule SPDU as the conveyed DLSDU.

c) Otherwise, when the subschedule identifier field of the received SPDU is non-zero and is not equal to the subschedule identifier of any of the subschedules of the currently active schedule, then the LAS DLE shall
   1) form a schedule-summary SPDU (see 9.3.5.1);
   2) schedule the transmission of a connectionless DT DLPDU with

– format 1S;
– TIME-AVAILABLE priority;
– a SHORT destination DL-address equal to the source DL-address of the DLPDU which conveyed the subschedule-request SPDU, or equal to the group DL-address for the DL-support functions of all LM DLE's on the local link, 0101_{16} (see 4.3.3.2);
– a SHORT source DL-address of the local LAS, 0400_{16} (see 4.3.3.2);
– a null SD-parameter field;
– the schedule-summary SPDU as the conveyed DLSDU.

If the schedule version-number field of the received SPDU is not equal to the schedule version-number field of the currently active schedule, then the LAS DLE shall send a schedule-summary SPDU as in c).

10.3 DL-support operation

A DLE's DL-support functions receive each DT DLPDU with the destination DL-address of NODE.0, V(TN).0 (see 6.7.4.1.1d)), addressed to that DLE. When the DLE receives such a DLPDU, the DLS-user data included in the received DT DLPDU shall be interpreted as an SPDU as specified in Clause 11:

a) If the received SPDU is a DL-address query SPDU, then the DLE shall process it as specified in 8.2.3.3 or 8.3.4.3 depending upon the DL-address in the SPDU.
b) If the received SPDU is a DL-address-report SPDU, then the DLE shall process it as specified in 8.2.3.4 or 8.3.4.4 depending upon the DL-address in the SPDU.
c) If the received SPDU is a schedule-cancelled SPDU, then the DLE shall process it as specified in 8.4.3.3 or 8.4.3.4.
d) If the source address of the received DT DLPDU is the default address for the local-link's LAS, 4.0, and if the received SPDU is a node-activation SPDU, then the DLE shall process it as specified in Clause 9.
e) If the received SPDU is an LAS-database-status SPDU, then the DLE shall process it as specified in Clause 9.
f) If the received SPDU is a live-list-change SPDU, then the DLE shall process it as specified in Clause 9.
g) If the received SPDU is a live-list-detail SPDU, then the DLE shall process it as specified in Clause 9.
h) If the received SPDU is a live-list-request SPDU, then the DLE shall process it as specified in Clause 9.
j) If the received SPDU is a relinquish-LAS-role SPDU, then the DLE shall process it as specified in Clause 9.
k) If the received SPDU is a link-master-parameters-reply SPDU, then the DLE shall process it as specified in Clause 9.
m) If the received SPDU is a token-hold-time-array SPDU, then the DLE shall process it as specified in Clause 9.
n) If the received SPDU is a link-master-parameters-request SPDU, then the DLE shall process it as specified in Clause 9.
p) If the received SPDU is a token-hold-time-request SPDU, then the DLE shall process it as specified in Clause 9.
q) If the received SPDU is a schedule-summary-request SPDU, then the DLE shall process it as specified in Clause 9.
r) If the received SPDU is a subschedule-request SPDU, then the DLE shall process it as specified in Clause 9.
s) If the received SPDU is a schedule-summary SPDU, then the DLE shall process it as specified in Clause 9.

t) If the received SPDU is a subschedule SPDU, then the DLE shall process it as specified in Clause 9.

u) If the received SPDU is a scheduling-completed SPDU, then the DLE shall process it as specified in 8.4.3.3.

v) If the received SPDU is any other SPDU, then the DLE shall process it as specified in Clause 9.

10.3.1 Receipt of an LAS-database-status SPDU by an LM DLE

The receiving LM DLE shall compare the live-list revision-number specified in the received SPDU with the revision-number of its own local copy of the live-list, V(LL). If the two revision-numbers differ, then the LM DLE shall

a) form a live-list-request SPDU (see 9.3.2.5);

b) schedule the transmission of a connectionless DT DLPDU with
   - format 1S;
   - NORMAL priority;
   - a SHORT destination DL-address for the local LAS, 0400_{16} (see 4.3.3.2);
   - a SHORT source DL-address equal to the NODE.0 DL-address, V(TN).0, of the sending DLE's DL-support functions;
   - a null SD-parameter field;
   - a live-list-request SPDU as the conveyed DLSDU.

   A reference to the DLE's NODE user-request queue, Q_N(UR), shall be appended to the LM DLE's unscheduled-service queue, Q(US).

The receiving LM DLE shall compare the schedule version-number specified in the received SPDU with the schedule version-number of its own local copy of the link schedule. If the two schedule version-numbers differ, and if the schedule-type field, T, in the received SPDU indicates that the LAS DLE is capable of transferring its schedule to other LM DLEs, then the LM DLE shall

c) form a schedule-summary-request SPDU;

d) schedule the transmission of a connectionless DT DLPDU with
   - format 1S;
   - NORMAL priority;
   - a SHORT destination DL-address for the local LAS, 0400_{16} (see 4.3.3.2);
   - a SHORT source DL-address equal to the NODE.0 DL-address, V(TN).0, of the sending DLE's DL-support functions;
   - a null SD-parameter field;
   - a schedule-summary-request SPDU as the conveyed DLSDU.

   A reference to the DLE's NODE user-request queue, Q_N(UR), shall be appended to the LM DLE's unscheduled-service queue, Q(US).

10.3.2 Receipt of a live-list-change SPDU by an LM DLE

If a received live-list-change SPDU (see 9.3.2.4) indicates a change in the live-list, then the receiving LM DLE shall update its own local version of the live-list in the manner indicated by the received SPDU, and shall increment its live-list revision-number.

10.3.3 Receipt of a live-list-detail SPDU by an LM DLE

The receiving LM DLE shall update

- its own local version of the local-link-live-list, V(LL);

– the live-list revision-number;
– the LAS DLE's NODE DL-address, V(LN);
– the expected-non-response-list, V(ENRL);
– the first-unpolled-node's NODE DL-address , V(FUN);
– the number-of-consecutive-unpolled-nodes, V(NUN)

as indicated by the received SPDU.

10.3.4 Request for LAS parameters by an LM DLE

A link master requires the following parameters to operate as the LAS:

a) the local link's configuration parameters defined in 4.7.5 and included in the link-master-parameters-reply SPDU (see 9.3.2.11);
b) the local-link-live-list, V(LL);
c) the expected-non-response-list, V(ENRL);
d) the token-circulation-list, V(TCL);
e) the maximum-token-holding-time-array, V(MTHA);
f) the local link's schedule;
g) a sense of the current LAS DLE's DL-time.

If a link master DLE does not have the necessary parameters required to operate as the LAS, as may be the case when the link master DLE has just changed its state to ONLINE, then the link master DLE shall schedule the transmission of the following SPDUs to the local LAS in connectionless DT DLPDUs each with format 1S, NORMAL priority, a destination address of the local LAS, 0400_{16} (see 4.3.3.2), and a source address of V(TN).00:

– a link-master-parameters-request SPDU (see 9.3.2.11);
– a token-hold-time-request SPDU (see 9.3.2.13);
– a live-list-request SPDU (see 9.3.2.5);
– a schedule-summary-request SPDU (see 9.3.5.3);

the link-master DLE shall note the need to transmit a CT DLPDU at its first opportunity.

After sending the SPDUs, the sending DLE shall wait for reception of the link-master-parameters-reply SPDU, token-hold-time-array SPDUs, live-list-detail SPDU and schedule-summary SPDU;

If these SPDUs are not received by the requesting LM DLE within 15 receptions of a PT DLPDU with a token-use-subfield equal to RESTART addressed to the DLE's NODE DL-address, then the requesting LM DLE shall again schedule the transmission of those request SPDUs appropriate for the missing reply SPDUs.

10.3.5 Receipt of a link-master-parameter-reply SPDU by an LM DLE

The DLE shall use the received SPDU to update the parameters required to operate as LAS and

– if the DLE's maximum-inactivity-to-claim-LAS-delay is less than or equal to the link's configured value of maximum-inactivity-to-claim-LAS-delay, V(MICD), then the DLE shall note that it is capable of operating as the local link's LAS.

10.3.6 Receipt of a token-hold-time-array SPDU by an LM DLE

If the starting NODE DL-address field of the received SPDU (see 9.3.2.14) is $N0_{16}$, where N = 1 to F_{16} then the DLE shall

– update the maximum-token-holding-time-array, V(MTHA) from the maximum-token-hold-time values received in the SPDU;

– exclude from the token-circulation-list, V(TCL), those DLEs for which the maximum-token-hold-time is zero.

Otherwise, when the starting NODE DL-address field of the received SPDU is 00, then the DLE shall

– check whether it has received the token-hold-time-array SPDUs for all NODE DL-addresses in the local-link-live-list, V(LL);

– if not, then send one or more token-hold-time-request SPDUs, specifying such values for the starting NODE DL-addresses in those SPDUs, that the corresponding reply SPDUs will contain the maximum-token-hold-times for all the NODE DL-addresses in local-link-live-list, V(LL), whose maximum-token-hold-times are not yet known by the requesting DLE;

Otherwise, when the starting NODE DL-address field of the received SPDU is of the form NM_{16}. where M is non-zero, then the received SPDU is invalid and shall be discarded.

10.3.7 Receipt of a schedule-summary SPDU by an LM DLE

If the schedule version-number field of the received SPDU (see 9.3.5.1) is non-zero, then

a) If either the DLE has no link schedule, or the schedule version-number of the link schedule stored in the DLE is different from the schedule version-number field of the just received SPDU, then the DLE shall

   1) check that it is capable of executing the link schedule whose summary was just received. The DLE shall

      i) check that it has the capability to execute the number of subschedules in the schedule-summary;

      ii) check that the DLE's value of maximum-scheduling-overhead, V(MSO), (see 4.7.5.6 is less than or equal to the value of V(MSO) specified in the schedule-summary;

      iii) check that the DLE has the required storage capacity specified in the schedule-summary;

      iv) check that the DLE has the required timing resolution specified in the schedule-summary;

      v) check that the DLE's value of time-source-link, V(TSL), (see 4.7.1.27) is the same as the value of V(TSL) specified in the schedule-summary.

      If any of these checks fail, then the DLE shall discard the just received schedule-summary SPDU.

   2) otherwise, when these checks are all passed, then the DLE shall

      i) form a subschedule-request SPDU with its schedule version-number field equal to the value of the schedule version-number field of the just received schedule-summary SPDU, and with a subschedule identifier equal to zero;

      ii) schedule the transmission of a connectionless DT DLPDU with

         – format 1S;

         – NORMAL priority;

         – a SHORT destination DL-address for the local LAS, 0400_{16} (see 4.3.3.2);

         – a SHORT source DL-address equal to the NODE.0 DL-address, V(TN).0, of the sending DLE's DL-support functions;

         – a null SD-parameter field;

         – the just-formed subschedule-request SPDU as the conveyed DLSDU.

b) Otherwise, when the DLE has a link schedule and the version-number of the link schedule stored in the DLE is equal to the schedule version-number field of the just received SPDU, but the DLE does not have the subschedule for all of the subschedules identified in the just received schedule-summary SPDU, then the DLE shall

   1) form one or more subschedule-request SPDUs, each with a schedule version-number field equal to the value of the schedule version-number field of the just received SPDU, and with a subschedule identifier equal to one of the subschedule-SPDU references of the missing subschedules;

   2) schedule the transmission of a connectionless DT DLPDU for each such just-formed SPDU, with

      – format 1S;
      – NORMAL priority;
      – a SHORT destination DL-address for the local LAS, 0400_{16} (see 4.3.3.2);
      – a SHORT source DL-address equal to the NODE.0 DL-address, V(TN).0, of the sending DLE's DL-support functions;
      – a null SD-parameter field;
      – one of the just-formed subschedule-request SPDUs as the conveyed DLSDU.

Otherwise, when the schedule version-number field of the just received SPDU is zero, then the DLE shall record that there is no active link schedule.

NOTE Where resources permit, an LM DLE which is incrementally updating its copy of the local link's schedule should retain the last self-consistent prior version of that schedule until such updating is complete.

10.3.8 Receipt of a subschedule SPDU by an LM DLE

If the schedule version-number field of the just received subschedule SPDU (see 9.3.5.2) is equal to the schedule version-number field of the last received schedule-summary SPDU (see 9.3.5.1), then

a) if the subschedule identifier field of the received SPDU is equal to a subschedule-SPDU reference included in the last received schedule-summary SPDU, then the DLE shall store the subschedule as part of the link schedule;

b) otherwise, when the subschedule identifier field of the received SPDU is equal to zero, then the DLE shall

   – check whether it has received the subschedule SPDUs for all of the subschedule-SPDU references in the last-received schedule-summary SPDU;
   – if not, then send one or more subschedule-request SPDUs, each with a value of the subschedule identifier requesting one of the missing subschedules;

c) otherwise, when neither a) nor b) applies, then the DLE has received a subschedule SPDU inconsistent with the DLE's last-received schedule-summary SPDU. Therefore the DLE shall

   1) form a schedule-summary-request SPDU;

   2) schedule the transmission of a connectionless DT DLPDU with

      – format 1S;
      – NORMAL priority;
      – a SHORT destination DL-address for the local LAS, 0400_{16} (see 4.3.3.2);
      – a SHORT source DL-address equal to the NODE.0 DL-address, V(TN).0, of the sending DLE's DL-support functions;
      – a NULL SD-PARAMETER FIELD;
      – A SCHEDULE-summary-request SPDU as the conveyed DLSDU.

Otherwise, when the schedule version-number field of the just received subschedule SPDU is not equal to the schedule version-number field of the last received schedule-summary SPDU, or if the DLE had never received a schedule-summary SPDU, then the DLE shall send a schedule-summary-request SPDU as in c).

NOTE See Note at the end of 10.3.7.

10.3.9 Request for LAS transfer by an LM DLE

If an LM DLE receives a request from local DL-management to assume the LAS role, then

– if the DLE does not have the necessary parameters required to operate as the LAS, then it shall follow the procedure of 10.3.4 to obtain those parameters.

After receiving the necessary parameters to operate as LAS, the LM DLE shall

a) form a Relinquish-LAS-role SPDU;
b) schedule the transmission of a connectionless DT DLPDU with
   – format 1S;
   – NORMAL priority;
   – a SHORT destination DL-address for the local LAS, 0400_{16} (see 4.3.3.2);
   – a SHORT source DL-address equal to the NODE.0 DL-address, V(TN).0, of the sending DLE's DL-support functions;
   – a null SD-parameter field;
   – a schedule-summary-request SPDU as the conveyed DLSDU;
c) wait for reception of a TL DLPDU;
d) if a TL DLPDU is not received within 15 receptions of a PT DLPDU with a token-use-subfield equal to RESTART, then the LM DLE shall retry this process commencing with step a);
e) if a TL DLPDU is received, but is rejected as specified in 6.20.4.2, then the LM DLE shall assume that it does not have the necessary parameters required to operate as the LAS and shall repeat the procedures of 10.3.9;
f) if a TL DLPDU is still not received after two retry attempts (as in d) or e)), then the LM DLE shall inform local DL-management of this failure.

10.4 DL-bridge elements of procedure and bridge sub-protocol

Two compatible versions of a bridge sub-protocol are defined. The first, based on ISO/IEC 10038, supports adaptive bridges which require almost no configuration, but requires all the nodes on the local link implement the full DL-support protocol of 10.3.

The second version of the bridge sub-protocol, based on a subset of ISO/IEC 10038, supports non-adaptive bridges which require extensive configuration, but eliminates the need for all the nodes on the local link to implement the full DL-support protocol of 10.3.

10.4.1 Common features of the Type 1 bridge protocols and elements of procedure

10.4.1.1 General

Bridges provide the following capabilities:

a) Selective forwarding of DLPDUs.
b) Selective subscription to, and republishing of, published DLSDUs.
c) Selective propagation of DL-time based on port state.

Bridges conforming to 10.4.1.5 also

d) Exchange information to self-configure into a spanning tree.

e) Observe traffic and query DLEs on the local links to self-configure their forwarding databases.

Bridges may conform in an operational sense to both this 10.4.1.5 and 10.4.3; the local method by which they reconcile conflicting management paradigms is not specified.

10.4.1.2 Overview

Local links of all data rates may be connected together with bridges, which are DL-relay entities. Each individual link has its own independent schedule and operation. The extended link created by the bridges allows the interconnection of end systems attached to separate links as if they were attached to a single unified link. However, time and event synchronization among end systems on the extended link is weaker than on any single constituent link.

Local links of different speeds and media types may be bridged together to form multi-link, or *bridged*, networks. Bridges are used in these networks to interconnect links. In all fieldbusbridged networks, at any moment there is only one operational DLPDU forwarding path between any two DLEs. To ensure this, routing tables within bridges are coordinated with each other to form a *spanning tree*.

The spanning tree represents a configuration of bridges, such that there are only two directions of data flow, either towards the root of the tree, or away from it. There are no loops and no parallel routes. That is, for each pair of links there must be one and only one active bridge leading to the root.

Each bridge in the spanning tree has a single root port and one or more downstream ports. Each bridge port represents a connection to one link. Root ports lead toward (or may themselves be) the root of the spanning tree; downstream ports lead away from the root.

When a message is received at a port, the bridge uses local routing tables to determine whether it is in the path connecting the link from which the message was received and one or more of the links to which the message is to be sent.

When a message is received on a downstream port, the bridge routes it to the appropriate other port(s) based on routing tables within the bridge. Figures 14 and 15 below illustrate a bridged network and a spanning tree configuration for it.

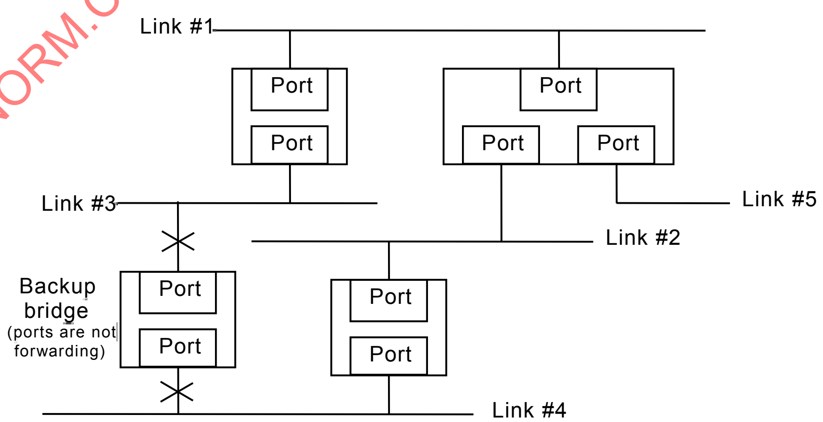


Figure 14 – Bridged network topology

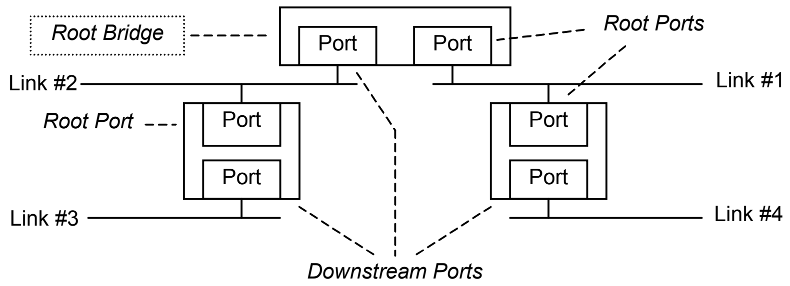


Figure 15 – Spanning tree representation

All bridge DLEs are responsible for performing these DLL activities:

a) forwarding,

b) republishing,

c) data link time synchronization.

Bridge DLEs conforming to 10.4.2 are also responsible for

d) self-configuring into a spanning tree,

e) dynamically establishing most of the routing information in their forwarding databases.

To support c), it is necessary for bridge ports to be the Link Active Scheduler (LAS) of their downstream links. Thus the spanning tree connectivity drives the LAS selection procedure. In the figure above, each active downstream port of a bridge functions (and is required to function) as the LAS for its associated link.

Forwarding occurs when a bridge receives a DLPDU with a first DL-address for which the bridge has a rule in its forwarding database. If the address indicates that the message is to be forwarded, the bridge *appends* the message to the FIFO output queue for the appropriate port(s). When the bridge receives the right to transmit from the LAS associated with that port, the bridge transmits entries from the queue in priority order.

Republishing occurs when the bridge is configured to republish DLSDUs. When the bridge receives a DLSDU which it is to republish, it buffers the DLSDU for scheduled transmission on the appropriate outbound ports as specified in its republishing database. The DLSDU is *republished* separately onto each of the outbound links.

Synchronization of DL-time occurs when a DL-time DLPDU is received by a bridge on its root port. DL-time originates with the root port of the root link. When a DLE receives a TD (DL-time) DLPDU, it recalculates the offset for the link scheduling time of each of its outbound ports for which it is the LAS. If the bridge contains a downstream port for which it is not the LAS, then the DL-time cannot be redistributed to that link by that bridge.

A bridge provides the normal DL-functions for each port, plus specialized forwarding, republishing, time propagation and bridge coordination functions. A bridge operates below the DL-Service boundary, and is transparent to protocols operating above this boundary. An extended link may provide the following:

1) An effective increase in the physical extent, the number of permissible attachments, or the total performance of a link over that possible without use of bridges.

2) Partitioning of the physical link for administrative, maintenance, fault limiting or security reasons.

3) Integration of individual links of different data rates into a single coherent network.

NOTE A *bridge* is a DL-relay entity, and should not be confused with a *repeater*, which is a PhL-relay entity. Bridges selectively forward DLPDUs and usually introduce significant delay in the forwarding process; repeaters forward PhPDUs unselectively with almost no delay. Bridges have substantial internal state; repeaters have almost none. Bridges can interconnect *links* of different data rates; repeaters interconnect *segments* of identical data rate, possibly with different modulation or types of media.

10.4.1.3 Extent of specification

This subclause specifies a general method for the operation of bridges. Bridges provide compatible interconnection of automation equipment that implement the corresponding Type 1 protocol. Subclause 10.4.1.3

a) positions the bridging function within an architectural description of the DLL;
b) defines the principles of operation of a bridge in terms of the support and preservation of the DL-Service, and the maintenance of Quality of Service (QoS);
c) identifies the functions to be performed by bridges, and provides an architectural model of the internal operation of a bridge in terms of processes and entities that provide these functions;
d) specifies a minimal protocol between bridges in the extended link, such that the resulting bridges (see 10.4.3) interoperate with bridges that meet the full requirements of 10.4.1.5.

The specification of Remote Bridges, which interconnect Type 1 links using other communications protocols for the transmission of DLPDUs among bridges, is outside the scope of this specification.

10.4.1.4 Support of the DL-Service

Bridges interconnect separate links into an integrated network, known as the *extended link*, by

a) forwarding DLPDUs between the separate links;
b) subscribing to information being published on one link, and republishing it on one or more of the other links to which the bridge connects;
c) providing a synchronized sense of DL-time among the links to which the bridge is directly connected and currently forwarding, or for which the bridge is currently acting as LAS.

The DL-Service is provided to the DLS-user in the end stations and is supported by the forwarding function in the bridge. This subclause discusses provisions of the DL-Service to end stations, support of the DL-Service by bridges, preservation of the DL-Service offered on the extended link, and maintenance of QoS in the extended link.

The DL-service provided to end stations attached to an extended link is supported by the bridges in that extended link. Bridges support all of the DL-services that involve multi-end-station communication.

The style of bridge operation maximizes the availability of the DL-Service to end stations and assists in the maintenance of the extended link. It is therefore desirable that bridges be capable of being configured in the extended link:

1) To provide redundant paths between end stations to enable the extended link to continue to provide the DL-Service in case of component failure (of bridge or basic link).
2) So that the paths supported between end stations are predictable and configurable given the availability of components of the extended link.
3) So that DLPDUs on one link are forwarded selectively to other links, with the forwarding decisions based on configured criteria.
4) So that DLSDUs of a DLC on one link are republished selectively to DLCs on other links, with the subscribing and republishing relationships based on configured criteria.
5) So that bridges implementing the full Spanning Tree protocol of 10.4.1.5 defer to the non-adaptive bridges of 10.4.3 when constructing the spanning tree for the extended link.

10.4.1.4.1 Preservation of the DL-Service

The DL-Service offered by an extended link, consisting of single links interconnected by bridges, is similar to that offered by a single link. In consequence,

a) A bridge is not directly addressed by communicating end stations when forwarding DLPDUs between links; DLPDUs transmitted between end-stations carry DL-addresses of end-stations in their address fields, not the DL-address of the bridge. However, the bridge itself serves as an end-station when republishing, and when providing access to management agents and higher-layer entities collocated with the bridge.
b) All DL-addresses must be unique and addressable within the extended link.
c) The topology and configuration of the extended link do not otherwise restrict the DL-addresses of end stations.

10.4.1.4.2 Quality of Service (QoS) maintenance

Bridges are designed so that the quality of the DL-Service supported by a bridge is not significantly inferior to that provided by a single link. The QoS parameters considered are those related to:

a) Service availability
b) Shared sense of DL-time
c) DLPDU and DLSDU loss
d) DLPDU and DLSDU misordering
e) DLPDU and DLSDU duplication
f) DLPDU transit delay
g) DLPDU lifetime
h) Undetected error rate in forwarded DLPDUs and republished DLSDUs
i) DLPDU and DLSDU priority
j) DLSDU timeliness
k) Throughput
l) Schedule skew when republishing DLSDUs

NOTE Bridges forward DLPDUs and republish DLSDUs. Therefore, to a first approximation, QoS related to DLPDUs applies to forwarding, while QoS related to DLSDUs applies to republishing.

10.4.1.4.2.1 Service availability

The DL layer provides the DL-Service to end stations attached to a single link or extended link. Service availability is measured as that fraction of some total time during which the service is provided. The operation of a bridge can increase or decrease the service availability.

The service availability can be increased by automatic reconfiguration of the extended link in order to avoid the use of a failed component (that is, a repeater, cable or connector) in the data path. The service availability can be lowered by failure of a bridge itself, through denial of service by a bridge, or through undesired DLPDU discard by a bridge during intervals of bridge resource saturation.

To maximize the service availability, no loss of service or delay in service provision should be caused by bridges, except as a consequence of a failure, removal or insertion of a component of the extended link. This is regarded as an extraordinary event.

NOTE A loss of service caused by congestion at an outbound bridge port is not induced by the bridge, but by the structure and use of the extended link. Such loss can be expected, particularly where a higher-capacity link is forwarding to a much lower capacity one. In contrast, a loss of service caused by congestion at an inbound bridge port is attributable to the design of the bridge itself, in conflict with the above recommendation.

10.4.1.4.2.2 Shared sense of DL-time

The DLL provides a shared sense of DL-time. The degree of time synchronization of any two end-stations on the extended link is limited by the characteristics of the communications path between them. Synchronization degrades with an increasing number of intermediate bridges.

10.4.1.4.2.3 DLPDU and DLSDU loss

The DL-Service does not guarantee the delivery of DLSDUs. However, some DLCs notify their associated DLS-user of the detection of DLSDU loss.

DLPDUs transmitted by a source station arrive, uncorrupted, at the destination station with high probability. The operation of a bridge introduces minimal additional DLPDU loss.

A DLPDU transmitted by a source station can fail to reach its destination station as a result of

a) DLPDU corruption during Physical Layer transmission, relaying or reception;
b) DLPDU discard by a bridge because
   1) it is unable to transmit the DLPDU within some maximum specified time and, hence, must discard the DLPDU to prevent the maximum DLPDU lifetime (10.4.1.4.2.7) from being exceeded;
   2) it is unable to continue to store the DLPDU due to exhaustion of internal buffering capacity as DLPDUs continue to arrive at a rate in excess of that at which they can be processed and forwarded.

10.4.1.4.2.4 DLPDU and DLSDU misordering

The DL-Service does not guarantee the delivery order of DLSDUs unless that ordering has been negotiated as part of the QoS for a DLC. However, DLPDUs of the same DL-priority for any given sending and receiving port pair are not misordered within a bridge; when neither DLPDU is discarded, the earlier-received DLPDU will be the earlier forwarded.

Misordering can occur immediately after any reconfiguration of the DL-path between two end stations involving changes to two or more bridges in the DL-path. Reconfiguration involving one bridge substituting for another parallel bridge that is directly connected to the same links does not cause misordering.

10.4.1.4.2.5 DLPDU and DLSDU duplication

Except for UNORDERED DLCs, the DL-Service does not permit the duplication of DLSDUs. Normal operation of bridges does not introduce duplication of DLPDUs.

The potential for DLPDU duplication in an extended link arises from the possibility of multiple paths between source and destination end stations. This can only arise due to erroneous configuration or malfunction of a bridge, including failover of a bridge to a replacement bridge that is using an anticipatory forwarding strategy.

10.4.1.4.2.6 DLSDU transit delay.

The DL-Service introduces a DLSDU transit delay that is dependent on the particular media and link transmission utilization of the DL-path employed. DLSDU transit delay is measured at the DLS interface (see Figure 16). DLSDU transit delay is the elapsed time between

a) the sending DLE's receipt of a DLS primitive requesting the transmission of DLS-user data, or the equivalent event triggering transmission from a publisher DLCEP when no DLSDU send primitive is involved, and
b) the receiving DLE's signaling of the availability of the DLS-user data at its DLS interface, or the equivalent event signaling reception at a subscriber DLCEP when no DLSDU delivery primitive is involved.

Elapsed time values are computed only on DLSDUs that are successfully transferred.

Since the DLS generally is provided at an abstract interface within the end station, it is not possible to specify precisely the total DLSDU transit delay. It is, however, possible to measure those components of delay associated with access to the medium and with transmission and reception. Likewise, those components of the transit delay introduced by a bridge can be measured.

The minimal additional transit delay introduced by a bridge is the sum of:

1) the time taken to (wholly or partially) receive a DLPDU conveying the DLSDU and determine that it needs to be forwarded;
2) any time required to prepare the DLPDU for forwarding;
3) the time taken to access the medium onto which the DLPDU is to be forwarded.

NOTE 1 If the DLPDU is completely received before it is forwarded, and if a frame check sequence (FCS) error is detected in the received DLPDU, then that DLPDU will be discarded without forwarding.

NOTE 2 If the DLPDU is only partially received before forwarding commences (known as "cut-through switching"), the delay induced by the bridge can be as small as a few octet-durations of the link from which the DLPDU is received.

NOTE 3 The concept of transit delay applies only to the bridge's forwarding activities, and not to its republishing or time-propagation activities.

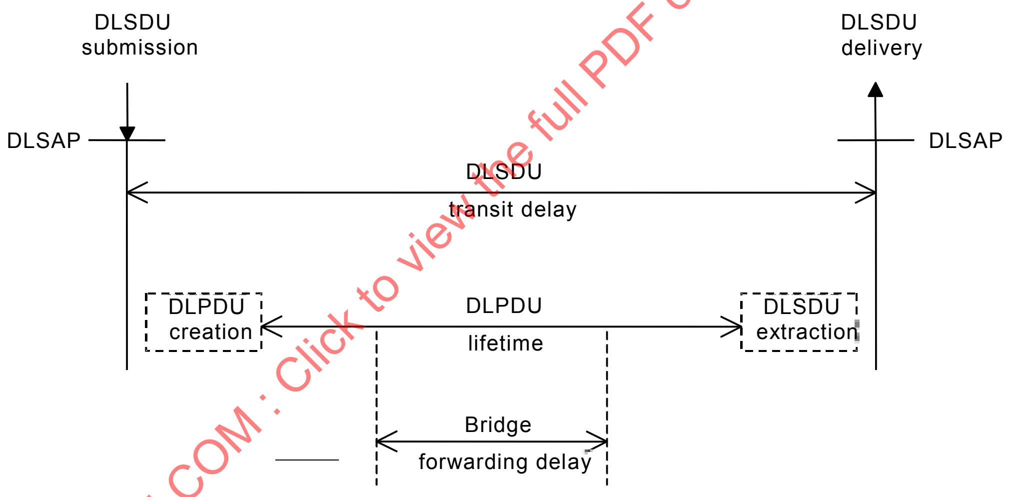


Figure 16 – DLSDU transit delay, DLPDU lifetime and bridge forwarding delay

10.4.1.4.2.7 DLPDU lifetime

The DL-Service ensures that there is an upper bound to the transit delay experienced for each DLS-instance. To ensure this, the DLS imposes a maximum lifetime on each DLPDU conveying a DLSDU. DLPDU lifetime is measured within the distributed DLE, from the moment when a DLSDU is packaged into DLPDUs for transmission, until the DLSDU is extracted after DLPDU reception (see Figure 16).

To ensure an upper bound to the DLPDU lifetime, both originating end stations and bridges are required to discard any DLPDU that has been queued for transmission for an excessive amount of time. Within a bridge, this queuing time is known as *forwarding delay*, and is measured from the moment when the DLPDU is completely received (the end-of-reception event) to the moment that the DLPDU retransmission commences (the start-of-transmission or transmit-commit event).

Since the information provided by the DL-Protocol to a bridge does not include the transit delay already experienced by any individual DLPDU, bridges must enforce a configurable maximum forwarding delay in each bridge, based only on DLPDU priority. They do this by discarding a DLPDU based on the length of time the DLPDU has been in the bridge, waiting for retransmission as a forwarded DLPDU.

The value of the maximum permissible forwarding delay for any given bridge is configurable for each DLPDU priority, and is based on

a) the desired maximum DLPDU lifetime for that priority class of DLPDUs, and
b) the set of DL-paths on the extended link which go through that bridge.

10.4.1.4.2.8 Undetected error rate in forwarded DLPDUs and republished DLSDUs

Protection against undetected errors is provided by the use of a FCS that is

a) included at the end of each DLPDU by the sending DLE prior to transmission;
b) propagated by any bridges, possibly after altering the FCS algorithmically to compensate for the complementation of a single fixed bit in the first octet of the DLPDU, as specified in 5.2.5.3;
c) checked by all receiving DLE(s).

DLSDUs retained within the bridge for republishing are also subject to errors due to malfunction of the bridge hardware or software, which can corrupt the DLSDU while within the bridge. The FCS propagation algorithm for forwarded DLPDUs, which is the basis for the end-to-end error detection of this protocol, is not readily applicable to DLSDUs that are republished. A bridge need not detect corruption of DLSDUs awaiting republication, nor need the FCS of any republishing DLPDU be a transformation of the FCS of a corresponding subscribing DLPDU.

10.4.1.4.2.9 DLPDU and DLSDU priority

Bridges do not alter the priority of received and forwarded DLPDUs. Bridges do not alter the priority of republished DLSDUs.

10.4.1.4.2.10 DLSDU timeliness

Bridges implement DLSDU timeliness computations on republished DLSDUs according to their configuration as subscribers and republishers. In most cases bridges provide TRANSPARENT timeliness, propagating whatever sense of untimeliness was received to those DLEs to whom they republish.

10.4.1.4.2.11 Throughput

Bridges localize traffic within the extended link by forwarding only selected DLPDUs and republishing only selected DLSDUs. This permits the total capacity of the extended link (in DLSDUs transferred per second) to be significantly greater than that provided by an equivalent single link.

10.4.1.4.2.12 Schedule skew when republishing DLSDUs

A bridge may require some minimum time after receipt of a DLSDU at a subscriber DLCEP before that same DLSDU is available for republication at a publisher DLCEP on another port. The scheduler for the republishing link may need to have an upper bound on this time interval. Therefore each bridge has a MinRepublishingDelay attribute that specifies the minimum time which the scheduler must allocate for propagation of the DLSDU from the port of the bridge at which it was received to the port of the same bridge at which it is to be republished.

10.4.1.4.3 Forwarding of DLPDUs

A bridge receiving a DLPDU examines the DLPDU's frame control octet and first address, from which the bridge determines whether and to where the DLPDU should be forwarded. Forwarding decisions are based on information in the bridge's filtering database.

A bridge that claims full conformance to 10.4.1.5 forwards a CA, CD, DC, DT, EC, ED or RC DLPDU, together with its received FCS, toward those links that have DLEs that should receive the DLPDU's first DL-address. If the received DLPDU is a CA, CD or ED DLPDU, then such a bridge replies immediately with a SR DLPDU on the link from which the CA, CD or ED DLPDU was received.

No other forwarding of DLPDUs occurs.

10.4.1.4.4 Republishing of DLSDUs

Bridges are configured as subscribers to receive DLSDUs which they also may be configured to republish.

A subscribing bridge completes DLC establishment with its publisher (which could be another bridge acting as a republisher), after which it receives published DT DLPDUs, as do other subscribers. The bridge extracts the DLSDUs and timeliness status from these received DLPDUs, buffers the extracted information, and republishes it on DLCs for which the bridge is configured as a publisher and has completed DLC establishment.

The bridge's Republishing Database is configured with the DLC characteristics of each republishing DLC for which the bridge is responsible, and of the corresponding subscriber DLC which provides the DLSDUs for republishing.

10.4.1.4.5 Propagation of DL-time

Bridges are configured so that the aggregate set of bridge ports that are in the forwarding state form an acyclic graph, known as a *spanning tree*, which connects all of the links of the extended link.

A bridge attempts to synchronize the sense of DL-time among those ports that are in the FORWARDING state (see 10.4.1.5 and 10.4.3) or are acting as LAS.

If all of the bridge's forwarding ports are acting as LAS, then the bridge is at the *root* of the spanning tree and it propagates its own internal sense of DL-time to all of those ports which are acting as LAS. A bridge that also has a gateway to another network may synchronize its internal sense of DL-time to a time sense received from that other network. In this case, the bridge functions as the root of the spanning tree for its local subnetwork.

Otherwise, the bridge has precisely one port that is forwarding and that is not acting as LAS; this port is designated the *root* port of the bridge. The bridge propagates DL-time from its root port to the ports that are acting as LAS, which includes all of its other ports which are in the FORWARDING state.

NOTE Any ports which are not in the FORWARDING state and which are not acting as LAS will have their time sense synchronized to their attached link, rather than to the internal time sense of the bridge.

10.4.1.5 Principles of operation

Subclause 10.4.1.5 establishes the principles of operation, and a model for the operation, of a bridge as follows:

a) Provides a model of operation of a bridge, explains the principal elements of bridge operation and lists the functions that support these.

b) Discusses the architectural model for a bridge, found in 4.1.2, which governs the provision of these functions. This includes specifying the states of a bridge port, and the format and associated protocol for BPDUs that facilitate interoperation with spanning tree bridges.

c) Details the addressing requirements of an extended link and specifies the addressing of entities in a bridge.

10.4.1.5.1 Bridge operation

The principal elements of bridge operation that differ from the operation of other DLEs are

a) Selectively forwarding received DLPDUs.

b) Selectively republishing received DLSDUs on other DLCs, and subscribing to other DLCs to receive these DLSDUs.

c) Selective propagation of a shared sense of DL-time.

d) Minimal participation in the inter-bridge spanning tree protocol of 10.4.1.5.

NOTE 1 Conceptual models of DLPDU reception, DLPDU forwarding, local DLPDU origination and DLSDU republishing are intended to facilitate an understanding of these processes.

NOTE 2 DLPDUs addressed to any port of a bridge are addressed to the bridge itself. The filtering process is not involved in their local delivery, and the requirements for forwarding do not apply.

10.4.1.5.1.1 Filtering and forwarding

A bridge forwards individual DLPDUs between the separate links connected to its *ports*. The decision of whether to forward, and to which ports to forward, is based on information in a filtering database.

NOTE The term filtering is used in this sense as a permissive filter, because that is the sense of the information configured in the filtering database. The more common language sense of filtering for discard applies to the set of unselected ports, which is the inverse of the database information.

The order of DLPDUs of a given DL-priority received on one port and retransmitted on another is preserved.

The functions that support the forwarding of DLPDUs and maintain the QoS supported by the bridge are

a) Reception of all DLPDUs that have no detected error, including retention of the DLPDU's FCS as received.

NOTE This is a generalization of the basic reception function found in all DLEs.

b) Submission of received DLPDUs to the filtering process for potential forwarding if
   1) the DLPDU contains one or more long addresses whose link designators are all non-zero,
   2) and
      i) the DLPDU type is CA, CD, DC, DT, EC, ED or RC, or
      ii) the DLPDU is a DT DLPDU that was created in accordance with 6.7.4.3.1 or 6.8.4.3 b).

c) Filtering of locally originated DLPDUs based on the first address of the DLPDU, with forwarding to any or all of the ports of the bridge as specified in the filtering database.

d) DLPDU discard to ensure that a maximum forwarding delay is not exceeded.

e) Possible modification of a DLPDU's frame control octet, with compensatory modification of the DLPDU's FCS.

f) FCS calculation on those DT DLPDUs created as specified by b)2)ii).

   NOTE This is part of the basic transmission function found in all DLEs.

g) DLPDU transmission, with transmission order determined both by DLPDU priority and by forwarding delay (for example, the order in the forwarding queue), and with the (modified) forwarded FCS used where available.

NOTE This is a generalization of the basic transmission function found in all DLEs.

10.4.1.5.1.1.1 Forwarding of a received DLPDU

A lower-level port-specific process receives a DLPDU, validates its FCS, and classifies the DLPDU based on its frame control octet. If the received DLPDU contained one or more explicit DL-addresses, and all of those addresses were LONG with non-zero link designators, and the link-designator of the first DL-address is in the forwarding database of the DLE, then the DLPDU record is passed to the lower-level transmission process of each port associated with that link designator in the forwarding database, provided that that port is in the FORWARDING state and that it is not the port on which the DLPDU was received.

DLPDUs are forwarded based on DLPDU priority and their order or length of time in the forwarding queue. A new FCS is never calculated for a forwarded DLPDU. However, the received FCS may be modified before forwarding as specified in 5.2.5.3.

DLPDUs that have been in the forwarding queue longer than the configured MaxForwardingDelay for the DLPDU's priority are discarded.

Statistics and diagnostic information should be kept as specified in 10.5.6.3.

Other aspects of DLPDU reception and subsequent local delivery are identical to those of a non-bridge DLE.

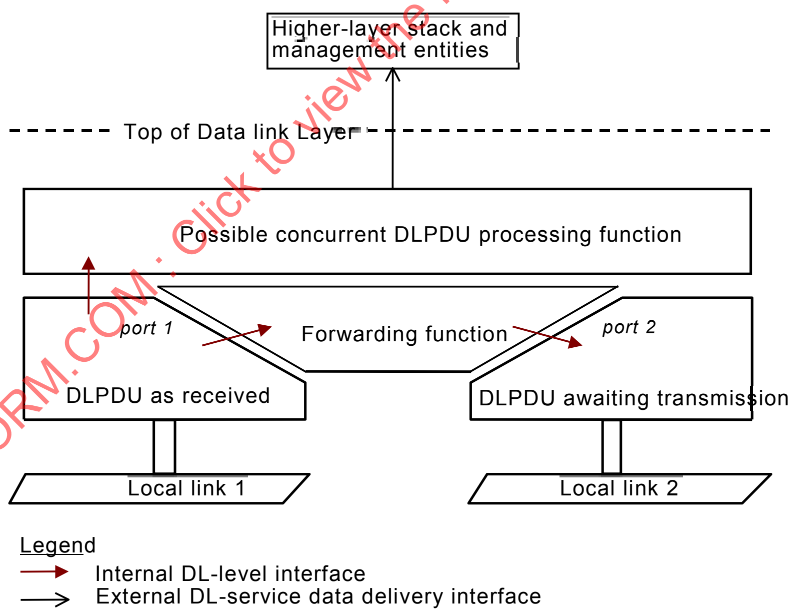


Figure 17 – Forwarding and delivering a received DLPDU

NOTE Figure 17 shows possibly-concurrent forwarding and delivery paths.

10.4.1.5.1.1.2 Local origination of a DLPDU

A higher-level process creates a DLPDU and passes it to the lower-level transmission process of each port associated in the forwarding database with the link designator of the DLPDU's first DL-address, whether or not that port is in the FORWARDING state. Other aspects of local origination and local delivery are identical to those of a non-bridge DLE. See Figure 18.

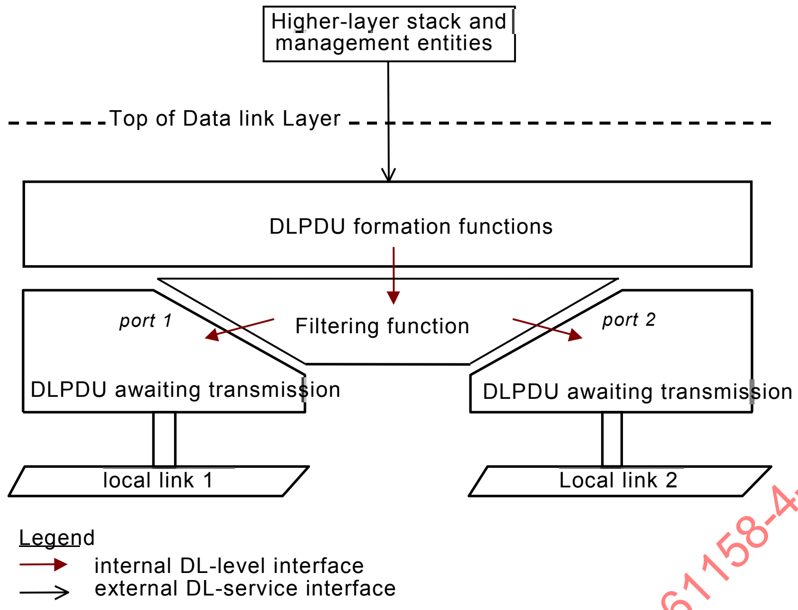


Figure 18 – Forwarding a locally-originated DLPDU

10.4.1.5.1.2 Subscribing to one DLCEP and republishing on another DLCEP

Each bridge may function as a republisher of published data. The bridge subscribes to one DLC and republishes received DLSDUs on one or more other DLCs as specified by its Republishing Database.

Each entry in the Republishing Database specifies 32-bit delocalized subscriber and publisher DLCEP-addresses and the characteristics of the subscribed-to DLC. The latter are also used as the characteristics of the republishing DLC.

NOTE 1 When periodic republishing is required on a port, the link schedule for that port's link must contain an entry to compel data transmission by the republishing DLCEP.

At port initialization, or whenever the state of a port is changed to FORWARDING, the bridge management entity (BME) initiates all (re)publishing DLCEPs described in its Republishing Database that have a delocalized DLCEP-address whose DL-link designator is associated with that port by the Filtering Database.

At port initialization, the BME initiates all subscribing DLCEPs described in its Republishing Database. Multiple entries in the Republishing Database for a single subscriber DLCEP-address may result in the establishment of only a single subscribing DLCEP.

DL-buffers are allocated as necessary during this process, so that each subscribing DLCEP and all associated (re)publishing DLCEPs are bound to the same DL-buffer.

If the bridge's DL-management changes the state of a port from FORWARDING to another state, the bridge management entity terminates republishing on the DLCEPs associated with that no-longer-forwarding port. The bridge may, but need not, send DC DLPDUs on those republishing DLCs which are being terminated.

NOTE 2 When management action is the mechanism for switching among the bridges of a redundant set of bridges, transmission of a separate DC DLPDU on each of the bridge's open republishing DLCs is not recommended.

The bridge management entity is not involved in the actual process of republishing DSLDUs. As each new DLSDU is received at a subscribing DLCEP, it is written to the DL-buffer. Writing the DL-buffer and its associated timeliness information is the event which signals each connected republishing DLCEP that the DL-buffer has newly-updated contents.

NOTE 3 Reception and detection of a duplicate DLSDU at a subscribing DLCEP results in no update of the buffered DLSDU and no change in that buffered DLSDU's timeliness attribute.

The rates at which DLSDUs are written to the DL-buffer by the subscribing DLCEP, and are read from the buffer by the publishing DLCEP(s), are determined by the associated link schedules. The two rates need not be identical.

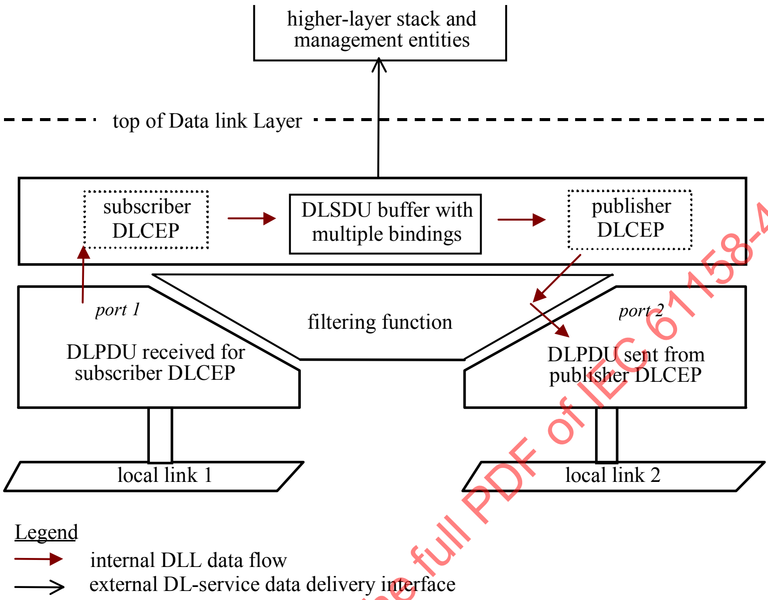


Figure 19 – Republishing a DLSDU received from another link

NOTE 4 The possibly-concurrent forwarding path, from one port to another, for the DLPDU received at the subscribing DLCEP is not shown. It is identical to the forwarding path of Figure C.2.

10.4.1.5.1.3 DL-time propagation

Each bridge functions as a propagator of DL-Time. The bridge adjusts its internal sense of DL-time to that of the single port, if any, which is in the FORWARDING state and is not acting as LAS. (This is necessarily the root port of the bridge.)

The bridge adjusts the sense of DL-time of each port that is acting as LAS to the bridge's internal sense of DL-time.

The functions performed by the bridge to support DL-time propagation are:

a) If the port is not acting as LAS, receive TD DLPDUs and adjust local DL-time in the receiving bridge port, regardless of the port's state.

   NOTE This is part of the standard functions found in all basic DLEs.

b) If the receiving port is in the FORWARDING state, then use the port's DL-time as the bridge's internal sense of DL-time.

   NOTE At most one port can be in the forwarding state and not acting as LAS.

c) Adjust local DL-time in each of the bridge ports that are acting as LAS, based on the bridge's internal DL-time and time-rate-of-change.

d) Create and transmit new TD DLPDUs on any port at which the bridge is acting as LAS, possibly because of a significant adjustment of the port's DL-time.

   NOTE This is part of the standard functions found in all LM DLEs.

10.4.1.5.1.4 Access to higher-layer entities within the bridge

Each bridge port also functions as an end station providing access to its local management agent (through System Management protocols), and possibly to other applications within the device.

10.4.1.5.2 Bridge Architecture

The bridge consists of normal DL-functions for each port, plus specialized forwarding, republishing, time propagation and bridge coordination functions. See Figure 20.

The normal low-level DL-functions of each port are responsible for receiving DLPDUs from the associated link addressed to that port. They are also responsible for sending DLPDUs queued for transmission at that port.

The filtering functions of the bridge use the configured filtering information to determine whether a received DLPDU should be discarded or should be forwarded to one or more ports other than the port on which it was received. They also determine to which ports a locally originated DLPDU should be sent.

The bridge coordination functions of 4.1.2 include the bridge management entity, which manages republishing DLCs and interacts with the bridge management entities of other bridges in the formation and maintenance of the bridge spanning tree. In non-adaptive bridges, only a minimal subset of the 10.4.1.5 spanning tree functionality is required, to permit non-adaptive bridges to coexist and interoperate with adaptive bridges conforming fully to 10.4.1.5.

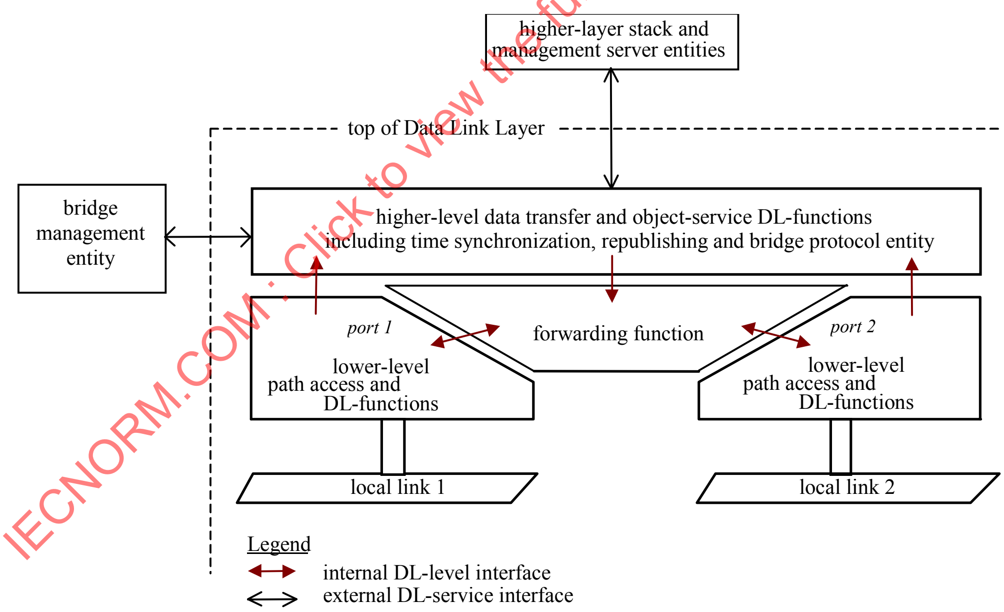


Figure 20 – Bridge architecture

10.4.1.5.2.1 Bridge management entity (BME)

The BME is responsible for

a) subscribing to DLCs which are to be republished,

b) initiating republishing DLCs and connecting subsequent subscribers to those republishing DLCs, and

c) interacting with other bridges on the connected links in establishing the topology of the extended network and determining which non-disabled bridge ports should be forwarding and which should be blocking.

The BME initiates each subscribing DLCEP and one or more associated republishing DLCs, as configured in the republishing database. by

1) allocating a DL-buffer to be used to hold each DLSDU received at the subscribing DLCEP before it is to be republished,
2) subscribing to the DLC which is providing the information to be republished, and
3) initiating each publishing DLC which is to republish the subscribed-to information.

The timing of these actions is as follows:

i) The DL-buffer is allocated before either the subscribing DLCEP or any of the republishing DLCs are established.
ii) The subscribing DLCEP is established during bridge initialization or change of port state for any port which either is in or enters the Blocking or Forwarding state.

Any connection establishment request addressed to the publisher of a republishing DLC is delivered to the BME, which responds by connecting the would-be subscriber to the existing republishing DLC.

The BME also responds to any connection establishment or disconnection DLPDUs received at those DLCEPs, and to any change in the operational status of the relevant ports.

The actual act of receiving DLSDUs as a subscriber and republishing them as a publisher is handled as part of the standard connection-mode functions of the IEC Data Link, using a DL-buffer that is bound to the subscribing DLCEP and to each republishing DLCEP.

NOTE Reception and detection of a duplicate DLSDU at a subscribing DLCEP results in no update of the buffered DLSDU and no change in that buffered DLSDU's timeliness attribute.

The BME also generates DLPDUs for, and responds to DLPDUs received from, the BMEs of other bridges on the immediately connected links. In 10.4.1.5 the BME (therein called the bridge protocol entity (BPE) manages the bridge spanning tree. In non-adaptive bridges, only a minimal subset of the 10.4.1.5 functionality is required, to permit non-adaptive bridges to coexist and interoperate with adaptive bridges conforming fully to 10.4.1.5.

10.4.1.5.2.2 Root port determination

A bridge which is not at the root of the bridge spanning tree has one port which most closely connects it to the root of the spanning tree. That port is known as the bridge's *root* port. In bridges conforming to 10.4.1.5, that port is determined automatically by the ISO/IEC Spanning Tree Protocol.

In non-adaptive bridges which do not implement the full spanning tree protocol, the bridge's root port, if any, is determined by the bridge's RootPort entry as configured by DL-management. For example, a non-zero value could identify that port of the bridge which is the root port; a zero value would indicate that the bridge is itself the root of the bridge spanning tree.

A bridge port which transitions to the FORWARDING state uses the RootPort information to determine whether it should also become LAS on its connected link; the port needs to become the LAS if and only if it is not the bridge's root port.

NOTE This requirement that a bridge which is not the root port become its local link's LAS minimizes the skew in DL-time among the set of connected bridges. The root port receives DL-time on its local link from that link's LAS; thus the root port must not itself act as LAS. The bridge at the root of the spanning tree acts as time source for the entire spanning tree; therefore none of its ports are designated a root port.

10.4.1.5.2.3 Port state information

State information associated with each bridge port governs its participation in the extended link. A port of a bridge is defined to be in one of three or more states set either by DL-Management or by the inter-bridge protocol of 10.4.1.5. The three basic states are:

a) **DISABLED** – The port is incapable of DL-operation as a bridge-class DLE, usually because the bridge portion of its database has not been configured. The port is always capable of operation as a basic-class DLE, at least to the extent of being brought into active participation on its connected link and providing access to any Systems Management agent. The default state for each bridge port, before being configured by DL-Management, is DISABLED.

   NOTE A disabled port may function as either a basic-class or link master-class DLE, as specified in its standard NMIB for the port. The safest default states for a port are DISABLED and basic class.

b) **BLOCKING** – The port is capable of DL-operation as a bridge-class DLE, but is configured to function as a link master rather than a bridge. It does not forward DLPDUs or republish DLSDUs, and does not participate in the bridge management of bridge annunciation protocol. It subscribes to DLSDUs that it would republish if it were in the FORWARDING state, so that it is prepared to transition to that state. If it is not acting as LAS on its attached link, then it synchronizes its local time sense to the attached link's, rather than to the bridge's internal time sense.

   A port that is a redundant backup for a corresponding port of a parallel bridge would be in the BLOCKING state.

c) **FORWARDING** – The port is operating as a bridge-class DLE, forwarding DLPDUs and republishing DLSDUs as directed by its DL-management configuration., synchronizing its time sense to that of the root port of the bridge if it is not itself the root port, and participating in the bridge annunciation protocol whose purpose is to enable coexistence and interoperation with bridges using the Spanning Tree protocol of 10.4.1.5.

NOTE 1 DLPDUs addressed to any port of a bridge are addressed to the bridge itself. The port states of the receiving port and the addressed port have no affect in their local delivery.

Transitions between these states can be commanded by DL-management. Transitions between the BLOCKING and FORWARDING states also can be commanded by a proprietary bridge-internal redundancy manager which coordinates the port states of parallel ports of parallel bridges in a redundant bridge configuration.

NOTE 2 Proprietary bridge-internal redundancy managers are expected to coordinate only like bridges from a single vendor.

Concurrent with entering the FORWARDING state, a bridge port may have to assume the LAS role for the connected link. This assumption can be by extra-protocol means when it occurs between parallel bridges managed by a common bridge-internal redundancy management protocol. Otherwise this assumption uses the mechanism for transferring the LAS role described in Clause 11, using the IEC DLP. A bridge port which is in, or entering, the BLOCKING or DISABLED state shall relinquish its LAS role upon request.

Change of bridge port state is not always the consequence of failure. When two bridges cooperatively exchange port states, with one transitioning from BLOCKING or LEARNING to FORWARDING, and the other from FORWARDING to LEARNING or BLOCKING, the bridges should transition their ports into the new state synchronously with their release of, or acceptance of, the LAS role at that port. This change of port state will affect future filtering decisions on DLPDUs received at or for that port, and future republishing actions. However, it is recommended that a port which has just transitioned to the LEARNING or BLOCKING state continue to transmit already-queued DLPDUs from its forwarding queues.

10.4.1.5.2.4 Filtering database

A bridge filters DLPDUs by determining to which ports, if any, a received DLPDU should be forwarded. Filtering is used to confine DLPDUs transmitted between end stations to the subset of the extended link that forms a DL-path between those end stations when such a path is known.

NOTE The term filtering is used in this sense as a permissive filter, because that is the sense of the information configured in the filtering database. The more common language sense of filtering for discard applies to the set of unselected ports, which is the inverse of the database information.

Each bridge contains a Filtering Database. Conceptually, the Filtering Database consists of two lists of records, where the first specifies default forwarding for links:

individual DL-addresses

a) a DL-link designator in the range 1 000 to V(ML), the maximum link designator value;
b) identification of a set of ports to which a DLPDU addressed to this link should be forwarded when all other enabling conditions are met.

The second list specifies forwarding for individual DL-addresses

c) a DLSAP-address, DLCEP-address or group DL-address;
d) identification of a set of ports to which DLPDUs with this destination DL-address should be forwarded when all other enabling conditions are met;
e) identification of a set of ports disjoint from those specified in d) to which DLPDUs containing just a source address (that is, publishing DLPDUs) should be forwarded when all other enabling conditions are met.

   NOTE This last set of ports is of use only for publisher DLCEP-addresses.

10.4.1.5.2.5 Republishing database

A bridge republishes DLSDUs by creating a DL-buffer, subscribing to a DLC on one port, establishing publishing DLCs on one or more other ports, and binding the receive DL-buffer for the subscribing DLCEP as the send DL-buffer for the republishing DLCEP(s).

Each bridge contains a Republishing Database. Conceptually, the Republishing Database consists of a list of records, each specifying

a) the 32-bit DLCEP-address of the DLC to which the bridge should subscribe for the purpose of republishing;
b) the characteristics of that DLC (maximum DLSDU size, priority, data delivery features, authentication, etc.);
c) the 32-bit DLCEP address(es) of the DLC (s) which the bridge should initiate as republisher.

10.4.1.5.3 Addressing

10.4.1.5.3.1 Link designators

Each of the constituent links, which together comprise the extended link, has its own unique link designator. The forwarding database of each bridge has an entry for each of these links, specifying the bridge port through which (direct or indirect) access to the link is obtained.

10.4.1.5.4 Statistics and diagnostic information

For each receiving port, statistics should be kept on the number of received DLPDUs

a) whose first delocalized DL-address specifies a link designator that is not in the Forwarding Database, or
b) that do not require forwarding, based on information in the Forwarding Database, or
c) that are queued for forwarding, frequently after being placed in a forwarding buffer, or

d) that are discarded due to lack of forwarding buffers.

NOTE Each received DLPDU will fall into exactly one of these categories.

For each transmitting port, statistics should be kept on the number of DLPDUs that were queued for forwarding on that port that have been discarded due to excessive forwarding delay.

The header of the last DLPDU whose first DL-address specifies a link designator that is not in the bridge's Forwarding Database should be retained for diagnostic use.

10.4.1.6 Detailed conceptual model of bridge functions (informative)

The forwarding functions of a bridge are a specialized extension of the address-based delivery functions intrinsic to all DLEs. The following conceptual models of processes, entities and data structures, which apply to the non-adaptive bridging subset of IEC DLP, facilitate an understanding of this process. Alternative conceptualizations are possible.

10.4.1.6.1 DLPDU reception

The lower-level DLL process associated with each bridge port examines all DLPDUs received from the link to which it is attached. DLPDUs that are detected to be in error are discarded; this includes DLPDUs whose FCS is in error.

Received DC, DT and EC DLPDUs with LONG DL-addresses whose link designators are all non-zero are submitted to the forwarding process. Other DLPDUs may be submitted to the forwarding process or not as permitted by the relevant DLPDU-specific subclause of the form 8.*nn*.4.3.

DLPDUs addressed to the bridge as an end station are submitted to the upper-level-DLL for processing. These may result in delivery of a DLSDU to one or more DLS-user entities at one or more DLSAPs.

10.4.1.6.2 Model of reception and forwarding of a DLPDU

A lower-level port-specific process receives a DLPDU, validates its FCS, and classifies the DLPDU based on its frame control octet. The receive process creates a received-DLPDU record consisting of:

a) the DLPDU class;
b) the delocalized DL-address(es) explicitly or implicitly specified by the DLPDU's header;
c) the octets representing the *body* of the DLPDU – the DL-parameters, or DLS-user data, or both;
d) the port number of the port from which the DLPDU was received; and
e) the DLPDU's lower-level header and trailer octets (that is, received frame control, address and FCS octets).

NOTE Components a) through c) are identical for, and apply to, both bridge and non-bridge DLEs.

The DLPDU class and first delocalized address are used to determine the other DL-processes, if any, which should receive the DLPDU before the DLPDU record is deleted.

1) If the first delocalized DL-address is an active DLSAP-address of the DLE, the DLPDU record is passed to the upper-level process that handles DLPDUs received at that DLSAP.
2) If the first delocalized DL-address is an active group DL-address of the DLE, for each local DLSAP which is a member of the group, the DLPDU record is passed to the upper-level process that handles DLPDUs received at that DLSAP.

3) If the first delocalized DL-address is an active DLCEP-address of the DLE, for each local DLCEP which is associated with that DLCEP-address, the DLPDU record is passed to the upper-level process associated with that DLCEP as follows:
   i) If the DLE has a publisher DLCEP for that DLCEP-address, and the DLPDU class is one which should be forwarded to publishers, then the DLPDU record is passed to the process associated with that publisher DLCEP.
   ii) If the DLE has one or more subscriber DLCEPs for that DLCEP-address, and the DLPDU class is one that should be forwarded to subscribers, then for each of those subscriber DLCEPs, the DLPDU record is passed to the process associated with that subscriber DLCEP.
4) If
   i) the received DLPDU contained one or more explicit DL-addresses, and
   ii) all of those addresses were LONG with non-zero link designators, and
   iii) the link-designator of the first delocalized DL-address is in the forwarding database of the DLE,

   then the DLPDU record is passed to the lower-level transmission process of each port associated with that link designator in the forwarding database that is in the FORWARDING state, except the port on which the DLPDU was received.

Lower-level port-specific transmission processes retransmit forwarded DLPDUs on their associated links. The actual octets of the received DLPDU, which were retained in fields c) and e) of the received DLPDU record, are used to retransmit the DLPDU. If the FINAL bit of the received frame control octet needs to be complemented before transmission, then the received FCS is modified as specified in 5.2.5.3 before that FCS is transmitted. A new FCS is never calculated for a forwarded DLPDU.

Receive statistics are kept as specified in 10.5.6.3.

NOTE Alternatives 1) through 3) are identical for, and apply to, both bridge and non-bridge DLEs. However, the set of DL-addresses recognized by bridge DLEs as relating to the bridge itself is generally larger than the similar set for non-bridge DLEs. Each bridge port is required to recognize the DL-addresses of all of that bridge's ports, regardless of the port states of those ports. Without such recognition, it would not be possible for System Management to configure a bridge port other than the one on which the DLPDU was received.

10.4.1.6.3 Model of local origination of a DLPDU

A higher-level process creates a locally originated DLPDU record consisting of

a) the DLPDU class;
b) the delocalized DL-address(es) which are to be specified, explicitly or implicitly, in localized or delocalized form, by the DLPDU's header;
c) the octets representing the *body* of the DLPDU – the DL-parameters, or DLS-user data, or both.

The DLPDU class and first delocalized DL-address are used to determine the other DL-processes that should receive the DLPDU before the DLPDU record is deleted.

1) If the first delocalized DL-address is an active DLSAP-address of the DLE, the DLPDU record is passed to the upper-level process that handles DLPDUs received at that DLSAP.
2) If the first delocalized DL-address is an active group DL-address of the DLE, for each local DLSAP which is a member of the group and which is not the sending DLSAP, the DLPDU record is passed to the upper-level process which handles DLPDUs received at that DLSAP.
3) If the first delocalized DL-address is an active DLCEP-address of the DLE, for each local DLCEP which is associated with that DLCEP-address and which is not the sending DLCEP, the DLPDU record is passed to the upper-level process associated with that DLCEP as follows:

a) If the DLE has a publisher DLCEP for that DLCEP-address, and the DLPDU class is one which should be forwarded to publishers, then the DLPDU record is passed to the process associated with that publisher DLCEP.

b) If the DLE has one or more subscriber DLCEPs for that DLCEP-address, and the DLPDU class is one which should be forwarded to subscribers, then for each of those subscriber DLCEPs, the DLPDU record is passed to the process associated with that subscriber DLCEP.

4) If the link-designator of the first delocalized DL-address is in the forwarding database of the DLE, then the DLPDU record is passed to the lower-level transmission process of each port associated with that link designator in the forwarding database.

Lower-level port-specific transmission processes transmit locally originated DLPDUs on their associated links by

– building a link-appropriate header from fields 1.a) and 1.b), possibly localizing and omitting all or part of the DL-addresses from the DLPDU in the process,
– appending the body from field 1.c), and
– computing the FCS over the DLPDU header and body as transmitted.

NOTE Alternatives 1) through 3) are identical for, and apply to, both bridge and non-bridge DLEs. Alternative 4) can always occur in a non-bridge DLE; the link designator can always be presumed to be in the implicit "forwarding database."

10.4.1.6.4 Model of subscribing to one DLCEP and republishing on one or more other DLCEPs

Each bridge may function as a republisher of published data. The entities that model this operation are:

a) the subscriber DLCEP, responsible for receiving published DLPDUs, detecting duplicate transmissions of sequence-numbered DLPDUs, and buffering the contained DLSDU from a non-duplicated transmission in a DL-buffer, together with the DLSDU's timeliness status.

b) the DL-buffer that holds the last DLSDU received as a subscriber, together with its timeliness status, and makes it available as the source DLSDU for republishing.

c) the publisher DLCEP, responsible for transmitting the DLSDU and its timeliness status on its new DLC, updating any sequence number associated with the transmission only when the buffer has been updated since the prior transmission on the DLCEP.

10.4.1.6.5 DLPDU transmission

The lower-level DLL process associated with each bridge port transmits DLPDUs on that port in accordance with the associated link's schedule.

DLPDUs which were received from another port and forwarded are retransmitted without alteration, except perhaps for complementation of the FINAL bit in the DLPDU's first octet (the frame control octet) and corresponding complementation of specific bits in the forwarded DLPDU's FCS, where the FCS complementation pattern is a function only of the number of octets in the DLPDU. (See 5.2.5.3.)

DLPDUs that are locally originated within the DLE are submitted by the upper-level DLL entity directly to the forwarding process for transmission on the port(s) associated with the first address of the DLPDU. DLPDUs that address the sending bridge as an end-station are also delivered to the upper-level DLL entity for processing.

10.4.2 Adaptive bridge sub-protocol and elements of procedure

The ISO/IEC bridge protocol, ISO/IEC 10038, specifies management and operation of the MAC-relay entities of a bridged local area network. This subclause specifies additions and modifications to ISO/IEC 10038:1993 which transform it into a bridge protocol suitable for use with fieldbus DL-relay entities.

In the following text references to clauses of this standard are preceded by [FB], references to clauses of ISO/IEC 10038:1993 are preceded by [IL].

10.4.2.1 Global changes

Throughout [IL] replace all references to

a) “MAC” *by* “DL” *or* “DL-”, *as appropriate*;
b) “Media Access Control” *by* “Data Link”;
c) “Sublayer” *by* “layer”;
d) “LAN” *by* “link”;
e) “Bridged Local Area Network” *by* “extended link”;
f) “user priority” *with* “DLL-priority”;
g) “frame” *with* “DLPDU”;
h) “LLC” *with* “upper DLL”;
i) “section” *with* “clause”.

Throughout [IL] change numeric references to subclauses of [IL] by preceding the reference, if appropriate, with the subclause identification “12.4.2.” to provide a means of discriminating between cross-references within this subclause [IL] and references within the subclause to this part of IEC 61158-4 [FB].

Make all other style, format and nomenclature changes required by IEC Directives, and renumber or reletter the items within each subclause as appropriate after the following changes.

10.4.2.2 Changes to [IL/1]: Introduction

Modify the following indicated paragraphs of [IL/1.5] as shown:

(1) *delete paragraph* (b)

(4) *delete this paragraph*

(7) *replace* “may” *with* “shall”

(9) *replace paragraph with* “shall use 32-bit DL-addresses and their 48-bit MAC-address representations as specified in [IL/3.1.2] and described in [FB/4.3.4].

(11) *replace* “may” *with* “shall” *and delete* “optional”

(12) *delete paragraphs* (a) *and* (b)

(16) *replace* “may” *with* “shall”

(17) *replace* “may” *with* “shall”

(19) *replace* “may” *with* “shall”

(20) *replace paragraph with* “shall support remote management, as specified in [IL/7] or by alternate means”

Delete [IL/1.6 and 1.7].

10.4.2.3 Changes to [IL/2]: Support of the DL-Service

Replace the first paragraph of [IL/2.1] with “The DL-service provided to end stations attached to an extended link is supported by the bridges in that extended link. Bridges support all of the DL-services which involve multi-end-station communication or scheduling.”

Delete "Destination" *from [IL/2.21)].*

Replace paragraph [IL/2.23)] with "the DL-addresses of end-stations are not restricted by the topology and configuration of the extended link, except as specified in [FB/4.3.2.1] for link-designator values between 1000_{16} and $FEFF_{16}$, inclusive."

Delete paragraph [IL/2.3.22c)].

Replace paragraph [IL/2.3.3] with "The operation of bridges is unlikely to misorder DLPDUs transmitted with the same DL-priority. Such misordering can occur

a) under conditions where DL-redundancy is employed to provide multiple simultaneously-usable paths between the two end-stations, or
b) immediately subsequent to reconfiguration of the DL-path between the two end stations.

NOTE Such reconfiguration can occur only as the result of DL-management action or during recovery from bridge or Ph-layer entity failures.

Replace the second paragraph of [IL/2.3.7] with "For DLPDUs relayed between links, the bridge shall modify the FC octet of each DLPDU and make compensatory modifications to the FCS of that DLPDU as specified in [5.2.5.3]."

Insert after [IL/2.3.10] "**2.3.11 Time synchronization**. The bridges establish and preserve the shared sense of DL-time throughout the extended link."

Replace [IL/2.4] with "**2.4 Forwarding of DLPDUs**

A bridge receiving a DLPDU examines the DLPDU's frame control octet and first address, from which the bridge determines whether and to where the DLPDU should be forwarded.

A bridge forwards a CA, CD, DC, DT, EC, ED or RC DLPDU, together with its received FCS, toward those links which have DLEs which should receive the DLPDU's first DL-address. No other forwarding of DLPDUs occurs.

If the received DLPDU is a CA, CD or ED DLPDU, then the bridge replies immediately with a SR DLPDU on the link from which the CA, CD or ED DLPDU was received."

Delete [IL/2.5].

10.4.2.4 Changes to [IL/3]: Principles of operation

Insert before [IL/3.1 (2)]:

"(2) Relay of the shared sense of DL-time."

Modify the indicated paragraphs of [IL/3.1.1] as shown:

(3) *Replace with* "DLPDU discard if the DLPDU type is not CA, CD, DC, DT, EC, ED, RC or SR; and DLPDU discard if the DLPDU is a detected duplicate of a prior CA, CD or ED DLPDU, presumably caused by loss of a prior immediate retry (see 6.4.4, 6.5.4 and 6.6.4)."

(3+) *Insert before (4)* "Possible DLPDU transformation or discard if the DLPDU type is DT or SR (see 6.7.4.3c) and 6.8.4.3b))."

(5) Delete this paragraph.

(8) Delete this paragraph.

(9) *Replace with* "modification of a frame control octet, with compensatory modification of the DLPDU's FCS."

(9+) *Insert before (10)* "FCS calculation on transformed DT or SR DLPDUs (both of which become DT DLPDUs)."

Append to [IL/3.1.2 (3)] "… and queries to other bridges and end-stations on the local link."

Replace [IL/3.2] with "**3.2 Bridge architecture**. Each bridge port receives and transmits DLPDUs to and from the link to which it is attached. The DL-relay-entity handles the functions of relaying DLPDUs between bridge ports, filtering DLPDUs, and learning filtering information."

"The bridge protocol entity handles calculation and configuration of the extended link topology. The bridge protocol entity and other higher-layer protocol users, such as DL-management, make use of higher-level DLL services, which are provided in a port-independent manner.

Figures 3-2 and 3-3 of [IL] illustrate a bridge and its ports, and the architecture of the bridge for a bridge with two ports. A bridge may have more than two ports. Where Fractional Duty Cycle (FDC) nodes exist on a link, it is also meaningful for a bridge to have just a single port, since a one-port bridge can supply the DLPDU store-and-forward services needed to permit communication with or among FDC nodes."

Delete the first paragraph of [IL/3.3].

Replace Figures 3-2 and 3-3 of [IL] with Figure 21 *and* Figure 22 *to replace the term* "LAN" *with* "local link" *and to delete the terms* "LLC entities" *and* "MAC service interfaces":

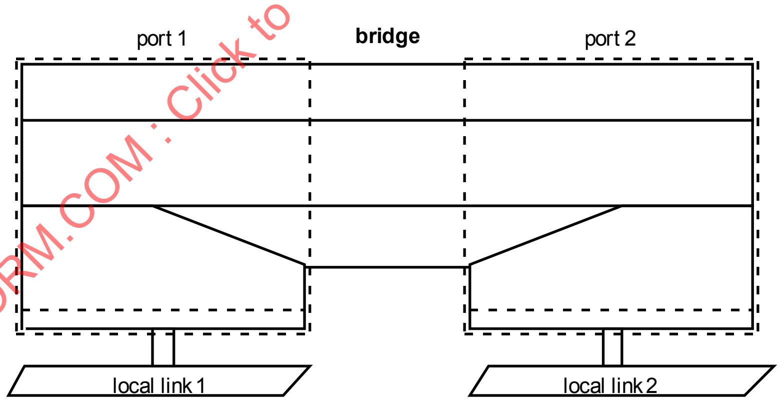


Figure 21 – Replacement for [IL] Fig 3-2 Bridge ports

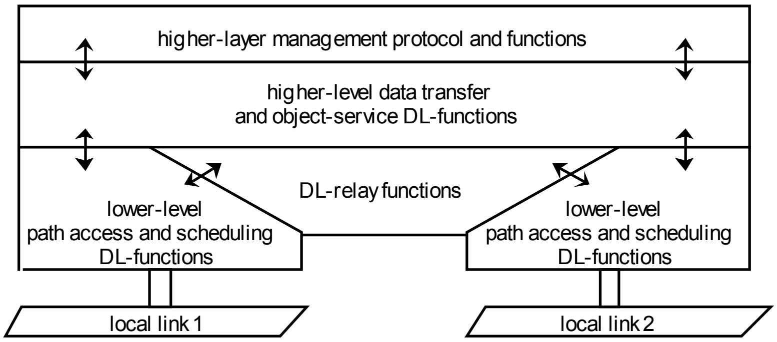


Figure 22 – Replacement for [IL] Fig 3-3 Bridge architecture

Delete the paragraph of [IL/3.3] which begins “Each Bridge Port shall support …”.

Replace the title of figure 3.4 of [IL] with “**Relaying DLPDUs**”.

Replace the title of figure 3.5 of [IL] with “**Observation of link traffic**”.

Delete the dotted blocks “LLC” *from figure 3-6 of [IL].*

Replace [IL/3.5] with “**3.5 DLPDU reception**. The lower-level DLL entity associated with each bridge port examines all DLPDUs received from the link to which it is attached. DLPDUs that are detected to be in error are discarded; this includes DLPDUs whose FCS is in error. DLPDUs that are detected to be duplicates of prior received DLPDUs, due to immediate retry by the current holder of a scheduled or delegated token, are also discarded after generation of the required immediate reply. All other DLPDUs shall be submitted to the learning process.

Received DC, EC and RC DLPDUs [IL/2.4] shall be submitted to the forwarding process. Received CA, CD and ED DLPDUs [IL/2.4] shall be submitted to the forwarding process except when they are discarded as duplicates of prior received DLPDUs. Received DT DLPDUs [IL/2.4] containing at least one explicit DL-address shall be submitted to the forwarding process.

Other DLPDU types shall not be relayed by the bridge. However, the bridge may be required to transform a received SR DLPDU, or a received DT DLPDU without explicit DL-addresses and with null user data, to a DT DLPDU with one explicit DL-address and null user data, and forward that transformed DLPDU as if it had just been received (see 6.7.4.3c) and 6.8.4.3b)).

DLPDUs addressed to the bridge as an end station shall be submitted to upper-DLL processing except when they are discarded as duplicates of prior received DLPDUs.

Locally-originated DLPDUs shall be submitted directly to the forwarding process by the upper-DLL functions.”

Replace [IL/3.6] with “**3.6 DLPDU transmission**. The lower-level DLL entity associated with each bridge port transmits DLPDUs submitted to it by higher DLL functions. Relayed, transformed and locally originated DLPDUs are submitted for transmission by the forwarding process.”

Replace [IL/3.7.1 (3) (a)] with “The DLPDU specifies a destination DL-address and the Filtering Database indicates that DLPDUs with this value of the destination address should be forwarded through the transmission Port (as would happen, for example, if the destination address specified a local link accessible through that transmission Port).”

Replace [IL/3.7.1 (3) (b)] with “The DLPDU specifies only a source DL-address and the Filtering Database indicates that source-only DLPDUs with this value of the source address should be forwarded through the transmission Port (as would happen, for example, if expected receivers of DLPDUs with that source address were attached to a local link accessible through that transmission Port).”

Delete [IL/3.7.2].

Delete [IL/3.7.4].

Replace [IL/3.7.5] with “3.7.5 Modification of DLPDUs. When forwarding a DLPDU, it is sometimes necessary for a bridge to alter the final-use subfield of a DLPDU’s frame control field. Specifically, the bridge may have to complement the value of the final-use subfield of the frame control field. This modification, with a corresponding FCS modification, shall be performed as specified in [5.2.5.3].”

Delete Tables 3-1 and 3-2 of [IL].

Replace the first paragraph of [IL/3.8] with “**3.8 The learning process**. The Learning Process observes the source DL-addresses, and in some cases the destination DL-addresses and CE-parameters, of DLPDUs received on each Port and updates the Filtering Database conditionally on the state of the receiving Port."

Replace the third paragraph of [IL/3.8] and its subparagraphs with “The Learning Process may deduce the path through the extended link to particular end stations, by inspection of the source DL-address fields of CA, CD, DC, DT, ED and RC DLPDUs, and by inspection of the EC-parameters and all of the DL-address fields of received EC DLPDUs.

(1) For the source DL-address of a CA, CD, DC, DT, ED or RC DLPDU, and for each of the two source DL-addresses of an EC DLPDU, if

(a) the link-designator component of that DL-address is in the range 1000_{16} to V(ML), inclusive, where V(ML) is as defined in [FB/4.7.6.1];

(b) the Port from which the DLPDU was received was in a state that allows learning [IL/4.4];

(c) a static entry [IL/3.9, 3.9.1] for the link-designator component of that source DL-address does not already exist; and

(d) the resulting number of entries would not exceed the capacity of the Filtering Database;

then the Learning Process shall create or update a dynamic entry [IL/3.9, 3.9.2] in the Filtering Database for the link-designator component of that source DL-address, associating the Port on which the DLPDU was received with that link-designator component.

(2) For the source DL-address of a CA, CD, DC, DT, ED or RC DLPDU, and for each of the two source DL-addresses of an EC DLPDU, if

(a) the link-designator component [FB/4.3.1] of the source DL-address is in the range 0080_{16} to $0FFF_{16}$, inclusive;

(b) the Port from which the DLPDU was received was in a state that allows learning [IL/4.4];

(c) a static entry [IL/3.9, 3.9.1] for that source DL-address does not already exist; and

(d) the resulting number of entries would not exceed the capacity of the Filtering Database;

“then the Learning Process shall create or update a dynamic entry [IL/3.9, 3.9.2] in the Filtering Database for that source DL-address, associating the Port on which the DLPDU was received with the DL-address.

(3) For an EC DLPDU, if

(a) the sender’s-DLCEP-class parameter of the DLPDU’s EC-parameters specifies PUBLISHER [FB/7.1a)4)];

(b) the Port from which the DLPDU was received was in a state that allows learning [IL/4.4];

(c) a static entry [IL/3.9, 3.9.1] for the DLPDU’s first source DL-address does not already exist; and

(d) the resulting number of entries would not exceed the capacity of the Filtering Database;

then the Learning Process shall create or update a dynamic Publisher entry [IL/3.9, 3.9.2] in the Filtering Database for that source DL-address, associating the Ports to which the DLPDU will be forwarded to the subscriber ports of that DL-address.

(4) For an EC DLPDU, if

(a) the sender’s-DLCEP-class parameter of the DLPDU’s EC-parameters specifies SUBSCRIBER [FB/7.1a)4)];

(b) the Port from which the DLPDU was received was in a state that allows learning [IL/4.4];

(c) a static entry [IL/3.9, 3.9.1] for the DLPDU’s destination DL-address does not already exist; and

(d) the resulting number of entries would not exceed the capacity of the Filtering Database;

then the Learning Process shall create or update a dynamic Publisher entry [IL/3.9, 3.9.2] in the Filtering Database for that destination DL-address, associating the Port on which the DLPDU was received as one of the subscriber ports of that DL-address.

The Learning Process may also query other DLEs on the local link by sending DL-address list query SPDUs [FB/Clause 11 to the DL-support functions of those DLEs and using the resultant DL-address list reply SPDUs [FB/Clause 11] as a basis for adding or removing entries from the Filtering Database.”

Replace the second to last paragraph of [IL/3.8] with “If the Filtering Database is already filled up to its capacity, but a new entry would otherwise be made, then DL-management shall be informed of the Filtering Database overflow, and the new entry shall not be made.”

Replace the last sentence of the first paragraph of [IL/3.9] with "It supports queries by the Forwarding Process as to whether DLPDUs with given values of the destination DL-address, or of the source DL-address in DLPDUs specifying only an explicit source DL-address, should be forwarded to a given Port."

Replace the second paragraph of [IL/3.9] with "The Filtering Database shall be capable of containing both static entries [IL/3.9.1] and dynamic entries [IL/3.9.2], for both full DL-addresses and DL-address link-designator components [FB/4.3.1]).

Static and dynamic entries shall not both exist for the same full DL-address; a dynamic entry shall not be created if a corresponding static entry for the same full DL-address already exists; and creation of a static entry shall cause the removal of a corresponding dynamic entry for the same full DL-address if one exists.

Static and dynamic entries shall not both exist for the same DL-address link-designator component; a dynamic entry shall not be created if a corresponding static entry for the same partial DL-address already exists; and creation of a static entry shall cause the removal of a corresponding dynamic entry for the same partial DL-address if one exists.

Each entry in the Filtering Database shall be of one of five classes:

(a) A DL-address link-designator component entry, which specifies the default single port to which a DLPDU with a destination DL-address containing this link-designator component should be forwarded when the Filtering Database does not contain a specific entry for the destination DL-address specified in the DLPDU.

(b) A group DL-address entry, which specifies, separately for each receiving port, the set of ports to which a DLPDU whose destination DL-address equals the specified DL-address should be forwarded.

(c) A DLSAP-address entry, which specifies the single port to which a DLPDU whose destination DL-address equals the specified DL-address should be forwarded.

(d) A peer DLCEP-address entry, which specifies the single port to which a DLPDU whose destination DL-address equals the specified DL-address should be forwarded.

(e) A publisher DLCEP-address entry, which specifies

(i) the single port to which a CA, CD, DT, EC or ED DLPDU whose destination DL-address equals the specified DL-address should be forwarded; and

(ii) the set of ports, all different from (i), to which a DC, DT, EC or RC DLPDU which contains no destination DL-address, and which specifies this DL-address as an explicit source DL-address, should be forwarded."

Replace the second and following paragraphs of [IL/3.9.1] with "A static entry may specify either a full DL-address or a link-designator component of a DL-address."

Replace the last five paragraphs of [IL/3.9.2], beginning "Dynamic entries specify …"*, with* "A dynamic entry may specify any full DL-address or DL-address link-designator component."

Replace all of [IL/3.12] with "**3.12 Addressing**." *That is, delete all text following the title up to but not including [IL/3.12.1].*

In the second paragraph of [IL/3.12.3], replace "LLC Entities" *with* "DLEs"*, and replace* "Standard LSAP … users of LLC" *with* "DL-bridge functions of DLEs (see 4.3)."

Replace the third paragraph of [IL/3.12.3] with "Each DL-UNITDATA request primitive causes the transmission of a DLPDU which conveys the BPDU in its user data field as a complete DLSDU."

Delete the fourth paragraph of [IL/3.12.3].

In the fifth paragraph of [IL/3.12.3], delete ", within the scope of the LSAP Assignment".

Delete the last sentence of the seventh paragraph of [IL/3.12.3], which states "This group address … are transmitted."

Delete the eighth paragraph of [IL/3.12.3].

Replace the last paragraph of [IL/3.12.4] with "[FB/4.3] specifies standard DL-addresses for the DL-bridge functions of bridge DLEs."

Delete the second and third paragraphs of [IL/3.12.5].

Delete [IL/3.12.6] and Tables 3-4, 3-5, 3-6 and 3-7 of [IL].

10.4.2.5 Changes to [IL/ 4]: The spanning tree algorithm and protocol

In the second paragraph of [IL/4.3.4], change "must" *to* "should".

In Tables 4-2 and 4-3, replace the first line after each table with "All times are in seconds when the local link data rate is 1 Mbit/s or greater; otherwise all times are in minutes."

In Table 4-3, replace the line

Bridge Forward Delay	15,0	—	4,0 – 30,0

with

Bridge Forward Delay	0,0	—	0,0 – 30,0

Insert before [IL/4.6.11] "**4.6.10.3.5** The port's DLE shall attempt to become the LAS for the local link to which it is attached."

10.4.2.6 Changes to [IL/5]: Encoding of bridge protocol data units

Replace the third paragraph of [IL/5.2.5] with "The third most significant octet is assigned the value of the initial octet of the 48-bit Bridge Address. The fourth to eighth octets are similarly assigned the values of the second to the sixth octets of the 48-bit Bridge Address, respectively."

Delete the fourth paragraph of [IL/5.2.5].

10.4.2.7 Changes to Clause 6: Bridge management

Replace [IL/6.35)] with "(5) IA5 String, for all text strings."

Replace [IL/6.7.5.1.2 (2)] with "(2) Address – full DL-address or DL-address link-designator component of the entry."

Replace [IL/6.7.5.1.2 (3)] with "(3) Entry class; associated port or port maps:

"(a) DL-address link-designator component – the default single port to which a DLPDU with a destination DL-address containing this link-designator component should be forwarded when the Filtering Database does not contain a specific entry for the destination DL-address specified in the DLPDU

“(b) group-DL-address – an array which specifies, separately for each receiving port, the set of ports to which a DLPDU specifying this DL-address as a destination DL-address should be forwarded.

“(c) DLSAP-address – the single port to which a DLPDU specifying this DL-address as a destination DL-address should be forwarded.

“(d) peer DLCEP-address – the single port to which a DLPDU specifying this DL-address as a destination DL-address should be forwarded.

“(e) publisher-DLCEP-address —

“(i) the single port to which a DLPDU specifying this DL-address as a destination DL-address should be forwarded; and

“(ii) the set of ports, all different from (i), to which a source-only DC, DT, EC or RC DLPDU specifying this DL-address as an explicit source DL-address should be forwarded.”

Replace [IL/6.7.5.2.2 (2)] with “(2) Address – full DL-address or DL-address link-designator component of the desired entry.”

Replace [/IL/6.7.5.3.2 (2)] with “(2) Address – full DL-address or DL-address link-designator component of the desired entry.”

Replace [/IL/6.7.5.3.3 (1)] with “(1) Address – full DL-address or DL-address link-designator component of the desired entry.”

Replace [IL/6.7.5.3.3 (3)] with “(3) Entry class; associated port or port maps; as specified in [IL6.7.5.1.2 (3)].”

Replace [IL/6.7.5.4.3 (3) (a)] with “(a) Address – full DL-address or DL-address link-designator component of the desired entry.”

Replace [IL/6.7.5.4.3 (3) (c)] with “(c) Entry class; associated port or port maps; as specified in [IL/6.7.5.1.2 (3)].”

10.4.2.8 Changes to[IL/7]: Management protocol

Replace the entire clause after the title with:

“The fieldbus management protocol is outside the scope of the IEC 61158 set of standards. Some DL-management capabilities are specified in 9.3 and 10.4.3.1.”

10.4.2.9 Changes to [IL/annex A]: PICS proforma

Delete items 1a, 1b and 2a.

Reword item 2c to read “Are CA, CD, DC, DT, EC, ED, RC and SR DLPDUs the only types of DLPDU relayed, possibly after transformation as specified in 6.7.4.3c) and 6.8.4.3b)?”

Delete items 2d, 2f, 2g, 2n, 2o, 2q, 2r, 2s, 4, 4a, 4b, and 5b.

Reword item 5g to read "Does each static or dynamic entry specify either a full DL-address or the link-designator component of a DL-address?"

Reword item 5i to read "Does each entry which specifies a link-designator component of a DL-address specify a port number?"

Renumber old item 5j to 5p.

Add new item 5j "Does each entry which specifies a group-DL-address specify an array of sets of ports, with one set for each receiving port?"

Add new item 5k "Does each entry which specifies a DLSAP-address specify a port number?"

Add new item 5m "Does each entry which specifies a peer DLCEP-address specify a port number?"

Add item 5n "Does each entry which specifies a publisher-DLCEP-address specify

"a) a port number; and

"b) separately, a set of ports?"

Reword item 7d to read "Can static entries be made for link-designators in the range 1000_{16} to $FEFF_{16}$?"

Replace item 7f with "Can static entries be made for DLSAP- addresses?"

Renumber old item 7g to 7j.

Add new item 7g "Can static entries be made for peer DLCEP-addresses?"

Add new item 7h "Can static entries be made for publisher DLCEP-addresses?"

Delete items 9a, 9b, 9c, 10b, 10e, 10f, 11a, 11b, 11d, 11e, 11f, and 12a.

Replace item 10c with "Are all BPDUs addressed to the DL-bridge functions of bridge DLEs?"

10.4.2.10 Delete [IL/annex C] and [IL/annex D], both of which relate to source-routing

10.4.3 Non-adaptive bridge sub-protocol and elements of procedure

Non-adaptive bridges shall implement the subset of bridge functionality described in 10.4.1. The means of configuring the required forwarding and republishing databases is a local issue.

10.4.3.1 Bridge annunciation protocol and BPDU structure

The bridge management entity is responsible for implementing the Bridge Annunciation protocol, which is a minimal compatible subset of the Spanning Tree protocol of 10.4.1.5. This subset serves to announce the non-adaptive bridge's presence to adaptive bridges conforming to 10.4.1.5 and to cause the latter bridges to defer to non-adaptive bridges in establishing the acyclic graph which is the spanning tree.

The bridge management entity achieves this by transmitting BPDUs (Bridge Protocol Data Units) as the DLS-user data of connectionless DLPDUs, addressed to the standard group DL-address for bridges on the local link (0x0103) as specified in Table 6, with the same format as in 10.4.1.5. The source DL-address of the DLPDUs is formed from the standard selector 0x01 for the bridge functions of a DLE, as specified in Table 7.

a) It sends a Topology Change Notification BPDU at startup from each bridge port that is configured to be in the FORWARDING state, and similarly after reconfiguration, from each bridge port that has just been placed in the FORWARDING state. Table 131 shows the coding of a Topology Change Notification BPDU.

Table 131 – Topology change notification BPDU format

Octets	1	2	3	4
01-04	0x00	0x00	0x00	0x80

b) It sends a Configuration BPDU on a port each time it receives a Topology Change Notification BPDU on that port, and each time it receives a Configuration BPDU on that port with a non-zero value in any of octets 6 through 13. Table 132 shows the coding of a Configuration BPDU, with non-zero values in octets 27 – 35, and with octet 27 specifying the port number of the bridge port from which the Configuration BPDU is being transmitted.

Table 132 – Configuration BPDU format

Octets	4N+1	4N+2	4N+3	4N+4
01-04	0x00	0x00	0x00	0x00
05-08	0x00	0x00	0x00	0x00
09-12	0x00	0x00	0x00	0x00
13-16	0x00	0x00	0x00	0x00
17-20	0x00	0x00	0x00	0x00
21-24	0x00	0x00	0x00	0x00
25-28	0x00	0x00	port #	0xFF
29-32	0xFF	0xFF	0xFF	0xFF
33-35	0xFF	0xFF	0xFF	

10.5 DL-management-information

10.5.1 General

This subclause enumerates the set of DL-parameters, defined as variables in 4.7, which need to be preconfigured before proper DLE operation is possible. It also summarizes DLE actions which can be requested through DL-management.

10.5.2 DLE configuration parameters

The following parameters, defined as variables in 4.7, are required to support the three classes of DLE operation.

10.5.2.1 Node-specific DL-configuration parameters

The following parameters shall be assigned the appropriate values for the DLE before the DLE is permitted to transmit any DLPDU on the local link:

10.5.3 Additional node-independent DL-configuration parameters

The following DL-parameters shall be assigned the then-current values for the local link before the DLE is permitted to transmit any DLPDU on the local link:

NOTE Each PN DLPDU conveys the values of the following parameters, plus those of critical PhL parameters. Thus a DLE which receives a PN DLPDU is able to configure itself, and its associated PhE, so that the DLE and associated PhE can reply to that or a subsequent PN DLPDU, starting the process of integrating the replying DLE into an already-functioning network.

10.5.3.1 V(MRD) maximum-response-delay, defined in 4.7.1.3;

The following DL-parameters shall be assigned the then-current values for the local link before the DLE is permitted to transmit any DLPDU other than a PR DLPDU on the local link:

NOTE A single PR DLPDU is able to convey a probe-response SPDU to the local LAS, which will cause the current local-link values of the following parameters to be returned in a DL-node-activation SPDU (see Clause 11).

10.5.3.2 V(MRC) maximum-retry-count (see 4.7.1.5);

10.5.3.3 V(NDL) network-DLPDU-lifetime (see 4.7.1.7);

10.5.3.4 V(MEP) MAC-embedding-prefix, defined in 4.7.1.10;

10.5.4 Additional node-independent DL-configuration parameters for link-master class DLEs

The following DL-parameters shall be assigned the then-current values for the local link before the DLE is permitted to function as a DLE of the Link-master (see 4.6.2) or bridge (see 4.6.3) class (that is, as a potential LAS ready to assume the LAS role), and transmit on the local link:

10.5.4.1 V(MST) maximum-scheduled-traffic, defined in 4.7.5.5;

10.5.4.2 V(DTHT) default-token-holding-time, defined in 4.7.5.8;

10.5.4.3 V(TTRT) target-token-rotation-time, defined in 4.7.5.11;

10.5.4.4 V(NUN) number-of-consecutive-unpolled-nodes ids, defined in 4.7.5.16;

10.5.4.5 V(TDP) time-distribution-period, defined in 4.7.5.18;

10.5.4.6 V(LDDP) LAS-data-base-distribution-period, defined in 4.7.5.20.

10.5.5 Additional node-independent DL-configuration parameters for bridge class DLEs

Since every bridge class DLE has link master capability, 10.5.4 also applies to bridge class DLEs.

The following DL-parameters shall be assigned the then-current values for the local link before the DLE is permitted to function as a DLE of the bridge (see 4.6.3) class and transmit on the local link:

10.5.5.1 – all of the additional parameters defined in 10.5.5;

10.5.5.2 – all of the configuration parameters defined in ISO/IEC 10038.

10.5.5.3 Node-independent Ph-configuration parameters required for minimal DL-communication

This subclause enumerates the minimal set of Ph-parameters, defined in IEC 61158-2, 6.2.2, which need to be preconfigured before proper PhE operation is possible.

NOTE 1 These Ph-parameters, defined in IEC 61158-2, Table 4 are included to make it possible for a device, newly connected to a fieldbus local link, to learn the minimal information required for successful communication with that local link's LAS, and thereby request any additional needed PhL and DLL configuration information, after which remote management becomes possible.

NOTE 2 Although this measure is named "post-transmission-gap-extension" in IEC 61158-2, it is a measure of the amount of observable non-transmission required between any two transmissions on the local medium, as observed from any point on that medium, whether the two transmissions are from the same source PhE or from two different source PhEs. Thus it could just as correctly be named "pre-transmission-gap-extension", or even more correctly "inter-transmission-gap-extension".

10.5.5.4 V(PhIS) maximum inter-channel signal skew.

10.5.6 DLE-collected fault-management data

NOTE DLE-collected fault-management data are designed to provide the maximum amount of information for network and end-system fault management while imposing the least performance burden on real implementations.

10.5.6.1 Required statistical measures

Each DLE should keep a minimum set of statistical measures to facilitate the identification of communications errors which render the local link inoperable, and the sources of those errors, where that information is available. These statistical measures provide a partial basis for network fault management.

NOTE 1 These statistical measures may be read through higher-layer network-management protocols.

In general, these statistical measures take the form of event counters, where a network transmission or reception event results in the incrementation of a counter. Conceptually, these counters are of unbounded length, are reset only on DLE activation or DLE reset, and record the number of events of the appropriate type which have occurred since the DLE was last activated or reset.

In general, counters which record error events need only be two octets in length, while counters that record events expected during normal error-free operation should be longer to permit infrequent loss-free readout by remote DL-management. The remainder of this subclause specifies both the minimum required size, and where different the recommended size, of each counter.

These counters shall be reset only upon DLE activation or DLE reset; they shall not be reset by any other local or remote means.

NOTE 2 This restriction on resetting the counters is necessary if they are to be meaningful in a multi-manager environment.

10.5.6.1.1 Transmission-related statistical measures

a) The total number of jabber (streaming) faults which were detected during transmission, if such faults are detectable separately from other modem or MAU faults, or zero if such faults are not separable. This counter shall be at least two octets in length.

b) The total number of DLPDUs which were transmitted as immediate retries of the immediately prior transaction (see 3.3.32). This includes DLPDUs retransmitted by the DLE when it is functioning in an initiator role, and DLPDUs retransmitted by the DLE when it is functioning in a responder role. This counter shall be at least two octets in length.

c) The total number of DLPDUs which were transmitted by the DLE, but excluding and not counting those DLPDUs counted in category b). This counter shall be at least two octets in length, and should be at least four octets in length.

10.5.6.1.2 Reception-related statistical measures

The following two statistical measures shall be kept

– separately for each PhL-medium attached to the DLE, if an implementation so permits, or

– aggregated for all of the PhL-media attached to the DLE.

NOTE 1 These statistical measures facilitate identification of PhE-related problems.

a) The total number of modem-internal or MAU-internal faults, which were detected during transmission or reception, plus the total number of jabber (streaming) faults not reported in category 10.5.6.1.1a). This counter shall be at least two octets in length.

   NOTE Implementations which do not distinguish jabber (streaming) faults from modem-internal or MAU-internal faults will count all such faults in this category.

b) The total number of **END-OF-ACTIVITY** indication events (see 4.4.1.2) at the PhE interface for which there was not an immediately preceding **END-OF-DATA** indication event. This counter shall be at least two octets in length.

NOTE This is the number of DLPDUs and noise intervals for which reception commenced, but in which a partially received DLPDU was discarded due to detected framing or coding errors.

When a DLE is implemented so that it does not actually receive its own DLPDUs when it is transmitting, but processes them as if they had been received, then the following counters shall not be incremented as a result of a transmission by the DLE.

NOTE 2 This is equivalent to saying that the DLE operates in a physically half-duplex mode.

The following additional statistical measures shall be kept for each DLE, where that information is available:

NOTE 3 These statistical measures facilitate identification of link-related or path-related, but not PhL-media-related, problems.

c) The total number of partially received DLPDUs which were discarded after examination of the DLPDU's frame control field and first address subfield, if any, and for which the associated FCS was not checked for correctness. This counter shall be at least two octets in length, and should be at least four octets in length.

NOTE If the receiving DLE takes any action based on the received DLPDU, such as an LAS DLE checking the frame control octet's final-use subfield, or any DLE noting the destination address of a CA, CD or ED DLPDU to enable reception of the reply DLPDU, then the receiving DLE necessarily has to check the correctness of the FCS before acting on the information in the DLPDU. In such a case, this specific statistical measure cannot, and does not, apply.

d) The total number of completely-received DLPDUs which were discarded due to a detected incorrect FCS (see 5.2.5.2). This counter shall be at least two octets in length.

e) The total number of DLPDUs which were discarded as inferred duplicates due to immediate retry of an immediately prior transaction. This counter shall be at least two octets in length.

f) The total number of received DLPDUs, with a correct FCS, which were not counted in categories a) to e). This counter shall be at least two octets in length, and should be at least four octets in length.

g) The total number of occurrences which the DLE has detected, of a timeout of one of the DLE's timers which is prescaled by the local link's slot-time, V(ST). This includes
   1) timeouts as initiator of a two-phase transaction requiring an immediate reply;
   2) timeouts as LAS while monitoring a delegated token holder;
   3) timeouts as a prior LAS while monitoring a transfer of the LAS role to another DLE;
   4) timeouts resulting in sending a CL DLPDU in an attempt to claim the LAS role and (re)-initialize the local link.

   This counter shall be at least two octets in length.

10.5.6.1.3 Additional reception-related statistical measures required of a bridge DLE

A DLE functioning as a port of a bridge shall keep the following statistical measures in addition to those specified in 10.5.6.1.2, where that information is available.

a) The total number of DLSDUs which were stored and forwarded for retransmission by the bridge. This counter shall be at least two octets in length, and should be at least four octets in length.

b) The total number of octets of the DLSDUs specified in a), not counting FCS octets. This counter shall be at least two octets in length, and should be at least four octets in length.

10.5.6.2 Additional required DLE-collected fault-management data

For each of the last four timeout events counted in 10.5.6.1.3g), the DLE shall retain the frame control octet and address subfields, if any, of the DLPDU which the DLE last transmitted or received (as appropriate) immediately before the occurrence of that event, where that information is available.

10.5.6.3 Additional statistical measures

A DLE may also keep additional statistical measures on other aspects of DLE operation to facilitate network and end-system fault management and performance monitoring, where that information is available.

10.5.7 DLE Variables which can be read and set by DL-management

DL-management may read and set

a) the desired DL-node class of operation: basic, link master, or bridge;
b) permitted ranges of QoS values and default QoS (see IEC 61158-3-1) values for:
   1) DLL priority
   2) DLL maximum confirm delay)
   3) DLPDU authentication
   4) DL-scheduling policy
   5) maximum DLSDU size
   6) residual activity (see also 8.2.1.1f)
   7) maximum send and receive window sizes (see also 8.2.1.3e)
   8) transmit window.
c) a default schedule for a link master or bridge class DLE to use if it needs to function as the LAS, in situations where the DLE is either incapable of, or unable to, get a more current schedule from a currently functioning LAS DLE.

10.5.8 DLE Actions Requestable by DL-management

The DLE actions which DL-management may request are:

a) invoke service primitives such as DL-CREATE, DL-BIND and DL-SCHEDULE-SEQUENCE (see IEC 61158-3-1), with the Action-qualifiers parameter conveying the input sub-parameters specified for the selected service, and the Additional-information parameter returning any resultant output subparameters to DL-management.
b) change DL-node status between online and offline, or for fractional-duty-cycle (FDC) nodes between “asleep” and ”awake” (see 6.21);
c) for DL-nodes functioning as a link master or a bridge DLE, to attempt to become the link’s active scheduler (LAS), or to request that the LAS role, when currently held, be transferred to another DLE (for example, in this latter case in anticipation of the DLE currently functioning as LAS going offline).

10.6 Implementation profiles

Profiles provide a simple multi-attribute means of summarizing an implementation’s capabilities, and thus its applicability to various time-critical communications needs.

10.6.1 DL-address, queue and buffer management services

The first profile categories specifies whether an implementation supports LONG DL-addresses (see 5.2.1.1) and thus supports direct communication and direct management through a bridge (see 3.3.1).

Permitted choices are:

YES LONG, SHORT and VERY-SHORT address representations are supported (that is, are recognized and can be generated);

NO only SHORT and VERY-short address representations are supported (that is, are recognized and can be generated).

10.6.2 DL-data delivery services

The first profile category specifies the degree to which an implementation supports those communications priorities which facilitate preemptive use of the shared communications resource. The remaining profile categories specify the degree to which an implementation supports those DLC data-delivery features which enhance the unaugmented DL-capability of conveying limited amounts of DLS-user information, with limited QoS, between DLSAPs:

a) Three priorities are defined in this standard for both connection-mode and connectionless data communications. The utility for time-critical communications of an implementation of this standard depends substantially on whether the implementation supports multiple priorities of data delivery to the DLS-user. Two levels of support, of decreasing difficulty and utility, have been specified. Which of them does the implementation support?

   Permitted choices are:

   MULTI URGENT, NORMAL and TIME-AVAILABLE priorities are all supported for both connection-mode and connectionless data communications;

   SINGLE only TIME-AVAILABLE priority is supported; URGENT and NORMAL priorities are not supported.

b) Four classes of DLC data delivery features are defined in the standard for both peer and multi-peer DLCs. Of these, three (CLASSICAL, ORDERED and UNORDERED) provide substantively different qualities of service. In addition, the quality of service of CLASSICAL peer-to-peer data delivery is different from that of it analogous multi-peer counterpart.

   NOTE In the absence of communications errors, DISORDERED and CLASSICAL data delivery services are indistinguishable. Thus the DISORDERED data delivery service is an implementation option when CLASSICAL service is provided, and does not provide a substantively different quality of service from the CLASSICAL service.

   The utility for time-critical communications of an implementation of this standard depends heavily on the degree to which the implementation supports the data delivery paradigms needed by the DLS-user. Four levels of support, of decreasing difficulty and utility, have been specified. Which of them does the implementation support?

   Permitted choices are:

   A CLASSICAL, ORDERED and UNORDERED peer and multi-peer DLCs are all supported; DISORDERED peer and multi-peer DLCs may be supported;

   B only ORDERED and UNORDERED peer and multi-peer DLCs, and CLASSICAL peer DLCs, are supported; CLASSICAL and DISORDERED multi-peer DLCs are not supported; DISORDERED peer DLCs may be supported;

   C only ORDERED and UNORDERED peer and multi-peer DLCs are supported; CLASSICAL and DISORDERED peer and multi-peer DLCs are not supported;

   D only UNORDERED peer and multi-peer DLCs are supported; ORDERED, CLASSICAL and DISORDERED peer and multi-peer DLCs are not supported.

c) Multi-peer DLCs can be capable of conveying DLSDUs from subscribers to their publisher, providing a capability roughly equivalent to the inverse of a connectionless multi-cast. Is this optional capability supported?

Permitted choices are:

YES DLSDUs may be conveyed from subscribers to their publisher;

NO DLSDUs cannot be conveyed from subscribers to their publisher.

d) CLASSICAL, ORDERED and DISORDERED peer and publisher DLCs are capable of conveying DLSDUs up to 16 times the size of the DLSDUs conveyable with UNORDERED DLCs or connectionless services. What is the ratio of the maximum size DLSDU conveyable on CLASSICAL and ORDERED peer and multi-peer DLCs to that conveyable with UNORDERED DLCs or connectionless services?

Permitted choices are the integers 1 to 16, inclusive. If CLASSICAL DLCs are supported and the ratios are different for CLASSICAL and ORDERED DLCs, or if the ratios are different for peer and multi-peer DLCs, then the smallest of the ratios shall be specified. If the choice specified for category b) above was **D**, then this value shall be 1 (one).

e) ORDERED peer and multi-peer DLCs are capable of computing TRANSPARENT, RESIDENCE, UPDATE and SYNCHRONIZED timeliness of DLSDU conveyance. Which of these forms of timeliness computations are supported?

Permitted choices are NONE, just TRANSPARENT, or TRANSPARENT plus any combination of the other three. If the choice specified for category b) above was **D**, then this value shall be NONE.

f) ORDERED peer and multi-peer DLCs are capable of conveying DL-time stamps with DLSDUs. Are this capability, and the capability of noting the DL-time of DL-PUT request primitives, and of providing TRANSPARENT timeliness, all supported?

Permitted choices are:

YES all three capabilities are supported, with best achievable time resolution dependent on the time class specified in 10.6.3;

NO not all of these capabilities are supported.

If the choice specified for category b) above was **D**, then this value shall be NO.

g) Three forms of connectionless data delivery are defined in the standard:

1) unitdata transfer with local confirmation;
2) unitdata transfer with remote confirmation;
3) unitdata exchange.

Which of these forms of connectionless data delivery are supported?

Permitted choices are:

X all three forms are supported;

Y both forms of unitdata transfer are supported; unitdata exchange is not supported;

Z only unitdata transfer with local confirmation is supported; unitdata transfer with remote confirmation and unitdata exchange are not supported;

10.6.3 DL-time and time-based scheduling services

The last three profile categories specify the degree to which an implementation supports a fieldbus-wide sense of time and time-based scheduling:

a) The maximum asynchronism in the fieldbus-shared sense of time determines the coarseness or fineness of such shared activities as distributed time-based scheduling and distributed sequence-of-events determination. Eight classes of time synchronism have been defined:

1 µs Under steady-state conditions, the DL-implementation can maintain a sense of DL-time whose maximum long-term mean error relative to the local LAS's sense of DL-time is at most 250 ns for basic and link master DLEs, and 125 ns for bridge DLEs, under the condition that a single TD DLPDU is received once each 5 ms, as shown in Table 133.

NOTE When connected to a similar reference node by a multi-link fieldbus with no more than two intervening bridges, in which the reference node, the intervening bridges, and the node under measurement all meet this requirement, such a DL-implementation can provide a sense of time differing from that in the reference node by less than 500 ns, and thus from any node similarly removed from the reference node by less than 1 µs.

10 µs Under steady-state conditions, the DL-implementation can maintain a sense of DL-time whose maximum long-term mean error relative to the local LAS's sense of DL-time is at most 3 µs for basic and link master DLEs, and 1 µs for bridge DLEs, under the condition that a single TD DLPDU is received once each 50 ms, as shown in Table 133.

NOTE When connected to a similar reference node by a multi-link fieldbus with no more than two intervening bridges, in which the reference node, the intervening bridges, and the node under measurement all meet this requirement, such a DL-implementation can provide a sense of time differing from that in the reference node by less than 5 µs, and thus from any node similarly removed from the reference node by less than 10 µs.

100 µs Under steady-state conditions, the DL-implementation can maintain a sense of DL-time whose maximum long-term mean error relative to the local LAS's sense of DL-time is at most 25 µs for basic and link master DLEs, and 10 µs for bridge DLEs, under the condition that a single TD DLPDU is received once each 500 ms, as shown in Table 133.

NOTE When connected to a similar reference node by a multi-link fieldbus with no more than two intervening bridges, in which the reference node, the intervening bridges, and the node under measurement all meet this requirement, such a DL-implementation can provide a sense of time differing from that in the reference node by less than 50 µs, and thus from any node similarly removed from the reference node by less than 100 µs.

1 ms Under steady-state conditions, the DL-implementation can maintain a sense of DL-time whose maximum long-term mean error relative to the local LAS's sense of DL-time is at most $\frac{1}{4}$ ms for basic and link master DLEs, and $\frac{1}{16}$ ms for bridge DLEs, under the condition that a single TD DLPDU is received once each 5 s, as shown in Table 133.

NOTE When connected to a reference node by a multi-link fieldbus, in which

a) the reference node, two bridges in series, and the node under measurement all meet this requirement, or

b) the reference node and three bridges in series meet the requirements of the 10 µs class, and an additional two bridges in series and the node under measurement all meet the requirement of the 1 ms class,

such a DL-implementation can provide a sense of time differing from that in the reference node by less than 0,5 ms, and thus from any node similarly removed from the reference node by less than 1 ms.

10 ms Under steady-state conditions, the DL-implementation can maintain a sense of DL-time whose maximum long-term mean error relative to the local LAS's sense of DL-time is at most 2,5 ms for basic and link master DLEs, and 0,5 ms for bridge DLEs, under the condition that a single TD DLPDU is received once each 10 s, as shown in Table 133.

NOTE When connected to a reference node by a multi-link fieldbus, in which

a) the reference node, two bridges in series, and the node under measurement all meet this requirement, or

b) the reference node and five bridges in series meet the requirements of the 1 ms class, and an additional two bridges in series and the node under measurement all meet the requirement of the 10 ms class,

such a DL-implementation can provide a sense of time differing from that in the reference node by less than 5 ms, and thus from any node similarly removed from the reference node by less than 10 ms.

100 ms Under steady-state conditions, the DL-implementation can maintain a sense of DL-time whose maximum long-term mean error relative to the local LAS's sense of DL-time is at most 25 ms for basic and link master DLEs, and 5 ms for bridge DLEs, under the condition that a single TD DLPDU is received once each 25 s, as shown in Table 133.

NOTE When connected to a reference node by a multi-link fieldbus, in which

a) the reference node, two bridges in series, and the node under measurement all meet this requirement, or

b) the reference node and five bridges in series meet the requirements of the 10 ms class, and an additional two bridges in series and the node under measurement all meet the requirement of the 100 ms class,

such a DL-implementation can provide a sense of time differing from that in the reference node by less than 50 ms, and thus from any node similarly removed from the reference node by less than 100 ms.

1 s Under steady-state conditions, the DL-implementation can maintain a sense of DL-time whose maximum long-term mean error relative to the local LAS's sense of DL-time is at most 250 ms for basic and link master DLEs, and 50 ms for bridge DLEs, under the condition that a single TD DLPDU is received once each 55 s, as shown in Table 133.

NOTE When connected to a reference node by a multi-link fieldbus, in which

a) the reference node, two bridges in series, and the node under measurement all meet this requirement, or

b) the reference node and five bridges in series meet the requirements of the 100 ms class, and an additional two bridges in series and the node under measurement all meet the requirement of the 1 s class,

such a DL-implementation can provide a sense of time differing from that in the reference node by less than 0,5 s, and thus from any node similarly removed from the reference node by less than 1 s.

NONE The DL-implementation does not provide a sense of time which is synchronized with that of other nodes on a connected fieldbus.

NOTE 1 In this last case, the DL-implementation may still be able to report the last time information heard during a fieldbus DL-time broadcast, and may even be able to estimate the intervening interval. However, the DL-implementation is unable to maintain a synchronized sense of DL-time with only low-frequency time broadcasts from a reference DLE on the fieldbus (that is, from the local link's LAS).

NOTE 2 The time-synchronism class of a link's LAS determines the best achievable time synchronism for the entire link. The time-synchronism class of a bridge determines the best achievable time synchronism for that portion of the extended link which inter-communicates through the bridge.

b) Time-based scheduling services are dependent on the time synchronism of the fieldbus DLE, and so can be provided only when some sense of time synchronism exists within the DLE and on the local link. Are local requests for time-based scheduling services, including consequent scheduling interactions with the local link's LAS DLE, supported?

Table 133 – Maximum permitted phase-tracking error in a DLE's sense of DL-time at the minimum requireable Time Distribution period

Time-synchronism class	Maximum permitted error by DLE class		Minimum requireable value for V(TDP)
	Basic or link master DLE	Bridge DLE	
1 µs	250 ns	125 ns	5 ms
10 µs	3 µs	1 µs	50 ms
100 µs	25 µs	10 µs	500 ms
1 ms	250 µs	1/16 ms	5 s
10 ms	2,5 ms	0,5 ms	10 s
100 ms	25 ms	5 ms	25 s
1 s	250 ms	50 ms	55 s
none	> 100 yr	> 100 yr	not applicable

Permitted choices are:

YES PERIODIC schedules, and ONE-TIME schedules with a specified starting time, are supported, with the best achievable time resolution dependent on the time class specified in 10.6.3;

NO these capabilities are not supported.

If the choice specified for category a) above was NONE, then this value shall be NO.

NOTE A DLE with no sense of DL-time may still be able to respond to time-based scheduled activities, such as periodic receipt of an ES DLPDU enabling periodic local schedule execution within the DLE. However, such a DLE cannot make a time-based scheduling request to the LAS, because it cannot specify the start time for the request with reference to the LAS' DL-time. Thus all such activities can only occur due to DL-management preconfiguration of the DLE through extra-protocol means, and not as a result of DLS-user requests for DL-service.

c) Time-based scheduling activities within an LAS DLE also are dependent on the time synchronism of the fieldbus DLE, and so can be provided only when some sense of time synchronism exists. Five classes of support for these services have been defined:

DYNAMIC The DL-implementation of LAS functions supports

1) time-based schedules which have been pre-constructed by DL-management;
2) requests for time-based scheduling services received from DLEs on the local link and modification of the current schedule to accommodate those requests, where feasible;
3) reception of the current schedule from a previous LAS DLE;
4) redistribution of the current schedule to other potential LAS DLEs;

SHARABLE The DL-implementation of LAS functions supports

1) time-based schedules which have been pre-constructed by DL-management;
2) time-based schedules which have been pre-constructed by a previous DYNAMIC-class LAS DLE;
3) reception of the current schedule from a previous LAS DLE;
4) redistribution of the current schedule to other potential LAS DLEs.

Such a DLE does not have the ability to construct schedules dynamically.

UPDATABLE The DL-implementation of LAS functions supports

1) time-based schedules which have been pre-constructed by DL-management;
2) time-based schedules which have been pre-constructed by a previous DYNAMIC-class LAS DLE;
3) reception of the current schedule from a previous LAS DLE.

Such a DLE does not have the ability to redistribute the current schedule to other potential LAS DLEs.

NOTE This inability to redistribute the schedule implies that a received schedule is transformed into an internal representation in which some of the received scheduling information has been discarded.

STATIC The DL-implementation of LAS functions supports time-based schedules which have been pre-constructed by DL-management. It does not have the ability to receive the current schedule from a previous LAS DLE.

NONE Either

1) the DLE is a BASIC DLE, or
2) the implemented LAS functions do not support time-based schedules, but do support free-running schedules which have been pre-constructed by DL-management.

11 PICS proforma

NOTE Clause 11 only partially follows ISO/IEC Directives Part 3 because it must be usable as a separable standalone document.

11.1 Introduction

To evaluate conformance to 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).

11.2 General

Clause 11 provides the PICS proforma for the Type 1 fieldbus Data Link Layer protocol as specified in the body of this standard, in compliance with the relevant requirements, and in accordance with the relevant guidelines, given in ISO/IEC 9646-2.

When the supplier completes this proforma, this standard becomes the PICS for the specified implementation. The PICS is then used to perform static conformance review of the implementation to determine that all mandatory features are implemented and that all conditional features are correctly supported dependent on the options claimed to be implemented. The PICS also is used as an aid in test case selection.

NOTE It is anticipated that this PICS will be extended in future revisions of this standard.

11.2.1 Audience

The PICS proforma shall be used by the supplier of an implementation to state the capabilities of the implementation. The completed PICS shall be used by the test operator and the test analyst to determine the static conformance of an implementation under test (IUT), to select test cases and test suites to use when testing the IUT, and to aid in test results analysis. The completed PICS may be used by a user to determine the adequacy of an implementation.

11.3 Normative references

See the current editions of ISO/IEC 9646-1 and ISO/IEC 9646-2, as listed in the Bibliography.

11.4 Definitions

Clause 11 uses the following terms defined in ISO/IEC 9646-1:

a) PICS proforma;

b) protocol implementation conformance statement (PICS);

c) static conformance review.

11.5 Abbreviations

The following abbreviations are used in defining the status of a feature, timer, parameter, or other capability:

M : mandatory;

O : optional;

X : prohibited;

N/A : not applicable;

C : conditional.

The following abbreviations are used in defining the support of a feature, timer, parameter, or other capability:

Y or **y** or **yes** : implemented;

N or **n** or **no** : not implemented

The following abbreviations are used in defining the range or existence of a feature, parameter, or other capability:

F : false;

T : true

References of the form "/n" or "/n.x" refer to the conformance claimed for row "n", or for row "n" column "x" respectively, in the same table. References of the form "m/n" or "m/n.x" are a corresponding reference to the conformance claimed in table "m".

11.6 Conformance

The supplier of a protocol implementation which is claimed to conform to this part of IEC 61158-4 is required to complete a copy of the PICS proforma provided in Clause 11 and is required to provide the information necessary to identify both the supplier and the implementation.

11.7 Instructions

The supplier of a fieldbus Data Link protocol implementation shall take the following steps to complete the PICS proforma.

The supplier shall complete the identification section.

The supplier shall enter either a 'Y' or an 'N' (or equivalents), as just defined under 'Abbreviations', in each entry in the support columns of the following tables. The supplier shall enter a 'Y' if the feature, timer, parameter, or capability is implemented. When a conformance requirement defines the allowable or required range or set of values for a capability or parameter, the "allowed" or "required" column contains the specified value(s), and the supplier of the implementation shall only enter a 'Y' in the support entry if the implementation completely supports the specified range or set of values. The supplier shall enter a 'Y' in the support entry even if access to the feature is not provided as an option (that is, the feature is unconditionally present).

The supplier shall enter an 'N' if the feature, timer, parameter, or capability is not implemented at all. If the supplier enters an 'N' for a mandatory feature, then the supplier shall attach an explanation as to why the feature was not implemented.

The supplier shall enter the specified ranges, values, default value (as set by the supplier of the implementation) and units of measure, where applicable, for all implemented parameters. Continuous ranges shall be entered in the form: lowest-value ':' highest-value. Series of discrete values or subranges shall be entered from lowest value to highest value, with each value or subrange separated by commas or semicolons.

If the units of measure, or any other column, is not applicable, then a '—' should be recorded in the appropriate entry.

11.8 Identification

Date:

Data Link Layer Implementation Information

Implementation Name:

Implementation Identifier (including version/release):

Vendor Information

Company Name:

Technical Representative:

Address:

Telephone:

Facsimile:

Telex:

E-mail

Protocol Identification

Conformance is hereby claimed to this part of IEC 61158-4-1.

The protocol versions supported are:

Overall Conformance Statement

Are all mandatory capabilities of the claimed version(s) of the above protocol fully supported? _______

Are all mandatory capabilities of the profiles(s) claimed under 13.9 fully supported? _______

Answering "No" to either of these questions indicates non-conformance to the protocol specification. Non-supported mandatory capabilities are to be identified in the PICS, with an explanation of why the implementation is non-conforming.

11.9 Implementation profile

This subclause identifies the major implementation choices which limit the DL-services and DL-service features which the implementation provides and supports. It is based on 12.6.

11.9.1 DL-address, queue and buffer management services

The implementation may, but need not, support LONG DL-addresses (see 12.6.1).

NOTE Direct communication and direct management through bridges is based upon the use of LONG DL-addresses.

Supported DL-address lengths				
Item number	**Supported DL-address lengths**	**A) Reference**	**B) Status**	**C) Support**
1	LONG DL-addresses	12.6.1	O	
2	SHORT DL-addresses	12.6.1	M	
3	VERY-SHORT DL-addresses	12.6.1	M	

11.9.2 Data delivery services

11.9.2.1 Data delivery priorities

The implementation must support the TIME-AVAILABLE priority of data transfer. It may also support other priorities: NORMAL and URGENT.

Supported data delivery priorities				
Item number	**Supported data delivery priorities**	**A) Reference**	**B) Status**	**C) Support**
1	TIME-AVAILABLE	12.6.2 a	M	
2	NORMAL	12.6.2 a	O	
3	URGENT	12.6.2 a	C1	
C1 : if /2 then M else X				

11.9.2.2 Connection-mode data delivery features

The implementation must support UNORDERED peer-to-peer and multi-peer DL-connections. It may also support other DLC data delivery features: ORDERED, CLASSICAL and DISORDERED.

Supported DLCEP classes and data delivery features				
Item number	**Supported DLCEP class(es) and data delivery features**	**A) Reference**	**B) Status**	**C) Support**
1 (**A**)	CLASSICAL, ORDERED and UNORDERED peer and multi-peer DLCs, with DISORDERED DLCs either unsupported, or supported for both peer and multi-peer	12.6.2 b	C1	
2 (**B**)	only ORDERED and UNORDERED peer and multi-peer DLCs, and CLASSICAL peer DLCs, with DISORDERED peer DLCs optional	12.6.2 b	C1	
3 (**C**)	only ORDERED and UNORDERED peer and multi-peer DLCs	12.6.2 b	C1	
4 (**D**)	only UNORDERED peer and multi-peer DLCs	12.6.2 b	C1	
C1 : exactly one of these alternatives must be chosen				

11.9.2.3 Support for subscribers-to-publisher DLSDUs

The implementation may provide the ability for subscribers on a multi-peer DLC to transmit DLSDUs to their publisher, providing a capability roughly equivalent to the inverse of a connectionless multi-cast.

Support for subscribers-to-publisher DLSDUs				
Item number	**Support for subscribers-to-publisher DLSDUs**	**A) Reference**	**B) Status**	**C) Support**
1	UNORDERED multi-peer DLCs	12.6.2 c	O	

11.9.2.4 Ratio of connection-mode maximum DLSDU size to maximum DLPDU-data size

The maximum-size DLSDU which can be transferred by an unordered DLC is the same as that which can be conveyed at the same priority by the connectionless data transfer service. ORDERED, CLASSICAL and DISORDERED DLCs provide the capability to transfer DLSDUs that are up to 16 times as large, by segmenting the DLSDU into 16 or fewer segments which will be transferred by separate DLPDUs. Thus the maximum ratio of DLSDU-size to DLPDU-data-size is 1 for UNORDERED DLCs and connectionless data transfer, and is between 1 and 16, inclusive for ORDERED, CLASSICAL and DISORDERED DLCs.

Supported maximum DLSDU : DLPDU-data ratios				
Item number	Maximum DLSDU-size / DLPDU-data size ratios	Reference	Allowed range	Support
1	CLASSICAL multi-peer DLCs	12.6.2 d	0:16 C1	
2	CLASSICAL peer DLCs	12.6.2 d	0:16 C1	
3	ORDERED peer and multi-peer DLCs	12.6.2 d	0:16	
4	UNORDERED peer and multi-peer DLCs	12.6.2 d	1	
C1 : if the supported value is non-zero then it must be equal to the value claimed in /3				

11.9.2.5 DL-timeliness

The implementation may provide the ability to record the DL-time at which a DLSDU is written to a buffer; to compute the timeliness of that buffer writing, or of subsequent buffer reading, or both, according to pre-established criteria; to convey that timeliness across a DLC; and to report both sending and receiving timeliness to a receiving DLS-user when delivering the DLSDU.

DL-timeliness support				
Item number	Supported types of timeliness	A) Reference	B) Status	C) Support
1	NONE	12.6.2 e	M	
2	TRANSPARENT	12.6.2 e	C1	
3	RESIDENCE	12.6.2 e	C2	
4	UPDATE	12.6.2 e	C2	
5	SYNCHRONIZED	12.6.2 e	C2	
C1 : if 11.9.2.2/4.C then X else O C2 : if /2.C then O else X				

11.9.2.6 DL-time-stamped DLSDUs

The implementation may provide the ability to record the DL-time at which a DLSDU is written to a buffer by a DL-PUT request primitive and to convey that time across a DLC for subsequent use by a receiving DLE or for reporting to a receiving DLS-user when delivering the DLSDU.

DL-time-stamped DLSDU support				
Item number	DL-time-stamped DLSDU support	A) Reference	B) Status	C) Support
1	NO	12.6.2 f	M	
2	YES	12.6.2 f	C1	
C1 : if 11.9.2.5/2.C then O else X				

11.9.2.7 Connectionless data delivery features

The implementation must support unitdata transfer with local confirmation. It may also support unitdata transfer with remote confirmation, and unitdata exchange.

Supported connectionless data delivery features				
Item number	**Supported connectionless data delivery features**	**A) Reference**	**B) Status**	**C) Support**
1 (**X**)	unitdata exchange, and unitdata transfer with both local and remote confirmation	12.6.2 g	C1	
2 (**Y**)	only unitdata transfer with both local and remote confirmation	12.6.2 g	C1	
3 (**Z**)	only unitdata transfer with local confirmation	12.6.2 g	C1	
C1 : exactly one of these alternatives must be chosen				

11.9.3 DL-time and scheduling services

11.9.3.1 DL-time synchronism

The implementation may provide time synchronism with other devices on the extended fieldbus.

Time synchronism support						
Item number	**Measure of provided time synchronism**	**A) Reference**	**B) Status**	**C) Support**	**Minimum required value for V(TDP)**	
					D) Range	**E) Support**
1	none	12.6.3 a	C1		none	
2	1 s	12.6.3 a	C1		≥ 55 s C2	
3	100 ms	12.6.3 a	C1		≥ 25 s C3	
4	10 ms	12.6.3 a	C1		≥ 10 s C4	
5	1 ms	12.6.3 a	C1		≥ 5 s C5	
6	100 µs	12.6.3 a	C1		≥ 500 ms C6	
7	10 µs	12.6.3 a	C1		≥ 50 ms C7	
8	1 µs	12.6.3 a	C1		≥ 5 ms C8	

C1 : exactly one of these alternatives must be chosen
C2 : if /2.C then M else X
C3 : if /3.C then M else X
C4 : if /4.C then M else X
C5 : if /5.C then M else X
C6 : if /6.C then M else X
C7 : if /7.C then M else X
C8 : if /8.C then M else X

11.9.3.2 Time-based scheduling services

The implementation may support local time-based scheduling services.

Time-based scheduling service support				
Item number	**Support for PERIODIC schedules and ONE-TIME schedules with a specified starting time**	**A) Reference**	**B) Status**	**C) Support**
1	YES	12.6.3 b	C1	
2	NO	12.6.3 b	C2	
C1 :if time_synchronism_supported then C2 else X C2 : exactly one of these alternatives must be chosen		time_synchronism_supported = 11.9.3.1/2 or /3 or /4 or /5 or /6 or /7 or /8		

11.9.3.3 LAS time-based scheduling activities

The fieldbus Data Link layer protocol requires at least one device on each local link to be functioning as a link master or a bridge. Both link master and bridge class devices are capable of initializing the token on the local link, and of serving as the local Link Active Scheduler (LAS). When functioning as an LAS, these devices provide centralized schedule-execution services, and may provide centralized schedule distribution and schedule construction services, for the entire link.

Bridge-class and link master-class devices can be full-capability devices, capable of creating a schedule for the local link, resources permitting, from any combination of scheduling requests received from the various devices on the link. Such devices instead may offer lesser capability, permitting construction of reduced-functionality networks. This reduced-functionality may be adequate for many devices which will become the LAS only if there are no bridges or higher-capability devices active on the link, or when the intended usage is only on preconfigured local links.

Five levels of scheduling support for link master-class and bridge-class devices have been defined:

LAS time-based scheduling activity support				
Item number	**LAS time-based scheduling activity support**	**A) Reference**	**B) Status**	**C) Support**
1	DYNAMIC – can construct, distribute, receive and execute time-based schedules for the local link	12.6.3 c	C1	
2	SHARABLE – can distribute, receive and execute time-based schedules for the local link, but cannot construct such schedules	12.6.3 c	C1	
3	UPDATABLE – can receive and execute time-based schedules for the local link, but cannot distribute such schedules	12.6.3 c	C1	
4	STATIC – can execute time-based schedules for the local link, but cannot receive such schedules	12.6.3 c	C1	
5	NONE – no time-based scheduling for link	12.6.3 c	C2	
C1 :if time_synchronism_supported then C2 else X C2 : exactly one of these alternatives must be chosen		time_synchronism_supported = 11.9.3.1/2 or /3 or /4 or /5 or /6 or /7 or /8		

11.10 Major low-level capabilities

This subclause identifies many of the major conformance requirements of the implementation. The conformance requirements specified in this subclause are those related to the lower-level medium-access functions of the DL-protocol. Subclause 11.11 specifies the conformance requirements for the higher-level functions of the protocol – those which are independent of the underlying medium-access functions.

In a number of cases (11.10.3.1 and 11.10.3.2, and 11.10.4.4 to 11.10.4.7), conformance requirements for reception are Mandatory, but with the added note “only a minimally responsive subset need be implemented.” This should be interpreted as meaning that when reception of the specified DLPDU initiates some sort of response, only a minimal-effort response, rejecting the requested action, needs to occur, and that complete handling of the received DLPDU’s parameter and DLS-user data fields is not required for such implementations.

11.10.1 DLE classes implemented

The fieldbus Data Link layer protocol defines three major lower-level classes of DLE operation, each a superset of the prior class. The support requirements for these classes differ.

DLE Classes Implemented				
Item number	**DLE Class**	**Reference**	**Status**	**Support**
1	Basic	6.6.1	M	
2	Link master	6.6.2	O	
3	Bridge	6.6.3	C1	
C1 : if /2 then O else X				

11.10.2 Basic addressing requirements

DL-addresses have multiple lengths, and potentially have two synonyms. Where such synonyms exist, the choice of the appropriate synonym for use in a specific DLPDU is dependent both on the required QoS and on the availability of similar synonyms for all of the other addresses in the DLPDU.

Basic addressing requirements				
Item number	**Addressing requirements**	**Reference**	**Status**	**Support**
1	Are all LONG DL-addresses four octets in length?	5.2.2.1, 6.3.1	M	
2	Are all SHORT DL-addresses two octets in length?	5.2.2.2, 6.3.2	M	
3	Are all DL-addresses interpreted as specified in Table 1, Table 2 and Table 4?	6.3.2	M	
4	Are all predefined DL-addresses for the implemented classes claimed under 11.10.1 recognized as specified in Table 5 through Table 7?	6.3.3	M	
5	Are all LONG DL-addresses with a link-designator subfield (HL) whose value is not equal to zero and is not equal to V(TL) considered non-local?	6.3.2	M	
6	Are all LONG DL-addresses with a link-designator subfield (HL) whose value is zero or is V(TL), and with a node-designator subfield whose value is not equal to zero and is not equal to V(TN), considered link-local?	6.3.2	M	
7	Are all LONG DL-addresses with a link-designator subfield (HL) whose value is zero or is V(TL), and with a node-designator subfield whose value is equal to zero or is equal to V(TN), considered node-local?	6.3.2	M	
8	Are all SHORT DL-addresses with a node-designator subfield (N) whose value is not equal to zero and is not equal to V(TN), considered link-local?	6.3.2	M	
9	Are all SHORT DL-addresses with a node-designator subfield (N) whose value is equal to zero or is equal to V(TN), considered node-local?	6.3.2	M	
10	During address recognition, are all node-local DL-addresses with the same subnode selector (S) considered equivalent?	6.3.2	M	
11	During address recognition, are all link-local DL-addresses with the same sublink selector (N.S) considered equivalent?	6.3.2	M	
12	During DLPDU formation, when the DLS-user's DLPDU-authentication QoS was SOURCE or MAXIMAL, then is a source DL-address included within each resultant DLPDU, when possible?	6.2.4	M	
13	During DLPDU formation, when the DLS-user's DLPDU-authentication QoS was MAXIMAL, then are all DL-addresses within each resultant DLPDU LONG?	6.2.4	M	
14	During DLPDU formation, when at least one of the DLPDU's addresses is non-local, then are all DL-addresses within that DLPDU LONG?	5.2.1.1	M	

Basic addressing requirements				
Item number	**Addressing requirements**	**Reference**	**Status**	**Support**
15	During DLPDU formation, when a DL-address is both link-local and LONG, then is the link component of the DL-address equal to the local link's link_id, and not zero unless that link_id is itself zero?	4.3.2.1	M	
16	During DLPDU formation, when a DL-address is node-local, then is the node component of the DL-address equal to the node's node_id, and not zero?	5.2.2.1, 4.3.2.2	M	
17	During DLPDU formation, when the DLS-user's DLPDU-authentication QoS was ORDINARY or SOURCE, and all DL-addresses within the DLPDU are link-local, then are those DL-addresses all SHORT?	6.2.4	M	

11.10.3 DLPDU support

This subclause enumerates the types of DLPDU which the implementation can receive and recognize, and, separately, which the implementation can transmit, as a function of the highest conformance class claimed under 11.10.1; and separately as a function of the momentary role within the local data link assumed by the implementation.

NOTE It is assumed that an implementation can receive (in a rudimentary sense) all DLPDUs, whether well-formed or not, but that well-formed DLPDUs which are not recognized are ignored.

11.10.3.1 Static DLPDU support

The fieldbus Data Link layer protocol defines three major classes of operation, each a superset of the prior class. This table specifies the static (that is, independent of momentary role) conformance requirements as a function of the claimed capabilities of the implementation. Only the entries for the highest claimed class (right-most in the table) need be completed.

Class dependency of supported DLPDUs														
	Column letter →		Basic class				Link master class				Bridge class			
			A) Receiving and recognizing		B) Forming and sending		C) Receiving and recognizing		D) Forming and sending		E) Receiving and recognizing		F) Forming and sending	
Item number	DLPDU type	Reference	Status	Sup-port	Status	Sup-port	Status	Sup-port	Status	Sup-port	Status	Sup-port	Status	Sup-port
1	EC	8.1	M1		M1		M1		M1		M1		M1	
2	DC	8.2	M1		M1		M1		M1		M1		M1	
3	RC	8.3	M1		C2		M1		C2		M1		M1	
4	CA	8.4	M1		C3		M1		C3		M		M	
5	CD	8.5	M		M		M		M		M		M	
6	ED	8.6	M1		C4		M1		C4		M		M	
7	DT	8.7	M		M		M		M		M		M	
8	SR	8.8	O		M		M		M		M		M	
9	CT	8.9	X		C5		M		C5		M		C5	
10	TD	8.10	C5		X		C5		M		C5		M	
11	RQ	8.11	M		C5		M		C5		M		C5	
12	RR	8.12	C5		M		C5		M		C5		M	
13	PN	8.13	M		X		M		M		M		M	
14	PR	8.14	M		M		M		M		M		M	
15	PT	8.15	M		X		M		M		M		M	
16	ES	8.16	M1		X		M1		M		M1		M	
17	RT	8.17	X		M		M		M		M		M	
18	RI	8.18	X		O		M		O		M		O	
19	CL	8.19	X		X		M		M		M		M	
20	TL	8.20	X		X		M		M		M		M	
21	WK	8.21	O		X		O		M		X		M	
22	IDLE	8.22	X		O		X		O		X		O	
23	spare	8.23	X		X		X		X		X		X	
24	don't use	8.24	X		X		X		X		X		X	

M1: only a minimally-responsive subset need be implemented

C2: if reliable_CO then M else X — reliable_CO = 11.9.2.2/1 or /2

C3: if acknowledged_CL then M elseif 11.9.2.3/1 then O else X — acknowledged_CL = 11.9.2.7/1 or /2

C4: if 11.9.2.7/1 then M else O

C5: if time_synchronism_supported then M else O — time_synchronism_supported = 11.9.3.1/2 or /3 or /4 or /5 or /6 or /7 or /8

11.10.3.2 Dynamic DLPDU support

The fieldbus Data Link layer protocol defines four major roles for operating devices. This table specifies the dynamic (that is, dependent on momentary role) conformance requirements of the implementation as a function of the then-appropriate role. For an implementation claiming only basic Class conformance, only the left-most four columns (A–D) need to be completed. For all other implementations, all columns need to be completed.

Role dependency of supported DLPDUs																		
			When DLE's operational Node address is unknown and DLE has assumed a temporary Node address				When DLE's operational Node address is known and DLE is functioning as a basic DLE				When DLE's operational Node address is known and DLE is functioning as a link master or bridge DLE and DLE is not acting as the LAS				When DLE's operational Node address is known and DLE is functioning as a link master or bridge DLE and DLE is acting as the LAS			
	Column letter →		A) Receiving and recognizing		B) Forming and sending		C) Receiving and recognizing		D) Forming and sending		E) Receiving and recognizing		F) Forming and sending		G) Receiving and recognizing		H) Forming and sending	
Item number	DLPDU type	Reference	Status	Support	Status	Support	Status	Support	Status	Support	Status	Support	Status	Support	Status	Support	Status	Support
1	EC	8.1	O		X		M1		M1		M1		M1		M1		M1	
2	DC	8.2	O		X		M1		M1		M1		M1		M1		M1	
3	RC	8.3	O		X		M1		C2		M1		C2		M1		M1	
4	CA	8.4	M1		X		M1		C3		M1		C3		M		M	
5	CD	8.5	M		X		M		M		M		M		M		M	
6	ED	8.6	M1		X		M1		C4		M1		C4		M		M	
7	DT	8.7	M		X		M		M		M		M		M		M	
8	SR	8.8	O		X		M		M		M		M		M		M	
9	CT	8.9	X		X		X		C5		M		C5		M		C5	
10	TD	8.10	C5		X		C5		X		C5		M		C5		M	
11	RQ	8.11	X		X		M		C5		M		C5		M		C5	
12	RR	8.12	X		X		C5		M		C5		M		C5		M	
13	PN	8.13	M		X		M		X		M		X		X		M	
14	PR	8.14	M		M		M		M		M		M		M		X	
15	PT	8.15	M		X		M		X		M		X		X		M	
16	ES	8.16	X		X		M1		X		M1		X		X		M	
17	RT	8.17	X		X		X		M		X		M		M		X	
18	RI	8.18	X		X		X		O		X		O		M		X	
19	CL	8.19	X		X		X		X		M		M		M		X	
20	TL	8.20	X		X		X		X		M		X		X		M	
21	WK	8.21	X		X		M		X		M		X		X		M	
22	IDLE	8.22	X		X		X		O		X		O		X		O	
23	spare	8.23	X		X		X		X		X		X		X		X	
	don't use	8.24	X		X		X		X		X		X		X		X	

M1 : only a minimally responsive subset need be implemented

C2 : if reliable_CO then M else X — reliable_CO = 11.9.2.2/1 or /2

C3 : if acknowledged_CL then M elseif 11.9.2.3/1 then O else X — acknowledged_CL = 11.9.2.7/1 or /2

C4 : if 11.9.2.7/1 then M else O

C5 : if time_synchronism_supported then M else O — time_synchronism_supported = 11.9.3.1/2 or /3 or /4 or /5 or /6 or /7 or /8

11.10.4 DLPDU-class specific fields and subfields

This subclause enumerates the ranges and states the static conformance requirements for each DLPDU-class-specific field and subfield of each DLPDU. Conformance is only required on those DLPDU classes which were claimed under 11.10.3.1 and 11.10.3.2, and then only for the direction(s) – receiving and recognizing, or forming and sending, or both – which were claimed. Thus any conditions or options specified in the relevant entry or entries in 11.10.3.1 and 11.10.3.2 apply uniformly to all of the specific requirements of the DLPDU class, as detailed in the following subclauses, for the corresponding direction of use. Requirements in the following DLPDU-specific subclauses for directions of use not claimed under either 11.10.3.1 or 11.10.3.2 can be ignored.

11.10.4.1 Specific fields and subfields of the EC DLPDU

Supported parameters						
	Column letter →		A) Receiving and recognizing		B) Forming and sending	
Item number	Parameter	Reference	Status or range	Support	Status or range	Support
1	address size designator	5.2.1.1,				
1.1	LONG	6.1.1.1	C1		C1	
1.2	SHORT		M		M	
2	final token use designator	5.2.1.2,				
2.1	NOT-FINAL	6.1.1.1	M		M	
2.2	FINAL		M		M	
3	DL address formats:	5.2.2,				
3.1	format 1L – LONG destination and two sources	6.1.1.1, 6.1.1.2	M		M	
3.2	format 1S – SHORT destination and two sources		M		M	
3.3	format 2L – two LONG sources only		M		M	
3.4	format 2S – two SHORT sources only		M		M	
4	EC-parameter:	6.1.1.3,				
4.0	version 1	7.1	M		M	
5	permitted octets of user data in DLPDU	5.2.4, 6.1.1.4	0 : 128 octets M		0 : 128 octets M	
C1 : if 11.9.1/1 then M else X						

11.10.4.2 Specific fields and subfields of the DC DLPDU

Supported parameters						
	Column letter →		A) Receiving and recognizing		B) Forming and sending	
Item number	Parameter	Reference	Status or range	Support	Status or range	Support
1	address size designator	5.2.1.1,				
1.1	LONG	6.2.1.1	C1		C1	
1.2	SHORT		M		M	
2	final token use designator	5.2.1.2,				
2.1	NOT-FINAL	6.2.1.1	M		M	
2.2	FINAL		M		M	
3	DL address formats:	5.2.2,				
3.1	format 1L – LONG destination and source	6.2.1.1, 6.2.1.2	M		M	
3.2	format 1S – SHORT destination and source		M		M	
3.3	format 2L – LONG source only		M		M	
3.4	format 2S – SHORT source only		M		M	
4	DC-parameter:	6.2.1.3,				
4.0	version 1	7.2	M		M	
5	permitted octets of user data in DLPDU	5.2.4, 6.2.1.4	0 : 128 octets M		0 : 128 octets M	
C1 : if 11.9.1/1 then M else X						

11.10.4.3 Specific fields and subfields of the RC DLPDU

Supported parameters						
	Column letter →		A) Receiving and recognizing		B) Forming and sending	
Item number	Parameter	Reference	Status or range	Support	Status or range	Support
1	address size designator	5.2.1.1,				
1.1	LONG	6.3.1.1	C1		C1	
1.2	SHORT		M		M	
2	final token use designator	5.2.1.2,				
2.1	NOT-FINAL	6.3.1.1	M		M	
2.2	FINAL		M		M	
3	priority designator	5.2.1.3,				
3.1	URGENT	6.3.1.1	C2		C2	
3.2	NORMAL		C2		C2	
3.3	TIME-AVAILABLE		M		M	
4	DL address formats:	5.2.2,				
4.1	format 1L – LONG destination and source	6.3.1.1, 6.3.1.2	M		M	
4.2	format 1S – SHORT destination and source		M		M	
4.3	format 2L – LONG source only		M		M	
4.4	format 2S – SHORT source only		M		M	
5	RC-parameter:	6.3.1.3,				
5.0	version 1	7.3	M		M	
Item number	**Parameter**	**Reference**	**Range (in octets)**	**Supported range**	**Range (in octets)**	**Supported range**
6	permitted octets of user data in DLPDU	5.2.4,				
6.1	URGENT priority	6.5.1.4	0 : 64 C2		0 : 64 C2	
6.2	NORMAL priority		0 : 128 C2		0 : 128 C2	
6.3	TIME-AVAILABLE priority		0 : 256 M		0 : 256 M	
C1 : if 11.9.1/1 then M else X						
C2 : if 11.9.2.1/2 then M else x						

11.10.4.4 Specific fields and subfields of the CA DLPDU

Supported parameters						
Item number	Column letter →		A) Receiving and recognizing		B) Forming and sending	
	Parameter	Reference	Status	Support	Status	Support
1	address size designator	5.2.1.1,				
1.1	LONG	6.4.1.1	C1		C1	
1.2	SHORT		M		M	
2	final token use designator	5.2.1.2,				
2.1	NOT-FINAL	6.4.1.1	M		M	
2.2	FINAL		M		M	
3	priority designator	5.2.1.3,				
3.1	URGENT	6.4.1.1	M2		C3	
3.2	NORMAL		M2		C3	
3.3	TIME-AVAILABLE		M		M	
4	DL address formats	5.2.2,				
4.1	format 1L – LONG destination and source	6.4.1.1, 6.4.1.2	M		M	
4.2	format 1S – SHORT destination and source		M		M	
4.3	format 2L – LONG destination, omitted source		M		M	
4.4	format 2S – SHORT destination, omitted source		M		M	
5	SD-parameter:	6.4.1.3,				
5.0	version 1 supported	7.4	M		M	
Item number	**Parameter**	**Reference**	**Range (in octets)**	**Supported range**	**Range (in octets)**	**Supported range**
6	permitted octets of user data in DLPDU	5.2.4,				
6.1	URGENT priority	6.4.1.4	0 : 64 M2		0 : 64 C3	
6.2	NORMAL priority		0 : 128 M2		0 : 128 C3	
6.3	TIME-AVAILABLE priority		0 : 256 M		0 : 256 M	
C1 :if 11.9.1/1 then M else X M2 :if 11.9.2.1/2 then M else only a minimally responsive subset need be implemented C3 :if 11.9.2.1/2 then M else X						

11.10.4.5 Specific fields and subfields of the CD DLPDU

Supported parameters						
	Column letter →		A) Receiving and recognizing		B) Forming and sending	
Item number	Parameter	Reference	Status	Support	Status	Support
1	address size designator	5.2.1.1,				
1.1	LONG	6.5.1.1	C1		C1	
1.2	SHORT		M		M	
2	final token use designator	5.2.1.2,				
2.1	NOT-FINAL	6.5.1.1	M		M	
2.2	FINAL		M		M	
3	priority designator	5.2.1.3,				
3.1	URGENT	6.5.1.1	M2		C3	
3.2	NORMAL		M2		C3	
3.3	TIME-AVAILABLE		M		M	
4	DL address formats	5.2.2,				
4.1	format 1L – LONG destination and source	6.5.1.1, 6.5.1.2	M		M	
4.2	format 1S – SHORT destination and source		M		M	
4.3	format 2L – LONG destination, omitted source		M		M	
4.4	format 2S – SHORT destination, omitted source		M		M	
5	SD-parameter:	6.5.1.3,				
5.0	version 1 supported	7.4	M		M	
6	user data in DLPDU	6.5.1.4	X		X	

C1 :if 11.9.1/1 then M else X
M2 :if 11.9.2.1/2 then M else only a minimally responsive subset need be implemented
C3 :if 11.9.2.1/2 then M else X

11.10.4.6 Specific fields and subfields of the ED DLPDU

Supported parameters						
Item number	Column letter →		A) Receiving and recognizing		B) Forming and sending	
	Parameter	Reference	Status	Support	Status	Support
1	address size designator	5.2.1.1,				
1.1	LONG	6.6.1.1	C1		C1	
1.2	SHORT		M		M	
2	final token use designator	5.2.1.2,				
2.1	NOT-FINAL	6.6.1.1	M		M	
2.2	FINAL		M		M	
3	priority designator	5.2.1.3,				
3.1	URGENT	6.6.1.1	M2		C3	
3.2	NORMAL		M2		C3	
3.3	TIME-AVAILABLE		M		M	
4	DL address formats	5.2.2,				
4.1	format 1L – LONG destination and source	6.6.1.1, 6.6.1.2	M		M	
4.2	format 1S – SHORT destination and source		M		M	
4.3	format 2L – LONG destination, omitted source		M		M	
4.4	format 2S – SHORT destination, omitted source		M		M	
5	SD-parameter:	6.6.1.3,				
5.0	version 1 supported	7.4	M		M	
Item number	Parameter	Reference	Range (in octets)	Supported range	Range (in octets)	Supported range
6	permitted octets of user data in DLPDU	5.2.4,				
6.1	URGENT priority	6.6.1.4	0 : 64 M2		0 : 64 C3	
6.2	NORMAL priority		0 : 128 M2		0 : 128 C3	
6.3	TIME-AVAILABLE priority		0 : 256 M		0 : 256 M	

C1 :if 11.9.1/1 then M else X
M2 :if 11.9.2.1/2 then M else only a minimally responsive subset need be implemented
C3 :if 11.9.2.1/2 then M else X

11.10.4.7 Specific fields and subfields of the DT DLPDU

Supported parameters						
Item number	Column letter →		A) Receiving and recognizing		B) Forming and sending	
	Parameter	Reference	Status	Support	Status	Support
1	address size designator	5.2.1.1,				
1.1	LONG	6.7.1.1	C1		C1	
1.2	SHORT		M		M	
2	final token use designator	5.2.1.2,				
2.1	NOT-FINAL	6.7.1.1	M		M	
2.2	FINAL		M		M	
3	priority designator	5.2.1.3,				
3.1	URGENT	6.7.1.1	M2		C3	
3.2	NORMAL		M2		C3	
3.3	TIME-AVAILABLE		M		M	
4	DL address formats	5.2.2,				
4.1	format 1L – LONG destination and source	6.7.1.1, 6.7.1.2	M		M	
4.4	format 1S – SHORT destination and source		M		M	
4.2	format 2L – LONG destination, omitted source		M		M	
4.5	format 2S – SHORT DESTINATION, omitted source		M		M	
4.3	format 3L – long SOURCE only		M		M	
4.6	format 3S – short SOURCE only		M		M	
4.7	format 4 – omitted DESTINATION and omitted source		M		M	
4.8	format 5 – omitted SOURCE only		M		M	
5	SD-parameter:	6.7.1.3,				
5.0	version 1 SUPPORTED	7.3	M		M	
Item number	Parameter	Reference	Range (in octets)	Supported range	Range (in octets)	Supported range
6	permitted octets of user data in DLPDU	5.2.4,				
6.1	URGENT priority	6.7.1.4	0 : 64 M2		0 : 64 C3	
6.2	NORMAL priority		0 : 128 M2		0 : 128 C3	
6.3	TIME-AVAILABLE priority		0 : 256 M		0 : 256 M	

C1 if 11.9.1/1 then M else X
M2 if 11.9.2.1/2 then M else only a minimally responsive subset need be implemented
C3 if 11.9.2.1/2 then M else X

11.10.4.8 Specific fields and subfields of the SR DLPDU

Supported parameters						
Item number	Column letter →		A) Receiving and recognizing		B) Forming and sending	
	Parameter	Reference	Status and range	Support	Status and range	Support
1	DL address format: omitted destination, NODE source	5.2.2.3, 6.8.1.2	M (16: 255)		M (16: 255)	
2	SR-parameter	6.8.1.3	M		M	

11.10.4.9 Specific fields and subfields of the CT DLPDU

The CT DLPDU has no variable fields or subfields.

11.10.4.10 Specific fields and subfields of the TD DLPDU

Supported parameters						
Item number	Column letter →		A) Receiving and recognizing		B) Forming and sending	
	Parameter	Reference	Status and range	Support	Status and range	Support
1	DL address format: NODE source, omitted destination	5.2.2.3, 6.10.1.2	M (16: 255)		M (16: 255)	
2	TD-parameter	6.10.1.3	M		M	

11.10.4.11 Specific fields and subfields of the RQ DLPDU

Supported parameters						
Item number	Column letter →		A) Receiving and recognizing		B) Forming and sending	
	Parameter	Reference	Status and range	Support	Status and range	Support
1	DL address format: NODE.0 destination and source	5.2.2.3, 6.11.1.2	M (4,16: 255).0		M (4,16: 255).0	
2	RQ-parameter	6.11.1.3	M		M	

11.10.4.12 Specific fields and subfields of the RR DLPDU

Supported parameters						
Item number	Column letter →		A) Receiving and recognizing		B) Forming and sending	
	Parameter	Reference	Status and range	Support	Status and range	Support
1	DL address format: NODE.0 destination and source	5.2.2.3, 6.12.1.2	M (4,16: 255).0		M (4,16: 255).0	
2	RR-parameter	6.12.1.3, 6.12.2	M		M	

11.10.4.13 Specific fields and subfields of the PN DLPDU

Supported parameters						
Item number	Column letter →		A) Receiving and recognizing		B) Forming and sending	
	Parameter	Reference	Status	Support	Status	Support
1	DL address format: NODE destination, omitted source	5.2.1.1, 6.13.1.2	M (16: 255)		M (16: 255)	
2	PN-parameters	6.13.1.3, 6.13.2				

11.10.4.14 Specific fields and subfields of the PR DLPDU

Supported parameters						
Item number	Column letter →		A) Receiving and recognizing		B) Forming and sending	
	Parameter	Reference	Range (in octets)	Supported range	Range (in octets)	Supported range
1	permitted octets of SPDU in DLPDU	5.2.4, 6.14.1.4	0 : 64 M		0 : 64 M	

11.10.4.15 Specific fields and subfields of the PT DLPDU

Supported parameters						
Item number	Column letter →		A) Receiving and recognizing		B) Forming and sending	
	Parameter	Reference	Status	Support	Status	Support
1	priority designator	5.2.1.2,				
1.1	URGENT	6.15.1.1	M		M	
1.2	NORMAL		M		M	
1.3	TIME-AVAILABLE		M		M	
2	DL address format: NODE destination, omitted source	5.2.1.1, 6.15.1.2	M (16: 255)		M (16: 255)	
3	DD-parameters	6.15.1.3, 6.15.2	M		M	

11.10.4.16 Specific fields and subfields of the ES DLPDU

Supported parameters						
Item number	Column letter →		A) Receiving and recognizing		B) Forming and sending	
	Parameter	Reference	Status and range	Support	Status and range	Support
1	address size designator	5.2.1.1,				
1.1	LONG	6.17.1.1	C1		C1	
1.2	SHORT		M		M	
2	restart/continue designator	5.2.1.3, 6.17.1.1				
2.1	RESTART		M		M	
2.2	CONTINUE		M		M	
3	DL address formats:	5.2.2,				
3.1	format 1L – LONG destination, omitted source	6.17.1.1, 6.17.1.2	M		M	
3.2	format 1S – SHORT destination, omitted source		M		M	
4	DD-parameters	6.17.1.3	M		M	
C1 : if 11.9.1/1 then M else X						

11.10.4.17 Specific fields and subfields of the RT DLPDUs

The RT DLPDU has no variable fields or subfields.

11.10.4.18 Specific fields and subfields of the RI DLPDU

Supported parameters						
Item number	Column letter →		A) Receiving and recognizing		B) Forming and sending	
	Parameter	Reference	Status	Support	Status	Support
1	DD-parameters	6.18.1.3	M		M	

11.10.4.19 Specific fields and subfields of the CL DLPDU

Supported parameters						
	Column letter →		A) Receiving and recognizing		B) Forming and sending	
Item number	Parameter	Reference	Status and range	Support	Status and range	Support
1	DL address format: omitted destination, NODE source	5.2.2.3, 6.19.1.2	M (16: 255)		M (16: 255)	

11.10.4.20 Specific fields and subfields of the TL DLPDU

Supported parameters						
	Column letter →		A) Receiving and recognizing		B) Forming and sending	
Item number	Parameter	Reference	Status and range	Support	Status and range	Support
1	DL address format: NODE destination, omitted source	5.2.2.3, 6.20.1.2	M (16: 255)		M (16: 255)	
Item number	Parameter	Reference	Range (in octets)	Supported range	Range (in octets)	Supported range
2	permitted octets of SPDU in DLPDU	5.2.4, 6.20.1.4	0 : 64 M		0 : 64 M	

11.10.4.21 Specific fields and subfields of the WK DLPDU

Supported parameters						
	Column letter →		A) Receiving and recognizing		B) Forming and sending	
Item number	Parameter	Reference	Status and range	Support	Status and range	Support
1	DL address format: NODE destination, omitted source	5.2.2.3, 6.21.1.2	M (16: 255)		M (16: 255)	

11.10.4.22 Specific fields and subfields of the Idle DLPDU

Supported parameters						
	Column letter →		A) Receiving and recognizing		B) Forming and sending	
Item number	Parameter	Reference	Range (in octets)	Supported range	Range (in octets)	Supported range
1	permitted octets of user data in DLPDU	5.2.4, 6.22.1.4	0 : 64 O		0 : 64 O	

11.11 Major high-level capabilities

This subclause identifies many of the major conformance requirements of the implementation. The conformance requirements specified in this subclause are those related to the higher-level medium-access-independent functions of the DL-protocol. Subclause 11.10 specifies the conformance requirements for the lower-level functions of the protocol – those which are related to the underlying medium-access functions.

11.11.1 DLCEP classes implemented

The fieldbus Data Link layer protocol defines four major classes of DLCEP operation, each a superset of the prior class. The support requirements for these classes differ.

DLCEP classes implemented					
Item number	**DLCEP class**	**DLC profiles**	**Reference**	**Status**	**Support**
1	UNORDERED peer, publisher and subscriber DLCs	A, B, C, D	10.2, 13.9.2.2	M	
2	ORDERED peer, publisher and subscriber DLCs	A, B, C	10.2, 13.9.2.2	O	
3	CLASSICAL peer DLCs	A, B	10.2, 13.9.2.2	C1	
4	CLASSICAL publisher and subscriber DLCs	A	10.2, 13.9.2.2	C2	
C1 : support of this class requires support of ORDERED peer, publisher and subscriber DLCEPs					
C2 : support of this class requires support of CLASSICAL peer DLCEPs					

11.11.2 Parameters of the EC, DC, RC, CA, CD, ED and DT DLPDUs

This subclause enumerates the ranges and states the static conformance requirements for each variable parameter of each EC, DC, RC, CA, CD, ED and DT DLPDU. Conformance is only required on those DLPDUs which were claimed under 11.10.3.1 and 11.10.3.2, and then only for the direction(s) – receiving and recognizing, or forming and sending, or both – which were claimed. Thus any conditions or options specified in the relevant entry or entries in 11.10.3.1 and 11.10.3.2 apply uniformly to all of the DLPDU's specific requirements, as detailed in the following subclauses, for the corresponding direction of use. Requirements in the following DLPDU-specific subclauses for directions of use not claimed under either 11.10.3.1 or 11.10.3.2 can be ignored.

11.11.2.1 Communications paradigms implemented

The fieldbus Data Link Layer protocol specifies a number of communications paradigms. The support requirements for many of these paradigms differ.

Communications paradigms implemented				
Item number	**Communications paradigm**	**Reference**	**Status**	**Support**
1	Unacknowledged Connectionless	6.5.2, 10.3	M	
2	Acknowledged Connectionless	6.5.2, 10.3	C1	
3	Connectionless Exchange	6.5.2, 10.3	C2	
4	Unordered Connection as Peer	6.5.3, 10.2,13.9.2.2	M	
5	Ordered Connection as Peer	6.5.3, 10.2,13.9.2.2	C3	
6	Classical Connection as Peer	6.5.3, 10.2,13.9.2.2	C4	
7	Disordered Connection as Peer	6.5.3, 10.2,13.9.2.2	C6	
8	Unordered Connection as Publisher	6.5.3, 10.2,13.9.2.2	M	
9	Ordered Connection as Publisher	6.5.3, 10.2,13.9.2.2	C3	
10	Classical Connection as Publisher	6.5.3, 10.2,13.9.2.2	C5	
11	Disordered Connection as Publisher	6.5.3, 10.2,13.9.2.2	C7	
12	Unordered Connection as Subscriber	6.5.3, 10.2,13.9.2.2	M	
13	Ordered Connection as Subscriber	6.5.3, 10.2,13.9.2.2	C3	
14	Classical Connection as Subscriber	6.5.3, 10.2,13.9.2.2	C5	
15	Disordered Connection as Subscriber	6.5.3, 10.2,13.9.2.2	C7	
C1 : if 11.9.2.7/3 then X else M.				
C2 : if 11.9.2.7/1 then M else X.				
C3 : if DLC_conformance_class = {A, B, or C} then M else O.				
C4 : if DLC_conformance_class = {A or B} then M else O.				
C5 : if DLC_conformance_class = A then M else O.				
C6 : if C4 then O else X.				
C7 : if C5 and C6 then M else X.				

11.11.2.2 Support for implied QoS for data link connections (DLCs)

Some aspects of data link connections also impact on conformance requirements.

Support for implied QoS (Quality of Service) parameters for data link connections							
Item number	Column letter →			A) Receiving data		B) Sending data	
	QoS parameter	Reference	Allowed range	Status	Support	Status	Support
1	segmentation – what is the ratio of the maximum DLSDU size to the maximum DLPDU size?	11.9.2.2	1 : 16	C1		C1	
2	what is the maximum window size on CLASSICAL DLCs?	7.1 Table 35	1 : 15	C2		C2	
C1 : if DLC_conformance_class = {A, B, or C} then 1:16 M else 1 M; the same value must be specified for both columns.							
C2 : if DLC_conformance_class = {A or B} then M else O; the same value must be specified for both columns.							

11.11.2.3 Parameters of the EC DLPDU – protocol version 1

11.11.2.3.1 DLC basic attributes

Supported parameters						
Item number	Column letter →		A) Receiving and recognizing		B) Forming and sending	
	Parameter	Reference	Status or range	Support	Status or range	Support
1	DLC protocol version number	9.1 a4	1 M		1 M	
2	reply requested	9.1 a1	F, T M		F, T M	
3	publisher-DLCEP-address reuse-discriminator	9.1 a2	F, T M		F, T M	
4	DLL path diversity	9.1 a3				
4.1	ANY-PATH		M		M	
4.2	THIS-PATH		M		M	
5	DLCEP class	9.1 a5				
5.1	peer DLCEP		M		M	
5.2	publisher DLCEP		M		M	
5.3	subscriber DLCEP		M		M	
6	DLL priority	9.1 a6				
6.1	URGENT		M		O	
6.2	NORMAL		M		O	
6.3	TIME-AVAILABLE		M		M	
7	DL address size	9.1 a7				
7.1	LONG		M		M	
7.2	SHORT		M		O	
7.3	VERY-SHORT		M		O	
8	DLPDU authentication	9.1 a8				
8.1	MAXIMAL		M		M	
8.2	SOURCE		M		M	
8.3	ORDINARY		M		M	
9	maximum DL-CONNECT, DL-RESET and DL-SUBSCRIBER-QUERY confirm delay	9.1 b				
9.1	UNLIMITED (encoded as 0xFFFF)		65 535 M		65 535 M	
9.2	minimum delay required (in ms)		1 : 60 000		1 : 60 000	
9.3	maximum delay supported (in ms)		1 : 60 000 M		1 : 60 000 M	
10	maximum DL-DATA confirm delay	9.1 b				
10.1	UNLIMITED (encoded as 0xFFFF)		65 535 M		65 535 M	

Supported parameters						
	Column letter →		A) Receiving and recognizing		B) Forming and sending	
Item number	Parameter	Reference	Status or range	Support	Status or range	Support
10.2	minimum delay required (in ms)		1 : 60 000		1 : 60 000	
10.3	maximum delay supported (in ms)		1 : 60 000 M		1 : 60 000 M	

11.11.2.3.2 DLC attributes as sender

Supported parameters						
	Column letter →		A) Receiving and recognizing		B) Forming and sending	
Item number	Parameter	Reference	Status or range	Support	Status or range	Support
1	DLC data delivery features	9.1 c1				
1.1	UNORDERED – peer, publisher and subscriber		M		M	
1.2	ORDERED – peer, publisher and subscriber		M		C1	
1.3	CLASSICAL – peer		M		C2	
1.4	DISORDERED – peer		M		O	
1.5	CLASSICAL – publisher and subscriber		M		C3	
1.6	DISORDERED – publisher and subscriber		M		C4	
2	DLC residual activity required	9.1 c2				
2.1	FALSE (no background activity)		M		M	
2.2	TRUE (background activity)		M		M	
3	DLSDU storage model at DLCEP	9.1 c3				
3.1	QUEUE		M		M	
3.2	BUFFER		M		M	
4	DLC maximum window size	9.1 c4	0:15 M		0:15 M1	
5	SD-parameter basic-DLC-parameters format	9.1 c5				
5.1	A – NULL format		M		C1	
5.2	B – SHORT format		M		C1	
5.3	C – LONG format		M		C1	
5.4	D – UNSEGMENTED LONG format		M		C1	
5.5	E – SUBSCRIBER format		M		C1	
5.6	F – PUBLISHER format		M		C1	
5.7	G – UNSEGMENTED PUBLISHER format		M		C1	
6	2-way data exchange permitted	9.1 c6				
6.1	FALSE		M		M	
6.2	TRUE		M		O	
7	timeliness included in DLPDU	9.1 c8				
7.1	FALSE		M		M	
7.2	TRUE		M		C5	
8	SD-parameter time-stamp format	9.1 c9				
8.1	J – NULL format		M		M	
8.2	K – two-octet format		M		C6	
8.3	L – three-octet format		M		C6	
8.4	M – six-octet format		M		C6	
9	DLC maximum DLSDU size	9.1 c10	0:4096 M		0:4096 M2	

C1 : if DLC_conformance_class = {A, B, or C} then M else O. C2 : if DLC_conformance_class = {A or B} then M else O. C3 : if DLC_conformance_class = A then M else O.	
C4 : if C3 and disordered_peer then M else O.	disordered_peer = /1.4
C5 : if DLC_conformance_class = {A or B} then M else O. C6 : if DLC_conformance_class = A then M else O. M1 : must be 1:*N*, where *N* ≥ the value claimed in 11.11.2.2/2 M2 : must be 1:*N*, where *N* = 256 times the value claimed in 11.11.2.2/1	

11.11.2.3.3 DLC attributes as receiver

Supported parameters						
	Column letter →		A) Receiving and recognizing		B) Forming and sending	
Item number	Parameter	Reference	Status or range	Support	Status or range	Support
1	DLC data delivery features	9.1 c1				
1.1	UNORDERED – peer, publisher and subscriber		M		M	
1.2	ORDERED – peer, publisher and subscriber		M		C1	
1.3	CLASSICAL – peer		M		C2	
1.4	DISORDERED – peer		M		O	
1.5	CLASSICAL – publisher and subscriber		M		C3	
1.6	DISORDERED – publisher and subscriber		M		C4	
2	DLC residual activity required	9.1 c2				
2.1	FALSE (no background activity)		M		M	
2.2	TRUE (background activity)		M		M	
3	DLSDU storage model at DLCEP	9.1 c3				
3.1	QUEUE		M		M	
3.2	BUFFER		M		M	
4	DLC maximum window size	9.1 c4	0:15 M		0:15 M1	
5	SD-parameter basic-DLC-parameters format	9.1 c5				
5.1	A – NULL format		M		C1	
5.2	B – SHORT format		M		C1	
5.3	C – LONG format		M		C1	
5.4	D – UNSEGMENTED LONG format		M		C1	
5.5	E – SUBSCRIBER format		M		C1	
5.6	F – PUBLISHER format		M		C1	
5.7	G – UNSEGMENTED PUBLISHER format		M		C1	
6	2-way data exchange permitted	9.1 c6				
6.1	FALSE		M		M	
6.2	TRUE		M		O	
7	timeliness included in DLPDU	9.1 c8				
7.1	FALSE		M		M	
7.2	TRUE		M		C5	
8	SD-parameter time-stamp format	9.1 c9				
8.1	J – NULL format		M		M	
8.2	K – two-octet format		M		C6	
8.3	L – three-octet format		M		C6	
8.4	M – six-octet format		M		C6	

Supported parameters						
	Column letter →		A) Receiving and recognizing		B) Forming and sending	
Item number	Parameter	Reference	Status or range	Support	Status or range	Support
9	DLC maximum DLSDU size	9.1 c10	0:4096 M		0:4096 M2	

C1 : if DLC_conformance_class = {A, B, or C} then M else O.
C2 : if DLC_conformance_class = {A or B} then M else O.
C3 : if DLC_conformance_class = A then M else O.
C4 : if C3 and disordered_peer then M else O. disordered_peer = /1.4
C5 : if DLC_conformance_class = {A or B} then M else O.
C6 : if DLC_conformance_class = A then M else O.
M1 : shall be 1:N, where $N \geq$ the value claimed in 11.11.2.2/2.
M2 : shall be 1:N, where N = 256 times the value claimed in 11.11.2.2/1.

11.11.2.4 Parameters of the DC DLPDU – protocol version 1

Supported parameters						
	Column letter →		A) Receiving and recognizing		B) Forming and sending	
Item number	Parameter	Reference	Status or range	Support	Status or range	Support
1	DLC protocol version number	7.2 a3	1 M		1 M	
2	reply requested	7.2 a1	F, T M		F, T M	
3	reason for requested action	7.2 b	M		M	

11.11.2.5 Parameters of the RC DLPDU – protocol version 1

Supported parameters						
	Column letter →		A) Receiving and recognizing		B) Forming and sending	
Item number	Parameter	Reference	Status or range	Support	Status or range	Support
1	DLC protocol version number	7.3 a, 7.2.a3	1 M		1 M	
2	reply requested	7.3 a, 7.2.a1	F, T M		F, T M	
3	reason for requested action	7.3 b	M		M	
4	sequence number residue	7.3 c	M		M	

11.11.2.6 SD-parameters of the CA, CD, DT and ED DLPDUs – protocol version 1

11.11.2.6.1 Unitdata transaction-support parameters

Supported parameters						
	Column letter →		A) Receiving and recognizing		B) Forming and sending	
Item number	Parameter	Reference	Status	Support	Status	Support
1	SD-parameter connectionless-mode parameters formats	7.4.1				
1.1	format P – NULL	7.4.1 a	M		M	
1.2	format R – INITIATOR	7.4.1 b	M		C1	
1.3	format U – RESPONDER	7.4.1 c	C1		M	

C1 if 11.9.2.7/3 then X else M.

11.11.2.6.2 Basic DLC parameters

Supported parameters						
Item number	Column letter →		A) Receiving and recognizing		B) Forming and sending	
	Parameter	Reference	Status	Support	Status	Support
1	SD-parameter basic-DLC-parameters formats	9.4.2.1				
1.1	format A – NULL	9.4.2.1 A	M		M	
1.2	format B – SHORT	9.4.2.1 B	C1		C1	
1.2.1	retransmit	9.4.2.1 B1	F, TC1		F, TC1	
1.2.2	acknowledge	9.4.2.1 B2	F, TC1		F, TC1	
1.2.3	timeliness	9.4.2.1 B3	F, TC1		F, TC1	
1.2.4	modulus number of post-acknowledged or requested DLSDU	9.4.2.1 B4	C1		C1	
1.2.5	modulus number of associated DLSDU or of highest-numbered DLSDU sent	9.4.2.1 B5	C1		C1	
1.3	format C – LONG	9.4.2.1 C	C1		C1	
1.3.1	modulus number of post-acknowledged or requested DLSDU	9.4.2.1 C1	C1		C1	
1.3.2	(zero-origin) requested segment number	9.4.2.1 C2	0:N C1		C1	
1.3.3	retransmit	9.4.2.1 C3	F, TC1		F, TC1	
1.3.4	acknowledge	9.4.2.1 C4	F, TC1		F, TC1	
1.3.5	timeliness	9.4.2.1 C5	F, TC1		F, TC1	
1.3.6	modulus number of associated DLSDU or of highest-numbered DLSDU sent	9.4.2.1 C6	C1		C1	
1.3.7	(zero-origin) total number of segments in DLSDU	9.4.2.1 C7	0:N C1		C1	
1.3.8	(zero-origin) associated segment number	9.4.2.1 C8	0:N C1		C1	
1.4	format D – UNSEGMENTED LONG	9.4.2.1 D	C1		C1	
1.5	format E – SUBSCRIBER	9.4.2.1 E	C1		C1	
1.6	format F – PUBLISHER	9.4.2.1 F	C1		C1	
1.7	format G – UNSEGMENTED PUBLISHER	9.4.2.1 G	C1		C1	

C1 : if DLC_conformance_class = {A, B, or C} then M else O.

N = : (the value claimed in 11.11.2.2/2) minus 1.

11.11.2.6.3 Timeliness parameters

Supported parameters						
Item number	Column letter →		A) Receiving and recognizing		B) Forming and sending	
	Parameter	Reference	Status	Support	Status	Support
1	SD-parameter time-stamp format	9.4.2.2				
1.1	format J – no-time format	9.4.2.2 J	M		M	
1.2	format K – 2-second format	9.4.2.2 K	M		C1	
1.3	format L – 8-minute format	9.4.2.2 L	M		C1	
1.4	format M – full-time format	9.4.2.2 M	M		C1	
C1 if 11.9.2.5/2 and 11.9.2.6/2 then M else O.						

11.11.3 Parameters of the SR DLPDU– protocol version 1

Supported parameters						
Item number	Column letter →		A) Receiving and recognizing		B) Forming and sending	
	Parameter	Reference	Status	Support	Status	Support
1	SR-parameter formats	9.5				
1.1	format X – NULL	9.5 a	M		C1	
1.2	format Y – ERROR	9.5 b	M		C1	
C1 if potential_LAS then M else O			potential_LAS = 11.10.1/2 or /3			

11.11.4 Parameters of the TD DLPDU – protocol version 1

Supported parameters						
Item number	Column letter →		A) Receiving and recognizing		B) Forming and sending	
	Parameter	Reference	Status	Support	Status	Support
1	link originating DL-time	9.67.6 a	M		C2	
2	DL-time-quality measures	9.67.6 b	M		C2	
2.1	limiting time-synchronism class	9.67.6 b1	0:7 M		0:N C2	
2.2	number of intervening links	9.67.6 b2	0:7 M		0:7 C2	
2.3	source type	9.67.6 b3	0:3 M		0:3 C2	
3	DL-time offset	9.67.6 c	C1		C2	
4	DL-time prior to end-of-transmission	9.67.6 d	C1		C2	
5	DL-time adjustment	9.67.6 e	C1		C2	
C1 if time_synchronism_supported then M else O			time_synchronism_supported = 11.9.3.1/2 or /3 or /4 or /5 or /6 or /7 or /8			
C2 if potential_LAS then M else O			potential_LAS = 11.10.1/2 or /3			
N = if 11.9.3.1/1 then 0 elseif 11.9.3.1/2 then 1 elseif 11.9.3.1/3 then 2 elseif 11.9.3.1/4 then 3 elseif 11.9.3.1/5 then 4 elseif 11.9.3.1/6 then 5 elseif 11.9.3.1/7 then 6 else 7.						

11.11.5 Parameters of the RQ DLPDU – protocol version 1

Supported parameters						
	Column letter →		A) Receiving and recognizing		B) Forming and sending	
Item number	Parameter	Reference	Status	Support	Status	Support
1	sending time-stamp	7.7 a	M		C1	
C1	**if** potential_LAS **then** M **else** O		potential_LAS = 11.10.1/2 or /3			

11.11.6 Parameters of the RR DLPDU – protocol version 1

Supported parameters						
	Column letter →		A) Receiving and recognizing		B) Forming and sending	
Item number	Parameter	Reference	Status	Support	Status	Support
1	DL-time-quality measures	9.87.8 a	C1		M	
1.1	limiting time-synchronism class	9.87.8.a1	0:7 M		0:N C2	
1.2	support for estimated C(NT)	9.87.8.a2	C2		M	
2	RQ sending time-stamp	9.87.8 b	C1		M	
3	RQ receiving time-stamp	9.87.8 b	C1		M	
4	RR sending time-stamp	9.87.8 c	C1		M	
C1	if potential_LAS then M else O		potential_LAS = 11.10.1/2 or /3			
C2	if potential_LAS or N≠0 then M else O					
N =	if 11.9.3.1/1 then 0 elseif 11.9.3.1/2 then 1 elseif 11.9.3.1/3 then 2 elseif 11.9.3.1/4 then 3 elseif 11.9.3.1/5 then 4 elseif 11.9.3.1/6 then 5 elseif 11.9.3.1/7 then 6 else 7.					

11.11.7 Parameters of the PN DLPDU – protocol version 1

Supported parameters						
	Column letter →		A) Receiving and recognizing		B) Forming and sending	
Item number	Parameter	Reference	Status or range	Support	Status or range	Support
1	DLC protocol version number	9.9 a3	1 M		1 M	
2	PhL inter-channel signal skew	9.9 a1	0:7 M		0:7 C1	
3	PhL post-transmission-gap extension units	9.9 a4	0:7 M		0:7 C1	
4	PhL preamble extension units	9.9 a5	0:7 M		0:7 C1	
5	slot-time	9.9 b1	1:4095 M		1:4095 C1	
6	maximum-response-delay	9.9 b2	1:12 M		1:12 C1	
7	minimum-inter-PDU-delay	9.9 b3	0:120 M		0:120 C1	
C1	**if** potential_LAS **then** M **else** O		potential_LAS = 11.10.1/2 or /3			

11.11.8 DD parameters of the PT, ES and RI DLPDUs – protocol version 1

Supported parameters						
	Column letter →		A) Receiving and recognizing		B) Forming and sending	
Item number	Parameter	Reference	Status	Support	Status	Support
1	delegated or requested duration	9.10 a	0:65000 M		0:65000 M	

Annex A
(informative)

Exemplary FCS implementation

This annex provides an example implementation of FCS generation and FCS syndrome checking.

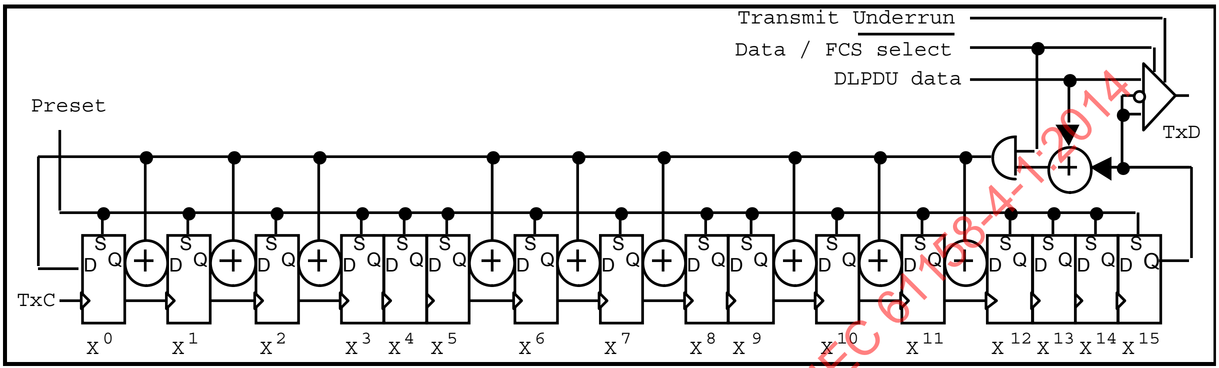


Figure A.1 – Example of FCS generation

In this example, shown in Figure A.1, the FCS is computed in a register consisting of 16 presettable master-slave flip-flops which are interconnected as a linear feedback shift register, with its least significant bit depicted on the left. The initial preset of the register before transmission serves to include the initial L(X) term in the FCS computation. Feedback is disabled during transmission of the FCS itself, and the FCS is transmitted complemented to provide the final L(X) inclusion in the FCS computation. Also shown is optional logic to inhibit the final complementation and transmit a massively incorrect FCS in the case of a transmitter underrun.

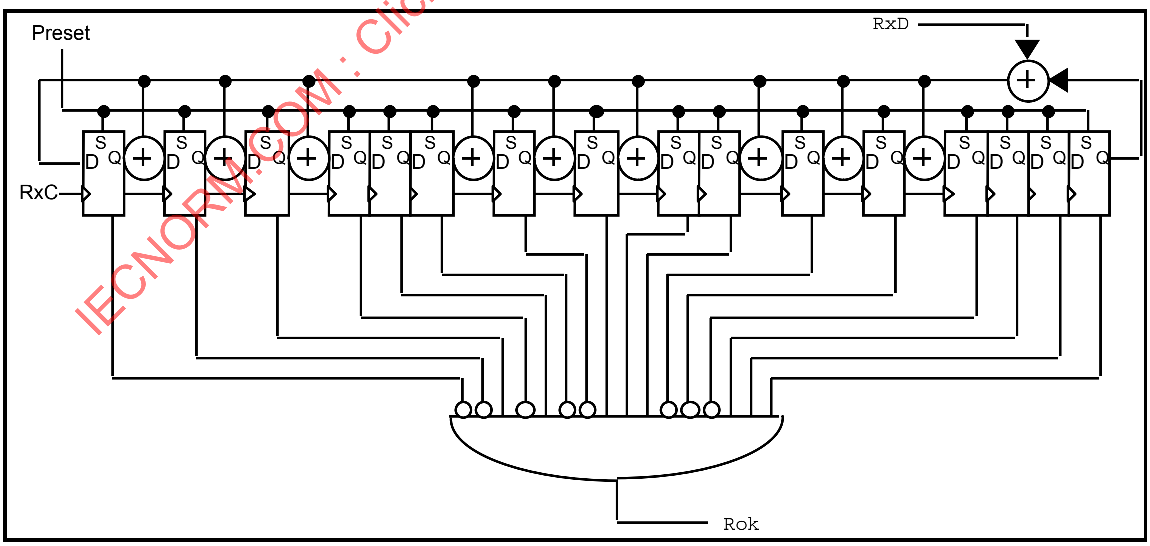


Figure A.2 – Example of FCS syndrome checking on reception

In this example, shown in Figure A.2, the residual FCS is computed in a similar register. The Q outputs of the 16 flip-flops are compared to the expected residual value by the 16-input “and” gate, half of whose inputs are complemented.

In this example, the FCS is computed in a register consisting of 16 presettable master-slave flip-flops which are interconnected as a linear feedback shift register, with its least significant bit depicted on the left. The initial pre-set of the register before transmission serves to include the initial L(X) term in the FCS computation. Feedback is disabled during transmission of the FCS it self, and the FCS is transmitted complemented to provide the final L(X) inclusion in the FCS computation. Also shown is optional logic to inhibit the final complementation and transmit a massively incorrect FCS in the case of a transmitter underrun.

Annex B
(informative)

Type 1: Formal protocol finite state machines

NOTE 1 This annex specifies a number of finite state machines used by the DLE to provide its low-level and high-level protocol functions. This specification is complementary to, and subordinate to, the textual specification in the body of this standard.

NOTE 2 The finite state machine descriptions given here are necessarily less than a complete description of an implementation. Additional requirements and considerations are found in the textual specification.

B.1 Basic reception and transmission FSMs

B.1.1 Nomenclature

X ⇐ Y assignment of value of expression Y to name-expression X

X ∧ Y logical AND of expressions X and Y with short-circuit evaluation (Y is evaluated only if X is TRUE)

X ∨ Y logical OR of expressions X and Y with short-circuit evaluation (Y is evaluated only if X is FALSE)

! Y logical NOT of value of expression Y

∃(X,Y) THERE EXISTS AN X SUCH THAT Y – which is true if an object of the type of name-expression X satisfying Boolean condition Y exists, and false otherwise.

If an object of the type of X satisfying Y exists, then one such object (usually the only one) is bound to the name-expression X for use throughout subsequent parts of the transition condition and the transition's actions.

X ∈{ Y … } CONTAINED IN – X is a member of the set of elements Y …

X ∉{ Y … } NOT CONTAINED IN – X is not a member of the set of elements Y …

X . Y field selection – the Y th component of name-expression X

(X)• pointer dereference (indirection) —the object to which name-expression X points

events, variables, procedures, and other FSM nomenclature

FCS_OK FCS condition at RxA\ of just-received DLPDU; false if no DLPDU just received

C(CT) 8-bit down-counter, prescaled by V(ST), for LAS measurement of periods of bus inactivity: "Token Recovery" – preset with P(TRD), 15, or V(TN)

C(RR) 8-bit down-counter, prescaled by V(ST), for token-holder measurement of periods of bus inactivity: "Reply Recovery" – preset with V(MRD)+1

C(FD) 8-bit down-counter, prescaled by V(ST), for responder measurement of reply-forming delay: "Forming Delay" – preset with V(MRD)

V(TL) 16-bit link identifier: "This Link"

V(TN) 8-bit node identifier: “This Node”

V(DTA) 32-bit DL-address used to identify the current token holder: “Delegated Token Address”

V(DA) 32-bit DL-address used to retain the last DLPDU’s “Destination Address” if a CA, CD, ED or RQ DLPDU, else null (invalid)

V(SA) 32-bit DL-address used to retain the last DLPDU’s “Source Address” if a CA, CD, ED or RQ DLPDU, else null (invalid)

V(LN) 8-bit node identifier used to record the node-id of the current LAS: “LAS Node”

V(TT) 8-bit FC as “Token Type”: TL, ES, PT, CA, CD, ED

NOTE CA, CD, ED are considered to be half-transaction tokens).

V(NTS) 16-bit node-time source link number

V(NTO) 56-bit DL-time offset with respect to sending DLE’s node-time

C(NT) 56-bit frequency-corrected up-counter: “Node Time”

C(UC) 16-bit down-counter: “Unscheduled Capacity” used by LAS to generate ES and PT DLPDUs

C(RD) 16-bit down-counter: “Remaining Duration” used by token-holder to limit use of ES-token

Q(LR) prioritized FIFO queue of unscheduled SPDUs for transmission to the local link’s LAS

QN(US) prioritized FIFO queue of unscheduled service requests for the DLE

QX.is_empty Boolean indicating that the specified queue is or is not empty

QX(Y,min=PP) subset of QX(Y) whose priority is PP or higher

rcv record holding parsed version of just-received DLPDU

next record holding parsed version of DLPDU in the selected transmit scheduling queue

reply record holding constructed immediate reply parameters

xxx.yyy field yyy of record xxx

xxx.FC.F Boolean field of FC field of xxx DLPDU which indicates final transaction of token use

xxx.FC.DA DA field is present in xxx DLPDU – determination based on FC value

xxx.FC.SA SA field is present in xxx DLPDU – determination based on FC value

xxx.DA.exists record xxx contains a DA field xxx.SA.exists record xxx contains an SA field

xxx.is_group variable xxx specifies a group DL-address

xxx.is_DLSAP_addr variable xxx specifies a DLSAP-address

xxx.is_DLCEP_id variable xxx specifies the DL-identifier of a DLCEP

is_usable(xxx) the implementation is capable of operation with the value of parameter xxx

falling_edge(xxx) event detecting falling (true → false) edge of signal xxx

dlm_queue_deliver report received DLPDU to DL-management for processing

dlm_event notify DL-management of specified event

process_xxx process specified type of received DLPDU

send_xxx(…) send specified DLPDU with specified explicit arguments and appropriate values for other arguments

send_item send specified DLPDU from parametric description, providing current values for the C(NT)-dependent arguments on TD, RQ and RR DLPDUs

local_addr DLE's address-recognition function – recognizes all but multi-peer subscriber DLCEPs

subscriber DLE's other address-recognition function – recognizes only multi-peer subscriber DLCEPs

create_object(X) create an object of the type of X and bind it to the name local name X

delete_object delete specified object

activate_addr_recogitionactivate recognition of the specified address

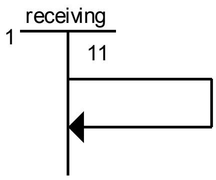
local_object local object addressed by specified DL-address, for which local_addr is true

FSM_state(FSM_name, FSM_state_number) Boolean condition, true if specified FSM is in specified state

FSM_transition(FSM_name, transition_state_number_pair) event, true when specified FSM executes the specified transition (from the first to the second specified state)

B.1.2 The receiver FSM

receiver FSM



receiver FSM

state	name	transitions	entry actions
1	receiving	11	

transition condition : [actions]

11a
rcv.FC=EC ∧ ((rcv.FC.DA ∧ local_addr(rcv.DA)
∨ (!rcv.FC.DA ∧ subscriber(rcv.SA))
: process_EC(rcv);
V(DA) ⇐ null;
V(SA) ⇐ null;

11b
rcv.FC=DC ∧ ((rcv.FC.DA ∧ local_addr(rcv.DA)
∨ (!rcv.FC.DA ∧ subscriber(rcv.SA))
: process_DC(rcv);
V(DA) ⇐ null;
V(SA) ⇐ null;

11c
rcv.FC=RC ∧ ((rcv.FC.DA ∧ local_addr(rcv.DA)
∨ (!rcv.FC.DA ∧ subscriber(rcv.SA))
: process_RC(rcv);
V(DA) ⇐ null;
V(SA) ⇐ null;

11d
rcv.FC=CA ∧ local_addr(rcv.DA)
: process_CA(rcv);
V(DA) ⇐ rcv.DA;
V(SA) ⇐ rcv.SA;

11e
rcv.FC=CD ∧ local_addr(rcv.DA)
: process_CD(rcv);
V(DA) ⇐ rcv.DA;
V(SA) ⇐ rcv.SA;

11f
rcv.FC=ED ∧ local_addr(rcv.DA)
: process_ED(rcv);
V(DA) ⇐ rcv.DA;
V(SA) ⇐ rcv.SA;

11g
rcv.FC ⇐ DT ∧ ((rcv.FC.DA ∧ local_addr(rcv.DA))
∨ (!rcv.FC.DA ∧ rcv.FC.SA ∧ subscriber(rcv.SA))
∨ (!rcv.FC.DA ∧ !rcv.FC.SA ∧ subscriber(V(DA)))
∨ (!rcv.FC.DA ∧ !rcv.FC.SA ∧ local_addr(V(SA))))
: process_DT(rcv);
V(DA) ⇐ null;
V(SA) ⇐ null;

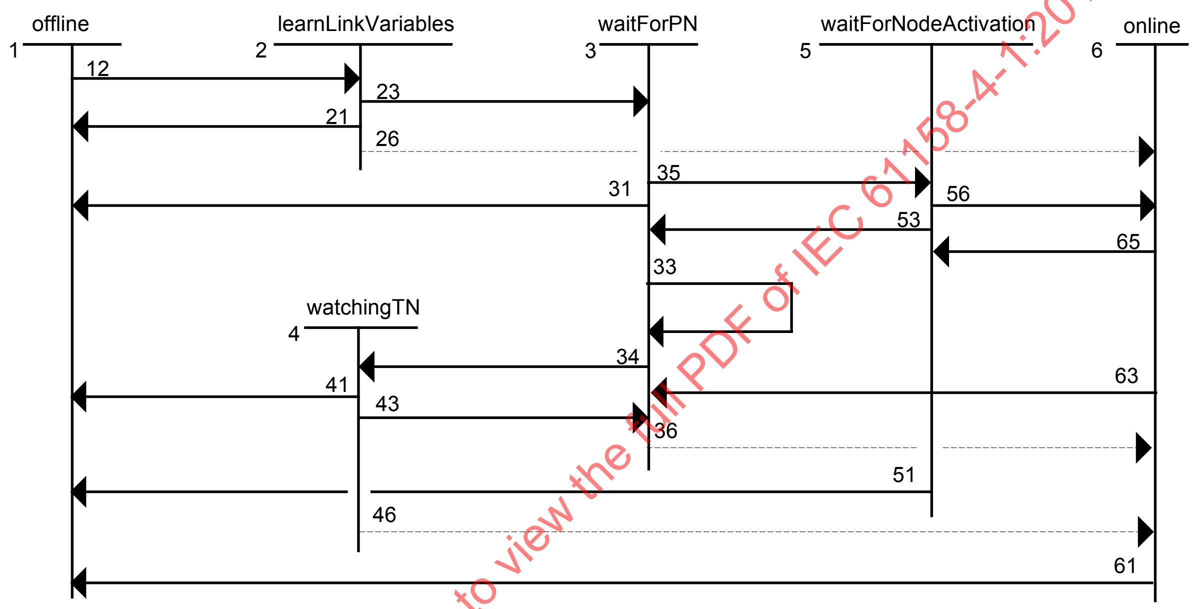
11h
rcv.FC=TD
: rcv.time_of_receipt ⇐ C(NT);
V(LN) ⇐ rcv.SA.N; /* track LAS node-id */
process_TD(rcv);
V(DA) ⇐ null;
V(SA) ⇐ null;

11j rcv.FC ∈{CL,TL}
: V(LN) ⇐ rcv.SA.N; /* track LAS node-id */
dlm_event(CL_or_TL_received);
V(DA) ⇐ null;
V(SA) ⇐ null;

11k rcv.FC=WK ∧ rcv.DA.N=V(TN)
: power_up_local_circuitry;
dlm_event(WK_received);
V(DA) ⇐ null;
V(SA) ⇐ null;

B.1.3 The DLE-state FSM

DLE-state FSM



<u>responding FSM</u>

state	name	transitions	entry actions
1	offline	12	
2	learnLinkVars	21, 23	
3	waitForPN	31, 33, 34, 35	**if** V(TN)≥x'F8' **then** $V(TN)_{1\text{-}0}$ ⇐ (random ∧ x'03');
4	watchingTN	41, 43	C(RR) ⇐ V(MRD)+1;
5	waitForNodeActivation	51, 53, 56	V(RID) ⇐ random; send PR(V(RID));
6	online	61, 65	

<u>transition</u> <u>condition : [actions]</u>

12 DLM_action("go online")
:

n1 DLM_action("go offline")
(2≤n≤6) :

23 rcv.FC=PN
: set essential PhL and DLL variables from PN-parameters;

33a rcv.FC=PT ∧ rcv.DA.N=V(TN) ∧ V(TN) ≥ x'F8'
:

33b rcv.FC=PN ∧ rcv.DA.N=V(TN) ∧ !(is_usable(V(MID)) ∧ is_usable(V(MRD)))
: V(RID) ⇐ random;
send PR(V(RID));

34 rcv.FC=PT ∧ rcv.DA.N=V(TN) ∧ V(TN) < x'F8'
:

35 rcv.FC=PN ∧ rcv.DA.N=V(TN) ∧ is_usable(V(MID)) ∧ is_usable(V(MRD))
:

43a rcv.FC=any
: V(TN) ⇐ x'F8';

43b C(RR)=0
:

53a rcv.FC ∈{PN,PT} ∧ rcv.DA.N=V(TN)
:

53b rcv.FC=DT ∧ !rcv.DA.is_group ∧ rcv.DA.N=V(TN) ∧ rcv.DA.S=00
∧ (rcv.data.SPDU_type≠"node activation"
∨ rcv.data.node_activation_SPDU.N(RID)≠V(RID))
:

56b rcv.FC=DT ∧ rcv.DA.N=V(TN) ∧ rcv.DA.S=00
∧ rcv.data.SPDU_type="node activation"
∧ rcv.data.node_activation_SPDU.N(RID)=V(RID))
: set remaining basic DLL variables from SPDU;

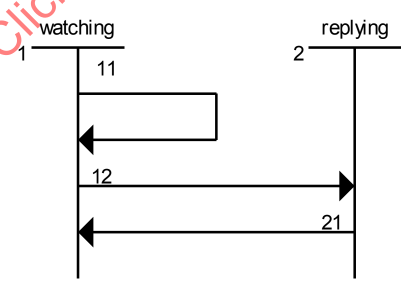
63 rcv.FC=PN ∧ rcv.DA.N=V(TN) ∧ !(is_usable(V(MID)) ∧ is_usable(V(MRD)))
:

65 rcv.FC=PN ∧ rcv.DA.N=V(TN) ∧ is_usable(V(MID)) ∧ is_usable(V(MRD))
:

***n*6a** DLE_is_LM ∧ DLE_knows_DLL_variables ∧ FSM_transition(link_master_FSM, 12)
(2≤***n***≤5) :

B.1.4 The responder FSM

responding FSM



<u>responding FSM</u>

state	name	transitions	entry actions
1	watching	11, 12	
2	replying	21	C(FD) ⇐ V(MRD)

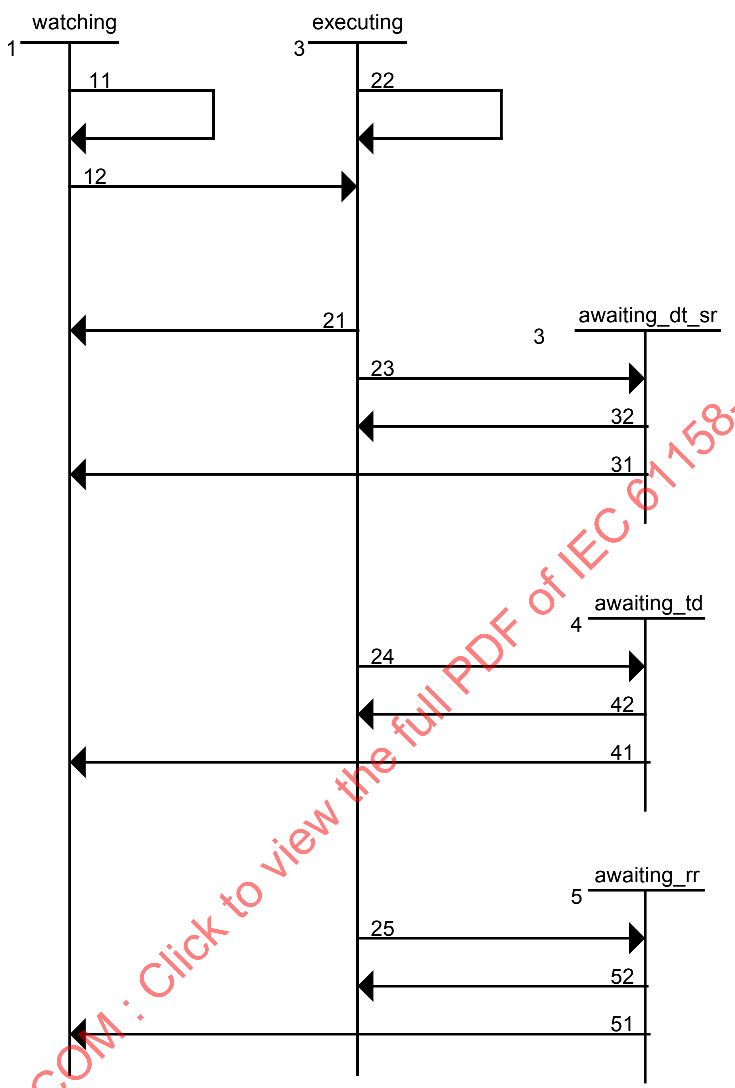
<u>transition</u> <u>condition : [actions]</u>

11a rcv.FC∉{CA,CD,ED,RQ} ∨ rcv.DA.is_group ∨ !local_addr(rcv.DA)
∨ rcv_data_length≠0
:

```
11b      rcv.FC ∈{CA,CD,ED} ∧ !rcv.DA.is_group ∧ local_addr(rcv.DA)
             ∧ rcv_data_length=0
             ∧ !(  local_object(rcv.DA).typ ∈{peer_DLCEP,publisher_DLCEP}
                 ∧ (!rcv.FC.SA ∨ rcv.SA=local_object(rcv.DA).remote_DLCEP_addr) )
         :   send_SR(V(TN));

12a      rcv.FC ∈{CA,CD,ED} ∧!rcv.DA.is_group ∧ local_addr(rcv.DA)
             ∧ (  local_object(rcv.DA).typ ∈{peer_DLCEP,publisher_DLCEP}
                 ∧ (!rcv.FC.SA ∨ rcv.SA=local_object(rcv.DA).remote_DLCEP_addr) )
         :   form_DT_reply per rcv.DA;

12b      rcv.FC=RR ∧ local_addr(rcv.DA)
         :   form_RR_reply;

21a      reply_formed
         :   send_item(next);

21b      C(FD)=0
         :

form_DT_reply per rcv.DA
  {   next ⇐ DT_reply_to(rcv.DA);
      if        Q_W(SS).is_empty ∧ !Q(LR).is_empty
      then      move first element of Q(LR) to end of appropriate priority of Q_W(SS);
      if        !Q_W(SS).is_empty
      then      next.FC.DD ⇐ priority(Q_W(SS).head);
  }

form_RR_reply
  {   next ⇐ RR_reply_to(rcv.DA);
      add value of C(NT) at the time of sending;
  }
```

B.1.5 The token-holder FSM

token holding FSM



token holding FSM

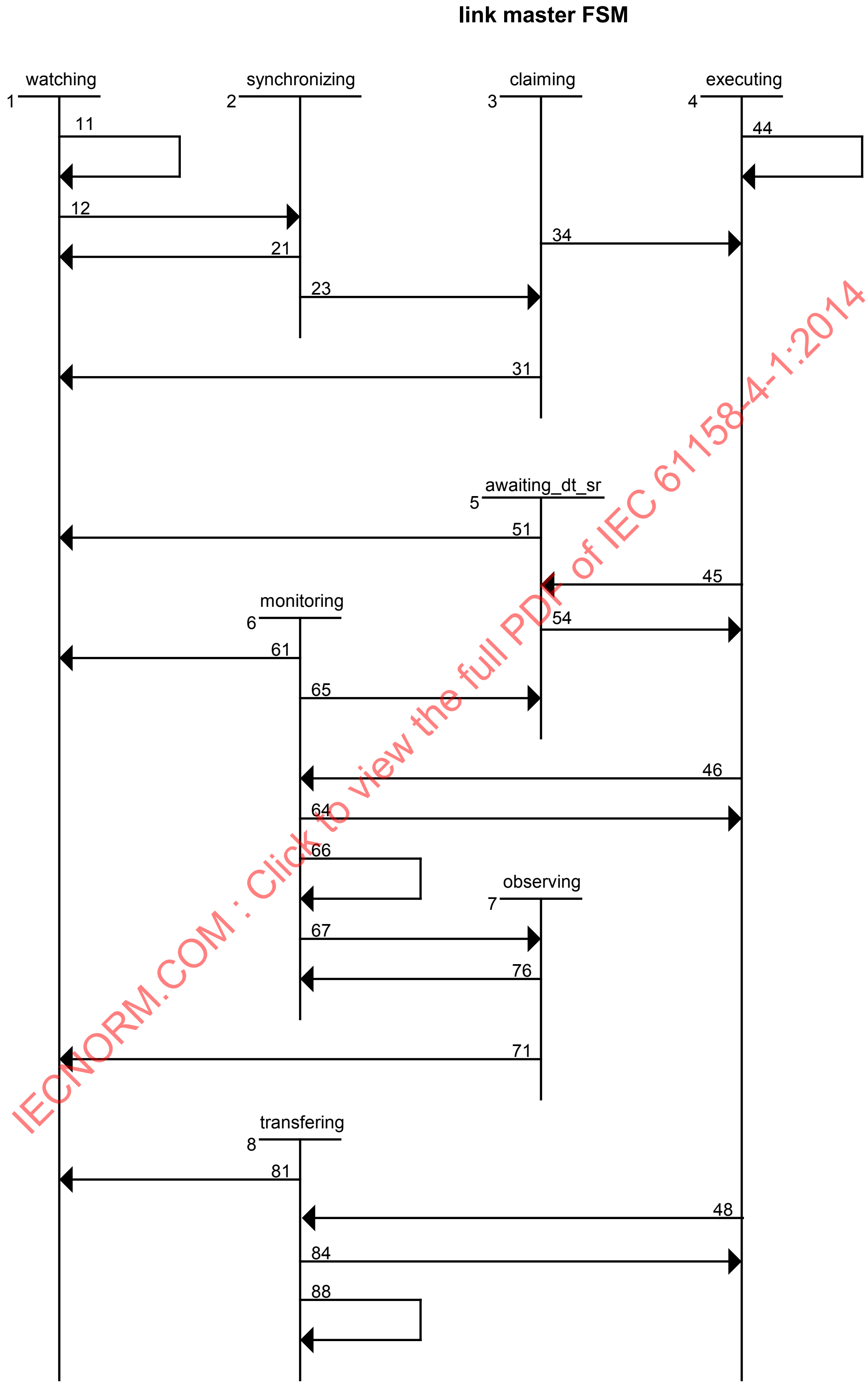
state	name	transitions	entry actions
1	watching	12	
2	executing	21, 22, 23, 24, 25	select_next
3	awaiting_dt_sr	31, 32	C(RR) ⇐ V(MRD)+1
4	awaiting_td	41, 42	C(RR) ⇐ V(MRD)+1
5	awaiting_rr	51, 52	C(RR) ⇐ V(MRD)+1

transition condition : [actions]

12a rcv.FC=PT $\wedge$ rcv.DA.N = TN
: V(TT) ⇐ rcv.FC;
C(RD) ⇐ rcv.PT_duration;
next.FC.F ⇐ NOT-FINAL;

```
12b     rcv.FC=ES ∧ !rcv.DA.is_group ∧ local_addr(rcv.DA) ∧
                local_object(rcv.DA).typ = DLSEP
            :   V(TT) ⇐ rcv.FC;
                C(RD) ⇐ rcv.ES_duration;
                next.FC.F ⇐ NOT-FINAL;

21      next.FC ∈{RI,RT} ∨ (next.FC ∈{EC,DC,RC,DT,IDLE} ∧ next.FC.F)
            :   send_item(next);

22      next.FC ∈{EC,DC,RC,DT,IDLE} ∧ !next.FC.F
            :   send_item(next);

23      next.FC ∈{CA,CD,ED}
            :   send_item(next);
                V(DA) ⇐ next.DA;
                V(SA) ⇐ next.SA;

24      next.FC=CT
            :   send_item(next);

25      next.FC=RQ
            :   send_item(next);

31      !(rcv.FC=SR ∨ (rcv.FC ∈{DT} ∧ !next.FC.F ∧ rcv.FC.DA ∧ rcv.DA=V(SA)))
            :

32a     rcv.FC=SR ∨ (rcv.FC ∈{DT} ∧ !next.FC.F ∧ rcv.FC.DA ∧ rcv.DA=V(SA))
            :

32b     C(RR)=0
            :

41      rcv.FC≠TD
            :   abort_token=TRUE;

42a     rcv.FC=TD
            :

42b     C(RR)=0
            :

51      rcv.FC≠RR ∨ rcv.DA≠V(SA)
            :   abort_token=TRUE;

52a     rcv.FC=RR ∧ rcv.DA=V(SA)
            :

52b     C(RR)=0
        :
```

B.1.6 The link-master FSM



LM FSM

state name transitions entry actions

1	watching	11, 12	C(CT) ⇐ V(TN)
2	synchronizing	21, 23	C(CT) ⇐ V(TN)
3	claiming	31, 34	C(CT) ⇐ random ∧ x'03'
4	executing	44, 45, 46, 48	select next per schedule
5	awaiting_dt_sr	51, 54	C(CT) ⇐ V(MRD)+1
6	monitoring	61, 64, 65, 66, 67	C(CT) ⇐ P(TRD)
7	observing	71, 76	C(CT) ⇐ P(TRD)
8	transferring	81, 84, 88	C(CT) ⇐ x'0F'

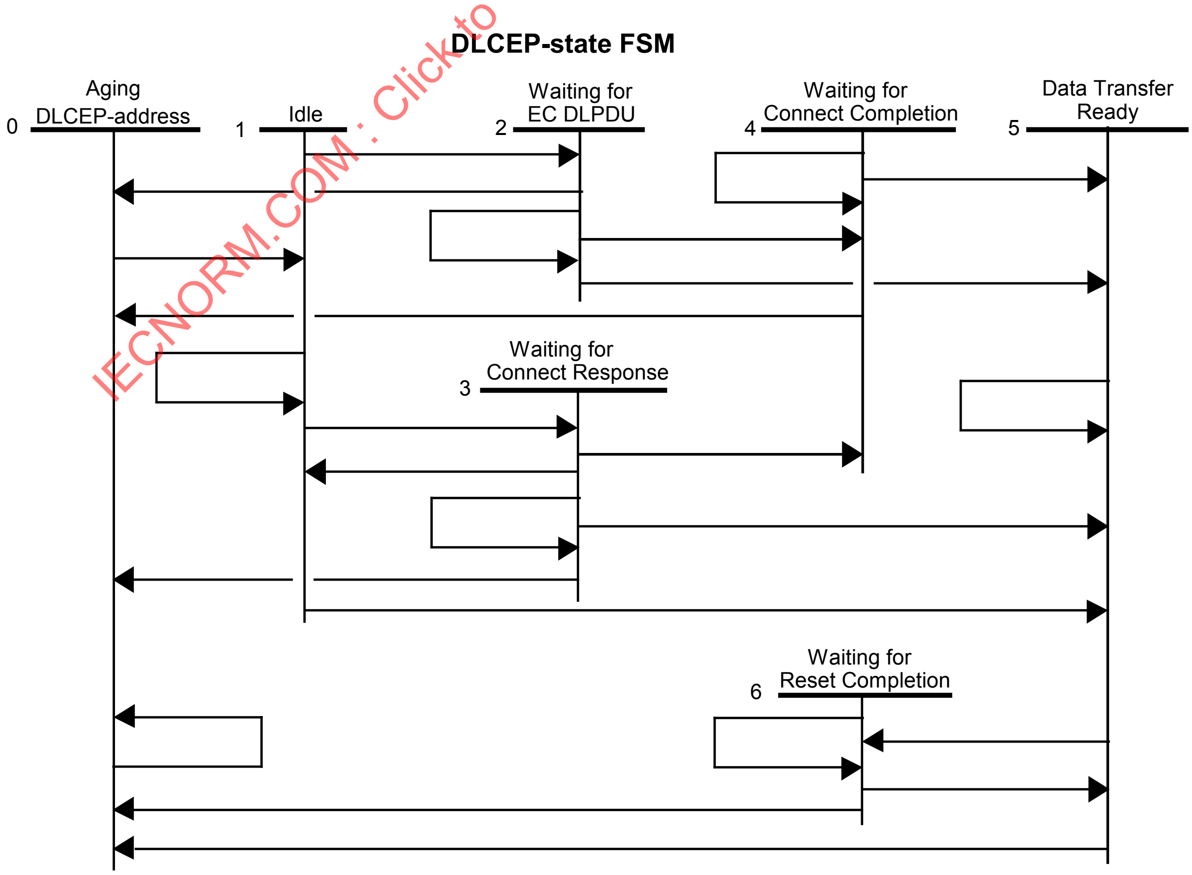
<u>transition</u> <u>condition : [actions]</u>

```
11      falling_edge(RxA)
            :

12      C(CT)=0 ∧ !FSM_state(link_master_FSM, 1)
            :   send_CL(V(TN));

21      falling_edge(RxA)
            :

23      C(CT)=0
            :   send_CL(V(TN));

31      falling_edge(RxA)
            :

34      C(CT)=0
            :

44      next.FC ∈{EC,DC,RC,DT,TD,WK,IDLE}
            :   send_item(next);

45      next.FC ∈{CA,CD,ED}
            :   send_item(next);
                V(TT) ⇐ next.FC;
                V(DA) ⇐ next.DA;

46a     next.FC=PT
            :   send_item(next);
                V(TT) ⇐ next.FC;
                V(DTA) ⇐ next.DA;

46b     next.FC=ES ∧ next.ES_duration≠0
            :   send_item(next);
                V(TT) ⇐ next.FC;
                V(DTA) ⇐ next.DA;

46c     next.FC=ES ∧ next.ES_duration=0
            :   next.ES_duration ⇐ C(UC); /* within the LAS' schedule an ES_duration */
                send_item(next);/*   of 0 requests a pseudo-circulated token */
                V(TT) ⇐ next.FC;
                V(DTA) ⇐ next.DA;

48      next.FC=TL
            :   send_item(next);
                V(TT) ⇐ next.FC;
                V(DTA) ⇐ next.DA;

51      falling_edge(RxA)
            ∧ !(rcv.FC=SR ∨ (rcv.FC=DT ∧ !rcv.FC.F ∧ (!rcv.FC.SA ∨ rcv.SA=V(DA))))
            :

54      rcv.FC=SR ∨ (rcv.FC=DT ∧ !rcv.FC.F ∧ (!rcv.FC.SA ∨ rcv.SA=V(DA)))
            :

61      rcv.FC ∈{WK,TD}
            :

64      rcv.FC ∈{EC,DC,RC,DT} ∧ rcv.FC.F
            :   V(DA) ⇐ rcv.DA;
```

65	rcv.FC ∊{CA,CD,ED} ∧ rcv.FC.F : V(DA) ⇐ rcv.DA;	
66a	rcv.FC ∊{EC,DC,RC,DT} ∧ !rcv.FC.F :	
66b	rcv.FC ∊{IDLE} :	
66c	rcv.FC=CT : send_TD(V(NTS),V(NTQ),V(NTO),C(NT));	
66d	C(CT)=0 : send_RT(V(TN));	
67	rcv.FC ∊{CA,CD,ED,RQ} ∧ !rcv.FC.F : V(DA) ⇐ rcv.DA;	
71	falling_edge(RxA) ∧ !(rcv.FC ∊{SR,RR} ∨ (rcv.FC=DT ∧ !rcv.FC.F ∧ (!rcv.FC.SA ∨ rcv.SA=V(DA)))) :	
76a	rcv.FC ∊{SR,RR} ∨ (rcv.FC=DT ∧ !rcv.FC.F ∧ (!rcv.FC.SA ∨ rcv.SA=V(DA))) :	
76b	C(CT)=0 :	
81	falling_edge(RxA) ∧ FCS_OK :	/* This is equivalent to rcv.FC in all */
84	C(CT)=0 :	
88	falling_edge(RxA) ∧ !FCS_OK :	

B.2 FSMs for DLCs

B.2.1 The DLCEP-state FSM



DLCEP-state FSM

state	name	transitions
0	agingDLCEPaddress	00, 01
1	idle	11, 12, 13, 15
2	waitingForEC_DLPDU	20, 22, 24, 25
3	waitingForConnectResponse	30, 31, 33, 34, 35
4	waitingForConnectCompletion	40, 44, 45
5	dataTransferReady	50, 55, 56
6	waitingForResetCompletion	60, 65, 66

NOTE Within states 2 to 6, non-hierarchical variable references are references to variables of the active DLCEP.

transition condition : [actions]

```
01          timeout
            :  delete_object(DLCEP);

11a         DL-CONNECT.request
                  ∧ ( (calling_id=DLCEP_id ∧ class≠PUBLISHER) ∨ !accepted_by_DLE)
            :  DL-DISCONNECT.indication;

11b         rcv.FC=EC ∧ !accepted_by_DLE ∧ rcv.DA_exists ∧ !rcv.DA.is_group
            :  send_DC(rcv.SA, rcv.DA, !reply_requested);

11c         rcv.FC=EC ∧ !accepted_by_DLE ∧ (!rcv.DA_exists ∨ rcv.DA.is_group)
            :

11d         rcv.FC=EC ∧ rcv.DA_exists ∧ (rcv.DA.is_DLSAP_addr ∨ rcv.DA.is_group)
                  ∧ ∃(DLCEP, DLCEP.remote_DLCEP_addr=rcv.SA)
                  ∧ DLCEP.remote_DLSAP_addr≠rcv.SA2
            :  send_DC(rcv.SA, rcv.DA, !reply_requested);

11e         rcv.FC=EC ∧ rcv.DA_exists ∧ (rcv.DA.is_DLSAP_addr ∨ rcv.DA.is_group)
                  ∧ ∃(DLCEP, DLCEP.remote_DLCEP_addr=rcv.SA)
                  ∧ DLCEP.remote_DLSAP_addr=rcv.SA2
            :

12          DL-CONNECT.request ∧ accepted_by_DLE ∧ class≠PUBLISHER
            :  create_object(DLCEP);
               DLCEP.DLCEP_addr ⇐ requested_or_new_DLCEP(req);
               DLCEP.DLSAP_addr ⇐ req.calling_DLSAP_addr;
               DLCEP.called_addr ⇐ req.called_addr;
               DLCEP.required_interactions ⇐ cnf;
               activate_addr_recogition(DLCEP.DLCEP_addr);
               if      DLCEP.called_addr≠UNKNOWN
               then    send_EC(DLCEP.called_addr, DLCEP.DLCEP_addr,
                                DLCEP.DLSAP_addr, reply_requested);
                       start_timer(DLCEP.send_params.max_confirm_delay_for_EC
                                  / (V(MRC)+1)  );

13          rcv.FC=EC ∧ rcv.DA_exists ∧ (rcv.DA.is_DLSAP_addr ∨ rcv.DA.is_group)
                  ∧ ! ∃(DLCEP, DLCEP.remote_DLCEP_addr=rcv.SA)
            :  create_object(DLCEP);
               DLCEP.remote_DLCEP_addr ⇐ rcv.SA;
               DLCEP.remote_DLSAP_addr ⇐ rcv.SA2;
               DLCEP.required_interactions ⇐ rsp_ind;
               DL-CONNECT.indication(DLCEP);
               start_timer(DLCEP, rcv.EC_p.send_params.max_confirm_delay_for_EC);
               --
               -- The following assignment is needed temporarily, until a DL-CONNECT.response is
               --   received, for use with any requested or internally-generated DC DLPDU.
               -- It will not be used if rcv.DA was a group DL-address.
               --
               DLCEP.DLCEP_addr ⇐ rcv.DA;
```

```
15      DL-CONNECT.request ∧ accepted_by_DLE ∧ class=PUBLISHER
        :  if    !req.DLCEP.calling_id.is_DLCEP_id
           then  create_object(DLCEP);
                 DLCEP.DLCEP_addr ⇐ requested_or_new_DLCEP(req);
                 activate_addr_recogition(DLCEP.DLCEP_addr);
                 DLCEP.DLSAP_addr ⇐ req.calling_DLSAP_addr;
                 DLCEP.called_addr ⇐ req.called_addr;
           if    req.called_addr=UNKNOWN
           then  send_EC(—, DLCEP.DLCEP_addr, DLCEP.calling_addr,
                         !reply_requested);
           else  send_EC(DLCEP.called_addr, DLCEP.DLCEP_addr,
                         DLCEP.DLSAP_addr, !reply_requested);
           DL-CONNECT.confirm;

20b     timeout ∧ (retry_count=max_retry_count) ∧ !called_addr.is_group
        :{see generic n0b transition}

20c     timeout ∧ (retry_count=max_retry_count) ∧ called_addr.is_group
        :{see generic n0c transition}

20e     rcv.FC=EC ∧ rcv.DA_exists ∧ rcv.DA=DLCEP_addr ∧ !accepted_by_DLE
              ∧ !called_addr.is_group
        :  DL-DISCONNECT.indication
           send_DC(rcv.SA, DLCEP_addr, !reply_requested);
           start_timer();

22a     timeout ∧ (retry_count < max_retry_count)
        :  send_EC(called_addr, DLCEP_addr, DLSAP_addr, reply_requested);
           start_timer(send_params.max_confirm_delay_for_EC / (V(MRC)+1) );

22b     rcv.FC=EC ∧ !accepted_by_DLE ∧ (!rcv.DA_exists ∨ called_addr.is_group)
        :  send_DC(rcv.SA, DLCEP_addr, !reply_requested);

24a     rcv.FC=EC ∧ rcv.DA_exists ∧ rcv.DA=DLCEP_addr ∧ accepted_by_DLE
              ∧ rcv.EC_p.reply_requested ∧ class=PEER
        :  remote_DLCEP_addr ⇐ rcv.SA;
           remote_DLSAP_addr ⇐ rcv.SA2;
           start_timer(send_params.max_confirm_delay_for_EC / (V(MRC)+1) );
           send_EC(remote_DLCEP_addr, DLCEP_addr, DLSAP_addr,
                   !reply_requested);

24b     merge_from_another_DLCEP
        :  start_timer(send_params.max_confirm_delay_for_EC / (V(MRC)+1) );
           send_EC(remote_DLCEP_addr, DLCEP_addr, DLSAP_addr,
                   !reply_requested);

25a     rcv.FC=EC ∧ rcv.DA_exists ∧ rcv.DA=DLCEP_addr ∧ accepted_by_DLE
              ∧ (class=SUBSCRIBER ∨ !rcv.EC_p.reply_requested)
        :  remote_DLCEP_addr ⇐ rcv.SA;
           remote_DLSAP_addr ⇐ rcv.SA2;
           stop_timer;
           if    class=PEER
           then  if    SA_is_required
                 then  send_DT(rcv.SA, DLCEP_addr);
                 else  send_DT(rcv.SA, —);
           DL-CONNECT.confirm

25b     rcv.FC=EC ∧ !rcv.DA_exists ∧ rcv.SA=DLCEP_addr ∧ class=SUBSCRIBER
              ∧ accepted_by_DLE
        :  remote_DLCEP_addr ⇐ rcv.SA;
           remote_DLSAP_addr ⇐ rcv.SA2;
           reset_seq_numbers_to_next_received ⇐ TRUE;
           DL-CONNECT.confirm

30b     (timeout ∨ (DL-CONNECT.response ∧ !accepted_by_DLE)) ∧ !called_addr.is_group
              ∧ class≠SUBSCRIBER
        :{see generic n0b transition}

30c     (timeout ∨ (DL-CONNECT.response ∧ !accepted_by_DLE))
              ∧ (called_addr.is_group ∨ class=SUBSCRIBER)
        :{see generic n0c transition}
```

```
31a   DL-CONNECT.response ∧ responding_id.is_DLCEP_id ∧ accepted_by_DLE
            ∧ responding_id.class=PEER
      :  (DLCEP.responding_addr)•.remote_DLCEP_addr
                  ⇐ DLCEP.remote_DLCEP_addr;
         (DLCEP.responding_addr)•.remote_DLSAP_addr
                  ⇐ DLCEP.remote_DLSAP_addr;
         signal   (DLCEP.responding_addr)•.DLCEP(merge_from_another_DLCEP);
         delete_object(DLCEP);

31b   DL-CONNECT.response ∧ responding_id.is_DLCEP_id ∧ accepted_by_DLE
            ∧ responding_id.class=PUBLISHER
      :  send_EC(remote_DLCEP_addr, (DLCEP.responding_addr)•.DLCEP_addr,
                  (DLCEP.responding_addr)•.DLSAP_addr, !reply_requested);
         delete_object(DLCEP);

33b   rcv.FC=EC ∧ !rcv.DA_exists ∧ rcv.SA_exists ∧ rcv.SA=remote_DLCEP_addr
            ∧ (rcv.publisher-DLCEP-address-reuse-discriminator
               =publisher-DLCEP-address reuse-discriminator)
      :

34    DL-CONNECT.response ∧ !responding_id.is_DLCEP_id ∧ accepted_by_DLE
            ∧ class=PEER
      :  DLCEP_addr ⇐ new_or_requested_DLCEP_addr(rsp);
         DLSAP_addr ⇐ rsp.calling_DLSAP_addr;
         activate_addr_recogition(DLCEP_addr);
         send_EC(remote_DLCEP_addr, DLCEP_addr, DLSAP_addr,
                  !reply_requested);
         start_timer(send_params.max_confirm_delay_for_EC / (V(MRC)+1)  );

35    DL-CONNECT.response ∧ !responding_id.is_DLCEP_id ∧ accepted_by_DLE
            ∧ class≠PEER
      :  DLCEP_addr ⇐ new_or_requested_DLCEP_addr(rsp);
         DLSAP_addr ⇐ rsp.calling_DLSAP_addr;
         stop_timer;
         activate_addr_recogition(DLCEP_addr);
         if      class=PUBLISHER
         then    send_EC(remote_DLCEP_addr, DLCEP_addr, DLSAP_addr,
                          !reply_requested);
         else    reset_seq_numbers_to_next_received ⇐ TRUE;
         DL-CONNECTION-ESTABLISHED.indication;

40b   timeout
      :{see generic n0b transition}

44b   rcv.FC=EC ∧ rcv.DA_exists ∧ rcv.EC_p.reply_requested
            ∧ (    rcv.DA=DLCEP_addr
              ∨ ( (rcv.DA.is_DLSAP_addr ∨ rcv.DA.is_group)
                ∧ ∃(DLCEP, DLCEP.remote_DLCEP_addr=rcv.SA)
                ∧ DLCEP.remote_DLSAP_addr=rcv.SA2 ) )
      :  send_EC(rcv.SA, DLCEP_addr, DLSAP_addr, !reply_requested);
         start_timer(send_params.max_confirm_delay_for_EC / (V(MRC)+1)  );

44c   timeout ∧ (retry_count < max_retry_count)
      :  send_EC(called_addr, DLCEP_addr, DLSAP_addr, reply_requested);
         start_timer(send_params.max_confirm_delay_for_EC / (V(MRC)+1)  );

44d   rcv.FC=EC ∧ !rcv.DA_exists ∧ rcv.SA_exists ∧ rcv.SA=remote_DLCEP_addr
            ∧ (rcv.publisher-DLCEP-address-reuse-discriminator
               =publisher-DLCEP-address reuse-discriminator)
      :

45a   rcv.FC=EC ∧ rcv.DA_exists ∧ rcv.DA=DLCEP_addr ∧ !rcv.EC_p.reply_requested
      :  stop_timer;
         if      SA_is_required
         then    send_DT(rcv.SA, DLCEP_addr);
         else    send_DT(rcv.SA, —);
         if      required_interactions=cnf
         then    DL-CONNECT.confirm;
         else    DL-CONNECTION-ESTABLISHED.indication;

45b   rcv.FC=DT
      :  stop_timer;
         if      required_interactions=cnf
         then    DL-CONNECT.confirm;
         else    DL-CONNECTION-ESTABLISHED.indication;
```

50c	rcv.FC=EC ∧ !rcv.DA_exists ∧ rcv.SA_exists ∧ rcv.SA=remote_DLCEP_addr ∧ (rcv.publisher-DLCEP-address-reuse-discriminator ≠publisher-DLCEP-address reuse-discriminator) : DL-DISCONNECT.indication; start_timer();
55b	rcv.FC=EC ∧ rcv.DA_exists ∧ rcv.DA=DLCEP_addr ∧ class=PUBLISHER ∧ accepted_by_DLE ∧ rcv.EC_p.reply_requested : send_EC(—, DLCEP_addr, DLSAP_addr, !reply_requested);
55c	rcv.FC=EC ∧ rcv.DA_exists ∧ rcv.DA=DLCEP_addr ∧ class=PUBLISHER ∧ !(accepted_by_DLE ∧ rcv.EC_p.reply_requested) : send_DC(rcv.SA, rcv.DA, !reply_requested);
55d	rcv.FC=EC ∧ rcv.DA_exists ∧ rcv.DA=DLCEP_addr ∧ class≠PUBLISHER :
55e	rcv.FC=EC ∧ !rcv.DA_exists ∧ rcv.SA_exists ∧ rcv.SA=remote_DLCEP_addr ∧ (rcv.publisher-DLCEP-address-reuse-discriminator =publisher-DLCEP-address reuse-discriminator) :
55f	rcv.FC ∈{CA,CD,ED} ∧rcv.DA=DLCEP_addr ∧ ((!rcv.SA_exists ∧ !SA_is_required) ∨ rcv.SA=remote_DLCEP_addr) : ...
55g	rcv.FC=DT ∧ rcv.DA_exists ∧ rcv.DA=DLCEP_addr ∧ ((!rcv.SA_exists ∧ !SA_is_required) ∨ rcv.SA=remote_DLCEP_addr) : ...
55h	rcv.FC=DT ∧ !rcv.DA_exists ∧ ((rcv.SA_exists ∧ rcv.SA=remote_DLCEP_addr) ∨ (rcv.SA_was_implied ∧ !SA_is_required ∧ rcv.implied_SA=remote_DLCEP_addr)) : ...
55j	DL-DATA.request ∨ DL-SUBSCRIBER-QUERY.request : ...
55k	timeout ∧ (retry_count < max_retry_count) : ...
55m	rcv.FC=RC ∧ rcv.DA_exists ∧ rcv.DA=DLCEP_addr ∧ class=PUBLISHER : send_DC(rcv.SA, rcv.DA, !reply_requested);
56a	rcv.FC=RC ∧ rcv.DA_exists ∧ rcv.DA=DLCEP_addr ∧ class=PEER ∧ rcv.SA=remote_DLCEP_addr : DL-RESET.indication; flush_internal_queues; required_interactions ⇐ rsp_ind; pended_DLPDUs ⇐ RC; expected_DLPDUs ⇐ DT; rcv_params.NDS ⇐ rcv.RC_p.NDS; rcv_params.NDR ⇐ rcv.RC_p.NDR; start_timer(send_params.max_confirm_delay_for_EC / (V(MRC)+1));
56b	rcv.FC=RC ∧ !rcv.DA_exists ∧ rcv.SA=DLCEP_addr ∧ class=SUBSCRIBER : DL-RESET.indication; flush_internal_queues; required_interactions ⇐ rsp_ind; pended_DLPDUs ⇐ none; expected_DLPDUs ⇐ none;

56c DL-RESET.request
: flush_internal_queues;
required_interactions ⇐ cnf;
pended_DLPDUs ⇐ none;
if class=PEER
then expected_DLPDUs ⇐ RC;
send_RC(remote_DLCEP_addr, DLCEP_addr, reply_requested,
V_C(M), V_C(L)+1);
start_timer(send_params.max_confirm_delay_for_EC / (V(MRC)+1));
elseif class=PUBLISHER
then expected_DLPDUs ⇐ none;
send_RC(—, DLCEP_addr, !reply_requested, V_C(N));
else expected_DLPDUs ⇐ none;

56d internal reset condition
: DL-RESET.indication;
flush_internal_queues;
required_interactions ⇐ rsp_ind;
pended_DLPDUs ⇐ none;
if class=PEER
then expected_DLPDUs ⇐ RC;
send_RC(remote_DLCEP_addr, DLCEP_addr, reply_requested,
V_C(M), V_C(L)+1);
start_timer(send_params.max_confirm_delay_for_EC / (V(MRC)+1));
elseif class=PUBLISHER
then expected_DLPDUs ⇐ none;
send_RC(—, DLCEP_addr, !reply_requested, V_C(N))
else expected_DLPDUs ⇐ none;

60b timeout ∧ (required_interactions=rsp
∨ (expected_DLPDUs≠none ∧ retry_count=max_retry_count))
:{see generic ***n***0b transition}

65 required_interactions≠rsp_ind ∧ expected_DLPDUs=none
: reset_seq_numbers_to_next_received ⇐ (class=SUBSCRIBER);
flush_internal_queues;
if required_interactions=cnf
then DL-RESET.confirm;
else DL-RESET-COMPLETED.indication;

66b rcv.FC=EC ∧ rcv.DA_exists ∧ rcv.DA=DLCEP_addr ∧ class=PUBLISHER
∧ rcv.EC_p.reply_requested
: expected_DLPDUs ⇐ none;
send_EC(rcv.SA, DLCEP_addr, DLSAP_addr, !reply_requested);

66c rcv.FC=EC ∧ rcv.DA_exists ∧ rcv.DA=DLCEP_addr
∧ (class≠PUBLISHER ∨ !rcv.EC_p.reply_requested)
:

66d rcv.FC=EC ∧ !rcv.DA_exists ∧ rcv.SA_exists ∧ rcv.SA=remote_DLCEP_addr
:

66e rcv.FC=RC ∧ rcv.DA_exists ∧ rcv.DA=DLCEP_addr ∧ class=PEER
∧ rcv.SA=remote_DLCEP_addr ∧ required_interactions≠rsp_ind
∧ rcv.RC_p.reply_requested
: **if** required_interactions=cnf
then expected_DLPDUs ⇐ RC;
else expected_DLPDUs ⇐ DT;
send_RC(rcv.SA, DLCEP_addr, !reply_requested, V_C(M), V_C(L)+1);
start_timer(send_params.max_confirm_delay_for_EC / (V(MRC)+1));

66f rcv.FC=RC ∧ rcv.DA_exists ∧ rcv.DA=DLCEP_addr ∧ class=PEER
∧ rcv.SA=remote_DLCEP_addr ∧ required_interactions≠rsp_ind
∧ expected_DLPDUs=RC ∧ !rcv.RC_p.reply_requested
: expected_DLPDUs ⇐ none;
if SA_is_required
then send_DT(rcv.SA, DLCEP_addr);
else send_DT(rcv.SA, —);

66g rcv.FC=RC ∧ rcv.DA_exists ∧ rcv.DA=DLCEP_addr ∧ class=PEER
∧ rcv.SA=remote_DLCEP_addr ∧ required_interactions=rsp_ind
: **if** rcv.RC_p.reply_requested
then pended_DLPDUs ⇐ RC;
else pended_DLPDUs ⇐ DT;

66h	rcv.FC=RC ∧ !rcv.DA_exists ∧ rcv.SA=DLCEP_addr ∧ class=SUBSCRIBER :
66j	rcv.FC=DT ∧ rcv.DA_exists ∧ rcv.DA=DLCEP_addr ∧ class=PEER ∧ ((!rcv.SA_exists ∧ !SA_is_required) ∨ rcv.SA=remote_DLCEP_addr) ∧ required_interactions≠rsp_ind ∧ expected_DLPDUs=DT; ∧ (rcv.SD_p.NDS ≥ rcv_params.NDS) : expected_DLPDUs ⇐ none;

66k	DL-RESET.response ∧ required_interactions=rsp_ind
	: required_interactions=ind;
	if pended_DLPDUs=RC
	then expected_DLPDUs ⇐ DT;
	send_RC(rcv.SA, DLCEP_addr, !reply_requested, V_C(M), V_C(L)+1);
	start_timer(send_params.max_confirm_delay_for_EC / (V(MRC)+1));
	elseif pended_DLPDUs=DT
	then expected_DLPDUs ⇐ none;
	if SA_is_required
	then send_DT(rcv.SA, DLCEP_addr);
	else send_DT(rcv.SA, —);

66m	timeout ∧ (expected_DLPDUs≠none) ∧ (retry_count < max_retry_count) ∧ required_interactions≠rsp_ind : **if** expected_DLPDUs=RC **then** send_RC(remote_DLCEP_addr, DLCEP_addr, !reply_requested, V_C(M), V_C(L)+1); **else** send_RC(remote_DLCEP_addr, DLCEP_addr, reply_requested, V_C(M), V_C(L)+1);
***n*0a** (2≤*n*≤6)	DL-DISCONNECT.request : **if** class=PEER **then** send_DC(remote_DLCEP_addr, DLCEP_addr, reply_requested); **elseif** class=PUBLISHER **then** send_DC(—, DLCEP_addr, !reply_requested); start_timer();
***n*0b** (2≤*n*≤6)	internal disconnect condition when a DC DLPDU is appropriate : **if** class=PEER **then** send_DC(remote_DLCEP_addr, DLCEP_addr, reply_requested); **elseif** class=PUBLISHER **then** send_DC(—, DLCEP_addr, !reply_requested); DL-DISCONNECT.indication; start_timer();
***n*0c** (2≤*n*≤3)	internal disconnect condition when a DC DLPDU is not appropriate : DL-DISCONNECT.indication; start_timer();
***n*0d** (2≤*n*≤6)	rcv.FC=DC ∧ rcv.DA=DLCEP_addr ∧ rcv.SA=remote_DLCEP_addr : **if** rcv.DC_p.reply_requested ∧ class=PEER **then** send_DC(remote_DLCEP_addr, DLCEP_addr, !reply_requested); DL-DISCONNECT.indication; start_timer();
***nn*a** (3≤*n*≤6) (or *n*=0)	rcv.FC ∈{EC,RC,CA,CD,ED,DT} ∧ rcv.DA_exists ∧ rcv.DA=DLCEP_addr ∧ rcv.SA_exists ∧ rcv.SA≠remote_DLCEP_addr ∧ class=PEER : send_DC(rcv.SA, DLCEP_addr, !reply_requested);

B.3 FSMs for scheduling

This is a future work item.

B.4 FSMs for bridges

B.4.1 Overview

This is a future work item.

B.4.2 The port-state FSM

See 11.4.6 and Clause 6 of ISO/IEC 10038:1993.

B.4.3 The bridge-state FSM

See 11.4.6 and Clause 6 of ISO/IEC 10038:1993.

B.4.4 The DLPDU forwarding FSM

This is a future work item.

B.4.5 The time distribution FSM

This is a future work item.

Annex C
(informative)

Type 1: DLPDU and DL-addressing short-form summaries

NOTE This annex provides a number of compact summaries of the various types of DLPDUs and DL-addresses.

C.1 Fields used in short-form summaries

C.1.1 Subfields of the frame control (FC) octet

PP DLPDU and transaction priority (01=URGENT, 10=NORMAL, 11=TIME-AVAILABLE)

L length of address fields in DLPDUs which have both a long and a short form (0=SHORT – two-octet node.selector with an implicit link=0000_{16},
1=LONG – four-octet link.node.selector)

F final transaction in sequence, explicitly returning the token to LAS at the end of the current transaction, : (0=NOT FINAL, 1=FINAL)

on sequence-initiating (ES) and token-passing (PT) DLPDUs a third meaning applies: (0= RESTART, 1=CONTINUE) sequence or repetitively-scheduled activities

C.1.2 Nomenclature for address and other fields of DLPDUs

– symbol indicating an omitted field

d- prefix indicating a destination address

s- prefix indicating a source address

o- prefix indicating an optional field (which is always the last non-omitted field in the DLPDU)

[PP] brackets indicating implied priority field

DLSAP a DL-service-access-point address, whose explicit length is specified by the FC's L subfield

DL(SAP) either a group DL-address or a DLSAP-address, whose explicit length is specified by the FC's L subfield

DLCEP a DL-address used as a DL-connection-end-point address, whose explicit length is specified by the FC's L subfield

DLCEP/SAP either a DLCEP-address or a DLSAP-address, whose explicit length is specified by the FC's L subfield

DLCEP/(SAP)either a group DL-address or a DLSAP-address or a DLCEP-address, whose explicit length is specified by the FC's L subfield

DLSEP a DL-address used as a DL-scheduling-end-point address, whose explicit length is specified by the FC's L subfield [d-DLCEP/SAP] brackets indicating implied d-DLCEP/SAP-address field, whose implied value is the same as that of the explicit or implied s-DLCEP/SAP-address field in the immediately prior DLPDU, which must have been a CA , CD or ED DLPDU.

[s-DLCEP/SAP] brackets indicating implied s-DLCEP/SAP-address field, whose implied value is the same as that of the explicit d-DLCEP/SAP-address field in the immediately prior DLPDU, which must have been a CA, CD or ED DLPDU.

C.1.3 Basic parameter and data fields of DLPDUs

link — individual link address (two octets), one component of LONG addresses; also used in time distribution (TD) DLPDUs

node — individual node address (one octet), one-component of SHORT and LONG addresses. Also used as a single octet address in some link-local DLPDUs used for inter-DLE coordination, where it is known as a node DL-address

DLSDU — complete Data Link Service Data Unit (non-null)

pDLSDU — partial or complete DLSDU (non-null)

SPDU — Support Protocol Data Unit (non-null) - a PDU used to convey scheduling and related DLS-support information between DLEs on the local or extended link.

SD-p — multi-octet SD (status and data-description) parameters of the CA, CD, ED and DT DLPDUs (see 7.4)

SR-p — parameters of the SR DLPDU (see 7.5)

EC-p — multi-octet parameters of the EC DLPDU (see 9.1)

DC-p — multi-octet parameters of the DC DLPDU (see 9.2)

RC-p — multi-octet parameters of the RC DLPDU (see 9.3)

TD-p — multi-octet parameters of the TD DLPDU (see 9.6)

RQ-p — multi-octet parameters of the RQ DLPDU (see 9.7)

RR-p — multi-octet parameters of the RR DLPDU (see 9.8)

PN-p — multi-octet parameters of the PN DLPDU (see 9.9)

DD-p — multi-octet parameters of the PT, ES and RI DLPDUs (see 9.10)

C.2 DLPDU short-form summary grouped by function

C.2.1 Connection-oriented DLPDUs

Establish Connection	1110 LF00	EC		s-DLCEP	s-DLSAP	EC-p	o-DLSDU
	1111 LF00	EC	d-DLCEP/(SAP)	s-DLCEP/SAP	s-DLSAP	EC-p	o-DLSDU
Disconnect Connection	0110 LF00	DC		s-DLCEP		DC-p	o-DLSDU
	0111 LF00	DC	d-DLCEP/(SAP)	s- DLCEP		DC-p	o-DLSDU
Reset Connection	0110 LFPP	RC		s-DLCEP		RC-p	o-DLSDU
	0111 LFPP	RC	d-DLCEP	s-DLCEP		RC-p	o-DLSDU

Compel Acknowledgment	1010 LFPP	CA	d-DLCEP	–	SD-p	o-pDLSDU
	1110 LFPP	CA	d-DLCEP	s-DLCEP/SAP	SD-p	o-pDLSDU
Compel Data	1011 LFPP	CD	d-DLCEP	–	o-SD-p	–
	1111 LFPP	CD	d-DLCEP	s-DLCEP/SAP	o-SD-p	–
Exchange Data	1000 LFPP	ED	d-DLCEP	–	SD-p	pDLSDU
	1100 LFPP	ED	d-DLCEP	s-DLCEP/SAP	SD-p	pDLSDU
Data	0101 0F00	DT [PP]		[s-DLCEP]	SD-p	o-pDLSDU
	0101 LFPP	DT		s-DLCEP	SD-p	o-pDLSDU
	1001 0F00	DT [PP]	[d-DLCEP/SAP]	–	SD-p	o- DLSDU
	1001 LFPP	DT	d-DLCEP	–	SD-p	o-pDLSDU
	1001 LFPP	DT	d-DLSAP	–	SD-p	o- DLSDU
	1101 LFPP	DT	d-DLCEP	s-DLCEP/SAP	SD-p	o-pDLSDU
Status Response	0001 0F11	SR	–	s-node	o-SR-p	–

C.2.2 Connectionless DLPDUs

Compel Acknowledgment	1110 LFPP	CA	d-DLSAP	s-DLSAP	SD-p	DLSDU
Compel Data	1111 LFPP	CD	d-DLSAP	s-DLSAP	SD-p	—
Exchange Data	1100 LFPP	ED	d-DLSAP	s-DLSAP	SD-p	DLSDU
Data	1001 0F00	DT [PP]	[d-DLSAP]	–	SD-p	o-DLSDU
	1001 LFPP	DT	d-DLSAP	–	SD-p	o-DLSDU
	1101 LFPP	DT	d-DL(SAP)	s-DLSAP	SD-p	o-DLSDU
Status Response	0001 0F11	SR	–	s-node	o-SR-p	–

NOTE The same FC values are used for connection-oriented and connectionless DLPDUs. In all cases, the structure of SD-parameters depends on characteristics of the DL-object addressed by the DLPDU's first explicit or implicit DL-address.

C.2.3 Time support DLPDUs

Compel Time	0001 0F00	CT	–	–	–	–
Time Distribution	0001 0F01	TD	–	s-node	TD-p	–
Round-trip-delay Query	1100 0F00	RQ	d-node.0	s-node.0	RQ-p	–
Round-trip-delay Reply	1101 0F00	RR	d-node.0	s-node.0	RR-p	–

C.2.4 Link management DLPDUs

Probe Node-address	0010 0110	PN	d-node	–	PN-p	–
Probe Response	0010 0111	PR	–	[s-node]	—	SPDU
Pass Token	0011 0FPP	PT	d-node	–	DD-p	–
Execute Sequence	1000 LF00	ES	d-DLSEP	–	DD-p	–
Return Token	0011 0100	RT	–	–	—	—
Request Interval	0010 0000	RI	–	–	DD-p	–
Claim LAS	0000 0001	CL	–	s-node	–	–
Transfer LAS	0000 0110	TL	d-node	–	TL-p	–
Status Response	0001 0F11	SR	–	s-node	o-SR-p	–
Wakeup	0000 0000	WK	d-node	–	–	–
Idle	0001 0F10	IDLE	–	–	—	o-DLSDU

C.2.5 Spare and unusable DLPDU code points

spare	0000 0x11
	0010 0x01
	00xx 1xxx
	0100 xxxx
	101x xx00
unusable	0000 0010
	0000 010x
	0010 001x
	0010 0100
	0011 0000
	0101 1x00
	1001 1x00
	110x 1x00

C.3 DLPDU short-form summary in alphabetic order of DLPDU names

Claim LAS	0000 0001	CL	–	s-node		–	–
Compel Acknowledgment	1010 LFPP	CA	d-DLCEP	–		SD-p	o-pDLSDU
	1110 LFPP	CA	d-DLCEP	s-DLCEP/SAP		SD-p	o-pDLSDU
	1110 LFPP	CA	d-DLSAP	s-DLSAP		SD-p	DLSDU
Compel Data	1011 LFPP	CD	d-DLCEP	–		o-SD-p	–
	1111 LFPP	CD	d-DLCEP	s-DLCEP/SAP		o-SD-p	–
	1111 LFPP	CD	d-DLSAP	s-DLSAP		SD-p	–
Compel Time	0001 0F00	CT	–	–		–	–
Disconnect Connection	0110 LF00	DC		s-DLCEP		DC-p	o-DLSDU
	0111 LF00	DC	d-DLCEP/(SAP)	s- DLCEP		DC-p	o-DLSDU
Data	0101 0F00	DT [PP]		[s-DLCEP]		SD-p	o-pDLSDU
	0101 LFPP	DT		s-DLCEP		SD-p	o-pDLSDU
	1001 0F00	DT [PP]	[d-DLCEP/SAP]	–		SD-p	o- DLSDU
	1001 LFPP	DT	d-DLCEP	–		SD-p	o-pDLSDU
	1001 LFPP	DT	d-DLSAP	–		SD-p	o- DLSDU
	1101 LFPP	DT	d-DLCEP	s-DLCEP/SAP		SD-p	o-pDLSDU
	1101 LFPP	DT	d-DL(SAP)	s-DLSAP		SD-p	o- DLSDU
Establish Connection	1110 LF00	EC		s-DLCEP	s-DLSAP	EC-p	o-DLSDU
	1111 LF00	EC	d-DLCEP/(SAP)	s-DLCEP/SAP	s-DLSAP	EC-p	o-DLSDU
Exchange Data	1000 LFPP	ED	d-DLCEP	–		SD-p	pDLSDU
	1100 LFPP	ED	d-DLCEP	s-DLCEP/SAP		SD-p	pDLSDU
	1100 LFPP	ED	d-DLSAP	s-DLSAP		SD-p	DLSDU
Execute Sequence	1000 LF00	ES	d-DLSEP	–		DD-p	–
Idle	0001 0F10	IDLE	–	–		—	o-DLSDU
Pass Token	0011 0FPP	PT	d-node	–		DD-p	–
Probe Node-address	0010 0110	PN	d-node	–		PN-p	–
Probe Response	0010 0111	PR	–	[s-node]		—	o-SPDU
Request Interval	0010 0000	RI	–	–		DD-p	–
Reset Connection	0110 LFPP	RC		s-DLCEP		RC-p	o-DLSDU
	0111 LFPP	RC	d-DLCEP	s-DLCEP		RC-p	o-DLSDU
Return Token	0011 0100	RT	–	–		—	—
Round-trip-delay Query	1100 0F00	RQ	d-node.0	s-node.0		RQ-p	–

Round-trip-delay Reply	1101 0F00	RR	d-node.0	s-node.0		RR-p	–
Status Response	0001 0F11	SR	–	s-node		o-SR-p	–
Time Distribution	0001 0F01	TD	–	s-node		TD-p	–
Transfer LAS	0000 0110	TL	d-node	–		TL-p	–
Wakeup	0000 0000	WK	d-node	–		–	–

NOTE The same FC values are used for connection-oriented and connectionless DLPDUs. In all cases, the structure of SD-parameters depends on characteristics of the DL-object addressed by the DLPDU's first explicit or implicit DL-address.

C.4 DLPDU short-form summary in alphabetic order of DLPDU acronyms

Compel Acknowledgment	1010 LFPP	CA	d-DLCEP	–		SD-p	o-pDLSDU
	1110 LFPP	CA	d-DLCEP	s-DLCEP/SAP		SD-p	o-pDLSDU
	1110 LFPP	CA	d-DLSAP	s-DLSAP		SD-p	DLSDU
Compel Data	1011 LFPP	CD	d-DLCEP	–		o-SD-p	–
	1111 LFPP	CD	d-DLCEP	s-DLCEP/SAP		o-SD-p	–
	1111 LFPP	CD	d-DLSAP	s-DLSAP		SD-p	–
Claim LAS	0000 0001	CL	–	s-node		–	–
Compel Time	0001 0F00	CT	–	–		–	–
Disconnect Connection	0110 LF00	DC		s-DLCEP		DC-p	o-DLSDU
	0111 LF00	DC	d-DLCEP/(SAP)	s-DLCEP		DC-p	o-DLSDU
Data	0101 0F00	DT [PP]		[s-DLCEP]		SD-p	o-pDLSDU
	0101 LFPP	DT		s-DLCEP		SD-p	o-pDLSDU
	1001 0F00	DT [PP]	[d-DLCEP/SAP]	–		SD-p	o- DLSDU
	1001 LFPP	DT	d-DLCEP	–		SD-p	o-pDLSDU
	1001 LFPP	DT	d-DLSAP	–		SD-p	o- DLSDU
	1101 LFPP	DT	d-DLCEP	s-DLCEP/SAP		SD-p	o-pDLSDU
	1101 LFPP	DT	d-DL(SAP)	s-DLSAP		SD-p	o- DLSDU
Establish Connection	1110 LF00	EC		s-DLCEP	s-DLSAP	EC-p	o-DLSDU
	1111 LF00	EC	d-DLCEP/(SAP)	s-DLCEP/SAP	s-DLSAP	EC-p	o-DLSDU
Exchange Data	1000 LFPP	ED	d-DLCEP	–		SD-p	pDLSDU
	1100 LFPP	ED	d-DLCEP	s-DLCEP/SAP		SD-p	pDLSDU
	1100 LFPP	ED	d-DLSAP	s-DLSAP		SD-p	DLSDU
Execute Sequence	1000 LF00	ES	d-DLSEP	–		DD-p	–
Idle	0001 0F10	IDLE	–	–		—	o-DLSDU
Probe Node-address	0010 0110	PN	d-node	–		PN-p	–
Probe Response	0010 0111	PR	–	[s-node]		—	o-SPDU
Pass Token	0011 0FPP	PT	d-node	–		DD-p	–
Reset Connection	0110 LFPP	RC		s-DLCEP		RC-p	o-DLSDU
	0111 LFPP	RC	d-DLCEP	s-DLCEP		RC-p	o-DLSDU
Request Interval	0010 0000	RI	–	–		DD-p	–
Round-trip-delay Query	1100 0F00	RQ	d-node.0	s-node.0		RQ-p	–
Round-trip-delay Reply	1101 0F00	RR	d-node.0	s-node.0		RR-p	–
Return Token	0011 0100	RT	–	–		—	—
Status Response	0001 0F11	SR	–	s-node		o-SR-p	–
Time Distribution	0001 0F01	TD	–	s-node		TD-p	–
Transfer LAS	0000 0110	TL	d-node	–		TL-p	–
Wakeup	0000 0000	WK	d-node	–		–	–

NOTE The same FC values are used for connection-oriented and connectionless DLPDUs. In all cases, the structure of SD-parameters depends on characteristics of the DL-object addressed by the DLPDU's first explicit or implicit DL-address.

C.5 DLPDU FC code-point assignment matrix – overview and detail

C.5.1 DLPDU FC code-point assignment overview

The first octet of each DLPDU is a frame control (FC) octet which specifies the format of the DLPDU, including the length of all address fields in the DLPDU, and the number and specific roles of the address fields explicitly present in the DLPDU. The FC octet also specifies, where appropriate, the DLPDU's priority.

The encoding of the FC octet is largely orthogonal, as is shown in figure C.1. The exact function of each FC code point is shown in tables C.1 and C.2. In this figure and these tables, the high-order four bits of the FC octet are shown in the row heading on the left side of the figure or table, and the low-order four bits are shown in the column heading at the top of the figure or table.

The blocks shown in figure C.1 indicate the major functions of that section of the FC code point assignment matrix; a small number of unrelated FC code points are also included in each block.

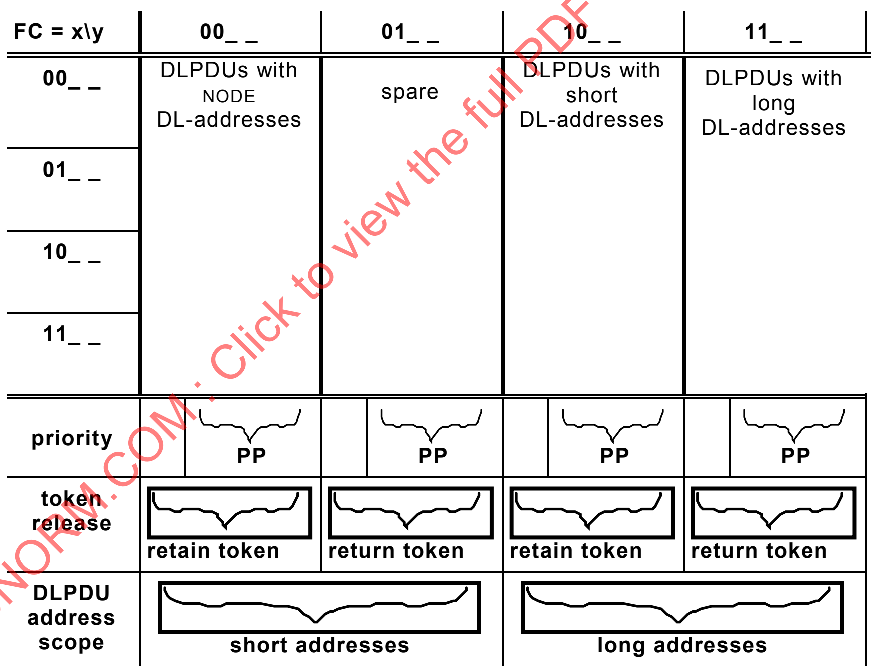


Figure C.1 – Gross structure of FC code points

In Figure C.1, the PP bits take only the values 01, 10 and 11. The PP bits specify the priority of the DLPDU.

Table C.1 shows the generic assignment of the FC code points to the different DLPDUs. It also shows

a) which address fields are explicitly present in the DLPDU (d for destination, s for source, and dss for the three addresses of a connection establishment DLPDU), and which are implicitly present (– for implicit source address);

b) the common length of each of those d and s address fields, either long (four-octet DL-addresses), or short (one-octet NODE DL-addresses in the upper one-quarter of the table; two-octet DL-addresses in the lower three-quarters of the table, all implicitly on the local link).

Table C.1 – Generic assignment of FC code points

FC=x\y	yyyy	00_ _				01_ _				10_ _				11_ _			
xxxx		00	01	10	11	00	01	10	11	00	01	10	11	00	01	10	11
00_ _	00	WK d-	CL -s	don't use		don't use	don't use	TL d-									
	01	CT	TD -s	IDLE	SR [d]s	CT	TD -s	IDLE	SR [d]s								
	10	RI		don't use	don't use	don't use		PN d-	PR -[s]								
	11	don't use	PT d-			RT	PT d-										
01_ _	00																
	01	DT [s]	DT s			DT [s]	DT s			don't use	DT s			don't use	DT s		
	10	DC s	RC s			DC s	RC s			DC s	RC s			DC s	RC s		
	11	DC ds	RC ds			DC ds	RC ds			DC ds	RC ds			DC ds	RC ds		
10_ _	00	ES d-	ED d-			ES d-	ED d-			ES d-	ED d-			ES d-	ED d-		
	01	DT [d]-	DT d-			DT [d]-	DT d-			don't use	DT d-			don't use	DT d-		
	10		CA d-				CA d-				CA d-				CA d-		
	11		CD d-				CD d-				CD d-				CD d-		
11_ _	00	RQ ds	ED ds			RQ ds	ED ds			don't use	ED ds			don't use	ED ds		
	01	RR ds	DT ds			RR ds	DT ds			don't use	DT ds			don't use	DT ds		
	10	EC ss	CA ds			EC ss	CA ds			EC ss	CA ds			EC ss	CA ds		
	11	EC dss	CD ds			EC dss	CD ds			EC dss	CD ds			EC dss	CD ds		

C.5.2 DLPDU code-point assignment rationale

The structure of the frame-control code points is quite regular.

For example

a) FC_3 is the L subfield, specifying the length of all addresses in the DLPDU: when $FC_3 = 0$ all addresses are SHORT (2 octets when $FC_{7654} \geq 0100$; 1 octet otherwise); when $FC_3 = 1$ all addresses are LONG (4 octets when $FC_{7654} \geq 0100$).

b) FC_2 is the F subfield, specifying the location of the token at the end of the current transaction: In general, when $FC_2 = 0$ the token stays with the current token holder; when $FC_2 = 1$ the token returns to the LAS, or transfers to a new LAS, at the end of the current transaction.

c) When $FC_{7654} \geq 0011$ and $FC_{10} \neq 00$, then FC_{10} is the PP subfield.

Table C.2 shows the individual assignment of the FC code points to the different DLPDUs.

Table C.2 – Individual assignment of FC code points

FC=x\y	yyyy	00_ _				01_ _				10_ _				11_ _			
xxxx		00	01	10	11	00	01	10	11	00	01	10	11	00	01	10	11
00_ _	00	WK	CL	don't use	spare	don't use	don't use	TL	spare	spare	spare	spare	spare	spare	spare	spare	spare
	01	CT	TD	IDLE	SR	CT	TD	IDLE	SR	spare	spare	spare	spare	spare	spare	spare	spare
	10	RI	spare	don't use	don't use	don't use	spare	PN	PR	spare	spare	spare	spare	spare	spare	spare	spare
	11	don't use	PT	PT	PT	RT	PT	PT	PT	spare	spare	spare	spare	spare	spare	spare	spare
01_ _	00	spare	spare	spare	spare	spare	spare	spare	spare	spare	spare	spare	spare	spare	spare	spare	spare
	01	DT	DT	DT	DT	DT	DT	DT	DT	don't use	DT	DT	DT	don't use	DT	DT	DT
	10	DC	RC	RC	RC	DC	RC	RC	RC	DC	RC	RC	RC	DC	RC	RC	RC
	11	DC	RC	RC	RC	DC	RC	RC	RC	DC	RC	RC	RC	DC	RC	RC	RC
10_ _	00	ES	ED	ED	ED	ES	ED	ED	ED	ES	ED	ED	ED	ES	ED	ED	ED
	01	DT	DT	DT	DT	DT	DT	DT	DT	don't use	DT	DT	DT	don't use	DT	DT	DT
	10	spare	CA	CA	CA	spare	CA	CA	CA	spare	CA	CA	CA	spare	CA	CA	CA
	11	spare	CD	CD	CD	spare	CD	CD	CD	spare	CD	CD	CD	spare	CD	CD	CD
11_ _	00	RQ	ED	ED	ED	RQ	ED	ED	ED	don't use	ED	ED	ED	don't use	ED	ED	ED
	01	RR	DT	DT	DT	RR	DT	DT	DT	don't use	DT	DT	DT	don't use	DT	DT	DT
	10	EC	CA	CA	CA	EC	CA	CA	CA	EC	CA	CA	CA	EC	CA	CA	CA
	11	EC	CD	CD	CD	EC	CD	CD	CD	EC	CD	CD	CD	EC	CD	CD	CD

C.6 SD-parameters (status and data-description parameters) of CA, CD, ED and DT DLPDUs

C.6.1 SD-p: Basic DLC parameters for CA, CD, ED and DT DLPDUs

Format code	Format summary	Description
A:	(null)	for use when all DLC directions are UNORDERED or not used, or when no pDLSDU or status is being conveyed
B:	(J, K, T, RR, SSS)	sent by a PEER DLC when sequence numbers are required, segmentation is never needed, and all window sizes are ≤3
C:	(RRRR, UUUU), (J, K, T, SSSSS), (TTTT, VVVV)	sent by a PEER DLC when one of the two directions is either DISORDERED or CLASSICAL, sequence numbers are required in the other direction, and segmentation may be needed. may also be used in place of format B, D or E
D:	(RRRR, UUUU), (J, K, T, SSSSS)	sent by a PEER DLC when one of the two directions is either DISORDERED or CLASSICAL, sequence numbers are required in the other direction, segmentation is never needed, and at least one window size is >3. may also be used in place of format E
E:	(RRRR, UUUU)	sent by a SUBSCRIBER, or the receiver of an ORDERED PEER DLC, to request a retransmission when segmentation may be needed and when the reverse direction is either UNORDERED or not used
F:	(J=0, K=0, T, SSSSS), (TTTT, VVVV)	sent by a PUBLISHER or PEER DLC when sequence numbers are required, segmentation may be needed, and the reverse direction is either UNORDERED or not used
G:	(J=0, K=0, T, SSSSS)	sent by a PUBLISHER or PEER DLC when sequence numbers are required, segmentation is never needed, and the reverse direction is either UNORDERED or not used

NOTE 1 Only formats A, C, F and G need be supported. Formats B, D and E are all optimizations of format C, and need not be supported.

NOTE 2 CA, CD, DT and ED DLPDUs which provide connection-mode services use formats A to G.

Subfield Name	Subfield Description
J	Boolean indicating whether retransmission of segment UUUU of the NDR'th DLSDU is requested.
K	Boolean indicating whether all DLSDUs prior to the NDR'th DLSDU are being acknowledged.
T	Boolean indicating whether the associated DLSDU originated in a buffer with associated timeliness criteria, and that those timeliness criteria were met.
RR	2-bit residue (that is, the three low-order bits) of the sequence number NDR of a received DLSDU, or of the DLSDU before a missing DLSDU.
RRRR	4-bit residue (that is, the four low-order bits) of the sequence number NDR of a received DLSDU, or of the DLSDU before a missing DLSDU.
SSS	3-bit residue (that is, the three low-order bits) of the sequence number NDS of this DLSDU, or of the highest-numbered DLSDU sent on the DLC.

SSSSS 5-bit residue (that is, the five low-order bits) of the sequence number NDS of this DLSDU, or of the highest-numbered DLSDU sent on the DLC.

UUUU 4-bit zero-based segment number RSN of the DLSDU segment which needs to be retransmitted.

TTTT 4-bit zero-based total number TNS of DLSDU segments in the DLSDU, one segment of which is conveyed in the associated DLPDU.

VVVV 4-bit zero-based segment number ASN of the DLSDU segment conveyed in the associated DLPDU.

Z…Z one or more bits of zero, as indicated by the number of “Z”s, reserved for future standard use.

C.6.2 SD-p: DLSDU source time stamp for CA, CD, ED and DT DLPDUs

<table>
<tr><th></th><th colspan="6">Low-order octets of DL-time, with a unit weight of 2^{-5} ms</th></tr>
<tr><td>J</td><td colspan="6">– (null) –</td></tr>
<tr><td>K</td><td colspan="4">– (null) –</td><td colspan="2">two low-order octets of DL-time</td></tr>
<tr><td>L</td><td colspan="3">– (null) –</td><td colspan="3">three low-order octets of DL-time</td></tr>
<tr><td>M</td><td colspan="6">DL-time</td></tr>
<tr><td></td><td>one octet</td><td>one octet</td><td>one octet</td><td>one octet</td><td>one octet</td><td>one octet</td></tr>
</table>

As in all OSI protocols, all multi-octet numeric fields are transmitted most-significant-octet first.

C.6.3 SD-p: Unitdata parameters and delayed-reply parameters

<table>
<tr><th>Format code</th><th>Format summary</th><th>Description</th></tr>
<tr><td>P:</td><td>(null)</td><td>for non-acknowledged unitdata DLPDUs with DL(SAP)s</td></tr>
<tr><td>R:</td><td>(ZZ,PP,YYYY)</td><td>for acknowledged-unitdata and exchange-unitdata request CA, CD and ED DLPDUs, all with DLSAPs</td></tr>
<tr><td>U:</td><td>(XXXX,YYYY)</td><td>for unitdata-acknowledgment and exchange-unitdata-reply DT and SR DLPDUs, all with DLSAPs</td></tr>
<tr><td colspan="3">CA, CD and ED DLPDUs which provide connectionless services use only format R;
DT DLPDUs with two DL-addresses which provide connectionless services use only format P;
DT DLPDUs with one explicit or implicit destination DL-address and an implicit source DL-address which provide connectionless services use only format U;
SR DLPDUs use format U.</td></tr>
</table>

Subfield Name Subfield Description

PP 2-bit DLL priority of the DLSDU conveyed in the associated CA, CD or ED DLPDU, as previously specified (01=URGENT, 10=NORMAL, 11=TIME-AVAILABLE).

XXXX 4-bit reply status as specified in Table C.3 for SR and Table C.4 for appropriate DT DLPDUs.

YYYY 4-bit not-yet-confirmed.

ZZ two bits of zero, reserved for future standard use.

C.6.4 SR-p: Status response parameters

Format code	Format summary	Description
X:	(null)	
Y:	(XXXX, ZZZZ)	for unitdata-acknowledgment and exchange-unitdata-reply error SR DLPDUs, all with DLSAPs

Subfield Name	Subfield Description
XXXX	4-bit reply status as specified in Table C.3 for SR DLPDUs.
ZZZZ	four bits of zero, reserved for future standard use.

C.6.5 Reply status for SR DLPDUs, and for unitdata-acknowledgment and exchange-unitdata-reply DT DLPDUs

Table C.3 – Reply status for SR DLPDUs

Short name	Definition	Hexadecimal coding
RR	failure – resource limitation in responder	4
RI	failure – responder DLE-class incompatible with this request	9
RTR	failure – LAS transfer rejected	B
BF	failure – fault in intermediary bridge	C
BR	temporary failure – resource limitation in intermediary bridge	D
BOK	interim success – intermediary bridge is forwarding transaction	E

Table C.4 – Reply status for unitdata-acknowledgment and exchange-unitdata-reply DT DLPDUs

Short name	Definition	Hexadecimal coding
OK	success – no reply data available at responder or service does not provide this information	0
OK_U	success – URGENT priority reply data available at responder	1
OK_N	success – NORMAL priority reply data available at responder	2
OK_TA	success – TIME-AVAILABLE priority reply data available at responder	3
RR	failure – resource limitation in responder – no reply data available at responder or service does not provide this information	4
RR_U	failure – resource limitation in responder – URGENT priority reply data available at responder	5
RR_N	failure – resource limitation in responder – NORMAL priority reply data available at responder	6
RR_TA	failure – resource limitation in responder – TIME-AVAILABLE priority reply data available at responder	7
RF	failure – fault in responder	8
RI	failure – responder DL(SAP)-role incompatible with this DLPDU	9
RA	failure – response restricted to a different peer DLSAP-address	A
—	reserved for compatibility with the SR DLPDU's reply status – not available for other use	B
BF	failure —fault in intermediary bridge	C
BR	failure —resource limitation in intermediary bridge	D
BOK	reserved for interim success – intermediary bridge is forwarding transaction	E
DR	interim success – delayed reply; end station needs more time to prepare response	F

C.7 EC parameters of EC DLPDUs

EC-p:	(sender's R, NNN, Q, VVV),	-- 1 octet
	(sender's CC, PP, SS, XX),	-- 1 octet
	(sender's MCD_{CR}),	-- 2 octets
	(sender's MCD_D),	-- 2 octets
	(sender's TT_S, A_S, B_S, $WWWW_S$),	-- 1 octet
	(sender's FFF_S, E_S, reserved, G_S, HH_S),	-- 1 octet
	(sender's $M...M_S$),	-- 2 octets
	(sender's TT_R,A_R, B_R, $WWWW_R$),	-- 1 octet
	(sender's FFF_R, E_R, reserved, G_R, HH_R),	-- 1 octet
	(sender's $M...M_R$)	-- 2 octets

As in all OSI protocols, all multi-octet numeric fields are transmitted most-significant-octet first.

R Boolean requiring return of a corresponding DLPDU (1=reply DLPDU requested)

NNN publisher-DLCEP-address reuse-discriminator, or zero for non-publisher DL-addresses

Q DLL path diversity (0=any-path, 1=this-path)

VVV DLL protocol version number (always = 1 for this version)

CC DLCEP class (01=PEER, 10=PUBLISHER, 11=SUBSCRIBER)

PP DLL (and transaction) priority, as previously specified (01=urgent, 10=normal, 11=time-available)

SS minimum DLL address size (01=LONG, 10=SHORT, 11=VERY-SHORT)

XX DLPDU authentication (00=ordinary, 10=source, 11=maximal)

MCD_{CR} negotiated maximum confirm delay for DL-Connect and DL-Reset requests $1 \leq D...D \leq 60\,000$, or 0xFFFF, where 0xFFFF represents the choice unlimited.

MCD_{D} negotiated maximum confirm delay for DL-DATA requests $1 \leq D...D \leq 60\,000$, or 0xFFFF, where 0xFFFF represents the choice unlimited.

For the following parameters, an S subscript indicates the sender's desired attributes as a source of DLS-user data on the DLC, and an R subscript indicates the sender's desired attributes as a receiver of DLS-user data on the DLC.

TT DLCEP features (00=unordered, 01=ordered, 10=disordered, 11=classical); =00 when M...M=00

A Boolean requiring residual activity in the specified (sender-to-receiver or receiver-to-sender) direction (1=residual activity required)

B Boolean local buffer/queue binding in the specified (sender-to-receiver or receiver-to-sender) direction is to a queue (=0) or to a buffer (=1)

WWWW window size (number of DLPDUs retained for possible retransmission; = 1 for BUFFER or UNORDERED; = min(*K*, 15) for QUEUE-*K*; default = 4 for implicit queue (unless overridden by local DL-management; =0 when M...M=0)

FFF SD-p format (see C.6) in the specified direction (0=format A, 1=format B, 2=format C, 3=reserved, 4=format D, 5=format E, 6=format F, 7=format G)

NOTE Formats R and U (see C.6) are not used with the connection-oriented DLS.

E Boolean indicating desire/ability to have sent/received ED DLPDUs contain a pDLSDU (1=pDLSDU desired/permitted in ED DLPDUs on this DLC)

reserved one or more bits of zero, reserved for future standard use

G Boolean indicating the presence (=1) or absence (=0) of a timeliness parameter in transmitted DLPDUs in the specified direction

HH time-stamp-format in the specified direction (0=format J, 1=format K, 2=format L, 3=format M)

M...M maximum DLSDU size $0 \leq M...M \leq 16 \times (M...M_{PP})$, based on the DLL priority PP, where $M...M_{Urgent}$=64, $M...M_{Normal}$=128 and $M...M_{Time\text{-}Available}$=256.

C.8 Parameters of DC and RC DLPDUs

C.8.1 DC parameters of DC DLPDUs

DC-p: (R, reserved, VVV), -- 1 octet
(RRRRRRRR) -- 1 octet

R Boolean requiring return of a corresponding DLPDU (1=return DLPDU requested)

reserved one or more bits of zero, reserved for future standard use

VVV DLL protocol version number (always = 1 for this version)

RRRRRRRR reason for disconnect (see C.8.3)

C.8.2 RC parameters of RC DLPDUs

RC-p: (R, reserved, VVV), -- 1 octet
(RRRRRRRR), -- 1 octet
(XXXSSSSS), -- 1 octet

R Boolean requiring return of a corresponding DLPDU (1=return DLPDU requested)

reserved one or more bits of zero, reserved for future standard use

VVV DLL protocol version number (always = 1 for this version)

RRRRRRRR reason for reset (see C.8.3)

XXXSSSSS 5-bit residue of the sequence number N_S preceding the next sequence number to be sent on the DLC, if any, with an arbitrary value for the three high-order bits of the octet.

C.8.3 DC and RC reason coding

Code	Reason for disconnect or reset	Reason class
00	user-originated disconnection – normal condition	user-originated disconnection (00..1F)
02	user-originated disconnection – abnormal condition	
1E	user-originated disconnection or connection rejection – reason unspecified	
20	user-originated connection rejection – connection not authorized, permanent condition	user-originated connection rejection (20..3F)
21	user-originated connection rejection – unacceptable QoS, permanent condition	
22	user-originated connection rejection – non-QoS reason, permanent condition	
24	user-originated connection rejection – transient condition	
40	provider-originated disconnection – incorrect DLCEP pairing, permanent condition	provider-originated disconnection (40..5F)
41	provider-originated disconnection – wrong publisher-DLCEP-address reuse-discriminator, permanent condition	
42	provider-originated disconnection – other permanent condition	
43	provider-originated disconnection – wrong DLPDU format or parameters, permanent condition	
44	provider-originated disconnection – wrong DLSDU size, permanent condition	
45	provider-originated disconnection – transient condition	
46	provider-originated disconnection – timeout	
5E	provider-originated disconnection or connection rejection – reason unspecified	

Code	Reason for disconnect or reset	Reason class
60	provider-originated connection rejection – DL(SAP) address unknown	provider-originated connection rejection (60..7D)
62	provider-originated connection rejection – DLSAP unreachable, permanent condition	
64	provider-originated connection rejection – DLSAP unreachable, transient condition	
65	provider-originated connection rejection – inconsistent DLCEP state, permanent condition	
66	provider-originated connection rejection – QoS unavailable, permanent condition	
68	provider-originated connection rejection – QoS unavailable, transient condition	
7E	disconnection or connection rejection, unknown origin – reason unspecified	unknown origin disconnect (7E..7F)
80	user-originated reset – resynchronization after user timeout	user-originated reset (80..BF)
82	user-originated reset – resynchronization after user-detected user-state inconsistencies	
9E	user-originated reset – reason unspecified	
C0	provider-originated reset – resynchronization after activation of a DL-management-established DLCEP	provider-originated reset (C0..FD)
C2	provider-originated reset – resynchronization after timeout	
C4	provider-originated reset – resynchronization after maximum number of retransmission requests or attempts	
C6	provider-originated reset – resynchronization after detected sequence number error	
C8	provider-originated reset – resynchronization after other detected DLCEP state inconsistencies	
FC	provider-originated reset – reason unspecified	
FE	reset, unknown origin – reason unspecified	unknown origin reset (FE..FF)

C.9 Parameters of TD, RQ and RR DLPDUs

C.9.1 General

C(NT) is the current value of the nodetimer, local to each DLE, zeroed at DLE startup.

NOTE The three low-order octets of C(NT), or of a value derived from C(NT),

a) are appended to the TD, RQ and RR DLPDUs at the moment of transmission, with an approximately fixed delay relative to transmission of the last octet of the DLPDU;

b) are appended to the DLPDU at the moment of reception, with an approximately fixed delay relative to receipt of the last octet of the DLPDU .

Thus the TD, RQ and RR DLPDUs can be considered to have three distinct stages in their formation:

i) the DLPDU header as it exists in a transmit queue awaiting the opportunity to transmit, without either of its last two C(NT)-based fields appended;

ii) the DLPDU as it is presented to and reported by the PhL, without its last C(NT)-based field appended;

iii) the DLPDU as it exists in a receive queue awaiting post-reception processing by upper-DLL software, with all C(NT)-based fields appended.

N(NT) is a previously recorded value of some DLE's C(NT).

V(DLTO) (see 4.7.1.21) is the DL-time-offset, which equals the current value of DL-time minus (V(LSTO) + C(NT).

As in all OSI protocols, all multi-octet numeric fields are transmitted most-significant-octet first.

C.9.2 TD-parameters of TD DLPDUs

TD-p:	(L…L),	-- 2 octets
	(TTT, LLL, SS),	-- 1 octet
	VS(DLTO),	-- 7 octets
	(VS(DLTO) + VS(LSTO)+CS(NT)),	-- 7 octets
	(A…A),	-- 3 octets, appended at moment of transmission
	(CR(NT)23..0)	-- 3 octets, appended at moment of reception

V_S(TSL) 16-bit link-id of the root link of the spanning (sub)tree from which the DL-time sense originated, that is, the link-id of the time-source, V_S(TSL)

L…L 16-bit link-id of the root link of the spanning (sub)tree from which the DL-time sense originated, that is, the link-id of the time-source, V_S(TSL)

TTT limiting time-synchronism class between time-source and sender, $N_S(TQ)_{7-5}$ (000=NONE, 001=1 s, 010=100 ms, 011=10 ms, 100=1 ms, 101=100 µs, 110=10 µs, 111=1 µs)

LLL number of intervening links on the DL-time propagation path from the DL-time source DLE to the sending DLE, $N_S(TQ)_{4-2}$,

0 indicates that the DL-time originates with the DLE itself,

1 indicates that the DL-time originates with another DLE on the local link,

2 indicates that the DL-time originates with a DLE on a link one bridge removed,

and so forth;

SS time-source-type, expressing the method of synchronization of the DL-time source DLE with Universal Coordinated Time (UTC), the worldwide time standard, $N_S(TQ)_{1-0}$

00 some DLE's node-time, locally generated and not received from an extra-DLS-provider source,

01 local time (not UTC) received (directly or indirectly) from a human source,

10 UTC received (directly or indirectly) from a human source or an unreliable electronic source, or previously (but no longer) received (directly or indirectly) from a reliable electronic source,

11 UTC continuously received (directly or indirectly) from a reliable electronic source.

V_S(DLTO) the DL-time-offset of the sending DLE during DLPDU formation shall be expressed as a signed seven-octet integer encoded most-significant-octet first, where the least significant bit represents a time granularity of approximately 2^{-13} ms (see Table C.5)

V_S(DLTO) + V_S(LSTO)+$C_S(NT)_1$ the DL-time of the sending DLE at a moment during DLPDU formation, less than 1,1 s prior to transmission, where the least significant bit represents a time granularity of approximately 2^{-13} ms

A…A a small adjustment to account for systemic offsets in the previous subfield, where the least significant bit represents a time granularity of approximately 2^{-13} ms, computed as $C_S(NT)_2 - C_S(NT)_1$, where $C_S(NT)_2$ represents the moment of end-of-transmission

$C_R(NT)_{23..0}$ low-order three octets of TD receiver's nodetimer at the moment of reception. This subfield does not appear on the medium (see Table C.6).

Table C.5 – Approximate numeric significance of the bits of seven-octet DL-time

Octet of DL-time in transmission order	Symbolic contents	Approximate weight of the low-order bit of the octet
1	0YYYYYYY	1,09 year
2	DDDDDDDD	1,55 day
3	HHHHHMMM	8,74 min
4	MMMSSSSS	2,05 s
5	Smmmmmmm	8,00 ms
6	mmm•µµµµµ	31,25 µs
7	µµµµµnnn	122 ns

Table C.6 – Approximate numeric significance of the bits of N(NT), A...A, and three-octet C(NT)

Octet of short time in transmission order	Symbolic contents	Approximate weight of the low-order bit of the octet
1	Smmmmmmm	8,00 ms
2	mmm•µµµµµ	31,25 µs
3	µµµµµnnn	122 ns

C.9.3 RQ-parameters of RQ DLPDUs

RQ-p: (CS(NT)23..0) -- 3 octets, appended at the moment of transmission
(CR(NT)23..0) -- 3 octets, appended at the moment of reception

CS(NT)23..0 low-order three octets of RQ sender's nodetimer at the moment of RQ DLPDU formation just prior to (or during) transmission, representing the low-order three octets of the DLS-provider's current node-time in units of 2-13 ms

CR(NT)23..0 low-order three octets of RQ receiver's nodetimer at moment of reception. This subfield does not appear on the medium

C.9.4 RR-parameters of RR DLPDUs

RR-p: (TTT, E, reserved), -- 1 octet
(NR(NT)23..0), -- 3 octets
(NS(NT)23..0) -- 3 octets
(CS(NT)23..0), -- 3 octets, appended at moment of transmission
(CR(NT)23..0) -- 3 octets, appended at moment of reception

TTT time-synchronism class of the replying DLE, $N_S(TQ)_{7\text{-}5}$, encoded as in C.9.2

E Boolean indicating whether a DLE maintains a real or estimated C(NT); true unless $N_S(TQ)_{7\text{-}5}$ specifies NONE

reserved one or more bits of zero, reserved for future standard use

$N_R(NT_{23..0})$ low-order three octets of RR receiver's (RQ sender's) nodetimer at the moment of transmission of the interrogating RQ DLPDU

$N_S(NT_{23..0})$ low-order three octets of RR sender's (RQ receiver's) nodetimer at the moment of reception of the interrogating RQ DLPDU, adjusted for any systemic offset in the value of the this and the next subfield

$C_S(NT)_{23..0}$ low-order three octets of RR sender's nodetimer at the moment of RR DLPDU formation just prior to (or during) transmission, representing the low-order three octets of the DLS-provider's current node-time in units of 2^{-13} ms

$C_R(NT)_{23..0}$ low-order three octets of RR receiver's nodetimer at the moment of reception. This subfield does not appear on the medium.

C.10 Parameters of PN, PT, ES and RI DLPDUs

C.10.1 General

As in all OSI protocols, all multi-octet numeric fields are transmitted most-significant-octet first.

C.10.2 PN-parameters of PN DLPDUs

PN-p:	(SSSS, reserved, VVV),	-- 1 octet
	(GGGG, PPPP),	-- 1 octet
	(TTTTTTTT TTTTTTTT),	-- 2 octets
	(DDDDDDDD),	-- 1 octet
	(MMMMMMMM)	-- 1 octet

SSSS PhL maximum inter-channel signal skew

reserved one or more bits of zero, reserved for future standard use

VVV DLL protocol version number (always = 001 for this version)

GGGG PhL post-transmission-gap extension units

PPPP PhL preamble extension units

T…T slot-time, V(ST), 1 ≤ T…T ≤ 4 095 octet-durations

D…D maximum-response-delay, V(MRD), 0 ≤ D…D ≤ 11 slot-times

M…M minimum-inter-PDU-delay, V(MID), 0 ≤ M…M ≤ min(120, (D…D –1) × T…T)

C.10.3 DD-parameters (delegation-duration parameters) of PT, ES and RI DLPDUs

DD-p: (DDDDDDDD DDDDDDDD) -- 2 octets

D…D requested or delegated duration of token usage, V(RD), measured in octet-durations

C.11 Addressing summary extracted from figures and tables of 4.3

The figures and tables in this annex are reproduced from 4.3. They provide a convenient summary of the structure of DL-addresses and the preassigned ranges and uses of those DL-addresses.

Link designator	Sublink selector
2 octets	2 octets

**Figure C.2 [Figure 2] –
Basic structure of a DL-address**

Implied link designator (0000)	Node designator	Sub-node selector
	1 octet	1 octet

**Figure C.3 [Figure 3] –
Basic structure of a sublink selector**

Link	Node	Selector
2 octets	1 octet	1 octet

or

Link	Node || selector
2 octets	2 octets

or

Link || Node || Selector
4 octets

**Figure C.4 [Figure 4] –
DL-address alternative structures**

Universally-administered MAC-addresses			
I/G	0	Globally-administered	Vendor-administered
1	1	22 bits	24 bits

Locally-administered MAC-addresses		
I/G	1	User-administered
1	1	46 bits

**Figure C.5 [Figure 5] –
Basic structure of MAC-addresses**

A DL-address as a locally-administered MAC-address			
I/G	1	Set by DL-management	Extended DL-address
1	1	14 bits	32 bits

Figure C.6 [Figure 6] – Representation of a DL-address as a MAC-address

Table C.7 [Table 1] – Link || node || selector addressing

Link	N | S	Assigned use for specified DL-address range
0000	0000 … FFFF	local link DL-addresses as specified in Table 2
0001	0000 … 0007	flat non-local group DL-addresses reserved for DLL use
0001	0008 … 003F	flat non-local group DL-addresses reserved for standard AEs
0001 … 003F	0040 … FFFF	flat non-local group DL-addresses available for vendor, user and dynamic assignment
0040 … 007F	0000 … FFFF	flat non-local globally-administered (see Note) group DL-addresses – potentially one per vendor
0080	0000 … 0007	flat non-local individual “DLSAP”- addresses reserved for DLL use
0080	0008 … 003F	flat non-local individual DLSAP- addresses reserved for standard AEs
0080 … 00FF	0040 … FFFF	flat non-local individual DLSAP- addresses available for vendor, user and dynamic assignment
0100 0FFF	0000 FFBF	flat non-local individual DL-addresses available for vendor, user and dynamic assignment as DLCEP-addresses and DLSEP-addresses
0FFF	FFC0 … FFF7	flat non-local individual DL-addresses reserved for standard AE DLCEP-addresses and DLSEP-addresses
0FFF	FFF8 … FFFF	flat non-local individual DL-addresses reserved for DLL use as DLCEP-addresses and DLSEP-addresses
1000 … FEFF	0000 … FFFF	Individual links of DL-addresses as specified in Table 2
NOTE These group DL-addresses are based on the Organizationally Unique Identifiers specified in the ISO/IEC 8802 family of Local Area Network standards.		

Table C.8 [Table 2] – Link-local node || selector addressing

N	S	Assigned use for specified DL-address range
00	00 … FF	local node DL-addresses as specified in Table 4
01	00 … 07	flat link-local group DL-addresses reserved for DLL use
01	08 … 3F	flat link-local group DL-addresses reserved for standard AEs
01 … 03	40 … FF	flat link-local group DL-addresses available for vendor, user and dynamic assignment
04	00 … 07	flat link-local individual “DLSAP”- addresses reserved for DLL use
04	08 … 3F	flat link-local individual DLSAP- addresses reserved for standard AEs
04	40 … FF	flat link-local individual DLSAP- addresses available for vendor, user and dynamic assignment
05 0F	00 BF	flat link-local individual DL-addresses available for vendor, user and dynamic assignment as DLCEP-addresses and DLSEP-addresses
0F	C0 … F7	flat link-local individual DL-addresses reserved for standard AE DLCEP-addresses and DLSEP-addresses
0F	F8 … FF	flat link-local individual DL-addresses reserved for DLL use as DLCEP-addresses and DLSEP-addresses
10 … FF	00 … FF	Individual nodes of DL-addresses as specified in Table 4

Table C.9 [Table 5] – Predefined flat non-local DL-addresses

link || N || S		Assigned use for specified DL-address
0001	0000	the DL-support functions of "all" (see Note) DLEs on the extended link
0001	0001	the DL-support functions of "all" (see Note) LM DLEs on the extended link
0001	0002	the DL-support functions of "all" (see Note) bridge DLEs on the extended link
0001	0003	the DL-bridge functions of "all" (see Note) bridge DLEs on the extended link
0001	0008	the SMAEs of "all" (see Note) unconfigured DLEs on the extended link
0001	0009	the SMAEs of "all" (see Note) DLEs on the extended link
0001	000A	the SMAEs of "all" (see Note) LM DLEs on the extended link
0001	000B	the SMAEs of "all" (see Note) bridge DLEs on the extended link
0001	000C	the SLAEs of "all" (see Note) LoadServers on the extended link
0001	000D	the SLAEs of "all" (see Note) Loadable Devices on the extended link
0080	0000	reserved for DLL use for a DL-support "DLSAP"-address (see Note)
0080	0004	the "DLSAP"-address for the DL-bridge functions of the bridge DLE on the extended link which is the root bridge of the spanning tree
0FFF	FFFE	the DLCEP-address for the DLC from the bridge functions of the bridge DLE on the extended link which is the root bridge of the spanning tree to the bridge functions in all other bridge DLEs on the extended link
0FFF	FFFF	reserved for DLL use for a DL-support DLCEP-address or DLSEP-address (see Note)
NOTE DLEs which do not recognize LONG DL- addresses are necessarily excluded from these sets.		

Table C.10 [Table 6] – Predefined flat link-local DL-addresses

node || selector		assigned use for specified DL-address
01	00	the DL-support functions of all DLEs on the link
01	01	the DL-support functions of all LM DLEs on the link
01	02	the DL-support functions of all bridge DLEs on the link
01	03	the DL-bridge functions of all bridge DLEs on the link
01	08	the SMAEs of all unconfigured DLEs on the link
01	09	the SMAEs of all DLEs on the link
01	0A	the SMAEs of all LM DLEs on the link
01	0B	the SMAEs of all bridge DLEs on the link
01	0C	the SLAEs of all LoadServers on the link
01	0D	the SLAEs of all LoadableDevices on the link
04	00	the "DLSAP"-address for the DL-support functions of the DLE on the link which is serving as LAS
04	04	the "DLSAP"-address for the DL-bridge functions of the bridge DLE on the link which is dominant (closest to the root) in the bridge spanning tree
0F	FE	the DLCEP-address for the DLC from the bridge functions of the bridge DLE on the link which is dominant (closest to the root) in the bridge spanning tree to the bridge functions in all other bridge DLEs on the link
0F	FF	the DLCEP-address for the DLC from the DL-support functions of the DLE on the link which is serving as LAS to the DL-support functions of all of the other LM DLEs on the link

Table C.11 [Table 3] – Link-local node designators

Node	Assigned use for specified DL-address range
00	local DLE
01 … 0F	unusable
10 … 13	bridge-class DLEs
14	link-master-class DLEs
15 … F6	unused
… F7	basic-class DLEs
F8 … FB	non-"visitor" DLEs awaiting proper node designator assignment
FC … FF	"visitor" DLEs

Table C.12 [Table 4] – Node-local selector addressing

Selector	Assigned use for specified DL-address range
00 … 01	node-local individual "DLSAP"-addresses reserved for DLL use
02 … 07	node-local individual DLSAP-addresses reserved for standard AEs
08 … 1F	node-local individual DLSAP-addresses available for vendor, user and dynamic assignment
20 … F7	node-local individual DL-addresses available for vendor, user and dynamic assignment as DLCEP-addresses and DLSEP-addresses
F8 … FF	node-local individual DL-addresses reserved for use as standard AE DLCEP-addresses and DLSEP-addresses

Table C.13 [Table 7] – Predefined node-local DL-addresses

Selector	Assigned use for specified DL-address
00	the "DLSAP"-address for the DL-support functions of the node's DLE
01	the "DLSAP"-address for the DL-bridge functions of the node's DLE
02	the DLSAP-address for the node's SMAE
03	the DLSAP-address for the node's SLAE

Bibliography

IEC 60847:1988[3], *Characteristics of LANs*

IEC 61158-5-5:2014, *Industrial communication networks – Fieldbus specifications – Part 5-9: Application layer service definition – Type 9 elements*

IEC 61158-6-5:2014, *Industrial communication networks – Fieldbus specifications – Part 6-9: Application layer protocol specification – Type 9 elements*

IEC 61784-1, *Industrial communication networks – Profiles – Part 1: Fieldbus profiles*

IEC 61784-2, *Industrial communication networks – Profiles – Part 2: Additional fieldbus profiles for real-time networks based on ISO/IEC 8802-3*

ISO/IEC 8802 (all parts), *Information technology – Telecommunications and information exchange between systems – Local and metropolitan area networks – Specific requirements*

ISO/IEC TR 8802-1, *Information technology – Telecommunications and information exchange between systems – Local and metropolitan area networks – Specific requirements – Part 1: Overview of Local Area Network Standards*

ISO/IEC 8802-2, *Information technology – Telecommunications and information exchange between systems – Local and metropolitan area networks – Specific requirements – Part 2: Logical link control*

ISO/IEC 8802-3, *Information technology – Telecommunications and information exchange between systems – Local and metropolitan area networks – Specific requirements – Part 3: Carrier sense multiple access with collision detection (CSMA/CD) access method and physical layer specifications*

ISO/IEC 8802-4:1990, *Information processing systems – Local area networks – Part 4: Token-passing bus access method and physical layer specifications*[4]

ISO/IEC 8802-5:1998, *Information technology – Telecommunications and information exchange between systems – Local and metropolitan area networks – Specific requirements – Part 5: Token ring access method and physical layer specifications*

ISO/IEC 9314-2, *Information processing systems – Fibre Distributed Data Interface (FDDI) – Part 2: Token Ring Media Access Control (MAC)*

ISO/IEC 9646-1, *Information technology – Open Systems Interconnection – Conformance testing methodology and framework – Part 1: General concepts*

ISO/IEC 9646-2, *Information technology – Open Systems Interconnection – Conformance testing methodology and framework – Part 2: Abstract test suite specification*

ISO/IEC 15802-1, *Information technology – Telecommunications and information exchange between systems – Local and metropolitan area networks – Common specifications – Part 1: Medium Access Control (MAC) service definition*

[3] Withdrawn

[4] Withdrawn

ISO/IEC 15802-3, *Information technology – Telecommunications and information exchange between systems – Local and metropolitan area networks – Common specifications – Part 3: Media Access Control (MAC) Bridges*

ISO 8509:1987, *Information processing systems – Open System Interconnection – Service conventions*[5]

[5] Withdrawn

SOMMAIRE

AVANT-PROPOS ... 419
0 INTRODUCTION ... 421
0.1 Généralités ... 421
0.2 Nomenclature pour les références dans la présente norme ... 421
1 Domaine d'application ... 422
1.1 Généralités ... 422
1.2 Spécifications ... 422
1.3 Procédures ... 422
1.4 Applicabilité ... 423
1.5 Conformité ... 423
2 Références normatives ... 423
3 Termes, définitions, symboles et abréviations ... 424
3.1 Termes et définitions du modèle de référence ... 424
3.2 Termes et définitions de convention pour les services ... 426
3.3 Termes et définitions ... 427
3.4 Symboles et abréviations ... 436
4 Vue d'ensemble du protocole de DL ... 441
4.1 Modèle à trois niveaux de la DLL ... 441
4.2 Service fourni par la DLL ... 444
4.3 Structure et définition des adresses de DL ... 452
4.4 Service assumé à partir de la PhL ... 465
4.5 Fonctions de la DLL ... 468
4.6 Classes fonctionnelles ... 470
4.7 Paramètres, variables, compteurs, temporisateurs et files d'attente locaux ... 472
5 Structure générale et codage des PhIDU et des DLPDU, et éléments de procédure connexes ... 487
5.1 Structure et codage des PhIDU ... 487
5.2 Structure, codage et éléments de procédure communs de DLPDU ... 488
6 Structure, codage et éléments de procédure spécifiques à une DLPDU ... 498
6.1 DLPDU "Establish connection" ("Établir une connexion" (EC)) ... 501
6.2 DLPDU "Disconnect connection" (déconnecter la connexion (DC)) ... 504
6.3 DLPDU "Reset connection" (Réinitialiser une connexion (RC)) ... 506
6.4 DLPDU "Compel acknowledgement" (Forcer un acquittement (CA)) ... 508
6.5 DLPDU "Compel data" (Forcer les données (CD)) ... 515
6.6 DLPDU "Exchange data" (Échanger des données (ED)) ... 522
6.7 DLPDU Data (Données(DT)) ... 531
6.8 DLPDU "Status response" (réponse de statut (SR)) ... 539
6.9 DLPDU "Compel time" (Forcer le temps (CT)) ... 542
6.10 DLPDU Time distribution (Distribution du temps (TD)) ... 544
6.11 DLPDU Round-trip-delay query (interrogation de temps de propagation aller-retour (RQ)) ... 546
6.12 DLPDU Round-trip-delay reply (réponse de temps de propagation aller-retour (RR)) ... 548
6.13 DLPDU Probe node DL-address (Sonder l'adresse de DL de nœud (PN)) ... 550
6.14 DLPDU Probe response (Sonder une réponse (PR)) ... 553
6.15 DLPDU Pass token (Passer un jeton (PT)) ... 555
6.16 DLPDU Execute sequence (Exécuter une séquence (ES)) ... 564

6.17 DLPDU Return token (Retourner un jeton (RT)) 571
6.18 DLPDU Request interval (Demander un intervalle (RI)) 572
6.19 DLPDU Claim LAS (Réclamer le LAS (CL)) 573
6.20 DLPDU Transfer LAS (Transférer le LAS (TL)) 575
6.21 DLPDU Wakeup (Réveil (WK)) 579
6.22 DLPDU Idle (Repos (IDLE)) 580
6.23 DLPDU Spare (De rechange) 582
6.24 DLPDU Reserved (not to be used) (Réservées, non utilisées) 583
7 Structure et codage de paramètres de DLPDU 584
7.1 Structure et codage des EC-PARAMETERS (paramètres EC) 584
7.2 Structure et codage des DC-PARAMETERS (paramètres DC) 589
7.3 Structure et codage des RC-PARAMETERS (paramètres RC) 590
7.4 Structure et codage des SD-Parameters (paramètres SD) 593
7.5 Structure et codage des SR-Parameters (paramètres SR) 600
7.6 Structure et codage des TD-Parameters (paramètres TD) 601
7.7 Structure et codage des RQ-Parameters (paramètres RQ) 603
7.8 Structure et codage des RR-Parameters (paramètres RR) 604
7.9 Structure et codage des PN-Parameters (paramètres PN) 605
7.10 Structure et codage des DD-Parameters (paramètres DD) 607
8 Éléments de procédure du service de DL 607
8.1 Fonctionnement de services de gestion d'adresses de DL(SAP), de tampons et de files d'attente 607
8.2 Fonctionnement des services en mode connexion 611
8.3 Fonctionnement des services en mode sans connexion 655
8.4 Fonctionnement des services de conseils de programmation 667
9 Sous-protocole de prise en charge de DL 677
9.1 Généralités 677
9.2 Vue d'ensemble du fonctionnement de LAS 677
9.3 Définition de sous-protocole de prise en charge de DL 678
9.4 Éléments de procédures pour les SPDU de réception 714
10 Autres éléments de procédure de DLE 715
10.1 Initialisation de DLE 715
10.2 Comportement et fonctionnement de LAS 720
10.3 Fonctionnement de la prise en charge de DL 727
10.4 Éléments de procédure ponts de DL et sous-protocole de pont 732
10.5 Informations de gestion de DL 768
10.6 Profils des mises en œuvre 773
11 Formulaire PICS 779
11.1 Introduction 779
11.2 Généralités 779
11.3 Références normatives 779
11.4 Définitions 779
11.5 Abréviations 779
11.6 Conformité 780
11.7 Instructions 780
11.8 Identification 781
11.9 Profil de mise en œuvre 782
11.10 Capacités majeures de bas niveau 786

11.11 Capacités majeures de haut niveau ... 800
Annexe A (informative) Mise en œuvre exemplaire de FCS ... 811
Annexe B (informative) Type 1: Diagrammes d'états finis formels de protocole ... 813
B.1 FSM de réception et de transmission de base ... 813
B.2 FSM pour DLC ... 825
B.3 FSM pour programmation ... 831
B.4 FSM pour les ponts ... 831
Annexe C (informative) Type 1: Résumés succincts des DLPDU et de l'adressage de DL ... 833
C.1 Champs utilisés dans les résumés succincts ... 833
C.2 Résumé succinct de DLPDU groupées par fonction ... 835
C.3 Résumé succinct des DLPDU dans l'ordre alphabétique des noms de DLPDU ... 836
C.4 Résumé succinct des DLPDU dans l'ordre alphabétique des acronymes de DLPDU ... 837
C.5 Matrice d'affectation des points de code du FC des DLPDU – vue d'ensemble et détail ... 838
C.6 Paramètres SD (statut et description de données) des DLPDU CA, CD, ED et DT ... 842
C.7 Paramètres EC des DLPDU EC ... 845
C.8 Paramètres des DLPDU DC et RC ... 847
C.9 Paramètres des DLPDU TD, RQ et RR ... 848
C.10 Paramètres des DLPDU PN, PT, ES et RI ... 852
C.11 Résumé d'adressage extrait à partir des figures et des tableaux en 4.3 ... 852
Bibliographie ... 857

Figure 1 – Relations des DLSAP, des adresses DLSAP, des DLCEP, des adresses DLCEP, des adresses DLSEP et des adresses DL de groupe ... 429
Figure 2 – Structure de base d'une adresse de DL ... 452
Figure 3 – Structure de base d'un sélecteur de sous-liaison ... 452
Figure 4 – Variantes de structures d'adresse de DL ... 453
Figure 5 – Structure de base des adresses MAC ... 464
Figure 6 – Représentation d'une adresse de DL sous forme d'adresse MAC ... 464
Figure 7 – Relations linéaires des variables numéro de séquence de DLCEP d'envoi et de réception ... 478
Figure 8 – Variantes de structures d'adresse de DL ... 490
Figure 9 – Champ d'adresse de DL SHORT – Variantes de structures implicites ... 490
Figure 10 – Champ d'adresse de DL NODE – structure implicite ... 491
Figure 11 – Diagramme de transition d'états pour un DLCEP ... 612
Figure 12 – Projection des variables «numéro de séquence» de DLCEP expéditeur et destinataire de la Figure 7 sur les paramètres numéro de séquence cyclique des DLPDU CA, CD, DT, ED et RC, avec la détermination consécutive des actions requises ... 630
Figure 13 – Transitions d'états d'une DLE ... 716
Figure 14 – Topologie de réseau ponté ... 734
Figure 15 – Représentation d'arborescence ... 734
Figure 16 – Retard de transit de DLSDU, durée de vie de DLPDU et retard de transmission de pont ... 740
Figure 17 – Transmission et livraison d'une DLPDU reçue ... 744

Figure 18 – Transmission d'une DLPDU d'origine locale 745
Figure 19 – Réédition d'une DLSDU reçue en provenance d'une autre liaison 747
Figure 20 – Architecture de pont 748
Figure 21 – Remplacement pour [IL] Fig 3-2 Ports de pont 758
Figure 22 – Remplacement pour [IL] Fig. 3-3 Architecture de pont 759
Figure A.1 – Exemple de création de FCS 811
Figure A.2 – Exemple de vérification de syndrome de FCS à la réception 812
Figure C.1 – Structure grossière des points de code de FC 839
Figure C.2 [Figure 2] – Structure de base d'une adresse de DL 853
Figure C.3 [Figure 3] – Structure de base d'un sélecteur de sous-liaison 853
Figure C.4 [Figure 4] – Variantes de structures d'adresse de DL 853
Figure C.5 [Figure 5] – Structure de base des adresses MAC 853
Figure C.6 [Figure 6] – Représentation d'une adresse de DL sous forme d'une adresse MAC 853

Tableau 1 – Adressage liaison || nœud || sélecteur 454
Tableau 2 – Adressage liaison locale nœud || sélecteur 457
Tableau 3 – Appellations de nœud locales à une liaison 459
Tableau 4 – Adressage de sélecteur local à un nœud 461
Tableau 5 – Adresses de DL non locales, plates et prédéfinies 462
Tableau 6 – Adresses de DL, locales à une liaison, plates et prédéfinies 463
Tableau 7 – Adresses de DL locales à un nœud prédéfinies 463
Tableau 8 – Corrélation des DLPDU avec classes fonctionnelles 470
Tableau 9 – Longueur de FCS, polynôme et résidu prévu 493
Tableau 10 – Structure sommaire des DLPDU 500
Tableau 11 – Restrictions de DLPDU en fonction du jeton dominant 501
Tableau 12 – Structure des DLPDU EC 502
Tableau 13 – Structure des DLPDU DC 504
Tableau 14 – Structure des DLPDU RC 506
Tableau 15 – Structure des DLPDU CA 508
Tableau 16 – Structure des DLPDU CD 515
Tableau 17 – Structure des DLPDU ED 523
Tableau 18 – Structure des DLPDU DT 532
Tableau 19 – Structure des DLPDU SR 540
Tableau 20 – Structure des DLPDU CT 543
Tableau 21 – Structure des DLPDU TD 544
Tableau 22 – Structure des DLPDU RQ 546
Tableau 23 – Structure des DLPDU RR 549
Tableau 24 – Structure des DLPDU PN 551
Tableau 25 – Structure des DLPDU PR 553
Tableau 26 – Structure des DLPDU PT 555
Tableau 27 – Structure des DLPDU ES 564
Tableau 28 – Structure des DLPDU RT 571

Tableau 29 – Structure des DLPDU RI 572
Tableau 30 – Structure des DLPDU CL 573
Tableau 31 – Structure des DLPDU TL 575
Tableau 32 – Structure des DLPDU WK 579
Tableau 33 – Structure des DLPDU IDLE 580
Tableau 34 – Structure supposée des DLPDU (spare) indéfinies 582
Tableau 35 – Structure supposée des DLPDU RESERVED (NOT TO BE USED) (Réservées, ne devant pas être utilisées) 583
Tableau 36 – Structure des paramètres d'une DLPDU EC 584
Tableau 37 – Paramètres EC: 1er octet 585
Tableau 38 – Paramètres EC: 2ème octet 585
Tableau 39 – Paramètres EC: 3ème et 4ème octets 586
Tableau 40 – Paramètres EC: 5ème et 6ème octets 586
Tableau 41 – Paramètres EC: 7ème octet 587
Tableau 42 – Paramètres EC: 8ème octet 587
Tableau 43 – Paramètres EC: 9ème et 10ème octets 588
Tableau 44 – Paramètres EC: 11ème octet 588
Tableau 45 – Paramètres EC: 12ème octet 589
Tableau 46 – Paramètres EC: 13ème et 14ème octets 589
Tableau 47 – Paramètres DC et paramètres RC: 1er octet 590
Tableau 48 – Paramètres DC et paramètres RC: 2ème octet 590
Tableau 49 – Causes de déconnexion 591
Tableau 50 – Causes de réinitialisation 592
Tableau 51 – Paramètres RC: 3ème octet 592
Tableau 52 – Paramètres RC: 4ème octet 592
Tableau 53 – Structure des DLPDU CA, CD, DT et ED en mode sans connexion 593
Tableau 54 – Paramètres SD de format short (court) pour les initiateurs de transaction en mode sans connexion 594
Tableau 55 – Paramètres SD de format short (court) pour les répondeurs en mode sans connexion 595
Tableau 56 – Statut Reply (réponse) pour les DLPDU DT d'acquittement d'unitdata et de réponse d'échange d'unitdata 595
Tableau 57 – Structure des DLPDU CA, CD, DT et ED orientées connexion 597
Tableau 58 – Paramètres SD de format Short (court) pour l'état de DLCEP 598
Tableau 59 – Paramètres SD de format Long pour l'état de DLCEP 1er octet 598
Tableau 60 – Paramètres SD de format Long pour l'état de DLCEP 2ème octet 598
Tableau 61 – Paramètres SD de format Long pour l'état de DLCEP 3ème octet 599
Tableau 62 – Statut Reply (réponse) pour les DLPDU SR 600
Tableau 63 – Paramètres SR de format Short (court) 601
Tableau 64 – Structure des TD-parameters (paramètres TD) 601
Tableau 65 – Structure et codage pour les mesures de qualité de temps de DL 601
Tableau 66 – Poids numérique approché des bits du temps de DL de sept octets 602
Tableau 67 – Poids numérique approché des bits du temps court de trois octets 603
Tableau 68 – Structure des RQ-parameters (paramètres RQ) 604

Tableau 69 – Structure des RR-parameters (paramètres RR) 604
Tableau 70 – Structure et codage pour les mesures de qualité de temps de RR 604
Tableau 71 – Structure des PN-parameters (paramètres PN) 605
Tableau 72 – Paramètres PN: 1^{er} octet 605
Tableau 73 – Paramètres PN: $2^{ème}$ octet 606
Tableau 74 – Paramètres PN: $3^{ème}$ et $4^{ème}$ octets 606
Tableau 75 – Paramètres PN: $5^{ème}$ octet 606
Tableau 76 – Paramètres PN: $6^{ème}$ octet 606
Tableau 77 – Structure des DD-parameters (paramètres DD) 607
Tableau 78 – Composantes du temps de DL retourné 668
Tableau 79 – Calcul de la synchronisation du temps 670
Tableau 80 – 1^{er} octet de SPDU: Classe de SPDU, et version de protocole ou sous-classe 679
Tableau 81 – Probe-response SPDU 680
Tableau 82 – Versions de protocole de DL prises en charge 680
Tableau 83 – SPDU PR: $3^{ème}$ et $4^{ème}$ octets 680
Tableau 84 – SPDU Node-activation 682
Tableau 85 – SPDU Node-activation $4^{ème}$ octet 682
Tableau 86 – SPDU LAS-data-base-status 683
Tableau 87 – SPDU LAS-data-base-status: $2^{ème}$ octet 683
Tableau 88 – SPDU Live-list- change 684
Tableau 89 – Structure du statut de DLE 684
Tableau 90 – SPDU Live-list-detail 685
Tableau 91 – SPDU DL-conformance-reply 686
Tableau 92 – Versions de protocole de DL prises en charge 686
Tableau 93 – Codage de conformité de DL (partie 1) 687
Tableau 94 – Codage de conformité de DL (partie 2) 687
Tableau 95 – Codage de conformité de DL (partie 3) 688
Tableau 96 – Codage de conformité de DL (partie 4) 688
Tableau 97 – SPDU Link-basic-parameters-reply 689
Tableau 98 – SPDU Link-master-parameters-reply 690
Tableau 99 – SPDU Token-hold-time-request 691
Tableau 100 – SPDU Token-hold-time-array 691
Tableau 101 – Code d'en-tête d'élément de séquence 693
Tableau 102 – Adresse de DL SHORT et élément de séquence Durée 694
Tableau 103 – Adresse de DL LONG et élément de séquence Durée 694
Tableau 104 – Élément de séquence de demande de réveil 695
Tableau 105 – SPDU Schedule-request 695
Tableau 106 – Codage du type de séquence, du type de programme et de la priorité 696
Tableau 107 – SPDU Scheduling-completed 697
Tableau 108 – Statut et codes de cause 698
Tableau 109 – SPDU Cancel-request 698
Tableau 110 – SPDU Schedule-cancelled 699

Tableau 111 – Programme de liaison 700
Tableau 112 – SPDU Schedule-summary 700
Tableau 113 – Référence de SPDU subschedule 701
Tableau 114 – SPDU subschedule 702
Tableau 115 – Sous-SPDU "sequence" 703
Tableau 116 – Element-description (description d'élément) 703
Tableau 117 – SPDU Schedule-summary-request 704
Tableau 118 – SPDU Subschedule-request 705
Tableau 119 – Codage de l'en-tête d'élément de liste de paramètres 705
Tableau 120 – Élément begin/end-of-list 706
Tableau 121 – Élément continuation-of-list 706
Tableau 122 – élément de liste adresse de DL SHORT 706
Tableau 123 – Élément adresse de DL LONG DL 707
Tableau 124 – Élément DLSAP-address-characteristics 708
Tableau 125 – Élément DLCEP-characteristics 708
Tableau 126 – SPDU Address-query 709
Tableau 127 – SPDU address-report 710
Tableau 128 – SPDU Address-list-query 711
Tableau 129 – Critères de sélection d'adresse de DL 712
Tableau 130 – SPDU Address-list-reply 713
Tableau 131 – Format d'une BPDU de notification de changement de topologie 767
Tableau 132 – Format d'une BPDU Configuration 768
Tableau 133 – Erreur maximale de poursuite de phase autorisée dans un sens du temps de DL d'une DLE à la période de Time Distribution (distribution du temps) minimale exigible 777
Tableau C.1 – Affectation générique des points de code FC 839
Tableau C.2 – Affectation individuelle des points de code FC 841
Tableau C.3 – Statut de réponse pour les DLPDU SR 844
Tableau C.4 – Statut de réponse pour les DLPDU DT d'acquittement d'unitdata et de réponse d'échange d'unitdata 845
Tableau C.5 – Poids numérique approché des bits du temps de DL de sept octets 850
Tableau C.6 – Poids numérique approché des bits de N(NT), A…A, et de C(NT) de trois octets 851
Tableau C.7 [Tableau 1] – Adressage liaison || nœud || sélecteur 854
Tableau C.8 [Tableau 2] – Adressage liaison locale || nœud || sélecteur 854
Tableau C.9 [Tableau 5] – Adresses de DL non locales, plates et prédéfinies 855
Tableau C.10 [Tableau 6] – Adresses de DL non locales à une liaison, plates et prédéfinies 855
Tableau C.11 [Tableau 3] – Appellations de nœud locales à une liaison 856
Tableau C.12 [Tableau 4] – Adressage de sélecteur local à un nœud 856
Tableau C.13 [Tableau 7] – Adresses de DL locales à un nœud prédéfinies 856

COMMISSION ÉLECTROTECHNIQUE INTERNATIONALE

RÉSEAUX DE COMMUNICATION INDUSTRIELS – SPÉCIFICATIONS DES BUS DE TERRAIN –

Partie 4-1: Spécification du protocole de la couche liaison de données – Éléments de type 1

AVANT-PROPOS

1) La Commission Electrotechnique Internationale (CEI) est une organisation mondiale de normalisation composée de l'ensemble des comités électrotechniques nationaux (Comités nationaux de la CEI). La CEI a pour objet de favoriser la coopération internationale pour toutes les questions de normalisation dans les domaines de l'électricité et de l'électronique. A cet effet, la CEI – entre autres activités – publie des Normes internationales, des Spécifications techniques, des Rapports techniques, des Spécifications accessibles au public (PAS) et des Guides (ci-après dénommés "Publication(s) de la CEI"). Leur élaboration est confiée à des comités d'études, aux travaux desquels tout Comité national intéressé par le sujet traité peut participer. Les organisations internationales, gouvernementales et non gouvernementales, en liaison avec la CEI, participent également aux travaux. La CEI collabore étroitement avec l'Organisation Internationale de Normalisation (ISO), selon des conditions fixées par accord entre les deux organisations.
2) Les décisions ou accords officiels de la CEI concernant les questions techniques représentent, dans la mesure du possible, un accord international sur les sujets étudiés, étant donné que les Comités nationaux de la CEI intéressés sont représentés dans chaque comité d'études.
3) Les Publications de la CEI se présentent sous la forme de recommandations internationales et sont agréées comme telles par les Comités nationaux de la CEI. Tous les efforts raisonnables sont entrepris afin que la CEI s'assure de l'exactitude du contenu technique de ses publications; la CEI ne peut pas être tenue responsable de l'éventuelle mauvaise utilisation ou interprétation qui en est faite par un quelconque utilisateur final.
4) Dans le but d'encourager l'uniformité internationale, les Comités nationaux de la CEI s'engagent, dans toute la mesure possible, à appliquer de façon transparente les Publications de la CEI dans leurs publications nationales et régionales. Toutes divergences entre toutes Publications de la CEI et toutes publications nationales ou régionales correspondantes doivent être indiquées en termes clairs dans ces dernières.
5) La CEI elle-même ne fournit aucune attestation de conformité. Des organismes de certification indépendants fournissent des services d'évaluation de conformité et, dans certains secteurs, accèdent aux marques de conformité de la CEI. La CEI n'est responsable d'aucun des services effectués par les organismes de certification indépendants.
6) Tous les utilisateurs doivent s'assurer qu'ils sont en possession de la dernière édition de cette publication.
7) Aucune responsabilité ne doit être imputée à la CEI, à ses administrateurs, employés, auxiliaires ou mandataires, y compris ses experts particuliers et les membres de ses comités d'études et des Comités nationaux de la CEI, pour tout préjudice causé en cas de dommages corporels et matériels, ou de tout autre dommage de quelque nature que ce soit, directe ou indirecte, ou pour supporter les coûts (y compris les frais de justice) et les dépenses découlant de la publication ou de l'utilisation de cette Publication de la CEI ou de toute autre Publication de la CEI, ou au crédit qui lui est accordé.
8) L'attention est attirée sur les références normatives citées dans cette publication. L'utilisation de publications référencées est obligatoire pour une application correcte de la présente publication.
9) L'attention est attirée sur le fait que certains des éléments de la présente Publication de la CEI peuvent faire l'objet de droits de brevet. La CEI ne saurait être tenue pour responsable de ne pas avoir identifié de tels droits de brevets et de ne pas avoir signalé leur existence.

L'attention est attirée sur le fait que l'utilisation du type de protocole associé est restreinte par les détenteurs des droits de propriété intellectuelle. En tout état de cause, l'engagement de renonciation partielle aux droits de propriété intellectuelle pris par les détenteurs de ces droits autorise l'utilisation d'un type de protocole de couche avec les autres protocoles de couche du même type, ou dans des combinaisons avec d'autres types autorisées explicitement par les détenteurs des droits de propriété intellectuelle pour ce type.

NOTE Les combinaisons de types de protocole sont spécifiées dans la CEI 61784-1 et la CEI 61784-2.

La Norme internationale CEI 61158-4-1 a été établie par le sous-comité 65C: Réseaux industriels, du comité d'études 65 de la CEI: Mesure, commande et automation dans les processus industriels.

Cette deuxième édition annule et remplace la première édition, parue en 2007. Cette édition constitue une révision technique.

La modification majeure par rapport à l'édition précédente est le suivante.

- amélioration terminologique.

Le texte de cette norme est issu des documents suivants:

FDIS	Rapport de vote
65C/762/FDIS	65C/772/RVD

Le rapport de vote indiqué dans le tableau ci-dessus donne toute information sur le vote ayant abouti à l'approbation de cette norme.

Cette publication a été rédigée selon les Directives ISO/CEI, Partie 2.

NOTE 2 De légers écarts par rapport aux directives ont été autorisés par le Bureau Central de la CEI afin d'assurer la continuité de la numérotation des paragraphes avec les éditions antérieures.

Une liste de toutes les parties de la série CEI 61158, publiées sous le titre général *Réseaux de communication industriels – Spécifications des bus de terrain*, peut être consultée sur le site web de la CEI.

Le comité a décidé que le contenu de cette publication ne sera pas modifié avant la date de stabilité indiquée sur le site web de la CEI sous http://webstore.iec.ch dans les données relatives à la publication recherchée. À cette date, la publication sera:

- reconduite;
- supprimée;
- remplacée par une édition révisée, ou
- amendée.

0 INTRODUCTION

0.1 Généralités

La présente partie de la CEI 61158 est l'une d'une série produite pour faciliter l'interconnexion de composants de systèmes d'automation. Elle est liée à d'autres normes de la série telle que définie par le modèle de référence des bus de terrain "à trois couches" décrit dans la CEI 61158-1.

Le protocole de liaison de données fournit le service de liaison de données en utilisant les services disponibles dans la couche physique. Le but principal de la présente norme est de fournir un ensemble de règles pour la communication exprimées en termes des procédures devant être accomplies par des entités de liaison de données (DLE) d'homologues au moment de la communication. Ces règles pour la communication visent à fournir une base solide pour le développement afin de servir une diversité de besoins:

a) comme un guide pour les réalisateurs et les concepteurs;

b) pour une utilisation dans les essais et achats d'équipements;

c) comme partie intégrante d'un accord pour l'admission de systèmes dans l'environnement des systèmes ouverts;

d) comme affinement pour la compréhension de communications prioritaires au sein de l'OSI.

La présente norme est concernée, en particulier, par la communication et l'interfonctionnement des capteurs, des effecteurs et autres appareils d'automatisation. L'utilisation de la présente norme conjointement à d'autres normes positionnées dans les modèles de référence de l'OSI ou de bus de terrain permet à n'importe quelle combinaison de systèmes autrement incompatibles de fonctionner.

0.2 Nomenclature pour les références dans la présente norme

Les articles, y compris les annexes, peuvent être référencés dans leur totalité, y compris tous les éventuels paragraphes subordonnés, sous la forme “Article N” ou “Annexe N”, N étant le numéro de l'article ou la lettre de l'annexe.

Les paragraphes peuvent être référencés dans leur totalité, y compris tous les éventuels paragraphes subordonnés, sous la forme “N.M” ou “N.M.P” et ainsi de suite, selon le niveau du paragraphe, N étant le numéro du paragraphe ou la lettre de l'annexe tandis que M, P, etc. représentent les niveaux successifs de paragraphe jusqu'au paragraphe d'intérêt inclus.

Lorsqu'un article ou un paragraphe contient un ou plusieurs paragraphes subordonnés, le texte entre le titre de l'article ou du paragraphe et son premier paragraphe subordonné peut être référencé dans sa totalité sous la forme “N.0” ou “N.M.0” ou “N.M.P.0” etc., N, M et P étant comme ci-dessus. Énoncé différemment, une référence terminée par “.0” désigne le texte et les figures entre un titre d'article ou de paragraphe et son premier paragraphe subordonné.

NOTE Cette nomenclature fournit un moyen de référencer un texte dans des articles en sommaire. De tels articles existaient dans des éditions plus anciennes des articles de la CEI 61784-3, Type 1. Ces articles en sommaire sont conservés dans la présente édition afin de réduire au maximum la perturbation à des normes nationales et multinationales existantes ainsi qu'à des documents consortiaux qui référencent cette numérotation antérieure des paragraphes.

RÉSEAUX DE COMMUNICATION INDUSTRIELS – SPÉCIFICATIONS DES BUS DE TERRAIN –

Partie 4-1: Spécification du protocole de la couche liaison de données – Éléments de type 1

1 Domaine d'application

1.1 Généralités

La couche liaison de données fournit des communications de messagerie de base prioritaire entre les appareils dans un environnement d'automation.

Ce protocole fournit le service de liaison de données en utilisant les services disponibles dans la couche physique. La relation entre les Normes internationales pour le service de liaison de données de bus de terrain, le protocole de liaison de données de bus de terrain, le service physique de bus de terrain et les systèmes de gestion est décrite dans la CEI 61158-1.

Ce protocole fournit des opportunités de communication à toutes les entités de liaison de données participantes

a) d'une manière asynchrone cyclique, séquentiellement pour chacune de ces entités de liaison de données, et
b) d'une manière synchrone, de façon cyclique ou acyclique, selon un programme préétabli.

Le protocole spécifié fournit également le moyen de changer le jeu d'entités de liaison de données participantes et de modifier le jeu d'opportunités de communications programmées. Lorsque le jeu d'opportunités de communications programmées est vide, la distribution d'opportunités de communication aux entités de liaison de données participantes est complètement asynchrone.

Par conséquent, ce protocole peut être caractérisé comme en étant un qui fournit un accès asynchrone, mais avec un recouvrement synchrone.

1.2 Spécifications

La présente norme spécifie

a) les procédures pour le transfert en temps utile de données et d'informations de commande d'une entité d'utilisateur de liaison de données vers une entité d'utilisateur homologue, et parmi les entités de liaison de données formant le fournisseur de service de liaison de données distribué;
b) la structure des DLPDU du bus de terrain utilisée pour le transfert de données et d'informations de commande par le protocole de la présente norme, et leur représentation comme unités de données d'interface physique.

NOTE Dans la CEI 61158-4-1, il a été utilisé des zones grisées dans les tableaux pour indiquer que le champ spécifié n'est pas une partie intégrante conceptuelle de la DLPDU spécifique.

1.3 Procédures

Les procédures sont définies en termes des

a) interactions entre les entités DL (DLE) homologues à travers des échanges de DLPDU de bus de terrain;

b) interactions entre un fournisseur de service DL (DLS) et un utilisateur de DLS dans le même système à travers les échanges de primitives DLS;

c) interactions entre un fournisseur de DLS et un fournisseur de service Ph dans le même système à travers l'échange de primitives de services Ph.

1.4 Applicabilité

Ces procédures sont applicables aux instances de communication entre des systèmes qui prennent en charge les services de communications prioritaires au sein d'une couche de liaison de données des modèles de référence de l'OSI ou de bus de terrain et qui requièrent l'aptitude à s'interconnecter dans un environnement d'interconnexion des systèmes ouverts.

Les profils sont un moyen simple à plusieurs attributs de récapituler les capacités d'une mise en œuvre, et donc son applicabilité à divers besoins de communications prioritaires.

1.5 Conformité

La présente norme spécifie également les exigences de conformité pour les systèmes mettant en œuvre ces procédures. Cette norme ne contient pas les essais pour démontrer la conformité à ces exigences.

2 Références normatives

Les documents suivants sont cités en référence de manière normative, en intégralité ou en partie, dans le présent document et sont indispensables pour son application. Pour les références datées, seule l'édition citée s'applique. Pour les références non datées, la dernière édition du document de référence s'applique (y compris les éventuels amendements).

NOTE Toutes les parties de la série CEI 61158, ainsi que la CEI 61784-1 et la CEI 61784-2 font l'objet d'une maintenance simultanée. Les références croisées à ces documents dans le texte se rapportent par conséquent aux éditions datées dans la présente liste de références normatives.

CEI 61158-1:2014, *Réseaux de communication industriels – Spécifications des bus de terrain – Partie 1: Présentation et lignes directrices des séries CEI 61158 et CEI 61784*

CEI 61158-2:2014, *Réseaux de communication industriels – Spécifications des bus de terrain – Partie 2: Spécifications et définitions des services de la couche physique*

CEI 61158-3-1:2014, *Réseaux de communication industriels – Spécifications des bus de terrain – Partie 3-1: Définition des services de la couche liaison de données – Éléments de Type 1*

ISO/CEI 7498-1, *Technologies de l'information – Interconnexion de systèmes ouverts (OSI) – Modèle de référence de base: Le modèle de base*

ISO/CEI 7498-3, *Technologies de l'information – Interconnexion de systèmes ouverts (OSI) – Modèle de référence de base: Dénomination et adressage*

ISO/CEI 8886, *Technologies de l'information – Interconnexion de systèmes ouverts (OSI) – Définition du service de liaison de données*

ISO/CEI 10038:1993, *Technologies de l'information – Télécommunications et échange d'informations entre systèmes – Réseaux locaux – Contrôle d'accès au milieu (MAC) – Ponts*

NOTE Cette édition a été supprimée et remplacée par l'ISO/CEI 15802-3:1998. Cependant, les références particulières dans la présente norme se rapportent à l'édition de 1993.

ISO/CEI 10731, *Technologies de l'information – Interconnexion de systèmes ouverts – Modèle de référence de base – Conventions pour la définition des services OSI*

3 Termes, définitions, symboles et abréviations

Pour les besoins du présent document, les termes, définitions, symboles et abréviations suivants s'appliquent.

3.1 Termes et définitions du modèle de référence

La présente norme, basée en partie sur les concepts développés dans l'ISO/CEI 7498-1 et l'ISO/CEI 7498-3, utilise les termes suivants y définis.

3.1.1 adresse de DL appelée [7498-3]

3.1.2 adresse de DL appelante [7498-3]

3.1.3 connexion centralisée à points d'extrémité multiples [7498-1]

3.1.4 entités (N) correspondantes [7498-1]
entités de DL correspondantes (N=2)
entités de Ph correspondantes (N=1)

3.1.5 démultiplexage [7498-1]

3.1.6 adresse de DL [7498-3]

3.1.7 mapping d'adresse de DL [7498-1]

3.1.8 connexion de DL [7498-1]

3.1.9 point d'extrémité de connexion de DL [7498-1]

3.1.10 identifiant de point d'extrémité de connexion de DL [7498-1]

3.1.11 transmission en mode connexion de DL [7498-1]

3.1.12 transmission en mode sans connexion de DL [7498-1]

3.1.13 collecteur de données de DL [7498-1]

3.1.14 source de données de DL [7498-1]

3.1.15 transmission duplex de DL [7498-1]

3.1.16 fonctionnalité de DL [7498-1]

3.1.17 vue locale de DL [7498-3]

3.1.18 nom de DL [7498-3]

3.1.19 protocole de DL [7498-1]

3.1.20 identifiant de connexion de protocole de DL [7498-1]

3.1.21 informations de contrôle protocolaires de DL [7498-1]

3.1.22 unité de données de protocole de DL [7498-1]

3.1.23 identifiant de version de protocole de DL [7498-1]

3.1.24 relais de DL [7498-1]

3.1.25 identifiant de connexion de service de DL [7498-1]

3.1.26 unité de données de service de DL [7498-1]

3.1.27 transmission simplex de DL [7498-1]

3.1.28 sous-système de DL [7498-1]

3.1.29 données d'utilisateur de DL [7498-1]

3.1.30 contrôle de flux [7498-1]

3.1.31 gestion de couche [7498-1]

3.1.32 multiplexage [7498-3]

3.1.33 autorité (d'adressage) de dénomination [7498-3]

3.1.34 domaine (d'adressage) de dénomination [7498-3]

3.1.35 sous-domaine (d'adressage) de dénomination [7498-3]

3.1.36 entité (N) [7498-1]
entité de DL
entité de Ph

3.1.37 unité de données d'interface (N) [7498-1]
unité de données de service de DL (N=2)
unité de données d'interface de Ph (N=1)

3.1.38 couche (N) [7498-1]
couche DL (N=2)
couche Ph (N=1)

3.1.39 service(N) [7498-1]
service de DL (N=2)
service de Ph (N=1)

3.1.40 point d'accès au service (N) [7498-1]
point d'accès au service de DL (N=2)
point d'accès au service de Ph (N=1)

3.1.41 adresse de point d'accès au service (N) [7498-1]
adresse de point d'accès au service de DL (N=2)
adresse de point d'accès au service de Ph (N=1)

3.1.42 entités homologues [7498-1]

3.1.43 informations de contrôle d'interface Ph [7498-1]

3.1.44 données d'interface de Ph [7498-1]

3.1.45 nom de primitive [7498-3]

3.1.46 réassemblage [7498-1]

3.1.47 recombinaison [7498-1]

3.1.48 réinitialisation [7498-1]

3.1.49 adresse de DL en réponse [7498-3]

3.1.50 acheminement [7498-1]

3.1.51 segmentation [7498-1]

3.1.52 séquencement [7498-1]

3.1.53 subdivision [7498-1]

3.1.54 nom synonyme [7498-3]

3.1.55 gestion-systèmes [7498-1]

3.2 Termes et définitions de convention pour les services

La présente norme utilise également les termes suivants définis dans l'ISO/CEI 10731 tels qu'ils s'appliquent à la couche liaison de données:

3.2.1 acceptant/accepteur

3.2.2 service asymétrique

3.2.3 (primitive) "confirm";
requestor.deliver (primitive)

3.2.4 (primitive) "deliver"

3.2.5 fonctionnalité confirmée de DL

3.2.6 fonctionnalité de DL

3.2.7 vue locale de DL

3.2.8 fonctionnalité obligatoire de DL

3.2.9 fonctionnalité non confirmée de DL

3.2.10 fonctionnalité de DL lancée par le fournisseur

3.2.11 fonctionnalité facultative de fournisseur de DL

3.2.12 primitive de service de DL;
primitive

3.2.13 fournisseur de service de DL

3.2.14 utilisateur de service de DL

3.2.15 fonctionnalité facultative d'utilisateur de DL

3.2.16 indication (primitive)
acceptor.deliver (primitive)

3.2.17 multipoint homologue/à plusieurs homologues

3.2.18 (primitive) "request";
requestor.submit (primitive)

3.2.19 demandeur

3.2.20 (primitive) "response";
acceptor.submit (primitive)

3.2.21 (primitive) "submit"

3.2.22 service symétrique

3.3 Termes et définitions

Pour les besoins du présent document, les termes et définitions suivants s'appliquent.

3.3.1
pont, routeur DL
entité relais de DL qui accomplit des fonctions sélectives de mise en mémoire et acheminement ultérieur et d'acheminement

a) pour relier deux ou plusieurs sous-réseaux de DL (liaisons) afin de former un sous-réseau de DL unifiée (la liaison étendue);et

b) pour fournir un moyen par lequel deux systèmes d'extrémité peuvent communiquer, lorsque l'un au moins des systèmes d'extrémité est périodiquement inattentif au sous-réseau DL d'interconnexion;

et fournit également la synchronisation du temps parmi les liaisons vers lesquelles elle transmet

3.3.2
délocalisation
processus interne à une DLE de conversion d'adresses DL synonymes en une forme canonique en vue de la transmission ou au cours de la réception (voir 5.2.2)

3.3.3
file d'attente de demandes utilisateur d'adresses de DL
file d'attente de priorité au sein d'un FIFO multipriorités, associée à une adresse de DL spécifique d'une DLE, de demandes d'utilisateur de DLS internes au processus, partitionnée en trois sections disjointes:

a) les demandes déjà émises, mais restant disponibles pour être réémises, confirmées, ou les deux – c'est-à-dire qui n'ont pas encore atteint leur délai de confirmation maximal et qui
   – sont encore dans la fenêtre de transmission de DLCEP, ou
   – sont en attente d'une DLPDU de réponse d'acquittement au DLSAP;

b) les demandes qui sont prêtes pour transmission, mais qui n'ont pas complètement été émises;

c) les demandes qui sont soit en attente une demande de DL-COMPEL-SERVICE (service Compel de DL) pour être libérées pour transmission, soit qui se situent à l'extérieur de la fenêtre de transmission du DLCEP, soit les deux

Note 1 à l'article: La section 1 de la file d'attente contient les demandes utilisateur de DLS qui peuvent avoir été communiquées avec succès à un autre utilisateur de DLS. La retransmission de tout ou partie des données d'utilisateur de DLS associées à ces demandes peut être requise, et peut être tentée avant que les demandes ne soient purgées de la partition (par réinitialisation ou acquittement d'homologue sur une DLC d'homologue; par réinitialisation ou temporisation de demande d'utilisateur ou le besoin d'une réutilisation de numéro de séquence sur une DLC à plusieurs homologues).

Note 2 à l'article: La section 2 de la file d'attente contient les demandes d'utilisateur DLS qui sont prêtes pour la transmission, mais pourraient ne pas avoir été complètement communiquées à un autre utilisateur de DLS

Note 3 à l'article: La section 3 de la file d'attente file peut être non vide seulement lorsque la politique de programmation de DL pour l'adresse de DL est EXPLICIT (explicite), ou lorsque le nombre des DLSDU en file d'attente excède la fenêtre de transmission du DLCEP, ou les deux. =

Note 4 à l'article: Les membres d'une priorité donnée sont avancés d'une section à l'autre jusqu'à ce qu'ils soient retirés de la file d'attente. (Dans la pratique, il est permis de déplacer les sections des partitions, plutôt que les membres.) L'ordre FIFO dans chaque priorité est strictement maintenu.

Note 5 à l'article: Une telle file d'attente est associée à chaque adresse de DLSAP, chaque adresse de DLCEP d'homologue ou d'éditeur et chaque DLCEP d'abonné (qui peut être considéré comme étant associé à une adresse de DLCEP implicite) de la DLE, et avec l'adresse de DL de NODE (nœud) de la DLE (voir 3.3.1 et 5.2.2.3).

3.3.4
segment de DL, liaison,
liaison locale
sous-réseau de DL unique dans n'importe lesquelles des DLE connectées peuvent communiquer directement, sans relayage de DL intermédiaire, chaque fois que toutes celles des DLE qui participent à une instance de communication sont simultanément attentives au sous-réseau de DL pendant la/les période(s) de communication tentée

3.3.5
file d'attente de services non programmés de DLE
file d'attente de priorité au sein d'une FIFO multipriorités de

a) références à des files d'attente de demandes utilisateur d'adresse de DL (voir 3.3.3);
b) références à des séquences actives programmées localement (voir Article 11) résultant de demandes de DL-SCHEDULE-SEQUENCE (séquence de programme de DL) (voir 8.4.3.1);
c) DLPDU DT contenant des données utilisateur de DLS qui sont des réponses différées à des DLPDU CD et ED reçues, mises en file d'attente en appui au service DL-UNITDATA-EXCHANGE (échange d'unités de données de DL).

Note 1 à l'article: Voir 4.7.1.17 a) pour une définition plus élaborée.

Note 2 à l'article: Puisqu'il s'agit d'une file d'attente de priorité dans une FIFO multipriorités, les membres sont retirés dans l'ordre des priorités et, au sein d'une même priorité, dans l'ordre FIFO.

3.3.6
DLSAP
point distinctif en lequel des services de DL sont fournis par une entité de DL unique à une unique entité de couche supérieure

Note 1 à l'article: Cette définition, dérivée de l'ISO/CEI 7498-1, est reprise ici pour faciliter la compréhension de la distinction critique entre les DLSAP et leurs adresses de DL. (Voir Figure 1.)

3.3.7
adresse de DL(SAP)
soit une adresse de DLSAP individuelle, désignant un unique DLSAP d'un unique utilisateur de DLS, soit une adresse de DL de groupe désignant potentiellement plusieurs DLSAP, chacun d'un unique utilisateur de DLS

Note 1 à l'article Cette terminologie est choisie parce que l'ISO/CEI 7498-3 ne permet pas l'utilisation du terme "adresse de DLSAP" pour désigner plus d'un seul DLSAP au niveau d'un utilisateur de DLS unique.

3.3.8
adresse de DLSAP (individuelle)
adresse de DL qui désigne un unique DLSAP au sein d'une liaison étendue

Note 1 à l'article: Une entité de DL unique peut avoir plusieurs adresses de DLSAP associées à un seul DLSAP.

3.3.9
adresse de DLCEP
adresse de DL qui désigne soit

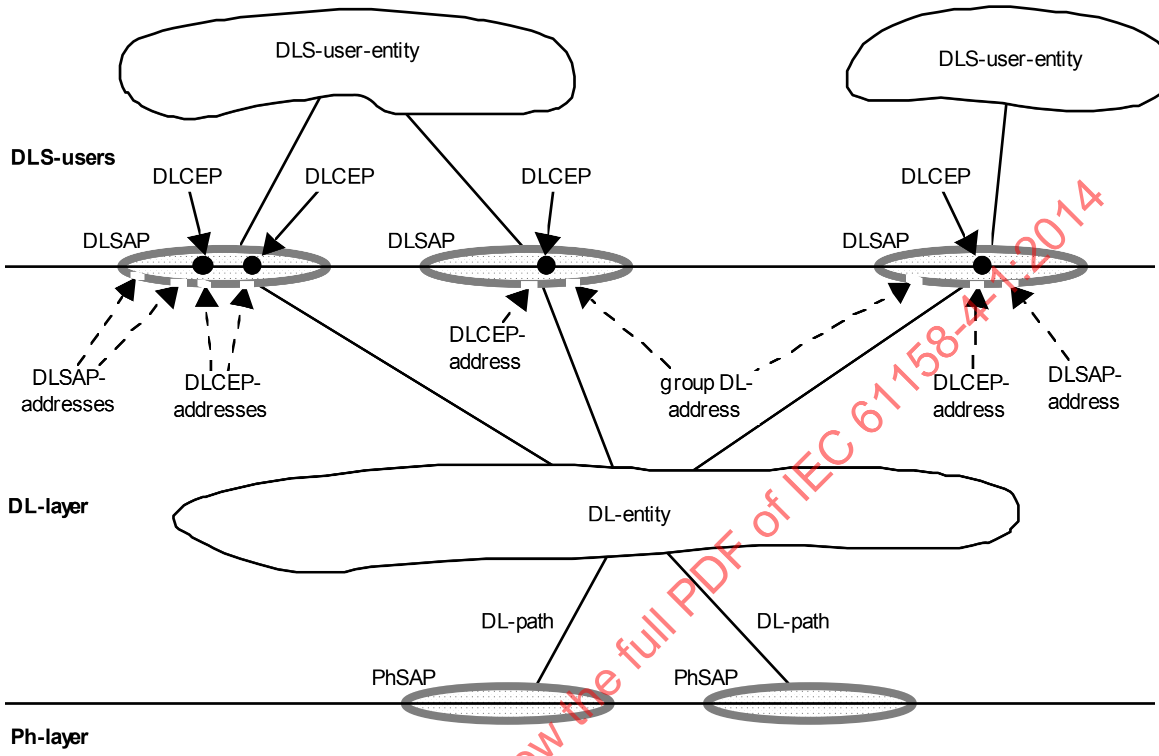
a) un point d'extrémité de connexion de DL d'homologue; soit
b) un point d'extrémité de connexion de DL de plusieurs homologues, et implicitement l'ensemble correspondant de points d'extrémité de connexion de DL

où chaque point d'extrémité de connexion de DL existe au sein d'un DLSAP distinct et est associé à une adresse de DLSAP distincte correspondante

Note 1 à l'article: Il s'agit d'une extension de l'utilisation des adresses de DL au-delà de celle spécifiée dans l'ISO/CEI 7498-3. (Voir Figure 1.)

3.3.10
adresse de DLSEP
adresse de DL qui désigne un point d'extrémité de programmation DL au sein d'une DLE

Note 1 à l'article: Il s'agit d'une extension de l'utilisation des adresses de DL au-delà de celle spécifiée dans l'ISO/CEI 7498-3.



NOTE 1 Les DLSAP et les PhSAP sont illustrés sous la forme d'ovales enjambant la frontière entre deux couches adjacentes.

NOTE 2 Les adresses de DL sont illustrées comme désignant de petits trous (points d'accès) dans la partie DLL d'un DLSAP. Une adresse de DLCEP désigne également un point spécifique de flots d'informations (son DLCEP) au sien du DLSAP.

NOTE 3 Une entité de DL unique peut avoir plusieurs adresses de DLSAP et adresses de DL de groupe associées à un même DLSAP.

NOTE 4 Cette figure montre également les relations des chemins (conduits) de DL et des PhSAP.

Légende

Anglais	Français
DLS-user-entity	Entité utilisateur de DLS
DLS-users	Utilisateurs-de DLS
DLCEP-addresses	Adresses de DLCEP
DLSAP-addresses	Adresses de DLSAP
DL-Layer	Couche DL
DL-entity	Entité de DL
DL-path	Chemin de DL
Group DL-address	Adresse DL de groupe
Ph-layer	Couche Ph (couche physique)
DLCEP	DELCEP
DLSAP	DLSAP
PhSAP	PhSAP

Figure 1 – Relations des DLSAP, des adresses DLSAP, des DLCEP, des adresses DLCEP, des adresses DLSEP et des adresses DL de groupe

3.3.11
jeton dominant
droit unique d'initier la prochaine transmission sur la liaison locale (voir jeton)

Note 1 à l'article: Un jeton de réponse est le jeton dominant sur la liaison locale pendant la période après sa création et avant son expiration ou retour, et le détenteur du jeton de réponse (le répondeur) détient le droit unique d'émettre.

Note 2 à l'article: Autrement, s'il n'existe aucun jeton de réponse, un jeton délégué est le jeton dominant sur la liaison locale pendant la période après sa création et avant son expiration ou retour, et son détenteur du jeton (l'initiateur) détient le droit unique d'émettre.

Note 3 à l'article: À tous les autres moments, le jeton programmeur est le jeton dominant sur la liaison locale, et la DLE LAS (qui fonctionne comme initiateur) détient le droit unique d'émettre.

3.3.12
liaison étendue
sous-réseau de DL, constitué de l'ensemble maximal de liaisons interconnectées par des relais de DL, partageant un même espace (d'adresses de DL) de noms de DL, dans lequel n'importe lesquelles des entités de DL connectées peuvent communiquer, les unes avec les autres, soit directement, soit avec l'aide d'une ou plusieurs entités relais de DL intermédiaires

Note 1 à l'article Une liaison étendue peut être composée juste d'une seule liaison.

3.3.13
facteur d'utilisation fractionnaire
FDC, fractional-duty-cycle
DLE qui n'est pas continuellement attentive à la signalisation reçue, habituellement rencontrée dans un système d'extrémité avec un budget de puissance électrique austère

Note 1 à l'article: Une telle DLE requiert généralement l'assistance d'une DLE pont (relais de DL) pour communiquer avec d'autres DLE sur la même liaison, pour prévoir les cas où les deux DLE ne sont pas actives en même temps.

Trois classes de DLE FDC ont été identifiées:

> **classe A –** les DLE FDC qui sont complètement inattentives à leurs liaisons locales pendant leurs périodes de «sommeil», et commande la temporisation de leur « réveil » temporaire aux communications de liaison seulement en se basant sur le temps de DL interne ou une autre mesure;
>
> **classe B –** les DLE FDC qui surveillent la liaison pendant leur période de «sommeil» pour une DLPDU de réveil particulière (voir 6.21) adressée spécifiquement à l'adresse de DL de nœud de DL, «se réveillant» à la suite de la réception d'une telle DLPDU;
>
> **classe C** – les DLE FDC qui surveillent la liaison pendant leurs périodes de «sommeil» pour toutes les DLPDU adressées spécifiquement à l'adresse DL de nœud de DLE, «se réveillant» à la suite de la réception d'une telle DLPDU

Note 2 à l'article: les DLE qui "sommeillent" entre des réceptions de DLPDU, mais qui répondent à toutes les DLPDU adressées à n'importe lesquelles de leurs adresses de DL, ne sont pas des DLE FDC.

3.3.14
trame
synonyme déconseillé de DLPDU

3.3.15
adresse de DL de groupe
adresse de DL qui désigne potentiellement plus d'un DLSAP au sein d'une liaison étendue

Note 1 à l'article: Une entité de DL unique peut avoir plusieurs adresses de DL de groupe associées à un DLSAP unique. Une entité de DL unique peut aussi une unique adresse de DL de groupe associée à plus d'un DLSAP

3.3.16
réponse immédiate
DLPDU envoyée par le répondeur dans une transaction en tant que réponse sollicitée à la DLPDU immédiatement précédente

Note 1 à l'article: La DLPDU d'origine envoyée par l'initiateur de la transaction spécifie les contraintes sur la DLPDU de réponse.

3.3.17
retard de récupération de réponse immédiate
mesure, similaire à un retard de réponse immédiate, de la période de pire cas d'inactivité PhE d'homologue qui peut être observée

a) par une DLE détenant un jeton délégué pendant l'attente d'une réponse immédiate d'une autre DLE fonctionnant correctement sur la liaison locale;
b) par toute autre DLE sur la liaison locale, après avoir reçu la transaction initiatrice d'une transaction à deux DLPDU, avant de détecter l'activité de liaison induite par la transmission de cette DLPDU de réponse immédiate;
c) par la DLE LAS pendant l'attente de l'activité de liaison initiale après
   – un jeton de réponse (par l'intermédiaire d'une DLPDU PROBE NODE-ADDRESS (PN));
   – un jeton délégué (par l'intermédiaire d'une DLPDU PASS TOKEN (PT) ou EXECUTE SEQUENCE (ES));
   – le jeton programmeur (par l'intermédiaire d'une DLPDU TRANSFER LAS (TL))

   vers une autre DLE fonctionnant correctement sur la liaison locale

Note 1 à l'article: Plus formellement, un retard de récupération de réponse immédiate fournit une borne pour le retard de cas le plus défavorable entre la réception de la primitive "confirm" de Ph-Data pour une primitive "request" de Ph-Data dont la PhIDU a spécifié une fin de données et d'activité (voir 4.4.3), et la réception consécutive de la prochaine PhIDU de début d'activité (voir 4.4.3), qui peut être observé par la DLE d'envoi sur la liaison locale pendant

1) une interaction initiateur-répondeur, dans laquelle une DLE initiatrice de transaction envoie une DLPDU exigeant une réponse immédiate à une DLE de réponse, et cette DLE répondeuse répond en envoyant une seconde DLPDU;
2) une interaction de passage de jeton, dans laquelle la DLE LAS envoie une DLPDU de jeton délégué à la prochaine DLE détentrice de jeton prévue ou le jeton programmeur à une DLE successeur (comme LAS) et cette DLE adressée répond en envoyant une seconde DLPDU.

Note 2 à l'article: La valeur du immediate-response-recovery-delay (c'est-à-dire retard de récupération de réponse immédiate), en unités d'un intervalle de temps, est (maximum-response-delay + 1), c'est-à-dire (retard de réponse maximal + 1).

3.3.18
initiateur
rôle de DLE dans lequel une DLE envoie une DLPDU à une DLE répondeuse homologue, qui envoie immédiatement une DLPDU de réponse vers la DLE initiatrice (et potentiellement vers d'autres DLE) comme partie intégrante de la même transaction

Note 1 à l'article: Certaines normes nationales antérieures ont fait référence à ce rôle comme un rôle "de maître".

3.3.19
programmeur actif de liaison
LAS (Link Active Scheduler)
rôle électif d'une DLE à but spécial qui programme la liaison locale et sert de source locale de temps de DL pour la liaison

Note 1 à l'article: Les fonctions LAS existent dans chaque DLE maître de liaison (LM), qui lutte pour ces fonctions LAS et ensuite les active après avoir détecté l'absence sur la liaison d'une DLE dont les fonctions LAS sont actives. La DLE servant de LAS, parfois appelée "DLE LAS", reçoit et répond aux demandes de programmation issues de toutes les DLE sur la liaison, y compris elle-même. Elle reçoit également et répond aux demandes pour le temps de DL courant.

3.3.20
link-id (identifiant de liaison)
identifiant primaire de deux octets pour la liaison locale, au sein de la liaison étendue, dont les valeurs sont contraintes comme cela est spécifié en 4.3.2.1

3.3.21
Link Master (maître de liaison)
LM
DLE qui peut aussi fournir les fonctions de LAS pour la liaison, y compris l'initialisation et la programmation de la liaison

3.3.20.1
maximum-inactivity-to-claim-LAS-delay
retard maximal d'inactivité pour réclamer le LAS
paramètre de configuration de chaque liaison locale, sa valeur minimum étant le retard interne de cas le plus défavorable de n'importe quelle DLE de maître de liaison et des PhE associées connectées à cette liaison

Note 1 à l'article: Ce retard est calculé comme étant la somme des deux composantes suivantes:

a) le retard interne entre
   1) le moment de la présentation de la dernière PhPDU non silencieuse d'une transmission vers la PhE associée à la DLE en ce point de connexion de la PhE à la liaison locale
   2) le démarrage résultant du temporisateur interne de la DLE qui surveille l'inactivité d'une liaison
b) le retard interne entre
   1) l'expiration du temporisateur en question;
   2) le moment qui en résulte pour la présentation de la première PhPDU non silencieuse d'une transmission de la DLPDU Claim Las (CL) requise par le PhE associée à la DLE en ce point de connexion de la PhE à la liaison locale

L'unité de mesure de ce retard agrégé est un huitième de la période de transmission d'un octet (c'est-à-dire une «période de bit» nominale). La plage des valeurs de ce paramètre va de 1 à 4 095 «périodes de bit» nominales.

3.3.22
maximum-response-delay (retard de réponse maximal)
mesure, en unités d'un intervalle de temps, supérieure à la période de cas le plus défavorable d'une inactivité de PhE locale qui peut être observée par une DLE

a) qui vient juste de recevoir un jeton délégué, avant de lancer la transmission d'une DLPDU;
b) qui vient juste de recevoir un jeton de réponse, avant de lancer la transmission d'une DLPDU de réponse immédiate

Note 1 à l'article: La valeur minimale requise d'une DLE pour un retard de réponse maximal est déterminée en mesurant, à la DLE répondeuse, le retard entre la conclusion de la réception d'une DLPDU demandeuse et le lancement de la transmission de la DLPDU répondeuse immédiatement consécutive.

Note 2 à l'article: Le retard de réponse maximal est un paramètre de configuration de chaque liaison locale, et possède une valeur de 1 à 11. Lorsqu'il est multiplié par la durée d'un intervalle de temps, le produit doit être au moins égal à la plus grande des valeurs de maximum-response-delay requises par chaque DLE actuelle ou anticipée sur la liaison locale.

3.3.23
DLC à homologues multiples
connexion centralisée de DL à points d'extrémité multiples proposant une transmission duplex DL entre un seul utilisateur de DLS distingué, appelé l'éditeur ou l'utilisateur de DLS éditeur, et un ensemble d'utilisateurs de DLS homologues mais pas distingués, appelés collectivement les abonnés ou utilisateurs de DLS abonnés, dans laquelle l'utilisateur de DLS éditeur peut effectuer des envois vers les utilisateurs de DLS abonnés en tant que groupe (mais pas individuellement) et les utilisateurs de DLS abonnés peuvent effectuer des envois vers l'utilisateur de DLS éditeur (mais pas les uns vers les autres)

Note 1 à l'article: Une DLC à homologues multiples fournit toujours un service asymétrique. Il peut également être négocié de fournir seulement un service simplexe de DL, soit de l'éditeur vers les abonnés, soit des abonnés vers l'éditeur. Dans le dernier cas, les caractérisations en éditeur et abonné sont des termes inexacts.

Note 2 à l'article: L'utilisateur de DLS éditeur peut avoir besoin d'employer le contrôle de sa fréquence d'édition, car un utilisateur de DLS abonné ne peut pas exercer le contrôle par flux ou par fréquence sur son entité homologue éditrice. Des considérations similaires s'appliquent aux utilisateurs de DLS abonnés en ce qui concerne leur envoi de DLSDU vers l'utilisateur de DLS éditeur.

3.3.24
nœud
simple entité de DL telle qu'elle apparaît sur une liaison locale

3.3.25
adresse DL de NODE (nœud)
adresse de DL qui désigne l'entité de DL (unique) associée à un nœud unique (voir 3.3.23) sur une liaison locale spécifique

3.3.26
node-id (identifiant de nœud)
identifiant primaire d'un octet pour la DLE sur la liaison locale, dont les valeurs sont contraintes comme cela est spécifié en 4.3.2.2

Note 1 à l'article: Il est également permis de donner une valeur zéro, qui neutralise toute transmission par cette DLE.

3.3.27
temporisateur de nœud
compteur à fréquence réglable, maintenu par la DLE locale, qui est utilisé pour fournir à l'utilisateur de DLS un sens multipartite local du temps de DL (voir 4.5.4 et 8.4.1.1) de sorte qu'une composante du temps de DL en question forme un sens monotone croissant de l'heure locale disponible au sein du système final rattaché

Note 1 à l'article: Conceptuellement, un temporisateur de nœud compte en unités nominales de 2 ms à 13 ms et a une période de plus de 100 ans. Donc, tout compteur réel doit être un compteur binaire dont le bit de poids faible a un poids nominal de 2±N, avec une période de repassage par zéro supérieure à l'intervalle maximal attendu entre les réinitialisations de la DLE

Lorsqu'aucune information relative à l'intervalle maximal n'est disponible, un intervalle de cinq ans peut être supposé.

Note 2 à l'article: Le temporisateur de nœud est également utilisé dans le protocole de DL pour fournir un sens partagé du temps de DL qui est utilisé tant pour synchroniser les actions de programmation de DLE, le cas échéant, que pour synchroniser le taux de dérive des tous les temporisateurs de nœud sur la liaison étendue. Ce dernier est obtenu en ajustant la fréquence de chaque temporisateur de DLE afin que le sens monotone de l'heure locale de la DLE maintienne une relation de phase approximativement constante avec celui de la DLE servant de DLE de référence temporelle.

Cet ajustement est la raison pour laquelle le poids de l'unité de comptage du temporisateur de nœud n'est nominalement que de $2^{\pm N}$ ms.

3.3.28
DLC d'homologues
connexion DL point à point proposant une transmission duplex DL entre deux utilisateurs de DLS homologues dans laquelle chacun peut être un utilisateur de DLS expéditeur et chacun comme utilisateur de DLS destinataire peut être capable d'exercer le contrôle par flux sur son homologue expéditeur

Note 1 à l'article: Une DLC d'homologues est négociée pour fournir soit un service symétrique, soit un service asymétrique. Une DLC d'homologues peut aussi être négociée pour fournir uniquement un service DL simplex.

3.3.29
utilisateur de DLS destinataire
utilisateur de service de DL qui agit comme un destinataire de données d'utilisateur de DL

Note 1 à l'article: Un utilisateur de service de DL peut être simultanément un utilisateur de DLS expéditeur et destinataire.

3.3.30
répondeur
rôle de DLE dans lequel une DLE envoie une DLPDU comme réponse immédiate à une DLPDU reçue en provenance d'une DLE initiatrice homologue, toutes étant partie intégrante d'une seule transaction

Note 1 à l'article: Certaines normes nationales antérieures ont appelé ce rôle un rôle "d'esclave".

3.3.31
utilisateur de DLS expéditeur
utilisateur de service de DL qui agit comme une source de données d'utilisateur de DL

3.3.32
(durée d')intervalle de temps
paramètre de configuration de chaque liaison locale, mesuré en tant que multiple entier de la période de transmission d'un octet complet, avec une valeur intégrale comprise entre 1 octet et 4 095 octets:

Note 1 à l'article: L'intervalle de temps est un paramètre de liaison fondamental ayant plusieurs usages:

– L'intervalle de temps est utilisé pour chaque DLE maître de liaison connectée à la liaison pour déterminer la durée pendant laquelle cette DLE surveille liaison à la recherche d'une inactivité avant d'envoyer une DLPDU CLAIM LAS (CL). L'intervalle de temps est défini de sorte que les périodes nominales de surveillance d'inactivité pour deux DLE qui ont des adresses DL de NODE consécutives et qui ne détiennent aucun jeton diffèrent exactement d'un intervalle de temps.
– L'intervalle de temps est utilisé par chaque DLE sur la liaison locale pour calculer les durées de toutes les autres périodes d'inactivité de liaison que la DLE doit surveiller. Toutes ces durées sont spécifiées sous forme de multiples normalisés ou confitures d'un intervalle de temps
– L'intervalle de temps est une borne supérieure minimum configurée sur l'asynchronisme bidirectionnel maximum dans des communications immédiates entre des DLE interagissantes sur la liaison locale lors d'une tentative de (ré)initialisation de la liaison, maximisé à travers de toutes les paires de DLE sur la liaison locale en question. De ce point de vue, l'intervalle de temps est une mesure agrégée des retards de la mise en œuvre de cas le plus défavorable dans les supports intermédiaires, la PhL, et les interfaces PhL/DLL, qui limitent tous la rapidité de l'interaction bidirectionnelle des DLE sur la liaison locale.

L'intervalle de temps se calcule comme étant la somme

a) des retards de propagation bidirectionnelle de cas le plus défavorable à travers les supports intermédiaires et les PhE intermédiaires, telles que les répétiteurs, entre deux PHE quelconques associées à leurs DLE respectives sur la liaison locale, mesurés entre les points de connexion à la liaison locale de chacune de ces PHE, en tenant compte des retards de cas le plus défavorable des circuits logiques et analogiques internes dans les éventuels répétiteurs intermédiaires;
b) du retard maximal d'inactivité pour réclamer le LAS (maximum-inactivity-to-claim-LAS-delay) conformément à 3.3.32;
c) d'un facteur de sécurité, qui est utilisé pour prendre en compte
   1) la différence de deux retards internes pertinents constitués par
      i) le retard entre
         A) la présentation de la première PhPDU non silencieuse à la PhE associée à la DLE en ce point d'attache de PhE,
         B) l'indication du début d'activité de cette PhE à cette DLE;
      ii) le retard entre
         A) la présentation de la dernière PhPDU non silencieuse à la PhE associée à la DLE en ce point d'attache de PhE,
         B) l'indication de la fin d'activité de cette PhE à cette DLE;
   2) les différences de fréquence dans les horloges des temporisateurs internes parmi les DLE sur la liaison locale;
   3) la résolution limitée des mesures et des erreurs potentielles de mesure;
   4) tout retard supplémentaire nécessaire pour arrondir la somme de a), b) et c) à un multiple entier de la période de transmission d'un octet.

Ces retards sont définis d'une manière permettant de les mesurer.

Une résolution plus fine qu'un octet n'est pas possible sans une connaissance de la PhL associée spécifique, car la temporisation de Ph provient de la PhL et l'interface formelle PhL-DLL ne donne qu'une temporisation d'un octet (voir 4.4).

Les définitions de 3.3.31 à 3.3.34 ne sont pas récursives, car la définition 3.3.31 est basée sur une autre mesure que 3.3.21 à 3.3.34. Les définitions de 3.3.21 à 3.3.34 sont en unités d'un intervalle de temps pour réduire la complexité des mises en œuvre de ce protocole. En conséquence, un seul temporisateur matériel de 8 bits, prénormalisé par l'intervalle de temps, activé sur inactivité du bus, avec des conditions d'arrêt et de redéclenchement spécifiques à une utilisation, peut fournir la fonctionnalité pour tous les temporisateurs à base d'intervalle de temps de ce protocole.

3.3.33
opportunité,
opportunité de DL
attribut d'une référence qui fournit une évaluation du cours temporel de la référence en question

Note 1 à l'article Cet attribut a une importance particulière dans des systèmes de données échantillonnées, qui peuvent avoir besoin de prendre des décisions en fonction de l'opportunité, ou de la non-opportunité, des échantillons de données courantes.

Note 2 à l'article: En règle générale, l'opportunité est un attribut d'utilisateur qui peut être affecté négativement par les diverses couches du système de transport de données. Autrement dit, une référence qui était opportune lorsqu'un utilisateur demandeur l'a présentée à un sous-système de communications de données peut devenir inopportune en raison de retards dans le sous-système de communications.

Note 3 à l'article: L'opportunité de DL est un attribut d'une référence d'utilisateur de DLS reliant la temporisation d'une interaction DLE/ utilisateur de DLS qui écrit ou lit la référence en question dans une ou plusieurs autres interactions DLE/utilisateur de DLS.

Note 4 à l'article: Ces concepts prennent aussi en charge la migration à partir de normes nationales antérieures.

3.3.34
jeton
droit d'émettre sur la liaison locale

Note 1 à l'article: Ce droit est assumé par une DLE lorsqu'elle active ses fonctions LAS. Ce droit peut être délégué à des DLE individuelles, sous réserve de contraintes spécifiées sur son utilisation. Dans tous les cas, ce droit retourne en fin de compte à la DLE qui a activé ses fonctions LAS (la DLE LAS). Chaque jeton est implicitement qualifié par la méthode de son transmission ou appropriation:

a) Un *jeton programmeur* est accaparé et détenu par la DLE LAS, et peut être envoyé à une autre DLE LM sur la liaison locale pour transférer l'activation des fonctions LAS à cette DLE destinataire.

b) Un *jeton délégué* est créé par la DLE LAS et envoyé à une DLE sur la liaison locale, et est retourné à la suite de l'achèvement de son utilisation, ou assumé par la DLE LAS à son expiration.

c) Un *jeton de réponse* est créé par l'actuel détenteur de jeton délégué (ou détenteur de jeton programmateur s'il n'y a pas de détenteur de jeton délégué) et envoyé à une DLE sur la liaison locale, demandant une réponse immédiate; il est retourné avec cette réponse immédiate ou assumé par l'actuel détenteur de jeton délégué à l'expiration de la période de réponse

Note 2 à l'article: Il est possible, bien que pas absolument indispensable, qu'une DLE fonctionne en même temps comme LAS, initiateur et répondeur, déléguant un jeton et recevant ce jeton, demandant une réponse immédiate et recevant cette demande, puis en se répondant à elle-même en tant que demandeuse et se retournant le jeton à elle-même en tant que LAS. Ceci exigerait la transmission d'au moins trois DLPDU – une provenant de chaque rôle.

3.3.35
retard de récupération de jeton
mesure, similaire au retard de réponse maximal, plus grande que la période de cas le plus défavorable d'inactivité de PhE homologue qui peut être observée par la DLE LAS pendant qu'une autre DLE fonctionnant correctement utilise un jeton

Note 1 à l'article: Plus formellement, le retard de récupération de jeton fournit une borne qui est plus élevée que le retard de cas le plus défavorable entre la réception d'une PhIDU de fin d'activité ou de fin de données et d'activité (4.4.4) et la réception ultérieure de la prochaine PhIDU de démarrage d'activité, qui peut être observée par n'importe quelle DLE sur la liaison locale pendant le fonctionnement normal de la liaison. Ce retard ne peut être dépassé seulement que si une DLE échoue alors qu'elle détient un jeton délégué ou programmeur.

Note 2 à l'article: La valeur du token-recovery-delay (c'est-à-dire retard de récupération de jeton), en unités d'un intervalle de temps, peut être n'importe quelle valeur entre (maximum-response-delay + 3) et 14.

3.3.36
transaction
DLPDU unique, ou séquence de DLPDU connexes immédiatement consécutives, résultant d'une seule demande d'utilisateur de DLS

Note 1 à l'article: La DLE envoyant la première DLPDU de la transaction est appelée l'initiatrice; la DLE qui envoie la deuxième DLPDU de la transaction, le cas échéant, est appelée le répondeur/la répondeuse.

Note 2 à l'article: Une entité de DL peut être à la fois une initiatrice et une répondeuse dans la même transaction.

3.4 Symboles et abréviations

3.4.1 Unités de données

3.4.1.1 **BPDU** Bridge Protocol Data Unit (unité de données de protocole de pont) utilisée dans le protocole entre ponts de l'ISO/CEI 10038

3.4.1.2 **pDLSDU** partial DL-Service Data Unit (unité de données de service de DL partielle) – un segment d'une DLSDU à plusieurs segments

3.4.1.3 **SPDU** Support Protocol Data Unit (unité de données de protocole de prise en charge), utilisée pour prendre en charge le protocole DL complet

3.4.2 Variables locales, temporisateurs, compteurs et files d'attente

3.4.2.1	**V(ST)**	Durée d'intervalle de temps	voir 4.7.1.1
3.4.2.2	**V(PhLO)**	per DLPDU PhL Overhead (Surdébit de couche physique par DLPDU)	voir 4.7.1.2
3.4.2.3	**V(MRD)**	maximum response delay (retard de réponse maximal)	voir 4.7.1.3
3.4.2.4	**V(IRRD)**	Immediate Response Recovery Delay (retard de récupération de réponse immédiate)	voir 4.7.1.4
3.4.2.5	**V(MRC)**	Maximum Retry Count (nombre maximal de répétitions de tentative)	voir 4.7.1.5
3.4.2.6	**V(NRC)**	Network Retry Count (Nombre de répétitions de tentative réseau)	voir 4.7.1.6
3.4.2.7	**V(NDL)**	Network DLPDU Lifetime (durée de vie de DLPDU réseau)	voir 4.7.1.7
3.4.2.8	**V(TN)**	this Node (le présent nœud)	voir 4.7.1.8
3.4.2.9	**V(TL)**	this Link (la présente liaison)	Voir 4.7.1.9
3.4.2.10	**V(MEP)**	DL MAC Address Embedding Prefix (préfixe d'imbrication d'adresse MAC de DL)	voir 4.7.1.10
3.4.2.11	**C(RD)**	Remaining Duration Down-counter (compteur dégressif de durée restante)	voir 4.7.1.11
3.4.2.12	**V(MID)**	Minimum Inter-DLPDU Delay (retard minimal entre des DLPDU)	voir 4.7.1.12
3.4.2.13	**T(IRRD)**	Immediate Response Recovery Delay monitor (moniteur de retard de récupération de réponse immédiate)	voir 4.7.1.13
3.4.2.14	**V(RA)**	Reply Address (adresse de réponse)	voir 4.7.1.14
3.4.2.15	**V(OTA)**	Outstanding Transaction Array (matrice de transactions en cours)	voir 4.7.1.15
3.4.2.16	**V(LTI)**	Last Transaction Index (dernier indice de transaction)	voir 4.7.1.16
3.4.2.17	**Q(US)**	Unscheduled Service Queue (file d'attente de service non programmée)	voir 4.7.1.17

3.4.2.18	**V(RID)**	Random Identifier (identifiant aléatoire)	voir 4.7.1.18
3.4.2.19	**C(NT)**	Node Time up-counter (compteur progressif de temps de nœud)	voir 4.7.1.19
3.4.2.20	**V(LSTO)**	Local Link Scheduling Time Offset (décalage temporel de programmation de liaison locale)	voir 4.7.1.20
3.4.2.21	**V(DLTO)**	DL-Time Offset (décalage de temps de DL)	voir 4.7.1.21
3.4.2.22	**V(TQ)**	Time Quality (qualité du temps)	voir 4.7.1.22
3.4.2.23	**V(MD)**	Measured Delay (retard mesuré)	voir 4.7.1.23
3.4.2.24	**V(LN)**	LAS Node (Nœud LAS)	voir 4.7.1.24
3.4.2.25	**V(TSC)**	Classe Time Synchronization (synchronisation du temps)	voir 4.7.1.25
3.4.2.26	**T(TDP)**	Time Distribution Period monitor (moniteur de période de distribution de temps)	voir 4.7.1.26
3.4.2.27	**V(TSL)**	Time Source Link (liaison source de temps)	voir 4.7.1.27
3.4.2.28	**P_U(SDUL)**	DLSDU Length request parameter (paramètre demande de longueur de DLSDU)	voir 4.7.2.1
3.4.2.29	**P_U(SDU)**	DLSDU request parameter (paramètre demande de DLSDU)	voir 4.7.2.2
3.4.2.30	**P_U(MCD)**	Maximum Confirm Delay parameter (paramètre retard de confirmation maximal)	voir 4.7.2.3
3.4.2.31	**T_U(MCD)**	Maximum Confirm Delay monitor (moniteur de retard de confirmation maximal)	voir 4.7.2.4
3.4.2.32	**Q_A(UR)**	User Request Queue (file d'attente des demandes utilisateur)	voir 4.7.3.1
3.4.2.33	**V_C(ST)**	DLCEP State (état de DLCEP)	voir 4.7.4.1
3.4.2.34	**V_C(NP)**	Negotiated DLCEP parameters (paramètres de DLCEP négociés)	voir 4.7.4.2
3.4.2.35	**V_C(N)**	prochain numéro de séquence à attribuer à une DLSDU	voir 4.7.4.3
3.4.2.36	**V_C(R)**	numéro de séquence maximal de DLSDU non transmissible	voir 4.7.4.4
3.4.2.37	**V_C(A)**	numéro de séquence maximal de DLSDU acquittée	voir 4.7.4.5
3.4.2.38	**V_C(M)**	numéro de séquence minimum de DLSDU non émise	voir 4.7.4.6
3.4.2.39	**V_C(MS)**	numéro de segment non émis minimum	voir 4.7.4.7
3.4.2.40	**$V_{C,K}$(SS)**	segments d'une DLSDU devant être envoyés	voir 4.7.4.8

3.4.2.41	**$T_{C,K}$(SS)**	moniteur de segments envoyés pour une DLSDU	voir 4.7.4.9
3.4.2.42	**T_C(SS)**	moniteur simplifié de segments envoyés pour une DLSDU	voir 4.7.4.9.1
3.4.2.43	**V_C(L)**	numéro de séquence de la dernière DLSDU rapportée	voir 4.7.4.10
3.4.2.44	**V_C(H)**	numéro de séquence le plus élevée des DLSDU détectées	voir 4.7.4.11
3.4.2.45	**V_C(HS)**	numéro de segment détecté le plus élevé de la DLSDU de numéro de séquence le plus élevé	voir 4.7.4.12
3.4.2.46	**$V_{C,K}$(MRS)**	segments reçus absents d'une DLSDU	voir 4.7.4.13
3.4.2.47	**$V_{C,K}$(RRS)**	segments de demande de transmission requise d'une DLSDU	voir 4.7.4.14
3.4.2.48	**$T_{C,K}$(RRS)**	moniteur de demande de transmission pour une DLSDU	voir 4.7.4.15
3.4.2.49	**T_C(RAS)**	Residual Activity Stimulus (stimulus d'activité résiduelle)	voir 4.7.4.16
3.4.2.50	**T_C(RAM)**	Residual Activity Monitor (moniteur d'activité résiduelle)	voir 4.7.4.17
3.4.2.51	**V_C(TNA)**	DL-Time of last Network Access (temps de DL du dernier accès réseau)	voir 4.7.4.18
3.4.2.52	**V_B(TW)**	DL-Time of last Buffer Write(temps de DL de la dernière écriture de tampon)	voir 4.7.4.19
3.4.2.53	**V_B(TP)**	DL-Time of production (temps de DL de production)	voir 4.7.4.20
3.4.2.54	**V_B(TS)**	Timeliness Status of Buffer write (statut d'opportunité de l'écriture de tampon)	voir 4.7.4.21
3.4.2.55	**V(DTA)**	Delegation Address (adresse de délégation)	voir 4.7.5.1
3.4.2.56	**V(LL)**	local link live list (liste active de liaison locale)	voir 4.7.5.2
3.4.2.57	**V(TCL)**	token circulation list (liste de circulation de jetons)	voir 4.7.5.3
3.4.2.58	**V(ENRL)**	Expected Non-Response List (liste de non-réponses attendues)	voir 4.7.5.4
3.4.2.59	**V(MST)**	Maximum Scheduled Traffic (trafic programmé maximal)	voir 4.7.5.5
3.4.2.60	**V(MSO)**	Maximum Scheduling Overhead (Surdébit de programmation maximal)	voir 4.7.5.6
3.4.2.61	**V(DMDT)**	Default Minimum Token Delegation Time (temps de délégation de jeton minimal par défaut)	voir 4.7.5.7

3.4.2.62	**V(DTHT)**	Default Token Holding time (temps de détention de jeton par défaut)	voir 4.7.5.8
3.4.2.63	**V(LTHT)**	Link Maintenance Token Holding Time (temps de détention de jeton de maintenance de liaison)	voir 4.7.5.9
3.4.2.64	**V(MTHA)**	Maximum Token Holding Time Array (matrice des temps de détention maximaux de jetons)	voir 4.7.5.10
3.4.2.65	**V(TTRT)**	Target Token Rotation Time (temps ciblé de rotation de jetons)	voir 4.7.5.11
3.4.2.66	**V(ATRT)**	Actual Token Rotation Time (temps effectif de rotation de jeton)	voir 4.7.5.12
3.4.2.67	**V(RTHA)**	Remaining Token Holding Time Array (matrice des temps de détention de jetons restants)	voir 4.7.5.13
3.4.2.68	**V(NTHN)**	Next Token Holding Node (prochain nœud de détention de jetons)	voir 4.7.5.14
3.4.2.69	**V(FUN)**	First Unpolled Node id (identifiant du premier nœud non interrogé)	voir 4.7.5.15
3.4.2.70	**V(NUN)**	Number of consecutive Unpolled Node ids (nombre d'identifiants de nœuds non interrogés consécutifs)	voir 4.7.5.16
3.4.2.71	**P(TRD)**	Token Recovery Delay (retard de récupération de jeton)	voir 4.7.5.17
3.4.2.72	**V(TDP)**	Time Distribution Period (période de distribution de temps)	voir 4.7.5.18
3.4.2.73	**V(MICD)**	Maximum-Inactivity-to-Claim-LAS-Delay retard maximal d'inactivité pour réclamer le LAS	voir 4.7.5.19
3.4.2.74	**V(LDDP)**	LAS Data Base Distribution Period (période de distribution de base des données LAS)	voir 4.7.5.20
3.4.2.75	**V(ML)**	Maximum Link (Liaison maximum)	voir 4.7.6.1

3.4.3 Classes de DLPDU

3.4.3.1	**CA**	Compel Acknowledgment (forcer l'acquittement)	Voir 6.4
3.4.3.2	**cd**	Compel Data (forcer les données)	Voir 6.5
3.4.3.3	**CL**	Claim LAS (réclamer le LAS)	voir 6.19
3.4.3.4	**CT**	Compel Time (forcer le temps)	Voir 6.9
3.4.3.5	**DC**	Disconnect Connection (déconnecter la connexion)	Voir 6.2
3.4.3.6	**DT**	Données	Voir 6.7

3.4.3.7	**EC**	Establish Connection (établir une connexion)	Voir 6.1
3.4.3.8	**ED**	Exchange Data (échanger les données)	Voir 6.6
3.4.3.9	**ES**	Execute Sequence (exécuter une séquence)	voir 6.16
3.4.3.10	**Repos**	Repos	voir 6.22
3.4.3.11	**PN**	Probe Node address (Sonder une adresse de nœud)	voir 6.13
3.4.3.12	**PR**	Probe Return(Retour de sondage)	voir 6.14
3.4.3.13	**Pt**	Pass Token (transmettre un jeton)	voir 6.15
3.4.3.14	**RC**	Reset Connection (réinitialiser une connexion)	Voir 6.3
3.4.3.15	**RI**	Request Interval (demander l'intervalle)	voir 6.18
3.4.3.16	**RQ**	Round Trip Delay Query (interrogation de temps de propagation aller-retour)	voir 6.11
3.4.3.17	**RR**	Round Trip Delay reply (réponse de temps de propagation aller-retour)	voir 6.12
3.4.3.18	**RT**	Return Token (retourner un jeton)	voir 6.17
3.4.3.19	**SR**	Status Response (statut réponse)	Voir 6.8
3.4.3.20	**TD**	Time distribution (distribution de temps)	voir 6.10
3.4.3.21	**TL**	Transfer LAS (transférer LAS)	voir 6.20
3.4.3.22	**WK**	wakeup (réveil)	voir 6.21

3.4.4 Divers

3.4.4.1	**BME**	Bridge Management Entity (entité de gestion de pont)
3.4.4.2	**DLC**	DL-connection (Connexion de DL)
3.4.4.3	**DLP-**	préfixe désignant le protocole DL
3.4.4.4	**DLPCI**	Data Link Protocol Control Information (informations de commande de protocole de liaison de données)
3.4.4.5	**DLSEP**	DL-Schedule Endpoint (point d'extrémité de programme de DL)
3.4.4.6	**FC**	Frame Control (commande de trame) (premier octet de chaque DLPDU)
3.4.4.7	**FDC**	Fractional Duty Cycle (facteur d'utilisation fractionnaire) (type de DLE)
3.4.4.8	**LAS**	Link Active Scheduler (programmeur actif de liaison)
3.4.4.9	**lm**	link master (maître de liaison)
3.4.4.10	**PhICI**	Physical Interface Control Information (informations de contrôle d'interface physique)
3.4.4.11	**PhID**	Physical Interface Data (données d'interface physique)
3.4.4.12	**SLAE**	Systems Load Application Entity (entité d'application de charge des systèmes)
3.4.4.13	**SMAE**	Systems Management Application Entity (entité d'application de gestion système)

4 Vue d'ensemble du protocole de DL

4.1 Modèle à trois niveaux de la DLL

La DLL est modélisée comme étant

a) un bas niveau des fonctions d'accès et de programmation de chemin, qui supporte,

b) un niveau intermédiaire de fonctions d'exploitation de pont, qui à son retour supporte,

c) un niveau supérieur de fonctions de service de DL, de coordination de ponts, de transfert de données en mode connexion et sans connexion.

Interfonctionnant avec tous les trois niveaux, les fonctions de gestion DL comprennent notamment les fonctions de gestion de chemins redondants et de ponts, le cas échéant.

NOTE 1 Le terme «sous-couche» n'est pas approprié pour la description de ces niveaux, car l'ISO/CEI 7498 exige que lorsque plusieurs sous-couches sont définies, toutes sauf une seule doivent être facultatives.

NOTE 2 Ce partitionnement sur trois niveaux ressemble étroitement au partitionnement en fonctions «MAC» de niveau inférieur, «exploitation de pont» de niveau intermédiaire et « LLC» de niveau supérieur rencontré dans les normes LAN de l'ISO/CEI (voir ISO/CEI TR 8802-1 pour la définition des services MAC), avec les deux différences significatives suivantes.

a) La fonctionnalité sur deux niveaux de ce protocole est entièrement contenue dans la Couche liaison de données telle que spécifiée par ISO/CEI 7498. Par contre, la fonctionnalité «MAC» des protocoles LAN de l'ISO/CEI s'étend sur la partie inférieure de la Couche liaison de données de l'OSI et sur la partie supérieure de la Couche physique de l'OSI.

b) Ce protocole utilise un niveau unique d'adressage DL dans la Couche liaison de données. Par contre, les protocoles LAN de l'ISO/CEI utilisent deux niveaux d'adressage DL, l'un au sein de la fonctionnalité «MAC» et «exploitation de pont» et le second au sein de fonctionnalité «LLC

4.1.1 Niveau d'accès et de programmation de chemin

Le niveau d'accès et programmation de chemin fournit une communication de base d'une DLE vers une autre. La temporisation de cette communication est régulée

a) pour fournir une distribution équitable des occasions pour des communications arbitraires à toutes les DLE sur la liaison locale, généralement de façon cyclique mais asynchrone,

b) pour fournir une distribution spécifique d'occasions pour des communications désignées aux DLE sur la liaison locale selon un programme préétabli pour la liaison

C'est la dernière exigence relative aux communications programmées, en appui aux communications prioritaires, qui amène ce protocole à s'appuyer plus lourdement sur la gestion temps réel centralisée des occasions de communication que les protocoles antérieurs tels que l'ISO/CEI 8802-3, l'ISO/CEI 8802-4 et ISO/CEI 8802-5. Le gestionnaire centralisé est sélectionné dans l'ensemble des DLE pleines fonctions opérantes sur la liaison locale par une procédure similaire à celle utilisée dans l'ISO/CEI 8802-4 et l'ISO/CEI 8802-5. Si ce gestionnaire venait à échouer, un remplacement est sélectionné d'une manière similaire.

NOTE Plusieurs algorithmes traditionnels de commande utilisés dans le domaine d'applicabilité de ce protocole requièrent des contraintes strictes sur la temporisation cycle à cycle de l'acquisition d'informations. Dans les systèmes distribués, de telles contraintes se traduisent fréquemment par des limites sur la périodicité admissible de cycle en cycle de la messagerie qui achemine ces informations acquises de façon répétitive.

c) Le niveau d'accès et de programmation de chemin crée forme chaque DLPDU à partir des informations de contrôle de protocole DL et des données utilisateur de DLS; calcule et aboute une séquence de contrôle de trame (FCS) appropriée; et transmet le tout sous la forme d'une séquence de PhIDU (voir 4.4) à la PhE en vue d'une transmission vers les PhE homologues pour un rapport aux DLE homologues.

   Dans certains cas, il aboute aussi les trois octets de faible ordre, soit

   – de la valeur courante du temporisateur de nœud local, C(NT); soit

   – de la valeur calculée moins la valeur courante du temporisateur de nœud local, C(NT),

   pendant la formation de la DLPDU, précédant immédiatement la FCS aboutée.

d) Le niveau d'accès et de programmation de chemin reçoit une séquence de PhIDU issues de la PhE, concatène ces PhIDU en une DLPDU reçue, calcule une séquence de contrôle de trame sur toute la séquence de données reçues, et vérifie la valeur résiduelle correcte. Le premier octet de la séquence reçue est examiné pour déterminer le type de DLPDU, et une tentative est effectuée pour analyser cette DLPDU en ses composantes "informations de contrôle de protocole de DL " et "données utilisateur DLS". Si la FCS résiduelle était correcte et l'analyse a été couronnée de succès, les actions appropriées de bas niveau inférieur sont entreprises, éventuellement y compris le fait de rapporter la DLPDU analysée à un niveau supérieur.

   Dans certains cas, la valeur des trois octets d'ordre inférieur du temporisateur de nœud local, C(NT), au moment de la réception, est aboutée à cette DLPDU analysée.

e) Le niveau d'accès et de programmation de chemin fournit les fonctions de base tant du répondeur que de l'initiateur. Comme répondeur, il fournit les fonctions de séquencement nécessaires

   1) pour recevoir une DLPDU, acheminant éventuellement un jeton de réponse;

   2) dans le dernier cas (du jeton de réponse reçu), pour envoyer une DLPDU comme réponse immédiate à la DLPDU juste reçue.

   Comme initiateur, il fournit les fonctions de séquencement nécessaires pour

   3) recevoir un jeton délégué;

   4) envoyer une ou plusieurs DLPDU, y compris celles exigeant une réponse immédiate;

   5) recevoir une telle réponse immédiate ou inférer son absence;

   6) retourner un jeton délégué.

f) Le niveau d’accès et de programmation de chemin fournit une fonctionnalité de programmation de bas niveau requise pour programmer des transmissions de DLPDU sur un chemin spécifié, y compris toute interaction avec le programmeur actif de liaison (LAS) relatif à la liaison locale pour coordonner le programme avec d’autres DLE ou demander la capacité nécessaire de transmission de chemin.

Les actions de c) et d) sont augmentées au sein un relais de pont DLE pour permettre la retransmission d'une séquence reçue d'octets de données, en incluant la FCS reçue, avec d'éventuelles altérations contraintes du premier octet et en compensant les altérations (voir 5.2.5.3) de la FCS reçue, avant la retransmission.

Les actions en e) et f) sont fondées en partie sur deux files d'attente de gestion et de programmation de demandes:

- une file d'attente de demandes spécifiques à une adresse DL (voir 3.3.3), associée à l’adresse d’envoi, qui est utilisée pour gérer les demandes d'utilisateurs de DLS provenant de cette adresse DL;
- la file d'attente de services non programmés de DLE commune (voir 4.7.1.17), qui est utilisée pour gérer la prise en charge de demandes non programmées à la suite de la réception du jeton "mis en circulation".

Certaine des fonctionnalités plus complexes de la programmation de ce niveau requièrent, et utilisent, les services du niveau supérieur (voir 4.1.3) en agissant comme quasi-utilisateur de DLS interne à la DLE.

4.1.2 Niveau d'exploitation de pont

Le niveau d'exploitation de pont fournit la fonctionnalité de niveau intermédiaire consistant à

a) interconnecter logiquement plusieurs liaisons locales en une seule liaison étendue en interconnectant physiquement plusieurs chemins;

b) servir de commutateur "temps-espace-temps" éventuellement distribué:

1) en fournissant des fonctions de mise en mémoire et acheminement ultérieur de DLPDU pour permettre la communication entre des DLE sur la liaison étendue qui ne pourraient autrement pas communiquer;

NOTE Ceci inclut la coordination avec des DLE de facteur d'utilisation fractionnaire (FDC) (voir 3.3.32) nécessaires pour permettre des périodes alternées de fonctionnement en «DLE FDC éveillée» et «DLE FDC endormie».

2) en fournissant des fonctions de substitution pour permettre des interactions de réponse différée entre les DLE sur la liaison comme une extrapolation des transactions de «réponse immédiate» sur la liaison locale;

c) fournir un sens partagé du temps de DL tout au long de la liaison étendue;

d) coordonner la programmation de liaison locale parmi au moins deux liaisons locales afin de fournir toute coordination nécessaire de liaisons multiples de la programmation dans la liaison étendue.

Une grande partie de la fonctionnalité de niveau exploitation de pont n'est active que dans les DLE de «ponts» (voir 4.6).

4.1.3 Niveau de service de DL, de coordination d ponts, de transfert de données en mode connexion et sans connexion

Le niveau de service de DL, de coordination de ponts, de transfert de données en mode connexion et sans connexion fournit la fonctionnalité consistant à

a) gérer toutes les interactions de DLE avec l'utilisateur de DLS, convertir toutes les primitives "request" et "response" de l'utilisateur de DLS en la nécessaire séquence d'opérations de DLE; et générer des primitives "indication" et "confirm" de l'utilisateur de DLS s'il y a lieu;

b) gérer le séquencement de chaque DLCEP actif, y compris
   1) la détermination du type et de la séquence des DLPDU devant être émises à partir du DLCEP;
   2) la négociation de la QoS;
   3) la détermination des DLPCI devant être incluses dans chaque DLPDU;
   4) la segmentation et le réassemblage des grandes DLSDU;
c) gérer le séquencement des transactions d'unitdata qui exigent une réponse de la part d'une DLE homologue, y compris
   1) la détermination des DLPCI devant être incluses dans chaque DLPDU;
   2) la corrélation d'une DLPDU de réponse non immédiate avec la transaction demandeuse;
d) traiter toutes les DLPDU de distribution d'informations d'état entre DLE, telles que les DLPDU TIME DISTRIBUTION (TD) (distribution du temps, voir 6.10 et 8.4.1) et bridge configuration (configuration de ponts, voir 10.4) et les SPDU de secours de LAS (voir Article 9);
e) traiter toutes les interactions DLPDU d'interrogation/DLPDU de réponse avec d'autres DLE, autres que celles qui ont lieu comme «réponse immédiate» et qui sont réalisées sur une base réactive par les fonctions de niveau inférieur de 4.1.1.

Ces interactions DLPDU d'interrogation/DLPDU de réponse comprennent aussi le calcul des temps de propagation aller-retour (voir 6.11, 6.12, et 8.4.1), la prise en charge des demandes DL-SUBSCRIBER-QUERY (voir 8.2.3) et DL-LISTENER-QUERY (voir 8.3.4) lancées à distance, les échanges entre ponts d'informations relatives aux états de pont (voir 10.4).

4.2 Service fourni par la DLL

Le DLL fournit des services de transfert de données sans connexion pour les DLSDU de taille limitée, des services de transfert de données en mode connexion pour les DLSDU de taille limitée, un service de temps synchronisé en interne, des services de programmation pour commander l'allocation de temps du service PhL partagé sous-jacent, et un service de gestion d'adresses DL(SAP), de files d'attente et de tampons.

NOTE La CEI 61158-3-1 montre plusieurs des relations entre les attributs QoS de DLC.

Certains attributs de QoS pertinents sont comme suit:

4.2.1 QoS – Classe de DLCEP

Chaque demande d'établissement de DLCEP spécifie la classe du DLCEP. Les trois choix pour la classe de DLCEP sont:

a) **PEER** – l'utilisateur de DLS peut échanger des DLSDU avec un autre utilisateur de DLS homologue;
b) **PUBLISHER** – l'utilisateur de DLS peut envoyer des DLSDU vers un ensemble de zéro à plusieurs utilisateurs de DLS abonnés associés et recevoir des DLSDU issues de n'importe lesquels de ces utilisateurs de DLS abonnés;
c) **SUBSCRIBER** – l'utilisateur de DLS peut recevoir, et demander, des DLSDU issues de l'utilisateur de DLS éditeur associé et peut envoyer des DLSDU vers cet utilisateur de DLS éditeur.

4.2.2 QoS – Caractéristiques de livraison de données de DLCEP

Les deux membres d'une DLC d'homologues ou l'utilisateur de DLS éditeur d'une DLC à plusieurs homologues spécifie(nt) les caractéristiques de livraison de données du/des DLCEP de la DLC. Les cinq choix pour les caractéristiques de livraison de données de DLCEP, et leurs effets, sont:

a) **CLASSICAL** – l'utilisateur de DLS peut envoyer des données qui seront livrées sans perte, duplication ou ordre incorrect. Tous les utilisateurs de DLS concernés recevront notification de toute perte de synchronisation sur la DLC.

b) **DISORDERED** – l'utilisateur de DLS peut envoyer des données qui seront livrées immédiatement dès la réception, sans duplication, mais potentiellement dans un ordre différent de celui de l'utilisateur de DLS expéditeur. Tous les utilisateurs de DLS concernés recevront notification de toute perte irrécupérable de données d'utilisateur de DLS ou perte de synchronisation sur la DLC.

c) **ORDERED** – l'utilisateur de DLS peut envoyer des données qui seront livrées immédiatement dès réception, sans duplication ou ordre incorrect, mais avec perte potentielle de certaines données d'utilisateur de DLS. La perte de données d'utilisateur de DLS ne sera pas rapportée et il ne sera pas tenté de récupérer les données d'utilisateur de DLS perdues avant les données d'utilisateur de DLS rapportées en dernier.

d) **UNORDERED** – l'utilisateur de DLS peut envoyer des données qui seront livrées immédiatement dès réception. La perte, la duplication et l'ordre incorrect de données d'utilisateur ne seront ni détectés ni rapportés. Aucune tentative de redressement sur de tels événements ne sera faite par le fournisseur de DLS.

e) **NONE** – l'utilisateur de DLS ne peut pas envoyer de données dans ce sens de transfert de données.

Sur une DLC d'homologues, la valeur de QoS pour la caractéristique de livraison de données du DLCEP expéditeur peut être choisie de façon indépendante pour chaque sens de transfert de données. Sur une DLC à plusieurs homologues, la valeur de QoS pour les caractéristiques de livraison de données de DLCEP pour le sens "abonnés vers éditeur" du transfert de données est limitée à UNORDERED et à NONE. La valeur de QoS par défaut pour les caractéristiques de livraison de données de DLCEP dans le sens "éditeur vers abonnés" est UNORDERED.

4.2.3 QoS – Priorité de DLL

Toutes les demandes et réponses d'établissement de DLCEP, toutes les demandes de transfert de données en mode sans connexion et un grand nombre de demande de programmation de DL spécifient une priorité de DLL associée qui est utilisée pour programmer des services de transfert de données de DL. Cette priorité de DLL détermine également la quantité maximale de données utilisateur de DLS qui peut être acheminée dans une seule DLPDU. Les trois priorités de DLL avec leurs gammes correspondantes de données utilisateur de DLS (par DLPDU) acheminables sont, de la priorité la plus élevée à la plus basse:

a) **URGENT** – ≤ 64 octets utilisateur de DLS par DLPDU;

b) **NORMAL** – ≤ 128 octets utilisateur de DLS par DLPDU;

c) **TIME-AVAILABLE** – ≤ 256 octets utilisateur de DLS par DLPDU.

NOTE 1 Les niveaux URGENT et NORMAL sont considérés comme étant des niveaux de priorité **à temps critique**; Le niveau TIME-AVAILABLE (temps disponible) est considéré comme étant un niveau de priorité **à temps non critique**.

NOTE 2 Les DLPDU d'établissement de DLC et de libération de DLC qui sont envoyées à la priorité TIME-AVAILABLE sont limitées à ne pas acheminer plus de données utilisateur de DLS qu'il ne serait permis à la priorité NORMAL – 128 octets.

4.2.4 QoS – Authentification de DLPDU

Chaque négociation pour l'établissement de DLCEP (et chaque transfert de données en mode sans connexion) utilise cet attribut pour déterminer

a) une limite inférieure sur la quantité d'informations d'adressage de DL utilisées dans les DLPDU qui fournissent les services associés de transfert de données de DLL;

NOTE Cela a un léger impact sur le taux résiduel de livraisons erronées de DLPDU; un plus grand nombre d'informations d'adressage réduit le potentiel d'erreurs de livraison.

b) s'il convient, oui ou non, que l'état courant d'un DLCEP d'homologue expéditeur ou d'éditeur soit envoyé à basse fréquence au(x) DLCEP d'homologue(s) ou d'abonné(s) de DLC même lorsque aucune demande d'utilisateur de DLS non confirmée n'est en attente en cours au niveau du DLCEP expéditeur;

NOTE Cette transmission en arrière-plan continue est appelée "**residual activity**" (activité résiduelle).

c) s'il convient, oui ou non, que des actions de programmation connexes soient exécutées localement.

NOTE Ces deux derniers aspects sont particulièrement importants dans les systèmes de sécurité.

Les trois niveaux spécifiables, avec leurs quantités d'informations d'adressage de DL, sont

1) **ORDINARY** – chaque DLPDU doit inclure la quantité minimale d'informations d'adressage nécessaires;
2) **SOURCE –** chaque DLPDU doit inclure une adresse de DL source lorsque cela est possible;
3) **MAXIMAL** – chaque adresse de DL doit inclure la quantité maximale d'informations d'adressage possibles. En outre, il convient que toutes les actions de programmation connexes soient exécutées localement; et il convient que chaque DLCEP d'homologue expéditeur ou d'éditeur de la DLC maintienne un rapport basse fréquence des informations d'état lorsqu'il n'y a pas d'activité d'utilisateur de DLS.

4.2.5 QoS – Retard de confirmation maximal de DLL

Chaque demande d'établissement de DLC/DLCEP (et chaque réponse) spécifie des limites supérieures sur la durée maximale permise pour l'achèvement

a) d'une primitive de service DL-CONNECT, DL-RESET ou DL-SUBSCRIBER-QUERY, et, séparément;
b) d'une primitive de service DL-DATA.

Chaque demande de service en mode sans connexion spécifie une limite supérieure sur la durée maximale permise pour l'achèvement

c) d'une primitive de service DL-UNITDATA confirmé localement, et, séparément;
d) une primitive de service DL-UNITDATA confirmé à distance, une primitive de service DL-LISTENER-QUERY ou une instance du service DL-UNITDATA-EXCHANGE.

Chaque paramètre soit a la valeur UNLIMITED, soit spécifie un intervalle, en unités de 1 ms, allant de 1 ms à 60 s compris. La valeur UNLIMITED assure la compatibilité avec des protocoles OSI antérieurs et fournit un moyen pour que les demandes de DL-CONNECT restent dans un état “listening” (à l'écoute) ou “half-open” (à moitié ouvertes). Le statut d'achèvement “timeout” (temporisation) ne peut pas se produire sur une demande d'utilisateur de DLS qui spécifie UNLIMITED.

Les paramètres pour les primitives de DL-DATA et de DL-UNITDATA confirmées localement spécifient des intervalles inférieurs ou égaux à celui pour les primitives de DL-CONNECT, DL-RESET, DL-SUBSCRIBER-QUERY, DL-UNITDATA confirmées à distance et DL-LISTENER-QUERY.

Les intervalles spécifiés sont les retards admissibles maximaux

- entre la transmission des primitives "request" spécifiées et la transmission des primitives "confirm" correspondantes;
- entre le déclenchement et l'achèvement d'une seule instance du service d'édition ou d'échange d'unidata spécifié.

NOTE Pour les DLE qui ne prennent pas en charge une résolution temporelle de 1 ms, l'intervalle de temps demandé peut être arrondi par excès au prochain plus grand multiple de la résolution que la DLE prend effectivement en charge ou à environ 60 s si la DLE n'a aucun sens du temps.

L'échec à parachever une "request" de DL-CONNECT ou de DL-RESET dans l'intervalle spécifié doit entraîner une libération du DLCEP initiée par le fournisseur de DLS et éventuellement celle de la DLC.

4.2.6 QoS – Politique de programmation de DL

Pour chaque adresse de DLSAP, et chaque DLCEP, l'utilisateur de DLS peut neutraliser la politique de DLL normale (implicitement programmée) consistant à fournir le service de DL demandé dès que possible, et, à la place, il peut différer n'importe quelle communication entre utilisateurs de DLS exigée par une primitive DLS "request" de DL-DATA ou de DL-UNITDATA jusqu'à ce que le différé soit libéré par un utilisateur de DLS impliqué. Chaque libération de ce type, par l'exécution d'une demande de DL-COMPEL-SERVICE, spécifiant l'adresse de DLSAP ou le DLCEP, permet l'achèvement d'une et une seule primitive de DLS "request" différée. Seuls les services de DL qui fournissent l'intercommunication d'utilisateurs de DLS sont affectés par cet attribut.

Les deux choix sont:

a) **IMPLICIT** – toutes les éventuelles communications exigées avec un ou plusieurs utilisateurs de DLS homologues à partir de cette adresse de DLSAP ou de ce DLCEP, auront lieu dès que possible;

b) **EXPLICIT** – une communication exigée de données ou de unitdata avec un ou plusieurs utilisateurs de DLS homologues à partir de cette adresse de DLSAP, ou de ce DLCEP, aura lieu seulement lorsque le différé est explicitement libéré par un utilisateur de DLS impliqué.

NOTE Le programmation de la transmission de DLPDU pour prendre en charge les services DL-CONNECT, DL-RESET et DL-DISCONNECT et pour prendre en charge les réponses différées par un répondeur dans les services DL-UNITDATA-EXCHANGE et DL-UNITDATA confirmé à distance, est toujours IMPLICIT. La programmation pour la transmission de DLPDU pour lancer le service DL-UNITDATA-EXCHANGE est toujours EXPLICIT.

4.2.7 QoS – tailles maximales de DLSDU

Chaque demande d'établissement de DLC/DLCEP (et chaque réponse) spécifie une limite supérieure pour la taille (en octets) des DLSDU qui seront proposées à la transmission et une limite supérieure pour la taille des DLSDU qui sont acceptables pour la réception.

Pour les DLC d'homologues, la taille maximale de DLSDU négociée doit être déterminée indépendamment pour chaque sens du transfert de données comme étant la plus petite parmi celle qui est offerte par l'expéditeur, celle qui est permise par la gestion de DL locale de l'expéditeur, celle qui est permise par la gestion de DL locale du destinataire et celle qui est permise par le destinataire.

La taille offerte par l'expéditeur peut être de n'importe quelle valeur comprise entre zéro et 16 fois le nombre maximum d'octets d'utilisateur de DLS par DLPDU, tel que spécifié en 4.2.3. La DLE de réception et tous les agents de gestion DL doivent choisir leurs tailles maximales permises dans la liste de tailles suivante:
0, 64, 128, 256 $\times$ N où $1 \leq N \leq 16$.

NOTE 1 La taille maximum de DLSDU prise en charge par ce protocole est 16 fois le nombre maximal d'octets des données d'utilisateur de DLS acheminables dans une seule DLPDU, tel que déterminé par la DLL Priority (priorité de DLL) de la DLC (voir 4.2.3).

NOTE 2 L'ensemble de tailles maximales de DLSDU permises est limité à la petite liste de valeurs ci-dessus afin de promouvoir l'interfonctionnement. Il est permis à la taille maximale spécifiée par l'expéditeur de prendre n'importe quelle valeur dans la plage permise par la priorité DLL de la DLC afin de faciliter l'optimisation de la capacité des communications partagées de la DLL.

NOTE 3 Une valeur de zéro (0) correspond au choix du service simplex, tel que spécifié par l'utilisateur de DLS par le choix NONE tel que décrit en 4.2.2e).

Pour les DLC à plusieurs homologues, la taille maximale de DLSDU négociée doit être la plus petite parmi celle proposée par l'éditeur et celle autorisée par la gestion locale de DL de l'éditeur. Pour les abonnés de DLC à plusieurs homologues, la DLC doit être refusée par le fournisseur de DLS (la DLE de l'abonné) si la taille maximale de DLSDU établie par l'éditeur est supérieure à la plus petite taille parmi celle qui est permise par l'abonné et celle qui est permise par la gestion locale de DL de l'abonné.

La taille offerte par l'éditeur dans le sens "éditeur vers abonnés" peut être de n'importe quelle valeur comprise entre zéro et 16 fois le nombre maximum d'octets d'utilisateur de DLS par DLPDU, tel que spécifié en 4.2.3. La taille offerte par l'éditeur dans le sens "abonné vers éditeur" peut être de n'importe quelle valeur comprise entre zéro et le nombre maximum d'octets d'utilisateur de DLS par DLPDU, tel que spécifié en 4.2.3.

Les abonnés et tous les agents de gestion DL doivent choisir leurs tailles maximales permises dans la liste de tailles suivante: 0, 64, 128, 256 × *N* où $1 \leq N \leq 16$.

NOTE 4 La taille maximum de DLSDU prise en charge par ce protocole dans le sens "éditeur vers abonnés" est 16 fois le nombre maximal d'octets des données d'utilisateur de DLS acheminables dans une seule DLPDU, tel que déterminé par la DLL Priority (priorité de DLL) de la DLC (voir 4.2.3).

NOTE 5 La taille maximum de DLSDU prise en charge par ce protocole dans le sens "abonné vers éditeur" est le nombre maximal d'octets des données d'utilisateur de DLS acheminables dans une seule DLPDU, tel que déterminé par la DLL Priority (priorité de DLL) de la DLC (voir 4.2.3).

NOTE 6 L'ensemble de tailles maximales de DLSDU permises est limité à la petite liste de valeurs ci-dessus afin de promouvoir l'interfonctionnement. Il est permis à la taille maximale de DLSDU spécifiée par l'éditeur de prendre n'importe quelle valeur dans la plage permise par la priorité DLL de la DLC afin de faciliter l'optimisation de la capacité des communications partagées de la DLL.

NOTE 7 Une valeur de zéro (0) correspond au choix du service simplex, tel que spécifié par l'utilisateur de DLS par le choix NONE tel que décrit en 4.2.2e).

La valeur par défaut pour la taille maximale des DLSDU tant de l'expéditeur que du destinataire est le nombre maximal d'octets d'utilisateur de DLS qui peuvent être transportés par une seule DLPDU ayant la priorité de DLL spécifiée. Le fournisseur de DLS doit prendre toujours en charge cette taille de DLSDU.

4.2.8 QoS – Liaison de tampon et file d'attente de DLCEP et d'adresse de DLSAP

Chaque demande d'établissement de DLCEP (et chaque réponse) peut lier au DLCEP un(e) ou deux tampons rétenteurs locaux ou files d'attentes FIFO de profondeur spécifiée, créé(e)s par des primitives de gestion DL-CREATE buffer and queue (ou par la gestion de DL).

NOTE Lorsque ces liaisons sont effectuées pour un utilisateur de DLS d'une DLC d'homologues, ou pour un utilisateur de DLS éditeur d'une DLC à plusieurs homologues, elles déterminent la fenêtre maximale de transmission (à savoir le nombre des DLSDU émises mais non acquittées) pour ce sens du transfert de données de DLC. Comme la taille de la fenêtre de transmission peut également être limitée par la gestion de DL, ou par une mise en œuvre, la profondeur de la file d'attente impose seulement une limite supérieure pour la taille de la fenêtre.

a) Une file d'attente ou un tampon rétenteur peut être lié(e) à un DLCEP dans le but d'acheminer des DLSDU de l'utilisateur de DLS vers le fournisseur de DLS.
b) Une file d'attente ou un tampon rétenteur peut être lié(e) à un DLCEP dans le but d'acheminer des DLSDU du fournisseur de DLS vers l'utilisateur de DLS.
c) Il est également possible de lier une file d'attente ou un tampon rétenteur afin qu'il/elle soit écrit(e) en un DLCEP donné et qu'il/elle soit la source de données en un autre DLCEP. Un(e) tel(le) tampon ou file d'attente intermédiaire peut servir à franchir des limites de programmation ou redistribuer à un second ensemble d'utilisateurs de DLS les données d'utilisateur de DLS reçues.

Une telle liaison est effectuée en spécifiant, pour le paramètre approprié, un paramètre buffer-or-queue DL-identifier qui a découlé d'une "request" antérieure de DL-CREATE (ou d'une action de la gestion de DL) et qui n'a pas encore été supprimée.

Lorsque des caractéristiques de livraison de données du DLCEP expéditeur spécifient UNORDERED ou ORDERED, l'expéditeur et aussi le(s) destinataire(s) peuvent spécifier une politique de mise en file d'attente BUFFER-R ou QUEUE. Lorsque des caractéristiques de livraison de données d'envoi du DLCEP spécifient DISORDERED ou CLASSICAL, l'expéditeur et aussi le(s) destinataire(s) peuvent spécifier une politique de mise en file d'attente QUEUE. la politique de mise en file d'attente BUFFER-R n'est pas permise. La politique de mise en file d'attente BUFFER-NR n'est jamais permise.

Chaque demande de liaison d'adresse de DLSAP peut lier à l'adresse de DLSAP jusqu'à six tampons rétenteurs locaux ou files d'attente non rétentrices locales ou files d'attente FIFO de profondeur spécifiée, créé(e)s par des primitives de gestion DL-CREATE buffer and queue (ou par la gestion de DL).

d) Un tampon ou une file d'attente peut être lié(e) au sens d'expédition d'une adresse de DLSAP à chaque priorité dans le but d'acheminer des DLSDU de l'utilisateur de DLS vers le fournisseur de DLS. Les tampons ne peuvent être liés qu'aux adresses de DLSAP dont le rôle de DLSAP est INITIATOR (initiateur) ou CONSTRAINED RESPONDER (répondeur contraint) ou UNCONSTRAINED RESPONDER (répondeur non contraint). Les files d'attente ne peuvent être liées qu'aux adresses de DLSAP dont le rôle de DLSAP BASIC (de base).

e) Un tampon ou une file d'attente peut être lié(e) au sens de réception d'une adresse de DLSAP à chaque priorité dans le but d'acheminer des DLSDU du fournisseur de DLS vers l'utilisateur de DLS. Les tampons ne peuvent être liés qu'aux adresses de DLSAP dont le rôle de DLSAP est INITIATOR (initiateur) ou CONSTRAINED RESPONDER (répondeur contraint) ou UNCONSTRAINED RESPONDER (répondeur non contraint). Les files d'attente peuvent être liées à toutes les adresses de DLSAP.

4.2.8.1 Liaison à un tampon

Lorsqu'un tampon expéditeur est lié à un DLCEP, ou à une adresse de DLSAP et une priorité de DLL, par un utilisateur de DLS,

a) une primitive "request" de service DL-PUT écrase le tampon avec une DLSDU ou peut vider le tampon;

   NOTE Après la création, le tampon est vide.

b) une primitive "request" de service DL-COMPEL, spécifiant soit

   1) un DLCEP, soit

   2) un échange d'unitdata à une adresse de DLSAP

   entraîne la transmission, à la première occasion, du contenu le plus récent du tampon à cet instant; la primitive ne spécifie pas elle-même une DLSDU.

c) Une primitive "indication" de DL-BUFFER-SENT notifie à l'utilisateur de DLS du DLCEP spécifique sur lequel le tampon rétenteur a été émis, et auquel le tampon est lié, que le tampon venait juste d'être émis.

d) une primitive "indication" de DL-UNITDATA-EXCHANGE notifie au seul utilisateur de DLS attaché à l'adresse de DLSAP spécifique à partir de laquelle le tampon avait été émis (et à laquelle le tampon est lié) que le tampon venait juste d'être émis sur cette adresse de DLSAP, et vidé s'il est non rétenteur (BUFFER-NR).

Lorsqu'un tampon récepteur est lié à un DLCEP, ou à une adresse de DLSAP et une priorité de DLL, par un utilisateur de DLS,

e) une primitive "indication" de DL-BUFFER-RECEIVED ou DL-UNITDATA-EXCHANGE notifie à l'utilisateur de DLS l'écrasement du tampon par la DLSDU nouvellement reçue; la primitive ne spécifie pas elle-même une DLSDU.

f) une primitive "indication" de DL-GET copie la DLSDU du tampon et vide le tampon s'il est non rétenteur (BUFFER-NR).

Plusieurs liaisons de sortie simultanées à un tampon rétenteur sont permises comme option de mise en œuvre et de conformité (voir 10.6.1), mais ne sont pas requises par ce protocole.

4.2.8.2 Liaison à une file d'attente FIFO

Lorsqu'une file d'attente FIFO d'envoi d'une profondeur maximale *K* est liée à un DLCEP, ou à une adresse de DLSAP et une priorité de DLL, par un utilisateur de DLS,

a) une primitive "request" de DL-PUT n'est pas permise;

b) une primitive "request" de DL-DATA ou DL-UNITDATA tente d'abouter une DLSDU à la file d'attente, mais échoue si la file d'attente contient déjà *K* DLSDU. Si l'opération d'aboutement a réussi, la DLSDU sera émise à la première occasion, après toutes les DLSDU qui la précèdent dans la file d'attente.

Lorsqu'une file d'attente FIFO de réception d'une profondeur maximale *K* est liée à un DLCEP, ou à une adresse de DLSAP et une priorité de DLL, par un utilisateur de DLS,

c) Une primitive "request" de DL-GET tente de retirer une DLSDU de la file d'attente, mais échoue si la file d'attente est vide.

d) Une primitive "indication" de DL-DATA ou DL-UNITDATA ou DL-UNITDATA-EXCHANGE notifie à l'utilisateur de DLS destinataire le résultat de l'aboutement d'une DLSDU nouvellement reçue à la file d'attente de réception; la primitive ne spécifie pas elle-même une DLSDU.

Plusieurs liaisons de sortie simultanées à une file d'attente FIFO sont permises comme option de mise en œuvre, mais ne sont pas requises par ce protocole.

4.2.8.3 Liaisons par défaut

Lorsque ces options de liaison ne sont pas spécifiées, ou pour des primitives de DLS pour lesquelles une liaison explicite n’est pas applicable, les interfaces conventionnelles d’expédition explicitement placées en file d'attente et de réception directe entre l’utilisateur de DLS et le fournisseur de DLS sont utilisées. Dans ce cas, chaque primitive DL-DATA et DL-UNITDATA achemine une DLSDU, et chaque primitive DL-CONNECT, DL-DISCONNECT et DL-RESET peut acheminer une DLSDU:

a) les primitives "request" de DL-PUT et DL-GET ne sont pas permises;

b) une primitive "request" de DL-DATA ou DL-UNITDATA ou DL-DISCONNECT, ou une primitive "request" ou "response" de DL-CONNECT ou DL-RESET, émise par l'utilisateur de DLS expéditeur tente d'abouter une DLSDU à la file d'attente implicite, mais échoue si la file d'attente est pleine. Si l'opération d'aboutement a réussi, la DLSDU sera émise à la première occasion, après toutes les DLSDU qui la précèdent dans la file d'attente.

c) une primitive "indication de DL-DATA ou DL-UNITDATA ou DL-DISCONNECT, ou une primitive "indication" ou "confirm" de DL-CONNECT ou DL-RESET, notifie à un utilisateur de DLS destinataire une DLSDU nouvellement reçue et achemine cette DLSDU vers l'utilisateur de DLS. Aucune mise en file d'attente apparente n'est fournie au sein de la DLL.

4.2.9 Opportunité de DL

Cet attribut s'applique seulement à des tampons de DL, à des DLCEP auxquels des tampons de DL sont liés, et aux primitives de DLS qui transfèrent des données d'utilisateur de DLS vers/depuis des tampons de DL au niveau de ces DLCEP.

Chaque demande d'établissement de DLCEP (et chaque réponse) peut spécifier des critères d'opportunité de DL qui doivent s'appliquer aux informations envoyées à partir des tampons rétenteurs, ou reçues en provenance de ceux-ci, au niveau du DLCEP en question. Quatre types d'opportunité de DL peuvent être pris en charge: opportunité RESIDENCE (résidence), opportunité UPDATE (mise à jour), opportunité SYNCHRONIZED (synchronisée), et opportunité TRANSPARENT (transparente). Tous les quatre types d'opportunité, et le cas où il n'y a pas d'opportunité, sont montrés à la Figure 6 de la CEI 61158-3-1.

a) L'opportunité *RESIDENCE* est une évaluation basée sur la durée pendant laquelle une référence d'utilisateur de DLS a été résidente dans un tampon, qui est l'intervalle de temps entre

1) le moment où il est écrit dans le tampon (par une primitive "request" DL-PUT ou par réception dans le tampon au niveau d'un DLCEP);
2) le moment où il est lu dans le tampon (par une primitive "request" DL-GET ou par transmission à partir tampon au niveau d'un DLCEP);

$$\text{Opportunité de DL} \equiv 0 \leq (R_T - W_T) < \Delta T \qquad (1)$$

NOTE Ce type d'opportunité est également appelée “*Asynchronous*” (asynchrone).

b) L'opportunité *UPDATE* (mise à jour) est une évaluation basée sur l'intervalle de temps entre
   1) le moment de la survenue d'un événement de synchronisation à plusieurs DLE (une primitive "indication" de DL-BUFFER-RECEIVED ou une primitive "indication" de DL-BUFFER-SENT);
   2) le moment où il est écrit dans le tampon (par une primitive "request" DL-PUT ou par réception dans le tampon au niveau d'un DLCEP);

$$\text{Opportunité de DL} \equiv 0 \leq (WT - ST) < \Delta T \qquad (2)$$

NOTE Un type d'opportunité étroitement liée à celle-ci est également appelée “*Punctual*” (ponctuelle).

c) L'opportunité *SYNCHRONIZED* (synchronisée) est une évaluation basée sur les intervalles de temps et les relations de temporisation entre
   1) le moment de la survenue d'un événement de synchronisation à plusieurs DLE (une primitive "indication" DL-BUFFER-RECEIVED ou une primitive "indication" DL-BUFFER-SENT);
   2) le moment où il est écrit dans le tampon (par une primitive "request" DL-PUT ou par réception dans le tampon au niveau d'un DLCEP);
   3) le moment où il est lu dans le tampon (par une primitive "request" DL-GET ou par transmission à partir tampon au niveau d'un DLCEP);

$$\text{Opportunité de DL} \equiv 0 \leq (WT - ST) \leq (RT - ST) < \Delta T \qquad (3)$$

NOTE Ce type d'opportunité est également appelée “*Synchronous*” (synchrone).

d) L'opportunité *TRANSPARENT* (transparente) a lieu lorsque l'opportunité est sélectionnée sur un DLCEP, mais aucune des évaluations ci-dessus n'a été effectuée. Dans ce cas, la connexion de liaison de données (DLC) préserve toute opportunité de tampon antérieure, mais n'invalide pas elle-même l'opportunité en question Lorsqu'il n'existe aucune opportunité de tampon antérieure, la valeur de l'opportunité par défaut doit être TRUE (vraie).
e) L'opportunité *NONE* (aucune) a lieu lorsque l'opportunité n'est pas sélectionnée sur un DLCEP. Dans ces cas, l'attribut DL-timeliness (opportunité de DL) des données d'utilisateur de DLS doit être FALSE (faux).

Lorsqu'une lecture ou écriture de tampon a lieu sur un intervalle de temps significatif, et pas seulement en tant qu'événement momentané, l'opportunité globale de l'opération de lecture ou d'écriture doit être calculée comme étant l'opportunité au début de l'opération de lecture ou d'écriture, soumise à une opération AND (ET logique) avec l'opportunité à la fin de l'opération de lecture ou d'écriture, le tout en utilisant les mêmes critères d'opportunité.

4.2.10 Confirmation par la DLE distante

Chaque demande de transfert d'unitdata peut spécifier si, oui ou non, la confirmation de réception de la DLSDU associée par la DLE distante (implicitement adressée) est exigée. Ses valeurs permises sont **TRUE** et **FALSE**.

NOTE Le choix de la valeur TRUE utilise inévitablement plus de capacité de liaison que ne le fait le choix de la valeur FALSE.

4.3 Structure et définition des adresses de DL

Les adresses de DL sont utilisées comme les adresses de DLSAP et les adresses de DL de groupe, comme les adresses de DLCEP, et comme les adresses de DLSEP Les adresses de DL se conforment aux limitations de structure et de codage suivantes.

Le Paragraphe 4.3 définit la forme d'adresses de DL et l'utilisation de diverses plages de composantes d'adresse de DL. Il inclut des définitions spécifiques pour certaines adresses de DL normalisées.

Ce protocole DL utilise des adresses DL individuelles (non de groupe) pour d'autres objectifs que simplement comme 'adresses DLSAP. La même terminologie et les considérations suivantes s'appliquent également à ces adresses DL autres que de DLSAP.

NOTE 1 Cet usage étend la définition des adresses de DL au-delà de celle qui est spécifiée dans l'ISO/CEI 7498-3.

NOTE 2 Les adresses de DL sont également traitées dans une certaine mesure en 5.1.

4.3.1 Forme des adresses DL

Une adresse de DL normalisée peut être considérée comme étant constituée de deux parties: appellation de liaison et sélecteur de sous-liaison. L'appellation de liaison est un entier non signé ayant une longueur de deux octets. Le sélecteur de sous-liaison a également une longueur de deux octets.

Appellation de liaison	**Sélecteur de sous-liaison**
2 octets	2 octets

Figure 2 – Structure de base d'une adresse de DL

La plupart des valeurs d'une appellation de liaison spécifient une liaison individuelle. Dans ces cas, le sélecteur de sous-liaison spécifie une adresse de DL unique dans la liaison locale désignée, telle que décrite en plus de détail en 4.3.2.2 et 4.3.2.3. Lorsqu'il manque des informations de transmission spécifiques à une adresse, les ponts (les DLE de relais) doivent baser leurs décisions de transfert uniquement sur la partie "appellation de liaison" de ces adresses.

Toutefois, plusieurs valeurs de l'appellation de liaison spécifient que la partie "appellation de liaison" de l'adresse est concaténée avec la partie "sélecteur de sous-liaison" pour former une adresse non hiérarchique (c'est-à-dire plate). Lorsqu'il manque des informations spécifiques à une adresse, les ponts ne doivent pas transmettre de telles adresses.

Un sélecteur de sous-liaison normalisé en soi est constitué de deux sous-parties: appellation de nœud et sélecteur de sous-nœud. Chacune de ses sous-parties est un petit nombre entier non signé ayant une longueur d'un octet.

Appellation de liaison implicite (0000)	**Appellation de nœud**	**Sélecteur de sous-nœud**
	1 octet	1 octet

Figure 3 – Structure de base d'un sélecteur de sous-liaison

La plupart des valeurs d'une appellation de nœud spécifient un nœud individuel. Dans ces cas, le sélecteur de sous-nœud spécifie une adresse de DL unique au sein du nœud local désigné.

Toutefois, certaines valeurs de l'appellation de nœud spécifient que la partie "appellation de nœud" de l'adresse est concaténée avec la partie "sélecteur de sous-nœud" pour former une sous-adresse non hiérarchique (c'est-à-dire plate).

Cette structure quasi hiérarchique des appellations de liaison, des appellations de nœud et des sélecteurs de sous-nœud avait été choisie en premier lieu pour faciliter l'administration des adresses DL et en second lieu pour réduire le nombre d'entités dans les tables de transmission d'adresses de ponts.

NOTE Les mises en œuvre de protocole peuvent trouver que cette structure partiellement hiérarchique est utile au cours de la reconnaissance d'adresses.

Les possibilités structurelles qui en résultent pour les adresses DL sont (“||” indiquant la concaténation des sous-champs d'appellations nominales de liaison, de nœud et de sélecteur):

Liaison	**Nœud**	**Sélecteur**
2 octets	1 octet	1 octet

ou

Liaison	**Nœud || Sélecteur**
2 octets	2 octets

ou

Liaison || Nœud || Sélecteur
4 octets

Figure 4 – Variantes de structures d'adresse de DL

Une adresse DL dans laquelle l'appellation de liaison a la valeur est appelée adresse *locale*, car son contexte pour l'interprétation est limité à la liaison locale. Par contre, d'autres adresses DL sont appelées adresses *étendues*, car leur contexte d'interprétation est la liaison étendue complète.

Il y a quatre types distincts d'adresses DL visibles aux utilisateurs de DLS:

a) Les adresses DLSAP, qui sont associées à un seul DLSAP dans tout le contexte d'interprétation d'adresse entier;

b) Les adresses DL de groupe, qui sont potentiellement associées à plusieurs DLSAP dans une ou plusieurs DLE, ou avec plusieurs DLE;

c) les adresses DLCEP, qui sont associées

   1) à un seul DLCEP et DLSAP associé d'une DLC peer-to-peer, ou

   2) directement par le DLCEP éditeur et le DLSAP associé d'une DLC à plusieurs homologues, et indirectement avec tous les DLCEP abonnés et DLSAP associés d'une DLC à plusieurs homologues;

d) les adresses de DLSEP, qui sont associées à un point d'extrémité de séquence d'une seule DLE.

Les adresses de DLSAP peuvent être utilisées avec toutes les primitives du DLS qui spécifient une adresse de DLSAP. Les adresses de DL de groupe ne doivent pas être

1) liées à une file d'attente comme adresse de DLSAP source, ou

2) utilisées en tant qu'adresse de DLSAP source dans n'importe quelle primitive de DLS orienté connexion ou sans connexion.

Les adresses DL individuelles autres que de DLSAP peuvent être utilisées au besoin au sein du protocole de DL et, lorsque cela est permis, comme paramètres spécifiques au sein des services de DL.

4.3.2 Valeurs et plages prédéfinies pour les composantes d'une adresse de DL

Certaines valeurs de la composante "appellation de liaison" ou "appellation de nœud" des adresses de DL désignent une adresse de DL de groupe. Toutes les autres combinaisons de valeurs des composantes "appellation de liaison" et "appellation de nœud" désignent des adresses de DL individuelles. Cela est récapitulé du Tableau 1 au Tableau 3.

NOTE Toutes les valeurs de liaison, de nœud et de sélecteur sont spécifiées en hexadécimal (base 16).

4.3.2.1 Appellations de liaison

La composante "appellation de liaison" d'une adresse DL désigne habituellement une seule liaison locale spécifique. Les valeurs de la composante "appellation de liaison" sont prédéfinies, ou leur utilisation est contrainte, comme suit:

0000 La liaison locale. La valeur zéro (0000) est fournie pour la commodité de l'unification des primitives de service, pour permettre aux mises en œuvre de se référer à l'adresse de la liaison locale par une constante zéro (0000), et pour permettre le fonctionnement en liaison isolée lorsqu'aucun pont n'est présent pour identifier la composante de liaison d'une liaison étendue. Lorsque la QoS d'authentification de DLPDU ne spécifie pas *MAXIMAL* et les adresses de DL elles-mêmes le permettent, les composantes "appellation de liaison" de toutes les adresses DL doivent être omises avant transmission. Autrement, l'identifiant de liaison nominale relatif à la liaison locale source, s'il est connu, doit remplacer la valeur de liaison zéro (0000) avant la transmission.

0001 … 003F Adresses de DL de groupe non locales. Ces valeurs d'appellation de liaison, concaténées aux valeurs d'appellation de nœud et valeurs de sélecteur associées – *liaison* || *nœud* || *sélecteur* – forment un espace adresse de 4 128 768 identifiants utilisables comme adresses de DL de groupe. Les adresses de DL qui en résultent ne désignent implicitement aucune liaison spécifique.

Les adresses DL de groupe non locales doivent être allouées individuellement ou en blocs contigus. Lorsqu'elles sont considérées comme étant des nombres entiers non signés de 32 bits, les premières huit adresses (dont les valeurs sont les plus faibles) de ce type sont réservées à une utilisation par le protocole de DL; les 56 adresses suivantes de ce type sont réservées pour des adresses de DL de groupe normalisées. Toutes les autres adresses sont disponibles pour être utilisées comme adresses de DL de groupe non normalisées.

0040 … 007F Adresses de DL de groupe globalement administrées non locales. Ces valeurs d'appellation de liaison, concaténées aux valeurs d'appellation de nœud et valeurs de sélecteur associées – *liaison* || *nœud* || *sélecteur* – forment un espace adresse de 4 194 304 identifiants pour les adresses de DL de groupe globalement administrées. Ces adresses de DL sont formées en préfixant la chaîne de 10 bits "0000 0000 01" aux 22 bits inférieurs des identifiants propres à une organisation (Organizationally Unique Identifiers) spécifiés dans la famille de normes ISO/CEI 8802 relatives aux réseaux locaux (Local Area Network). Ces adresses de DL ne désignent implicitement aucune liaison spécifique.

Tableau 1 – Adressage liaison || nœud || sélecteur

Liaison	N | S	Usage assigné pour la plage spécifiée d'adresses de DL
0000	0000 … FFFF	Adresses de DL de liaison locale telles que spécifiées dans le Tableau 2
0001	0000 … 0007	Adresses de DL de groupe plates non locales, réservées pour une utilisation de DLL

Liaison	**N | S**	**Usage assigné pour la plage spécifiée d'adresses de DL**
0001	0008 … 003F	Adresses de DL de groupe plates non locales, réservées pour des AE normalisées
0001 … 003F	0040 … FFFF	Adresses de DL de groupe plates non locales disponibles pour l'affectation du vendeur, de l'utilisateur et l'assignation dynamique
0040 … 007F	0000 … FFFF	Adresses de DL de groupe plates globalement administrées non locales – potentiellement une par vendeur (Voir Note 1)
0080	0000 … 0007	Adresses de DLSAP individuelles plates non locales pour une utilisation de DLL
0080	0008 … 003F	Adresses de DLSAP individuelles plates non locales, réservées pour des AE normalisées
0080 … 00FF	0040 …	Adresses de DLSAP individuelles plates non locales disponibles pour l'affectation du vendeur, de l'utilisateur et l'assignation dynamique
0100 … 0FFF	0000 … FFBF	Adresses de DL individuelles plates non locales disponibles pour l'affectation du vendeur, de l'utilisateur et l'assignation dynamique comme adresses des DLCEP et adresses de DLSEP
0FFF	FFC0 … FFF7	Adresses de DL individuelles plates non locales réservées pour les adresses de DLCEP et les adresses de DLSEP des AE normalisées
0FFF	FFF8 … FFFF	Adresses de DL individuelles plates non locales réservées pour l'utilisation de DLL comme adresses de DLCEP et adresses de DLSEP
1000 … FEFF	0000 … FFFF	Liaisons individuelles avec des adresses de DL telles que spécifiées dans le Tableau 2

NOTE Ces adresses de DL de groupes sont fondées sur les identifiants propres à une organisation spécifiés dans la famille de normes ISO/CEI 8802 relatives au réseau local

0080 … 00FF	Adresses de DLSAP individuelles non locales. Ces valeurs d'appellation de liaison, concaténées aux valeurs d'appellation de nœud et valeurs de sélecteur associées – *liaison* || *nœud* || *sélecteur* – forment un espace adresse de 8 388 608 identifiants utilisables comme adresses de DLSAP individuelles. Les adresses de DLSAP qui en résultent ne désignent implicitement ni liaison spécifique ni nœud spécifique. Les adresses de DLSAP individuelles non locales doivent être allouées individuellement ou en blocs contigus. Lorsqu'elles sont considérées comme étant des nombres entiers non signés de 32 bits, les premières huit adresses (dont les valeurs sont les plus faibles) de ce type sont réservées à une utilisation comme adresses "DLSAP" par le protocole de DL; les 56 adresses suivantes de ce type sont réservées pour les adresses de DLSAP individuelles normalisées pour des AE normalisées. Toutes les autres adresses sont disponibles pour être utilisées comme adresses de DLSAP individuelles non normalisées.
0100 … 0FFF	Adresses de DL non-DLSAP individuelles non locales, utilisables comme adresses de DLCEP et adresses de DLSEP. Ces valeurs d'appellation de liaison, concaténées aux valeurs d'appellation de nœud et valeurs de sélecteur associées – *liaison* || *nœud* || *sélecteur* – forment un espace adresse de 251 658 240 identifiants utilisables comme adresses de DLCEP et de DLSEP individuelles. Les adresses de DL qui en résultent ne désignent implicitement ni liaison spécifique ni nœud spécifique. Les adresses de DL non-DLSAP individuelles non locales doivent être allouées individuellement ou en blocs contigus. Lorsqu'elles sont considérées comme étant des nombres entiers non signés de 32 bits, les dernières huit adresses (dont les valeurs sont les plus élevées) de ce type sont réservées à une utilisation par le protocole de DL; les dernières 56 adresses suivantes de ce type sont réservées aux adresses de DLCEP ou adresses de DLSEP individuelles normalisées pour les AE normalisées. Toutes les autres adresses sont disponibles pour être utilisées comme adresses de DLCEP et adresses de DLSEP individuelles non normalisées.
1000 … FEFF	Appellations de liaisons individuelles. Des appellations de liaison doivent être allouées, individuellement ou en blocs contigus, afin d'augmenter les valeurs d'adresses lorsqu'elles sont considérées comme étant des nombres entiers non signés de 16 bits. La valeur du paramètre de configuration de gestion de DL V(ML) doit être supérieure ou égale à la plus grande des valeurs d'appellation de liaison assignées aux liaisons de la liaison étendue. NOTE Ce paramètre est utilisé par les ponts pour aider à la fourniture d'actions de transmission par défaut pour chaque liaison.

NOTE 1 Les valeurs potentielles d'appellation de liaison FF00 à FFFF sont réservées pour la prise en charge de la mise en œuvre de références d'identifiants à des adresses DL, comme cela est noté dans la CEI 61158-3-1.

En résumé, chaque DLE capable de reconnaître des adresses de DL LONG (voir 5.2.2) reconnaît les valeurs d'appellation de liaison 0001 à 0FFF comme désignant les espaces d'adresses individuelles et de groupe étendues – *liaison* || *nœud* || *sélecteur* – qui sont communs à la liaison étendue.

Chaque DLE capable de reconnaître des adresses de DL LONG DL reconnaît la valeur d'appellation de liaison 0000 comme désignant la liaison nominale, qui est une valeur unique d'appellation de liaison d'une liaison locale ≥ 1 000. Les valeurs d'identifiant qui en résultent – *id de liaison nominale* || *nœud* || *sélecteur* – forment un espace adresse contenant un nombre pouvant atteindre 65 280 d'identifiants utilisables comme adresses de DL individuelles et de groupe telles que décrites en 4.3.2.2.

NOTE 2 Les composantes *liaison* || *nœud* || *sélecteur* de ces identifiants d'adresse ont la forme hexadécimale WWWWXXYY, où 1000 ≤ WWWW, 01 ≤ XX ≤ FF et 00 ≤ YY ≤ FF. Lorsque 01 ≤ XX ≤ 03, ces adresses sont des adresses de DL de groupe; toutes les autres adresses de ce type sont des adresses de DL individuelles. Les adresses dans lesquelles 1 000 ≤ WWWW et XX = 00 se réfèrent au nœud local et ne sont valides qu'au sein du logiciel de ce nœud.

Lorsque les besoins en espace adresse d'une liaison donnée excède l'espace fourni par un id de liaison nominale unique, des identifiants de liaison secondaires peuvent être alloués à la liaison en question. Pour chacun de ces identifiants de liaison secondaires, les valeurs qui en résultent – *identifiant de liaison secondaire* || *nœud* || *sélecteur* – forment un espace adresse complémentaire de 57 600 identifiants utilisables comme adresses de DL individuelles et de groupe supplémentaires pour la liaison telles que décrites en 4.3.2.2.

NOTE 3 Les composantes *liaison* || *nœud* || *sélecteur* de ces identifiants d'adresse supplémentaires ont la forme hexadécimale WWWWXXYY, où 1 000 ≤ WWWW, et soit 0140 ≤ XXYY ≤ 03FF, soit 0440 ≤ XXYY ≤ 0FBF, soit 10 ≤ XX ≤ FF et 08 ≤ YY ≤ F7. Lorsque 01 ≤ XX ≤ 03, ces adresses sont des adresses de DL de groupe; toutes les autres adresses de ce type sont des adresses de DL individuelles.

Les DLE qui ne sont pas capables de reconnaître des adresses de DL LONG reconnaissent seulement des adresses SHORT et VERY-SHORT (voir 5.2.2), qui désignent toujours la liaison nominale. Les DLE de ce type ne sont pas capables de communiquer directement ou d'être directement gérées par le biais d'un pont.

4.3.2.2 Appellations de nœud

La composante "appellation de nœud" d'une adresse de DL désigne habituellement une seule DLE. Lorsque la composante "appellation de liaison" n'a pas la valeur 0001 à 0FFF, les valeurs pour la composante "appellation de nœud" sont prédéfinies ou bien leur utilisation est contrainte, comme suit:

Tableau 2 – Adressage liaison locale nœud || sélecteur

n	S	Usage assigné pour la plage spécifiée d'adresses de DL
00	00 … FF	Adresses de DL de nœud locales telles que spécifiées dans le Tableau 4
01	00 … 07	Adresses de DL de groupe plates locales à une liaison réservées pour une utilisation de DLL
01	08 … 3F	Adresses de DL de groupe plates locales à une liaison, réservées pour des AE normalisées
01 … 03	40 … FF	Adresses de DL de groupe plates locales à une liaison disponibles pour l'affectation du vendeur, de l'utilisateur et l'assignation dynamique
04	00 … 07	Adresses de DLSAP individuelles plates locales à une liaison pour une utilisation de DLL
04	08 … 3F	Adresses de DLSAP individuelles plates locales à une liaison, réservées pour des AE normalisées
04	40 … FF	Adresses de DLSAP individuelles plates locales à une liaison disponibles pour l'affectation du vendeur, de l'utilisateur et l'assignation dynamique
05 … 0F	00 … BF	Adresses de DL individuelles plates locales à une liaison disponibles pour l'affectation du vendeur, de l'utilisateur et l'assignation dynamique comme adresses des DLCEP et adresses de DLSEP

n	S	Usage assigné pour la plage spécifiée d'adresses de DL
0F	C0 … F7	Adresses de DL individuelles plates locales à une liaison réservées pour les adresses de DLCEP et les adresses de DLSEP des AE normalisées
0F	F8 … FF	Adresses de DL individuelles plates locales à une liaison réservées pour l'utilisation de DLL comme adresses de DLCEP et adresses de DLSEP
10 … FF	00 … FF	Nœuds individuels avec des adresses de DL telles que spécifiées dans le Tableau 4

00 La DLE locale. L'équivalent d'une adresse interne de DLE. Cette appellation est fournie pour la commodité de l'unification des primitives de service et pour permettre à des mises en œuvre de se référer à l'adresse de DLE locale par une constante zéro (00). Lorsque la transmission est appropriée, une mise en œuvre doit remplacer cette adresse zéro (00) par le propre identifiant de nœud nominal de la DLE.

NOTE 1 Par conséquent, les DLPDU avec une composante "appellation de nœud" de zéro (00) dans une adresse de DL (autre qu'une adresse de DL plate) ne sont jamais émises et sont éliminées à réception.

Cette appellation de nœud ne doit pas être utilisée avec des identifiants de liaison secondaire.

01 … 03 Adresses de DL de groupe locales à une liaison. Ces valeurs d'appellation de nœud, concaténées à l'identifiant de liaison nominal ou secondaire de la liaison locale et les valeurs de sélecteur associées —*identifiant de liaison* || *nœud* || *sélecteur* – forment des espaces adresses de 768 identifiants chacun, utilisables comme adresses de DL de groupe.

Des adresses de DL de groupe locales à une liaison doivent être allouées, individuellement ou en blocs contigus, afin d'augmenter les valeurs d'adresses lorsqu'elles sont considérées comme étant des nombres entiers non signés de 16 bits. Les premières (dont les valeurs sont les plus faibles) des adresses de ce type sont réservées à une utilisation par le protocole de DL; les 56 adresses suivantes de ce type sont réservées pour des adresses de DL de groupe normalisées. Toutes les autres adresses sont disponibles pour être utilisées comme adresses de DL de groupe non normalisées.

04 Adresses de DLSAP individuelles locales à une liaison. Ces valeurs d'appellation de nœud, concaténées à l'identifiant de liaison nominal ou secondaire de la liaison locale et les valeurs de sélecteur associées – *identifiant de liaison* || *nœud* || *sélecteur* – forment des espaces adresses de 256 identifiants chacun, utilisables comme adresses de DLSAP individuelles. Les adresses de DLSAP qui en résultent ne désignent implicitement aucun nœud spécifique.

Des adresses de DLSAP individuelles locales à une liaison doivent être allouées, individuellement ou en blocs contigus, afin d'augmenter les valeurs d'adresses lorsqu'elles sont considérées comme étant des nombres entiers non signés de 16 bits. Les premières (dont les valeurs sont les plus faibles) des adresses de ce type sont réservées à une utilisation comme adresses de DLSAP par le protocole de DL; les 56 adresses suivantes de ce type sont réservées pour les adresses de DLSAP individuelles normalisées pour des AE normalisées.

05 … 0F Adresses de DL non-DLSAP individuelles locales à une liaison, utilisables comme adresses de DLCEP et adresses de DLSEP. Ces valeurs d'appellation de nœud, concaténées à l'identifiant de liaison nominal ou secondaire de la liaison locale et les valeurs de sélecteur associées – *identifiant de liaison* || *nœud* || *sélecteur* – forment

des espaces adresses de 2 816 identifiants chacun, utilisables comme adresses de DLCEP et de DLSEP individuelles. Les adresses de DL qui en résultent ne désignent implicitement aucun nœud spécifique.

Les adresses de DL non-DLSAP individuelles locales à une liaison, utilisables avec des services de DL connectés doivent être allouées, individuellement ou en blocs contigus, afin de diminuer les valeurs d'adresse lorsqu'elles sont considérées comme étant des nombres entiers non signés de 16 bits. Les premières (dont les valeurs sont les plus élevées) huit adresses de ce type sont réservées à une utilisation par le protocole de DL; les 56 adresses suivantes de ce type sont réservées aux adresses de DLCEP ou adresses de DLSEP individuelles normalisées pour les AE normalisées. Toutes les autres adresses sont disponibles pour être utilisées comme adresses de DLCEP et adresses de DLSEP individuelles non normalisées.

10 … FF Appellations de nœuds individuels Ces valeurs d'appellation de nœud, concaténées à l'identifiant de liaison nominal de la liaison locale, fournissent l'équivalent d'une "adresse de matériel unique" pour chaque DLE physique individuelle. Les appellations de nœud doivent être allouées, individuellement ou en blocs contigus, comme le montre le Tableau 3.

Tableau 3 – Appellations de nœud locales à une liaison

n	Usage assigné pour la plage spécifiée d'appellations de nœud
00	DLE locale
01 … 0F	inutilisable
10 … 13	DLE de la classe Bridge (pont)
14 …	DLE de la classe Link-master (maître de liaison)
15 … F6	Inutilisé
… F7	DLE de la classe Basic (de base)
F8 … FB	DLE non "invitées" attendant une affectation correcte des appellations de nœud
FC … FF	DLE "Visitor" (“Invitées”)

00 La DLE locale. L'équivalent d'une adresse interne de DLE. Cette appellation est fournie pour la commodité de l'unification des primitives de service et pour permettre à des mises en œuvre de se référer à l'adresse de DLE locale par une constante zéro (00). Lorsque la transmission est appropriée, une mise en œuvre doit remplacer cette adresse zéro (00) par le propre identifiant de nœud nominal de la DLE.

NOTE 2 Par conséquent, les DLPDU avec une composante "appellation de nœud" de zéro (00) dans une adresse de DL (autre qu'une adresse de DL plate) ne sont jamais émises et sont éliminées à réception.

01 … 0F Réservées à une utilisation sans rapport au sein du protocole de DL - non disponible pour les adresses de DL NODE (nœud) de DL.

10 … 13 Réservée pour l'allocation à des DLE qui pontent (interconnectent) la liaison locale (dont l'identifiant de liaison est l'identifiant de liaison nominal) à d'autres liaisons, en particulier à des liaisons de plus grande capacité

NOTE 3 De tels ponts sont censés servir de gestionnaires locaux à une liaison pour le protocole de DL.

14 … Réservée pour l'allocation à d'autres DLE qui sont capables d'initialiser et de gérer la liaison locale, et qui sont destinées à être attachées en permanence à la liaison locale (et non, par exemple, des DLE de maintenance telles que les "communicateurs tenus à la main”).

Ces appellations de nœud doivent être allouées, individuellement ou en blocs contigus, afin d'augmenter les valeurs d'adresses lorsqu'elles sont considérées comme étant des nombres entiers non signés de 8 bits.

15 … F6 Réservée pour l'allocation à des DLE qui exigent plus d'adresses de DL qu'il n'en est fourni hiérarchiquement par l'appellation de nœud primaire de la DLE.

… F7 Réservée pour l'allocation à des DLE qui ne sont pas capables (ou qui sont configurées pour être incapables) d'initialiser la liaison locale.

Ces appellations de nœud doivent être allouées, individuellement ou en blocs contigus, afin de diminuer les valeurs d'adresses lorsqu'elles sont considérées comme étant des nombres entiers non signés de 8 bits.

F8 … FB Réservée à une utilisation temporaire par d'autres DLE qui ne sont pas destinées à être attachées en permanence à la liaison locale alors qu'elles déterminent leur appellation appropriée de nœud final.

FC … FF Réservée pour l'allocation à des DLE qui ne sont pas destinées à être attachées en permanence à la liaison locale (par exemple, des DLE de maintenance telles que les "communicateurs tenus à la main”).

NOTE 4 Ces DLE “visitor” (invitées) sont normalement capables d'initialiser et gérer la liaison locale, mais sont les DLE de ce type les moins souhaitables à cet effet parce que leur présence à long terme sur la liaison n'est pas prévue.

Lorsqu'elles ne sont pas utilisées pour désigner une DLE unique, les valeurs d'appellation de nœud entre celles utilisées pour les DLE capables d'initialiser la liaison locale et celles incapables (ou configurées pour être incapables) d'initialiser la liaison locale peuvent être concaténées avec l'identifiant de liaison nominal ou secondaire de la liaison locale, et avec les composantes "sélecteur" dont la valeur est $\geq$ 08 et $\leq$ F7, pour former des espaces d'identifiants supplémentaires de 240 adresses de DL individuelles chacun – *identifiant de liaison*|| *nœud* || *sélecteur*. Chaque espace adresse supplémentaire de ce type peut être assigné à un seul nœud ou peut être partagé entre les nœuds de la liaison.

NOTE 5 Les composantes *nœud* || *sélecteur* de ces identifiants d'adresse supplémentaires ont la forme hexadécimale XXYY, où 10 < XX < FC et 08 ≤ YY ≤ F7.

En résumé, lorsqu'une valeur d'appellation de liaison désigne la liaison locale, chaque DLE reconnaît les valeurs d'appellation de nœud 01 à 0F comme désignant des espace adresses individuelles et de groupes locales à une liaison – *identifiant de liaison*|| *nœud* || *sélecteur* – communs à tous les nœuds sur la liaison locale.

Chaque DLE reconnaît la valeur d'appellation de nœud 00 comme désignant une valeur d'appellation d'un seul nœud local ≥ 10 (hexadécimale), le nœud nominal. Les valeurs d'identifiant qui en résultent – *id de liaison nominale* || *id de nœud nominal* || *sélecteur* – forment un espace d'identifiants utilisables comme adresses de DL individuelles telles que décrites en 4.3.2.3.

Lorsque les besoins en espace adresse d'un nœud donné excède l'espace fourni par un id de nœud nominal unique, des identifiants de nœud secondaires peuvent être alloués au nœud en question. Pour chacun de ces identifiants de nœud secondaires, les valeurs qui en résultent – *identifiant de liaison* || *id de nœud secondaire* || *sélecteur* – avec des valeurs de composante "sélecteur" ≥08 et ≤F7, forment un espace complémentaire de 240 identifiants utilisables comme adresses de DL individuelles supplémentaires pour le nœud telles que décrites en 4.3.2.3. La même méthode peut être utilisée pour étendre le jeu d'adresses individuelles non spécifiques à un nœud pour la liaison.

4.3.2.3 Sélecteurs

La composante "sélecteur" d'une adresse DL désigne habituellement une seule entité de protocole au sein de la DLE.

Lorsque la composante "appellation de liaison" n'a pas la valeur 0001 – 0FFF et lorsque la composante "appellation de nœud" n'a pas la valeur 01 – 0F, les valeurs de la composante "sélecteur" sont prédéfinies et leur utilisation est contrainte, comme le spécifie dans le Tableau 4.

Tableau 4 – Adressage de sélecteur local à un nœud

Sélecteur	Usage assigné pour la plage spécifiée d'adresses de DL
00 … 01	Adresses de DLSAP individuelles locales à un nœud pour une utilisation de DLL
02 … 07	Adresses de DLSAP individuelles locales à un nœud, réservées pour des AE normalisées
08 … 1F	Adresses de DLSAP individuelles locales à un nœud disponibles pour l'affectation du vendeur, de l'utilisateur et l'assignation dynamique
20 … F7	Adresses de DL individuelles locales à un nœud disponibles pour l'affectation du vendeur, de l'utilisateur et l'assignation dynamique comme adresses des DLCEP et adresses de DLSEP
F8 … FF	Adresses de DL individuelles locales à un nœud réservées pour une utilisation comme adresses de DLCEP et adresses de DLSEP des AE normalisées

00 … 07 Réservée pour les adresses de DLSAP normalisées relatives à un nœud, et pour les adresses de DLSAP au sein d'une DLE.

08 … 1F Adresses de DLSAP individuelles non normalisées.

20 … F7	Adresses de DLCEP et adresses de DLSEP non normalisées.
F8 … FF	Réservée pour les adresses de DLCEP et adresses de DLSEP non normalisées relatives à un nœud

4.3.3 Adresses de DL prédéfinies

Comme indiqué du Tableau 1 au Tableau 3, certaines adresses de DL spécifiques sont définies en 4.3.3. Ces adresses de DL sont définies pour faciliter la fourniture des services de DL et l'initialisation des communications OSI au sein d'un système réel physiquement distribué.

4.3.3.1 Adresses de DL non locales, plates et prédéfinies

Plusieurs adresses de DL non locales et plates sont définies dans le présent paragraphe, telles que spécifiées dans le Tableau 5.

NOTE SLAE est l'acronyme de System Load Application Entity (entité d'application de chargement de système); SMAE est l'acronyme de System Management Application Entity (entité d'application de gestion système).

Tableau 5 – Adresses de DL non locales, plates et prédéfinies

Liaison || N || S		**Usage assigné pour l'adresse de DL spécifiée**
0001	0000	Les fonctions de prise en charge de DL de "toutes" (voir NOTE) les DLE sur la liaison étendue
0001	0001	Les fonctions de prise en charge de DL de "toutes" (voir NOTE) les DLE de LM sur la liaison étendue
0001	0002	Les fonctions de prise en charge de DL de "toutes" (voir NOTE) les DLE ponts sur la liaison étendue
0001	0003	Les fonctions de pontage de DL de "toutes" (voir NOTE) les DLE ponts sur la liaison étendue
0001	0008	Les SMAE de "toutes" (voir NOTE) les DLE non configurées sur la liaison étendue
0001	0009	Les SMAE de "toutes" (voir NOTE) les DLE sur la liaison étendue
0001	000A	Les SMAE de "toutes" (voir NOTE) les DLE de LM sur la liaison étendue
0001	000B	Les SMAE de "toutes" (voir NOTE) les DLE ponts sur la liaison étendue
0001	000C	Les SLAE de "tous" (voir NOTE) les LoadServers (serveurs de chargement) sur la liaison étendue
0001	000D	Les SLAE de "tous" (voir NOTE) les LoadableDevices (appareils chargeables) sur la liaison étendue
0080	0000	Réservée à l'utilisation de la DLL pour une adresse de "DLSAP" de prise en charge de DL (Voir Note 1)
0080	0004	L'adresse "DLSAP" pour les fonctions de pont de DL de la DLE pont sur la liaison étendue qui est le pont racine de l'arborescence (spanning tree)
0FFF	FFFE	L'adresse de DLCEP pour la DLC à partir des fonctions de pont de la DLE pont sur la liaison étendue qui est le pont racine de l'arborescence jusqu'aux fonctions de pont dans toutes les autres DLE ponts sur la liaison étendue
0FFF	FFFF	Réservée à l'utilisation de la DLL pour une adresse de DLCEP ou adresse de DLSEP de prise en charge de DL (Voir NOTE)
NOTE Les DLE qui ne reconnaissent pas les adresses DL LONG sont nécessairement exclues de ces jeux.		

4.3.3.2 Adresses de DL locales, plates et prédéfinies

Plusieurs adresses de DL locales plates sont définies dans le présent paragraphe, telles que spécifiées dans le Tableau 6. Elles correspondent de façon univoque (une pour une) aux adresses DL plates, non locales et prédéfinies qui sont spécifiées dans le Tableau 5. L'étendue des adresses dans le Tableau 6 est la liaison locale; l'étendue des adresses dans le Tableau 5 est la liaison locale entière;

Les adresses de DL dans le Tableau 6 ont une correspondance univoque (une à une) avec les adresses de DL analogues dans le Tableau 5; alors que les adresses dans le Tableau 6 se réfèrent à des fonctions qui sont locales à chaque liaison, les adresses analogues du Tableau 5 se réfèrent à des fonctions similaires qui sont globales à la liaison étendue entière. Cette correspondance d'adressage est intentionnelle et peut servir à diminuer la complexité de mise en œuvre. Par conséquent, il convient de préserver cette correspondance dans les futures assignations d'adresses.

Tableau 6 – Adresses de DL, locales à une liaison, plates et prédéfinies

Nœud || sélecteur		Usage assigné pour l'adresse de DL spécifiée
01	00	Les fonctions de prise en charge de DL de toutes les DLE sur la liaison
01	01	Les fonctions de prise en charge de DL de toutes les DLE de LM sur la liaison
01	02	Les fonctions de prise en charge de DL de toutes les DLE ponts sur la liaison
01	03	Les fonctions de ponts de DL de toutes les DLE ponts sur la liaison
01	08	Les SMAE de toutes les DLE non configurées sur la liaison
01	09	Les SMAE de toutes les DLE sur la liaison
01	0A	Les SMAE de toutes les DLE de LM sur la liaison
01	0B	Les SMAE de toutes les DLE ponts sur la liaison
01	0C	Les SLAE de tous les LoadServers (serveurs de chargement) sur la liaison
01	0D	Les SLAE de tous les LoadableDevices (appareils chargeables) sur la liaison
04	00	L'adresse de "DLSAP" pour les fonctions de prise en charge de DL de la DLE sur la liaison qui sert comme LAS
04	04	L'adresse "DLSAP" pour les fonctions de pont de DL de la DLE pont sur la liaison qui est dominante (la plus proche de la racine) dans l'arborescence du pont
0F	FE	L'adresse de DLCEP pour la DLC à partir des fonctions de pont de la DLE pont sur la liaison étendue qui est dominante (la plus proche de la racine) dans l'arborescence du pont jusqu'aux fonctions de pont dans toutes les autres DLE ponts sur la liaison
0F	FF	L'adresse de DLCEP pour la DLC à partir des fonctions de prise en charge de DL de la DLE sur la liaison qui sert comme LAS jusqu'aux fonctions de prise en charge de DL de toutes les autres DLE de LM sur la liaison

4.3.3.3 Adresses de DL locales à un nœud prédéfinies

Plusieurs adresses de DL locales à un nœud sont définies dans le présent paragraphe, telles que spécifiées dans le Tableau 7.

Tableau 7 – Adresses de DL locales à un nœud prédéfinies

Sélecteur	Usage assigné pour l'adresse de DL spécifiée
00	L'adresse de "DLSAP" pour les fonctions de prise en charge de DL pour la DLE du nœud
01	L'adresse de "DLSAP" pour les fonctions de pont de DL pour la DLE du nœud
02	L'adresse de "DLSAP" pour la même du nœud
03	L'adresse de "DLSAP" pour la SLAE du nœud

4.3.4 Représentations des adresses de DL sous forme d'adresses MAC de 48 bits administrées localement

Les protocoles de réseau local ISO/CEI (ISO/CEI 8802) fournissent une base pour les parties de protocoles de bus de terrain CEI, tels que le diagramme d'états de redondance de couche physique, le protocole (de pontage) inter relais de DL et le protocole de chargement de système pour la gestion système. Les deux derniers protocoles, qui spécifient des codages de six octets pour chaque adresse MAC de l'ISO/CEI, sont en cours d'adaptation pour une utilisation dans les bus de terrain avec le minimum de changements nécessaires; par conséquent, il existe un besoin de spécifier la représentation des adresses de DL de quatre octets du bus de terrain en vue de l'utilisation avec les adaptations de ces protocoles dans les bus de terrain.

Une adresse MAC de six octets de l'ISO/CEI, telle que spécifiée dans l'ISO/CEI TR 8802-1, contient, dans l'ordre de transmission,

a) un bit initial, I/G, indiquant si l'adresse MAC est une adresse de SAP MAC individuelle (I/G=0) désignant une seule entité de sous-couche MAC, ou une adresse MAC de groupe (I/G=1) désignant un groupe d'un nombre supérieur ou égal à zéro d'entités de sous-couche MAC;
b) un second bit, U/L, indiquant si l'adresse MAC contient une composante administrée à un niveau global (U/L=0) ou si elle est complètement administrée localement (U/L=1);
c) une composante de 22 bits administrée à un niveau global et une composante de 24 bits administrée par un vendeur lorsque U/L = 0 ou une composante de 46 bits administrée par un utilisateur lorsque U/L=1 (voir Figure 5).

Adresses MAC administrées à un niveau global			
I/G	U/L	Administrée à un niveau global	Administrée par un vendeur
1 bit	1 bit	22 bits	24 bits

Adresses MAC administrées localement		
I/G	U/L	Administrée par un utilisateur
1 bit	1 bit	46 bits

Figure 5 – Structure de base des adresses MAC

Sachant que les adresses de DL ne sont pas administrées pour être uniques à un niveau global, elles doivent être représentées sous forme d'adresses MAC de six octets administrées localement telles que montrées à la Figure 6.

Adresse de DL représentée comme une adresse MAC administrée localement			
I/G	U/L	Établie par la gestion de DL	Adresse de DL étendue
1 B	1 B	14 bits	32 bits

Figure 6 – Représentation d'une adresse de DL sous forme d'adresse MAC

La composante 14 bits de l'adresse MAC établie par la gestion DL doit être mise à la valeur du paramètre *DL-MAC-address-embedding-prefix* (préfixe d'imbrication d'adresse MAC-DL) de la liaison étendue, V(MEP), qui doit être distribuée comme une partie intégrante des informations de configuration relatives à la liaison étendue. La valeur par défaut de ce paramètre doit être la valeur l'hexadécimale 3FF (des uns partout), mais doit être modifiable par la gestion de DL pour permettre à plus d'une liaison étendue d'être directement adressable depuis les réseaux locaux ISO/CEI en interfonctionnement.

Chaque adresse de DLSAP, de DLCEP et de DLSEP de 32 bits doit être représentée sous la forme d'une adresse MAC de six octets en préfixant l'adresse de DL autre que de groupe de 32 bits par la chaîne de caractères binaires "01" suivie du DL-MAC-address-embedding-prefix (préfixe d'imbrication d'adresse MAC-DL), V(MEP) de 14 bits de la liaison étendue.

Chaque adresse de DL de groupe de 32 bits doit être représentée sous la forme d'une adresse MAC de six octets en préfixant l'adresse de DL de groupe de 32 bits par la chaîne de caractères binaires "11" suivie du préfixe d'imbrication d'adresse MAC-DL, V(MEP), de 14 bits de la liaison étendue.

De plus, s'il existe le besoin de spécifier une adresse MAC de six octets pour le groupe des DLCEP abonnés à plusieurs homologues associés à un DLCEP éditeur à plusieurs homologues spécifié, l'adresse implicite de ce groupe de DLCEP abonnés doit être représentée sous forme d'une adresse MAC de six octets en préfixant l'adresse de DLCEP individuelle de 32 bits de l'éditeur par la chaîne de caractères binaires "11" suivie du préfixe d'imbrication d'adresse MAC DL de 14 bits de la liaison étendue, V(MEP).

4.4 Service assumé à partir de la PhL

Le présent paragraphe définit les primitives de Service Physique (PhS) assumées et leurs contraintes d'utilisation par la DLE.

Une stratification correcte exige qu'une entité de couche (N+1) ne soit pas concernée par, et qu'une interface de service (N) ne contraigne pas exagérément, les moyens par lesquels une couche (N) fournit ses services (N). Donc, l'interface de service Ph n'exige pas que les DLE connaissent les détails internes de la PhE (par exemple: le préambule, le synchroniseur final et les séquences de signaux de délimiteur de DLPDU, le nombre de bits par baud), et il convient qu'elles n'empêchent pas la PhE d'utiliser des technologies évolutives appropriées.

4.4.1 Primitives assumées du PhS

La granularité de la transmission dans le protocole de bus de terrain est d'un octet. Il s'agit de la granularité des données utilisateur de PhS et des informations de temporisation échangées à l'interface PhL - DLL.

4.4.1.1 Service de rapports relatifs aux caractéristiques du PhS

Le PhS est supposé fournir la primitive de service suivante pour rapporter les caractéristiques de PhS essentielles utilisées dans les activités de transmission, de réception et de programmation de la DLL:

> "Indication" de Ph-CHARACTERISTICS (débit minimum de données, surdébit de verrouillage de trame)

où

> **débit minimum de données**— spécifie la vitesse minimale réelle d'acheminement des données en bits par seconde (y compris toutes les éventuelles tolérances relatives aux temps) de toute PhE sur la liaison locale.
>
> NOTE 1 Une PhE avec un débit nominal de données de 1 Mbit/s ± 0,01 % spécifie un débit minimal de données de 0,999 9 Mbit/s.
>
> **surdébit de verrouillage de trame** – spécifie le nombre maximum de périodes de bits, où la période 1/(débit de données)), utilisée pour n'importe quelle transmission pour des PhPDU qui n'acheminent pas directement des données (par exemple, les PhPDU acheminant un préambule, des délimiteurs de DLPDU, un synchroniseur final, un «silence» inter-DLPDU, et ainsi de suite).

NOTE 2 Si le surdébit de verrouillage de trame est F et les deux longueurs de message de DL sont L_1 et L_2, le temps pour envoyer deux messages immédiatement consécutifs de longueurs L_1 et L_2 sera au moins aussi long que le temps nécessaire pour envoyer un message de longueur $L_1 + F + L_2$.

Si le surdébit de verrouillage de trame est supérieur au surdébit de PhL par DLPDU ("per-DLPDU-PhL-overhead"), V(PhLO), configuré de la DLE, la DLE doit rapporter cette discordance à la gestion de DL et ne doit pas émettre de demandes de "Ph-DATA " tant que cette discordance existe.

NOTE 3 Cette restriction empêche la transmission de la DLE pendant que cette discordance existe. La station de gestion locale de la DLE peut remédier à cette discordance en reconfigurant le V(PhLO) à une valeur plus élevée.

4.4.1.2 Services de transmission et de réception du PhS

Le PhS est supposé fournir les primitives de service suivantes pour la transmission et la réception;

"request" de Ph-DATA (class, data);

"indication" de Ph-DATA (class, data);

" confirm" de Ph-DATA (status)

où

class – spécifie la composante Ph-interface-control-information (PhICI) de la Ph-interface-data-unit (PhIDU). Pour une "request" (demande) de Ph-DATA, ses valeurs possibles sont

START-OF-ACTIVITY – il convient que la transmission des PhPDU qui précèdent les données commence;

DATA – il convient que la valeur à un seul octet du paramètre data associé soit émise comme partie intégrante de la transmission continue correctement formée;

END-OF-DATA-AND-ACTIVITY – il convient que les PhPDU qui terminent les données d'utilisateur de Ph soient émises après le dernier octet précédent des données d'utilisateur de Ph, en culminant dans la cessation de la transmission active;

Pour une "indication" de Ph-DATA, ses valeurs possibles sont

START-OF-ACTIVITY – la réception d'une transmission évidente à partir d'une ou plusieurs PhE a commencé;

DATA – le paramètre data associé a été reçu comme partie intégrante d'une réception continue correctement formée;

END-OF-DATA – la réception continue correctement formée en cours des données d'utilisateur de Ph s'est terminée avec la réception correcte des PhPDU impliquant des END-OF-DATA;

END-OF-ACTIVITY – la réception en cours (d'une transmission apparente à partir d'une ou plusieurs PhE) s'est terminée, sans autre preuve de transmission de PhE;

END-OF-DATA-AND-ACTIVITY – apparition simultanée de END-OF-DATA et de END-OF-ACTIVITY;

data – spécifie la composante Ph-interface-data (PhID) de la PhIDU. Elle est constituée d'un octet de données d'utilisateur de Ph devant être émises ("request" de Ph-DATA) ou qui ont été reçues avec succès ("indication" de Ph-DATA).

status – spécifie soit le succès, soit la cause détectée localement pour inférer l'échec.

La primitive "confirm" de Ph-DATA fournit le critique retour d'informations relatives à la temporisation physique nécessaire pour empêcher que la DLE démarre une seconde transmission avant que la première ne soit achevée. La primitive finale "confirm " du service Ph-DATA d'une transmission ne doit pas être émise tant que la PhE n'a pas parachevé la transmission courante.

4.4.2 Notification des caractéristiques du PhS

Le PhE a la responsabilité de notifier à la DLE les caractéristiques du PhS qui sont pertinentes au fonctionnement de la DLE. Cette notification est accomplie par la PhE en émettant une seule primitive "indication" Ph-CHARACTERISTICS en chacun des PhSAP de la PhE au démarrage de la PhE.

4.4.3 Transmission de données utilisateur de Ph

La PhE détermine la temporisation de toutes les transmissions. Lorsqu'une DLE a une DLPDU à émettre et le protocole de DL fournit à cette DLE le droit d'émettre, la DLE doit envoyer la DLPDU, y compris une FCS concaténée, en formant une séquence bien formée de demandes de Ph-DATA, consistant en une seule demande spécifiant START-OF-ACTIVITY; suivie de trois à 300 (maximum) demandes consécutives, spécifiant DATA; et terminée par une seule demande spécifiant END-OF-DATA-AND-ACTIVITY.

Le PhE signale son achèvement de chaque "request" de Ph-DATA et sa disposition à accepter une nouvelle "request" de Ph-DATA, avec une primitive "confirm" de Ph-DATA; le paramètre "status" de la primitive "confirm" de Ph-DATA achemine le succès ou l'échec de la "request" Ph-DATA associée. Il convient de ne pas émettre une seconde "request" de Ph-DATA tant que la "confirm" de Ph-DATA correspondant à la première "request" n'a pas été reçue en provenance de la PhE.

4.4.4 Réception de données utilisateur de Ph

La PhE rapporte une transmission reçue avec une séquence bien formée de primitives "indication" de Ph-DATA qui doit être constituée soit

a) d'une seule indication spécifiant START-OF-ACTIVITY; suivie d'indications consécutives spécifiant DATA; suivies d'une seule indication spécifiant END-OF-DATA; et terminée par une indication unique spécifiant END-OF-ACTIVITY, soit
b) d'une seule indication spécifiant START-OF-ACTIVITY; suivie d'indications consécutives spécifiant DATA; suivies d'une indication unique spécifiant END-OF-DATA-AND-ACTIVITY, soit
c) d'une seule indication spécifiant START-OF-ACTIVITY; suivie facultativement par une ou plusieurs indications consécutives spécifiant DATA; et terminée par une indication unique spécifiant END-OF-ACTIVITY.

NOTE Cette dernière séquence est indicative d'une réception incomplète ou incorrecte. La détection d'une erreur dans la séquence des PhPDU reçues, ou dans le processus de réception de la PhE, désactive les indications Ph-DATA ultérieures avec un paramètre de classe spécifiant DATA, END-OF-DATA, ou END-OF-DATA-AND-ACTIVITY jusqu'à ce que la fin de la période d'activité courante et le début de la période ultérieure aient été rapportés par des indications DATA spécifiant respectivement END-OF-ACTIVITY et START-OF-ACTIVITY.

Dans les deux premiers cas, la DLE concatène les données reçues et tente ensuite de les analyser en une DLPDU suivie d'une FCS concaténée. Dans le dernier cas, la DLE rejette toutes les données rapportées et rapporte l'événement à la gestion DL.

4.5 Fonctions de la DLL

4.5.1 Vue d'ensemble des fonctions

Les fonctions de la DLL sont les fonctions nécessaires pour ponter le trou entre les services disponibles auprès de la PhL et celles offertes aux utilisateurs de DLS. Lorsqu'elles sont utilisées dans un environnement OSI normal, les fonctions nécessaires de la DLL sont celles qui sont spécifiées dans l'ISO/CEI 8886. Lorsqu'elles sont utilisées dans un environnement OSI prioritaire, les fonctions nécessaires de la DLL sont un surensemble de celles qui sont spécifiées dans l'ISO/CEI 8886. les améliorations sont principalement apportées à

a) la disponibilité d'une primitive "confirm" et d'une dernière limite de confirmation de service pour chaque service de DL de transfert de données orienté connexion et sans connexion;

b) l'ajout de services acquittement de DLE destinataire sans connexion et de services d'échange de données bidirectionnel au service de DL de transfert de données OSI de base;

c) l'aptitude à différer et programmer les services de DL de transfert de données;

d) la distribution efficace de données utilisateur de DLS, d'un utilisateur DLS éditeur vers un ensemble d'utilisateurs DLS abonnés;

e) la convergence efficace de données utilisateur de DLS, d'un ensemble d'utilisateurs DLS abonnés vers un seul utilisateur DLS éditeur;

f) la fourniture d'un sens synchronisé de temps interne parmi les DLE et disponible pour les utilisateurs DLS de la liaison étendue;

g) la disponibilité normalisée de capacités de gestion d'adresses DL(SAP) locales, de tampon et de file d'attente.

4.5.2 Fonctions de transfert de données sans connexion

Les fonctions de transfert de données sans connexion ont pour but de transporter des DLSDU de taille limitée entre un utilisateur de DLS et un ou plusieurs autres utilisateurs de DLS sur la même liaison, sans l'exigence d'établir ou de maintenir une DLC avec chacun de ces autres utilisateurs DLS. Ce but est atteint au moyen d'une transmission de DLPDU fournissant le transfert d'une quantité limitée de données utilisateur vers un ou plusieurs utilisateurs DLS, avec une protection limitée contre la perte de DLSDU, ou la perte d'ordre des DLPDU émises successivement. Les services tant de transfert de DLSDU que d'échange de DLSDU sont fournis.

4.5.3 Fonctions orientées connexion

Les fonctions orientées connexion permettent la prise en charge de l'établissement, de l'utilisation, de la resynchronisation et de la cessation abrupte d'une connexion entre les utilisateurs DLS sur une liaison étendue. Le type de la connexion peut être sélectionné pour prendre en charge le flux de données utilisateur, soit

a) de façon bidirectionnelle, entre deux utilisateurs de DLS homologues,

b) de façon unidirectionnelle, d'un utilisateur de DLS homologue vers un autre,

c) de façon bidirectionnelle, entre un utilisateur de DLS éditeur et l'ensemble de zéro à plusieurs utilisateurs de DLS abonnés,

c) de façon unidirectionnelle, d'un un utilisateur de DLS éditeur vers l'ensemble de zéro à plusieurs utilisateurs de DLS abonnés, ou

e) de façon unidirectionnelle, de l'ensemble de zéro à plusieurs utilisateurs de DLS vers l'utilisateur de DLS éditeur.

Les caractéristiques de la connexion peuvent être sélectionnées pour prendre en charge le transfert des DLSDU de la taille maximale négociée, avec soit

1) une livraison fiable en séquence et non dupliquée avec réinitialisation à la suite d'une perte de DLSDU;

2) une livraison à retard réduit, potentiellement hors séquence mais non dupliquée, et réinitialisation à la suite d'une la perte de DLSDU;
3) une livraison à retard minimal, en séquence et non dupliquée, mais avec une perte potentielle de DLSDU;
4) une livraison à retard minimal, non séquencée, duplication potentielle, perte d'ordre potentielle, mais improbable et sans notification de perte de DLSDU; soit
5) absence de transfert dans un sens spécifié.

Seuls les cas 4) et 5) sont permis pour le sens abonné vers éditeur d'une DLC à plusieurs homologues

4.5.3.1 Phase d'établissement de connexion

Le but de la phase d'établissement de connexion est

a) d'établir une DLC entre deux utilisateurs de DLS,

   NOTE L'établissement d'une DLC d'éditeur est le mieux modélisé sous la forme de l'établissement indépendant simultané par paires de la DLC entre l'éditeur commun et chaque abonné séparé.

b) de déterminer les attributs QoS de la DLC;
c) de distinguer entre les DLC.

4.5.3.2 Phase de transfert de données

Le but de la phase de transfert de données est

a) de transporter des DLSDU entre deux utilisateurs de DLS reliés par une DLC. Ce but est obtenu par la transmission de DATA (DT) et de DLPDU connexes (voir 6.7, 6.6 et 6.4), qui peut comprendre la segmentation des DLSDU l'acheminement de plusieurs DLPDU et le réassemblage par la DLE de destination;
b) de resynchroniser le flux des DLSDU entre les utilisateurs DLS et 'notifier à ces utilisateurs DLS la perte d'information après une erreur non récupérée.

Ces concepts sont définis dans la CEI 61158-3-1.

4.5.3.3 Phase de terminaison de connexion

Le but de la phase de terminaison de connexion est de terminer abruptement une connexion entre deux ou plusieurs utilisateurs DLS et acheminer la cause de la terminaison.

4.5.4 Fonction de synchronisation du temps

Le but de la fonction de synchronisation du temps est de fournir une référence partagée de temps interne approximativement synchronisée pour tous les utilisateurs DLS. Cette fonction comporte deux composantes:

a) une composante monotone croissante en fonction du temps, avec une valeur de zéro au moment de l'établissement du système final local;
b) une seconde composant qui, lorsqu'elle est ajoutée à la première, rend la somme approximativement égale aux sommes correspondantes des DLE fonctionnant correctement sur la liaison étendue.

Tableau 8 – Corrélation des DLPDU avec classes fonctionnelles

Classes fonctionnelles						Code	Nom de DLPDU	Paragraphe
Basic		Link master		Bridge				
R[a]	S[a]	R[a]	S[a]	R[a]	S[a]	EC	ESTABLISH CONNECTION	6.1
R[a]	S[a]	R[a]	S[a]	R[a]	S[a]	DC	DISCONNECT CONNECTION	6.2
R[a]	S[a]	R[a]	S[a]	R[a]	S[a]	RC	RESET CONNECTION	6.3
R	S	R	S	R	S	CA	COMPEL ACKNOWLEDGMENT	6.4
R	S	R	S	R	S	CD	COMPEL DATA	6.5
R	S	R	S	R	S	ED	EXCHANGE DATA	6.6
R	S	R	S	R	S	DT	DATA	6.7
R	—	R	S	R	S	SR	STATUS RESPONSE	6.8
–	s[b]	R	s[b]	R	s[b]	CT	COMPEL TIME	6.9
r[b]	—	R[b]	S	R[b]	S	TD	TIME DISTRIBUTION	6.10
R	s[b]	R	s[b]	R	s[b]	RQ	ROUND-TRIP-DELAY QUERY	6.11
r[b]	S	r[b]	S	r[b]	S	RR	ROUND-TRIP-DELAY REPLY	6.12
R	—	R	S	R	S	PN	PROBE NODE DL-ADDRESS	6.13
–	S	R[c]	S	R[c]	S	PR	PROBE RESPONSE	6.14
R	—	R	S	R	S	PT	PASS TOKEN	6.15
R	—	R	S	R	S	ES	EXECUTE SEQUENCE	6.16
–	S	R[c]	S	R[c]	S	RT	RETURN TOKEN	6.17
–	S	R[c]	S	R[c]	S	RI	REQUEST INTERVAL	6.18
–	—	R	S	R	S	CL	CLAIM LAS	6.19
–	—	R	S	R	S	TL	TRANSFER LAS	6.20
r	—	r	S	–	S	WK	WAKEUP	6.21
–	s[d]	–	s[d]	–	s[d]	IDLE	IDLE	6.22
–	—	–	—	–	—	spare	SPARE	6.23

où

- r La classe spécifiée permet l'aptitude à recevoir et agir sur le type spécifié de DLPDU.
- R La classe spécifiée fournit toujours l'aptitude à recevoir et agir sur le type spécifié de DLPDU.
- s La classe spécifiée permet l'aptitude à envoyer le type spécifié de DLPDU.
- S La classe spécifiée fournit toujours l'aptitude à envoyer le type spécifié de DLPDU.
- – La classe spécifiée ne permet pas le type spécifié de DLPDU.

[a] La prise en charge mandatée exige seulement des éléments minimaux de procédure.

[b] Tous les éléments doivent être pris en charge lorsque la DLE a une classe de synchronisme temporel (voir 10.6.3) autre que NONE.

[c] Doivent répondre au contenu de la DLSDU lorsqu'elles agissent comme LAS.

[d] Aucune classe n'est requise pour recevoir la DLPDU IDLE. Son destinataire prévu est un moniteur ou analyseur de liaison spécialisé, ne relevant du domaine d'application de la présente norme.

4.6 Classes fonctionnelles

Dans la présente norme, une classe fonctionnelle de DLE détermine ses capacités pour l'activité DLL autonome et donc la complexité minimum des mises en œuvre conformes. Chaque classe comprend toutes les classes ayant un numéro plus faible. Les trois classes fonctionnelles, dans l'ordre de complexité croissante, sont

a) Basic (de base);

b) Link Master (Maître de liaison (LM));

c) Bridge (Pont).

Toutes les classes fonctionnelles prennent en charge tous les services d'utilisateur de DLS et sont entièrement interopérables. Les DLPDU associées aux éléments de la procédure pour chaque classe fonctionnelle sont spécifiées dans le Tableau 8.

4.6.1 Classe Basic

La classe Basic (de base) inclut les éléments protocolaires élémentaires de procédure nécessaires

a) pour fournir une interopérabilité lorsqu'elle répond à des DLPDU envoyées par un homologue de DLS ou une DLE pont;
b) pour initier, réinitialiser et terminer les DLC avec une DLE homologue, pour prendre en charge l'acheminement ordonné des DLSDU;
c) pour envoyer et recevoir des DLSDU sans connexion et orientées connexion, et répondre aux DLSDU reçues, le cas échéant;
d) pour demander des services issus du LAS;
e) pour exécuter une séquence ininterrompue d'opérations de liaison;
f) pour optimiser l'utilisation locale de la liaison.

Cette classe est le minimum nécessaire pour l'interopérabilité de bus de terrain.

4.6.2 Classe Link Master

La classe "maître de liaison" comprend toutes les fonctions de la classe de base. Elle comprend également les éléments protocolaires de procédure nécessaires

a) pour coopérer avec des DLE similaires en établissant et partageant l'autorité sur la liaison;
b) pour détecter l'absence de LAS sur la liaison et activer les fonctions de LAS dans son propre nœud,

et lorsqu'elle fournit les fonctions du LAS,

c) pour maintenir un accès ordonné à la ressource de communication de liaison partagée, répondant aux requêtes d'autres DLE pour l'utilisation de cette ressource partagée;
d) pour servir de source de temps interne pour les autres DLE sur la liaison.

Cette classe est nécessaire pour le fonctionnement autonome sur la liaison. Au moins une DLE sur la liaison doit fonctionner dans cette classe

4.6.3 Classe Bridge (relais de DL)

La classe "Bridge" comprend toutes les fonctions de la classe Link-Master. Elle comprend également les éléments protocolaires de procédure nécessaires

a) pour permettre les communications entre les DLE sur une même liaison qui sont elles-mêmes périodiquement incapables de communiquer directement sur la liaison (c'est-à-dire, des DLE de facteur d'utilisation fractionnaire (FDC)), et dans certains cas servant de substitut pour une DLE sur la liaison locale;
b) pour interconnecter deux ou plusieurs liaisons locales, en les pontant en une liaison étendue, et dans certains cas, en servant de substitut pour une DLE distante sur une des liaisons locales du pont;
c) pour fournir un sens commun du temps interne DL coordonné à travers la liaison étendue;.

Cette classe est nécessaire lors de l'interconnexion de deux ou plusieurs liaisons locales pour former une liaison étendue à plusieurs liaisons, ou lorsqu'une ou plusieurs DLE sur la liaison locale sont des DLE de facteur d'utilisation fractionnaire (FDC). Lorsqu'une liaison étendue de multiliaison existe, les liaisons locales individuelles doivent être interconnectées seulement par les DLE qui fonctionnent dans cette classe.

4.7 Paramètres, variables, compteurs, temporisateurs et files d'attente locaux

La présente norme utilise des paramètres de demande d'utilisateur de DLS P(...) et des variables locales V(...) comme un moyen de clarifier l'effet de certaines actions et les conditions dans lesquelles ces actions sont valides, des temporisateurs locaux T(...) comme un moyen de surveiller les actions du fournisseur de DLS distribué et d'assurer une réponse de DLE locale à l'absence de ces actions, et des temporisateurs locaux C(...) pour l'accomplissement des fonctions de mesure de débit. Elle utilise également les files d'attente locales Q(...) comme un moyen d'ordonnancer certaines activités, de clarifier les effets de certaines actions, et de clarifier les conditions dans lesquelles ces activités sont valides.

Sauf spécification contraire, au moment de leur création ou de l'activation de la DLE,

– toutes les variables doivent être initialisées à leur valeur par défaut, ou à leur valeur minimum permise si aucune valeur par défaut n'est spécifiée;
– tous les compteurs doivent être initialisés à zéro;
– tous les temporisateurs doivent être initialisés à "inactive" (inactifs);
– toutes les files d'attente doivent être initialisées à "empty" (vides).

La gestion de DL peut changer les valeurs de variables de configuration.

4.7.1 Paramètres, variables, compteurs, temporisateurs et files d'attente pour prendre en charge la classe "Basic"

Les paramètres, variables, temporisateurs, compteurs et files d'attente définis de 4.7.1.1 à 4.7.1.18 sont requis dans toutes les DLE. Les variables, compteurs et temporisateurs définis de 4.7.1.19 à 4.7.1.27 sont requis dans toutes les DLE à l'exception de celles ayant une classe de synchronisme temporel (voir 10.6.3) de NONE.

4.7.1.1 V(ST) intervalle de temps

V(ST) est utilisé par la DLE pour enregistrer l'intervalle de temps de liaison (voir 3.3.31), qui est un paramètre de configuration de la liaison locale. Sa plage s'étend de 1 à 4 095, et son unité est la durée de transmission d'un octet.

4.7.1.2 V(PhLO) per-DLPDU-PhL-overhead (surdébit de PhL par DLPDU)

V(PhLO) est utilisé par la DLE pour rendre compte du retard de délai induit par la PhL entre la fin du dernier octet d'une DLPDU tel qu'il apparaît sur la liaison et le début du premier octet de n'importe quelle autre DLPDU tel qu'il apparaît sur la liaison, mesuré en unités de durée d'un octet. Sa plage s'étend de 2 à 63.

4.7.1.3 V(MRD) maximum-response-delay (retard de réponse maximal)

V(MRD) est utilisé par la DLE pour enregistrer le retard de réponse minimal de la liaison (voir 3.3.21), qui est un paramètre de configuration de la liaison locale. La valeur par défaut pour cette variable est 3. Sa plage s'étend de 1 à 11, et son unité est un intervalle de temps.

NOTE Cette unité est choisie afin que toutes les mesures d'une liaison inactive concernent des multiples d'un intervalle de temps.

4.7.1.4 V(IRRD) immediate-response-recovery-delay (retard de récupération de réponse immédiate)

V(IRRD) est utilisé par la DLE pour enregistrer le retard de récupération de réponse immédiate de la liaison (voir 3.3.16), qui est un paramètre calculé de la liaison locale. La valeur par défaut pour cette variable est V(MRD) + 1. Sa plage s'étend de 2 à 12, et son unité est un intervalle de temps.

4.7.1.5 V(MRC) maximum-retry-count (compte maximal de répétitions de tentative)

V(MRC) est utilisé par la DLE pour enregistrer le compte maximal de répétitions de tentative de la liaison, qui limite le nombre permis de répétitions de tentative immédiates d'une transaction. V(MRC) est un paramètre de configuration de la liaison locale. Sa valeur par défaut est 0, signifiant que les répétitions de tentative immédiates ne sont pas permises. Sa plage s'étend de 0 à 7.

4.7.1.6 V(NRC) network-repeat-count (compte de répétitions de tentative de réseau)

V(NRC) est utilisé par la DLE pour enregistrer le compte de répétitions de tentative de réseau de la liaison étendue, qui spécifie le nombre maximal de répétitions de tentative différées d'une communication à plusieurs liaisons qu'il convient de tenter. V(NRC) est un paramètre de configuration de la liaison étendue, fondé dans une certaine mesure sur les valeurs de V(MRC) pour les diverses liaisons locales sur la liaison étendue. Sa valeur par défaut est 1, signifiant qu'il convient de tenter une seule répétition de tentative différée lorsque d'autres considérations n'interviennent pas. Sa plage s'étend de 0 à 7.

4.7.1.7 V(NDL) network-DLPDU-lifetime (durée de vie de DLPDU de réseau)

V(NDL) est utilisé par la DLE pour enregistrer la durée de vie de DLPDU du réseau de la liaison étendue, qui spécifie la période maximum durant laquelle toute DLPDU peut rester en transit au sein de la liaison étendue. Sa plage va de 1 ms à 60 s, en unités de 1 ms, et sa valeur par défaut est de 30 s.

4.7.1.8 V(TN) this-node (ce nœud)

V(TN) est utilisé par la DLE pour enregistrer l'identifiant de nœud local (voir 3.3.25), qui est un paramètre de configuration de la DLE locale. Sa valeur non initialisée par défaut est zéro, ce qui ne permet pas à la DLE d'émettre. Ses valeurs sont 0, et 10_{16} à FF_{16} (voir 4.3.2.2).

4.7.1.9 V(TL) this-link (cette liaison)

V(TL) est utilisé par la DLE pour enregistrer l'identifiant de liaison locale (voir 3.3.19), qui est un paramètre de configuration de la liaison locale. Sa valeur non initialisée par défaut est zéro, ce qui ne permet pas à un pont de transmettre vers une autre liaison. Ses valeurs sont 0, et 1000_{16} à $FEFF_{16}$ (voir 4.3.2.1).

4.7.1.10 V(MEP) DL-MAC-address-embedding-prefix (préfixe d'imbrication d'adresses MAC DL)

V(MEP) est utilisé par la DLE (voir 4.3.4) pour imbriquer l'espace adresse de 32 bits de la liaison étendue au sein de l'espace adresse de 48 bits utilisé par les normes LAN (réseau local) de l'ISO/CEI (ISO/CEI TR 8802-1). La valeur par défaut pour cette variable est zéro. Sa plage s'étend de 0 à $3FFF_{16}$.

4.7.1.11 C(RD) remaining-duration counter (compteur de durée restante)

C(RD) est utilisé par la DLE pour enregistrer la durée restante de la délégation de jeton lorsqu'il est utilisé un jeton délégué par une DLPDU PASS TOKEN (PT) ou EXECUTE SEQUENCE (ES), ou la capacité de liaison restante jusqu'à la prochaine activité programmée lorsqu'il est utilisé un jeton programmeur au cours de l'activité de maintenance de la liaison. C(RD) est initialisé à la suite de la réception d'une telle DLPDU à partir du paramètre DD de la DLPDU et il est décrémenté à la fréquence d'au moins un compte par octet de capacité de transmission écoulée jusqu'à ce qu'il atteigne la valeur zéro ou jusqu'à ce que le jeton soit retourné.

NOTE L'expression "au moins" permet des mises en œuvre pour approcher le C(RD) sous réserve qu'il soit exact ou pessimiste dans son estimation du temps effectif requis pour la prise de décision et pour la préparation et l'envoi d'une transmission. Cette permission peut réduire le nombre de compteurs requis dans une mise en œuvre réelle.

4.7.1.12 V(MID) minimum-inter-DLPDU-delay (retard minimal entre des DLPDU)

V(MID) est utilisé pour spécifier la durée minimale d'un intervalle de non-transmission qu'une DLE émettrice doit fournir après avoir reçu, ou émis, une DLPDU. Cet intervalle est mesuré en unités de durée de transmission d'un octet; l'intervalle de mesure commence par soit

a) la réception d'une primitive "confirm" de PH-DATA par la DLE qui confirme une primitive "request" de PH-DATA qui spécifiait END-OF-DATA-AND-ACTIVITY, soit
b) la réception d'une primitive "indication" PH-DATA par la DLE spécifiant END-OF-DATA, END-OF-DATA-AND-ACTIVITY ou END-OF-ACTIVITY;

celui des deux événements qui arrive le premier étant déterminant, et se termine par la présentation d'une primitive "request" de PH-DATA par la DLE spécifiant start-of-activity. La valeur par défaut pour cette variable est 0 et sa plage s'étend de 0 à la plus petite des deux valeurs 120 ou (V(MRD)-1) × V(ST).

NOTE La PhL peut spécifier sa propre valeur du retard minimal entre transmissions. Une telle contrainte est dépendante de la contrainte spécifiée ci-dessus ou peut s'appliquer en même temps que celle-ci.

4.7.1.13 T(IRRD) immediate-response-recovery-delay monitor (moniteur de retard de récupération de réponse immédiate)

T(IRRD) est utilisé par la DLE qui initie une transaction à deux DLPDU pour surveiller la liaison locale à la recherche d'une DLPDU de réponse immédiate anticipée, et pour réaffirmer que son propre jeton est le jeton actif sur la liaison si aucune réponse n'est détectée.

4.7.1.14 V(RA) reply-address (adresse de réponse)

V(RA) est utilisé par une DLE pour enregistrer l'adresse de DL vers laquelle un jeton de réponse a été passé la dernière fois par une DLPDU COMPEL ACKNOWLEDGMENT (CA), COMPEL DATA (CD) ou EXCHANGE DATA (ED). Il peut aussi être utilisé par la DLE LAS pour enregistrer l'adresse de DL vers laquelle un jeton de réponse a été passé la dernière fois par une DLPDU PROBE NODE-ADDRESS (PN).

4.7.1.15 V(OTA) outstanding-transaction-array (matrice de transactions en cours)

V(OTA) est utilisé par la DLE, dans des transactions exigeant une réponse immédiate, pour corréler une DLPDU de réponse immédiate ou de réponse différée avec la demande d'utilisateur de DLS initiateur. Il est également utilisé par la DLE pour déterminer, pour tout indice de transaction donné, si, oui ou non, cet indice est actuellement assigné par la DLE à une transaction en cours incomplète. V(OTA) est une matrice avec une plage d'indices de 0..F_{16}.

4.7.1.16 V(LTI) last-transaction-index (indice de dernière transaction)

V(LTI) est utilisé par la DLE, dans des transactions exigeant une réponse immédiate, pour s'assurer qu'une nouvelle transaction ne réutilise pas l'indice de la transaction immédiatement antérieure lancée par la même DLE. Sa plage est 0.. F_{16} et sa valeur initiale est F_{16}.

4.7.1.17 Q(US) unscheduled-service queue (file d'attente des services non programmés)

Q(US) est utilisé par la DLE pour gérer

a) les références aux files d'attente de demandes utilisateur d'adresse DL de la DLE (Q_A(UR) pour tout *A* au sein de la DLE (voir 4.7.3.1), y compris

1) les références qui signalent la nécessité de forcer une transmission de DLSDU à partir du DLCEP homologue ou éditeur correspondant lorsque *A* est une adresse de DLCEP homologue ou éditeur;
2) les références qui signalent la nécessité de forcer une instance du service d'échange d'unitdata avec l'adresse de DLSAP spécifiée, lorsque *A* est une adresse de DLSAP dont le rôle DL(SAP) est INITIATOR;
3) les références qui signalent la nécessité d'émettre le contenu d'un tampon expéditeur lié à un DLCEP local, lorsque *A* est une adresse de DLCEP locale;
4) les références qui signalent la nécessité d'envoyer une DLSDU de la file d'attente expéditrice liée à l'adresse locale de DLCEP ou de DLSAP, lorsque *A* est une adresse locale de DLCEP ou de DLSAP locale;

b) les références à des séquences actives programmées localement (voir Article 10) résultant de demandes de DL-SCHEDULE-SEQUENCE (séquence de programme de DL) (voir 8.4.3.1);
c) les DLPDU DT contenant des données utilisateur de DLS qui sont des réponses différées à des DLPDU CD et ED reçues, mises en file d'attente en appui au service DL-UNITDATA-EXCHANGE (échange d'unités de données de DL).

Cette file d'attente est utilisée à la suite de la réception d'une DLPDU PASS TOKEN (PT) adressée à la DLE (voir 3.3.24 et 5.2.2.3). La structure de cette file d'attente est décrite en 3.3.5.

4.7.1.18 V(RID) random identifier (identifiant aléatoire)

(V(RID) est un identifiant, choisi dans une distribution aléatoire approximativement uniforme, utilisé par la DLE en réponse à une DLPDU Probe NODE-address ("Sonder l'adresse de nœud"(PN)) adressée à la DLE, pour introduire un élément aléatoire dans la réponse, qui est, à son tour, utilisé pour permettre à la DLE de valider une SPDU d'activation de nœud reçue par la suite en étant adressée à la DLE.

Chaque fois qu'elle est utilisée dans une DLPDU Probe Reply (PR), avant que sa valeur ne soit copiée dans la DLPDU PR, la variable doit être mise à une nouvelle valeur aléatoire choisie uniformément dans la plage 0 à FF_{16}. Le choix aléatoire effectif doit être statistiquement indépendant des choix similaires effectués par d'autres DLE.

NOTE Cette exigence d'indépendance statistique réduit au maximum la probabilité de répétition de choix identiques par des appareils réels de construction identique, et fournit donc une base pour une éventuelle discrimination parmi les multiples DLE qui se trouvent répondre à la même adresse de DL NODE, soit par une panne, soit par une configuration erronée.

4.7.1.19 C(NT) node-time counter (compteur de temps de nœud)

C(NT) est utilisé par la DLE pour enregistrer la composante monotone croissante du temps de nœud local de la DLE (voir 3.3.35). En l'absence de tout ajustement correctif de la fréquence (voir 8.4.1.3, sa fréquence nominale à long terme de comptage doit se traduire par une incrémentation de 8 192 000 unités par seconde (= $2^{13} \times 10^3$ unités par seconde). Sa valeur initiale doit être zéro. C(NT) doit être dimensionné de façon à ne pas repasser par zéro avant la réinitialisation.

NOTE Lorsque d'autres données sont absentes, les réalisateurs peuvent supposer que la période maximale entre les réinitialisations dans les environnements industriels est de cinq ans. La fréquence de comptage spécifiée n'implique pas la granularité du comptage, qui dépend de la mise en œuvre et de la fonctionnalité de synchronisation de temps de la DLE.

4.7.1.20 V(LSTO) local-link-scheduling-time-offset (décalage de temps de programmation d'une liaison locale)

V(LSTO) est utilisé par la DLE pour enregistrer le décalage signé (différence) entre la somme N(NT) + N(LSTO) du temps de programmation de liaison locale du LAS local, tel que reçu dans une DLPDU TIME DISTRIBUTION (TD), et du C(NT) de la DLE destinataire, si bien que V(LSTO) = N_{LAS}(LSTO) + N_{LAS}(NT) – C(NT). La valeur initiale de V(LSTO) doit être zéro.

4.7.1.21 V(DLTO) DL-time-offset (décalage de temps de DL)

V(DLTO) est utilisé par la DLE pour enregistrer le décalage signé (différence) entre le temps de DL et le temps de programmation de liaison locale, ce qui conduit à DL-time = C(NT) + V(LSTO) + V(DLTO). Cette variable est également reçue dans une DLPDU TIME DISTRIBUTION (TD). Sa valeur initiale doit être zéro.

4.7.1.22 V(TQ) time-quality (qualité du temps)

V(TQ) est utilisé par la DLE pour enregistrer la qualité multipartite de la DLE et de la source et du chemin de distribution de temps sur la liaison étendue. Sa valeur initiale doit indiquer une source de temps local.

4.7.1.23 V(MD) measured-delay (retard mesuré)

V(MD) est utilisé par la DLE pour enregistrer le retard mesuré filtré dans des communications bidirectionnelles entre la DLE et le LAS courant, tel qu'il est mesuré avec une série de DLPDU ROUND-TRIP-DELAY QUERY / ROUND-TRIP-DELAY REPLY. Sa valeur initiale et sa valeur lorsque les propres fonctions LAS de la DLE sont actives doivent être zéro. Sa valeur est invalidée dans les conditions spécifiées en 8.4.1.3.

4.7.1.24 V(LN) LAS-node (nœud LAS)

V(LN) est utilisé par la DLE pour enregistrer l'identifiant de nœud (voir 3.3.19) du LAS de la liaison locale, tel que reçu dans une DLPDU TD (TIME DISTRIBUTION) (voir 8.4.1.3), ou zéro si l'identifiant de nœud du LAS sur la liaison locale n'est pas connu. Sa valeur initiale doit être zéro. V(LN) est utilisé pour déterminer la validité et la pertinence de la valeur calculée de V(MD). Sa plage est la même que celle de V(TN) (voir 4.7.1.8).

4.7.1.25 V(TSC) time-synchronization-class (classe de synchronisation du temps)

V(TSC) est utilisé par la DLE pour enregistrer la classe de synchronisation du temps de la liaison et pour déterminer la propre exigence de la DLE relative à la période minimale de distribution du temps que la DLE requiert pour maintenir le niveau spécifié de synchronisation du temps. V(TSC) doit spécifier l'une des classes de synchronisation de temps définies en 10.6.3. Sa valeur par défaut doit être la classe de synchronisation de temps 10 ms (voir 10.6.3).

4.7.1.26 T(TDP) time-distribution-period monitor (moniteur de période de distribution de temps)

T(TDP) est utilisé par la DLE pour mesurer le temps écoulé depuis le dernier envoi ou la dernière réception de DLPDU TIME DISTRIBUTION (TD).

4.7.1.27 V(TSL) time-source-link (liaison de source de temps)

V(TSL) est utilisé par la DLE pour enregistrer l'identifiant de liaison de la source de distribution de temps sur la liaison étendue. Sa plage est la même que celle de V(TL). Sa valeur initiale doit être zéro.

4.7.2 Paramètres et temporisateurs pour prendre en charge une demande d'un utilisateur de DLS

Chaque instance spécifique de demande ou de réponse de service d'un utilisateur DLS peut avoir des paramètres et temporisateurs associés, selon le type de demande ou de réponse.

4.7.2.1 P_U(SDUL) DLSDU-length request parameter (paramètre de demande de longueur de DLSDU)

P_U(SDUL) désigne la longueur de la DLSDU associée à la demande ou réponse spécifiée de l'utilisateur DLS.

4.7.2.2 P_U(SDU) DLSDU request parameter (paramètre de demande de DLSDU)

P_U(SDU) désigne la DLSDU associée à la demande ou réponse spécifiée de l'utilisateur DLS. P_U(SDU) est conceptualisé comme étant une matrice d'octets P_U(SDUL), avec des indices de 1 à P_U(SDUL).

4.7.2.3 P_U(MCD) maximum-confirm-delay parameter (paramètre de retard maximal de confirmation)

P_U(MCD) désigne un retard maximal de confirmation établi par un utilisateur DLS pour une demande spécifique, où la classe du retard est toujours déterminée par un texte associé dans la présente norme.

4.7.2.4 T_U(MCD) maximum-confirm-delay monitor (moniteur de retard maximal de confirmation)

T_U(MCD) est utilisé par la DLE pour surveiller l'achèvement des actions associées à la demande spécifiée de l'utilisateur de DLS, afin de s'assurer qu'une "confirm" de DLS correspondante est donnée à l'utilisateur DLS dans l'intervalle de temps alloué par l'utilisateur de DLS, tel que spécifié dans le retard maximal de confirmation établi par un utilisateur DLS pour la demande spécifique.

4.7.3 Files d'attente pour prendre en charge la programmation de DL basée sur une adresse de DL

Chaque instance spécifique d'une adresse de DLSAP ou adresse de DLCEP ou de DLCEP abonné au sein de la DLE doit avoir une file d'attente de demandes utilisateur associée (voir 3.3.3 et 4.7.3.1).

4.7.3.1 Q_A(UR) user-request queue (file d'attente de demandes utilisateur)

Chaque instance spécifique de Q_A(UR) est utilisée par la DLE pour gérer les demandes et réponses d'utilisateur de DLS qui exigent des transmissions de DLPDU dont l'origine est l'adresse de DL ou le DLCEP associé(e). La structure de cette file d'attente est décrite en 3.3.3.

4.7.4 Variables et temporisateurs pour prendre en charge un DLCEP

L'état de chaque DLCEP est maintenu dans des variables et des temporisateurs spécifiques au DLCEP en question. L'ensemble spécifique de variables et de temporisateurs requis pour un DLCEP donné est dépendant de la classe du DLCEP et des caractéristiques de livraison de données.

Les relations suivantes (voir Figure 7 existent entre les variables L (voir 4.7.4.10) et H (voir 4.7.4.11) et la taille de fenêtre de réception négociée W_R à un DLCEP destinataire, et les variables N (voir 4.7.4.3), R (voir 4.7.4.4), A (voir 4.7.4.5), M (voir 4.7.4.6), et la taille de fenêtre d'envoi négociée W_S (voir 4.7.4.2) au DLCEP d'envoi correspondant:

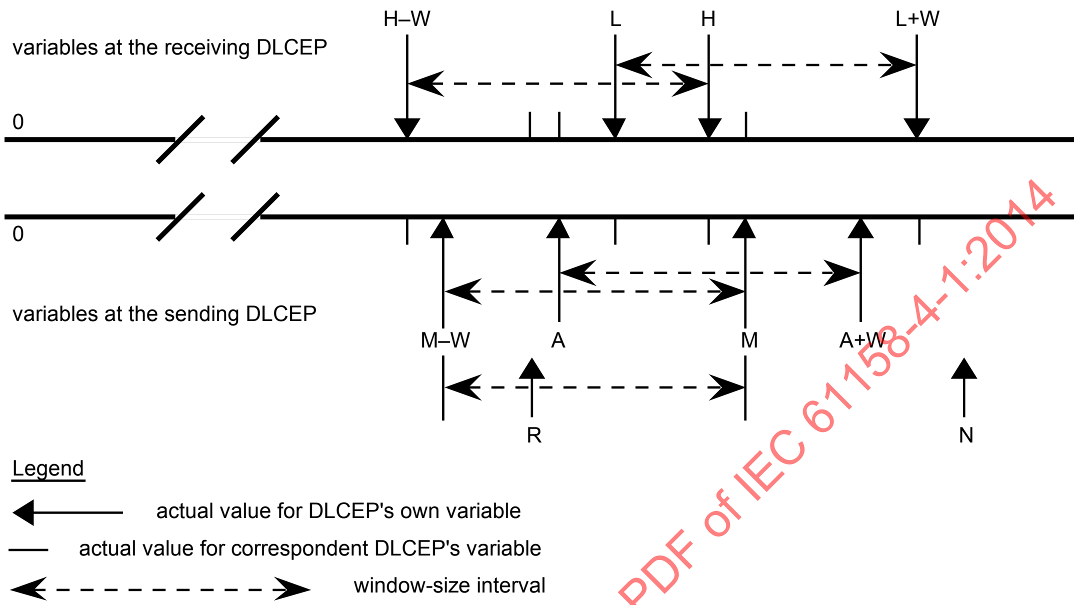
Si les indices "SC" désignent une DLE expéditrice d'une DLC, et les indices "RC" désignent une DLE destinataire de la même DLC, les relations entre les réelles variables de DLCEP, V_C(...), VC(...), et les paramètres négociés, P_C(...), du point de vue d'un seul sens de transfert de données, sont

$$P_{RC}(W_{RC}) = P_{SC}(W_{SC}) = W \quad (4)$$

$$V_{RC}(H)–W \leq V_{SC}(M)–W \leq V_{SC}(A) \leq V_{RC}(L) \leq V_{RC}(H) \leq V_{SC}(M) \leq V_{SC}(A)+W \leq V_{RC}(L)+W \qquad (5)$$

$$V_{SC}(M)–W \leq V_{SC}(R) \leq V_{SC}(M) < V_{SC}(N) \qquad (6)$$

$$V_{RC,K}(RRS) \subseteq V_{RC,K}(MRS) \qquad (7)$$



Légende

Anglais	Français
variables at the receiving DLCEP	variables au DLCEP récepteur
variables at the sending DLCEP	variables au DLCEP expéditeur
Legend	Légende
actual value for DCLEP's own variable	valeur réelle pour la variable propre au DLCEP
actual value for correspondent DLCEP's variable	valeur réelle de la variable DLCEP correspondante
window-size interval	intervalle de taille de fenêtre

Figure 7 – Relations linéaires des variables numéro de séquence de DLCEP d'envoi et de réception

4.7.4.1 V_C(ST) DLCEP state (état de DLCEP)

V_C(ST) est utilisé par une DLE pour maintenir l'état d'utilisateur courant du DLCEP. Ces états sont les mêmes que ceux qui sont montrés à la Figure 11.

$$W_R = W_S = W \qquad (8)$$

$$H–W \leq M–W \leq A \leq L \leq H \leq M \leq A+W \leq L+W \qquad (9)$$

$$M–W \leq R \leq M < N \qquad (10)$$

4.7.4.2 V_C(NP) negotiated DLCEP parameters (paramètres de DLCEP négociés)

V_C(NP) est utilisé par une DLE pour maintenir les paramètres de DLCEP courants sous la forme d'une seule variable structurée, pendant et après le processus d'établissement de DLCEP. Ces paramètres sont définis par les paramètres EC (voir. 7.1) de la DLPDU ESTABLISH CONNECTION (EC).

Dans la présente norme, les références à des paramètres spécifiques au sein des paramètres EC sont désignées par la syntaxe "V.F", où V est une variable V_C(NP) et F est un champ des paramètres EC tels que marqués en 7.1.

4.7.4.3 V_C(N) prochain numéro de séquence à assigner à une DLSDU

V_C(N) est utilisé par une DLE pour maintenir le nombre ordinal supérieur d'une unité au nombre des DLSDU dont la transmission sur le DLCEP a été demandée par l'utilisateur de DLS. Sa valeur à l'établissement du DLCEP est de un et elle augmente de façon monotone chaque fois que la transmission d'une nouvelle DLSDU est demandée par l'utilisateur de DLS. Elle n'augmente pas lorsqu'une demande d'utilisateur de DLS pour une transmission à partir d'un tampon (qui doit être sur un DLCEP dont les caractéristiques de livraison de données d'envoi sont UNORDERED ou ORDERED) entraîne la retransmission d'une DLSDU émise précédemment.

NOTE Cette dernière exigence peut être réalisée par un processus d'incrémentation à deux phases, dans lequel l'écriture dans le tampon arme le processus d'incrémentation pour tous les DLCEP qui effectuent un envoi à partir du tampon, et l'incrémentation pour un DLCEP donné a lieu pendant la formation de DLPDU immédiatement avant une transmission (mais non une retransmission) du premier segment du tampon sur le DLCEP en question.

4.7.4.4 V_C(R) numéro maximal de séquence de DLSDU non transmissible

V_C(R) est utilisé par une DLE pour un DLCEP DISORDERED ou CLASSICAL PUBLISHER et, facultativement, pour un DLCEP ORDERED PEER ou PUBLISHER DLCEP, pour maintenir le numéro ordinal de la DLSDU de plus grand numéro qui n'est plus disponible pour la transmission ou la retransmission de DLE, ou zéro lorsqu'il n'existe pas une telle DLSDU. Sa valeur à l'établissement du DLCEP est zéro, et elle n'est pas décroissante. Elle s'incrémente chaque fois qu'une DLSDU, pour laquelle une transmission a été demandée, devient indisponible pour la DLE en vue de la retransmission, soit parce que la DLSDU a été mise en file d'attente dans la DLL pendant une durée plus longue que le retard maximal de confirmation, soit parce que le numéro de séquence de la DLSDU diffère du plus grand numéro de séquence émis par la taille de fenêtre d'envoi $V_C(W_S)$, soit en raison d'une réinitialisation de la DLC.

NOTE Le fait d'incrémenter V_C(R) peut faire passer un nombre correspondant de demandes de la troisième partition à la deuxième de la Q_A(UR) concernée.

4.7.4.5 V_C(A) numéro de séquence maximal de DLSDU acquittée

V_C(A) est utilisé par une DLE pour un DLCEP DISORDERED ou CLASSICAL PEER DLCEP afin de maintenir le numéro ordinal de la DLSDU du plus grand numéro émise à partir de ce DLCEP qui a été acquittée par la DLE correspondante. Sa valeur à l'établissement du DLCEP est zéro, et elle n'est pas décroissante. Une DLPDU CA, CD, DT, ED ou RC nouvellement reçue peut la faire augmenter.

NOTE Le fait d'incrémenter V_C(A) peut faire passer un nombre correspondant de demandes de la troisième partition à la deuxième de la Q_A(UR) concernée.

4.7.4.6 V_C(M) numéro de séquence maximal de DLSDU émise

V_C(M) est utilisé par une DLE pour un DLCEP d'envoi afin de maintenir le numéro ordinal de la DLSDU de plus grand numéro qui a au moins un segment qui est, ou a été, émis à partir de ce DLCEP. Sa valeur au moment de l'établissement du DLCEP est zéro, et elle augmente de façon monotone pendant la transmission initiale du premier segment de chaque DLSDU.

4.7.4.7 V_C(MS) numéro de segment maximal de DLSDU émis

V_C(MS) est utilisé par une DLE pour un DLCEP expéditeur ORDERED, DISORDERED ou CLASSICAL pour maintenir le numéro ordinal base zéro du segment de plus grand numéro de la DLSDU de rang V_C(M) qui est, ou a été, émise à partir de ce DLCEP. Sa valeur au moment de l'établissement du DLCEP, avant la transmission d'une quelconque DLSDU, est zéro.

4.7.4.8 $V_{C,K}$(SS) segments à envoyer

$V_{C,K}$(SS) est utilisé par une DLE pour un DLCEP expéditeur pour maintenir le jeu de segments de la DLSDU de rang *K* qui a encore besoin d'être émise. Pour chaque nouvelle DLSDU (de rang *N*) présentée par l'utilisateur de DLS pour transmission, une nouvelle liste $V_{C,N}$ (SS) est créée.

Pour A<*K*≤M et R<*K*<N, des ajouts sont apportés à la liste à la suite de la réception d'une demande de retransmission pour un segment de la DLSDU de rang *K*, ou à la suite de la temporisation du temporisateur $T_{C,K}$(SS) (ou à la suite de la temporisation du temporisateur T_C(SS) si elle s'applique à la DLSDU de rang *K*, lorsque la permission de 4.7.4.9.1 est utilisée).

NOTE Le segment de plus faible numéro de la DLSDU de plus faible numéro restante est habituellement le prochain devant être émis; la DLE supprime ce segment de la liste à la suite de la transmission. Pour les DLCEP expéditeurs ORDERED ou PUBLISHER, le premier segment non émis de la première DLSDU qui n'a pas été complètement émise peut être envoyé à la place, pour maintenir une fréquence assurée de livraison de nouvelles DLSDU sur des DLC périodiques.

4.7.4.9 $T_{C,K}$(SS) moniteur de segments envoyés

$T_{C,K}$(SS) est utilisé par une DLE pour un DLCEP DISORDERED ou CLASSICAL PEER pour surveiller l'acquittement en temps opportun par un homologue de la DLSDU émise de rang *K*, avec A<*K*≤M. Il est lancé lorsque la DLSDU a été émise en totalité, et il est annulé lorsque la DLSDU est acquittée. À la suite de sa temporisation, le dernier segment de la DLSDU de rang *K* est de nouveau ajouté à $V_{C,K}$(SS), entraînant sa retransmission et déclenchant potentiellement la DLE destinataire et l'amenant à demander la retransmission d'éventuels segments de numéro plus faible qui avaient également été manqués.

4.7.4.9.1 $T_{C,K}$(SS) moniteur simplifié de segments envoyés

Il est permis de remplacer le jeu de temporisateurs par DLSDU { $T_{C,K}$(SS) } par un seul temporisateur par DLCEP, T_C(SS). Un tel remplacement conduira à un service DL équivalent lorsque le nombre des DLSDU en cours est limité par d'autres contraintes, telles qu'une petite taille de fenêtre négociée, ou lorsque le taux d'erreur résiduel du chemin de communications est très faible. Lorsque de telles contraintes ne s'appliquent pas, ce remplacement peut conduire à un service de DL médiocre.

Si ce remplacement est effectué, la définition suivante s'applique:

T_C(SS) est utilisé par une DLE pour un DLCEP DISORDERED ou CLASSICAL PEER pour surveiller l'acquittement en temps opportun par un homologue des DLSDU émises. Il est lancé, mais n'est pas redémarré, lorsqu'une DLSDU a été émise en totalité, et il est annulé lorsqu'une DLSDU est acquittée. Il est ensuite relancé s'il existe encore des DLSDU non acquittées.

À la suite de sa temporisation, le dernier segment de la DLSDU_K non acquittée ayant le plus petit numéro de séquence est de nouveau ajouté au $V_{C,K}$(SS) correspondante à la DLSDU en question, entraînant sa retransmission et déclenchant potentiellement la DLE destinataire et l'amenant à demander la retransmission d'éventuels segments de numéro plus faible qui avaient également été manqués.

4.7.4.10 V_C(L) numéro de séquence de DLSDU rapportée en dernier

V_C(L) est utilisé par une DLE pour un DLCEP destinataire ORDERED, DISORDERED ou CLASSICAL afin de maintenir le numéro ordinal de la dernière (de plus grand numéro) DLSDU reçue au DLCEP et rapportée au DLCEP distant. Sa valeur à l'établissement du DLCEP est zéro et elle augmente de façon monotone avec le rapport relatif à chaque DLSDU complètement reçue donné à l'utilisateur de DLS.

4.7.4.11 $V_C(H)$ numéro de séquence de DLSDU détecté le plus élevé

$V_C(H)$ est utilisé par une DLE pour un DLCEP PEER ou SUBSCRIBER DLCEP pour maintenir le numéro ordinal de la DLSDU de plus grand numéro reçue, ou détectée comme absente, au DLCEP. Sa valeur à l'établissement du DLCEP est zéro et elle augmente avec la réception du premier reçu des segments de chaque DLSDU nouvellement reçue ayant un plus grand numéro de séquence ou avec la détection d'une DLSDU absente ayant un plus grand numéro de séquence.

4.7.4.12 $V_C(HS)$ plus grand numéro de segment détecté du plus grand numéro de séquence de DLSDU détecté

$V_C(HS)$ est utilisé par une DLE pour un DLCEP PEER ou SUBSCRIBER pour maintenir le numéro ordinal base zéro du segment de plus grand numéro de la DLSDU de rang $V_C(H)$ qui a été reçue, ou détectée comme absente, au DLCEP. Sa valeur à l'établissement du DLCEP est zéro:

a) Elle est mise à l'indice commençant par zéro de chaque segment nouvellement reçu, $N_R(ASN)$ (voir 7.4.2.1C)6)), reçu dans une DLPDU à condition que
   1) le numéro de DLSDU reçue soit plus grand que la valeur antérieure de $V_C(H)$, ou
   2) le numéro de DLSDU reçue est le même que la valeur antérieure de $V_C(H)$, et le numéro de segment reçu est plus grand que la valeur antérieure de $V_C(HS)$.
b) Elle est mise à zéro lorsque $V_C(H)$ est avancé à la suite de la détection d'une DLSDU absente et le nombre de segments dans cette DLSDU est inconnu.

4.7.4.13 $V_{C,K}(MRS)$ missing received segments (segments reçus absents)

$V_{C,K}(MRS)$ est utilisé par une DLE pour un DLCEP ORDERED, DISORDERED ou CLASSICAL PEER ou SUBSCRIBER pour maintenir le jeu de segments absents pour la DLSDU de rang K, pour $L<K\leq H$. Des ajouts sont apportés à cette liste, ou une ou plusieurs nouvelles listes sont créées, à la suite de la réception de n'importe quel segment d'une nouvelle DLSDU. Des suppressions sont effectuées lorsque des segments inclus dans la liste sont reçus.

NOTE 1 Lorsqu'au moins un segment de la DLSDU de rang K a été reçu, le nombre total de segments dans la DLSDU est connu et la liste des segments absents est exacte. Autrement, il peut être supposé que la DLSDU absente est constituée d'un seul segment dont la retransmission doit être demandée; à la suite de la réception de n'importe quel segment de la DLSDU, la liste des segments encore absents peut être rendue exacte.

NOTE 2 Les abonnés d'un DLCEP éditeur commun ont la permission, mais ne sont pas tenus, d'écouter subrepticement les transmissions des uns des autres vers l'éditeur Une telle écoute subreptice peut assurer une détection précoce de la perte de message. Par conséquent, une ou plusieurs nouvelles listes peuvent aussi être créées à la suite de réception d'une demande de retransmission d'un segment de la DLSDU de rang K, avec $K>H$.

4.7.4.14 $V_{C,K}(RRS)$ retransmission-request required segments (segments requis pour demande de retransmission)

$V_{C,K}(RRS)$ est utilisé par une DLE pour un DLCEP ORDERED, DISORDERED ou CLASSICAL, PEER ou SUBSCRIBER pour maintenir le jeu de segments absents pour la DLSDU de rang K pour laquelle des demandes de retransmission sont requises, pour $L<K\leq H$. Des ajouts sont apportés à cette liste lorsqu'une DLPDU reçue implique qu'un ou plusieurs segments de DLSDU ou des DLSDU entières ont été manqué(e)s, et (pour les DLCEP SUBSCRIBER) à la suite de la temporisation du temporisateur $T_{C,K}(RRS)$ utilisé par la DLE pour surveiller une réponse réussie à une demande précédente de retransmission pour un ou plusieurs segments de la même DLSDU.

Des suppressions sont effectuées à la suite de la réception (y compris la réception pendant la transmission) d'une DLPDU qui soit contient un segment absent, soit en demande la retransmission.

NOTE De telles suppressions à la suite de la réception d'une demande de retransmission pour un autre abonné sur la même DLC à plusieurs homologues sont facultatives (voir Note 2 en 4.7.4.13).

4.7.4.15 $T_{C,K}$(RRS) moniteur de demande de retransmission

$T_{C,K}$(RRS) est utilisé par une DLE pour un DLCEP DISORDERED ou CLASSICAL SUBSCRIBER DLCEP, et facultativement, pour un DLCEP ORDERED PEER ou SUBSCRIBER DLCEP, pour surveiller à la recherche d'une réponse réussie à une demande précédente de retransmission pour un ou plusieurs segments de la DLSDU de rang *K*, avec $L<K\leq H$. Le temporisateur est démarré s'il n'est pas en marche, mais n'est pas redémarré s'il est en marche, lorsqu'une demande de retransmission est envoyée pour un des segments de la DLSDU incomplète. Il est annulé lorsque tous les segments de la DLSDU ont été reçus. À la suite de sa temporisation, $V_{C,K}$(RRS) est mis égal à la valeur alors courante de $V_{C,K}$(MRS).

NOTE Le temporisateur peut aussi être lancé, mais n'est pas redémarré, lorsqu'une demande de retransmission d'un autre abonné de la même DLC à plusieurs homologues, pour l'un des segments de la DLSDU incomplète, est entendue par hasard (voir Note 2 en 4.7.4.13).

4.7.4.16 T_C(RAS) residual activity stimulus (stimulus d'activité résiduelle)

T_C(RAS) doit être utilisé par une DLE en un DLCEP ORDERED, DISORDERED ou CLASSICAL, PUBLISHER ou PEER expéditeur, lorsque l'établissement de la DLC (voir 8.2.1.1) demandait une activité résiduelle sur la DLC dans ce sens (expéditeur vers destinataire) de transfert de données, pour assurer l'activité à partir de ce DLCEP pendant les périodes où toutes les DLSDU présentées pour transmission en ce DLCEP depuis l'établissement de la DLC, ou depuis la plus récente réinitialisation de la DLC, ont été émises

a) en un DLCEP DISORDERED ou CLASSICAL PEER expéditeur, lorsque toutes ces DLSDU ont été acquittées (à savoir $V_C(A) = V_C(M)$);
b) en un DLCEP PUBLISHER, ou DLCEP ORDERED PEER expéditeur, lorsque aucune de ces DLSDU n'a besoin d'être réémise (totalement ou partiellement).

Ce temporisateur redémarre à chaque transmission partant du DLCEP. À la temporisation, il envoie l'état courant du DLCEP devant être envoyé au(x) DLCEP correspondant(s).

4.7.4.17 T_C(RAM) residual activity monitor (moniteur d'activité résiduelle)

T_C(RAM) est utilisé par une DLE pour un DLCEP ORDERED, DISORDERED ou CLASSICAL, SUBSCRIBER ou PEER expéditeur pour détecter l'inactivité sur la DLC. Il n'est requis que lorsque l'établissement de la DLC (voir 8.2.1.1) demandait une activité résiduelle sur la DLC dans ce sens (destinataire vers expéditeur) du transfert de données. Il fonctionne continuellement et redémarre à la suite de la réception de n'importe quelle DLPDU sur le DLCEP. À la temporisation, il amène un DLCEP (et éventuellement la DLC) à se réinitialiser, ce qui peut, à son tour, amener un DLCEP (et éventuellement la DLC) à se déconnecter.

4.7.4.18 V_C(TNA) heure de DL du dernier accès réseau

V_C(TNA) est utilisé par la DLE pour enregistrer le temps de DL auquel une indication de DL-BUFFER-RECEIVED ou indication DL-BUFFER-SENT a été générée pour la dernière fois au DLCEP associé. Il est utilisé seulement lorsque le DLCEP a été spécifié comme étant un DLCEP de synchronisation pendant l'établissement d'un ou plusieurs autres DLCEP locaux

4.7.4.19 V_B(TW) heure de DL de la dernière écriture de tampon

V_B(TW) est utilisé par la DLE pour enregistrer le temps de DL auquel une DLSDU entière a été écrite la dernière fois dans le tampon associé (par une primitive "request" de DL-PUT ou par l'achèvement de la réception dans le tampon à un DLCEP). Il est utilisé seulement lorsque l'établissement du DLCEP spécifiait une classe d'opportunité de DL autre que NONE.

4.7.4.20 V_B(TP) temps de DL de production

V_B(TP) est utilisé par la DLE pour enregistrer le temps de DL auquel un utilisateur de DLS a transféré la DLSDU associée à la DLE. Il s'agit du temps de DL auquel l'utilisateur de DLS expéditeur a émis la demande de DL-PUT qui a placé la DLSDU associée dans le tampon de

DL d'envoi. Il est utilisé seulement lorsque l'établissement du DLCEP spécifiait une classe d'opportunité de DL autre que NONE.

4.7.4.21 V_B(TS) Statut d'opportunité de l'écriture de tampon

V_B(TS) est utilisé par la DLE pour enregistrer l'état d'opportunité associé à la DLSDU stockée dans un tampon. Si le tampon est écrit en raison d'une demande de DL-PUT, la valeur de cette variable doit être égale à l'opportunité de données de l'utilisateur de DLS. Si le tampon est écrit en raison de la réception d'une DLPDU, la valeur de cette variable doit être mise à jour comme partie intégrante de la procédure de réception.

4.7.5 Variables et temporisateurs pour prendre en charge la classe Link-Master (maître de liaison)

Les variables définies en 4.7.5 sont requises dans toutes les DLE Link Master (maître de liaison) et Bridge (pont).

4.7.5.1 V(DTA) adresse de délégation

V(DTA) est utilisé par la DLE LAS pour enregistrer l'adresse de DL vers laquelle le jeton de réponse a été passé la dernière fois par une DLPDU PASS TOKEN (PT), EXECUTE SEQUENCE (ES), ou TRANSFER LAS (TL).

4.7.5.2 V(LL) liste active de liaison locale

V(LL) est utilisé par la DLE LAS pour enregistrer le jeu d'adresses de DL NODE (voir 3.3.24 et 5.2.2.3) sur la liaison locale qui paraît être utilisée par des DLE homologues. Il est utilisé et mis à jour par la DLE LAS au cours des activités de sondage d'adresse et de première occasion pour le LAS, telles que décrites en 10.2.

NOTE Certaines normes nationales déjà existantes appellent un tel jeu une «liste active», parce qu'elle indique le jeu courant de DLE interagissantes présentes sur la liaison locale.

4.7.5.3 V(TCL) liste de circulation de jetons

V(TCL) est utilisé par la DLE LAS pour enregistrer le jeu d'adresses de DL NODE (voir 3.3.24 et 5.2.2.3) sur la liaison locale vers laquelle il convient de faire circuler une DLPDU PT afin de fournir une occasion pour des transmissions non programmées, y compris des demandes à la DLE LAS pour la programmation de communications. Cette circulation de jeton émule le passage de jetons distribués indiqué dans les normes nationales et internationales antérieures.

NOTE Cette variable est utilisée pour restreindre aux seuls nœuds “maîtres” (à savoir autosuffisants) de la liaison locale la circulation d'un jeton sans limitation. Cette fonctionnalité aide à la migration à partir de certaines normes nationales déjà existantes.

V(TCL) est utilisé et mis à jour par la DLE LAS au cours des activités de sondage d'adresse et de première occasion pour le LAS, telles que décrites en 10.2. Les DLE représentées dans V(TCL) doivent toujours être un sous-ensemble des DLE représentées dans V(LL).

4.7.5.4 V(ENRL) expected-non-response list (liste de non-réponses attendues)

V(ENRL) est utilisé par la DLE LAS pour enregistrer le jeu d'adresses de DL NODE (voir 3.3.24 et 5.2.2.3) sur la liaison locale des DLE de facteur d'utilisation fractionnaire (FDC) qui sont censées être non réactives (c'est-à-dire "endormies"). Il est un sous-ensemble du jeu des DLE FDC sur la liaison et il change de façon dynamique à mesure que les DLE FDC donnent à la DLE LAS des informations relatives à

a) leur état d'attention (“état de veille”), ou
b) leur intention de devenir non réactives (de "s'endormir").

Le manquement d'une DLE énumérée dans le V(ENRL) à répondre à une DLPDU PT adressée à la DLE, ou à une DLPDU ES adressée à un DLSEP structuré hiérarchiquement (voir en 5.2.2.1 le format 1 ou en 5.2.2.2 le format 1) de la DLE,

1) ne doit pas être traité comme une erreur;
2) la transmission de la DLPDU PT ou ES ne doit pas être tentée de nouveau.

NOTE Cette dernière exigence équivaut à l'utilisation d'une valeur de zéro (0) pour V(MRC) lors de l'envoi d'une DLPDU PT (voir 6.15.3) vers une telle DLE.

4.7.5.5 V(MST) maximum-scheduled-traffic (trafic programmé maximal)

V(MST) est utilisé par la DLE LAS, au cours de toute construction de programme dynamique, pour déterminer la fraction maximale de l'intervalle théorique (V(TTRT) (voir 4.7.5.11) qui peut être dédiée à l'activité de DL explicitement programmée.

V(MST) est un paramètre de configuration de la liaison locale qui prend la forme d'une fraction binaire d'un octet. La plage pour ce paramètre s'étend de $0{,}00_{16}$ à $0{,}BF_{16}$; sa valeur par défaut est $0{,}40_{16}$.

4.7.5.6 V(MSO) maximum-scheduling-overhead -surdébit maximal de programmation)

V(MSO) spécifie le surdébit maximal de programmation permis à une DLE LAS par le programme de liaison existant. Sa plage s'étend de 0 à $3F_{16}$; sa valeur par défaut est $3F_{16}$ et son unité est la durée de transmission d'un octet. Ce surdébit est inclus dans le temps alloué pour chaque activité programmée, et il est donc utilisé seulement pendant la construction d'un programme et la détermination si, oui ou non, une DLE peut servir de LAS pour un programme existant.

4.7.5.7 V(DMDT) default-minimum-token-delegation-time (temps de délégation de jeton minimal par défaut)

V(DMDT) est utilisé par la DLE LAS pour déterminer la quantité par défaut minimale de capacité de liaison locale que le LAS doit allouer à une DLE dans une seule DLPDU PT envoyée à la DLE.

V(DMDT) est un paramètre de configuration de la liaison locale. La plage pour cette variable s'étend de 20_{16} à $7FFF_{16}$, sa valeur par défaut est 54_{16} + V(PhLO), ce qui permet l'envoi d'une DLPDU URGENT, et son unité est la durée de transmission d'un octet

4.7.5.8 V(DTHT) default-token-holding-time (temps de détention d'un jeton par défaut)

V(DTHT) est utilisé par la DLE LAS pour spécifier la quantité par défaut initiale de capacité de liaison locale qu'il convient que le LAS alloue à chaque DLE, en un seul et même cycle de "circulation du jeton", lorsque le LAS envoie une ou plusieurs DLPDU PT vers la DLE.

V(DTHT) est un paramètre de configuration de la liaison locale. La plage pour cette variable s'étend de 114_{16} à 65 000, sa valeur par défaut est 114_{16} + V(PhLO), ce qui permet l'envoi d'une DLPDU TIME-AVAILABLE, et son unité est la durée de transmission d'un octet.

4.7.5.9 V(LTHT) link-maintenance-token-holding-time (temps de détention d'un jeton de maintenance de liaison)

V(LTHT) est utilisé par la DLE LAS pour spécifier la quantité initiale de capacité de liaison locale qu'il convient d'allouer à des activités de maintenance de liaison relatives au LAS dans un seul et même cycle de "circulation de jeton". Ces activités consistent notamment à

– sonder une adresse de DL NODE non utilisée par le biais d'une DLPDU PN pour l'apparition d'une DLE actuellement non active sur la liaison, et l'envoi consécutif d'une

DLPDU DT contenant une SPDU d'activation de nœud vers n'importe quelle DLE de ce type;

– mesurer le retard de communications aller-retour avec une autre DLE sur la liaison locale par le biais d'une DLPDU RQ;
– transférer des informations de programme mises à jour à d'autres DLE maîtresses de liaison sur la liaison locale afin de permettre la poursuite du programme après le transfert du rôle de LAS à une DLE de ce type;
– coordonner un programme multiliaison avec d'autres DLE LAS de la liaison étendue.

V(LTHT) est un paramètre de configuration de la liaison locale. La plage pour cette variable s'étend de 124_{16} à 65 000, et son unité est la durée de transmission d'un octet. Sa valeur par défaut est

$124_{16} + (3 \times V(PhLO)) + (V(IRRD) \times V(ST))$, ce qui permet

le sondage d'une adresse de DL NODE et l'envoi d'une DLPDU DT TIME-AVAILABLE.

4.7.5.10 V(MTHA) maximum-token-holding-time-array (matrice des temps maximaux de détention de jeton)

V(MTHA) est utilisé par le LAS pour spécifier, séparément pour chaque DLE énumérée dans V(TCL) (voir 4.7.5.3), la quantité initiale de capacité de liaison locale qu'il convient que le LAS alloue à la DLE en question, en un seul et même cycle de "circulation du jeton", lorsque le LAS envoie une ou plusieurs DLPDU PT vers la DLE en question. La plage et les unités de ce paramètre sont les mêmes que ceux de V(DTHT) (voir 4.7.5.8). Sa valeur par défaut est également V(DTHT).

4.7.5.11 V(TTRT) target-token-rotation-time (temps ciblé de rotation de jeton)

V(TTRT) est utilisé par la DLE LAS pour spécifier la borne supérieure du temps requis pour un cycle de "circulation de jeton" à toutes les DLE de la liaison locale. Un cycle de "circulation de jeton" est mesuré comme étant l'intervalle entre des occurrences successives d'envoi d'une DLPDU PT, avec un sous-champ d'utilisation de jeton spécifiant RESTART (voir 6.15.2), par la DLE LAS vers l'adresse DL NODE ayant le plus petit numéro représentée dans V(LL).

V(TTRT) est un paramètre de configuration de la liaison locale. La plage pour cette variable s'étend de 1 à 60 000, sa valeur par défaut est 60 000 et son unité est 1 ms.

4.7.5.12 V(ATRT) actual-token-rotation-time (temps effectif de rotation de jeton)

V(ATRT) est utilisé par la DLE LAS pour déterminer le temps effectif utilisé pour chaque cycle de "circulation du jeton" vers toutes les DLE de la liaison locale, tel que mesuré par l'intervalle entre des occurrences successives d'envoi de la DLPDU PT, avec un sous-champ d'utilisation de jeton spécifiant RESTART (voir 6.15.2), par la DLE LAS vers l'adresse DL NODE ayant le plus petit numéro représentée dans V(LL).

4.7.5.13 V(RTHA) remaining-token-holding-time-array (matrice des temps de détention de jeton restants)

V(RTHA) est utilisé par le LAS pour spécifier, séparément pour chaque DLE énumérée dans V(TCL) (voir 4.7.5.3), la quantité restante de capacité de liaison locale qu'il convient que le LAS alloue à la DLE en question, au cours du cycle courant "circulation du jeton", lorsque le LAS envoie une ou plusieurs DLPDU PT vers la DLE en question. La plage et les unités de cette matrice sont les mêmes que celles de V(MTHA), et elle est réinitialisée à partir de V(MTHA) au début de chaque cycle de "circulation du jeton".

4.7.5.14 V(NTHN) next-token-holding-node (prochain nœud de détention de jeton)

V(NTHN) est utilisé pour enregistrer l'adresse de DL NODE de la prochaine DLE vers laquelle "il convient de circuler le jeton", qui est la prochaine adresse de DL NODE vers laquelle une DLPDU PT sera envoyée Sa plage s'étend de 10_{16} à FF_{16} (voir 4.3.2.2), et elle progresse de façon cyclique à travers l'ensemble des DLE spécifiées par V(TCL) (voir 4.7.5.3).

4.7.5.15 V(FUN) first-unpolled-node (premier nœud non interrogé)

V(FUN) (FUN) spécifie la première adresse de DL NODE d'une série d'adresses de DL NODE consécutives qui doivent être omises du sondage ordonné d'adresses DL NODE pour les DLE non spécifiées par V(LL). Sa page s'étend de 14_{16} à $F7_{16}$ (voir 4.3.2.2).

4.7.5.16 V(NUN) number-of-consecutive-unpolled-nodes (nombre de nœuds non interrogés consécutifs)

V(NUN) spécifie le nombre d'adresses de DL NODE consécutives qui doivent être omises du sondage ordonné des adresses de DL NODE pour les DLE non spécifiées par V(LL). Sa plage s'étend de 00_{16} à $E4_{16}$ et sa valeur par défaut est 0.

4.7.5.17 P(TRD) token-recovery-delay (retard de récupération de jeton)

P(TRD) est utilisé par la DLE LAS pour enregistrer le retard de récupération de jeton (voir 3.3.34) de la DLE, qui est déterminé uniquement par la DLE elle-même. La valeur par défaut pour cette variable est 14. Sa plage s'étend de V(MRD)+3 à 14, et son unité est un intervalle de temps.

NOTE Cette unité est choisie afin que toutes les mesures d'une liaison inactive concernent des multiples d'un intervalle de temps.

4.7.5.18 V(TDP) time-distribution-period (période de distribution de temps)

V(TDP) est utilisé pour déterminer la fréquence minimale de distribution de temps sur la liaison locale. Sa valeur initiale doit être la valeur minimale requise pour la classe 'time-synchronisation (synchronisation du temps) de la liaison, V(TSC) telle que spécifiée en 10.6.3. Sa plage s'étend de 5 ms à 55 s, et son unité est 1 ms.

4.7.5.19 V(MICD) maximum-inactivity-to-claim-LAS-delay (retard maximal d'inactivé pour réclamation de LAS)

V(MICD) est utilisé par la DLE LAS pour enregistrer le retard maximal d'inactivité pour réclamer le LAS (voir 3.3.32), qui est un paramètre de configuration de la liaison locale. Sa plage s'étend de 1 à 4 095, et son unité est un huitième de la période de transmission d'un octet.

4.7.5.20 V(LDDP) LAS-data-base-distribution-period (période de distribution de base de données LAS)

V(LDDP) est utilisé par le LAS pour déterminer le temps entre deux distributions successives de la base de données LAS au moyen des SPDU "statut de base de données LAS" envoyées sur la liaison locale. La plage de V(LDDP) s'étend de 100 ms à 55 s, et son unité est 1 ms. La valeur par défaut de cette variable est 5 s.

4.7.6 Variables et temporisateurs pour prendre en charge la classe Bridge (pont)

Les variables et temporisateurs définis en 4.7.6 et en 10.4 sont requis dans toutes les DLE Bridge (pont).

4.7.6.1 V(ML) maximum-link (liaison maximale)

V(ML) peut être utilisé par une DLE pont pour limiter le nombre d'entrées dans la table d'informations de transmission par défaut du pont, qui peut avoir une entrée pour chaque liaison de la liaison étendue. Sa plage s'étend de 1000_{16} à $FEFF_{16}$ (voir 4.3.2.1), qui est l'identifiant de liaison maximale pour le réseau étendu.

Des variables et temporisateurs supplémentaires sont définis en 10.4, la plupart par référence à l'ISO/CEI 10038.

5 Structure générale et codage des PhIDU et des DLPDU, et éléments de procédure connexes

Dans l'Article 5, toute référence au bit *K* d'un octet est une référence au bit dont le poids dans un nombre entier non signé d'un octet est 2^K.

NOTE Elle est parfois appelé numérotation de bit "little endian" (petit boutiste).

5.1 Structure et codage des PhIDU

Chaque PhIDU est constituée d'une information de commande d'interface Ph (PhICI) et, dans certains cas, d'un octet de données d'interface Ph (voir 4.4). Lorsque la DLE émet une DLPDU, elle calcule une séquence de contrôle de trame pour la DLPDU telle que spécifiée en 5.2.5.1, concatène la DLPDU et la séquence de contrôle de trame, et émet la paire concaténée sous la forme d'une séquence de PhIDU comme suit:

a) La DLE produit une seule primitive "request" de Ph-DATA avec PhICI spécifiant START-OF-ACTIVITY et attend la consécutive primitive "confirm" de Ph-DATA.
b) La DLE produit une séquence de primitives "request" de Ph-DATA avec PhICI spécifiant DATA, chacune étant accompagnée d'un octet de la DLPDU en tant que données d'interface Ph, du premier au dernier octet de la DLPDU, puis chaque primitive "request" de DATA attend la consécutive primitive "confirm" de Ph-DATA.
c) La DLE produit une séquence de deux primitives "request" de Ph-DATA avec PhICI spécifiant DATA, chacune étant accompagnée d'un octet de la FCS en tant que données d'interface Ph, du premier au dernier octet de la FCS, puis chaque primitive "request" de DATA attend la consécutive primitive "confirm" de Ph-DATA.
d) La DLE produit une seule primitive "request" de Ph-DATA avec PhICI spécifiant END-OF-DATA-AND-ACTIVITY et attend la consécutive primitive "confirm" de Ph-DATA.

La DLE forme une DLPDU reçue en concaténant la séquence d'octets reçue en tant qu'informations de commande d'interface Ph de primitives "indication" de Ph-DATA consécutives, en calculant une séquence de contrôle de trame pour ces octets reçus tels que spécifiés en 5.2.5.2, et vérifie la justesse du syndrome (résiduel) de la FCS calculée comme suit:

e) La DLE reçoit une seule primitive "indication" de Ph-DATA avec PhICI spécifiant START-OF-ACTIVITY, et initialise son calcul d'une FCS pour la DLPDU reçue.
f) La DLE reçoit une séquence de primitives "indication" de Ph-DATA avec PhICI spécifiant DATA, chacune étant accompagnée d'un octet de la DLPDU reçue en tant que données d'interface Ph, calcule par incréments une FCS sur l'octet reçu, et concatène tous les octets reçus, sauf les deux derniers, pour former la DLPDU reçue.
g) La DLE reçoit une seule primitive "indication" de Ph-DATA avec PhICI spécifiant soit END-OF-DATA, soit END-OF-DATA-AND-ACTIVITY, soit END-OF-ACTIVITY, et vérifie la justesse du syndrome de la FCS calculée:
   1) Si les PhICI spécifiaient END-OF-DATA ou END-OF-DATA-AND-ACTIVITY, et le syndrome de la FCS calculée était correct, la DLE rapporte la DLPDU reconstruite et les deux octets de la FCS reçue comme étant une DLPDU correctement reçue apte à une analyse ultérieure.

2) Autrement, la DLE incrémente ses statistiques de gestion pour refléter la DLPDU incorrectement reçue.

5.2 Structure, codage et éléments de procédure communs de DLPDU

Chaque DLPDU est constituée d'un champ de contrôle de trame qui spécifie le type de DLPDU et achemine des paramètres de petite taille (octet fractionnaire) de la DLPDU; de zéro à trois champs d'adresse explicite, contenant chacun une adresse de DL, tous de la même longueur; des paramètres supplémentaires de la DLPDU; et pour la plupart des DLPDU, un champ de données utilisateur acheminant toute ou partie d'une DLSDU. À cela est abouté avant transmission et retiré après réception, un champ FCS (voir 5.2.5) utilisé pour vérifier l'intégrité de la DLPDU reçue.

5.2.1 Champ "Frame control" (contrôle de trame (FC))

Le champ "contrôle de trame" (FC) est constitué d'un seul octet. Il spécifie le type de la DLPDU. Pour de nombreux types de DLPDU, il achemine également un certain nombre de paramètres d'octets fractionnaires, appelés sous-champ de contrôle de trame, spécifiques au type de DLPDU.

Certains types de DLPDU exigent une réponse immédiate. De telles DLPDU ne peuvent être envoyées qu'en cas de détention d'un jeton programmé ou délégué.

Lorsqu'un jeton est délégué, la priorité spécifiée dans la DLPDU jeton est la priorité minimale requise de toutes les DLPDU envoyées au cours de la période suivante d'utilisation de jeton. Les DLPDU de plus faible priorité ne doivent pas être envoyées pendant la période en question.

5.2.1.1 Sous-champ "Address size" (taille d'adresse)

Un champ "taille d'adresse" est utilisé pour spécifier le nombre d'octets dans chaque champ adresse de la DLPDU.

Dans les DLPDU qui prennent en charge plusieurs tailles d'adresse, ce sous-champ occupe le bit 3 du champ " frame control". Son codage est:

0: **SHORT** – les champs d'adresse de la DLPDU sont de deux octets chacun;

1: **LONG** – les champs d'adresse de la DLPDU sont de quatre octets chacun et leurs champs "appellation de liaison" sont autres que zéro, si possible.

La DLPDU DT a aussi une forme spéciale ne comportant que des adresses de DL implicites:

– **VERY-SHORT** – la DLPDU contient logiquement un unique champ d'adresse de DL, qui est vide (zéro octet de longueur).
  i) Si l'unique champ d'adresse DL vide est une adresse source, l'adresse DL source de la DLPDU courante est implicitement l'adresse DL de destination de la DLPDU CA, CD ou ED immédiatement antérieure sur la liaison (qui doit avoir été une adresse DLCEP).
  ii) Si l'unique champ d'adresse de DL vide est une adresse de destination, l'adresse DL de destination de la DLPDU courante est implicitement l'adresse de DL source de la DLPDU CA, CD ou ED immédiatement antérieure sur la liaison (qui peut elle-même être une adresse de DLSAP ou de DLCEP implicite).

NOTE Cette forme VERY SHORT est indiquée par d'autres composantes de l'octet de contrôle de trame de la DLPDU, et non par son sous-champ "taille d'adresse".

5.2.1.2 Sous-champ d'utilisation finale de jeton

L'appellation "final-token-use" (utilisation finale de jeton) est utilisée pour optimiser le retour d'un jeton délégué à la fin de la dernière transaction d'une instance d'utilisation de jeton délégué. Ce sous-champ est présent dans la plupart des DLPDU, y compris toutes les DLPDU qui peuvent être envoyées par une DLE qui détient un jeton délégué ou jeton de réponse. Lorsqu'il est présent, ce sous-champ occupe le bit 2 du champ de contrôle de trame. Son codage et sa sémantique dépendent du type de DLPDU comme suit.

a) Le codage et la sémantique pour les DLPDU CA, CD, CT, ED et RQ sont comme suit:

   0: **NOT-FINAL** – le jeton délégué n'est pas retourné au LAS à la fin de la transaction courante;

   1: **FINAL** – le jeton délégué est retourné au LAS à la fin de la transaction courante, après la transmission de la DLPDU et sa DLPDU de réponse immédiate demandée, qu'elle soit détectée ou non, et aucune utilisation supplémentaire du jeton retourné n'est nécessaire à ce moment;

b) Le codage et la sémantique des DLPDU DC, DT, EC, RC, RR, SR et TD sont comme suit:

   0: **NOT-FINAL** – le jeton délégué n'est pas retourné au LAS à la fin de la transaction courante;

   1: **FINAL** – le jeton délégué est retourné au LAS à la fin de la transaction courante, après la transmission de la DLPDU, et aucune utilisation supplémentaire du jeton retourné n'est nécessaire à ce moment.

c) Le codage et la sémantique des DLPDU PT et ES sont comme suit:

   0: **RESTART** – il s'agit de la délégation de jeton initiale dans le cycle courant de "circulation du jeton", ou d'exécution de séquence programmée, et donc il convient de redémarrer la séquence indiquée et toutes les éventuelles transactions programmées à répétitions;

   1: **CONTINUE** – il s'agit d'une délégation de jeton ultérieure (c'est-à-dire secondaire) dans le cycle courant de "circulation du jeton" ou d'exécution de séquence programmée, et donc il convient de poursuivre la séquence lancée précédemment de transactions placées en file d'attente ou programmées.

Lorsqu'un pont reçoit pour transmission une DLPDU qui contient un sous-champ "final-token-use" dont la valeur est différente de celle requise par le pont, le pont doit compléter le sous-champ "final-token-use" de la DLPDU reçue avant de la transmettre et doit apporter une modification compensatoire au champ FCS de la DLPDU reçue (voir 5.2.5.3) pour préserver la protection d'intégrité de FCS assurée par l'émetteur original de la DLPDU.

NOTE Seules les DLPDU CA, CD, DC, DT, EC, ED et RC (voir 6.1 à6.7) sont transmises par des ponts.

5.2.1.3 Sous-champ "priority" (priorité)

Une appellation Priority est utilisée pour spécifier la priorité de la DLPDU ou de la transaction, afin de limiter la taille du champ de données utilisateur de la DLPDU, et contraindre la priorité minimale (la plus basse) de toute DLPDU envoyée comme réponse immédiate demandée par cette DLPDU ou comme utilisation de jeton déléguée par cette DLPDU. Ce sous-champ est présent dans toutes les DLPDU CA, CD, DT, ED, PT et RC qui contiennent des adresses de DL explicites Lorsqu'il est présent, ce sous-champ occupe les bits 1 et 0 du champ de contrôle de trame. Son codage est:

   01: **URGENT** (haute) priorité;

   10: **NORMAL** (moyenne) priorité;

   11: **TIME-AVAILABLE** (basse) priorité.

5.2.2 Champs d'adresses de DL

La structure des adresses DL est spécifiée en 4.3, qui spécifie aussi les adresses et plages d’adresses pré-assignées normalisées.

5.2.2.1 Champ d'adresse LONG

Un champ d'adresse **LONG** est habituellement constitué d’une séquence fixe de trois parties telles que spécifiées en 4.3. Les parties sont

a) une composante "appellation de liaison" explicite;

b) une composante "appellation de nœud" explicite;

c) une composante "sélecteur" explicite.

Séparément, les trois parties reflètent une structure d’adresse hiérarchique. Cette hiérarchie peut être partiellement ou totalement aplatie. Ces variantes sont montrées à la Figure 8.

format 1	**liaison**	**Nœud**	**Sélecteur**
	2 octets	1 octet	1 octet

soit

format 2	**Liaison**	**Nœud || Sélecteur**
	2 octets	2 octets

soit

format 3	**Liaison || Nœud || Sélecteur**
	4 octets

Figure 8 – Variantes de structures d'adresse de DL

Lorsqu’il est présent, le champ Link doit être délocalisé à la transmission et à la réception, comme spécifié en 5.2.2.4. De même, lorsque le champ Link est présent et spécifie la liaison locale, et lorsque le champ de Node est présent, le champ Node doit être délocalisé à la transmission.

5.2.2.2 Champ d'adresse SHORT

Un champ d'adresse **SHORT** est habituellement constitué d’une séquence fixe d’une partie implicite et de deux parties explicites, telles que spécifiées 4.3 et à la Figure 9. Les parties sont

a) une composante "appellation de liaison" implicite, spécifiant la liaison locale, qui est toujours présente;

b) une composante "appellation de nœud" explicite;

c) une composante "sélecteur" explicite.

Séparément, les parties reflètent une structure d’adresse hiérarchique. Cette hiérarchie peut être partiellement aplatie. Les variantes sont montrées à la Figure 9.

format 1	0000	**Nœud**	**Sélecteur**
		1 octet	1 octet

soit

format 2	0000	**Nœud || Sélecteur**
		2 octets

Figure 9 – Champ d'adresse de DL SHORT – Variantes de structures implicites

Lorsqu'il est présent, le champ Node doit toujours être délocalisé à la transmission, comme spécifié en 5.2.2.4.

5.2.2.3 Champ d'adresse de DL NODE

Une adresse de DL **NODE** est une version à un seul octet d'un champ d'adresse SHORT qui désigne certaines fonctions de prise en charge de DL de la DLE. Elle est constituée d'une séquence fixe de deux parties implicites et d'une partie explicite, telles que spécifiées en 4.3 et à la Figure 10. Les parties sont

a) une composante "appellation de liaison" implicite, spécifiant la liaison locale;

b) une composante "appellation de nœud" explicite, spécifiant la DLE;

c) une composante "sélecteur" implicite de zéro, spécifiant les fonctions de prise en charge DL de la DLE.

format 1	0000	**Nœud**	00
		1 octet	

Figure 10 – Champ d'adresse de DL NODE – structure implicite

Le champ Node doit toujours être délocalisé à la transmission, comme spécifié en 5.2.2.4.

5.2.2.4 Délocalisation

Les processus de transmission et de réception suivants sont collectivement appelés "délocalisation" (voir 3.3.2), qui est le processus interne à la DLE de conversion d'adresses de DL synonymes en une forme canonique en vue de la transmission, ou pour la reconnaissance d'adresses de DL pendant la réception.

Deux valeurs du champ Link – la valeur zéro et la valeur égale à V(TL) – sont équivalentes pour désigner la liaison locale:

a) À la transmission, lorsque la valeur d'un champ Link explicite serait autrement zéro, la valeur de ce champ Link doit toujours être mis à V(TL).

NOTE 1 La variable V(TL) peut avoir la valeur zéro, auquel cas cette substitution n'effectue aucun changement.

NOTE 2 Cette équivalence permet à une DLE d'émettre sur la liaison locale sans connaître la valeur correcte de V(TL) pour la liaison, ou pendant les périodes où cette valeur est en transition (auquel cas, elle est changée administrativement de zéro à une valeur autre que zéro).

NOTE 3 Cette disposition d'équivalence est une aide aux utilisateurs DLS, et potentiellement dans les mises en œuvre de ce protocole DL; elle permet à des références à la composante d'adresse DL de la liaison locale d'être représentées uniformément par la valeur zéro.

b) À la réception, en ce qui concerne le champ Link, la valeur V(TL) doit être considérée comme équivalant à la valeur zéro, excepté qu'une DLPDU contenant une adresse de DL avec un champ Link effectivement égal à zéro ne doit pas être transmise par un pont sur une liaison différente.

Au sein d'une DLE, et lors de l'adressage de la liaison locale, deux valeurs du champ Node – la valeur zéro et la valeur égale à V(TN) – sont équivalentes pour désigner la DLE locale sur la liaison locale.

c) À la transmission, lorsque la valeur du champ Link est équivalente à V(TL), et la valeur d'un champ Node serait autrement zéro, la valeur du champ Node en question doit toujours être mise à V(TN).

NOTE Cette disposition d'équivalence est une aide aux utilisateurs DLS, et potentiellement dans les mises en œuvre de ce protocole DL; elle permet à des références à la composante d'adresse DL de nœud local d'être représentées uniformément par la valeur zéro.

5.2.3 Champ "Parameter" (paramètre)

Chaque classe de DLPDU peut avoir un champ de paramètre spécifique à la classe de DLPDU; ils sont tous décrits à l'Article 7.

Lorsqu'il est souhaitable de distinguer entre les valeurs d'une variable DLE, V(xx), ou d'un compteur, C(xx), qui est copié(e) dans un champ de paramètre, et la valeur courante de la même variable ou du même compteur, la valeur dans le champ de paramètre est appelée N(xx), parce qu'elle ne suit plus à trace les modifications apportées à la valeur de la variable ou du compteur source. La nécessité de cette séparation de nomenclature est particulièrement évidente dans le cas des compteurs tels que C(NT), qui n'arrêtent jamais de compter, et dans les diverses DLPDU relatives au temps (TD, RQ et RR), qui peuvent contenir plusieurs champs basés sur différents échantillonnages de C(NT)

5.2.4 Champ de données d'utilisateur (User data)

Les DLPDU RC, CA, DT et ED, associées aux services de transfert de données en mode sans connexion et en mode orienté connexion, contiennent un champ de données utilisateur qui est utilisé pour acheminer une DLSDU partielle ou complète d'un utilisateur de DLS à un autre. La taille de ce champ de données utilisateur est contrainte à ne pas être supérieure à celle qui est permise par la priorité de la DLPDU convoyeuse:

01: **URGENT** (haute) priorité: ≤ 64 octets;

10: **NORMAL** (moyenne) priorité: ≤ 128 octets;

11: **TIME-AVAILABLE** (basse) priorité: ≤ 256 octets.

Les DLPDU PR contiennent un champ de données utilisateur qui est utilisé pour acheminer une SPDU "sonder réponse" (voir 9.3.2.1) de la DLE détenant le jeton vers la DLE LAS courante; sa taille est contrainte à ne pas être supérieure à celle qui est permise à la priorité URGENT – à savoir 64 octets.

Les DLPDU TL contiennent un champ de données utilisateur qui est utilisé pour acheminer une SPDU "statut de base de donnes de LAS" (voir 9.3.2.3) de la DLE LAS vers la DLE adressée; sa taille est contrainte à ne pas être supérieure à celle qui est permise à la priorité URGENT – à savoir 64 octets.

Les DLPDU DC et EC contiennent un champ de données utilisateur. Leur priorité implicite est TIME-AVAILABLE; toutefois, la taille de leurs champs de données est contrainte à ne pas être supérieure à celle qui est permise à la priorité NORMAL – à savoir 128 octets.

5.2.5 Champ "Frame check sequence" (séquence de contrôle de trame)

Dans le présent paragraphe, toute référence au bit K d'un octet est une référence au bit dont le poids dans un nombre entier non signé d'un octet est 2^K.

NOTE 1 Elle est parfois appelé numérotation de bit "little endian" (petit boutiste).

Pour le Type de protocole de la présente norme, tout comme dans d'autres Normes internationales (voir Note 2), la détection d'erreur au niveau des DLPDU est assurée en calculant et aboutant une séquence de contrôle de trame (FCS) de plusieurs bits aux autres champs de DLPDU au cours de la transmission afin de former un "mot de code systématique" [1] de longueur n constitué de k bits de message de DLPDU suivis de n - k (égal à 16) bits redondants, et en calculant au cours de la réception que le message et la FCS concaténée forment un mot de code légitime (n,k). Le mécanisme pour cette vérification est comme suit:

[1] W. W. Peterson and E. J. Weldon, Jr., *Error Correcting Codes* (2nd edition), MIT Press, Cambridge, 1972.

NOTE 2 L'ISO/CEI 8802 et l'ISO/CEI 9314-2, par exemple.

La forme générique du polynôme générateur pour cette construction de FCS est spécifiée dans l'Équation (6) et le polynôme pour le résidu prévu du récepteur est spécifié dans l'Équation (11). Les polynômes spécifiques pour la présente norme sont spécifiés dans le Tableau 9. Une mise en œuvre exemplaire est débattue à l'Annexe A.

Tableau 9 – Longueur de FCS, polynôme et résidu prévu

Item	Valeur	
n-k	16	
G(x)	$X^{16} + X^{12} + X^{11} + X^{10} + X^8 + X^7 + X^6 + X^3 + X^2 + X + 1$	(Notes 1, 2, 3)
R(x)	$X^{15} + X^{14} + X^{13} + X^9 + X^8 + X^7 + X^4 + X^2$	(Note 4)

NOTE 1 Les mots de code D(X) construits à partir de ce polynôme G(X) ont une distance de Hamming de 4 pour les longueurs ≤ 344 octets et une distance de Hamming de 5 pour les longueurs ≤ 15 octets.

NOTE 2 Ce polynomial G(X) est relativement premier par rapport à tous les polynômes habituellement utilisés dans les ETD (équipements de transmission de données) (modems) et n'est donc corrompu par aucun de ceux-ci: le polynôme de codage différentiel $1 + X^{-1}$ et tous les polynômes primitifs d'embrouillage de la forme $1 + X^{-j} + X^{-k}$.

NOTE 3 Ce polynôme G(X) est le polynôme optimal de 16 bits pour la détection d'erreurs en rafales sur les DLPDU de 300 octets ou moins lorsque les statistiques de la rafale d'erreurs ont une distribution de Poisson (comme cela est le cas habituel).

NOTE 4 Il convient que le reste R(x) soit 1110 0011 1001 0100 (X^{15} à X^0, respectivement) en l'absence d'erreurs.

5.2.5.1 Au niveau de la DLE d'envoi

Le message original (c'est-à-dire, la DLPDU sans une FCS), la FCS, et le mot de code du message composite (la DLPDU et la FCS concaténées) doivent être considérés comme des vecteurs(M(X), F(X), et D(X), de dimension respective *k*, *n* - *k*, et *n*, dans un champ d'extension sur GF(2). Si les bits de message sont $m_1 \dots m_k$ et les bits FCS sont $f_{n-k-1} \dots f_0$, où

$m_1 \dots m_8$	forment le premier octet envoyé,
$m_{8N-7} \dots m_{8N}$	forment le *N*e octet envoyé,
$f_7 \dots f_0$	forment te dernier octet envoyé, et
m_1	est envoyé par le(s) premier(s) symbole(s) PhL du message et f_0 est envoyé par le(s) dernier()symbole(s) PhL du message (sans compter les informations de verrouillage de trame PhL)

NOTE Cet ordre «tel qu'émis» est critique pour les propriétés de détection d'erreur de la FCS.

alors, le vecteur message M(X) doit être considéré comme étant

$$M(X) = m_1X^{k-1} + m_2X^{k-2} + \dots + m_{k-1}X^1 + m_k \quad (11)$$

et le vecteur message F(X) doit être considéré comme étant

$$\begin{aligned} F(X) &= f_{n-k-1}X^{n-k-1} + \dots + f_0 \quad (12) \\ &= f_{15}X^{15} + \dots + f_0 \end{aligned}$$

Le vecteur composite D(X), pour la DLPDU complète, doit être construit comme la concaténation des vecteurs message et FCS

$$\begin{aligned} D(X) &= M(X)\,X^{n-k} + F(X) \quad (13) \\ &= m_1X^{n-1} + m_2X^{n-2} + \dots + m_kX^{n-k} + f_{n-k-1}X^{n-k-1} + \dots + f_0 \\ &= m_1X^{n-1} + m_2X^{n-2} + \dots + m_kX^{16} + f_{15}X^{15} + \dots + f_0 \quad \text{(pour le cas de k = 15)} \end{aligned}$$

La DLPDU présentée à la PhL doit être constituée d'une séquence d'octets dans l'ordre spécifié.

Les bits de contrôle redondants f_{n-k-1} … f_0 de la FCS doivent être les coefficients du reste F(X), après la division par G(X), de L(X) $(X^k + 1)$ + M(X) X^{n-k}

où G(X) est le polynôme générateur de degré ***n-k*** pour les mots de code

$$G(X) = X^{n-k} + g_{n-k-1}X^{n-k-1} + \ldots + 1 \tag{14}$$

et L(X) est le polynôme de poids maximal (des uns partout) de degré ***n-k-1***

$$L(X) = \frac{X^{n-k}+1}{X+1} = X^{n-k-1} + X^{n-k-2} + \ldots + X + 1 \tag{15}$$

$$= X^{15} + X^{14} + X^{13} + X^{12} + \ldots + X^2 + X + 1 \qquad \text{(pour le cas de k = 15)}$$

C'est-à-dire,

$$F(X) = L(X)\,(X^k + 1) + M(X)\,X^{n-k} \;\; (\text{modulo } G(X)) \tag{16}$$

NOTE 1 Les termes L(X) sont inclus dans le calcul pour détecter la troncature ou extension de message initiale ou terminale en ajoutant un facteur dépendant de la longueur à la FCS.

NOTE 2 Comme mise en œuvre typique lorsque *n-k* = 16, le reste initial de la division est préréglé à des uns partout. Le train de bits de message émis est multiplié par X^{n-k} et divisé (modulo 2) par le polynôme générateur G(X), spécifié dans l'Équation (7). Le complément de uns du reste obtenu est émis comme étant la FCS de (n-k) bits, avec le coefficient de X^{n-k-1} émis le premier.

5.2.5.2 Au niveau de la DLE destinataire

La séquence d'octet indiquée par la PhE doit être concaténée dans la DLPDU et la FCS reçues et être considérée comme un vecteur V(X) de la dimension *u*

$$V(X) = v_1X^{u-1} + v_2X^{u-2} + \ldots + v_{u-1}X + v_u \tag{17}$$

NOTE 1 En raison d'erreurs, u peut différer de n, la dimension du vecteur code émis.

Un reste R(X) doit être calculé pour V(X), la DLPDU et la FCS reçues, par une méthode similaire à celle qui est utilisée par la DLE d'envoi (voir 5.2.5.1) dans le calcul de F(X).

$$R(X) = L(X)\,X^u + V(X)\,X^{n-k} \;\; (\text{modulo } G(X)) \tag{18}$$

$$= r_{n-k-1}X^{n-k-1} + \ldots + r_0$$

Définir E(X) comme étant le vecteur code d'erreur des différences additives (modulo 2) entre le vecteur code émis D(X) et le vecteur reçu V(X) résultant d'erreurs rencontrées (dans le fournisseur PhS et dans les ponts) entre les DLE expéditrices et destinataires.

$$E(X) = D(X) + V(X) \tag{19}$$

Si aucune erreur ne s'est produite, si bien que E(X) = 0, alors R(X) sera égal à un polynôme reste qui est constant et non nul.

$$R_{ok}(X) = L(X)\,X^{n-k} \;\; (\text{modulo } G(X)) \tag{20}$$

dont la valeur est indépendante de D(X). Malheureusement R(X) sera aussi égal à $R_{ok}(X)$ dans les cas où E(X) est un multiple exact non-zéro de G(X), auquel cas il y a des erreurs "indétectables". Dans tous les autres cas, R(X) ne sera pas égal à $R_{ok}(X)$; de telles DLPDU sont erronées et doivent être rejetées sans autre analyse.

NOTE 2 Comme mise en œuvre typique, le reste initial de la division est préréglé à des uns partout. Le train de bits reçus est multiplié par X^{n-k} et divisé (modulo 2) par le polynôme générateur G(X), spécifié dans l'Équation (7).

5.2.5.3 Modification au sein des ponts

Lors de la transmission d'une DLPDU de Type 1, il est parfois nécessaire pour un pont d'altérer un ou plusieurs sous-champs d'un champ de contrôle de trame de la DLPDU. Lorsqu'il apporte ces modifications, le pont peut être requis de modifier la FCS reçue afin de compenser les changements apportés à l'octet de contrôle de trame; dans ce cas, le pont ne rejette pas la FCS reçue et recalcule une nouvelle FCS après que le champ de contrôle de trame de la DLPDU a été altéré[2].

Lorsque la longueur de la DLPDU reçue, plus celle de son champ FCS, est de N octets, le pont peut compenser une modification du bit K dans le premier octet en calculant le résidu du polynôme

$$X^{8N+K-8} \ (\text{modulo} \ \ G(X)) \tag{21}$$

et en mettant ensuite à jour le champ FCS de la DLPDU en soumettant à une opération OU-EXCLUSIF le résidu calculé dans ce champ.

NOTE Lorsque le pont s'initialise, il peut précalculer les résidus de toutes les longueurs de DLPDU admissibles et les positions de bits nécessitant une altération, c'est-à-dire, pour toutes les valeurs de N comprises entre 3 et 272 et de K égales à 2. Ensuite, pour toute DLPDU, le pont a uniquement besoin d'appliquer à la FCS de la DLPDU ce résidu qui correspond au changement réellement apporté à l'octet de contrôle de la trame de la DLPDU.

5.2.6 Éléments de procédure pour le retard minimal entre des DLPDU

La DLE qui détient le jeton dominant doit lancer la transmission seulement après avoir fourni un retard au moins égal au retard maximum entre DLPDU, V(MID), durées d'octet, où le retard est mesuré comme spécifié en 4.7.1.12.

5.2.7 Éléments de procédure pour l'abandon d'un jeton par le détenteur de jeton dominant

Si la DLE qui détient le jeton dominant vient d'achever soit

a) la transmission d'une DLPDU, telle qu'indiquée par la réception d'une primitive "confirm" de Ph-DATA correspondant à la plus récente primitive "request" de Ph-DATA, qui doit avoir spécifié END-OF-DATA-AND-ACTIVITY, soit
b) la réception d'une DLPDU, telle qu'indiquée par la réception d'une primitive "indication" de Ph-Data spécifiant END-OF-DATA-AND-ACTIVITY ou END-OF-DATA ou END-OF-ACTIVITY

mais n'a pas encore lancé sa prochaine transmission, soit

c) parce qu'elle attend l'intervalle requis de retard minimal entre DLPDU, soit
d) parce qu'elle n'est pas prête à émettre en raison du retard dans la préparation de la prochaine transmission;

alors, si cette DLE est capable de recevoir dans cet intervalle et si une "indication" de Ph DATA (voir 4.4.4) rapportant DATA est reçue, la DLE doit abandonner le jeton.

5.2.8 Éléments commun de la procédure de surveillance de l'activité d'une liaison

Un certain nombre de types de DLPDU (CA, CD, ED, CT, RQ, PT, ES, TL) envoyées par un détenteur de jeton demandent une réponse immédiate. Les procédures communes s'appliquent aux DLE qui

a) ont besoin d'exercer une surveillance afin de détecter cette réponse immédiate;
b) ont besoin de déterminer
   – quand cesser la surveillance visant à détecter cette réponse;
   – si, oui ou non, cette réponse a eu lieu.

5.2.8.1 Surveillance pour détecter une réponse immédiate par la DLE initiatrice

Après avoir envoyé une DLPDU CA, CD, ED, CT ou RQ qui demande une réponse immédiate,

[2] D. R. Irvin, *Preserving the integrity of cyclic-redundancy checks when protected text is intentionally altered*, IBM Journal of Research and Development, Vol. 33, No. 6, novembre 1989, pp. 618-626.

a) lorsque la DLE expéditrice n'est pas également la DLE répondeuse, la DLE expéditrice doit
   1) surveiller la liaison locale pendant une période de immediate-response-recovery-delay intervalles de temps, à savoir V(IRRD) × V(ST) durées d'octet, attendant une réponse;
   2) puis entreprendre l'action appropriée en fonction du résultat de cette surveillance;
b) lorsque la DLE expéditrice est également la DLE répondeuse, elle doit entreprendre une action comme si la surveillance avait eu lieu et avait détecté avec succès la DLPDU de réponse immédiate demandée.

La procédure de surveillance est comme suit:

1) Si une "indication" Ph-Data (voir 4.4.4) rapportant DATA est reçue, alors la DLE doit
   i) arrêter toute surveillance;
   ii) attendre ensuite de recevoir l'indication de Ph-Data rapportant END-OF-DATA-AND-ACTIVITY ou END-OF-ACTIVITY;
2) Si 1) ne s'applique pas et la période de surveillance expire alors
   i) si le bus n'est pas actif à cet instant-là (c'est-à-dire que la dernière "indication" de Ph-Data reçue rapportait END-OF-ACTIVITY), la DLE doit arrêter toute surveillance.
   ii) si i) ne s'applique pas, ce qui implique que la liaison est active à cet instant-là (c'est-à-dire que la dernière "indication" de Ph-Data reçue rapportait START-OF-ACTIVITY), alors
      α) Pour les mises en œuvre qui n'étaient pas démontrables le 31 décembre 1995 ou avant, la DLE doit surveiller la liaison locale pendant une période d'un intervalle de temps supplémentaire, à savoir V (ST) durées d'octet, attendant une "indication" de Ph-Data
         A) Si une "indication" de Ph-Data rapportant DATA est reçue, alors la DLE doit poursuivre comme en 1).
         B) Si une "indication" de Ph-Data rapportant END-OF-ACTIVITY est reçue, et A) ne s'applique pas, alors la DLE doit arrêter toute surveillance.
         C) Si ni A) ni B) ne s'appliquent
         D) et la période de surveillance expire avant qu'une "indication" Ph-Data ne soit reçue, alors a DLE doit attendre la réception de l'indication de Ph-Data rapportant END-OF-DATA-AND-ACTIVITY ou END-OF-ACTIVITY.
      β) Les mises en œuvre qui étaient démontrables le 31 décembre 1995, ou avant, peuvent en variante juste poursuivre comme dans α).C) lorsque ni α).A) ni α).B) ne s'appliquent.

      NOTE Après que cette norme a atteint le statut ACDV, les réalisateurs ont été invités à développer des puces pour aider à l'évaluation de ce protocole complexe. En conséquence, il a été trouvé souhaitable d'améliorer les caractéristiques de réjection de bruit du processus de passage de jeton, et le résultat a été le texte 2)ii)α), qui nécessite des changements dans ces puces existantes. β) constitue l'antépénultième approche de réjection de bruit relative au premier ACDV (Projet approuvé pour Projet de comité pour vote) pour ces mises en œuvre antérieures et pour ces mises en œuvre seulement

À la fin de la surveillance, la DLE expéditrice doit agir en fonction du résultat de cette surveillance:

3) Si 2)i) ou 2)ii)α)B) s'appliquait, la DLE expéditrice doit
   – rapporter à la gestion locale l'échec à détecter une DLPDU;
   – si le sous-champ "utilisation finale de jeton" de la DLPDU CA, CD, ED, CT ou RQ initiatrice spécifiait NOT-FINAL, la DLE doit démarrer la prochaine transmission dans les limites de (immediate-response-recovery-delay plus un) intervalles de temps, à savoir (V(IRRD) + 1) × V(ST) durées d'octet, par rapport à la fin de transmission de la DLPDU CA, CD, ED, CT ou RQ en question.
4) Si 1) ou 2)ii) α)A) ou 2)ii) α)C) s'appliquait et l'activité de la liaison n'a pas donné de DLPDU, la DLE expéditrice doit

– rapporter à la gestion locale l'échec à détecter une réponse valide;
– si le sous-champ "'utilisation finale de jeton" de la DLPDU CA, CD, ED, CT ou RQ expéditrice spécifiait NOT-FINAL, la DLE doit démarrer la prochaine transmission dans les limites de maximum-reply-delay intervalles de temps, à savoir $V(MRD) \times V(ST)$ durées d'octet, par rapport au début de la période actuelle d'inactivité de la liaison;

5) Si
   – l'activité de liaison a donné une DLPDU;
   – si la DLPDU reçue n'était pas une DLPDU de réponse admissible,

   alors tout jeton détenu doit être abandonné et la gestion DL locale doit recevoir notification de l'événement.

   NOTE Ces rapports de gestion DL peuvent prendre la forme consistant à incrémenter un compteur d'erreurs de gestion de DL.

6) Si
   – l'activité de liaison a donné une DLPDU;
   – la DLPDU reçue était DLPDU de réponse admissible;
   – le sous-champ "utilisation finale de jeton" de la DLPDU CA, CD, ED, CT ou RQ initiatrice spécifiait NOT-FINAL,

   alors la DLE doit démarrer la prochaine transmission dans les limites de maximum-reply-delay intervalles de temps,
   à savoir $V(MRD) \times V(ST)$ durées d'octet, par rapport au début de la période courante d'inactivité de la liaison.

5.2.8.2 Surveillance pour détecter une réponse immédiate par la DLE LAS

Si le sous-champ "utilisation finale de jeton" de la DLPDU CA, CD, ED ou RQ reçue a la valeur FINAL, alors

a) si la DLE LAS n'est pas la DLE de réponse immédiate adressée, la DLE LAS doit appliquer les procédures de surveillance selon 5.2.8.1 1) et 5.2.8.1 2);
b) si la DLE LAS est également la DLE répondeuse, elle doit entreprendre une action comme si la surveillance avait eu lieu et avait détecté avec succès la DLPDU de réponse immédiate.

Après avoir achevé la surveillance a), ou après avoir envoyé la réponse immédiate demandée b), la DLE LAS doit

1) supposer que l'utilisation courante du jeton délégué a cessé et que le jeton programmeur est de nouveau dominant sur la liaison locale; et
2) traiter cette cessation comme si le jeton avait été renvoyé par une DLPDU RT (voir 6.17);
3) reprendre le fonctionnement actif comme le LAS.

5.2.8.3 Surveillance pour détecter une réponse immédiate comme abonné à une DLC

Une DLE qui

– reçoit une DLPDU CA, CD ou ED demandant une réponse immédiate;
– est un destinataire prévu de cette DLPDU de réponse immédiate demandée;
– n'est pas elle-même le répondeur adressé par la DLPDU CA, CD ou ED

doit déclencher un temporisateur avec une durée de immediate-response-recovery-delay intervalles de temps, à savoir
$V(IRRD) \times V(ST)$ durées d'octet:

a) Si une "indication" de Ph Data (4.4.4) rapportant DATA est reçue, la DLE doit arrêter toute surveillance et attendre la réception de l'indication de Ph-Data rapportant END-OF-DATA-AND-ACTIVITY ou END-OF-ACTIVITY et poursuivre comme en e).
b) Si la période de surveillance expire et le e bus n'est pas actif à la fin de cette période de surveillance, la DLE doit arrêter toute surveillance et poursuivre comme en d).
c) Si la période de surveillance expire et le bus est actif à la fin de cette période de surveillance, la DLE doit attendre la réception de l'indication de Ph Data rapportant END-OF-DATA-AND-ACTIVITY ou END-OF-ACTIVITY, et ensuite poursuivre comme en d) ou e), selon le cas.
d) Si la surveillance n'a pas donné de DLPDU comme en b) ou éventuellement en c), la DLE doit invalider V(RA).
e) Si l'activité de la liaison a donné une DLPDU comme en a) ou éventuellement en c), et si la DLPDU reçue était une DLPDU DT dont l'adresse de DL source est implicitement ou explicitement l'adresse de DLCEP de destination spécifiée par la DLPDU CA, CD ou ED, alors l'adresse enregistrée V(RA) doit être supposée être l'adresse source impliquée de cette DLPDU CA, CD ou ED. Autrement l'adresse enregistrée V(RA) doit être invalidée.

5.2.9 Fonctionnement alterné bidirectionnel (semi-duplex)

Il n'y a aucune exigence qu'une DLE expéditrice reçoive des indications de PH-DATA pendant qu'elle émet une DLPDU; mais lorsque de telles indications ne sont pas recevables (c'est-à-dire lorsque la DLE fonctionne en mode alterné bidirectionnel ou semi-duplex), une DLE expéditrice doit traiter chaque DLPDU émise comme si elle avait été reçu en même temps par cette DLE.

6 Structure, codage et éléments de procédure spécifiques à une DLPDU

NOTE 1 Les éléments de procédure dans le présent article correspondent généralement au sous-niveau de DLL le plus bas, tel que spécifié en 4.1.1.

L'Article 6 définit la structure, le contenu et le codage de chaque type et format de DLPDU, et spécifie les éléments de procédure pour le type et le format de DLPDU en question. (Le Tableau 10 récapitule leur structure.)

Dans chaque paragraphe, la structure, le contenu et le codage de la DLPDU sont décrits en premier lieu. Puis, les aspects relatifs aux utilisateurs de DLS expéditeurs et destinataires et à leurs DLE sont abordés, suivis des aspects, le cas échéant, qui sont uniques aux fonctions de mise en mémoire et acheminement ultérieur de ponts (DLE de relais).

Dans tout l'Article 6, chaque fois qu'une action conditionnelle est spécifiée et la condition habilitante spécifiée ne se produit pas, l'action correspondante ne se produit pas non plus. L'effet net produit est que les événements qui ne satisfont à aucune des conditions habilitantes spécifiées dans une procédure de service n'auront aucune conséquence en ce qui concerne cette procédure de service spécifique.

Une DLE qui n'est pas ONLINE par rapport à la liaison locale, mais qui tente de changer son état de DL pour le faire passer à ONLINE, ne peut envoyer une DLPDU qu'en réponse à une DLPDU PN adressée aux fonctions de prise en charge de DL de la DLE, telles que spécifiées en 6.13.4, 6.14.3 et 10.2.1. Au cours de cet intervalle, le protocole de DL spécifie la réponse de la DLE aux DLPDU PN, PT et DT reçues adressées aux fonctions de prise en charge de DL de la DLE. Le comportement de la DLE après avoir changé son état à ONLINE ne doit dépendre en aucune façon de la réception, avant qu'elle ne soit passée à ONLINE, d'autres DLPDU que celles (PN, PT, DT) spécifiées ci-dessus.

NOTE 2 La réception d'une DLPDU sans effets secondaires observables relève purement d'une initiative locale et se situe donc hors du domaine d'application de la normalisation.

Une DLE qui est ONLINE par rapport à la liaison locale ne peut envoyer de DLPDU que lorsqu'elle détient le jeton dominant sur la liaison locale. L'ensemble des DLPDU qui peuvent être envoyées pendant la détention du jeton dominant dépendent de la classe du jeton dominant (voir la récapitulation dans Tableau 11):

a) Si le jeton dominant est un jeton programmeur, les classes autorisées de DLPDU sont CA, CD, DC, DT, EC, ED, ES, PN, PT, RC, RQ, TD, TL, WK et IDLE.
b) Si le jeton dominant est un jeton délégué, les classes autorisées de DLPDU CA, CD, CT, DC, DT, EC, ED, RC, RI, RQ, RT, WK et IDLE.
c) Si le jeton dominant est un jeton de réponse, les classes autorisées de DLPDU sont
   1) DT et SR en réponse à une DLPDU CA, CD ou ED;
   2) TD en réponse à une DLPDU CT;
   3) PR en réponse à une DLPDU PN;
   4) RR en réponse à une DLPDU RQ.
d) Les jetons ne sont créés que par les DLPDU suivantes:
   1) Un jeton programmeur n'est créé que par une DLPDU CL;
   2) Un jeton délégué n'est créé que par des DLPDU ES et PT;
   3) Un jeton de réponse n'est créé que par des DLPDU CA, CD, CT, ED, PN, RQ et TL.

Tableau 10 – Structure sommaire des DLPDU

Classe de DLPDU	Contrôle de trame	Adresses de DL			Paramètres	Données utilisateur
		Destination	Source	2e source		
EC 1	1111 LF00	[HL.]N.S	[HL.]N.S	[HL.]N.S	EC-p	o-DLSDU
EC 2	1110 LF00		[HL.]N.S	[HL.]N.S	EC-p	o-DLSDU
DC 1	0111 LF00	[HL.]N.S	[HL.]N.S		DC-p	o-DLSDU
DC 2	0110 LF00		[HL.]N.S		DC-p	o-DLSDU
RC 1	0111 LFPP	[HL.]N.S	[HL.]N.S		RC-p	o-DLSDU
RC 2	0110 LFPP		[HL.]N.S		RC-p	o-DLSDU
CA 1	1110 LFPP	[HL.]N.S	[HL.]N.S		SD-p	o-pDLSDU
CA 2	1010 LFPP	[HL.]N.S	—		SD-p	o-pDLSDU
CD 1	1111 LFPP	[HL.]N.S	[HL.]N.S		o-SD-p	—
CD 2	1011 LFPP	[HL.]N.S	—		o-SD-p	—
ED 1	1100 LFPP	[HL.]N.S	[HL.]N.S		SD-p	pDLSDU
ED 2	1000 LFPP	[HL.]N.S	—		SD-p	pDLSDU
DT 1	1101 LFPP	[HL.]N.S	[HL.]N.S		SD-p	o-pDLSDU
DT 2	1001 LFPP	[HL.]N.S	—		SD-p	o-pDLSDU
DT 3	0101 LFPP		[HL.]N.S		SD-p	o-pDLSDU
DT 4	1001 0F00	[PSA]	—		SD-p	o-pDLSDU
DT 5	0101 0F00		[PDA]		SD-p	o-pDLSDU
SR	0001 0F11	[PSA]	N		o-SR-p	—
TC	0001 0F00	—	—		—	—
TD	0001 0F01	—	N		TD-p	—
RQ	1100 0F00	N.0	N.0		RQ-p	—
RR	1101 0F00	N.0	N.0		RR-p	—
PN	0010 0110	N	—		PN-p	—
PR	0010 0111	—	—		—	SPDU
Pt	0011 0FPP	N	—		DD-p	—
ES	1000 LF00	[HL.]N.S	—		DD-p	—
RT	0011 0100	—	[DTH]		—	—
RI	0010 0000	—	[DTH]		DD-p	—
CL	0000 0001	—	N		—	—
TL	0000 0110	N	—		—	SPDU
WK	0000 0000	N	—		—	—
IDLE	0001 0F10	—	—		—	o-DLSDU

où

L indique la longueur des adresses de DL associées (0 = SHORT, 1 = LONG).

F indique l'utilisation finale d'un jeton ou qu'il convient qu'une séquence soit terminée plutôt que redémarrée.

PP spécifie la priorité de la DLPDU et de tout jeton transmis.

l'ombrage indique un champ logiquement inexistant.

— indique un champ logiquement existant dont le contenu doit être vide.

[HL.]N.S est une adresse de DL LONG de quatre octets (HLNS) lorsque L = 1 ou une adresse de DL SHORT de deux octets (NS) avec HL = 00 implicite lorsque L = 0.

N est une adresse de DL NODE d'un octet.

N.0 est la forme adresse de DL SHORT de deux octets d'une adresse de DL NODE d'un octet.

[PDA] est l'adresse de DL implicite égale à l'adresse de DL de destination explicite de la DLPDU immédiatement antérieure sur la liaison, qui doit avoir été une DLPDU CA, CD ou ED. Il s'agit d'un champ logiquement existant dont le contenu réel doit être vide.

[PSA] est l'adresse de DL implicite égale à l'adresse de DL source implicite ou explicite de la DLPDU immédiatement antérieure sur la liaison. Il s'agit d'un champ logiquement existant dont le contenu réel doit être vide.

[DTH] est l'adresse de DL implicite égale à l'adresse de DL de destination explicite de la plus récente DLPDU PT ou ES sur la liaison – à savoir la DLPDU qui a délégué le jeton retourné par la DLPDU RT ou RI. Il s'agit d'un champ logiquement existant dont le contenu réel doit être vide.

o- indique le contenu de champ facultatif.

xx-p indique les paramètres des DLPDU de classe xx.

DLSDU est une DL-service-data-unit (Unité de données de service liaison de données).

pDLSDU est une DLSDU complète ou partielle.

SPDU est une Support Protocol Data Unit (unité de données de protocole de prise en charge).

Tableau 11 – Restrictions de DLPDU en fonction du jeton dominant

Classe de DLPDU	Type de jeton créé	Peut être envoyée pendant l'utilisation d'un jeton programmeur	Peut être envoyée pendant l'utilisation d'un jeton délégué	Peut être envoyée en réponse à
EC	—	Y	Y	—
DC	—	Y	Y	—
RC	—	Y	Y	—
CA	Reply (Réponse)	Y	Y	—
CD	Reply	Y	Y	—
ED	Reply	Y	Y	—
DT	—	Y	Y	CA, CD, ED
SR	—	N	N	CA, CD, ED, TL
CT	Reply	N	Y	—
TD	—	Y	N	CT
RQ	Reply	Y	Y	—
RR	—	N	N	RQ
PN	Reply	Y	N	—
PR	—	N	N	PN
PT	Delegated (Délégué)	Y	N	—
ES	Delegated	Y	N	—
RT	—	N	Y	—
RI	—	N	Y	—
CL	—	N	N	—
TL	Scheduler (Programmeur)	Y	N	—
WK	—	Y	Y	—
IDLE	—	Y	Y	—
où – signifie qu'aucun jeton n'a été créé.				

6.1 DLPDU "Establish connection" ("Établir une connexion" (EC))

Une DLPDU ESTABLISH CONNECTION (EC) est utilisée pour établir une DLC d'homologue entre deux utilisateurs de DLS, une DLC à plusieurs homologues entre un utilisateur de DLS éditeur et des utilisateurs DLS abonnés.

6.1.1 Structure de la DLPDU EC

Tableau 12 – Structure des DLPDU EC

Format	Contrôle de trame	Adresse de destination	Adresse source	2e adresse source	Paramètres	Données d'utilisateur
1L	1111 1F00	HL.N.S	HL.N.S	HL.N.S	EC-p	o-DLSDU
1s	1111 0F00	N.S	N.S	N.S	EC-p	o-DLSDU
2L	1110 1F00		HL.N.S	HL.N.S	EC-p	o-DLSDU
2s	1110 0F00		N.S	N.S	EC-p	o-DLSDU

6.1.1.1 Champ "Frame control" (contrôle de trame)

Le champ "contrôle de trame" doit spécifier

a) la fonction de la DLPDU;
b) la priorité implicite de la DLPDU, qui est TIME-AVAILABLE;
c) la longueur, le nombre et le type d'adresses de DLPDU;
d) si, oui ou non, la transmission de la DLPDU courante met fin à l'utilisation d'un jeton délégué.

6.1.1.2 Champ "Address" (adresse)

Le champ "adresse" doit être constitué soit

a) d'une adresse de DL de destination explicite et de deux adresses de DL source explicites, dans cet ordre, pour les formats 1L et 1S, soit
b) de seulement deux adresses de DL source explicites, pour les formats 2L et 2S.

Pour les formats 1L et 2L, toutes les adresses doivent être LONG; pour les formats 1S et 2S, toutes les adresses doivent être SHORT;

6.1.1.3 Champ "Parameters" (paramètres)

Le champ "establish-connection parameters" (EC-parameters (paramètres EC) spécifie les attributs QoS proposés ou sélectionnés de la DLC, y compris la version du protocole de DL utilisé. Ce champ doit être structuré et codé tel que décrit en 7.1.

6.1.1.4 Champ "User data" (données d'utilisateur)

Le champ "données d'utilisateur" doit être constitué d'une seule DLSDU facultative dont la taille maximale est limitée à 128 octets.

6.1.2 Contenu de la DLPDU EC

Le champ "contrôle de trame" doit être codé comme spécifié dans le Tableau 12.

Pour les formats 1L et 1S,

a) la première adresse de DL doit être une adresse de DL(SAP) ou une adresse de DLCEP;
b) la deuxième adresse de DL doit être une adresse de DLCEP ou une adresse de DLSAP;
c) la troisième adresse DL doit être une adresse de DLSAP du DLSAP associé à cette deuxième adresse de DL.

Pour les formats 2L et 2S, la première adresse DL doit être une adresse de DLCEP éditeur et la deuxième adresse de DL doit être une adresse de DLSAP du DLSAP associé à cette première adresse de DL.

Les paramètres EC doivent spécifier les attributs QoS proposés ou sélectionnés de la DLC, y compris la version du protocole DL utilisé. Le contenu de ce champ doit être tel que décrit en 7.1.

6.1.3 Envoi de la DLPDU EC

Une DLPDU EC peut être envoyée sur la liaison lorsque la DLE expéditrice détient un jeton programmeur ou un jeton délégué qui est le jeton dominant sur la liaison locale, et lorsque la durée allouée restante d'utilisation du jeton, C(RD), permet l'achèvement de la transmission de la DLPDU EC avant l'expiration du jeton.

Chaque adresse de DL dans la DLPDU doit être délocalisée (voir 5.2.2) avant transmission.

Si la DLE détient un jeton délégué et aucune utilisation supplémentaire de ce jeton après l'envoi de cette DLPDU n'est nécessaire à cet instant-là, la DLE peut mettre à la valeur FINAL le sous-champ "utilisation finale du jeton" de la DLPDU EC; sinon, ce sous-champ doit avoir la valeur NOT-FINAL.

6.1.4 Réception de la DLPDU EC

Chaque adresse de DL dans la DLPDU doit être délocalisée (voir 5.2.2) à la suite de la réception.

Une DLPDU EC reçue doit être traitée comme suit par la DLE destinataire.

6.1.4.1 Actions requises de toutes les DLE

Si la première adresse de DL spécifiée par la DLPDU désigne une adresse de DL(SAP) active ou un DLCEP actif de la DLE, la DLPDU reçue doit être transmise aux fonctions de niveau supérieur de la DLE (voir 8.3) pour un traitement ultérieur.

6.1.4.2 Actions supplémentaires requises d'une DLE de classe Link-Master

Aucune.

6.1.4.3 Actions supplémentaires requises d'une DLE de classe Bridge

Les considérations suivantes s'appliquent:

a) Étant donné que chaque DLE de classe Bridge a une capacité de maître de liaison, toutes les actions spécifiées en 6.1.4.2 s'appliquent également à une DLE de la classe Bridge.
b) Si la première adresse DL spécifiée dans la DLPDU est une adresse qu'il convient que le pont transmette et le pont a été capable de mettre la DLPDU en tampon sans erreur, alors la DLPDU reçue doit être, au besoin, transmise avec la modification du champ "contrôle de trame" dans la DLPDU transmise (voir 5.2.5.3).
c) Le pont doit tenter de mettre à jour son entrée de table de routage pour l'adresse de DLCEP source et l'adresse de DLSAP source spécifiées dans la DLPDU afin de refléter le port du pont à partir duquel la DLPDU a été reçue.
d) Si les paramètres "Establish-Connection" de la DLPDU indiquent que le(s) destinataire(s) de la DLPDU sera/seront l'/les abonné(s) d'une DLC à plusieurs homologues, le pont doit tenter de mettre à jour son entrée de table de routage pour l'adresse de DLCEP source spécifiée dans la DLPDU en ajoutant le(s) port(s) de pont vers le(s)quel(s) la DLPDU est transmise à l'ensemble des ports de DLCEP collecteur associés à cette adresse de DLCEP source.

NOTE Cette dernière procédure n'a de sens que pour les formats 1L et 1S de la DLPDU EC.

6.1.4.4 Actions supplémentaires requises de la DLE LAS courante

Si le sous-champ "utilisation finale de jeton" de la DLPDU reçue a la valeur FINAL, la DLE LAS doit

a) supposer que l'utilisation courante du jeton délégué est terminée comme si le jeton avait été retournée par une DLPDU RT (voir 6.17);
b) supposer que le jeton programmeur est de nouveau dominant sur la liaison locale et reprendre le fonctionnement actif comme LAS.

6.2 DLPDU "Disconnect connection" (déconnecter la connexion (DC))

Une DLPDU DISCONNECT CONNECTION (DC) est utilisée pour déconnecter une DLC existante ou proposée.

6.2.1 Structure de la DLPDU DC

Tableau 13 – Structure des DLPDU DC

Format	Contrôle de trame	Adresse de destination	Adresse source	Paramètres	Données utilisateur
1L	0111 1F00	HL.N.S	HL.N.S	DC-p	o-DLSDU
1s	0111 0F00	N.S	N.S	DC-p	o-DLSDU
2L	0110 1F00		HL.N.S	DC-p	o-DLSDU
2s	0110 0F00		N.S	DC-p	o-DLSDU

6.2.1.1 Champ "Frame control" (contrôle de trame)

Le champ "contrôle de trame" doit spécifier

a) la fonction de la DLPDU;
b) la priorité implicite de la DLPDU, qui est TIME-AVAILABLE;
c) la longueur, le nombre et le type d'adresses de DLPDU;
d) si, oui ou non, la transmission de la DLPDU courante met fin à l'utilisation d'un jeton délégué.

6.2.1.2 Champ "Address" (adresse)

Le champ "adresse" doit être constitué soit

a) d'une adresse de DL de destination explicite et d'une adresse de DL source explicite, dans cet ordre, pour les formats 1L et 1S, soit
b) de seulement une adresse de DL source explicite, pour les formats 2L et 2S.

Pour les formats 1L et 2L, toutes les adresses doivent être LONG; pour les formats 1S et 2S, toutes les adresses doivent être SHORT.

6.2.1.3 Champ "Parameters" (paramètres)

Le champ "disconnect-connexion parameters" (paramètres de déconnexion de connexion (paramètres DC)) doit spécifier la version du protocole de DL utilisé, l'action souhaitée et la raison de cette action. Ce champ doit être structuré et codé tel que décrit en 7.27.2.

6.2.1.4 Champ "User data" (données d'utilisateur)

Le champ "données d'utilisateur" doit être constitué d'une seule DLSDU facultative dont la taille maximale est limitée à 128 octets.

6.2.2 Contenu de la DLPDU DC

Le champ "contrôle de trame" doit être codé comme spécifié dans le Tableau 13.

Pour les formats 1L et 1S, soit

a) les deux adresses de DL doivent être des adresses de DLCEP homologues, soit
b) la première adresse de DL doit être une adresse de DL(SAP) et la deuxième doit être une adresse de DLCEP, soit
c) la première adresse de DL doit être une adresse de DLCEP et la deuxième doit être une adresse de DL(SAP), soit

Pour les formats 2L et 2S, la seule adresse de DL doit être une adresse de DLCEP éditeur.

Les paramètres DC doivent spécifier la version du protocole DL utilisé, l'action souhaitée et sa raison, ainsi que d'autres informations. Le contenu de ce champ doit être codé tel que décrit en 7.27.2.

6.2.3 Envoi de la DLPDU DC

Une DLPDU DC peut être envoyée sur la liaison lorsque la DLE expéditrice détient un jeton programmeur ou un jeton délégué qui est le jeton dominant sur la liaison locale, et lorsque la durée allouée restante d'utilisation du jeton, C(RD), permet l'achèvement de la transmission de la DLPDU avant l'expiration du jeton.

Chaque adresse de DL dans la DLPDU doit être délocalisée (voir 5.2.2) avant transmission.

Si la DLE détient un jeton délégué et aucune utilisation supplémentaire de ce jeton après l'envoi de cette DLPDU n'est nécessaire à cet instant-là, la DLE peut mettre à la valeur FINAL le sous-champ "utilisation finale du jeton" de la DLPDU DC; sinon, ce sous-champ doit avoir la valeur NOT-FINAL.

6.2.4 Réception de la DLPDU DC

Chaque adresse de DL dans la DLPDU doit être délocalisée (voir 5.2.2) à la suite de la réception.

Une DLPDU DC reçue doit être traitée comme suit par la DLE destinataire.

6.2.4.1 Actions requises de toutes les DLE

Si la première adresse de DL spécifiée par la DLPDU désigne une adresse de DLSAP active ou un DLCEP actif de la DLE destinataire, la DLPDU reçue doit être transmise aux fonctions de niveau supérieur de la DLE (voir 8.2.1.7) pour un traitement ultérieur.

6.2.4.2 Actions supplémentaires requises d'une DLE de classe Link-Master

Aucune.

6.2.4.3 Actions supplémentaires requises d'une DLE de classe Bridge

Les considérations suivantes s'appliquent:

a) Étant donné que chaque DLE de classe Bridge a une capacité de maître de liaison, toutes les actions spécifiées en 6.2.4.2 s'appliquent également à une DLE de classe Bridge.
b) Si la première adresse DL spécifiée dans la DLPDU est une adresse qu'il convient que le pont transmette et le pont a été capable de mettre la DLPDU en tampon sans erreur, alors

la DLPDU reçue doit être, au besoin, transmise avec la modification du champ "contrôle de trame" dans la DLPDU transmise (voir 5.2.5.3).

c) Si les paramètres de déconnexion de connexion de la DLPDU indiquent qu'une réponse n'est pas demandée, le pont peut tenter de supprimer les entrées de table de routage pour toutes les adresses DLCEP explicites spécifiées dans la DLPDU.

6.2.4.4 Actions supplémentaires requises de la DLE LAS courante

Si le sous-champ "utilisation finale de jeton" de la DLPDU reçue a la valeur FINAL, la DLE LAS doit

a) supposer que l'utilisation courante du jeton délégué est terminée comme si le jeton avait été retournée par une DLPDU RT (voir 6.17);
b) supposer que le jeton programmeur est de nouveau dominant sur la liaison locale et reprendre le fonctionnement actif comme LAS.

6.3 DLPDU "Reset connection" (Réinitialiser une connexion (RC))

Une DLPDU RESET CONNECTION (RC) est utilisée pour réinitialiser une DLC existante ou proposée.

6.3.1 Structure de la DLPDU RC

Tableau 14 – Structure des DLPDU RC

Format	Frame control (Contrôle de trame)	Adresse de destination	Adresse source	Paramètres	User data (données d'utilisateur)
1L	0111 1FPP	HL.N.S	HL.N.S	RC-p	o-DLSDU
1s	0111 0FPP	N.S	N.S	RC-p	o-DLSDU
2L	0110 1FPP		HL.N.S	RC-p	o-DLSDU
2s	0110 0FPP		N.S	RC-p	o-DLSDU

6.3.1.1 Champ "Frame control" (contrôle de trame)

Le champ "contrôle de trame" doit spécifier

a) la fonction de la DLPDU;
b) la priorité de la DLPDU;
c) la longueur, le nombre et le type d'adresses de DLPDU;
d) si, oui ou non, la transmission de la DLPDU courante met fin à l'utilisation d'un jeton délégué.

6.3.1.2 Champ "Address" (adresse)

Le champ "adresse" doit être constitué soit

a) d'une adresse de DL de destination explicite et d'une adresse de DL source explicite, dans cet ordre, pour les formats 1L et 1S, soit
b) de seulement une adresse de DL source explicite, pour les formats 2L et 2S.

Pour les formats 1L et 2L, toutes les adresses doivent être LONG; pour les formats 1S et 2S, toutes les adresses doivent être SHORT;

6.3.1.3 Champ "Parameters" (paramètres)

Le champ "reset -connexion parameters" (paramètres de réinitialisation de connexion (paramètres RC)) doit spécifier la version du protocole de DL utilisé, l'action souhaitée et la raison, ainsi que d'autres informations. Ce champ doit être structuré et codé tel que décrit en 7.3.

6.3.1.4 Champ "User data" (données d'utilisateur)

Le champ "données utilisateur" doit consister en une seule DLSDU facultative dont la taille maximale est limitée à la taille maximale des données utilisateur de DLS autorisée pour une DLPDU de la priorité spécifiée en 6.3.1.1b).

6.3.2 Contenu de la DLPDU RC

Le champ "contrôle de trame" doit être codé comme spécifié dans le Tableau 14.

Pour les formats 1L et 1S, les deux adresses de DL doivent être des adresses de DLCEP homologues. Pour les formats 2L et 2S, la seule adresse de DL doit être une adresse de DLCEP éditeur.

Les paramètres RC doivent spécifier la version du protocole DL utilisé, l'action souhaitée et sa raison, ainsi que d'autres informations. Le contenu de ce champ doit être codé tel que décrit en 7.3.

6.3.3 Envoi de la DLPDU RC

Une DLPDU RC peut être envoyée sur la liaison lorsque la DLE expéditrice détient un jeton programmeur ou un jeton délégué qui est le jeton dominant sur la liaison locale, et lorsque la durée allouée restante d'utilisation du jeton, C(RD), permet l'achèvement de la transmission de la DLPDU EC avant l'expiration du jeton.

Chaque adresse de DL dans la DLPDU doit être délocalisée (voir 5.2.2) avant transmission.

Si la DLE détient un jeton délégué et aucune utilisation supplémentaire de ce jeton après l'envoi de cette DLPDU n'est nécessaire à cet instant-là, la DLE peut mettre à la valeur FINAL le sous-champ "utilisation finale du jeton" de la DLPDU RC; sinon, ce sous-champ doit avoir la valeur NOT-FINAL.

6.3.4 Réception de la DLPDU RC

Chaque adresse de DL dans la DLPDU doit être délocalisée (voir 5.2.2) à la suite de la réception.

Une DLPDU RC reçue doit être traitée comme suit par la DLE destinataire.

6.3.4.1 Actions requises de toutes les DLE

Si la première adresse de DL spécifiée par la DLPDU désigne un DLCEP actif de la DLE destinataire, la DLPDU reçue doit être transmise aux fonctions de niveau supérieur de la DLE (voir 8.2) pour un traitement ultérieur.

6.3.4.2 Actions supplémentaires requises d'une DLE de classe Link-Master

Aucune.

6.3.4.3 Actions supplémentaires requises d'une DLE de classe Bridge

Les considérations suivantes s'appliquent:

a) Étant donné que chaque DLE de classe Bridge a une capacité de maître de liaison, toutes les actions spécifiées en 6.3.4.2 s'appliquent également à une DLE de la classe Bridge.

b) Si la DLE destinataire est un pont et la première adresse DL spécifiée dans la DLPDU est une adresse qu'il convient que le pont transmette et le pont a été capable de mettre la DLPDU en tampon sans erreur, alors la DLPDU reçue doit être, au besoin, transmise avec la modification du champ "contrôle de trame" dans la DLPDU transmise (voir 5.2.5.3).

6.3.4.4 Actions supplémentaires requises de la DLE LAS courante

Si le sous-champ "utilisation finale de jeton" de la DLPDU reçue a la valeur FINAL, la DLE LAS doit

a) supposer que l'utilisation courante du jeton délégué est terminée comme si le jeton avait été retournée par une DLPDU RT (voir 6.17);

b) supposer que le jeton programmeur est de nouveau dominant sur la liaison locale et reprendre le fonctionnement actif comme LAS.

6.4 DLPDU "Compel acknowledgement" (Forcer un acquittement (CA))

Une DLPDU COMPEL ACKNOWLEDGMENT (Forcer un acquittement (CA)) est utilisée

a) pour transférer (ou retransférer) une quantité limitée de données utilisateur transparentes d'un utilisateur de DLS demandeur vers un autre utilisateur de DLS;

b) pour demander que le transfert soit acquitté dès que possible,

sans exiger que la transaction (demande et acquittement) s'effectue dans le contexte d'une DLC. Elle est également utilisée à des fins similaires dans le contexte d'une DLC, et pour aider à la synchronisation des DLCEP et de leurs utilisateurs DLS.

Une DLPDU CA crée et transmet un jeton de réponse à la DLE destinataire adressée, qui, à la suite de la réception, devient le jeton dominant sur sa liaison. Une DLPDU CA exige une réponse immédiate soit d'une DLPDU DATA (DT) sans données utilisateur de DLS, soit d'une DLPDU STATUS RESPONSE (SR). Si aucune DLPDU n'est reçue en réponse, la transaction est V(MRC) fois au maximum.

6.4.1 Structure des DLPDU CA

Tableau 15 – Structure des DLPDU CA

Format	Frame control (Contrôle de trame)	Adresse de destination	Adresse source	Paramètres	User data (données d'utilisateur)
1L	1110 1FPP	HL.N.S	HL.N.S	SD-p	o-pDLSDU
1s	1110 0FPP	N.S	N.S	SD-p	o-pDLSDU
2L	1010 1FPP	HL.N.S	—	SD-p	o-pDLSDU
2s	1010 0FPP	N.S	—	SD-p	o-pDLSDU

6.4.1.1 Champ "Frame control" (contrôle de trame)

Le champ "contrôle de trame" doit spécifier

a) la fonction de la DLPDU, y compris qu'une réponse immédiate est demandée;

b) la priorité de la transaction et de la DLPDU;

c) la longueur, le nombre et le type d'adresses de DLPDU;

d) si, oui ou non, la transmission de la DLPDU courante, et soit sa réponse immédiate escomptée, soit la temporisation de repos de liaison appropriée, mettent fin à l'utilisation d'un jeton délégué.

NOTE Ce champ a nécessairement la valeur NOT-FINAL lorsqu'une répétition immédiate de tentative de la transaction courante est possible. Seule une transaction qui est garantie ne pas avoir besoin d'une répétition immédiate de tentative peut avoir la valeur FINAL spécifiée dans sa DLPDU associée.

6.4.1.2 Champ "Address" (adresse)

Le champ "adresse" doit être constitué soit

a) d'une adresse de DL de destination explicite et d'une adresse de DL source explicite, dans cet ordre, pour les formats 1L et 1S, soit

b) de seulement une adresse de DL destination explicite, pour les formats 2L et 2S.

Pour les formats 1L et 2L, toutes les adresses doivent être LONG; pour les formats 1S et 2S, toutes les adresses doivent être SHORT;

6.4.1.3 Champ "Parameters" (paramètres)

Le champ "status-data-parameters" (paramètres de données de statut (paramètres SD)) spécifie les informations appropriées à l'adresse de DL de destination associée:

a) Si cette adresse de DL est une adresse de DLSAP, alors les paramètres SD spécifient
   1) un id de transaction utilisé par la DLE d'origine pour corréler à la demande d'origine une réponse différée retournée;
   2) une priorité de DLSDU utilisée pour acheminer la priorité réelle de la DLSDU d'accompagnement vers la DLE de réponse, ou qu'il n'y a pas de DLSDU d'accompagnement.

   Ce champ doit être structuré et codé tel que décrit en 7.4.1.

b) Si cette adresse de DL est une adresse de DLCEP, les paramètres SD spécifient des informations d'état pour le DLCEP adressé. Ce champ doit être structuré et codé tel que décrit en 7.4.2.

   NOTE La taille et la structure de ce champ dépendent des attributs QoS associés au DLCEP adressé par l'adresse de DL de destination spécifiée dans cette DLPDU, et sont déterminées au cours de l'établissement du DLCEP.

6.4.1.4 Champ "User data" (données d'utilisateur)

La taille et le contenu du champ "données utilisateur" sont limités par l'adresse de DL de destination associée;

a) Si l'adresse de DL est une adresse de DLSAP, le champ "données utilisateur" doit être constitué de données utilisateur de DLS dont la taille maximale est limitée à la plus petite des tailles maximales de données utilisateur de DLS autorisées pour une DLPDU des priorités spécifiées en 6.4.1.1b) et en 6.4.1.3a)2), et ne doit pas être vide.

b) Si cette adresse de DL est une adresse de DLCEP, le champ "données utilisateur" doit être constitué des données utilisateur de DLS dont la taille maximale est limitée à la plus petite parmi
   1) la taille maximale de données utilisateur de DLS autorisée pour une DLPDU de la priorité spécifiée en 6.4.1.1b);
   2) la taille de DLSDU maximale négociée sur la DLC pour la transmission de données vers ce DLCEP,

   et ce champ peut être vide.

6.4.2 Contenu de la DLPDU CA

Le champ "contrôle de trame" doit être codé comme spécifié dans le Tableau 15.

Soit les adresses de DL doivent être

a) deux adresses de DLSAP explicites, soit

b) une adresse de DLCEP explicite et une seconde adresse de DLCEP explicite ou implicite, soit

c) une adresse de DLCEP explicite, suivie d'une adresse de DLSAP explicite ou implicite.

6.4.2.1 Contenu de la DLPDU CA lors de la spécification d'une adresse de DLSAP de destination

Lorsque la première adresse de DL est une adresse de DLSAP comme en 6.4.2a), alors

a) si le format de DLPDU est le format 1L ou 1S, alors
   1) cette DLPDU est utilisée pour mettre en œuvre le service de transfert des unitdata avec confirmation de DLE distante;
   2) le rôle de DL(SAP) pour l'adresse de DLSAP de destination doit être BASIC;
   3) la seconde adresse doit être présente, doit être une adresse de DLSAP, et le rôle de DL(SAP) pour cette adresse DLSAP doit être BASIC;
   4) le champ "paramètres SD" doit spécifier une priorité de DLSDU et un id de transaction utilisés par la DLE d'origine pour corréler à la demande d'origine une réponse différée renvoyée, où
      i) le contenu de ce champ doit être tel que décrit en 7.4;
      ii) la priorité de la DLSDU doit être la priorité des données utilisateur d'accompagnement et doit être la même que la priorité de la DLPDU spécifiée en 6.4.1.1b);
   5) les données utilisateur doivent être une DLSDU unique dont la taille est limitée à la taille maximale pour la priorité spécifiée en 6.4.1.4a), et ne doivent pas être vides;

b) aucun autre format de DLPDU ne peut être utilisé.

6.4.2.2 Contenu de la DLPDU CA lors de la spécification d'une adresse de DLCEP de destination

Lorsque la première adresse est une adresse de DLCEP comme en 6.4.2b) et 6.4.2c), alors

a) cette DLPDU peut transporter une DLSDU unique ou partielle
   – d'un DLCEP homologue vers son DLCEP homologue correspondant, ou
   – d'un DLCEP abonné vers son DLCEP éditeur correspondant,

   doit demander des informations d'état issues du DLCEP adressé, et ne doit pas autoriser d'inclure des données utilisateur de DLS dans la DLPDU de réponse;

b) la seconde adresse, si elle est présente,
   – doit être l'adresse de DLCEP homologue de la même DLC que l'adresse de DLCEP homologue de destination, ou
   – doit être une adresse de DLSAP appelante du DLCEP abonné sur la même DLC que l'adresse de DLCEP éditeur de destination;

c) le champ "paramètres SD" doit spécifier les informations d'état pour le DLCEP adressé, et le contenu de ce champ doit être tel que décrit en 7.4;

   NOTE La taille et la structure de ce champ dépendent des attributs QoS associés au DLCEP adressé par l'adresse de DL de destination spécifiée dans cette DLPDU, et sont déterminées au cours de l'établissement du DLCEP.

d) les données utilisateur doivent spécifier les octets d'une DLSDU cohérents avec la taille de DLSDU négociée et les informations de segmentation spécifiées dans les paramètres SD d'accompagnement, et peuvent être vides.

   NOTE Une DLPDU CA avec des données utilisateur vides peut être utilisé par une DLE avec un DLCEP abonné pour solliciter, auprès du DLCEP éditeur correspondant, des informations d'état de la DLC courante.

NOTE Les formats 1L et 1S sont utilisés dans des communications peer-to-peer (voir 6.4.2b)) et abonné vers éditeur (voir 6.4.2c)) lorsque l'attribut d'authentification de la DLPDU est SOURCE ou MAXIMAL. Les formats 2L et 2S sont utilisés pour des communications peer-to-peer et abonné vers éditeur lorsque l'attribut d'authentification de la DLPDU est ORDINARY.

Il y a erreur de protocole si les conditions ci-dessus ne sont pas remplies.

6.4.3 Envoi de la DLPDU CA

Une DLPDU CA peut être sélectionnée pour transmission sur la liaison quand

a) la DLE expéditrice détient un jeton programmeur ou jeton délégué qui est le jeton dominant sur la liaison locale;

b) la durée allouée restante de l'utilisation du jeton, C (RD), permet l'achèvement de V(MRC)+1 transactions implicites avant l'expiration du jeton, chaque transaction consistant à envoyer la DLPDU CA qui exige une réponse immédiate et à attendre une DLPDU SR de cas le plus défavorable ou une DLPDU de réponse DT autorisée de cas le plus défavorable ne contenant pas de données utilisateur de DLS;

c) s'il est prévu d'adresser la DLPDU CA à une adresse de DLSAP, alors le tableau des transactions en cours, V(OTA) (voir 4.7.1.15), scruté de façon circulaire à partir du dernier indice de transaction, V(LTI) (voir 4.7.1.16), possède une entrée non assignée dont l'indice n'est pas V(LTI).

Une fois sélectionnée, s'il est prévu d'adresser la DLPDU CA à une adresse de DLSAP, alors

– V(LTI) doit être mis à l'indice de cette entrée non assignée dans V(OTA);

– cette entrée dans V(OTA) doit être assignée à la DLPDU CA sélectionnée et doit enregistrer des informations qui permettent de corréler une DLPDU DT de réponse attendue à l'invocation spécifique du service de transfert d'unidata qui a généré la DLPDU CA.

Une fois sélectionnée, la transmission de la DLPDU CA doit être tentée de nouveau jusqu'à ce que soit

1) une DLPDU de réponse immédiate admissible soit reçue, soit

2) une DLPDU inadmissible soit reçue quand une réponse immédiate était attendue, soit

3) la transmission originale et le nombre maximal autorisé de répétitions de tentative de transmission, V(MRC) (voir 4.7.1.5), aient échoué à obtenir l'une des DLPDU de réponse admissibles.

Si la DLE détient un jeton délégué et aucune utilisation supplémentaire de ce jeton après envoi de cette DLPDU et attente de sa réponse immédiate n'est nécessaire à cet instant-là, la DLE peut mettre à la valeur FINAL le sous-champ "utilisation finale du jeton" de la DLPDU CA; sinon, ce sous-champ doit avoir la valeur NOT-FINAL.

NOTE La transaction FINAL est nécessairement celle de rang (V(MRC)+1) dans une série, autrement, si une réponse n'est pas reçue, une autre DLPDU CA est à envoyer avant la fin du cycle courant d'utilisation du jeton, rendant NOT-FINAL la transaction courante.

Chaque adresse de DL explicite dans la DLPDU CA doit être délocalisée (voir 5.2.2) avant transmission.

Après l'envoi d'une DLPDU CA, la DLE expéditrice doit surveiller la liaison locale pour détecter une réponse telle que spécifiée en 5.2.8.1. La DLPDU de réponse admissible est soit

– une DLPDU DT dont l'adresse de DL de destination est implicitement ou explicitement l'adresse de DLSAP d'origine ou le DLCEP spécifié par la DLPDU CA, soit

– une DLPDU DT sans adresse de DL de destination, soit

– une DLPDU SR.

Si V(LTI) a été assigné à la transaction, comme résultat de la recherche décrite en c), et si une DLPDU de réponse admissible n'a pas été reçue, l'entrée de rang V(LTI) doit être annulée.

6.4.4 Réception de la DLPDU CA

Chaque adresse de DL dans la DLPDU doit être délocalisée (voir 5.2.2) à la suite de la réception.

Une DLPDU CA reçue doit être traitée comme suit par la DLE destinataire.

6.4.4.1 Actions requises de toutes les DLE

NOTE La prochaine variante tente de détecter la réception d'une DLPDU CA en doublon résultant d'une répétition immédiate de tentative par la DLE détentrice de jeton courant, qui elle-même était probablement due à une erreur détectée au cours de la réception d'une DLPDU DT de réponse antérieure. Dans un tel cas, la DLPDU DT de réponse doit être identique à la première dans les champs et sous-champs qui acheminent les données utilisateur de DLS et des informations connexes.

Les considérations suivantes s'appliquent:

a) Si
   1) l'adresse de DL de destination spécifiée par la DLPDU désigne comme une adresse de DLSAP active de la DLE destinataire, ou une adresse de DLCEP active d'une DLC pour laquelle la DLE destinataire est un homologue ou un éditeur;
   2) la DLPDU immédiatement antérieure a été émise comme réponse immédiate à une DLPDU CA reçue dont l'adresse de DL de destination était la même adresse de DLSAP ou la même adresse de DLCEP;
   3) si une adresse de DL source est présente dans la DLPDU CA juste reçue, elle était également présente et identique dans cette DLPDU CA reçue antérieure;
   4) en commençant par le champ "paramètres SD", les trois premiers octets de la DLPDU CA juste reçue, ou le reste de la DLPDU s'il y a moins de trois octets, sont identiques aux octets correspondants de cette DLPDU CA reçue antérieure;
   5) une période d'inactivité de liaison de (immediate-response-recovery-delay + 1) intervalles de temps, à savoir (V(IRRD) + 1) $\times$ V(ST) durées d'octet, n'a pas eu lieu depuis la réception de cette DLPDU CA reçue antérieure,

   alors, la DLE destinataire

   i) doit rejeter la DLPDU reçue et ne pas la transmettre aux fonctions de niveau supérieur de la DLE pour un traitement ultérieur;
   ii) doit réémettre la DLPDU DT de réponse immédiate émise antérieure, dans les limites d'une période de maximum-response-delay intervalles de temps, à savoir V(MRD) $\times$ V(ST) durées d'octet, par rapport à la réception de la DLPDU CA, inchangée par rapport à la transmission antérieure, sauf pour les sous-champs
      – du sous-champ "utilisation finale du jeton" de l'octet de contrôle de trame qui achemine les informations au LAS d'écoute;
      – les sous-champs de tous les éventuels paramètres SD de DLCEP présents qui acquittent la réception des DLSDU reçues ou en demandent la rtransmission.

   NOTE Cette exigence est destinée à assurer que la priorité de DL, le contenu et l'identité de toutes les éventuelles données utilisateur de DLS acheminées dans la DLPDU DT de réponse immédiate sont identiques à celles de la DLPDU DT immédiatement antérieure envoyée à partir de cette même adresse de DL, incluant l'id de transaction de l'émetteur s'il y en a un présent dans les paramètres SD reçus et émis.

b) Si a) ne s'applique pas et l'adresse de DL de destination spécifiée par la DLPDU désigne une adresse de DLSAP de la DLE destinataire, alors la DLPDU reçue doit être transmise aux fonctions de niveau supérieur de la DLE (8.3) pour un traitement ultérieur.

   NOTE Les adresses de DL de groupe ne sont pas couvertes ici; de telles DLPDU CA sont erronées et ne sont pas reconnues à la réception.

La DLE destinataire doit initier une réponse dans les limites d'une période de maximum-response-delay intervalles de temps, à savoir V(MRD) × V(ST) durées d'octet, par rapport à la réception de la DLPDU CA La DLPDU de réponse doit être une DLPDU DT avec une adresse de DL de destination explicite quelconque dans la DLPDU DT ayant la même valeur que l'adresse de DL source de la DLPDU CA reçue, et une longueur et un format comme suit:

– DT (voir en 6.7.1 le format 2L) en réponse à CA (voir en 6.4.1 le format 1L), ou
– DT (voir en 6.7.1 le format 4) en réponse à CA (voir en 6.4.1 le format 1S),

et doit inclure les paramètres SD spécifiant

– l'id de transaction de l'émetteur tel que reçu dans la DLPDU CA de stimulation, tel que spécifié en 6.4.2.1a)4);
– le statut de réception de la DLSDU acheminée par cette DLPDU CA.

La DLPDU DT de réponse ne doit contenir aucune donnée d'utilisateur.

c) Si a) ne s'applique pas et l'adresse de DL de destination spécifiée par la DLPDU désigne une adresse de DLCEP active d'une DLC pour laquelle la DLE destinataire
   – est un homologue ou un éditeur et la priorité DLL du DLCEP n'est pas identique à la priorité indiquée dans la DLPDU reçue, ou
   – est un homologue et la longueur et le nombre d'adresses de DL ne sont pas tels que prévus ou la DLPDU spécifie une adresse source explicite qui n'est pas égale à l'adresse de DLCEP du DLCEP homologue distant,

   alors

   1) la DLPDU reçue doit être transmise aux fonctions de niveau supérieur de la DLE (voir 8.2.2.9) pour un traitement ultérieur;
   2) la DLE destinataire doit initier une réponse dans les limites d'une période de maximum-response-delay intervalles de temps, à savoir V(MRD) × V(ST) durées d'octet, par rapport à la réception de la DLPDU CA La DLPDU de réponse doit être une DLPDU DT au format négocié pour la DLC pour le sens choisi de transmission et doit contenir les paramètres SD appropriés au DLCEP expéditeur, mais ne doit pas contenir de donnée utilisateur de DLS.

d) Si ni a) ni c) ne s'appliquent et l'adresse DL de destination spécifiée par la DLPDU désigne une adresse de DLCEP active d'une DLC pour laquelle la DLE destinataire est un homologue ou un éditeur, alors
   1) la DLPDU reçue doit être transmise aux fonctions de niveau supérieur de la DLE (voir 8.2.2.9) pour un traitement ultérieur;
   2) la DLE destinataire doit initier une réponse dans les limites d'une période de maximum-response-delay intervalles de temps, à savoir V(MRD) × V(ST) durées d'octet, par rapport à la réception de la DLPDU CA La DLPDU de réponse doit être une DLPDU DT au format négocié pour la DLC pour le sens choisi de transmission, doit contenir les paramètres SD appropriés au DLCEP expéditeur, mais ne doit pas contenir de donnée utilisateur de DLS.
   3) la DLPDU CA demandeuse peut contenir des informations concernant l'état du DLCEP demandeur. La DLPDU DT de réponse a la permission, mais n'est pas tenue, de refléter ces informations d'état dans sa réponse; le traitement immédiat de ces informations d'état avant l'envoi de la réponse immédiate doit être permis, mais ne doit pas être exigé.

e) Si a) ne s'applique pas et l'adresse de DL de destination spécifiée par la DLPDU désigne une adresse de DLCEP active d'une DLC pour laquelle la DLE destinataire est un abonné, alors
   1) la partie autre que des données utilisateur de DLS de la DLPDU reçue doit être transmise aux fonctions de niveau supérieur de la DLE (voir 8.2.2.9) pour un traitement ultérieur;

2) la DLE destinataire doit enregistrer l'adresse de DL de destination issue de la DLPDU CA reçue dans V(RA) pour association ultérieure avec la DLPDU DT de réponse immédiate attendue, qui est recommandée être la prochaine DLPDU reçue;

3) la DLE destinataire doit surveiller la liaison locale pour détecter une réponse, puis agir selon le résultat de cette surveillance, tout cela tel que spécifié en 5.2.8.3.

6.4.4.2 Actions supplémentaires requises d'une DLE de classe Link-Master

Aucune.

6.4.4.3 Actions supplémentaires requises d'une DLE de classe Bridge

Les considérations suivantes s'appliquent:

a) Étant donné que chaque DLE de classe Bridge a une capacité de maître de liaison, toutes les actions spécifiées en 6.1.4.2 s'appliquent également à une DLE de la classe Bridge.

b) Si toutes les conditions suivantes sont remplies

   1) l'adresse de DL de destination spécifiée dans la DLPDU est une adresse qu'il convient que le pont transmette, mais vers laquelle la DLE pont elle-même ne générerait pas autrement de DLPDU de réponse immédiate;

   2) la DLPDU immédiatement antérieure a été transmise comme réponse immédiate à une DLPDU CA reçue avec la même adresse de DL de destination;

   3) si une adresse de DL source est présente dans la DLPDU CA juste reçue, elle était également présente et identique dans cette DLPDU CA reçue antérieure;

   4) la partie "paramètres DLC de base" des paramètres SD de la DLPDU CA juste reçue est identique à celle de cette DLPDU CA reçue antérieure;

   5) une période d'inactivité de liaison de (immediate-response-recovery-delay plus un) intervalles de temps, à savoir (V(IRRD) +1) × V(ST) durées d'octet, n'a pas eu lieu depuis la réception de cette DLPDU CA reçue antérieure,

   alors le pont

   i) doit rejeter la DLPDU reçue et ne pas la transmettre aux autres fonctions du pont pour un traitement ultérieur;

   ii) doit lancer la retransmission de la DLPDU SR de réponse immédiate émise antérieure, inchangée par rapport à la transmission antérieure.

   NOTE Cette exigence est destinée à garantir que tout statut acheminé dans la DLPDU SR de réponse immédiate est identique à celui qui est contenue dans la DLPDU SR antérieure.

   Autrement,

   A) Si l'adresse de DL de destination spécifiée dans la DLPDU est une adresse qu'il convient que le pont transmette, mais que la DLE pont elle-même ne recevrait autrement pas, le pont doit former et envoyer une DLPDU SR

      – dans les limites d'une période de maximum-response-delay intervalles de temps, à savoir V(MRD) × V(ST) durées d'octet, par rapport à la réception de la DLPDU CA;

      – avec un statut indiquant si, oui ou non, le pont a été capable de placer en tampon la DLPDU reçue;

   B) Si l'adresse DL de destination spécifiée dans la DLPDU est une adresse qu'il convient que le pont transmette et le pont a été capable de recevoir et placer en tampon la DLPDU sans erreur, alors la DLPDU reçue doit être, au besoin, transmise avec la modification du champ "contrôle de trame" dans la DLPDU transmise (voir 5.2.5.3).

   C) Si la DLPDU contient une adresse de DL source explicite, le pont doit tenter de mettre à jour son entrée de table de routage pour l'adresse de DL source spécifiée dans la DLPDU afin de refléter le port du pont à partir duquel la DLPDU a été reçue.

c) C'est une erreur de protocole pour une DLE pont qui transmet la DLPDU CA que de ne pas envoyer une DLPDU SR de réponse SR lorsqu'une réponse est exigée.

NOTE Il convient qu'au maximum une seule DLE pont sur la liaison locale transmette la DLPDU CA reçue.

6.4.4.4 Actions supplémentaires requises de la DLE LAS courante

La DLE LAS doit agir comme spécifié en 5.2.8.2.

6.5 DLPDU "Compel data" (Forcer les données (CD))

Une DLPDU COMPEL DATA (Forcer les données (CD)) est utilisée pour demander le transfert (ou le retransfert) d'une quantité limitée de données utilisateur transparentes à partir d'un autre utilisateur de DLS vers un utilisateur de DLS demandeur sans exiger que la transaction (demande et acquittement) s'effectue dans le contexte d'une DLC. Elle est également utilisée à des fins similaires dans le contexte d'une DLC, et pour aider à la synchronisation des DLCEP et de leurs utilisateurs DLS. Lorsque la DLPDU CD est envoyée vers un DLCEP éditeur, les données utilisateur de DLS dont le transfert est demandé seront distribuées à tous les abonnés de la DLC.

Une DLPDU CD crée et transmet un jeton de réponse à la DLE destinataire adressée, qui, à la suite de la réception, devient le jeton dominant sur la liaison. Une DLPDU CD exige une réponse immédiate soit d'une DLPDU DATA (DT), soit d'une DLPDU STATUS RESPONSE (SR). Si aucune DLPDU n'est reçue en réponse, la transaction est V(MRC) fois au maximum.

6.5.1 Structure des DLPDU CD

Tableau 16 – Structure des DLPDU CD

Format	Frame control (Contrôle de trame)	Adresse de destination	Adresse source	Paramètres
1L	1111 1FPP	HL.N.S	HL.N.S	o-SD-p
1s	1111 0FPP	N.S	N.S	o-SD-p
2L	1011 1FPP	HL.N.S	—	o-SD-p
2s	1011 0FPP	N.S	—	o-SD-p

6.5.1.1 Champ "Frame control" (contrôle de trame)

Le champ "contrôle de trame" doit spécifier

a) la fonction de la DLPDU, y compris qu'une réponse immédiate est demandée;

b) la priorité de la transaction et la priorité implicite de la DLPDU;

c) la longueur, le nombre et le type d'adresses de DLPDU;

d) si, oui ou non, la transmission de la DLPDU courante, et soit sa réponse immédiate escomptée, soit la temporisation de repos de liaison appropriée, mettent fin à l'utilisation d'un jeton délégué.

NOTE Ce champ a nécessairement la valeur NOT-FINAL lorsqu'une répétition immédiate de tentative de la transaction courante est possible. Seule une transaction qui est garantie ne pas avoir besoin d'une répétition immédiate de tentative peut avoir la valeur FINAL spécifiée dans sa DLPDU associée.

6.5.1.2 Champ "Address" (adresse)

Le champ "adresse" doit être constitué soit

a) d'une adresse de DL de destination explicite et d'une adresse de DL source explicite, dans cet ordre, pour les formats 1L et 1S, soit

b) de seulement une adresse de DL destination explicite, pour les formats 2L et 2S.

Pour les formats 1L et 2L, toutes les adresses doivent être LONG; pour les formats 1S et 2S, toutes les adresses doivent être SHORT;

6.5.1.3 Champ "Parameters" (paramètres)

Le champ "status-data-parameters" (paramètres de données de statut (paramètres SD)) spécifie les informations appropriées à l'adresse de DL de destination associée:

a) Si cette adresse de DL est une adresse de DLSAP liée dans un rôle de DL(SAP) répondeur, les paramètres SD spécifient un id de transaction utilisé par la DLE d'origine pour corréler à la demande d'origine une réponse différée renvoyée. Ce champ doit être structuré et codé tel que décrit en 7.4.1.

b) Si cette adresse de DL est une adresse de DLCEP, les paramètres SD spécifient des informations d'état pour le DLCEP adressé. Ce champ peut être vide (null). S'il n'est pas vide, ce champ doit être structuré et codé tel que décrit en 7.4.2.

NOTE La taille et la structure de ce champ dépendent des attributs QoS associés au DLCEP adressé par l'adresse de DL de destination spécifiée dans cette DLPDU, et sont déterminées au cours de l'établissement du DLCEP.

Lorsque la DLPDU CD est envoyée par la DLE LAS, ce qui se produit alors que le jeton dominant est un jeton programmeur, ce champ doit être vide.

6.5.1.4 Champ "User data" (données d'utilisateur)

Le champ "données d'utilisateur" doit être vide.

6.5.2 Contenu de la DLPDU CD

Le champ "contrôle de trame" doit être codé comme spécifié dans le Tableau 16.

Soit les adresses de DL doivent être

a) deux adresses de DLSAP explicites, soit

b) une adresse de DLCEP explicite et une seconde adresse de DLCEP explicite ou implicite ou absente, soit

c) une adresse de DLCEP explicite, suivie d'une adresse de DLSAP explicite ou implicite.

6.5.2.1 Contenu de la DLPDU CD lors de la spécification d'une adresse de DLSAP de destination

Lorsque la première adresse de DL est une adresse de DLSAP comme en 6.5.2a), alors

a) si le format de DLPDU est le format 1L ou 1S, alors
   1) cette DLPDU est utilisée pour mettre en œuvre le service d'échange d'unitdata;
   2) le rôle de DL(SAP) pour l'adresse de DLSAP de destination doit être CONSTRAINED RESPONDER ou UNCONSTRAINED RESPONDER;
   3) la seconde adresse doit être présente, doit être une adresse de DLSAP, et le rôle de DL(SAP) pour cette adresse DLSAP doit être INITIATOR;
   4) le champ "paramètres SD" doit spécifier l'id de transaction utilisé par la DLE d'origine pour corréler à la demande d'origine une réponse différée envoyée, où le contenu de ce champ doit être tel que décrit en 7.4;
   5) les données utilisateur doivent être vides.

b) aucun autre format de DLPDU ne peut être utilisé.

6.5.2.2 Contenu de la DLPDU CD lors de la spécification d'une adresse de DLCEP de destination

Lorsque la première adresse est une adresse de DLCEP comme en 6.5.2b) et 6.5.2c), alors

a) cette DLPDU doit demander des informations d'état issues du DLCEP adressé et doit demander que des données utilisateur de DLS soient incluses dans la DLPDU de réponse;

b) la seconde adresse, si elle est présente,

   – doit être l'adresse de DLCEP homologue de la même DLC que l'adresse de DLCEP homologue de destination, ou

   – doit être une adresse de DLSAP appelante du DLCEP abonné sur la même DLC que l'adresse de DLCEP éditeur de destination;

c) le champ "paramètres SD", s'il est présent, doit spécifier les informations d'état pour le DLCEP adressé, et le contenu de ce champ doit être tel que décrit en 7.4;

   NOTE La taille et la structure de ce champ dépendent des attributs QoS associés au DLCEP adressé par l'adresse de DL de destination spécifiée dans cette DLPDU, et sont déterminées au cours de l'établissement du DLCEP.

d) les données utilisateur doivent être vides.

   NOTE Si le champ de données utilisateur n'est pas vide, il convient d'utiliser une DLPDU ED (voir 6.6) à la place.

NOTE Les formats 1L et 1S sont utilisés dans des communications peer-to-peer (voir 6.5.2b)) et abonné vers éditeur (voir 6.5.2c)) lorsque l'attribut d'authentification de la DLPDU est SOURCE ou MAXIMAL. Les formats 2L et 2S sont utilisés pour des communications peer-to-peer et abonné vers éditeur lorsque l'attribut d'authentification de la DLPDU est ORDINARY.

6.5.3 Envoi de la DLPDU CD

Une DLPDU CD peut être sélectionnée pour transmission sur la liaison quand

a) la DLE expéditrice détient un jeton programmeur ou jeton délégué qui est le jeton dominant sur la liaison locale;

b) la durée allouée restante de l'utilisation du jeton, C (RD), permet l'achèvement de V(MRC)+1 transactions implicites avant l'expiration du jeton, chaque transaction consistant à envoyer la DLPDU CD qui exige une réponse immédiate et à attendre une DLPDU SR de cas le plus défavorable ou une DLPDU de réponse DT autorisée de cas le plus défavorable ne contenant pas de données utilisateur de DLS;

c) s'il est prévu d'adresser la DLPDU CD à une adresse de DLSAP, alors le tableau des transactions en cours, V(OTA) (voir 4.7.1.15), scruté de façon circulaire à partir du dernier indice de transaction, V(LTI) (voir 4.7.1.16), possède une entrée non assignée dont l'indice n'est pas V(LTI).

Une fois sélectionnée, s'il est prévu d'adresser la DLPDU CD à une adresse de DLSAP, alors

– V(LTI) doit être mis à l'indice de cette entrée non assignée dans V(OTA);

– cette entrée dans V(OTA) doit être assignée à la DLPDU CD sélectionnée et doit enregistrer des informations qui permettent de corréler une DLPDU DT de réponse attendue à l'invocation spécifique du service d'échange d'unidata qui a généré la DLPDU CD.

Une fois sélectionnée, la transmission de la DLPDU CD doit être tentée de nouveau jusqu'à ce que soit

1) une DLPDU de réponse immédiate admissible soit reçue, soit

2) une DLPDU inadmissible soit reçue quand une réponse immédiate était attendue, soit

3) la transmission originale et le nombre maximal autorisé de répétitions de tentative de transmission, V(MRC) (voir 4.7.1.5), aient échoué à obtenir l'une des DLPDU de réponse admissibles.

Si la DLE détient un jeton délégué et aucune utilisation supplémentaire de ce jeton après envoi de cette DLPDU et attente de sa réponse immédiate n'est nécessaire à cet instant-là, la DLE peut mettre à la valeur FINAL le sous-champ "utilisation finale du jeton" de la DLPDU CD; sinon, ce sous-champ doit avoir la valeur NOT-FINAL.

NOTE La transaction FINAL est nécessairement celle de rang (V(MRC)+1) dans une série, autrement, si une réponse n'est pas reçue, une autre DLPDU CD est à envoyer avant la fin du cycle courant d'utilisation du jeton, rendant NOT-FINAL la transaction courante.

Chaque adresse de DL explicite dans la DLPDU CD doit être délocalisée (voir 5.2.2) avant transmission.

Après l'envoi d'une DLPDU CD, la DLE expéditrice doit surveiller la liaison locale pour détecter une réponse telle que spécifiée en 5.2.8.1. La DLPDU de réponse admissible est soit

– une DLPDU DT dont l'adresse de DL de destination est implicitement ou explicitement l'adresse de DLSAP d'origine ou le DLCEP spécifié par la DLPDU CD, soit
– une DLPDU DT sans adresse de DL de destination, soit
– une DLPDU SR.

Si V(LTI) a été assigné à la transaction, comme résultat de la recherche décrite en c), et si une DLPDU de réponse admissible n'a pas été reçue, l'entrée de rang V(LTI) doit être annulée.

6.5.4 Réception de la DLPDU CD

Chaque adresse de DL dans la DLPDU doit être délocalisée (voir 5.2.2) à la suite de la réception.

Une DLPDU CD reçue doit être traitée comme suit par la DLE destinataire.

6.5.4.1 Actions requises de toutes les DLE

NOTE La prochaine variante tente de détecter la réception d'une DLPDU CD en doublon résultant d'une répétition immédiate de tentative par la DLE détentrice de jeton courant, qui elle-même était probablement due à une erreur détectée au cours de la réception d'une DLPDU DT de réponse antérieure. Dans un tel cas, la DLPDU DT de réponse doit être identique à la première dans les champs et sous-champs qui acheminent les données utilisateur de DLS et des informations connexes.

Les considérations suivantes s'appliquent:

a) Si
   1) l'adresse de DL de destination spécifiée par la DLPDU désigne comme une adresse de DLSAP active de la DLE destinataire, ou une adresse de DLCEP active d'une DLC pour laquelle la DLE destinataire est un homologue ou un éditeur;
   2) la DLPDU immédiatement antérieure a été émise comme réponse immédiate à une DLPDU CD reçue dont l'adresse de DL de destination était la même adresse de DLSAP ou la même adresse de DLCEP;
   3) si une adresse de DL source est présente dans la DLPDU CD juste reçue, elle était également présente et identique dans cette DLPDU CD reçue antérieure;
   4) en commençant par le champ "paramètres SD", les trois premiers octets de la DLPDU CD juste reçue, ou le reste de la DLPDU s'il y a moins de trois octets, sont identiques aux octets correspondants de cette DLPDU CD reçue antérieure;
   5) une période d'inactivité de liaison de (immediate-response-recovery-delay + 1) intervalles de temps, à savoir (V(IRRD) + 1) × V(ST) durées d'octet, n'a pas eu lieu depuis la réception de cette DLPDU CD reçue antérieure,

   alors, la DLE destinataire

i) doit rejeter la DLPDU reçue et ne pas la transmettre aux fonctions de niveau supérieur de la DLE pour un traitement ultérieur;

ii) doit réémettre la DLPDU DT de réponse immédiate émise antérieure, dans les limites d'une période maximum-response-delay intervalles de temps, à savoir V(MRD) × V(ST) durées d'octet, par rapport à la réception de la DLPDU CD, inchangée par rapport à la transmission antérieure, sauf pour les sous-champs

– du sous-champ "utilisation finale du jeton" de l'octet de contrôle de trame qui achemine les informations au LAS d'écoute;

– les sous-champs de tous les éventuels paramètres SD de DLCEP présents qui acquittent la réception des DLSDU reçues ou en demandent la rtransmission.

NOTE Cette exigence est destinée à assurer que la priorité de DL, le contenu et l'identité de toutes les éventuelles données utilisateur de DLS acheminées dans la DLPDU DT de réponse immédiate sont identiques à celles de la DLPDU DT immédiatement antérieure envoyée à partir de cette même adresse de DL, incluant l'id de transaction de l'émetteur s'il y en a un présent dans les paramètres SD reçus et émis.

b) Si a) ne s'applique pas et l'adresse de DL de destination spécifiée par la DLPDU désigne une adresse de DLSAP de la DLE destinataire, alors le traitement de la DLPDU reçue doit être fondé sur le rôle de DL(SAP) spécifié pour cette adresse de DLSAP.

NOTE Les adresses de DL de groupe ne sont pas couvertes ici; de telles DLPDU CD sont erronées et ne sont pas reconnues à la réception.

La DLE destinataire doit initier une réponse dans les limites d'une période de maximum-response-delay intervalles de temps, à savoir V(MRD) × V(ST) durées d'octet, par rapport à la réception de la DLPDU CD

1) Si la DLE peut la préparer à temps, la DLPDU de réponse doit être une DLPDU DT avec une adresse DL de destination explicite quelconque dans la DLPDU DT ayant la même valeur que l'adresse de DL source de la DLPDU CD reçue, et une longueur et un format comme suit:

– DT (voir en 6.7.1 le format 2L) en réponse à CD (voir en 6.5.1 le format 1L), ou

– DT (voir en 6.7.1 le format 2S) en réponse à CD (voir en 6.5.1 le format 1S) si la DLPDU DT contient des données utilisateur de DLS, ou

– DT (voir en 6.7.1 le format 4) en réponse à CD (voir en 6.5.1 le format 1S) si la DLPDU DT ne contient pas de données utilisateur de DL,

et doit inclure les paramètres SD spécifiant

– l'id de transaction de l'émetteur tel que reçu dans la DLPDU CD de stimulation, tel que spécifié en 6.5.2.1a)4);

– statut pour la DLSDU demandée par cette DLPDU CD.

La DLPDU DT de réponse peut comporter une DLSDU qui a déjà été mise en tampon dans cette DLE répondeuse réponse au moment de la réception de CD.

NOTE Cette restriction interdit à la DLPDU de réponse de refléter un quelconque traitement de couche supérieure (à la DLL) de la DLSDU reçue. Cette restriction est nécessaire pour permettre la migration à partir des normes nationales antérieures.

Si le rôle de DL(SAP) de l'adresse de DLSAP de destination spécifiait

i) BASIC ou INITIATOR, la DLE doit rejeter la DLPDU CD reçue en fonction de ce rôle de DL(SAP), et un statut d'erreur approprié sans DLSDU d'accompagnement doit être inclus dans la DLPDU DT de réponse;

ii) UNCONSTRAINED RESPONDER, le statut de paramètre SD doit indiquer le tampon expéditeur avec la priorité DLL la plus élevée avec une DLSDU non vide qui était disponible au DLSAP adressé au moment de la réception de la DLPDU CD;

A) si cette priorité DLL est supérieure ou égale à la priorité spécifiée en 6.5.1.1b), cette DLSDU doit être incluse dans la DLPDU DT de réponse et son tampon mis à vide si configuré ainsi, et la priorité DLL de cette DLPDU DT de réponse doit être la priorité DLL du DLSDU acheminée; ou

B) si cette priorité DLL est inférieure à la priorité spécifiée en 6.5.1.1b), aucune DLSDU ne doit être incluse dans la DLPDU DT de réponse, et la priorité DLL de cette DLPDU DT de réponse doit être la priorité DLL spécifiée en 6.5.1.1b); ou

C) s'il n'y a pas un tel tampon expéditeur avec une DLSDU non vide, la DLPDU DT de réponse doit indiquer un statut d'erreur approprié, la priorité DLL de cette DLPDU DT de réponse doit être la priorité DLL spécifiée en 6.5.1.1b), et cette DLPDU DT de réponse ne doit pas contenir de données utilisateur de DLS

iii) constrained responder, alors

A) si l'adresse de DLSAP source de la DLPDU CD est égale à l'adresse de DLSAP distante qui était spécifiée dans la primitive "request" de DL-BIND antérieure pour l'adresse de DLSAP destinataire (ou son équivalent de gestion de DL), après délocalisation pour les deux, alors la procédure indiquée en ii) doit être suivie;

B) autrement, la DLSDU reçue doit être supprimée, et un statut d'erreur approprié sans DLSDU d'accompagnement doit être inclus dans la DLPDU DT de réponse;.

2) Si la DLE ne peut préparer à temps la DLPDU DT de réponse requise, la DLE doit envoyer une DLPDU DT avec une adresse DL de destination explicite quelconque dans la DLPDU DT ayant la même valeur que l'adresse de DL source de la DLPDU CD reçue, et une longueur et un format comme suit:

– DT (voir en 6.7.1 le format 2L) en réponse à CD (voir en 6.5.1 le format 1L), ou

– DT (voir en 6.7.1 le format 4) en réponse à CD (voir en 6.5.1 le format 1S),

et doit inclure les paramètres SD

– avec un id de transaction identique à l'id de transaction issu de la DLPDU CD reçue, telle que spécifiée en 6.5.2.1a)4);

– avec un statut (DR – “delayed reply”, c'est-à-dire"réponse différée") indiquant que la DLE exige du temps supplémentaire

pour préparer la réponse requise;

– avec un champ "données utilisateur" vide,

et la DLE

– doit préparer cette DLPDU DT telle que spécifiée en 1) dès que possible;

– doit inclure une adresse de destination explicite dans la DLPDU DT de réponse;

– doit abouter cette DLPDU DT de réponse à la Q(US) de la DLE devant être émise dès que possible;

c) Si a) ne s'applique pas et l'adresse de DL de destination spécifiée par la DLPDU désigne une adresse de DLCEP active d'une DLC pour laquelle la DLE destinataire

– est un homologue ou un éditeur et la priorité DLL du DLCEP n'est pas identique à la priorité indiquée dans la DLPDU reçue, ou

– est un homologue et la longueur et le nombre d'adresses de DL ne sont pas tels que prévus ou la DLPDU spécifie une adresse source explicite qui n'est pas égale à l'adresse de DLCEP du DLCEP homologue distant,

alors

   1) la DLPDU reçue doit être transmise aux fonctions de niveau supérieur de la DLE (voir 8.2.2.9) pour un traitement ultérieur;
   2) la DLE destinataire doit initier une réponse dans les limites d'une période de maximum-response-delay intervalles de temps, à savoir V(MRD) × V(ST) durées d'octet, par rapport à la réception de la DLPDU CD. La DLPDU de réponse doit être une DLPDU DT au format négocié pour la DLC pour le sens choisi de transmission et doit contenir les paramètres SD appropriés au DLCEP expéditeur, mais ne doit pas contenir de donnée utilisateur de DLS.

d) Si ni a) ni c) ne s'appliquent et l'adresse DL de destination spécifiée par la DLPDU désigne une adresse de DLCEP active d'une DLC pour laquelle la DLE destinataire est un homologue ou un éditeur, alors
   1) la DLPDU reçue doit être transmise aux fonctions de niveau supérieur de la DLE (voir 8.2.2.9) pour un traitement ultérieur;
   2) la DLE destinataire doit initier une réponse dans les limites d'une période de maximum-response-delay intervalles de temps, à savoir V(MRD) × V(ST) durées d'octet, par rapport à la réception de la DLPDU CD. La DLPDU de réponse doit être une DLPDU DT au format négocié pour la DLC pour le sens sélectionné de transmission, doit contenir les paramètres SD appropriés pour le DLCEP expéditeur, et doit contenir les données utilisateur de DLS s'il y en a qui étaient disponibles et attendaient la transmission ou la retransmission à partir du DLCEP;

   NOTE La migration à partir de normes nationales antérieures exige que ces données soient toujours incluses lorsqu'elles sont disponibles.

   3) La DLPDU CD demandeuse peut contenir des informations concernant l'état du DLCEP demandeur. La DLPDU DT de réponse a la permission, mais n'est pas tenue, de refléter ces informations d'état dans sa réponse; le traitement immédiat de ces informations d'état avant l'envoi de la réponse immédiate doit être permis, mais ne doit pas être exigé.

e) Si a) ne s'applique pas et l'adresse de DL de destination spécifiée par la DLPDU désigne une adresse de DLCEP active d'une DLC pour laquelle la DLE destinataire est un abonné, alors
   1) la partie autre que des données utilisateur de DLS de la DLPDU reçue doit être transmise aux fonctions de niveau supérieur de la DLE (voir 8.2.2.9) pour un traitement ultérieur;
   2) la DLE destinataire doit enregistrer l'adresse de DL de destination issue de la DLPDU CD reçue dans V(RA) pour association ultérieure avec la DLPDU DT de réponse immédiate attendue, qui est recommandée être la prochaine DLPDU reçue;
   3) la DLE destinataire doit surveiller la liaison locale pour détecter une réponse, puis agir selon le résultat de cette surveillance, tout cela tel que spécifié en 5.2.8.3.

6.5.4.2 Actions supplémentaires requises d'une DLE de classe Link-Master

Aucune.

6.5.4.3 Actions supplémentaires requises d'une DLE de classe Bridge

Les considérations suivantes s'appliquent:

a) Étant donné que chaque DLE de classe Bridge a une capacité de maître de liaison, toutes les actions spécifiées en 6.5.4.2 s'appliquent également à une DLE de la classe Bridge.
b) Si toutes les conditions suivantes sont remplies
   A) l'adresse de DL de destination spécifiée dans la DLPDU est une adresse qu'il convient que le pont transmette, mais vers laquelle la DLE pont elle-même ne générerait pas autrement de DLPDU de réponse immédiate;
   B) la DLPDU immédiatement antérieure a été transmise comme réponse immédiate à une DLPDU CD reçue avec la même adresse de DL de destination;

C) si une adresse de DL source est présente dans la DLPDU CD juste reçue, elle était également présente et identique dans cette DLPDU CD reçue antérieure;

D) la partie "paramètres DLC de base" des paramètres SD de la DLPDU CD juste reçue est identique à celle de cette DLPDU CD reçue antérieure;

E) une période d'inactivité de liaison de (immediate-response-recovery-delay plus un) intervalles de temps, à savoir (V(IRRD) +1) × V(ST) durées d'octet, n'a pas eu lieu depuis la réception de cette DLPDU CD reçue antérieure,

alors le pont

1) doit rejeter la DLPDU reçue et ne pas la transmettre aux autres fonctions du pont pour un traitement ultérieur;

2) doit lancer la retransmission de la DLPDU SR de réponse immédiate émise antérieure, inchangée par rapport à la transmission antérieure.

NOTE Cette exigence est destinée à garantir que tout statut acheminé dans la DLPDU SR de réponse immédiate est identique à celui qui est contenue dans la DLPDU SR antérieure.

Autrement

i) Si l'adresse de DL de destination spécifiée dans la DLPDU est une adresse qu'il convient que le pont transmette, mais que la DLE pont elle-même ne recevrait autrement pas, le pont doit former et envoyer une DLPDU SR

– dans les limites d'une période de maximum-response-delay intervalles de temps, à savoir V(MRD) × V(ST) durées d'octet, par rapport à la réception de la DLPDU CD;

– avec un statut indiquant si, oui ou non, le pont a été capable de placer en tampon la DLPDU reçue.

ii) Si l'adresse DL de destination spécifiée dans la DLPDU est une adresse qu'il convient que le pont transmette et le pont a été capable de recevoir et placer en tampon la DLPDU sans erreur, alors la DLPDU reçue doit être, au besoin, transmise avec la modification du champ "contrôle de trame" dans la DLPDU transmise (voir 5.2.5.3).

iii) Si la DLPDU contient une adresse de DL source explicite, le pont doit tenter de mettre à jour son entrée de table de routage pour l'adresse de DL source spécifiée dans la DLPDU afin de refléter le port du pont à partir duquel la DLPDU a été reçue.

c) C'est une erreur de protocole pour une DLE pont qui transmet la DLPDU CD que de ne pas envoyer une DLPDU de réponse SR lorsqu'une réponse est exigée.

NOTE Il convient qu'au maximum une seule DLE pont sur la liaison locale transmette la DLPDU CD reçue.

6.5.4.4 Actions supplémentaires requises de la DLE LAS courante

La DLE LAS doit agir comme spécifié en 5.2.8.2.

6.6 DLPDU "Exchange data" (Échanger des données (ED))

Une DLPDU EXCHANGE DATA (échanger des données (ED)) est utilisée

a) pour transférer (ou retransférer) une quantité limitée de données utilisateur transparentes de l'utilisateur de DLS demandeur vers un autre utilisateur de DLS;

b) pour demander le transfert (ou retransfert) d'une quantité limitée de données utilisateur transparentes de cet autre utilisateur de DLS vers l'utilisateur de DLS demandeur,

sans exiger que la transaction (demande et acquittement) s'effectue dans le contexte d'une DLC. Elle est également utilisée à des fins similaires dans le contexte d'une DLC, et pour aider à la synchronisation des DLCEP et de leurs utilisateurs DLS. Lorsque la DLPDU ED est envoyée vers un DLCEP éditeur, les données utilisateur de DLS dont le transfert est demandé en b) seront distribuées à tous les abonnés de la DLC.

Une DLPDU ED crée et transmet un jeton de réponse à la DLE destinataire adressée, qui, à la suite de la réception, devient le jeton dominant sur la liaison. Une DLPDU ED exige une réponse immédiate soit d'une DLPDU DATA (DT), soit d'une DLPDU STATUS RESPONSE (SR). Si aucune DLPDU n'est reçue en réponse, la transaction est répétée un maximum de V(MRC) fois.

6.6.1 Structure des DLPDU ED

Tableau 17 – Structure des DLPDU ED

Format	Contrôle de trame	Adresse de destination	Adresse source	Paramètres	User data (données d'utilisateur)
1L	1100 1FPP	HL.N.S	HL.N.S	SD-p	pDLSDU
1s	1100 0FPP	N.S	N.S	SD-p	pDLSDU
2L	1000 1FPP	HL.N.S	—	SD-p	pDLSDU
2s	1000 0FPP	N.S	—	SD-p	pDLSDU

6.6.1.1 Champ "Frame control" (contrôle de trame)

Le champ "contrôle de trame" doit spécifier

a) la fonction de la DLPDU, y compris qu'une réponse immédiate est demandée;
b) la priorité de la transaction, et la priorité implicite de la DLPDU à moins qu'elle ne soit explicitement remplacée par 6.6.1.3a)2);
c) la longueur, le nombre et le type d'adresses de DLPDU;
d) si, oui ou non, la transmission de la DLPDU courante, et soit sa réponse immédiate escomptée, soit la temporisation de repos de liaison appropriée, mettent fin à l'utilisation d'un jeton délégué.

NOTE Ce champ a nécessairement la valeur NOT-FINAL lorsqu'une répétition immédiate de tentative de la transaction courante est possible. Seule une transaction qui est garantie ne pas avoir besoin d'une répétition immédiate de tentative peut avoir la valeur FINAL spécifiée dans sa DLPDU associée.

6.6.1.2 Champ "Address" (adresse)

Le champ "adresse" doit être constitué soit

a) d'une adresse de DL de destination explicite et d'une adresse de DL source explicite, dans cet ordre, pour les formats 1L et 1S, soit
b) de seulement une adresse de DL destination explicite, pour les formats 2L et 2S.

Pour les formats 1L et 2L, toutes les adresses doivent être LONG; pour les formats 1S et 2S, toutes les adresses doivent être SHORT.

6.6.1.3 Champ "Parameters" (paramètres)

Le champ "status-data-parameters" (paramètres de données de statut (paramètres SD)) spécifie les informations appropriées à l'adresse de DL de destination associée:

a) si cette adresse de DL est une adresse de DLSAP liée dans le rôle de DL(SAP) répondeur, les paramètres SD spécifient
   1) un id de transaction utilisé par la DLE d'origine pour corréler à la demande d'origine une réponse différée retournée;

2) une priorité de DLSDU utilisée pour acheminer la priorité réelle de la DLSDU d'accompagnement à la DLE répondeuse.

Ce champ doit être structuré et codé tel que décrit en 7.4.1.

b) Si cette adresse de DL est une adresse de DLCEP, les paramètres SD spécifient des informations d'état pour le DLCEP adressé. Ce champ doit être structuré et codé tel que décrit en 7.4.2.

NOTE La taille et la structure de ce champ dépendent des attributs QoS associés au DLCEP adressé par l'adresse de DL de destination spécifiée dans cette DLPDU, et sont déterminées au cours de l'établissement du DLCEP.

6.6.1.4 Champ "User data" (données d'utilisateur)

La taille et le contenu du champ "données utilisateur" sont limités par l'adresse de DL de destination associée;

a) Si l'adresse de DL est une adresse de DLSAP, le champ "données utilisateur" doit être constitué de données utilisateur de DLS dont la taille maximale est limitée à la plus petite des tailles maximales de données utilisateur de DLS autorisées pour une DLPDU de la priorité spécifiée en 6.6.1.1b) et en 6.6.1.3a)2), et ne doit pas être vide.

b) Si cette adresse de DL est une adresse de DLCEP, les attributs de DLC négociés pour le sens voulu de transmission doivent permettre de transporter des données utilisateur de DLS dans une DLPDU ED (voir 7.1c)6) et 7.1d)6)), et le champ "données utilisateur" doit consister en des données utilisateur de DLS dont la taille maximale est limitée par la plus petite parmi

   1) la taille maximale de données utilisateur de DLS autorisée pour une DLPDU de la priorité spécifiée en 6.6.1.1b);

   2) la taille de DLSDU maximale négociée sur la DLC pour la transmission de données vers ce DLCEP,

   et ne doit pas être vide.

6.6.2 Contenu de la DLPDU ED

Le champ "contrôle de trame" doit être codé comme spécifié dans le Tableau 17.

Soit les adresses de DL doivent être

a) deux adresses de DLSAP explicites, soit

b) une adresse de DLCEP explicite et une seconde adresse de DLCEP explicite ou implicite, soit

c) une adresse de DLCEP explicite, suivie d'une adresse de DLSAP explicite ou implicite.

6.6.2.1 Contenu de la DLPDU ED lors de la spécification d'une adresse de DLSAP de destination

Lorsque la première adresse de DL est une adresse de DLSAP comme en 6.6.2a), alors

a) si le format de DLPDU est le format 1L ou 1S, alors

   1) cette DLPDU est utilisée pour mettre en œuvre le service d'échange d'unitdata;

   2) le rôle de DL(SAP) pour l'adresse de DLSAP de destination doit être CONSTRAINED RESPONDER ou UNCONSTRAINED RESPONDER;

   3) la seconde adresse doit être présente, doit être une adresse de DLSAP, et le rôle de DL(SAP) pour cette adresse DLSAP doit être INITIATOR;

   4) le champ "paramètres SD" doit spécifier une priorité de DLSDU et un id de transaction utilisés par la DLE initiatrice pour corréler à la demande d'origine une réponse différée retournée, le contenu de ce champ doit être tel que décrit en 7.4, et la priorité de

DLSDU doit être la priorité des données utilisateur d'accompagnement, doit être la même telle que spécifiée en 6.6.1.3a)2) et doit être au moins aussi élevée que la priorité de DLPDU spécifiée en 6.6.1.1b);

5) les données utilisateur doivent être une DLSDU unique dont la taille est limitée à la taille maximale pour la priorité spécifiée en 6.6.1.4a), et ne doivent pas être vides.

b) aucun autre format de DLPDU ne peut être utilisé.

6.6.2.2 Contenu de la DLPDU ED lors de la spécification d'une adresse de DLCEP de destination

Lorsque la première adresse est une adresse de DLCEP comme en 6.6.2b) et 6.6.2c), alors

a) cette DLPDU peut transporter une DLSDU unique ou partielle
   – d'un DLCEP homologue vers son DLCEP homologue correspondant, ou
   – d'un DLCEP abonné vers son DLCEP éditeur correspondant,

   doit demander des informations d'état issues du DLCEP adressé et doit demander que des données utilisateur de DLS soient incluses dans la DLPDU de réponse, et

b) la seconde adresse, si elle est présente,
   – doit être l'adresse de DLCEP homologue de la même DLC que l'adresse de DLCEP homologue de destination, ou
   – doit être une adresse de DLSAP appelante du DLCEP abonné sur la même DLC que l'adresse de DLCEP éditeur de destination;

c) le champ "paramètres SD" doit spécifier les informations d'état pour le DLCEP adressé, et le contenu de ce champ doit être tel que décrit en 7.4;

   NOTE La taille et la structure de ce champ dépendent des attributs QoS associés au DLCEP adressé par l'adresse de DL de destination spécifiée dans cette DLPDU, et sont déterminées au cours de l'établissement du DLCEP.

d) les données utilisateur doivent spécifier les octets d'une DLSDU cohérents avec la taille de DLSDU négociée et les informations de segmentation spécifiées dans les paramètres SD d'accompagnement, et ne doivent pas être vides.

   NOTE Si le champ de données utilisateur n'est pas vide, il convient d'utiliser une DLPDU CD (voir 6.5) à la place.

NOTE Les formats 1L et 1S sont utilisés dans des communications peer-to-peer (voir 6.6.2b) et abonné vers éditeur (voir 6.6.2c)) lorsque l'attribut d'authentification de la DLPDU est SOURCE ou MAXIMAL. Les formats 2L et 2S sont utilisés pour des communications peer-to-peer et abonné vers éditeur lorsque l'attribut d'authentification de la DLPDU est ORDINARY.

6.6.3 Envoi de la DLPDU ED

Une DLPDU ED peut être sélectionnée pour transmission sur la liaison quand

a) la DLE expéditrice détient un jeton programmeur ou jeton délégué qui est le jeton dominant sur la liaison locale;

b) la durée allouée restante de l'utilisation du jeton, C (RD), permet l'achèvement de V(MRC)+1 transactions implicites avant l'expiration du jeton, chaque transaction consistant à envoyer la DLPDU ED qui exige une réponse immédiate et à attendre une DLPDU SR de cas le plus défavorable ou une DLPDU de réponse DT autorisée de cas le plus défavorable ne contenant pas de données utilisateur de DLS;

c) s'il est prévu d'adresser la DLPDU ED à une adresse de DLSAP, alors le tableau des transactions en cours, V(OTA) (voir 4.7.1.15), scruté de façon circulaire à partir du dernier indice de transaction, V(LTI) (voir 4.7.1.16), possède une entrée non assignée dont l'indice n'est pas V(LTI).

Une fois sélectionnée, s'il est prévu d'adresser la DLPDU ED à une adresse de DLSAP, alors

– V(LTI) doit être mis à l'indice de cette entrée non assignée dans V(OTA);

– cette entrée dans V(OTA) doit être assignée à la DLPDU ED sélectionnée et doit enregistrer des informations qui permettent de corréler une DLPDU DT de réponse attendue à l'invocation spécifique du service d'échange d'unidata qui a généré la DLPDU ED.

Une fois sélectionnée, la transmission de la DLPDU ED doit être tentée de nouveau jusqu'à ce que soit

1) une DLPDU de réponse immédiate admissible soit reçue, soit
2) une DLPDU inadmissible soit reçue quand une réponse immédiate était attendue, soit
3) la transmission originale et le nombre maximal autorisé de répétitions de tentative de transmission, V(MRC) (voir 4.7.1.5), aient échoué à obtenir l'une des DLPDU de réponse admissibles.

Si la DLE détient un jeton délégué et aucune utilisation supplémentaire de ce jeton après envoi de cette DLPDU et attente de sa réponse immédiate n'est nécessaire à cet instant-là, la DLE peut mettre à la valeur FINAL le sous-champ "utilisation finale du jeton" de la DLPDU ED; sinon, ce sous-champ doit avoir la valeur NOT-FINAL.

NOTE La transaction FINAL est nécessairement celle de rang (V(MRC)+1) dans une série, autrement, si une réponse n'est pas reçue, une autre DLPDU ED est à envoyer avant la fin du cycle courant d'utilisation du jeton, rendant NOT-FINAL la transaction courante.

Chaque adresse de DL explicite dans la DLPDU ED doit être délocalisée (voir 5.2.2) avant transmission.

Après l'envoi d'une DLPDU ED, la DLE expéditrice doit surveiller la liaison locale pour détecter une réponse telle que spécifiée en 5.2.8.1. La DLPDU de réponse admissible est soit

– une DLPDU DT dont l'adresse de DL de destination est implicitement ou explicitement l'adresse de DLSAP d'origine ou le DLCEP spécifié par la DLPDU ED, soit
– une DLPDU DT sans adresse de DL de destination, soit
– une DLPDU SR.

Si V(LTI) a été assigné à la transaction, comme résultat de la recherche décrite en c), et si une DLPDU de réponse admissible n'a pas été reçue, l'entrée de rang V(LTI) doit être annulée.

6.6.4 Réception de la DLPDU ED

Chaque adresse de DL dans la DLPDU doit être délocalisée (voir 5.2.2) à la suite de la réception.

Une DLPDU ED reçue doit être traitée comme suit par la DLE destinataire.

6.6.4.1 Actions requises de toutes les DLE

NOTE La prochaine variante tente de détecter la réception d'une DLPDU ED en doublon résultant d'une répétition immédiate de tentative par la DLE détentrice de jeton courant, qui elle-même était probablement due à une erreur détectée au cours de la réception d'une DLPDU DT de réponse antérieure. Dans un tel cas, la DLPDU DT de réponse doit être identique à la première dans les champs et sous-champs qui acheminent les données utilisateur de DLS et des informations connexes.

Les considérations suivantes s'appliquent:

a) Si
   1) l'adresse de DL de destination spécifiée par la DLPDU désigne comme une adresse de DLSAP active de la DLE destinataire, ou une adresse de DLCEP active d'une DLC pour laquelle la DLE destinataire est un homologue ou un éditeur;

2) la DLPDU immédiatement antérieure a été émise comme réponse immédiate à une DLPDU ED reçue dont l'adresse de DL de destination était la même adresse de DLSAP ou la même adresse de DLCEP;

3) si une adresse de DL source est présente dans la DLPDU ED juste reçue, elle était également présente et identique dans cette DLPDU ED reçue antérieure;

4) en commençant par le champ "paramètres SD", les trois premiers octets de la DLPDU ED juste reçue, ou le reste de la DLPDU s'il y a moins de trois octets, sont identiques aux octets correspondants de cette DLPDU ED reçue antérieure;

5) une période d'inactivité de liaison de ("retard de récupération de réponse immédiate "+ 1) intervalles de temps,
à savoir (V(IRRD) + 1) × V(ST) durées d'octet, n'a pas eu lieu depuis la réception de cette DLPDU ED reçue antérieure,

alors, la DLE destinataire

i) doit rejeter la DLPDU reçue et ne pas la transmettre aux fonctions de niveau supérieur de la DLE pour un traitement ultérieur;

ii) doit réémettre la DLPDU DT de réponse immédiate émise antérieure, inchangée par rapport à la transmission antérieure, excepté pour.

   – du sous-champ "utilisation finale du jeton" de l'octet de contrôle de trame qui achemine les informations au LAS d'écoute;

   – les sous-champs de tous les éventuels paramètres SD de DLCEP présents qui acquittent la réception des DLSDU reçues ou en demandent la rtransmission.

NOTE Cette exigence est destinée à assurer que la priorité de DL, le contenu et l'identité de toutes les éventuelles données utilisateur de DLS acheminées dans la DLPDU DT de réponse immédiate sont identiques à celles de la DLPDU DT immédiatement antérieure envoyée à partir de cette même adresse de DL, incluant l'id de transaction de l'émetteur s'il y en a un présent dans les paramètres SD reçus et émis.

b) Si a) ne s'applique pas et l'adresse de DL de destination spécifiée par la DLPDU désigne une adresse de DLSAP de la DLE destinataire, alors le traitement de la DLPDU reçue doit être fondé sur le rôle de DL(SAP) spécifié pour cette adresse de DLSAP.

NOTE 1 Les adresses de DL de groupe ne sont pas couvertes ici; de telles DLPDU ED sont erronées et ne sont pas reconnues à la réception.

La DLE destinataire doit initier une réponse dans les limites d'une période de "retard maximal de réponse" intervalles de temps, à savoir V(MRD) × V(ST) durées d'octet, par rapport à la réception de la DLPDU ED

1) Si la DLE peut la préparer à temps, la DLPDU de réponse doit être une DLPDU DT

   – avec une adresse DL de destination explicite quelconque dans la DLPDU DT ayant la même valeur que l'adresse de DL source de la DLPDU ED reçue, et une longueur et un format comme suit:

      • DT (voir en 6.7.1 le format 2L) en réponse à ED (voir en 6.6.1 le format 1L), ou

      • DT (voir en 6.7.1 le format 2S) en réponse à ED (voir en 6.6.1 le format 1S) si la DLPDU DT contient des données utilisateur de DLS, ou

      •– DT (voir en 6.7.1 le format 4) en réponse à ED (voir en 6.6.1 le format 1S) si la DLPDU DT ne contient pas de données utilisateur de DL,

   et doit inclure les paramètres SD spécifiant

   – l'id de transaction de l'émetteur tel que reçu dans la DLPDU ED de stimulation, tel que spécifié en 6.6.2.1a)4);

   – le statut d'erreur de réception pour cette DLSDU acheminée par ED si la DLSDU n'a pas pu être placée en tampon ou en file d'attente; autrement, le statut pour la DLSDU demandée par cette DLPDU ED.

La DLSDU reçue doit être transmise aux fonctions de niveau supérieur de la DLE (voir

8.3) pour un traitement ultérieur;

Si la DLSDU reçue a été placée en tampon ou en file d'attente avec succès, la DLPDU de réponse peut inclure une DLSDU qui a déjà été mise en tampon dans cette DLE de répondeuse au moment de la réception de l'ED.

NOTE 2 Cette restriction interdit à la DLPDU de réponse de refléter un quelconque traitement de couche supérieure (à la DLL) de la DLSDU reçue. Cette restriction est nécessaire pour permettre la migration à partir des normes nationales antérieures.

Si le rôle de DL(SAP) de l'adresse de DLSAP de destination spécifiait

i) BASIC ou INITIATOR, la DLE doit rejeter la DLPDU ED reçue en fonction de ce rôle de DL(SAP), la DLSDU reçue doit être rejetée, et un statut d'erreur approprié sans DLSDU d'accompagnement doit être inclus dans la DLPDU DT de réponse;

ii) UNCONSTRAINED RESPONDER, alors

A) si aucun tampon ou aucune file d'attente de réception n'est explicitement lié(e) à l'adresse DLSAP de destination à la priorité DLL de la DLSDU reçue, ou si une file d'attente de réception est explicitement liée, mais est pleine, la DLSDU reçue doit être rejetée, et un statut d'erreur approprié sans DLSDU d'accompagnement doit être inclus dans la DLPDU DT de réponse; ou

B) autrement, la DLSDU reçue doit être placée dans ce tampon ou aboutée à cette file d'attente non pleine et le statut des paramètres SD doit indiquer le tampon expéditeur de la priorité DLL la plus élevée avec une DLSDU non vide qui était disponible au DLSAP adressé au moment de la réception de la DLPDU ED, et

I) si cette priorité DLL est supérieure ou égale à la priorité spécifiée en 6.6.1.1b), cette DLSDU doit être incluse dans la DLPDU DT de réponse et son tampon mis à vide si configuré ainsi, et la priorité DLL de cette DLPDU DT de réponse doit être la priorité DLL du DLSDU acheminée;

II) si cette priorité DLL est inférieure à la priorité spécifiée en 6.6.1.1b), aucune DLSDU ne doit être incluse dans la DLPDU DT de réponse, et la priorité DLL de cette DLPDU DT de réponse doit être la priorité DLL spécifiée en 6.6.1.1b); soit

III) s'il n'y a pas un tel tampon expéditeur avec une DLSDU non vide, la DLPDU DT de réponse doit indiquer un statut d'erreur approprié, la priorité DLL de cette DLPDU DT de réponse doit être la priorité DLL spécifiée en 6.6.1.1b), et cette DLPDU DT de réponse ne doit pas contenir de données utilisateur de DLS

iii) constrained responder, alors

A) si l'adresse de DLSAP source de la DLPDU ED est égale à l'adresse de DLSAP distante qui était spécifiée dans la primitive "request" de DL-BIND antérieure pour l'adresse de DLSAP destinataire (ou son équivalent de gestion de DL), après délocalisation pour les deux, alors la procédure indiquée en ii) doit être suivie; soit

B) autrement, la DLSDU reçue doit être supprimée, et un statut d'erreur approprié sans DLSDU d'accompagnement doit être inclus dans la DLPDU DT de réponse;

2) Si la DLE ne peut préparer à temps la DLPDU DT de réponse requise, la DLE doit envoyer une DLPDU DT avec une adresse DL de destination explicite quelconque dans la DLPDU DT ayant la même valeur que l'adresse de DL source de la DLPDU ED reçue, et une longueur et un format comme suit:

– DT (voir en 6.7.1 le format 2L) en réponse à ED (voir en 6.6.1 le format 1L), ou

– DT (voir en 6.7.1 le format 4) en réponse à ED (voir en 6.6.1 le format 1S),

et doit inclure les paramètres SD

– avec un id de transaction identique à l'id de transaction issu de la DLPDU ED reçue, telle que spécifiée en 6.6.2.1a)4);
– avec un statut (DR – “delayed reply”, c'est-à-dire "réponse différée") indiquant que la DLE exige du temps supplémentaire pour préparer la réponse requise;
– avec un champ "données utilisateur" vide,

et la DLE

– doit préparer cette DLPDU DT requise dès que possible, telle que juste décrite;
– doit inclure une adresse de destination explicite dans la DLPDU de réponse;
– doit abouter cette DLPDU de réponse à la Q(US) de la DLE devant être émise dès que possible;

c) Si a) ne s'applique pas et l'adresse de DL de destination spécifiée par la DLPDU désigne une adresse de DLCEP active d'une DLC pour laquelle la DLE destinataire
   – est un homologue ou un éditeur et la priorité DLL du DLCEP n'est pas identique à la priorité indiquée dans la DLPDU reçue; soit
   – est un homologue et la longueur et le nombre d'adresses de DL ne sont pas tels que prévus ou la DLPDU spécifie une adresse source explicite qui n'est pas égale à l'adresse de DLCEP du DLCEP homologue distant,

   alors

   1) la DLPDU reçue doit être transmise aux fonctions de niveau supérieur de la DLE (voir 8.2.2.9) pour un traitement ultérieur;
   2) la DLE destinataire doit initier une réponse dans les limites d'une période de "retard maximal de réponse" intervalles de temps, à savoir V(MRD) × V(ST) durées d'octet, par rapport à la réception de la DLPDU CD. La DLPDU de réponse doit être une DLPDU DT au format négocié pour la DLC pour le sens choisi de transmission et doit contenir les paramètres SD appropriés au DLCEP expéditeur, mais ne doit pas contenir de donnée utilisateur de DLS.

d) Si ni a) ni b) ne s'appliquent et l'adresse DL de destination spécifiée par la DLPDU désigne une adresse de DLCEP active d'une DLC pour laquelle la DLE destinataire est un homologue ou un éditeur, alors
   1) la DLPDU reçue doit être transmise aux fonctions de niveau supérieur de la DLE (voir 8.2.2.9) pour un traitement ultérieur;
   2) la DLE destinataire doit initier une réponse dans les limites d'une période de "retard maximal de réponse" intervalles de temps, à savoir V(MRD) × V(ST) durées d'octet, par rapport à la réception de la DLPDU ED. La DLPDU de réponse doit être une DLPDU DT au format négocié pour la DLC pour le sens choisi de transmission et doit contenir les paramètres SD appropriés au DLCEP expéditeur, et doit contenir des données utilisateur de DLS, s'il y en a qui
      – étaient disponibles et attendaient pour transmission ou retransmission à partir du DLCEP;
      – peuvent être incluses dans la DLPDU DT tout en respectant le retard maximal de réponse requis;
   3) la DLPDU DD demandeuse peut contenir des informations concernant l'état du DLCEP demandeur. La DLPDU DT de réponse a la permission, mais n'est pas tenue, de refléter ces informations d'état dans sa réponse; le traitement immédiat de ces informations d'état avant l'envoi de la réponse immédiate doit être permis, mais ne doit pas être exigé.

   La DLPDU reçue doit être transmise aux fonctions de niveau supérieur de la DLE (voir

8.2.2.9) pour un traitement ultérieur;

e) Si a) ne s'applique pas et l'adresse de DL de destination spécifiée par la DLPDU désigne une adresse de DLCEP active d'une DLC pour laquelle la DLE destinataire est un abonné, alors
   1) la partie autre que des données utilisateur de DLS de la DLPDU reçue doit être transmise aux fonctions de niveau supérieur de la DLE (voir 8.2.2.9) pour un traitement ultérieur;
   2) la DLE destinataire doit enregistrer l'adresse de DL de destination issue de la DLPDU ED reçue dans V(RA) pour association ultérieure avec la DLPDU DT de réponse immédiate attendue, qui est recommandée être la prochaine DLPDU reçue;
   3) la DLE destinataire doit surveiller la liaison locale pour détecter une réponse, puis agir selon le résultat de cette surveillance, tout cela tel que spécifié en 5.2.8.3.

6.6.4.2 Actions supplémentaires requises d'une DLE de classe Link-Master

Aucune.

6.6.4.3 Actions supplémentaires requises d'une DLE de classe Bridge

Les considérations suivantes s'appliquent:

a) Étant donné que chaque DLE de classe Bridge a une capacité de maître de liaison, toutes les actions spécifiées en 6.6.4.2 s'appliquent également à une DLE de la classe Bridge.
b) Si toutes les conditions suivantes sont remplies
   A) l'adresse de DL de destination spécifiée dans la DLPDU est une adresse qu'il convient que le pont transmette, mais vers laquelle la DLE pont elle-même ne générerait pas autrement de DLPDU de réponse immédiate;
   B) la DLPDU immédiatement antérieure a été transmise comme réponse immédiate à une DLPDU ED reçue avec la même adresse de DL de destination;
   C) si une adresse de DL source est présente dans la DLPDU ED juste reçue, elle était également présente et identique dans cette DLPDU ED reçue antérieure;
   D) la partie "paramètres DLC de base" des paramètres SD de la DLPDU ED juste reçue est non vide et est identique à celle de cette DLPDU ED reçue antérieure;
   E) une période d'inactivité de liaison de (immediate-response-recovery-delay plus un) intervalles de temps, à savoir (V(IRRD) +1) × V(ST) durées d'octet, n'a pas eu lieu depuis la réception de cette DLPDU ED reçue antérieure,

   alors le pont

   1) doit rejeter la DLPDU reçue et ne pas la transmettre aux autres fonctions du pont pour un traitement ultérieur;
   2) doit lancer la retransmission de la DLPDU SR de réponse immédiate émise antérieure, inchangée par rapport à la transmission antérieure.

   NOTE Cette exigence est destinée à garantir que tout statut acheminé dans la DLPDU SR de réponse immédiate est identique à celui qui est contenue dans la DLPDU SR antérieure.

   Autrement,

   i) si l'adresse de DL de destination spécifiée dans la DLPDU est une adresse qu'il convient que le pont transmette, mais que la DLE pont elle-même ne recevrait autrement pas, le pont doit former et envoyer une DLPDU SR
      – dans les limites d'une période de maximum-response-delay intervalles de temps, à savoir V(MRD) × V(ST) durées d'octet, par rapport à la réception de la DLPDU ED;
      – avec un statut indiquant si, oui ou non, le pont a été capable de placer en tampon la DLPDU reçue.

ii) si l'adresse DL de destination spécifiée dans la DLPDU est une adresse qu'il convient que le pont transmette et le pont a été capable de recevoir et placer en tampon la DLPDU sans erreur, alors la DLPDU reçue doit être, au besoin, transmise avec la modification du champ "contrôle de trame" dans la DLPDU transmise (voir 5.2.5.3).

iii) si la DLPDU contient une adresse de DL source explicite, le pont doit tenter de mettre à jour son entrée de table de routage pour l'adresse de DL source spécifiée dans la DLPDU afin de refléter le port du pont à partir duquel la DLPDU a été reçue.

c) C'est une erreur de protocole pour une DLE pont qui transmet la DLPDU ED que de ne pas envoyer une DLPDU de réponse SR lorsqu'une réponse est exigée.

NOTE Il convient qu'au maximum une seule DLE pont sur la liaison locale transmette la DLPDU ED reçue.

6.6.4.4 Actions supplémentaires requises de la DLE LAS courante

La DLE LAS doit agir comme spécifié en 5.2.8.2.

6.7 DLPDU Data (Données(DT))

Une DLPDU DATA (DT) est utilisée pour transférer une quantité limitée de données utilisateur transparentes à partir d'un utilisateur de DLS vers un ou plusieurs autres utilisateurs de DLS; pour acquitter le transfert de ces données; et pour aider à la synchronisation des DLCEP et des utilisateurs de DLS.

Elle est également utilisée par une DLE répondeuse pour répondre à une DLPDU CA, CD ou ED reçue, lorsque le temps autorisé par V(ST) et V(MRD) (voir 4.7.1.1 et 4.7.1.2) est inadéquate pour que cette DLE génère la réponse requise à la DLPDU reçue.

Elle est également utilisée par une DLE pour envoyer une SPDU à une ou plusieurs autres DLE.

6.7.1 Structure des DLPDU DT

Tableau 18 – Structure des DLPDU DT

Format	Contrôle de trame	Adresse de destination	Adresse source	Paramètres	User data (données d'utilisateur)
1L	1101 1FPP	HL.N.S	HL.N.S	SD-p	o-pDLSDU
1s	1101 0FPP	N.S	N.S	SD-p	o-pDLSDU
2L	1001 1FPP	HL.N.S	—	SD-p	o-pDLSDU
2s	1001 0FPP	N.S	—	SD-p	o-pDLSDU
3L	0101 1FPP		HL.N.S	SD-p	o-pDLSDU
3s	0101 0FPP		N.S	SD-p	o-pDLSDU
4	1001 0F00	[PSA]	—	SD-p	o-pDLSDU
5	0101 0F00		[PDA]	SD-p	o-pDLSDU

où

[PDA] est l'adresse de DL de destination issue de la DLPDU CA, CD ou ED immédiatement antérieure,

[PSA] est l'adresse de DL de source implicite issue de la DLPDU CA, CD ou ED immédiatement antérieure.

6.7.1.1 Champ "Frame control" (contrôle de trame)

Le champ "contrôle de trame" doit spécifier

a) la fonction de la DLPDU;
b) la priorité de la DLPDU;
c) la longueur, le nombre et le type d'adresses de DLPDU;
d) si, oui ou non, la transmission de la DLPDU courante met fin à l'utilisation d'un jeton délégué.

6.7.1.2 Champ "Address" (adresse)

Le champ "adresse" doit être constitué soit

a) d'une adresse de DL de destination explicite et d'une adresse de DL source explicite, dans cet ordre, pour les formats 1L et 1S,
b) de seulement une adresse de DL destination explicite, pour les formats 2L et 2S,
c) de seulement une adresse de DL source explicite, pour les formats 3L et 3S,
d) de seulement une adresse de DL de destination implicite, pour le format 4, soit
e) de seulement une adresse de DL source implicite, pour le 5.

Pour les formats 1L, 2L et 3L, toutes les adresses doivent être LONG; pour les formats 1S, 2S et 3S, toutes les adresses doivent être SHORT; pour les formats 4 et 5 toutes les adresses doivent être VERY-SHORT (voir 5.2.1.1) (c'est-à-dire implicite).

6.7.1.3 Champ "Parameters" (paramètres)

Lorsque l'adresse de DL de destination spécifie explicitement une adresse DL de groupe, le champ "paramètres des données de statut" (paramètres SD) du champ est vide (formats 1L et 1S), tels que décrits en 7.4.1.

Lorsque l'adresse de DL de destination spécifie, explicitement ou implicitement, une adresse de DLSAP, le champ "paramètres SD" spécifie un id de transaction et un statut (formats 2L, 2S, 4), est vide (formats 1L et 1S), tels que décrits en 7.4.2.

Lorsque la première adresse de DL spécifie, explicitement ou implicitement, une adresse de DLCEP, le champ "paramètres SD" spécifie des informations d'état pour le DLCEP adressé par l'adresse de DL de destination (formats 1L, 1S, 2L, 2S, 4), ou l'adresse de DL source (formats 3L, 3S, 5), tels que décrits en 7.4. La taille et la structure de ce champ dépendent des attributs QoS associés au DLCEP adressé et sont déterminées au cours de l'établissement du DLCEP.

6.7.1.4 Champ "User data" (données d'utilisateur)

Le champ "données d'utilisateur" doit être constitué d'une seule ou partielle DLSDU facultative dont la taille maximale est limitée à la plus petite parmi

a) la taille maximale de données utilisateur de DLS autorisée pour une DLPDU de la priorité spécifiée en 6.7.1.1b);
b) lorsque l'adresse de destination (ou source) implicite ou explicite de la DLPDU est une adresse de DLCEP, la taille maximale de DLSDU négociée sur la DLC pour ce sens de transmission de données vers (ou depuis) le DLCEP.

6.7.2 Contenu de la DLPDU DT

Le champ "contrôle de trame" doit être codé comme spécifié dans le Tableau 18.

Soit les adresses de DL doivent être

a) la première une adresse de DL de groupe et la deuxième une adresse de DLSAP,
b) toutes des adresses de DLSAP,
c) toutes des adresses de DLCEP, ou
d) la première une adresse de DLCEP et la deuxième une adresse de DLSAP.

6.7.2.1 Contenu de la DLPDU DT lors de la spécification d'une adresse de DL(SAP) de destination

Lorsque la première adresse est une adresse de DL de groupe comme en 6.7.2a), alors

a) Si le format de DLPDU est le format 1L ou 1S, alors
   1) la DLPDU est utilisée pour mettre en œuvre le service de transfert d'unitdata;
   2) le rôle de DL(SAP) pour l'adresse de DLSAP de destination doit être GROUP;
   3) le rôle de DL(SAP) pour l'adresse de DLSAP source doit être BASIC;
   4) le champ "paramètres SD" doit être vide;
   5) les données utilisateur doivent être une DLSDU unique dont la taille est limitée à la taille maximale pour la priorité spécifiée en 6.7.1.1b), et ne doivent pas être vides.
b) Aucun autre format de DLPDU ne peut être utilisé.

Lorsque les adresses sont des adresses de DLSAP comme en 6.7.2b), alors

c) Si le format de DLPDU est le format 1L ou 1S, alors
   1) cette DLPDU est utilisée pour mettre en œuvre le service de transfert d'unitdata;
   2) le rôle de DL(SAP) pour l'adresse de DLSAP de destination doit être BASIC;
   3) le rôle de DL(SAP) pour l'adresse de DLSAP source doit être BASIC;
   4) le champ "paramètres SD" doit être vide;

5) les données utilisateur doivent être une DLSDU unique dont la taille est limitée à la taille maximale pour la priorité spécifiée en 6.7.1.1b), et ne doivent pas être vides.

d) Si le format de DLPDU est le format 2L, 2S ou 4, alors

1) cette DLPDU achemine un acquittement ou une réponse comme partie intégrante du service de transfert d'unitdata (avec confirmation de DLE à distance), ou du service d'échange d'unitdata;
2) le rôle de DL(SAP) pour l'adresse de DLSAP de destination doit être BASIC ou INITIATOR;
3) le rôle DL(SAP) pour l'adresse DLSAP source implicite est BASIC ou CONSTRAINED RESPONDER ou UNCONSTRAINED RESPONDER;
4) le champ "paramètres SD" doit spécifier un id de transaction et un statut de réponse pour le DLSAP adressé, et le contenu de ce champ doit être tel que décrit en 7.4;
5) les données utilisateur soit doivent être une DLSDU unique dont la taille est limitée à la taille maximale pour la priorité spécifiée en 6.7.1.1b), soit doivent être vides, comme suit:
   – les formats 2L peuvent être vides ou non vides;
   – le format 2S doit toujours être non vide;
   – le format 4 doit toujours être vide.

e) Aucun autre format de DLPDU ne peut être utilisé.

6.7.2.2 Contenu de la DLPDU DT lors de la spécification d'une adresse de DLCEP source ou de destination

Lorsque la première adresse est une adresse de DLCEP comme en 6.7.2b) et 6.7.2d), alors

a) cette DLPDU peut transporter une DLSDU unique ou partielle
   – d'un DLCEP homologue vers son DLCEP homologue correspondant,
   – d'un DLCEP abonné vers son DLCEP éditeur correspondant, ou
   – d'un DLCEP éditeur vers ses DLCEP abonnés correspondants,

et

b) la seconde adresse, si elle est présente,
   – doit être l'adresse de DLCEP homologue de la même DLC que l'adresse de DLCEP de destination, ou
   – doit être une adresse de DLSAP appelante d'un DLCEP abonné;

c) le champ "paramètres SD" doit spécifier les informations d'état pour le DLCEP adressé, et le contenu de ce champ doit être tel que décrit en 7.4.2;

   NOTE La taille et la structure de ce champ dépendent des attributs QoS associés au DLCEP adressé par l'adresse de DL de destination spécifiée dans cette DLPDU, et sont déterminées au cours de l'établissement du DLCEP.

d) les données utilisateur doivent spécifier les octets d'une DLSDU cohérents avec la taille de DLSDU négociée et les informations de segmentation spécifiées dans les paramètres SD d'accompagnement, et peuvent être vides.

NOTE 1 Les formats 1L, 2L, 1S et 2S sont utilisés pour des communications peer-to-peer et abonné vers éditeur; 1L est utilisé lorsque l'attribut DLPDU-authentication (authentification de DLPDU) est SOURCE ou MAXIMAL, 1S est utilisé lorsque l'attribut d'authentification de la DLPDU est SOURCE, 2L et 2S sont utilisés lorsque l'attribut d'authentification de la DLPDU est ORDINARY.

Le format 3L est utilisé pour les communications éditeur vers abonné lorsque l'attribut d'authentification de la DLPDU est MAXIMAL. Les formats 3L et 3S sont utilisés pour les communications éditeur vers abonné lorsque l'attribut d'authentification de la DLPDU est ORDINARY ou SOURCE.

Le format spécifique devant être utilisé (parmi les formats 1L à 3S) est déterminé comme partie intégrante de l'établissement du DLCEP.

NOTE 2 Les formats 4 et 5 peuvent respectivement être utilisés en lieu et place des formats 2s et 3s uniquement lorsque la DLE expéditrice détient un jeton de réponse et lorsque l'attribut d'authentification de la DLPDU est ORDINARY.

6.7.3 Envoi de la DLPDU DT

Une DT DLPDU peut être sélectionnée pour transmission sur la liaison lorsque la DLE expéditrice

a) vient juste de recevoir un jeton de réponse dans une DLPDU CA, CD ou ED, permettant une seule transmission d'une DLPDU DT ou SR; ou
b) détient un jeton programmeur ou un jeton délégué qui est le jeton dominant sur la liaison locale, et lorsque la durée allouée restante d'utilisation du jeton, C(RD), permet l'achèvement de la transmission de la DLPDU DT avant l'expiration du jeton.

Chaque adresse de DL explicite dans la DLPDU doit être délocalisée (voir 5.2.2) avant transmission.

6.7.3.1 Transmission lorsque le jeton de réponse est dominant

Les considérations suivantes s'appliquent:

a) Une DLPDU DT peut être envoyée sur la liaison lorsque la DLE expéditrice a reçu une DLPDU CD ou ED adressée
   – à l'une de ses adresses de DLSAP actives, ou
   – à l'une de des adresses de DLCEP actives pour laquelle elle a un DLCEP homologue ou éditeur,

   et la DLE expéditrice répond comme spécifié en 6.5.4.1 ou en 6.6.4.1, en formant comme réponse immédiate une DLPDU DT qui peut comporter une DLSDU qui était déjà placée tampon ou en file d'attente en cette DLE répondeuse au moment de la réception de la DLPDU CD ou ED.

   NOTE Cette restriction interdit à la DLPDU de réponse de refléter un quelconque traitement de couche supérieure de la DLSDU reçue. Cette restriction est nécessaire pour permettre la migration à partir des normes nationales antérieures.
b) Une DLPDU DT peut être envoyée sur la liaison lorsque la DLE expéditrice a reçu une DLPDU CA adressée
   – à l'une de ses adresses de DLSAP actives, ou
   – à l'une de des adresses de DLCEP actives pour laquelle elle a un DLCEP homologue ou éditeur,

   et la DLE expéditrice répond comme spécifié en 6.4.4.1, en formant comme réponse immédiate une DLPDU DT qui ne comporte pas de DLSDU.

Lorsqu'une réponse immédiate à une DLPDU CA, CD ou ED est requise, comme spécifié en a) ou b), la DLE de réponse doit envoyer une DLPDU DT de réponse dans les limites d'une période de maximum-response-delay intervalles de temps, à savoir V(MRD) $\times$ V(ST) durées d'octet, par rapport à la réception de la DLPDU CA, CD ou ED demandeuse.

Si la DLPDU CA, CD ou ED qui a exigé la réponse a été adressée à un DLCEP, cette DLPDU CA, CD ou ED peut avoir contenu des paramètres SD qui ont acheminé des informations relatives à l'état du DLCEP expéditeur vers le DLCEP destinataire. La DLE destinataire a la permission, mais n'est pas tenue, de refléter ces informations d'état dans sa DLPDU DT de réponse; le traitement immédiat de ces informations d'état avant l'envoi de la réponse immédiate doit être permis, mais ne doit pas être exigé.

Le champ "utilisation finale du jeton" de la DLPDU DT de réponse doit avoir la même valeur que celle qui est contenue dans la DLPDU CA, CD ou ED demandeuse.

Chaque adresse de DL explicite dans la DLPDU DT de réponse doit être délocalisée (voir 5.2.2) avant transmission.

C'est une erreur de protocole pour une DLE adressée que de ne pas envoyer une DLPDU de réponse DT lorsqu'une réponse est exigée.

NOTE Il convient qu'au maximum une DLE sur la liaison locale envoie une réponse à la DLPDU CA, CD ou ED reçue. Cette réponse peut être une DLPDU soit DT, soit SR.

6.7.3.2 Transmission lorsque le jeton délégué est dominant

Une DLPDU DT peut être envoyée sur la liaison lorsque la DLE expéditrice détient un jeton délégué qui est le jeton dominant sur la liaison locale, et lorsque la durée allouée restante d'utilisation du jeton, C(RD), permet l'achèvement de la transmission de la DLPDU avant l'expiration du jeton.

Si aucune utilisation supplémentaire de ce jeton après l'envoi de cette DLPDU n'est nécessaire à cet instant-là, la DLE peut mettre à la valeur FINAL le sous-champ "utilisation finale du jeton" de la DLPDU DT; sinon, ce sous-champ doit avoir la valeur NOT-FINAL.

Chaque adresse de DL explicite dans la DLPDU doit être délocalisée (voir 5.2.2) avant transmission.

6.7.3.3 Transmission lorsque le jeton programmeur est dominant

La DLE LAS peut envoyer une DLPDU DT sur la liaison lorsque la durée allouée restante d'utilisation du jeton avant la prochaine activité programmée et la durée allouée restante d'utilisation du jeton programmeur pour la maintenance de la liaison, qui est V(LTHT) durées d’octet moins la quantité de capacité de la liaison utilisée pour la maintenance de la liaison au cours de l'actuel cycle de "circulation du jeton", permettent toutes deux l'achèvement de la transmission de la DLPDU avant l'expiration du jeton.

Chaque adresse de DL explicite dans la DLPDU doit être délocalisée (voir 5.2.2) avant transmission.

6.7.4 Réception de la DLPDU DT

Chaque adresse de DL dans la DLPDU doit être délocalisée (voir 5.2.2) à la suite de la réception.

Une DLPDU DT reçue doit être traitée comme suit par la DLE destinataire.

6.7.4.1 Actions requises de toutes les DLE

6.7.4.1.1 Actions requises lorsque le jeton de réponse n'était pas dominant au début de la réception

a) Si la DLPDU DT reçu est au format 1L ou 1S, et son adresse de DL de destination désigne une adresse de DL(SAP) de la DLE destinataire, alors la DLPDU reçue doit être transmise aux fonctions de niveau supérieur de la DLE (voir 8.3.1.3) pour un traitement ultérieur.

b) Si la DLPDU DT reçue est au format 2L ou 2S et si son adresse de DL de destination désigne une adresse de DLSAP de la DLE destinataire et l'id de transaction rapporté par les paramètres SD indexe une transaction assignée dans la matrice des transactions en cours, V(OTA) (voir 4.7.1.15), alors
   - la DLE doit considérer que la transmission antérieure était exempte d'erreur;
   - les informations contenues dans l'entrée indexée dans V(OTA) doivent être associées à la DLPDU DT reçue;
   - la DLPDU DT reçue et ces informations associées doivent être transmises aux fonctions de niveau supérieur de la DLE (voir 8.3) pour un traitement ultérieur;

– l'entrée indexée dans V(OTA) doit être non affectée.

c) Si la DLPDU DT reçue

1) a le format 1L, 1S, 2L ou 2S et son adresse de DL de destination désigne une adresse de DLCEP désignant un DLCEP homologue ou éditeur de la DLE destinataire; ou

2) a le format 3L ou 3S et son adresse de DL source désigne une adresse de DLCEP désignant un DLCEP abonné de la DLE destinataire,

alors la DLPDU reçue doit être transmise aux fonctions de niveau supérieur de la DLE (voir 8.2) pour un traitement ultérieur.

d) Si la DLPDU DT a le format 1L ou 1S et son adresse de DL de destination désigne une adresse de DL NODE.0, V(TN).0, des fonctions de prise en charge DL de la DLE destinataire, alors la DLPDU reçue doit être transmise aux fonctions de niveau supérieur de la DLE (voir 10.3) pour un traitement ultérieur.

e) Si aucun des points a), b), c) ou d) ne s'applique, la DLPDU DT doit être rapportée à la gestion de DL locale comme étant une réponse inattendue, et doit être rejetée.

NOTE Ce rapport peut prendre la forme consistant à incrémenter un compteur d'erreurs de gestion de DL.

6.7.4.1.2 Actions requises lorsque le jeton de réponse était dominant au début de la réception et la DLE destinataire avait envoyé la DLPDU CA, CD ou ED qui a créé le jeton de réponse

a) Si la DLPDU DT reçue

– a le format 2L ou 2S et son adresse de DL de destination désigne une adresse de DLSAP de la DLE destinataire; ou

– a le format 4 et son adresse de DL source issue de la DLPDU CA, CD ou ED immédiatement antérieure était une adresse de DLSAP,

alors

1) Si le statut rapporté par les paramètres SD est "DR – delayed reply (réponse différée)" (c'est-à-dire qu'il a la valeur F_{16}), alors la DLE

– doit considérer que la transmission antérieure était exempte d'erreur;

– doit rejeter la DLPDU.

NOTE Il convient que la DLE s'attende à recevoir une autre DLPDU DT plus tard (avec la même adresse de DL de destination et le même format) comme réponse différée.

2) Si 1) ne s'applique pas, alors

– la DLE doit considérer que la transmission antérieure était exempte d'erreur;

– les informations contenues dans l'entrée de rang V(LTI) dans V(OTA) (voir 4.7.1.16 et 4.7.1.15) doivent être associées à la DLPDU DT reçue;

– la DLPDU DT reçue et ces informations associées doivent être transmises aux fonctions de niveau supérieur de la DLE (voir 8.3) pour un traitement ultérieur;

– l'entrée de rang V(LTI) dans V(OTA) doit être désassignée.

b) Si la DLPDU DT a le format 1L, 1S, 2L ou 2S et son adresse de DL de destination désigne une adresse de DLCEP d'un DLCEP homologue de la DLE destinataire, alors

– la DLE doit considérer que la transmission antérieure était exempte d'erreur;

– la DLPDU DT reçue doit être transmise aux fonctions de niveau supérieur de la DLE (voir 8.2) pour un traitement ultérieur.

c) Si la DLPDU DT reçue a le format 4 et l'adresse de DL source, implicite ou explicite, issue de la DLPDU CA, CD ou ED immédiatement antérieure était une adresse de DLCEP d'un DLCEP homologue de la DLE destinataire, alors

– la DLE doit considérer que la transmission antérieure était exempte d'erreur;

- la DLPDU DT reçue doit être transmise aux fonctions de niveau supérieur de la DLE (voir 8.2) pour un traitement ultérieur, avec son adresse DLCEP de destination implicite supposée être l'adresse de DL source, explicite ou implicite, issue de cette DLPDU CA, CD ou ED immédiatement antérieure.

d) Si la DLPDU DT reçue a le format 3L ou 3S et son adresse de DL source désigne l'adresse de DLCEP d'éditeur d'un DLCEP abonné de la DLE destinataire et cette adresse de DL source est égale à l'adresse de DL de destination issue de la DLPDU CA, CD ou ED immédiatement antérieure, alors

- la DLE doit considérer que la transmission antérieure était exempte d'erreur;
- la DLPDU DT reçue doit être transmise aux fonctions de niveau supérieur de la DLE (voir 8.2) pour un traitement ultérieur.

e) Si la DLPDU DT reçue a le format 5 et l'adresse de DL de destination explicite issue de la DLPDU CA, CD ou ED immédiatement antérieure était une adresse de DLCEP d'un DLCEP abonné de la DLE destinataire, alors

- la DLE doit considérer que la transmission antérieure était exempte d'erreur;
- la DLPDU DT reçue doit être transmise aux fonctions de niveau supérieur de la DLE (voir 8.2) pour un traitement ultérieur, avec son adresse DLCEP source implicite supposée être l'adresse de DL de destination explicite issue de cette DLPDU CA, CD ou ED immédiatement antérieure.

f) Si aucun des points a) à e) ne s'applique, la DLPDU DT doit être rapportée à la gestion de DL locale comme étant une réponse inattendue, et doit être rejetée.

NOTE Ce rapport peut prendre la forme consistant à incrémenter un compteur d'erreurs de gestion de DL.

6.7.4.1.3 Actions requises lorsque le jeton de réponse était dominant au début de la réception et la DLE destinataire n'avait pas envoyé la DLPDU CA, CD ou ED qui a créé le jeton de réponse

a) Si la DLPDU DT reçue a le format 3L ou 3S et son adresse de DL source désigne une adresse de DLCEP d'éditeur d'un DLCEP abonné de la DLE destinataire et cette adresse de DL source est égale à l'adresse de DL de destination, V(RA), issue de la DLPDU CA, CD ou ED immédiatement antérieure, alors

- la DLE doit considérer que la transmission antérieure était exempte d'erreur;
- la DLPDU DT reçue doit être transmise aux fonctions de niveau supérieur de la DLE (voir 8.2) pour un traitement ultérieur.

b) Si la DLPDU DT reçue a le format 5 et l'adresse de DL de destination explicite issue de la DLPDU CA, CD ou ED immédiatement antérieure était une adresse de DLCEP d'éditeur d'un DLCEP abonné de la DLE destinataire, alors

- la DLE doit considérer que la transmission antérieure était exempte d'erreur;
- la DLPDU DT reçue doit être transmise aux fonctions de niveau supérieur de la DLE (voir 8.2) pour un traitement ultérieur, avec son adresse DLCEP source implicite supposée être l'adresse de DL de destination explicite issue de cette DLPDU CA, CD ou ED immédiatement antérieure.

6.7.4.2 Actions supplémentaires requises d'une DLE de classe Link-Master

Aucune.

6.7.4.3 Actions supplémentaires requises d'une DLE de classe Bridge

a) Étant donné que chaque DLE de classe Bridge a une capacité de maître de liaison, toutes les actions spécifiées en 6.7.4.2 s'appliquent également à une DLE de la classe Bridge.

b) Si la première adresse DL spécifiée dans la DLPDU est une adresse de DL explicite vers laquelle il convient que le pont transmette la DLSDU et le pont a été capable de placer en tampon la DLPDU sans erreur, alors la DLPDU doit être, au besoin, transmise avec la modification du champ "contrôle de trame" dans la DLPDU transmise (voir 5.2.5.3).

c) Si la DLPDU contient une adresse de DL source explicite, le pont doit tenter de mettre à jour son entrée de table de routage pour l'adresse de DL source spécifiée dans la DLPDU afin de refléter le port du pont à partir duquel la DLPDU a été reçue.

d) Autrement, si ni b) ni c) ne s'applique, la DLE ne doit pas transmettre la DLPDU.

6.7.4.3.1 Actions requises lorsque le jeton de réponse était dominant au début de la réception et la DLE pont destinataire avait transmis, sans en être l'origine, la DLPDU CA, CD ou ED qui a créé le jeton de réponse

a) La DLE doit considérer que la transmission antérieure était exempte d'erreur.

b) Si

   – la DLPDU DT reçue a le format 4;

   – l'adresse de DL de destination issue de la DLPDU CA, CD ou ED immédiatement antérieure était une adresse de DLSAP;

   – le statut rapporté par les paramètres SD est autre que "DR – delayed reply (réponse différée)" (c'est-à-dire qu'il a une valeur autre que F_{16});

   – le champ "données utilisateur" est vide,

   alors la DLE doit former une DLPDU DT

   1) avec le format 2L lorsque la précédente DLPDU CA, CD ou ED était au format 1L, ou le format 2S lorsque la précédente DLPDU CA, CD ou ED était au format 1S;

   2) avec l'adresse de DL de destination explicite égale à l'adresse de DL source de cette DLPDU CA, CD ou ED immédiatement antérieure;

   3) avec un champ "paramètres SD" égal au champ "paramètres SD" de la DLPDU DT reçue;

   4) avec un champ "données utilisateur" vide;

   et doit transmettre la DLPDU DT juste formée comme si elle avait juste été créée par les fonctions de niveau supérieur de la DLE pont.

   NOTE Cela signifie que la DLE pont est à l'origine de la FCS pour la DLPDU DT nouvellement formée comme en 5.2.5.1, plutôt que de simplement modifier une FCS précédemment reçue au cours de la transmission normale comme en 5.2.5.3.

6.7.4.4 Actions supplémentaires requises de la DLE LAS courante

Si le sous-champ "utilisation finale de jeton" de la DLPDU reçue a la valeur FINAL, la DLE LAS doit

a) supposer que l'utilisation courante du jeton délégué est terminée comme si le jeton avait été retournée par une DLPDU RT (voir 6.17);

b) supposer que le jeton programmeur est de nouveau dominant sur la liaison locale et reprendre le fonctionnement actif comme LAS.

6.8 DLPDU "Status response" (réponse de statut (SR))

Une DLPDU STATUS RESPONSE (Réponse de statut (SR)) n'est envoyée qu'en détenant un jeton de réponse; elle est utilisée

a) pour indiquer la réception de la DLPDU CA, CD ou ED immédiatement antérieure par le pont qui transmettrait normalement cette DLPDU vers la DLE adressée, pour indiquer à la DLE expéditrice qu'aucune erreur ne s'est produite, ou que l'erreur indiquée s'est produite;;

b) pour rejeter une tentative de transfert du rôle LAS de la DLE LAS courante vers une autre DLE Link-Master (maître de liaison).

6.8.1 Structure de la DLPDU SR

Tableau 19 – Structure des DLPDU SR

Contrôle de trame	Adresse de destination	Adresse NODE source	Paramètres
0001 0F11	[PSA]	N	o-SR-p
où [PSA] est l'adresse de DL implicite égale à l'adresse de DL source implicite ou explicite de la DLPDU immédiatement antérieure sur la liaison.			

6.8.1.1 Champ "Frame control" (contrôle de trame)

Le champ "contrôle de trame" doit spécifier

a) la fonction de la DLPDU;

b) la priorité implicite de la DLPDU, qui est celle de la DLPDU immédiatement antérieure sur la liaison locale (pour laquelle la DLPDU SR est une réponse immédiate);

c) la longueur, le nombre et le type d'adresses de DLPDU;

d) si, oui ou non, la transmission de la DLPDU courante met fin à l'utilisation d'un jeton délégué.

6.8.1.2 Champ "Address" (adresse)

Le champ "adresse" doit consister en une adresse de DL NODE source explicite.

6.8.1.3 Champ "Parameters" (paramètres)

Le champ "paramètres de réponse différée" (paramètres SR) spécifie un statut et un id de transaction facultatif. S'il n'est pas vide, ce champ doit être structuré et codé tel que décrit en 7.5.

6.8.1.4 Champ "User data" (données d'utilisateur)

Le champ "données d'utilisateur" doit être vide.

6.8.2 Contenu de la DLPDU SR

Le champ "contrôle de trame" doit être codé comme spécifié dans le Tableau 19.

– Lorsqu'il est envoyé comme réponse immédiate à une DLPDU TL, le sous-champ "utilisation finale du jeton" de la DLPDU SR doit spécifier FINAL.

– Lorsqu'il est envoyé comme réponse immédiate à une DLPDU CA, CD ou ED, le sous-champ "utilisation finale du jeton" de la DLPDU SR doit avoir la même valeur que celui dans la DLPDU CA, CD ou ED demandeuse.

Le champ "adresse" doit être l'adresse de DL NODE de la DLE de réponse, dont la valeur est l'id de nœud de l'expéditeur, V(TN).

NOTE L'objectif principal de l'inclusion de l'adresse de DL NODE est d'identifier la DLE qui a assumé le jeton réponse à un analyseur de liaison d'observation.

Lorsqu'il est envoyé comme réponse immédiate à une DLPDU CA, CD ou ED,

a) les paramètres SR doivent être vides lorsque la DLE est une DLE pont qui a pu accepter la DLPDU de lancement de transaction pour transmission;

b) autrement, lorsque a) ne s'applique pas, les paramètres SR doivent êtres codés comme décrit en 7.5.

Lorsqu'il est envoyé comme réponse immédiate à une DLPDU TL, les paramètres SR doivent êtres codés comme décrit en 7.5.

6.8.3 Envoi de la DLPDU SR

a) Une DLPDU SR peut être envoyée sur la liaison lorsque la DLE expéditrice est une DLE pont qui vient juste de recevoir un jeton de réponse dans une DLPDU CA, CD ou ED, autorisant cette DLE à émettre un seule DLPDU DT ou SR, et la DLE répond comme spécifié par 6.4.4.3, 6.5.4.3 ou 6.6.4.3.

   C'est une erreur de protocole pour une DLE pont qui transmet la DLPDU CA, CD ou ED que de ne pas envoyer une DLPDU de réponse SR lorsqu'une réponse est exigée.

   NOTE Il convient qu'au maximum une DLE pont sur la liaison locale transmette la DLPDU CA, CD ou ED reçue.

b) Une DLPDU SR peut être envoyée sur la liaison lorsque la DLE expéditrice est une DLE pont ou maître de liaison qui vient juste de recevoir un jeton de réponse dans une DLPDU TL, et la DLE destinataire a besoin de rejeter le transfert du rôle LAS (voir 6.20.4.2).

Le sous-champ "'utilisation finale du jeton" de la DLPDU SR de réponse doit avoir la même valeur que celui dans la demande de DLPDU CA, CD ou ED, ou doit avoir la valeur FINAL pour une réponse à la DLPDU TL demandeuse.

Lorsqu'une réponse immédiate à une DLPDU CA, CD, ED, ou TL est requise, la DLE doit répondre dans les limites d'une période de maximum-response-delay intervalles de temps, à savoir V(MRD) × V(ST) durées d'octet, par rapport à la réception de la DLPDU CA, CD, ED, ou TL demandeuse.

6.8.4 Réception de la DLPDU SR

Une DLPDU SR reçue doit être traitée comme suit par la DLE destinataire.

6.8.4.1 Actions requises de toutes les DLE

Les considérations suivantes s'appliquent:

a) Une DLPDU SR reçue comme réponse à une DLPDU CA, CD ou ED immédiatement antérieure qui a été initiée par la DLE destinataire et qui a été adressée à une adresse de DLSAP, doit amener la DLE destinataire
   1) à considérer que la transmission antérieure était exempte d'erreur;
   2) à transmettre la DLPDU reçue aux fonctions de niveau supérieur de la DLE (voir 8.3 pour un traitement ultérieur.
b) Une DLPDU SR reçue, reçue comme réponse à une DLPDU TL immédiatement antérieure, qui a été initiée par la DLE (LAS) destinataire, doit amener la DLE destinataire
   1) à considérer que la transmission antérieure était exempte d'erreur;
   2) comme spécifié en 6.20.3
      – à réassumer le jeton programmeur;
      – à informer la gestion de DL locale de l'événement;
      – à reprendre le fonctionnement actif comme LAS, et à commencer la transmission sur la liaison.

6.8.4.2 Actions supplémentaires requises d'une DLE de classe Link-Master

Aucune.

6.8.4.3 Actions supplémentaires requises d'une DLE de classe Bridge

Les considérations suivantes s'appliquent:

a) Étant donné que chaque DLE de classe Bridge a une capacité de maître de liaison, toutes les actions spécifiées en 6.8.4.2 s'appliquent également à une DLE de la classe Bridge.

b) Une DLPDU SR reçue

   1) qui rapporte un statut différent de "BOK - bridge OK (pont correct)" (c'est-à-dire qu'il a une valeur différente de E_{16});

   2) qui a été reçue comme réponse à une DLPDU CA, CD ou ED immédiatement antérieure qui a été transmise (et donc pas initiée) par la DLE destinataire et qui a été adressée à une adresse DLSAP

   doit amener la DLE destinataire à former une DLPDU DT

      i) avec le format 2L lorsque la précédente DLPDU CA, CD ou ED était au format 1L, ou le format 2S lorsque la précédente DLPDU CA, CD ou ED était au format 1S;

      ii) avec l'adresse de DL de destination explicite égale à l'adresse de DL source de cette DLPDU CA, CD ou ED immédiatement antérieure;

      ii) avec un champ "paramètres SD" égal au champ "paramètres SR" de la DLPDU SR reçue;

      iv) avec un champ "données utilisateur" vide;

   et doit transmettre la DLPDU DT juste formée comme si elle avait juste été créée par les fonctions de niveau supérieur de la DLE pont.

   NOTE Cela signifie que la DLE pont est à l'origine de la FCS pour la DLPDU DT nouvellement formée comme en 5.2.5.1, plutôt que de simplement modifier une FCS précédemment reçue au cours de la transmission normale comme en 5.2.5.3.

c) Autrement, si b) ne s'applique pas, la DLE ne doit pas transmettre la DLPDU.

6.8.4.4 Actions supplémentaires requises de la DLE LAS courante

Les considérations suivantes s'appliquent:

a) Si le sous-champ "utilisation finale de jeton" de la DLPDU reçue a la valeur FINAL, la DLE LAS doit

   1) supposer que l'utilisation courante du jeton délégué est terminée comme si le jeton avait été retournée par une DLPDU RT (voir 6.17);

   2) supposer que le jeton programmeur est de nouveau dominant sur la liaison locale et reprendre le fonctionnement actif comme LAS.

b) Une DLPDU SR reçue, reçue comme réponse à une DLPDU TL immédiatement antérieure, qui a été initiée par la DLE LAS, doit amener la DLE LAS

   1) à considérer que la transmission antérieure était exempte d'erreur;

   2) à mettre fin au transfert tenté du rôle de LAS;

   3) à informer la gestion de DL de l'événement.

6.9 DLPDU "Compel time" (Forcer le temps (CT))

Une DLPDU COMPEL TIME (CT) (Forcer le temps) est utilisée par le détenteur courant d'un jeton délégué pour demander à la DLE LAS d'émettre comme réponse immédiate une DLPDU TIME DISTRIBUTION (TD), permettant ainsi à toutes les autres DLE sur la liaison locale

– de mettre à jour leur sens du temps de DL;

– de synchroniser les rythmes de progression de leurs sens du temps.

6.9.1 Structure de la DLPDU CT

Tableau 20 – Structure des DLPDU CT

Contrôle de trame
0001 0F00

6.9.1.1 Champ "Frame control" (contrôle de trame)

Le champ "contrôle de trame" doit spécifier

a) la fonction de la DLPDU;
b) la priorité implicite de la DLPDU, qui est URGENT;
c) la longueur, le nombre et le type d'adresses de DLPDU;
d) si, oui ou non, la transmission de la DLPDU courante, et soit sa réponse immédiate escomptée, soit la temporisation de repos de liaison appropriée, mettent fin à l'utilisation d'un jeton délégué.

6.9.1.2 Champ "Address" (adresse)

Le champ "adresse" doit être vide.

6.9.1.3 Champ "Parameters" (paramètres)

Le champ "paramètres" doit être vide.

6.9.1.4 Champ "User data" (données d'utilisateur)

Le champ "données d'utilisateur" doit être vide.

6.9.2 Contenu de la DLPDU CT

Le champ "contrôle de trame" doit être codé comme spécifié dans le Tableau 20.

6.9.3 Envoi de la DLPDU CT

Un besoin d'une DLE d'obtenir le temps de DL actuel, ou de se synchroniser avec la DLE LAS et le rythme de progression du temps de DL actuel de cette DLE, peut conduire à une transmission d'une DLPDU CT. Une DLPDU CT demande que la DLE LAS envoie une DLPDU TD en réponse immédiate.

Une DLPDU CT peut être envoyée sur la liaison lorsque la DLE expéditrice détient un jeton délégué dont la durée allouée restante d'utilisation de jeton, C(RD), permet l'achèvement de la transaction impliquée – en envoyant la DLPDU CT, qui exige une réponse immédiate, et en attendant la DLPDU TD de réponse – avant le retour du jeton.

Si la DLE détient un jeton délégué et aucune utilisation supplémentaire de ce jeton après envoi de cette DLPDU et attente de sa réponse immédiate n'est nécessaire à cet instant-là, la DLE peut mettre à la valeur FINAL le sous-champ "utilisation finale du jeton" de la DLPDU CT; sinon, ce sous-champ doit avoir la valeur NOT-FINAL.

Après l'envoi d'une DLPDU CT, la DLE expéditrice doit surveiller la liaison locale pour détecter une réponse telle que spécifiée en 5.2.8.1. Une DLPDU de réponse admissible est une DLPDU TD.

6.9.4 Réception de la DLPDU CT

Une DLPDU CT reçue doit être traitée comme suit par la DLE destinataire.

6.9.4.1 Actions requises de toutes les DLE

Aucune.

6.9.4.2 Actions supplémentaires requises d'une DLE de classe Link-Master

Aucune.

6.9.4.3 Actions supplémentaires requises d'une DLE de classe Bridge

a) Étant donné que chaque DLE de classe Bridge a une capacité de maître de liaison, toutes les actions spécifiées en 6.9.4.2 s'appliquent également à une DLE de la classe Bridge.

b) La DLE ne doit pas transmettre la DLPDU.

6.9.4.4 Actions supplémentaires requises de la DLE LAS courante

La DLE doit répondre immédiatement par une DLPDU TD.

Si le sous-champ "utilisation finale de jeton" de la DLPDU CT reçue a la valeur FINAL, la DLE LAS doit

a) supposer que l'utilisation courante du jeton délégué a cessé et que le jeton programmeur est de nouveau dominant sur la liaison locale;

b) traiter cette cessation comme si le jeton avait été renvoyé par une DLPDU RT (voir 6.17);

c) reprendre le fonctionnement actif comme le LAS.

6.10 DLPDU Time distribution (Distribution du temps (TD))

Une DLPDU TIME DISTRIBUTION (TD) est émise par la DLE LAS pour permettre aux DLE sur la liaison locale de coordonner et synchroniser les rythmes de progression de leurs sens de temps de DL.

6.10.1 Structure de la DLPDU TD

Tableau 21 – Structure des DLPDU TD

Contrôle de trame	Adresse NODE source	Paramètres
0001 0F01	N	TD-p

6.10.1.1 Champ "Frame control" (contrôle de trame)

Le champ "contrôle de trame" doit spécifier

a) la fonction de la DLPDU;

b) la priorité implicite de la DLPDU, qui est URGENT;

c) la longueur, le nombre et le type d'adresses de DLPDU;

d) si, oui ou non, la transmission de la DLPDU courante met fin à l'utilisation d'un jeton délégué.

6.10.1.2 Champ "Address" (adresse)

Le champ "adresse" doit consister en une adresse de DL NODE source explicite.

6.10.1.3 Champ "Parameters" (paramètres)

Le champ "paramètre" doit être structuré et codé tel que spécifié en 7.6.

6.10.1.4 Champ "User data" (données d'utilisateur)

Le champ "données d'utilisateur" doit être vide.

6.10.2 Contenu de la DLPDU TD

Le champ "contrôle de trame" doit être codé comme spécifié dans le Tableau 21.

Le champ "adresse" doit spécifier l'appellation de nœud de la DLE LM qui co-réside avec la DLE LAS.

Le champ "paramètres TD" doit être codé comme spécifié en 7.6, et doit refléter les valeurs courantes des variables relatives au temps de la DLE et du compteur de temps de nœud au moment de la formation de la DLPDU, qui doivent être inférieures à 1 s avant la transmission de la DLPDU.

Le champ "données d'utilisateur" doit être vide.

6.10.3 Envoi de la DLPDU TD

Une DLPDU TD peut être envoyée sur la liaison lorsque la DLE expéditrice (le LAS)

a) vient juste de recevoir un jeton de réponse dans une DLPDU CT, permettant une seule transmission d'une DLPDU TD; ou
b) détient un jeton programmeur qui est le jeton dominant sur la liaison locale et doit envoyer une DLPDU TD comme spécifié en 8.4.1.6, et la durée allouée restante d'utilisation du jeton avant la prochaine activité programmée permet l'achèvement de la transmission de la DLPDU avant l'expiration du jeton.

Si la DLPDU TD est envoyée comme réponse à une DLPDU CT reçue, le sous-champ "'utilisation finale du jeton" de la DLPDU TD de réponse doit avoir la même valeur que celui dans la DLPDU CT de demande.

Si la classe de synchronisation du temps de la DLE expéditrice n'est pas NONE, alors après avoir émis la DLPDU TD, la DLE LAS doit programmer une autre transmission d'une DLPDU TD telle que spécifiée en 8.4.1.2.

La réception d'une DLPDU CT par la DLE LAS doit amener la DLE LAS à répondre immédiatement par une DLPDU TD, envoyée à toutes les autres DLE sur la liaison locale.

6.10.4 Réception de la DLPDU TD

Une DLPDU TD reçue doit être traitée comme suit par la DLE destinataire.

6.10.4.1 Actions requises de toutes les DLE

Les considérations suivantes s'appliquent:

a) Chaque DLE qui prend en charge les variables, les temporisateurs et les compteurs définis de 4.7.1.19 à 4.7.1.26, autre que la DLE expéditrice, doit abouter à la DLPDU reçue les 24 bits de poids faible du temps de nœud local C(NT), appelé ci-après N_R(NT), auquel la réception de la DLPDU TD s'est achevée (à savoir la réception de la PhIDU END-OF-DATA ou END-OF-DATA-AND-ACTIVITY).
b) Chaque DLE autre que la DLE expéditrice doit traiter la DLPDU reçue comme spécifié en 8.4.1.3.

6.10.4.2 Actions supplémentaires requises d'une DLE de classe Link-Master

Aucune.

6.10.4.3 Actions supplémentaires requises d'une DLE de classe Bridge

Les considérations suivantes s'appliquent:

a) Étant donné que chaque DLE de classe Bridge a une capacité de maître de liaison, toutes les actions spécifiées en 6.10.4.2 s'appliquent également à une DLE de la classe Bridge.
b) La DLE ne doit pas transmettre la DLPDU. Cependant, si la DLPDU TD a été reçue au port racine de la DLE pont (c'est-à-dire le port vers la racine de l'arborescence du pont), la DLE doit transmettre l'événement de réception de la DLPDU au niveau opérationnel du pont (voir 4.1.2).

6.10.4.4 Actions supplémentaires requises de la DLE LAS courante

Si la DLE ne venait juste pas d'envoyer la DLPDU, la DLE doit abandonner le jeton programmeur et désactiver son rôle comme LAS, et doit informer la gestion de DL locale de l'événement.

Si le sous-champ "utilisation finale de jeton" de la DLPDU reçue a la valeur FINAL, la DLE LAS doit

a) supposer que l'utilisation courante du jeton délégué est terminée comme si le jeton avait été retournée par une DLPDU RT (voir 6.17);
b) supposer que le jeton programmeur est de nouveau dominant sur la liaison locale et reprendre le fonctionnement actif comme LAS.

6.11 DLPDU Round-trip-delay query (interrogation de temps de propagation aller-retour (RQ))

Une DLPDU ROUND-TRIP-DELAY QUERY (RQ) (interrogation de temps de propagation aller-retour) est envoyée d'une DLE à une autre sur la liaison locale pour lancer la mesure et le calcul du retard aller-retour intrinsèque à leur intercommunication. Sa réception a pour résultat le retour d'une DLPDU ROUND-TRIP-DELAY REPLY (RR) complémentaire terminant la mesure.

6.11.1 Structure de la DLPDU RQ

Tableau 22 – Structure des DLPDU RQ

Contrôle de trame	Adresse de destination	Adresse source	Paramètres
1100 0F00	N.0	N.0	RQ-p

6.11.1.1 Champ "Frame control" (contrôle de trame)

Le champ "contrôle de trame" doit spécifier

a) la fonction de la DLPDU;
b) la priorité implicite de la DLPDU, qui est NORMAL;
c) la longueur, le nombre et le type d'adresses de DLPDU;
d) si, oui ou non, la transmission de la DLPDU courante, et soit sa DLPDU de réponse immédiate escomptée, soit la temporisation de repos de liaison appropriée, mettent fin à l'utilisation d'un jeton délégué.

6.11.1.2 Champ "Address" (adresse)

Le champ "adresse" doit être constitué de deux adresses de DL SHORT explicites, destination et source, dans cet ordre.

6.11.1.3 Champ "Parameters" (paramètres)

Le champ "paramètres" doit être structuré et codé tel que spécifié en 7.7.

6.11.1.4 Champ "User data" (données d'utilisateur)

Le champ "données d'utilisateur" doit être vide.

6.11.2 Contenu de la DLPDU RQ

Le champ "contrôle de trame" doit être codé comme spécifié dans le Tableau 22.

Le champ "adresse" doit spécifier deux adresses DL NODE exprimées au format d'adresse de DL SHORT où l'adresse de DL de destination est

– l'adresse de DL SHORT plate pour le LAS local, 0400_{16} (voir 4.3.3.2); ou
– a la valeur formée par la concaténation de l'id de nœud du destinataire prévu, V_R(TN), et d'un octet de zéro, spécifiant ainsi les fonctions de prise en charge DL de cette DLE;

et l'adresse de DL source a la valeur formée par la concaténation de l'id de nœud de l'expéditeur, V_S(TN), et d'un octet de zéro, spécifiant ainsi les fonctions de prise en charge de DL de la DLE expéditrice.

Le champ "paramètres RQ" doit être codé tel que spécifié en 7.7 et doit refléter la valeur du compteur de temps de nœud de la DLE, s'il est maintenu par la DLE, à un moment de la formation de la DLPDU qui est un décalage temporel constant par rapport au moment où la PhIDU END-OF-DATA-AND-ACTIVITY sera émise pour la DLPDU. Autrement, la DLE doit utiliser n'importe quelle valeur pour ce champ.

Le champ "données d'utilisateur" doit être vide.

6.11.3 Envoi de la DLPDU RQ

Une DLPDU RQ peut être envoyée sur la liaison lorsque la DLE expéditrice

a) détient un jeton délégué qui est le jeton dominant sur la liaison locale, et lorsque la durée allouée restante d'utilisation du jeton, C(RD), permet l'achèvement de la transaction impliquée – envoi de la DLPDU RQ, qui exige une réponse immédiate, et attente de la DLPDU de réponse RR – avant l'expiration du jeton; ou
b) détient un jeton programmeur qui est le jeton dominant sur la liaison locale, et lorsque à la fois
   – la durée allouée restante d'utilisation du jeton avant la prochaine activité programmée;
   – la durée allouée restante d'utilisation du jeton programmeur pour la maintenance de la liaison, qui est V(LTHT) durées d’octet moins la quantité de capacité de liaison utilisée pour la maintenance de la liaison au cours du cycle actuel de "circulation du jeton",

   permettent l'achèvement de la transaction impliquée – envoi de la DLPDU RQ, qui exige une réponse immédiate, et attente de la DLPDU de réponse RR – avant l'expiration du jeton.

La DLPDU RQ doit être envoyée à la priorité NORMAL.

Si la DLE détient un jeton délégué et aucune utilisation supplémentaire de ce jeton après envoi de cette DLPDU et attente de sa DLPDU RR de réponse immédiate n'est nécessaire à cet instant-là, la DLE peut mettre à la valeur FINAL le sous-champ "utilisation finale du jeton" de la DLPDU RQ; sinon, ce sous-champ doit avoir la valeur NOT-FINAL.

Après l'envoi d'une DLPDU RQ, la DLE expéditrice doit surveiller la liaison locale pour détecter une réponse telle que spécifiée en 5.2.8.1. Une DLPDU de réponse admissible est une DLPDU RR.

6.11.4 Réception de la DLPDU RQ

Une DLPDU RQ reçue doit être traitée comme suit par la DLE destinataire.

6.11.4.1 Actions requises de toutes les DLE

Si l'adresse de DL NODE de destination spécifiée par la DLPDU désigne la DLE destinataire, alors

a) lorsque la classe "temps synchronisation "de la DLE destinataire (voir 10.6.3) est NONE) et la DLE ne maintient même pas de C(NT) estimé, alors la DLE doit abouter à la DLPDU reçu la valeur de 24 bits zéro, ou
b) sinon si a) ne s'applique pas, la DLE doit
   1) abouter à la DLPDU reçue les 24 bits de poids faible du temps de nœud local, C(NT), auquel la réception de la DLPDU RQ s'est achevée (c'est-à-dire, la réception de la PhIDU END-OF-DATA ou END-OF-DATA-AND-ACTIVITY);
   2) réajuster cette valeur de 24 bits pour éliminer toute différence systémique dans le retard de temps interne de la DLE entre les chemins de transmission et de réception de la DLE, provoqué par des considérations de mise en œuvre connues au sein du réel système final de réception.

   NOTE 1 Ce réajustement variera probablement en sens inverse du réel débit instantané de transmission de données de la DLE.

   NOTE 2 L'objectif de ce réajustement est d'assurer que V(MD), tel que calculé en 8.4.1.5, est aussi proche que possible du double de la somme des retards unidirectionnels de transmission, de propagation et de réception encourus dans la transmission d'une DLPDU TD entre le nœud source adressé et la DLE locale.

Dans un cas comme dans l'autre, la DLE destinataire doit initier une réponse dans les limites d'une période de maximum-response-delay intervalles de temps, à savoir V(MRD) × V(ST) durées d'octet, par rapport à la réception de la DLPDU RQ. La DLPDU de réponse doit être une DLPDU RR avec une adresse de DL de destination égale à l'adresse de DL source de la DLPDU RQ reçu, et avec une adresse de DL source spécifiant les fonctions de prise en charge de DL de la DLE de réponse.

6.11.4.2 Actions supplémentaires requises d'une DLE de classe Link-Master

Aucune.

6.11.4.3 Actions supplémentaires requises d'une DLE de classe Bridge

Les considérations suivantes s'appliquent:

a) Étant donné que chaque DLE de classe Bridge a une capacité de maître de liaison, toutes les actions spécifiées en 6.11.4.2 s'appliquent également à une DLE de la classe Bridge.
b) La DLE ne doit pas transmettre la DLPDU.

6.11.4.4 Actions supplémentaires requises de la DLE LAS courante

La DLE LAS doit agir comme spécifié en 5.2.8.2.

6.12 DLPDU Round-trip-delay reply (réponse de temps de propagation aller-retour (RR))

Une DLPDU ROUND-TRIP-DELAY REPLY (RR) (réponse de temps de propagation aller-retour) est envoyée d'une DLE à une autre sur la liaison locale pour permettre l'achèvement de la mesure et du calcul du retard aller-retour intrinsèque à leur intercommunication. Elle est seulement envoyée comme une réponse immédiate à une DLPDU RQ reçue.

6.12.1 Structure de la DLPDU RR

Tableau 23 – Structure des DLPDU RR

Contrôle de trame	Adresse de destination	Adresse source	Paramètres
1101 0F00	N.0	N.0	RR-p

6.12.1.1 Champ "Frame control" (contrôle de trame)

Le champ "contrôle de trame" doit spécifier

a) la fonction de la DLPDU;

b) la priorité implicite de la DLPDU, qui est NORMAL;

c) la longueur, le nombre et le type d'adresses de DLPDU;

d) si, oui ou non, la transmission de la DLPDU courante met fin à l'utilisation d'un jeton délégué.

6.12.1.2 Champ "Address" (adresse)

Le champ "adresse" doit être constitué de deux adresses de DL SHORT explicites, destination et source, dans cet ordre.

6.12.1.3 Champ "Parameters" (paramètres)

Le champ "paramètres" doit être structuré et codé tel que spécifié en 7.8.

6.12.1.4 Champ "User data" (données d'utilisateur)

Le champ "données d'utilisateur" doit être vide.

6.12.2 Contenu de la DLPDU RR

Le champ "contrôle de trame" doit être codé comme spécifié dans le Tableau 23.

Le champ "adresse" doit indiquer deux adresses de DL NODE exprimées au format d'adresse de DL SHORT, où l'adresse de DL de destination est égale à l'adresse de DL source reçue dans la DLPDU RQ immédiatement antérieure, et l'adresse de DL source a une valeur formée par la concaténation de l'id de nœud de l'expéditeur, V_S(TN), et un octet de zéro, spécifiant ainsi les fonctions de prise en charge de DL de la DLE expéditrice.

Le champ "paramètres RR" doit être codé tel que spécifié en 7.8 et doit refléter la valeur du compteur de temps de nœud de la DLE, s'il est maintenu par la DLE, à un moment de la formation de la DLPDU qui est un décalage temporel constant par rapport au moment où la PhIDU END-OF-DATA-AND-ACTIVITY sera émise pour la DLPDU. Autrement, la DLE doit utiliser la valeur zéro pour ce champ.

Le champ "données d'utilisateur" doit être vide.

6.12.3 Envoi de la DLPDU RR

Une DLPDU RR DLPDU peut être envoyée sur la liaison lorsque la DLE expéditrice vient juste de recevoir un jeton de réponse dans une DLPDU RQ, permettant une seule transmission d'une DLPDU RR. Le champ "utilisation finale du jeton" de la DLPDU RR de réponse doit avoir la même valeur que celle qui est contenue dans la DLPDU RQ demandeuse.

6.12.4 Réception de la DLPDU RR

Une DLPDU RR reçue doit être traitée comme suit par la DLE destinataire.

6.12.4.1 Actions requises de toutes les DLE

Si l'adresse de DL NODE de destination spécifiée par la DLPDU désigne la DLE destinataire, et si la DLPDU immédiatement antérieure est une DLPDU RQ envoyée par la DLE destinataire, alors la DLE doit considérer que la transmission antérieure était exempte d'erreur, et si la DLE maintient C(NT), alors la DLE doit

a) abouter à la DLPDU reçue les 24 bits de poids faible du temps de nœud local, C(NT), auquel la réception de la DLPDU RR s'est achevée (c'est-à-dire, la réception de la PhIDU END-OF-DATA ou END-OF-DATA-AND-ACTIVITY);

b) réajuster cette valeur de 24 bits pour éliminer toute différence systémique dans le retard de temps interne de la DLE entre les chemins de transmission et de réception de la DLE, provoqué par des considérations de mise en œuvre connues au sein du réel système final de réception.

   NOTE 1 Ce réajustement variera probablement en sens inverse du réel débit instantané de transmission de données de la DLE.

   NOTE 2 L'objectif de ce réajustement est d'assurer que V(MD), tel que calculé en 8.4.1.5, est aussi proche que possible du double de la somme des retards unidirectionnels de transmission, de propagation et de réception encourus dans la transmission d'une DLPDU TD entre le nœud de destination adressé et la DLE locale.

c) et la DLE doit traiter la DLPDU reçue comme spécifié en 8.4.1.5.

6.12.4.2 Actions supplémentaires requises d'une DLE de classe Link-Master

Aucune.

6.12.4.3 Actions supplémentaires requises d'une DLE de classe Bridge

Les considérations suivantes s'appliquent:

a) Étant donné que chaque DLE de classe Bridge a une capacité de maître de liaison, toutes les actions spécifiées en 6.12.4.2 s'appliquent également à une DLE de la classe Bridge.

b) La DLE ne doit pas transmettre la DLPDU.

6.12.4.4 Actions supplémentaires requises de la DLE LAS courante

Si le sous-champ "utilisation finale de jeton" de la DLPDU reçue a la valeur FINAL, la DLE LAS doit

a) supposer que l'utilisation courante du jeton délégué est terminée comme si le jeton avait été retournée par une DLPDU RT (voir 6.17);

b) supposer que le jeton programmeur est de nouveau dominant sur la liaison locale et reprendre le fonctionnement actif comme LAS.

6.13 DLPDU Probe node DL-address (Sonder l'adresse de DL de nœud (PN))

Une DLPDU PROBE NODE DL-ADDRESS (Sonder l'adresse de DL de nœud (PN)) est utilisée par la DLE LAS pour vérifier l'existence de DLE non reconnues précédemment sur la liaison locale. Une DLPDU PN crée et transmet un jeton de réponse à toute(s) DLE susceptible(s) d'avoir son/leur V(TN) (voir 4.7.1.8) égal à l'adresse de DL NODE spécifiée (configurée, mais inactive). La seule réponse admissible est une DLPDU PR (voir 6.14), qui doit être une réponse immédiate.

6.13.1 Structure de la DLPDU PN

Tableau 24 – Structure des DLPDU PN

Contrôle de trame	Adresse NODE de destination	Paramètres
0010 0110	N	PN-p

6.13.1.1 Champ "Frame control" (contrôle de trame)

Le champ "contrôle de trame" doit spécifier

a) la fonction de la DLPDU;

b) la longueur, le nombre et le type d'adresses de DLPDU.

6.13.1.2 Champ "Address" (adresse)

Le champ "adresse" doit consister en une adresse de DL NODE de destination explicite.

6.13.1.3 Champ "Parameters" (paramètres)

Le champ "paramètres" doit être structuré et codé tel que spécifié en 7.9.

6.13.1.4 Champ "User data" (données d'utilisateur)

Le champ "données d'utilisateur" doit être vide.

6.13.2 Contenu de la DLPDU PN

Le champ "contrôle de trame" doit être codé comme spécifié dans le Tableau 24.

Le champ "adresse" doit spécifier l'adresse de DL NODE en cours de sondage.

Le champ "Probe NODE DL-ADDRESS parameters" (paramètres de sondage d'adresse de DL de nœud) (paramètres PN) doit acheminer, comme spécifié en 7.9, les valeurs alors courantes pour la liaison locale de ces paramètres DLE et PhE nécessaires à la DLE destinataire pour se configurer elle-même et sa PhE associée afin qu'elles puissent répondre à la présente DLPDU PN ou à une ultérieure.

Les champs "données d'utilisateur" doivent être vides.

6.13.3 Envoi de la DLPDU PN

Le LAS de lui-même, au début de chaque cycle de transmissions relatives à la maintenance de la liaison, et éventuellement au cours de ce qui serait autrement des périodes d'inactivité de la capacité de liaison, sonde des adresses DL NODE configurées, mais apparemment inutilisées, en envoyant des DLPDU PN à ces adresses de DL et en attendant une réponse. Si une telle réponse se produit, cela indique qu'une nouvelle DLE a rejoint (ou a rejoint de nouveau) la liaison, et le LAS ajoute alors l'adresse de DL NODE de cette DLE

– à sa liste active, V(LL), pour être interrogée avec des DLPDU PT;

– à sa liste de circulation de jetons, V(TCL), si cela est demandé dans la DLPDU PR répondeuse.

Les DLPDU PN ne sont autrement pas envoyées aux adresses de DL NODE.

La DLE LAS doit envoyer une DLPDU PN lors du sondage d'adresses de DL NODE qui ne sont associées à aucune DLE active connue sur la liaison locale. Une DLPDU PN peut être envoyée sur la liaison par la DLE LAS lorsque

a) la DLE LAS utilise la capacité de la liaison allouée à la maintenance de la liaison telle que spécifiée à l'Article 9; ou

b) le jeton programmeur est le jeton dominant sur la liaison locale; et la durée allouée restante d'utilisation du jeton avant la prochaine activité programmée permet l'achèvement d'une transaction PN - PR telle que spécifiée à l'Article 9).

Avant d'envoyer la DLPDU PN, la DLE LAS doit déterminer la prochaine adresse de DL NODE devant être sondée en choisissant, sur des instances d'envoi de la DLPDU PN, parmi les deux ensembles suivants d'adresses de DL NODE, telles que définies en 4.3.2.2, et de façon cyclique dans l'ordre numérique croissant dans chacun des ensembles:

1) $\{ 10_{16}..F7_{16} \}$ – V(LL) – { V(FUN)..(V(FUN) + V(NUN) – 1) },
qui est l'ensemble d'adresses configurées de classe Bridge, de classe Link-Master et de classe Basic, moins l'ensemble d'adresses de DLE actives et moins l'ensemble d'adresses de DLE inutilisées exclues

2) $\{ F8_{16}..FF_{16} \}$ – V(LL),
qui est l'ensemble des adresses de DLE visiteuses, plus l'ensemble d'adresses de DLE pour des DLE non visiteuses qui ne connaissent pas leurs propres adresses DLE, moins l'ensemble des adresses de DLE actives.

Après avoir envoyé une DLPDU PN, la DLE LAS doit surveiller l'activité de la liaison locale pendant une période de immediate-response-recovery-delay intervalles de temps, à savoir V(IRRD) × V(ST) durées d'octet, en attente de réponse. Après cet intervalle, dès que la liaison est inactive, la DLE LAS doit poursuivre avec ses autres activités. Même lorsqu'une réponse a été reçue, le LAS doit attendre jusqu'à la fin de immediate-response-recovery-delay intervalles de temps, à savoir V(IRRD) × V(ST) durées d'octet, après la transmission de PN, et ensuite attendre que la liaison devienne inactive, avant de poursuivre avec sa transaction suivante, pour permettre une potentielle seconde réponse issue d'une DLE qui a la même adresse de DL NODE.

NOTE Cette exigence impose des considérations de temporisateur qui diffèrent subtilement de toutes les autres utilisations des temporisateurs à base d'intervalles de temps dans le protocole de DL. Pour ce temporisateur, l'action de temporisation ne s'arrête pas lorsqu'il est rapporté ACTIVITY ou DATA sur la liaison, mais se poursuit au contraire, même après ces rapports.

Si le LAS reçoit une DLPDU, qui n'est pas une DLPDU PR, pendant cet intervalle de surveillance, la DLE LAS doit abandonner son jeton programmeur et informer la gestion de DL locale de cet événement.

6.13.3.1 Considérations complémentaires

Du point de vue de la DLE LAS, la capacité de liaison en surdébit maximal requise par toute DLPDU PN et une limite supérieure pour les réponses maximales autorisées peuvent être pré-calculées comme suit

2 × (surdébit de verrouillage de trame)
+ 1 × PN-size
+ 1 × V(IRRD) × V(ST)
+ 1 × (taille maximale de P)
+ 1 × V(MID)

6.13.4 Réception de la DLPDU PN

La réception d'une DLPDU PN par une DLE nouvellement attachée ou nouvellement initialisée permet à la DLE d'apprendre les paramètres de configuration de PhL et de DLL qui sont essentiels à la transmission sur la liaison locale.

La réception d'une DLPDU PN par la DLE adressée permet à cette DLE

– d'indiquer sa présence et son activité désirée sur la liaison locale;

– de commencer le processus de son inclusion dans les autres activités de la liaison.

Une DLPDU PN reçue doit être traitée comme suit par la DLE destinataire.

6.13.4.1 Actions requises de toutes les DLE

La DLE doit utiliser la DLPDU PN reçue dans les procédures d'initialisation de la DLE selon 10.1.2.1 et 10.1.3. Si ces procédures exigent que la DLE reçue réponde à la DLPDU PN reçue, alors la DLE

a) doit assumer un jeton réponse;
b) doit initier une transmission d'une DLPDU PR dans les limites d'une période de maximum-response-delay intervalles de temps, à savoir V(MRD) × V(ST) durées d'octet (telles que mesurées à la DLE de réception de PN);
c) doit inclure une SPDU de sondage de réponse, telle que décrite à l'Article 9, dans le champ "données utilisateur" de cette DLPDU PR, et doit assurer que tous les éventuels champs choisis au hasard de cette SPDU sont choisis dans des distributions approximativement uniformes d'une manière qui est statistiquement indépendante des choix similaires effectués par d'autres DLE.

6.13.4.2 Actions supplémentaires requises d'une DLE de classe Link-Master

Aucune.

6.13.4.3 Actions supplémentaires requises d'une DLE de classe Bridge

Les considérations suivantes s'appliquent:

a) Étant donné que chaque DLE de classe Bridge a une capacité de maître de liaison, toutes les actions spécifiées en 6.13.4.2 s'appliquent également à une DLE de la classe Bridge.
b) La DLE ne doit pas transmettre la DLPDU.

6.13.4.4 Actions supplémentaires requises de la DLE LAS courante

Si la DLE ne venait juste pas d'envoyer la DLPDU, la DLE doit abandonner le jeton programmeur et désactiver son rôle comme LAS; et doit informer la gestion de DL locale de l'événement.

6.14 DLPDU Probe response (Sonder une réponse (PR))

Une DLPDU PROBE RESPONSE (PR) (Sondage de réponse) est envoyée au LAS en réponse à une DLPDU PROBE NODE DL-ADDRESS (PN) immédiatement antérieure pour acheminer vers le LAS des informations relatives au sondage de réponse.

6.14.1 Structure de la DLPDU PR

Tableau 25 – Structure des DLPDU PR

Contrôle de trame	User data (données d'utilisateur)
0010 0111	SPDU

6.14.1.1 Champ "Frame control" (contrôle de trame)

Le champ "contrôle de trame" doit spécifier la fonction de la DLPDU.

6.14.1.2 Champ "Address" (adresse)

Le champ "adresse" doit être vide.

6.14.1.3 Champ "Parameters" (paramètres)

Le champ "paramètres" doit être vide.

6.14.1.4 Champ "User data" (données d'utilisateur)

Le champ "données utilisateur" doit consister en une SPDU de sondage de réponse de niveau supérieur.

NOTE Cette limite basse sur la taille de la SPDU de sondage de réponse accélère le sondage des adresses de DL NODE pendant les intervalles au sein du programme de liaison qui seraient autrement inutilisables.

6.14.2 Contenu de la DLPDU PR

Le champ "contrôle de trame" doit être codé comme spécifié dans le Tableau 25. La valeur implicite du sous-champ "utilisation finale du jeton" doit être FINAL.

Le champ "adresse" doit être vide. L'adresse de DL NODE de LAS générique de 04 est implicitement adressée comme destination de la DLPDU, et l'adresse de destination issue de la DLPDU PN immédiatement antérieure envoyée sur la liaison locale est implicitement l'adresse source de la DLPDU PR.

Le champ "paramètres" doit être vide.

Le champ "données utilisateur" doit acheminer une SPDU de sondage de réponse de niveau supérieur dont la taille maximale est limitée comme cela est spécifié en 6.14.1.4.

6.14.3 Envoi de la DLPDU PR

Une DLPDU PR doit découler de la réception d'une DLPDU PN. La DLPDU PR doit être envoyée comme cela est spécifié à l'Article 9 et en 6.13.4.

6.14.4 Réception de la DLPDU PR

Une DLPDU PR reçue doit être traitée comme suit par la DLE destinataire.

6.14.4.1 Actions requises de toutes les DLE

Aucune.

6.14.4.2 Actions supplémentaires requises d'une DLE de classe Link-Master

Aucune.

6.14.4.3 Actions supplémentaires requises d'une DLE de classe Bridge

Les considérations suivantes s'appliquent:

a) Étant donné que chaque DLE de classe Bridge a une capacité de maître de liaison, toutes les actions spécifiées en 6.14.4.2 s'appliquent également à une DLE de la classe Bridge.

b) La DLE ne doit pas transmettre la DLPDU.

6.14.4.4 Actions supplémentaires requises de la DLE LAS courante

Si la DLE avait envoyé une DLPDU PN comme étant la DLPDU antérieure sur la liaison, alors la DLE

a) doit rapporter toute SPDU de niveau supérieur acheminée par la DLPDU reçue, comme si elle avait été reçue comme des données utilisateur de DLS contenues dans une DLPDU DT adressée à l'adresse de DL NODE du LAS et envoyée à partir de l'adresse de DL NODE qui était contenue dans la dernière DLPDU PN envoyée (voir 6.13.3), et doit traiter la SPDU comme cela est spécifié à l'Article 9;

b) doit supposer que le jeton programmeur est de nouveau dominant sur la liaison locale, reprendre le fonctionnement actif comme LAS et initier une transmission

   1) dans les limites d'une période de maximum-response-delay intervalles de temps, à savoir V(MRD) × V(ST) durées d'octet, par rapport à la fin de la réception de la DLPDU PR, telles que mesurées à la DLE destinataire,

   2) mais pas avant la fin de l'intervalle d'observation suivant la DLPDU PN, comme cela est spécifié en 6.13.3.

Si la DLE avait envoyé une DLPDU PN comme étant une DLPDU antérieure sur la liaison, mais une autre DLPDU PR avait été reçue après l'envoi de la DLPDU PN, et avant la réception de la DLPDU PR courante, alors la DLE doit ignorer la DLPDU PR courante et toute SPDU contenue.

Si la DLE n'avait pas envoyé de DLPDU PN comme étant la DLPDU non PR immédiatement antérieure sur la liaison, alors la DLE doit abandonner le jeton programmeur et désactiver son rôle comme LAS; et doit informer la gestion de DL locale de l'événement.

6.15 DLPDU Pass token (Passer un jeton (PT))

Une DLPDU PASS TOKEN (Passer un jeton (PT)) est utilisée pour passer un jeton délégué de la DLE agissant comme un LAS à une DLE située sur la liaison locale. En faisant cela à plusieurs reprises, la DLE LAS fournit un jeton délégué qui "circule" successivement, généralement dans l'ordre des adresses de DL NODE, vers toutes les DLE actives sur la liaison locale qui sont incluses dans la liste de circulation de jeton de la liaison, V(TCL) (voir 4.7.5.3).

NOTE Cette utilisation prend également en charge la migration à partir de normes nationales déjà existantes.

Cette DLPDU donne à la DLE destinataire le droit de lancer des transactions de DL pendant une durée spécifiée dans la DLPDU donnant délégation.

6.15.1 Structure de la DLPDU PT

Tableau 26 – Structure des DLPDU PT

Contrôle de trame	Adresse NODE de destination	Paramètres
0011 0FPP	N	DD-p

6.15.1.1 Champ "Frame control" (contrôle de trame)

Le champ "contrôle de trame" doit spécifier

a) la fonction de la DLPDU;

b) la priorité minimale requise d'utilisation du jeton;

c) la phase (initiale ou suivante) de la délégation de jeton;

d) la longueur, le nombre et le type d'adresses de DLPDU.

6.15.1.2 Champ "Address" (adresse)

Le champ "adresse" doit consister en une adresse de DL NODE de destination explicite.

6.15.1.3 Champ "Parameters" (paramètres)

Le champ "paramètres" doit être structuré et codé tel que spécifié en 7.10.

6.15.1.4 Champ "User data" (données d'utilisateur)

Le champ "données d'utilisateur" doit être vide.

6.15.2 Contenu de la DLPDU PT

Le champ "contrôle de trame" doit être codé comme spécifié dans le Tableau 26. La valeur du sous-champ "priorité" détermine la priorité autorisée minimale de DLPDU qui peut être émise par la DLE destinataire. La valeur du sous-champ "utilisation finale" doit être interprétée généralement comme cela est spécifié en 5.2.1.2c), et spécifiquement comme étant:

> RESTART – il s'agit de la délégation initiale du jeton dans le cycle courant de "circulation du jeton", et il convient donc de redémarrer toutes les transactions programmées à répétitions;
>
> CONTINUE – il s'agit d'une délégation de jeton ultérieure (c'est-à-dire secondaire) dans le cycle courant de "circulation du jeton" et il convient donc de poursuivre la séquence précédemment lancée des transactions placées en file d'attente.

Le champ "adresse" doit spécifier l'adresse de DL NODE vers laquelle le jeton est délégué.

Le champ "Delegation-Duration-parameter" (paramètre de durée de délégation) (paramètre DD) doit achemine, comme spécifié en 7.10, la durée pendant laquelle le jeton est délégué au service Q(US), mesuré en durées d'octet. Ses valeurs autorisées sont 0 à 65 000, mesurées en unités de durée de transmission d'un octet (c'est-à-dire en durées d'octet).

NOTE La limite inférieure de zéro est requise pour permettre le sondage des adresses de DL de nœud qui sont incluses dans la liste active, V(LL), de la liaison locale, mais pas dans la liste de circulation de jeton, V(TCL), de la liaison locale. La valeur minimale pratique pour l'utilisation de cette durée est d'environ 16.

Une durée de zéro ne doit être utilisée que lorsque l'adresse de DL NODE de la DLE adressée est contenue dans la liste active, V(LL), mais pas dans la liste des DLE, V(TCL), devant être interrogées avec des DLPDU PT. Dans ce cas, la DLE adressée indique sa présence continue en répondant par une DLPDU RT ou RI.

Le champ "données d'utilisateur" doit être vide.

6.15.3 Envoi de la DLPDU PT

La DLE LAS envoie une DLPDU PT pour "faire circuler le jeton», généralement dans l'ordre des adresses DL NODE, parmi toutes les DLE actives sur la liaison locale.

6.15.3.1 Détermination des champs de la PT DLPDU et des paramètres "rotation de jeton" connexes

Les considérations suivantes s'appliquent:

a) Une DLPDU PT peut être envoyée sur la liaison
   - lorsque la DLE expéditrice (le LAS) détient un jeton programmeur qui est le jeton dominant sur la liaison locale;
   - lorsque le nombre de durées d'octet de capacité de liaison restante jusqu'à la prochaine activité programmée permet l'envoi de la DLPDU PT, plus l'utilisation minimale requise par la DLE destinataire, plus le retour du jeton dans une DLPDU RI ou la récupération du jeton, s'il y a lieu;
   - lorsque la DLE expéditrice n'a pas besoin d'envoyer une autre DLPDU.
b) Le surdébit pour déléguer le jeton, qui n'est pas inclus dans la capacité de liaison déléguée spécifiée dans les paramètres DD de la DLPDU PT, mais qui est pris comme facteur dans la décision d'envoyer ou non la DLPDU PT, est calculé comme étant la somme de

1) la capacité de liaison requise pour envoyer la DLPDU PT;
2) V(IRRD) × V(ST) durées d'octet pour l'acceptation du jeton et le début de la première utilisation;
3) la capacité de liaison requise pour que le détenteur du jeton délégué envoie une DLPDU RI mettant fin à l'utilisation du jeton;
4) la plus grande valeur parmi
   i) le retard minimal entre des DLPDU, à savoir V(MID) durées d'octet (voir 4.7.1.12), qui suit la DLPDU RI,
   ii) toute période dépendant de la mise en œuvre, mesurée en unités de capacité de liaison, nécessaire pour récupérer le jeton à l'expiration de la capacité de la liaison effectivement déléguée.

c) Le nombre maximal de durées d'octet de capacité de liaison qui peut être déléguée est calculé comme étant la plus petite valeur parmi
   1) 65 000 durées d'octet,
   2) le nombre de durées d'octet de capacité de liaison restant jusqu'à la prochaine activité programmée moins le surdébit pour déléguer le jeton spécifié en b), avec le résultat réduit afin de rendre compte des possibles différences entre les débits de données des deux DLE (c'est-à-dire, multiplié par le rapport minimal possible entre les débits de données Ph de la DLE LAS et de la DLE destinataire, tels que rapportés en 4.4.1.1).

d) Le nombre minimal de durées d'octet de capacité de liaison qu'exige une DLE doit être calculé comme suit:
   1) zéro, lorsque l'adresse de DL NODE de la DLE n'est pas un membre de la V(TCL) établie, indiquant que la DLE ne requiert pas l'utilisation du jeton;
   2) V(DMDT) (voir 4.7.5.7), lorsque 1) ne s'applique pas, et lorsque
      – le sous-champ "utilisation finale" de la DLPDU PT indique RESTART, ou
      – lorsque le dernier jeton envoyé par une DLPDU PT à la même adresse DL n'a pas été retourné par une DLPDU RI; ou
   3) la valeur issue des paramètres DD de la DLPDU RI qui a retourné le dernier jeton envoyé par une DLPDU PT à la même adresse DL, lorsque 1) et 2) ne s'appliquent pas.

e) Le nombre maximal de durées d'octet de capacité de liaison qui peut être déléguée à DLE doit être calculé comme suit:
   1) zéro, lorsque l'adresse de DL NODE de la DLE n'est pas un élément de la V(TCL) établie, indiquant que la DLE ne requiert pas l'utilisation du jeton;
   2) l'élément de la matrice V(MTHA) correspondant à l'adresse de DL NODE de la DLE, lorsque 1) ne s'applique pas, et lorsque le sous-champ "utilisation finale" de la DLPDU PT indique RESTART;
   3) l'élément de la matrice V(RTHA) correspondant à l'adresse de DL NODE de la DLE, lorsque 1) et 2) ne s'appliquent pas.

   NOTE La capacité de liaison effectivement utilisée par les DLPDU RI et RT de renvoi de jeton n'est pas incluse dans ce calcul, car la capacité de liaison est allouée au surdébit du processus de passage de jeton.

f) Le jeton doit être délégué dans l'ordre croissant des adresses de DL NODE aux adresses de DL NODE de la DLE qui sont représentées dans la V(LL), en commençant par l'adresse de DL NODE ayant le plus petit numéro. Lorsqu'une DLE vient juste d'assumer le jeton programmeur (en envoyant une DLPDU CL ou en recevant une DLPDU TL) et de devenir le LAS, elle doit commencer le passage de jeton avec cette adresse DL NODE ayant le plus petit numéro, et doit redémarrer le passage de jeton avec la DLE ayant le plus petit numéro après que toutes les autres DLE représentées dans la V(LL) établie ont cessé l'utilisation de leur jeton
   – en spécifiant une utilisation finale FINAL,

– en étant incapable d'utiliser le jeton sans dépasser le nombre maximal de durées d'octet de capacité de liaison déterminé en e).

Une DLE peut, mais peut ne pas, être temporairement contournée dans cet ordre croissant, car ses exigences, tel que calculées en d), dépassent la capacité de liaison momentanément disponible pour la délégation du jeton, telle que calculée en c). Toute DLE ainsi contournée doit recevoir en délégation sa capacité de liaison allouée restante avant qu'une DLPDU PT, dont le champ "utilisation finale" spécifie RESTART ne soit de nouveau déléguée à la DLE en question.

NOTE Ce contournement relève d'un choix de mise en œuvre, car son existence améliore simplement l'efficacité momentanée de la liaison et n'affecte pas l'interopérabilité.

C'est une erreur de protocole que de spécifier toute adresse de DL autre qu'une adresse de DL NODE valide ou d'adresser de façon explicite une DLE qui est notoirement inactive sur la liaison locale.

g) La période réelle de délégation de jeton doit être calculée de façon à être la plus petite des valeurs calculées en c) et en e), et la délégation ne doit se produire que si cette période est supérieure ou égale à la valeur calculée en d).

h) La DLPDU PT initiale envoyée à chaque DLE sur la liaison au cours d'un cycle de "circulation du jeton" doit spécifier RESTART dans son champ "utilisation finale"; toutes les DLPDU PT ultérieures envoyées vers la même DLE au cours du même cycle, le cas échéant, doivent spécifier CONTINUE dans ce champ "utilisation finale". Lorsqu'il est spécifié RESTART, la DLE doit aussi initialiser l'élément correspondant de la V(RTHA) à la valeur de l'élément correspondant de la V(MTHA).

j) La durée réelle de rotation du jeton, V(ATRT), doit être mesurée par la DLE LAS comme étant l'intervalle entre des occurrences successives d'envoi par la DLE LAS de la DLPDU PT, avec un sous-champ "utilisation de jeton" spécifiant RESTART (voir 6.15.2), vers l'adresse de DL NODE DL de plus petit numéro représentée dans la V(LL).

k) La priorité de DLL spécifiée dans la DLPDU PT doit être déterminée avant chaque cycle de "circulation du jeton" comme suit:

   1) Après que la DLE LAS a assumé le jeton programmeur (en envoyant une DLPDU CL ou en recevant une DLPDU TL), la priorité doit être NORMAL.

   2) Si 1) ne s'applique pas et le temps V(ATRT) qui vient juste d'être calculé était supérieur à V(TTRT), alors

      – la gestion de DL doit recevoir notification de l'événement;

      – la priorité du jeton doit être montée au niveau supérieur d'urgence, si possible.

   3) Si 1) et 2) ne s'appliquent pas, et donc le V(ATRT) qui vient juste d'être calculé est inférieur ou égal à V(TTRT), alors la priorité du jeton doit être baissée au prochain niveau inférieur d'urgence, si possible.

   NOTE L'inclusion de l'hystérésis ou '"apprentissage" dans ce réajustement de la priorité est un domaine d'étude ultérieure.

   Une fois déterminée, la même priorité de DLL doit être incluse dans toutes les DLPDU PT émises au cours de ce cycle de "circulation du jeton”.

6.15.3.2 Envoi de la PT DLPDU et surveillance de la DLE à laquelle le jeton est délégué

Lors de l'envoi de la DLPDU PT, la DLE LAS doit enregistrer l'adresse de destination issue de la DLPDU comme étant l'adresse de délégation, V(DTA) (voir 4.7.5.1), pour une potentielle utilisation de la gestion de DL au cas où le jeton serait perdu et ne serait pas retourné, et pour toute potentielle association à une DLPDU RI ou RT reçue.

Après avoir envoyé une DLPDU PT, la DLE LAS doit surveiller l'activité de la liaison locale pendant une période de immediate-response-recovery-delay intervalles de temps, à savoir V(IRRD) × V(ST) durées d'octet, en attente de réponse.

a) Si une "indication" de Ph-Data (voir 4.4.4) rapportant DATA est reçue, alors la DLE doit poursuivre comme en c).

b) Si a) ne s'applique pas et la période de surveillance de immediate-response-recovery-delay intervalles de temps, à savoir V(IRRD) × V(ST) durées d'octet, a expiré, alors

   1) si le bus n'est pas actif à cet instant-là (c'est-à-dire que la dernière "indication" de Ph-Data reçue rapportait END-OF-ACTIVITY), la DLE doit poursuivre comme en d).

   2) si 1) ne s'applique pas, ce qui implique que la liaison est encore active à cet instant-là (c'est-à-dire que la dernière "indication" de Ph-Data reçue rapportait START-OF-ACTIVITY), alors

      A) pour les mises en œuvre qui n'étaient pas démontrables le 31 décembre 1995 ou avant, la DLE doit surveiller la liaison locale pendant une période d'un intervalle de temps supplémentaire, à savoir V (ST) durées d'octet, attendant une "indication" de Ph-Data

         i) si une "indication" de Ph-Data rapportant DATA est reçue, alors la DLE doit poursuivre comme en c).

         ii) si une "indication" de Ph-Data rapportant END-OF-ACTIVITY est reçue, et i) ne s'applique pas, alors la DLE doit arrêter toute surveillance et poursuivre comme en d).

         NOTE Sans la disposition ii), un bruit intempestif de faible niveau peut amener la DLE LAS à ne pas répéter la tentative de passage de jeton, avec la conséquence d'omettre la DLE du cycle courant de "circulation du jeton". La disposition ii) permet à la DLE LAS de distinguer les probables événements sonores des transmissions réelles.

         iii) si ni i) ni ii) ne s'appliquent et la période de surveillance expire avant qu'une "indication" de Ph Data ne soit reçue, la DLE doit poursuivre comme en c).

      B) Les mises en œuvre qui étaient démontrables le 31 décembre 1995, ou avant, peuvent en variante juste poursuivre comme en c).

c) Jusqu'à ce que le jeton délégué soit renvoyé, la DLE LAS doit continuellement surveiller l'activité de liaison locale, et si elle constate

   1) une période de token-recovery-delay intervalles de temps, à savoir P(TRD) × V(ST) durées d'octet, d'inactivité ininterrompue; ou

   2) pour les mises en œuvre qui n'étaient pas démontrables le 31 décembre 1995 ou avant, une période de token-recovery-delay intervalles de temps, à savoir P(TRD) × V(ST), pendant laquelle toute période continue d'activité (c'est-à-dire entre une "indication" de Ph-Data rapportant START-OF-ACTIVITY et la prochaine "indication" de Ph-Data suivante rapportant END-OF-ACTIVITY ou END-OF-DATA-AND-ACTIVITY) à la fois

      i) n'a pas donné une "indication" de Ph-Data (voir 4.4.4) rapportant DATA, et

      ii) a une durée plus courte qu'une à deux V(ST) durées d'octet

      NOTE L'imprécision concernant l'ampleur de la durée d'activité admissible autorise une certaine économie de mise en œuvre de la mesure.

   elle doit alors

   – supposer que le jeton programmeur est dominant sur la liaison locale;
   – informer la gestion de DL locale de l'événement;
   – commencer la transmission sur la liaison.

d) Si la période de surveillance de immediate-response-recovery-delay intervalles de temps, à savoir V(IRRD) × V(ST) durées d'octet, a expiré et a) ne s'applique pas; ou, pour les mises en œuvre qui n'étaient pas démontrables le 31 décembre 1995, la période de surveillance supplémentaire d'un intervalle de temps, à savoir V (ST) durées d'octet, expire et c) ne s'applique pas; alors la DLE LAS doit retenter d'envoyer DLPDU PT et de surveiller la liaison locale, jusqu'au nombre de fois spécifié comme étant le nombre maximal de répétitions de tentative, V(MRC) (voir 4.7.1.5), de la liaison locale, en utilisant à chaque fois comme valeur de son paramètre de durée la plus petite valeur parmi

   1) la valeur envoyée à l'origine; ou

2) le nombre de durées d’octet de capacité de liaison restant jusqu'à la prochaine activité programmée moins le surdébit pour déléguer le jeton spécifié en 6.15.3.1b), avec le résultat réduit afin de rendre compte des possibles différences de débits de données des deux DLE, tels que spécifiés en 6.15.3.1c)2),

à condition que la répétition de tentative et le nouvel essai de délégation ne doivent se produire que si ladite période est au moins aussi grande que la valeur calculée en 6.15.3.1d).

Cette répétition de tentative doit commencer dans la limite de token-recovery-delay intervalles de temps, à savoir P(TRD) × V(ST) durées d'octet, par rapport au début de la période courante d'inactivité de la liaison.

Si toutes les répétitions de tentative sont infructueuses, la DLE LAS doit

– informer la gestion de DL locale de l'événement;

NOTE Les DLE avec des adresses DL de nœud dans l'ensemble {F816..FF$_{16}$} sont censées mettre fin à l'opération en abandonnant le processus de circulation de jeton. Ainsi, il convient que la gestion de DL ne traite pas ces occurrences comme étant des preuves de dysfonctionnement de la DLE ou de la liaison locale.

– démarrer la prochaine transmission dans la limite de token-recovery-delay intervalles de temps, à savoir P(TRD) × V(ST) durées d'octet, par rapport au début de la période courante d'inactivité de la liaison.

Après trois rapports d'événement de ce type concernant une adresse de DL NODE spécifique quelconque au cours de trois cycles de "circulation de jeton", la gestion de DL doit supprimer l'adresse de DL NODE de la liste de circulation de jeton, V(TCL), sur la liaison locale du LAS et de la liste active, V(LL), de la liaison locale.

6.15.4 Réception de la DLPDU PT

La réception d'une DLPDU PT par la DLE adressée permet à la DLE d'émuler un "jeton circulé" tel que rencontré dans des normes antérieures de bus de passage de jeton, telles que l'ISO/CEI 8802-4, en permettant la transmission des DLPDU à la priorité spécifiée et supérieure, soit

– jusqu'à ce que toutes les transmissions en souffrance aient eu lieu, soit

– jusqu'à ce que la partie de la capacité de transmission de la liaison locale, qui était allouée par les paramètres DD de la DLPDU PT reçue exige le retour du jeton délégué.

Une DLPDU PT reçue doit être traitée par la DLE destinataire comme suit:

6.15.4.1 Actions requises de toutes les DLE

Si l'adresse de DL de destination spécifiée par la DLPDU désigne l'adresse de DL NODE de la DLE destinataire, alors si la DLE destinataire, en raison de sa construction, n'utilise pas le jeton délégué, alors la DLE destinataire doit lancer la transmission d'une DLPDU RT dans les limites d'une période de maximum-response-delay intervalles de temps, à savoir V(MRD) × V(ST) durées d'octet, par rapport à la fin de la réception de la DLPDU PT, telles que mesurées à la DLE destinataire; sinon, la DLE destinataire doit

a) copier la valeur du paramètre de durée de la DLPDU vers compteur dégressif local de durée restante, C(RD);

b) assumer le jeton délégué;

c) lancer la transmission dans les limites d'une période de maximum-response-delay intervalles de temps,
à savoir V(MRD) × V(ST) durées d'octet, par rapport à la fin de la réception de la DLPDU PT, telles que mesurées à la DLE destinataire;

d) employer de façon répétée les critères de sélection suivants jusqu'à ce que le jeton délégué soit retourné au LAS, en lançant à chaque fois la transmission dans les limites d'une période de maximum-response-delay intervalles de temps, à savoir V(MRD) × V(ST) durées d'octet, par rapport à la fin de la transmission antérieure (sauf s'il s'agissait d'une transmission de CA, CD, CT, ED ou RQ), telles que mesurées à la DLE de réception de PT.

Le compteur C(RD) doit décrémenter et être géré comme spécifié en 4.7.1.11.

Les transactions de la priorité spécifiée dans la DLPDU PT de délégation ou de priorité supérieure, doivent être lancées successivement, en commençant comme suit:

– démarrer à partir de la première transaction et inclure toutes les séquences REPETITIVES si le sous-champ "utilisation de jeton" de la DLPDU PT spécifiait RESTART, ou
– continuer à partir de la transaction lancée précédemment et en excluant ainsi toutes les séquences REPETITIVES qui ont été exécutées depuis que la dernière DLPDU PT indiquant RESTART a été reçu si le sous-champ "utilisation de jeton" de la DLPDU PT reçue spécifiait CONTINUE;

sous réserve de la contrainte que la DLE doit limiter son utilisation de jeton à une séquence de transactions qui est garantie s'achever dans la limite de la durée spécifiée dans le paramètre DD de la DLPDU PT reçue.

6.15.4.1.1 Sélection de la prochaine transaction à exécuter

Si la DLE a besoin d'envoyer une DLPDU CT, la DLE doit émettre une DLPDU CT et attendre sa réponse immédiate. Autrement, la DLE destinataire doit à répétitions choisir le prochain élément, le cas échéant, de la file d'attente de services non programmés, Q(US), de la DLE dont la priorité n'est pas inférieure à la priorité de la PT reçue, et doit appliquer tout le reste du présent paragraphe, selon le cas:

a) Si ce prochain élément est une référence à une file d'attente de demandes utilisateur d'adresses de DL, Q_A(UR), alors
   1) si l'élément est une référence qui signale la nécessité de forcer une transmission de DLSDU depuis le correspondant distant d'un DLCEP homologue ou abonné local (voir 4.7.1.17a)1)), alors la DLE doit
      i) émettre une DLPDU CD vers un point d'extrémité de DLC distant, avec des paramètres SD acheminant l'état du DLCEP local au DLCEP distant, comme spécifié en 8.2.2.1 et 8.2.2.4;
      ii) attendre une réponse immédiate;

      NOTE La taille maximale de DLPDU de cette réponse immédiate peut être déterminée uniquement en fonction des paramètres de DLCEP négociés.

      iii) supprimer l'élément de référence de la file d'attente de services non programmés, Q(US), de la DLE.
   2) si l'élément est une référence qui signale la nécessité de forcer une instance d'échange d'unitdata avec une adresse de DLSAP spécifiée à la de priorité de service spécifiée (voir 4.7.1.17a)2)), alors la DLE doit
      i) émettre une DLPDU ED ou CD vers cette adresse de DLSAP distante, à la priorité de service spécifiée, le cas échéant, comme spécifié en 8.2.2.1 et 8.2.2.4;
      ii) attendre une réponse immédiate;

      NOTE La taille maximale de DLPDU de cette réponse immédiate peut être déterminée uniquement en fonction des paramètres de DLCEP négociés.

      iii) supprimer l'élément de référence de la file d'attente de services non programmés, Q(US), de la DLE.
   3) si l'élément est une référence qui signale la nécessité d'émettre le contenu d'un tampon expéditeur qui est lié à un DLCEP local (voir 4.7.1.17a)3)), alors la DLE doit
      i) émettre la DLPDU appropriée à l'état du DLCEP;

ii) si la DLPDU émise qui en résulte achève la transmission d'une DLSDU (de telle sorte qu'aucun segment de cette DLSDU ne reste à émettre), alors la DLE doit

– notifier à l'utilisateur de DLS l'achèvement de la demande, si cela est approprié ;

– supprimer l'élément de référence de la file d'attente de services non programmés, Q(US), de la DLE.

4) si l'élément est une référence qui signale la nécessité d'envoyer une DLSDU à partir de la file d'attente d'envoi liée à un DLCEP local ou une "adresse DLSAP" (voir 4.7.1.17a)4)), ou d'envoyer une DLPDU à partir de l'adresse de DL NODE de la DLE, alors la DLE doit sélectionner un élément de la Q_A(UR), référencée, ayant la priorité spécifiée ou supérieure, qui a été libéré pour transmission et qui requiert une transmission ou retransmission complète ou partielle, comme suit:

– soit le premier élément de la seconde partition de Q_A(UR), soit

– le premier élément de la première partie de Q_A(UR) qui a un ou plusieurs segments marqués pour la retransmission.

NOTE En raison de la simultanéité potentielle des mises en œuvre réelles, l'ensemble des segments ainsi marqués peut ne pas refléter les informations de statut issues de DLPDU récemment reçues au DLCEP.

S'il n'existe pas un tel élément, alors

i) la DLE doit émettre la DLPDU avec un segment de la DLSDU appropriée au statut de segment de transmission, $V_{C,K}$(SS) (voir 4.7.4.8), si applicable, de cet élément de cette Q_A(UR);

ii) si la DLPDU émise qui en résulte achève la transmission d'une DLSDU (de telle sorte qu'aucun segment de cette DLSDU ne reste à émettre), alors la DLE doit

– faire progresser cet élément de Q_A(UR) vers la partition déjà envoyée, ou le supprimer, selon ce qui est approprié au type d'élément;

– notifier à l'utilisateur de DLS l'achèvement de la demande, si cela est approprié ;

– supprimer l'élément de référence de la file d'attente de services non programmés, Q(US), de la DLE.

Si aucun élément de ce type de la file d'attente de demandes d'adresses de DL, Q_A(UR), référencée n'existe, alors

iii) si l'adresse de DL est une adresse DLCEP d'un DLCEP homologue ou éditeur local, alors la DLE doit

A) émettre une DLPDU DT ou RC, selon ce qui est approprié à l'état du DLCEP, avec un champ "données utilisateur" vide, pour acheminer l'état du DLCEP au(x) DLCEP homologue(s) ou abonné(s) correspondant(s);

B) supprimer l'élément de référence de la file d'attente de services non programmés, Q(US), de la DLE.

iv) autrement, lorsque iii) ne s'applique pas, alors l'adresse DL n'est pas une adresse DLCEP et la DLE doit répéter le processus de sélection.

b) Si ce prochain élément est une référence à une séquence REPETITIVE (répétitive) (voir 8.4.3.1d)), alors

1) s'il y a un ou plusieurs éléments actifs de la séquence (c'est-à-dire, pas exclu(s) par la plus récente "request" de DL-SUBSET-SEQUENCE, le cas échéant, spécifiant la séquence);

2) si un ou plusieurs de ces éléments actifs n'ont pas été exécutés depuis que la dernière fois que la DLE a reçu une DLPDU PT indiquant RESTART,

alors, la DLE doit

i) insérer une référence au prochain élément actif de cette séquence sur la Q(US) immédiatement avant la référence à la séquence (avec comme résultat que la référence nouvellement insérée sera supprimée à l'achèvement de l'action qui vient d'être programmée);

ii) continuer comme dans le cas précédent a), tel que déterminé par le type de l'élément de la séquence qui vient juste d'être inséré.

NOTE À la fin de l'action qui vient juste d'être programmée, le prochain élément de Q(US) sera la référence à la séquence REPETITIVE, et la procédure ci-dessus se poursuivra jusqu'à ce que tous les éléments actifs de la séquence REPETITIVE aient été traités.

3) autrement, lorsque ni 1) ni 2) ne s'applique, la DLE doit répéter la séquence de sélection.

c) Si ce prochain élément est une référence à une séquence (voir 8.4.3.1a)) qui n'est pas une séquence REPETITIVE, et si la séquence consiste en un seul élément, alors la DLE doit

1) remplacer la référence à la séquence sur Q(US) par la référence à l'élément unique, si bien que le même élément de Q(US) est désormais l'élément de référence;

2) émettre une primitive "confirm" de service DL-SCHEDULE-SEQUENCE s'il n'a pas déjà été confirmé

3) supprimer la séquence;

4) continuer comme dans le cas précédent a), tel que déterminé par le type de l'élément unique de la séquence sélectionnée et juste supprimée.

Tous les autres cas sont erronés et sont interdits.

6.15.4.1.2 Considérations complémentaires

Les considérations suivantes s'appliquent également:

a) S'il n'y a pas éléments supplémentaires, ayant la priorité spécifiée dans la DLPDU PT reçue, ou toute priorité supérieure, dans la file d'attente de services non programmés, Q(US), de la DLE et si une répétition immédiate de tentative de la transaction actuellement sélectionnée n'est pas possible, alors le sous-champ "utilisation finale de jeton" dans cette DLPDU autre que RT peut être mis à la valeur FINAL avant la transmission, ce qui renvoie le jeton délégué à la DLE LAS à la fin de la transaction courante.

b) S'il n'y a aucun élément, ayant la priorité spécifiée dans la DLPDU PT reçue, ou toute priorité supérieure, dans la file d'attente de services non programmés, Q(US), de la DLE et le jeton délégué n'a pas été retourné par le sous-champ "utilisation finale de jeton" à la valeur FINAL dans la dernière DLPDU émise par la DLE, alors la DLE doit retourner le jeton délégué en émettant une DLPDU RT.

c) Si à un instant quelconque

1) la durée restante d'utilisation de jeton – telle qu'indiquée par le compteur dégressif de durée restante, C(RD) – est inadéquate pour permettre toute utilisation ultérieure par la DLE destinataire;

2) la file d'attente de services non programmés, Q(US), de la DLE contient des éléments supplémentaires dont la priorité est égale ou supérieure à la priorité spécifiée dans la DLPDU PT reçue en dernier;

alors la DLE destinataire doit retourner le jeton délégué au LAS en envoyant une DLPDU RI spécifiant l'intervalle minimal de délégation pour l'utilisation de jeton requis lorsque le jeton est ensuite retourné.

NOTE Une DLPDU RI, qui est envoyée pour mettre fin à l'utilisation d'un jeton qui a été délégué par une DLPDU PT, indique au LAS destinataire qu'un service supplémentaire est nécessaire et qu'il convient d'envoyer une autre DLPDU PT, avec un sous-champ "utilisation finale de jeton" indiquant CONTINUE, vers la DLE au cours du cycle courant de "circulation de jeton" si la capacité totale de liaison allouée à la DLE par l'élément de rang DLE de la matrice V(MTHA) (voir 4.7.5.10) le permet.

d) Si aucun des points a) à c) ne s'applique, alors la DLE doit émettre la DLPDU non RT et non RI avec une valeur NOT-FINAL pour le sous-champ "utilisation finale de jeton".

6.15.4.2 Actions supplémentaires requises d'une DLE de classe Link-Master

La DLE peut, mais n'est pas tenue de le faire, enregistrer l'adresse de destination issue de la DLPDU comme étant l'adresse de délégation, V(DTA) (voir 4.7.5.1), en vue d'une utilisation potentielle pour redémarrer le jeton mis en circulation si le LAS courant venait à défaillir et cette DLE LM devenait le nouveau LAS. La DLE peut également utiliser cette V(DTA) enregistrée pour détecter les ajouts, mais pas les suppressions, apportées à l'ensemble des DLE recevant le jeton mis en circulation, et mettre à jour sa copie de la liste de circulation de jeton, V (TCL) (voir 4.7.5.3) en conséquence.

NOTE Ces actions optimisent la reprise du processus de circulation de jeton en cas de défaillance de la DLE fonctionnant actuellement comme LAS.

6.15.4.3 Actions supplémentaires requises d'une DLE de classe Bridge

Les considérations suivantes s'appliquent:

a) Étant donné que chaque DLE de classe Bridge a une capacité de maître de liaison, toutes les actions spécifiées en 6.15.4.2 s'appliquent également à une DLE de la classe Bridge.
b) La DLE ne doit pas transmettre la DLPDU.

6.15.4.4 Actions supplémentaires requises de la DLE LAS courante

Si la DLE ne venait juste pas d'envoyer la DLPDU, la DLE doit abandonner le jeton programmeur et désactiver son rôle comme LAS, et doit informer la gestion de DL locale de l'événement.

6.16 DLPDU Execute sequence (Exécuter une séquence (ES))

Une DLPDU EXECUTE SEQUENCE (Exécution de séquence (ES)) est utilisée pour passer un jeton délégué de la DLE fonctionnant comme un LAS vers une DLE placée sur la liaison locale qui a précédemment demandé ou programmé de la capacité de liaison locale, ou pour laquelle un tel programme a été configuré par la gestion de DL.

NOTE Cette utilisation prend en charge la migration à partir de normes nationales déjà existantes.

Cette DLPDU donne à la DLE destinataire le droit de lancer des transactions de DL pendant une durée, spécifiée dans la DLPDU donnant délégation. L'adresse de DL de destination de cette dernière DLPDU est utilisée pour déterminer l'ensemble des transactions à lancer.

6.16.1 Structure de la DLPDU ES

Tableau 27 – Structure des DLPDU ES

Format	Contrôle de trame	Adresse de DLSEP de destination	Paramètres
1L	1000 1F00	HL.N.S	DD-p
1s	1000 0F00	N.S	DD-p

6.16.1.1 Champ "Frame control" (contrôle de trame)

Le champ "contrôle de trame" doit spécifier

a) la fonction de la DLPDU;
b) la phase (initiale ou suivante) de l'exécution de séquence;
c) la longueur, le nombre et le type d'adresses de DLPDU.

6.16.1.2 Champ "Address" (adresse)

Le champ "adresse" doit consister en une adresse de DL de destination explicite.

6.16.1.3 Champ "Parameters" (paramètres)

Le champ "paramètres" doit être structuré et codé tel que spécifié en 7.10.

6.16.1.4 Champ "User data" (données d'utilisateur)

Le champ "données d'utilisateur" doit être vide.

6.16.2 Contenu de la DLPDU ES

Le champ "contrôle de trame" doit être codé comme spécifié dans le Tableau 27. La valeur du sous-champ "utilisation finale" doit être interprétée généralement comme cela est spécifié en 5.2.1.2c), et spécifiquement comme étant:

RESTART – il s'agit de la délégation de jeton initiale dans le cycle courant de "circulation du jeton", ou d'exécution de séquence programmée, et donc il convient de redémarrer la séquence indiquée et toutes les éventuelles transactions programmées à répétitions;

CONTINUE – il s'agit d'une délégation de jeton ultérieure (c'est-à-dire secondaire) dans le cycle courant d'exécution de séquence programmée et, donc, il convient que la séquence lancée précédemment de transactions programmées soit poursuivie au point où l'exécution avait été suspendue l dernière fois et la DLPDU ES avait été redemandée (voir 6.18.3).

Le champ "adresse" doit spécifier l'adresse de DLSEP vers laquelle le jeton est délégué. Pour le format 1L, l'adresse doit être LONG; pour le format 1S, elle doit être SHORT.

Le champ "Delegation-Duration-parameter" (paramètre de durée de délégation) (paramètre DD) doit acheminer, comme spécifié en 7.10, la durée pendant laquelle le jeton est délégué pour exécuter la séquence programmée adressée, mesurée en unités de période de transmission d'un octet sur la liaison locale (c'est-à-dire huit fois la période de bit nominale de la liaison locale).Sa plage s'étend de 0 à 65 000 durées d'octet.

Le champ "données d'utilisateur" doit être vide.

6.16.3 Envoi de la DLPDU ES

La DLE LAS envoie une DLPDU ES lorsqu'il y est invité par le courant programme de la liaison locale, ou en réponse à des demandes de prolongation du temps alloué à une délégation de jeton antérieure programmée (et exécutée).

6.16.3.1 Envoi de la DLPDU ES comme instruit par le courant programme de la liaison locale

Une DLPDU ES peut être envoyée sur la liaison lorsque la DLE expéditrice (le LAS) détient un jeton programmeur qui est le jeton dominant sur la liaison locale, et le programme de la liaison requiert la transmission de la DLPDU ES. Le sous-champ "utilisation finale" d'un telle DLPDU ES doit être dicté par le programme.

6.16.3.2 Envoi de la DLPDU ES en réponse à des demandes antérieures d'extensions de la délégation de jeton programmée

Une DLPDU ES ne doit être envoyée que comme réponse à une DLPDU RI telle que spécifiée à l'Article 9. Si la DLE LAS envoie une DLPDU ES qui n'est pas dictée par le courant programme de la liaison, alors les procédures suivantes doivent s'appliquer:

a) Une DLPDU ES peut être envoyée sur la liaison lorsque la DLE expéditrice (le LAS) détient un jeton programmeur qui est le jeton dominant sur la liaison locale, et lorsque le nombre de durées d'octet de capacité de liaison restant jusqu'à la prochaine activité programmée permet l'envoi de la DLPDU ES, plus l'utilisation minimale requise par la DLE destinataire, plus le retour du jeton dans une DLPDU RI ou la récupération du jeton si nécessaire. Le sous-champ "utilisation finale" d'une telle DLPDU ES doit spécifier CONTINUE.

b) Le surdébit pour déléguer le jeton, qui n'est pas inclus dans la capacité de liaison déléguée spécifiée dans les paramètres DD de la DLPDU ES, mais qui est pris comme facteur dans la décision d'envoyer ou non la DLPDU ES, est calculé comme étant la somme de

   1) la capacité de liaison requise pour envoyer la DLPDU ES;

   2) V(IRRD) × V(ST) durées d'octet pour l'acceptation du jeton et le début de la première utilisation;

   3) la capacité de liaison requise pour que le détenteur du jeton délégué envoie une DLPDU RI mettant fin à l'utilisation du jeton;

   4) la plus grande valeur parmi

      i) le retard minimal entre des DLPDU, à savoir V(MID) durées d'octet (voir 4.7.1.12), qui suit la DLPDU RI; ou

      ii) toute période dépendant de la mise en œuvre, mesurée en unités de capacité de liaison, nécessaire pour récupérer le jeton à l'expiration de la capacité de la liaison effectivement déléguée.

c) Le nombre maximal de durées d'octet de capacité de liaison qui peut être déléguée est calculé comme étant la plus petite valeur parmi

   1) 65 000 durées d'octet; ou

   2) le nombre de durées d'octet de capacité de liaison restant jusqu'à la prochaine activité programmée moins le surdébit pour déléguer le jeton spécifié en b), avec le résultat réduit afin de rendre compte des possibles différences entre les débits de données des deux DLE (c'est-à-dire, multiplié par le rapport minimal possible entre les débits de données Ph de la DLE LAS et de la DLE destinataire, tels que rapportés en 4.4.1.1).

d) La période réelle de la délégation de jeton doit être la valeur retournée dans les paramètres DD de la DLPDU RI qui a mis fin à l'intervalle antérieur d'utilisation du jeton délégué à la même adresse DL.

6.16.3.3 Surveillance de la DLE vers laquelle le jeton est délégué et considérations supplémentaires

C'est une erreur de protocole que de spécifier toute adresse de DL autre qu'une adresse de DLSEP valide, ou d'adresser de façon explicite ou implicite une DLE inactive sur la liaison locale. Toutefois, un LAS qui reçoit une adresse de DLSEP d'une autre DLE dans une SPDU relative à la programmation peut présupposer qu'une telle adresse est valide.

Lors de l'envoi de la DLPDU ES, la DLE LAS doit enregistrer l'adresse de destination issue de la DLPDU comme étant l'adresse de délégation, V(DTA) (voir 4.7.5.1), pour une potentielle utilisation de la gestion de DL au cas où le jeton serait perdu et ne serait pas retourné, et pour toute potentielle association à une DLPDU RI ou RT reçue.

Après avoir envoyé une DLPDU ES, la DLE LAS doit surveiller l'activité de la liaison locale pendant une période de immediate-response-recovery-delay intervalles de temps, à savoir V(IRRD) × V(ST) durées d'octet, en attente de réponse.

a) Si une "indication" de Ph-Data (voir 4.4.4) rapportant DATA est reçue, alors la DLE doit poursuivre comme en c).

b) Si a) ne s'applique pas et la période de surveillance de immediate-response-recovery-delay intervalles de temps, à savoir V(IRRD) × V(ST) durées d'octet, a expiré, alors

1) si le bus n'est pas actif à cet instant-là (c'est-à-dire que la dernière "indication" de Ph-Data reçue rapportait END-OF-ACTIVITY), la DLE doit poursuivre comme en d).

2) si 1) ne s'applique pas, ce qui implique que la liaison est encore active à cet instant-là (c'est-à-dire que la dernière "indication" de Ph-Data reçue rapportait START-OF-ACTIVITY), alors

   i) Pour les mises en œuvre qui n'étaient pas démontrables le 31 décembre 1995 ou avant, la DLE doit surveiller la liaison locale pendant une période d'un intervalle de temps supplémentaire, à savoir V (ST) durées d'octet, attendant une "indication" de Ph-Data

      A) si une "indication" de Ph-Data rapportant DATA est reçue, alors la DLE doit poursuivre comme en c).

      B) si une "indication" de Ph-Data rapportant END-OF-ACTIVITY est reçue, et i) ne s'applique pas, alors la DLE doit arrêter toute surveillance et poursuivre comme en d);

      NOTE Sans la disposition ii), un bruit intempestif de faible niveau peut amener la DLE LAS à ne pas répéter la tentative de passage de jeton, avec la conséquence d'omettre la DLE du cycle courant d'exécution de programme. La disposition ii) permet à la DLE LAS de distinguer les probables événements sonores des transmissions réelles.

      C) si ni A) ni B) ne s'appliquent et la période de surveillance expire avant qu'une "indication" de Ph-Data ne soit reçue, la DLE doit poursuivre comme en c).

   ii) Les mises en œuvre qui étaient démontrables le 31 décembre 1995, ou avant, peuvent en variante juste poursuivre comme en c).

c) Jusqu'à ce que le jeton délégué soit renvoyé, la DLE LAS doit continuellement surveiller l'activité de liaison locale, et si elle constate

1) une période de "retard de récupération de jeton" intervalles de temps, à savoir P(TRD) × V(ST) durées d'octet, d'inactivité ininterrompue; ou

2) pour les mises en œuvre qui n'étaient pas démontrables le 31 décembre 1995 ou avant, une période de token-recovery-delay intervalles de temps, à savoir P(TRD) × V(ST), pendant laquelle toute période continue d'activité (c'est-à-dire entre une "indication" de Ph-Data rapportant START-OF-ACTIVITY et la prochaine "indication" de Ph-Data suivante rapportant END-OF-ACTIVITY ou END-OF-DATA-AND-ACTIVITY) à la fois

   i) n'a pas donné une "indication" de Ph-Data (voir 4.4.4) rapportant "data", et

   ii) a une durée plus courte qu'une à deux V(ST) durées d'octet,

   NOTE L'imprécision concernant l'ampleur de la durée d'activité admissible autorise une certaine économie de mise en œuvre de la mesure.

elle doit alors

– supposer que le jeton programmeur est dominant sur la liaison locale;
– informer la gestion de DL locale de l'événement;
– commencer la transmission sur la liaison.

d) Si la période de surveillance de immediate-response-recovery-delay intervalles de temps, à savoir V(IRRD) × V(ST) durées d'octet, a expiré et a) ne s'applique pas; ou, pour les mises en œuvre qui n'étaient pas démontrables le 31 décembre 1995 ou avant, la période de surveillance supplémentaire d'un intervalle de temps, à savoir V (ST) durées d'octet, expire et c) ne s'applique pas; alors la DLE LAS doit retenter d'envoyer DLPDU ES et de surveiller la liaison locale, jusqu'au nombre de fois spécifié comme étant le nombre maximal de répétitions de tentative, V(MRC) (voir 4.7.1.5), de la liaison locale, en utilisant à chaque fois comme valeur de son paramètre de durée la valeur initialement envoyée et la redélégation ne doit être effectuée que si cette période est au moins aussi grande que la valeur spécifiée en (6.16.3.1 ou 6.16.3.2d).

Cette répétition de tentative doit commencer dans la limite token-recovery-delay de jeton" intervalles de temps, à savoir P(TRD) × V(ST) durées d'octet, par rapport au début de la période courante d'inactivité de la liaison.

Si toutes les répétitions de tentative sont infructueuses, la DLE LAS doit

– informer la gestion de DL locale de l'événement;
– démarrer la prochaine transmission dans la limite de token-recovery-delay intervalles de temps, à savoir P(TRD) × V(ST) durées d'octet, par rapport au début de la période courante d'inactivité de la liaison.

6.16.4 Réception de la DLPDU ES

La réception d'une DLPDU ES par la DLE adressée permet à la DLE d'exécuter tout ou partie d'une séquence initialement programmée, dont le lancement est synchronisé par le LAS, soit

– jusqu'à ce que la séquence soit terminée, soit
– jusqu'à ce que la partie de la capacité de transmission de la liaison locale, qui était allouée par les paramètres DD de la DLPDU ES reçue exige le retour du jeton délégué.

NOTE Cette dernière situation peut se produire

A) sur de séquences permettant l'interruption;

B) sur des séquences permettant l'aboutement dynamique de demandes de DL-COMPEL-SERVICE;

C) lorsque la retransmission de la DLPDU ES initiatrice, ou éventuellement d'une autre DLPDU au sein de la séquence définie, a rendu nécessaire une capacité de liaison au-delà de celle qui a été attribuée.

Une DLPDU ES reçue doit être traitée par la DLE destinataire comme suit.

6.16.4.1 Actions requises de toutes les DLE

6.16.4.1.1 Lorsqu'elle est adressée à une adresse de DL autre qu'une adresse de DLSEP

Si l'adresse DL de destination spécifiée par la DLPDU désigne une adresse de DL de la DLE destinataire autre qu'une adresse de DLSEP, alors la DLE doit

a) informer la gestion DL locale de l'événement;
b) initier une transmission d'une DLPDU PR dans les limites d'une période de maximum-response-delay intervalles de temps,
à savoir V(MRD) × V(ST) durées d'octet, telles que mesurées à la DLE de réception d'une DLPDU RT.

NOTE La DLE destinataire rejette la DLPDU ES.

6.16.4.1.2 Lorsqu'elle est adressée à une adresse de DLSEP de la DLE

Si l'adresse DL de destination spécifiée par la DLPDU désigne une adresse de DLSEP de la DLE destinataire, alors la DLE destinataire doit

a) copier la valeur du paramètre de durée de la DLPDU vers compteur dégressif local de durée restante, C(RD);
b) assumer le jeton délégué;
c) lancer la transmission dans les limites d'une période de maximum-response-delay intervalles de temps,
à savoir V(MRD) × V(ST) durées d'octet, par rapport à la fin de la réception de la DLPDU ES, telles que mesurées à la DLE destinataire;
d) employer de façon répétée les critères de sélection suivants jusqu'à ce que le jeton délégué soit retourné au LAS, en lançant à chaque fois la transmission dans les limites d'une période de maximum-response-delay intervalles de temps, à savoir V(MRD) × V(ST) durées d'octet, par rapport à la fin de la transmission antérieure (sauf s'il s'agissait d'une transmission de CA, CD, CT ou ED), telles que mesurées à la DLE de réception de ES.

Le compteur C(RD) doit décrémenter et être géré comme spécifié en 4.7.1.11.

Les transactions dans la séquence associée à l'adresse de DLCEP désignée

1) doivent être lancées successivement;
2) débuter comme suit:
   - en commençant par la transaction initiale de la séquence, si le sous-champ" utilisation du jeton" de la DLPDU ES reçue spécifiait RESTART, ou
   - en continuant à partir du point auquel la séquence a été précédemment suspendue si le sous-champ "utilisation du jeton" de la DLPDU ES reçue spécifiait CONTINUE;
3) sous réserve de la contrainte que la capacité de liaison de cas le plus défavorable requise pour achever la transaction soit inférieure à la valeur alors courante du compteur dégressif de durée restante, C(RD).

6.16.4.1.2.1 Sélection de la prochaine transaction à exécuter

La DLE destinataire doit exécuter les transactions dans la séquence comme suit.

Si ce prochain élément de la séquence est une référence à une file d'attente de demandes utilisateur d'adresses de DL, Q_A(UR), alors

a) si l'élément est une référence au correspondant distant d'un DLCEP homologue ou abonné local (voir 8.4.2.1b)1)i)), alors la DLE doit
   1) émettre une DLPDU CD vers un point d'extrémité de DLC distant, avec des paramètres SD acheminant l'état du DLCEP local au DLCEP distant, comme spécifié en 8.2.2.1 et 8.2.2.4;
   2) attendre une réponse immédiate;

      NOTE La taille maximale de DLPDU de cette réponse immédiate peut être déterminée uniquement en fonction des paramètres de DLCEP négociés.
   3) avancer au prochain élément de la séquence qui avait été adressé par la DLPDU ES reçue;
b) si l'élément est une référence qui indique la nécessité d'accomplir un échange d'unitdata avec une adresse de DLSAP spécifiée à la de priorité de service spécifiée (voir 8.4.2.1c)1)i)), alors la DLE doit
   1) émettre une DLPDU ED ou CD vers cette adresse de DLSAP distante, à la priorité de service spécifiée, le cas échéant, comme spécifié en 8.2.2.1 et 8.2.2.4;
   2) attendre une réponse immédiate;

      NOTE La taille maximale de DLPDU de cette réponse immédiate peut être déterminée uniquement en fonction des paramètres de DLCEP négociés.
   3) avancer au prochain élément de la séquence qui avait été adressé par la DLPDU ES reçue.

   Autrement, la DLE doit sélectionner le premier élément de la Q_A(UR) référencée qui a été libéré pour transmission et qui exige une transmission ou retransmission totale ou partielle.
c) Si un tel élément existe, alors
   1) la DLE doit émettre la DLPDU appropriée à l'état de cet élément de cette Q_A(UR);
   2) si la DLPDU émise qui en résulte achève la transmission d'une DLSDU (de telle sorte qu'aucun segment de cette DLSDU ne reste à émettre), alors la DLE doit
      - faire progresser cet élément de Q_A(UR) vers la partition déjà envoyée, ou le supprimer, selon ce qui est approprié au type d'élément;
      - notifier à l'utilisateur de DLS l'achèvement de la demande, si cela est approprié ;
      - avancer au prochain élément de la séquence qui avait été adressé par la DLPDU ES reçue.
d) Si aucun élément de ce type de la file d'attente de demandes d'adresses de DL, Q_A(UR), référencée n'existe, alors

1) Si l'adresse de DL est
   – une adresse de DLCEP d'un DLCEP homologue ou éditeur local, ou
   – une adresse de DLCEP éditeur d'un DLCEP abonné local,

   et la liaison d'envoi du DLCEP est faite à un tampon qui n'est pas vide, alors
   i) la DLE doit émettre la DLPDU appropriée à l'état du DLCEP;
   ii) si la DLPDU émise qui en résulte achève la transmission d'une DLSDU (de telle sorte qu'aucun segment de cette DLSDU ne reste à émettre), alors la DLE doit notifier à l'utilisateur de DLS l'achèvement de la demande, le cas échéant.
2) Autrement, lorsque i) ne s'applique pas, et l'adresse de DL est une adresse de DLCEP d'un DLCEP homologue ou éditeur local, alors la DLE doit émettre une DLPDU DT ou RC, selon ce qui est approprié à l'état du DLCEP, avec un champ "données utilisateur" vide, pour acheminer l'état du DLCEP au(x) DLCEP homologue(s) ou abonné(s) correspondant(s).
3) Qu'une DLPDU ait été émise ou non, la DLE doit avancer au prochain élément de la séquence qui a été adressée par la DLPDU ES reçue.

Tous les autres cas sont erronés et sont interdits.

6.16.4.1.2.2 Considérations complémentaires

Les considérations suivantes s'appliquent également:

a) S'il n'y a pas d'éléments supplémentaires ans la séquence, et si une répétition immédiate de tentative de la transaction actuellement sélectionnée n'est pas possible, le sous-champ "utilisation finale de jeton" dans cette DLPDU autre que RT peut être mis à la valeur FINAL avant transmission, ce qui retourne le jeton délégué à la DLE LAS à la fin de la transaction courante.
b) Si la séquence est achevée, et le jeton délégué n'a pas été retourné en mettant un sous-champ "utilisation finale de jeton" à la valeur FINAL dans la dernière DLPDU émise par la DLE, alors la DLE doit retourner le jeton délégué en émettant une DLPDU RT.
c) Si à un instant quelconque
   1) le compteur dégressif de durée restante, C(RD), est inadéquat pour permettre toute utilisation ultérieure par la DLE destinataire;
   2) la séquence contient des éléments supplémentaires,

   alors la DLE destinataire doit retourner le jeton délégué au LAS en envoyant une DLPDU RI spécifiant l'intervalle minimal de délégation pour l'utilisation de jeton requis lorsque le jeton est ensuite retourné.

   NOTE Une DLPDU RI, qui est envoyée pour mettre fin à l'utilisation d'un jeton qui a été délégué par une DLPDU ES, indique au LAS destinataire que l'exécution de séquence était incomplète et qu'il convient qu'une autre DLPDU ES, avec un sous-champ "utilisation finale de jeton" spécifiant CONTINUE, soit envoyée à la même adresse de DLSEP.

6.16.4.2 Actions supplémentaires requises d'une DLE de classe Link-Master

Aucune.

6.16.4.3 Actions supplémentaires requises d'une DLE de classe Bridge

a) Étant donné que chaque DLE de classe Bridge a une capacité de maître de liaison, toutes les actions spécifiées en 6.16.4.2 s'appliquent également à une DLE de la classe Bridge.
b) La DLE ne doit pas transmettre la DLPDU.

6.16.4.4 Actions supplémentaires requises de la DLE LAS courante

Si la DLE ne venait juste pas d'envoyer la DLPDU, la DLE doit abandonner le jeton programmeur et désactiver son rôle comme LAS, et doit informer la gestion de DL locale de l'événement.

6.17 DLPDU Return token (Retourner un jeton (RT))

Une DLPDU RETURN TOKEN (RT) est utilisée pour retourner le jeton qui a été délégué par une DLPDU PT ou ES au LAS et indiquer qu'une délégation de jeton supplémentaire, à l'adresse de DL de la dernière DLPDU PT ou ES, est requise.

NOTE Le jeton délégué peut aussi être retourné en mettant l'appellation "utilisation finale de jeton" dans n'importe quelle DLPDU à la valeur FINAL, puis en cessant l'utilisation du jeton délégué après avoir émis cette DLPDU.

6.17.1 Structure de la DLPDU RT

Tableau 28 – Structure des DLPDU RT

Contrôle de trame
0011 0100

6.17.1.1 Champ "Frame control" (contrôle de trame)

Le champ "contrôle de trame" doit spécifier la fonction de la DLPDU.

6.17.1.2 Champ "Address" (adresse)

Le champ "adresse" doit être vide.

6.17.1.3 Champ "Parameters" (paramètres)

Le champ "paramètres" doit être vide.

6.17.1.4 Champ "User data" (données d'utilisateur)

Le champ "données d'utilisateur" doit être vide.

6.17.2 Contenu de la DLPDU RT

Le champ "contrôle de trame" doit être codé comme spécifié dans le Tableau 28.

Les champs "adresse", "paramètres" et "données d'utilisateur" doivent être vides.

6.17.3 Envoi de la DLPDU RT

Une DLPDU RT résulte uniquement de la réception d'une DLPDU PT ou ES. La DLPDU RT doit être envoyée comme cela est spécifié en 6.15.4 et en 6.16.4.

6.17.4 Réception de la DLPDU RT

Une DLPDU RT reçue doit être traitée comme suit par la DLE destinataire.

6.17.4.1 Actions requises de toutes les DLE

Aucune.

6.17.4.2 Actions supplémentaires requises d'une DLE de classe Link-Master

Aucune.

6.17.4.3 Actions supplémentaires requises d'une DLE de classe Bridge

Les considérations suivantes s'appliquent:

a) Étant donné que chaque DLE de classe Bridge a une capacité de maître de liaison, toutes les actions spécifiées en 6.17.4.2 s'appliquent également à une DLE de la classe Bridge.
b) La DLE ne doit pas transmettre la DLPDU.

6.17.4.4 Actions supplémentaires requises de la DLE LAS courante

Lorsque le jeton programmeur n'est pas dominant sur la liaison locale, alors

a) si le jeton de retour était délégué par une DLPDU PT, alors la DLE doit mettre à zéro la valeur de l'élément de rang V(DTA) de la matrice des temps de détention de jeton restants, V(RTHA) (voir 4.7.5.13);
b) la DLE doit supposer que le jeton programmeur est de nouveau dominant sur la liaison locale et reprendre le fonctionnement actif comme LAS.

6.18 DLPDU Request interval (Demander un intervalle (RI))

Une DLPDU REQUEST INTERVAL (RI) est utilisée pour retourner un jeton qui a été délégué par une DLPDU PT ou ES au LAS et à indiquer l'intervalle utile minimal de délégation requis pour la prochaine délégation suivante.

NOTE Le jeton délégué peut aussi être retourné en mettant l'appellation "utilisation finale de jeton" dans n'importe quelle DLPDU à la valeur FINAL, puis en cessant l'utilisation du jeton délégué après avoir émis cette DLPDU ou en envoyant une DLPDU RT.

6.18.1 Structure de la DLPDU RI

Tableau 29 – Structure des DLPDU RI

Contrôle de trame	Paramètres
0010 0000	DD-p

6.18.1.1 Champ "Frame control" (contrôle de trame)

Le champ "contrôle de trame" doit spécifier la fonction de la DLPDU.

6.18.1.2 Champ "Address" (adresse)

Le champ "adresse" doit être vide.

6.18.1.3 Champ "Parameters" (paramètres)

Le champ "paramètres" doit être structuré et codé tel que spécifié en 7.10.

6.18.1.4 Champ "User data" (données d'utilisateur)

Le champ "données d'utilisateur" doit être vide.

6.18.2 Contenu de la DLPDU RI

Le champ "contrôle de trame" doit être codé comme spécifié dans le Tableau 29. Le champ "adresse" doit être vide.

Le champ "paramètres" doit spécifier la durée utile minimale pour la prochaine délégation du jeton courant. Le champ "données d'utilisateur" doit être vide.

6.18.3 Envoi de la DLPDU RI

Une DLPDU RI résulte uniquement de la réception d'une DLPDU ES ou PT. La DLPDU RI doit être envoyée comme cela est spécifié en 6.15.4 et en 6.16.4.

6.18.4 Réception de la DLPDU RI

Une DLPDU RI reçue doit être traitée comme suit par la DLE destinataire.

6.18.4.1 Actions requises de toutes les DLE

Aucune.

6.18.4.2 Actions supplémentaires requises d'une DLE de classe Link-Master

Aucune.

6.18.4.3 Actions supplémentaires requises d'une DLE de classe Bridge

Les considérations suivantes s'appliquent:

a) Étant donné que chaque DLE de classe Bridge a une capacité de maître de liaison, toutes les actions spécifiées en 6.18.4.2 s'appliquent également à une DLE de la classe Bridge.
b) La DLE ne doit pas transmettre la DLPDU.

6.18.4.4 Actions supplémentaires requises de la DLE LAS courante

Lorsque le jeton programmeur n'est pas dominant sur la liaison locale, alors la DLE doit

a) si le jeton en cours de retour était délégué par une DLPDU PT, alors la DLE doit mettre la valeur de l'élément de rang V(DTA) de la matrice des temps de détention de jeton restants, V(RTHA) (voir 4.7.5.13) pour rendre compte de la capacité de liaison juste utilisée par la DLE;
b) si le jeton en cours de retour a été délégué par une DLPDU ES ou PT, alors la DLE doit transmettre le champ "paramètre DD" de cette DLPDU, accompagné de la V(DTA), aux fonctions de niveau supérieur de la DLE LAS (voir Article 9) pour reprogrammation tentée;
c) la DLE doit supposer que le jeton programmeur est de nouveau dominant sur la liaison locale et reprendre le fonctionnement actif comme LAS.

6.19 DLPDU Claim LAS (Réclamer le LAS (CL))

Une DLPDU CLAIM LAS (CL) (Réclamer le LAS) est utilisée par une DLE LM pour initialiser la liaison locale ou pour récupérer après une longue période de silence sur la liaison locale indiquant la défaillance du précédent LAS.

6.19.1 Structure de la DLPDU CL

Tableau 30 – Structure des DLPDU CL

Contrôle de trame	Adresse NODE source
0000 0001	N

6.19.1.1 Champ "Frame control" (contrôle de trame)

Le champ "contrôle de trame" doit spécifier

a) la fonction de la DLPDU;

b) la longueur, le nombre et le type d'adresses de DLPDU.

6.19.1.2 Champ "Address" (adresse)

Le champ "adresse" doit consister en une adresse de DL NODE source explicite.

6.19.1.3 Champ "Parameters" (paramètres)

Le champ "paramètres" doit être vide.

6.19.1.4 Champ "User data" (données d'utilisateur)

Le champ "données d'utilisateur" doit être vide.

6.19.2 Contenu de la DLPDU CL

Le champ "contrôle de trame" doit être codé comme spécifié dans le Tableau 30. Le champ "adresse" doit être l'adresse DL NODE de la DLE qui a détecté l'inactivité prolongée de la liaison, dont la valeur est l'id de nœud de l'expéditeur, V(TN). Les champs "paramètres" et "données d'utilisateur" doivent être vides.

NOTE Les objectifs principaux de l'adresse de DL NODE dans la DLPDU CL sont

a) permettre une détection améliorée des collisions au cours de la procédure de réclamation de LAS;

b) identifier la DLE qui a détecté l'inactivité à des DLE d'observation, ce qui peut avoir son utilité dans des systèmes ouverts réels pour identifier des DLE avec des types particulièrement obstructifs d'erreurs de PhE ou DLE, parfois caractérisées comme étant "émetteur défectueux" ou "récepteur sourd".

6.19.3 Envoi de la DLPDU CL

Une DLPDU CL doit résulter de la détection dans une DLE LM d'une durée de node-id intervalles de temps, à savoir V(TN) × V(ST) durées d'octet, d'inactivité continue de la liaison, impliquant la défaillance de la précédente DLE LAS, le cas échéant.

À la suite d'une telle détection, la DLE doit immédiatement envoyer une DLPDU CL, et doit de nouveau surveiller le support pendant node-id intervalles de temps, à savoir V(TN) × V(ST) durées d'octet, d'inactivité continue de la liaison après avoir envoyé la première DLPDU CL.

S'il n'est entendu de nouveau aucune activité (autre que potentiellement une DLPDU CL avec l'adresse de DL NODE qui venait juste d'être envoyée), alors la DLE LM expéditrice doit immédiatement envoyer une seconde DLPDU CL identique, après quoi la DLE doit choisir un nombre entier aléatoire uniformément distribué dans la plage zéro à trois, et doit surveiller le support pendant des intervalles de temps égaux à ce nombre, à savoir random(0..3) × V(ST) durées d'octet. Le choix aléatoire effectif doit être statistiquement indépendant des choix similaires effectués par d'autres DLE.

NOTE Cette exigence d'indépendance statistique réduit au maximum la probabilité de répétition de choix identiques par des appareils réels de construction identique.

S'il n'est entendu de nouveau aucune activité (autre que potentiellement une DLPDU CL avec l'adresse de DL NODE qui venait juste d'être envoyée), alors la DLE LM expéditrice doit activer ses fonctions LAS, assumer le jeton programmeur et démarrer le fonctionnement actif comme LAS.

6.19.4 Réception de la DLPDU CL

Une DLPDU CL reçue doit être traitée comme suit par la DLE destinataire.

6.19.4.1 Actions requises de toutes les DLE

Les considérations suivantes s'appliquent:

a) La DLPDU doit être rapportée à la gestion de DL locale de la DLE.
b) Si la DLE destinataire détient un jeton sur la liaison locale, la DLE destinataire doit abandonner immédiatement et doit informer la gestion de DL locale de l'événement.

6.19.4.2 Actions supplémentaires requises d'une DLE de classe Link-Master

Aucune.

6.19.4.3 Actions supplémentaires requises d'une DLE de classe Bridge

Les considérations suivantes s'appliquent:

a) Étant donné que chaque DLE de classe Bridge a une capacité de maître de liaison, toutes les actions spécifiées en 6.1.4.2 s'appliquent également à une DLE de la classe Bridge.
b) La DLE ne doit pas transmettre la DLPDU.

6.19.4.4 Actions supplémentaires requises de la DLE LAS courante

La DLE doit abandonner le jeton programmeur et désactiver son rôle comme LAS, et doit informer la gestion de DL locale de l'événement.

6.20 DLPDU Transfer LAS (Transférer le LAS (TL))

Une DLPDU TRANSFER LAS (TL) (Transfert de LAS) est utilisée par la DLE LAS courante pour transférer le jeton programmeur et le rôle de LAS à une autre DLE LM sur la liaison locale. La DLPDU TL n'est envoyée qu'à la suite d'une demande de la DLE LM adressée, et peut être rejetée si la DLE adressée décide que sa propre copie du programme de liaison locale n'est pas courante.

6.20.1 Structure de la DLPDU TL

Tableau 31 – Structure des DLPDU TL

Contrôle de trame	Adresse NODE de destination	User data (données d'utilisateur)
0000 0110	N	o-SPDU

6.20.1.1 Champ "Frame control" (contrôle de trame)

Le champ "contrôle de trame" doit spécifier

a) la fonction de la DLPDU;
b) la priorité implicite de la DLPDU, qui est URGENT;
c) la longueur, le nombre et le type d'adresses de DLPDU.

6.20.1.2 Champ "Address" (adresse)

Le champ "adresse" doit consister en une adresse de DL NODE de destination explicite.

6.20.1.3 Champ "Parameters" (paramètres)

Le champ "paramètres" doit être vide.

6.20.1.4 Champ "User data" (données d'utilisateur)

Le champ "données utilisateur" doit consister en une SPDU facultative de statut de base de données de LAS de niveau supérieur, de 64 octets ou moins.

6.20.2 Contenu de la DLPDU TL

Le champ "contrôle de trame" doit être codé comme spécifié dans le Tableau 31. Le champ "adresse" doit être l'adresse de DL NODE de la DLE vers laquelle le rôle de LAS est transféré. Le champ "paramètres" doit être vide.

Le champ "données utilisateur" achemine une SPDU d'état de base de données LAS de niveau supérieur (voir 9.3.2.3) dont la taille maximale est limitée comme spécifié en 6.20.1.4, ou doit être vide. Lorsqu'il est présent, le contenu de cette SPDU doit refléter les plus récentes activités de construction de programme, le cas échéant, avant d'envoyer la DLPDU TL.

6.20.3 Envoi de la DLPDU TL

La DLE LAS doit envoyer une DLPDU TL en réponse à la réception d'une demande de programmation explicite reçue par le LAS en provenance d'une autre DLE LM sur la liaison locale, demandant le transfert d'un jeton de programmation et le rôle de LAS associé. C'est une erreur de protocole que d'envoyer une DLPDU TL DLPDU vers une DLE non demandeuse.

NOTE 1 Une telle demande ne peut être faite que par le biais d'une SPDU de niveau supérieur envoyée aux fonctions de prise en charge de DL du LAS courant. La gestion DL n'a pas la permission de commander un tel transfert, même si elle peut inviter la DLE LM demandeuse à demander le transfert directement.

Si la gestion de DL le lui ordonne, la DLE LAS peut comparer l'adresse DL NODE de sa DLE LM associée à celle de la DLE demandeuse et rejeter la demande de transfert lorsque la DLE demandeuse détient une adresse DL NODE supérieure.

NOTE 2 Cet ordre de disposition améliore la prévisibilité de quelles DLE LM assumeront le rôle de LAS sur les liaisons locales qui n'emploient pas la détermination dynamique de cette DLE LM qui est la plus capable d'assumer le rôle de LAS.

Une DLPDU TL peut être envoyée sur la liaison lorsque

a) la DLE expéditrice, le LAS, détient un jeton programmeur qui est le jeton dominant sur la liaison locale;

b) la DLE LAS a achevé le dernier cycle de "circulation de jeton" et autrement lancerait le prochain cycle de "circulation de jeton" en envoyant explicitement une DLPDU PT à l'adresse DL NODE ayant le plus petit numéro représentée dans V(LL);

   NOTE Cette restriction sur le transfert du rôle LAS élimine la nécessité de transférer (de façon fiable) la valeur courante de la matrice V(RTHA) juste avant d'envoyer la DLPDU TL.

c) la durée allouée restante d'utilisation de jeton avant la prochaine activité programmée permet à la DLE LAS d'envoyer à répétitions une DLPDU TL et surveiller la liaison locale pendant une période de immediate-response-recovery-delay intervalles de temps, à savoir V(IRRD) × V(ST) durées d'octet, pour toute réponse immédiate issue de la DLE adressée.

Après avoir envoyé une DLPDU TL, la DLE LAS doit surveiller l'activité de la liaison locale pendant une période de immediate-response-recovery-delay intervalles de temps, à savoir V(IRRD) × V(ST) durées d'octet, en attente de réponse.

1) Si une "indication" Ph-Data (voir 4.4.4) rapportant DATA est reçue, alors la DLE doit
   – arrêter toute surveillance;
   – attendre ensuite de recevoir l'indication de Ph-Data rapportant END-OF-DATA-AND-ACTIVITY ou END-OF-ACTIVITY;
   – ensuite poursuivre comme en 3).

2) Si 1) ne s'applique pas et la période de surveillance expire alors

   i) si le bus n'est pas actif à cet instant-là (c'est-à-dire que la dernière "indication" de Ph-Data reçue rapportait END-OF-ACTIVITY), la DLE doit poursuivre comme en 4).

   ii) si i) ne s'applique pas, ce qui implique que la liaison est encore active à cet instant-là (c'est-à-dire que la dernière "indication" de Ph-Data reçue rapportait START-OF-ACTIVITY), alors

      α) Pour les mises en œuvre qui n'étaient pas démontrables le 31 décembre 1995 ou avant, la DLE doit surveiller la liaison locale pendant une période d'un intervalle de temps supplémentaire, à savoir V(ST) durées d'octet, attendant une "indication" de Ph-Data

         A) Si une "indication" de Ph-Data rapportant DATA est reçue, alors la DLE doit poursuivre comme en 1).

         B) Si une "indication" de Ph-Data rapportant END-OF-ACTIVITY est reçue, et A) ne s'applique pas, alors la DLE doit arrêter toute surveillance et poursuivre comme en 4d).

         C) Si ni A) ni B) ne s'appliquent et la période de surveillance expire avant qu'une "indication" de Ph-Data ne soit reçue, la DLE doit

            – attendre de recevoir l'indication de Ph-Data rapportant END-OF-DATA-AND-ACTIVITY ou END-OF-ACTIVITY;

            – ensuite poursuivre comme en 3).

      β) Les mises en œuvre qui étaient démontrables le 31 décembre 1995, ou avant, peuvent en variante juste poursuivre comme dans α).C) lorsque ni α).A) ni α).B) ne s'appliquent.

3) Si l'activité de la liaison était suffisante pour inférer que DATA a été reçu comme en 1), 2)ii)α)A) ou 2)ii)α)C), alors

   i) Si l'activité de la liaison n'a pas donné de DLPDU, alors la DLE LAS doit abandonner son jeton programmeur et doit continuer ou reprendre la surveillance de la liaison locale jusqu'à ce que 3)i)A) ou 3)i)B) se produise.

   NOTE Cette surveillance permet la récupération du rôle LAS lorsque le transfert échoue.

      A) Si la DLE de surveillance observe une DLPDU (avec FCS correcte), la DLE de surveillance doit cesser de surveiller pour la récupération du rôle de LAS, et doit informer la gestion de DL locale de l'événement.

      B) Si la DLE de surveillance observe une période de 15 intervalles de temps, à savoir 15 × V(ST) durées d'octet, d'inactivité continue sur la liaison locale, la DLE de surveillance doit

         – assumer de nouveau le jeton programmeur;

         – informer la gestion de DL locale de l'événement;

         – reprendre le fonctionnement actif comme LAS, et commencer la transmission sur la liaison.

   ii) Si l'activité de la liaison a donné une DLPDU (avec FCS correcte), alors

      A) si la DLPDU reçue est une DLPDU SR, spécifiant la cause de "défaillance – LAS transfer rejected (Transfert LAS rejeté)", alors la DLE de surveillance doit

         – assumer de nouveau le jeton programmeur;

         – informer la gestion de DL locale de l'événement;

         – reprendre le fonctionnement actif comme LAS, et commencer la transmission sur la liaison.

      B) sinon si la DLPDU reçue est n'importe quelle autre DLPDU, alors la DLE de surveillance doit cesser la surveillance pour la récupération du rôle LAS, et doit informer la gestion de DL locale de l'événement.

4) Il n'y a eu aucun activité de liaison ou l'activité sur la liaison n'était pas suffisante pour inférer que DATA a été reçu comme en 2)i) ou 2)ii)α)B), alors la DLE LAS peut répéter la

tentative d'envoyer de nouveau de la DLPDU TL, jusqu'au nombre de fois spécifié comme étant le nombre maximal de répétitions de tentative V(MRC) (voir 4.7.1.5), sur la liaison locale.

Si toutes les répétitions de tentative sont infructueuses, la DLE LAS doit

– doit conserver le jeton programmeur et continuer le fonctionnement actif comme LAS;
– informer la gestion de DL locale de l'événement;
– démarrer la prochaine transmission dans la limite de "token-recovery-delay" intervalles de temps, à savoir P(TRD) × V(ST) durées d'octet, par rapport au début de la période courante d'inactivité de la liaison.

6.20.4 Réception de la DLPDU TL

Une DLPDU TL reçue doit être traitée comme suit par la DLE destinataire.

6.20.4.1 Actions requises de toutes les DLE

La DLPDU doit être rapportée à la gestion de DL locale de la DLE.

6.20.4.2 Actions supplémentaires requises d'une DLE de classe Link-Master

Si l'adresse de DL de destination spécifiée par la DLPDU désigne l'adresse de DL NODE de la DLE destinataire, alors

a) Si
   1) la DLE destinataire n'est pas en attente de réception de la DLPDU TL;
   2) la DLE destinataire ne peut pas exécuter le programme existant, soit en raison de la complexité ou de la longueur du programme, soit parce que la DLE destinataire ne peut pas satisfaire à la contrainte de surdébit de programme, V(MSO) (voir 4.7.5.6), intégrée au programme existant; ou
   3) si les informations relatives à la construction du programme et à la liste active acheminées dans la SPDU de statut de base de données de LAS au sein de la DLPDU TL reçue indiquent que la DLE destinataire n'a pas une copie courante du programme et/ou de la liste active,

   alors la DLE destinataire doit répondre par une DLPDU SR dans une période de maximum-response-delay intervalles de temps, à savoir V(MRD) × V(ST) durées d'octet, telles que mesurées à la DLE destinataire, avec un statut de "défaillance – LAS transfer rejected (Transfert LAS rejeté)".

b) Autrement, lorsque a) ne s'applique pas, la DLE destinataire doit assumer le jeton programmeur, activer ses fonctions de LAS, et recommencer le fonctionnement comme LAS.

6.20.4.3 Actions supplémentaires requises d'une DLE de classe Bridge

Les considérations suivantes s'appliquent:

a) Étant donné que chaque DLE de classe Bridge a une capacité de maître de liaison, toutes les actions spécifiées en 6.20.4.2 s'appliquent également à une DLE de classe Bridge.
b) La DLE ne doit pas transmettre la DLPDU.

6.20.4.4 Actions supplémentaires requises de la DLE LAS courante

Si la DLE ne venait juste pas d'envoyer la DLPDU, la DLE doit abandonner le jeton programmeur et désactiver son rôle comme LAS, et doit informer la gestion de DL locale de l'événement.

6.21 DLPDU Wakeup (Réveil (WK))

Une DLPDU WAKEUP (WK) (Réveil) est utilisée par la DLE LAS courante pour stimuler une DLE de facteur d'utilisation fractionnaire (FDC) de classe B ou de classe C sur la liaison locale (voir.3.3.32) à passer à un état pleinement opérationnel, avec possibilité de communication directe.

La transmission de la DLPDU WK se produit habituellement à la suite d'une programmation explicite préalable, soit par la DLE FDC adressée, soit par la gestion de DL. Toutefois, la DLE LAS, qui doit aussi être une DLE pont pour prendre en charge les DLE FDC, peut émettre la DLPDU WK de sa propre initiative, généralement lorsque, dans sa fonction pont, la file d'attente de DLPDU attendant pour une transmission vers la DLE FDC dépasse un certain seuil interne.

6.21.1 Structure de la DLPDU WK

Tableau 32 – Structure des DLPDU WK

Contrôle de trame	Adresse NODE de destination
0000 0000	N

6.21.1.1 Champ "Frame control" (contrôle de trame)

Le champ "contrôle de trame" doit spécifier

a) la fonction de la DLPDU;

b) la priorité implicite de la DLPDU, qui est URGENT;

c) la longueur, le nombre et le type d'adresses de DLPDU.

6.21.1.2 Champ "Address" (adresse)

Le champ "adresse" doit consister en une adresse de DL NODE de destination explicite.

6.21.1.3 Champ "Parameters" (paramètres)

Le champ "paramètres" doit être vide.

6.21.1.4 Champ "User data" (données d'utilisateur)

Le champ "données d'utilisateur" doit être vide.

6.21.2 Contenu de la DLPDU WK

Le champ "contrôle de trame" doit être codé comme spécifié dans le Tableau 32. Le champ "adresse" doit être l'adresse de DL NODE de la DLE qui doit recevoir la DLPDU WK. Les champs "paramètres" et "données d'utilisateur" doivent être vides.

NOTE Le codage spécifique de la DLPDU WK réduit au maximum la quantité de circuits de reconnaissance de DLPDU actives requis dans une DLE FDC "endormie".

6.21.3 Envoi de la DLPDU WK

La DLE LAS doit envoyer une DLPDU WK lorsque la programmation explicite préalable le lui demande. La capacité de liaison utilisée pour l'envoi de la DLPDU WK doit être déduite (dans toute la mesure possible) de la durée allouée restante d'utilisation du jeton programmeur pour le trafic programmé, qui est de V(MST) × V(TTRT) durées d’octet, dans le cycle courant de "circulation du jeton".

Une DLE pont peut également envoyer une DLPDU WK quand les propres fonctions de transmission de pont de la DLE le lui demandent lorsque la DLE détient un jeton programmeur ou un jeton délégué qui est le jeton dominant sur la liaison locale, et lorsque la durée allouée restante d'utilisation du jeton avant à la prochaine activité programmée permet l'achèvement de la transmission de la DLPDU avant l'expiration du jeton

6.21.4 Réception de la DLPDU WK

Une DLPDU WK reçue doit être traitée comme suit par la DLE destinataire.

6.21.4.1 Actions requises de toutes les DLE

Si l'adresse de DL de destination spécifiée par la DLPDU désigne l'adresse DL NODE, la DLE doit activer le reste de ses fonctions DLE et se préparer à une communication active sur la liaison locale.

NOTE La réception de cette DLPDU n'a aucun impact sur les DLE non FDC.

6.21.4.2 Actions supplémentaires requises d'une DLE de classe Link-Master

Aucune.

6.21.4.3 Actions supplémentaires requises d'une DLE de classe Bridge

Les considérations suivantes s'appliquent:

a) Étant donné que chaque DLE de classe Bridge a une capacité de maître de liaison, toutes les actions spécifiées en 6.21.4.2 s'appliquent également à une DLE de classe Bridge.
b) La DLE ne doit pas transmettre la DLPDU.

6.21.4.4 Actions supplémentaires requises de la DLE LAS courante

Aucune.

6.22 DLPDU Idle (Repos (IDLE))

Une DLPDU IDLE est utilisée par le détenteur courant du jeton pour placer une DLPDU sur la liaison locale qui est ignorée par toutes les DLE, pour toute autre raison que son impact sur le sens d'activité de liaison de chaque DLE, mais peut acheminer des informations vers un appareil d'analyse de liaison de données ou en déclencher un.

6.22.1 Structure de la DLPDU IDLE

Tableau 33 – Structure des DLPDU IDLE

Contrôle de trame	Données utilisateur
0001 0F10	o-DLSDU

6.22.1.1 Champ "Frame control" (contrôle de trame)

Le champ "contrôle de trame" doit spécifier

a) la fonction de la DLPDU;
b) si, oui ou non, la transmission de la DLPDU courante met fin à l'utilisation d'un jeton délégué.

6.22.1.2 Champ "Address" (adresse)

Le champ "adresse" doit être vide.

6.22.1.3 Champ "Parameters" (paramètres)

Le champ "paramètres" doit être vide.

6.22.1.4 Champ "User data" (données d'utilisateur)

Le champ "données" doit consister en une seule DLSDU dont la taille maximale est le maximum autorisé pour une DLPDU de priorité URGENT. Ce champ "données" peut être vide (null).

6.22.2 Contenu de la DLPDU IDLE

Le champ "contrôle de trame" doit être codé comme spécifié dans le Tableau 33. Les champs "adresse" et "paramètre" doivent être vides.

Le contenu du champ "données utilisateur", éventuellement vide, n'est pas spécifié par la présente norme.

6.22.3 Envoi de la DLPDU IDLE

Une DLPDU IDLE peut être envoyée sur la liaison lorsque la DLE expéditrice détient un jeton programmeur ou un jeton délégué qui est le jeton dominant sur la liaison locale, et lorsque la durée allouée restante d'utilisation du jeton, C(RD), permet l'achèvement de la transmission de la DLPDU IDLE avant l'expiration du jeton.

NOTE La DLE LAS utilise la DLPDU IDLE pour remplir la capacité de liaison autrement inutilisée jusqu'au moment de la prochaine activité de liaison programmée, ce qui empêche d'autres DLE d'inférer une défaillance de la DLE LAS à partir de l'absence d'activité sur la liaison. Pour faciliter cette utilisation, la DLPDU IDLE a une taille minimale requise d'un octet.

Une demande de la gestion locale peut conduire à la transmission d'une DLPDU IDLE.

Si la DLE détient un jeton délégué et cette DLPDU est la DLPDU finale qu'il est nécessaire d'envoyer sur cette utilisation du jeton délégué, alors la DLE peut mettre à la valeur FINAL le sous-champ "utilisation finale du jeton" de la DLPDU IDLE; sinon, ce sous-champ doit avoir la valeur NOT-FINAL.

6.22.4 Réception de la DLPDU IDLE

Une DLPDU IDLE reçue doit être traitée comme suit par la DLE destinataire.

6.22.4.1 Actions requises de toutes les DLE

La DLE doit ignorer la DLPDU à la réception, excepté pour les possibles rapports de gestion locale ou besoins spécifiques d'instrumentation.

6.22.4.2 Actions supplémentaires requises d'une DLE de classe Link-Master

Aucune.

6.22.4.3 Actions supplémentaires requises d'une DLE de classe Bridge

Les considérations suivantes s'appliquent:

a) Étant donné que chaque DLE de classe Bridge a une capacité de maître de liaison, toutes les actions spécifiées en 6.22.4.2 s'appliquent également à une DLE de classe Bridge.
b) La DLE ne doit pas transmettre la DLPDU.

6.22.4.4 Actions supplémentaires requises de la DLE LAS courante

Si le sous-champ "utilisation finale de jeton" de la DLPDU reçue a la valeur FINAL, la DLE LAS doit

a) supposer que l'utilisation courante du jeton délégué est terminée comme si le jeton avait été retournée par une DLPDU RT (voir 6.17);

b) supposer que le jeton programmeur est de nouveau dominant sur la liaison locale et reprendre le fonctionnement actif comme LAS.

6.23 DLPDU Spare (De rechange)

Une DLPDU SPARE est réservée pour usage ultérieur normalisé et ne peut être envoyée par aucune DLE.

6.23.1 Points de code de la DLPDU SPARE

Tableau 34 – Structure supposée des DLPDU (spare) indéfinies

Format	Contrôle de trame
1	0000 0x11
2	0010 0x01
3	00xx 1xxx
4	0100 xxxx
5	101x xx00

Une DLPDU SPARE doit consister en un champ "contrôle de trame". Chaque point de code de FC d'une "spare" est réservé à une future utilisation normalisée.

Le format et le contenu de DLPDU des points de code d'une "spare" sont sujets à modification dès qu'ils seront utilisés dans une révision de la présente norme. Entre-temps, ils ne doivent pas être utilisés.

6.23.1.1 Champ "Frame control" (contrôle de trame)

Le champ "contrôle de trame" doit spécifier la fonction de la DLPDU – spare reserved (de rechange et réservée). Aucune priorité de DLPDU n'est impliquée.

6.23.1.2 Champ "Address" (adresse)

Le champ "adresse" doit être vide.

6.23.1.3 Champ "Parameters" (paramètres)

Le champ "paramètres" doit être vide.

6.23.1.4 Champ "User data" (données d'utilisateur)

Le champ "données d'utilisateur" doit être vide.

6.23.2 Contenu d'une DLPDU SPARE

Le champ "contrôle de trame" doit être codé comme spécifié dans le Tableau 34. Les champs "adresse", "paramètres" et "données d'utilisateur" doivent être vides.

6.23.3 Envoi d'une DLPDU SPARE

Les DLPDU SPARE ne peuvent pas être envoyées ou transmises.

6.23.4 Réception d'une DLPDU SPARE

Une DLPDU SPARE reçue doit être traitée comme suit par la DLE destinataire.

6.23.4.1 Actions requises de toutes les DLE

La DLE doit ignorer la DLPDU à la réception, excepté pour les possibles rapports de gestion locale ou besoins spécifiques d'instrumentation.

6.23.4.2 Actions supplémentaires requises d'une DLE de classe Link-Master

Aucune.

6.23.4.3 Actions supplémentaires requises d'une DLE de classe Bridge

Les considérations suivantes s'appliquent:

a) Étant donné que chaque DLE de classe Bridge a une capacité de maître de liaison, toutes les actions spécifiées en 6.23.4.2 s'appliquent également à une DLE de classe Bridge.
b) La DLE ne doit pas transmettre la DLPDU.

6.23.4.4 Actions supplémentaires requises de la DLE LAS courante

Aucune.

6.24 DLPDU Reserved (not to be used) (Réservées, non utilisées)

Une DLPDU RESERVED (NOT TO BE USED) (ne devant pas être utilisée) est réservée pour usage normalisé ultérieur et ne peut être envoyée par aucune DLE.

6.24.1 Points de code de DLPDU RESERVED (NOT TO BE USED) (ne devant pas être utilisées)

Tableau 35 – Structure supposée des DLPDU RESERVED (NOT TO BE USED) (Réservées, ne devant pas être utilisées)

Format	Contrôle de trame
1	0000 0010
2	0000 010X
3	0010 001X
4	0010 0100
5	0011 0000
6	0101 1x00
7	1001 1x00
8	110x 1x00

Le point de code de FC pour une DLPDU RESERVED (NOT TO BE USED) ne doit pas être utilisé; ces points de code sont réservés afin de simplifier le décodage des octets de FC octets des autres DLPDU définies. L'ensemble de ces points de code réservés est montré dans le Tableau 35.

Le champ "contrôle de trame" doit spécifier la fonction des DLPDU – reserved (not to be used) (c'est-à-dire: réservées ne devant pas être utilisées). Il ne doit pas être utilisé.

NOTE Étant donné que ces points de code de FC sont réservés pour simplifier le décodage des points de code de FC des DLE, leur reconnaissance comme étant distincts des points de code FC apparentés du point de vue de la structure ne peut être mandatée.

6.24.2 Contenu d'une DLPDU RESERVED (NOT TO BE USED)

Aucun contenu n'est défini, car les DLPDU ayant ces points de code ne peuvent pas être émises.

6.24.3 Envoi d'une DLPDU RESERVED (NOT TO BE USED)

Les DLPDU spécifiant des points de code de FC RESERVED (NOT TO BE USED) FC ne doivent pas être émises. Par contre, elles peuvent être transmises par un pont si elles n'avaient pas été rejetées à la réception.

6.24.4 Réception d'une DLPDU RESERVED (NOT TO BE USED)

Si la DLE destinataire établit une distinction entre les DLPDU RESERVED (NOT TO BE USED) et les autres DLPDU, il convient que la DLE ignore la DLPDU RESERVED (NOT TO BE USED) à la réception (excepté pour les possibles rapports de gestion locale ou besoins spécifiques d'instrumentation). Toutefois, l'aptitude à établir une telle distinction et à effectuer ces rejets n'est pas requise.

7 Structure et codage de paramètres de DLPDU

Les variables Boolean (booléennes) sont toutes codées avec une représentation commune:

0: FALSE (faux)

1: TRUE (vrai)

Le numéro de version de protocole DL de 3 bits (VVV), qui est utilisé dans un certain nombre des paramètres de DLPDU et des SPDU (voir Article 11), doit avoir la valeur un (1).

7.1 Structure et codage des EC-PARAMETERS (paramètres EC)

Une DLPDU EC est utilisée pour établir une DLC d'homologue entre deux utilisateurs de DLS, ou une DLC à plusieurs homologues entre un utilisateur de DLS éditeur et des utilisateurs DLS abonnés.

Tableau 36 – Structure des paramètres d'une DLPDU EC

Attributs de base de DLC	Attributs de DLC lors de l'envoi	Attributs de DLC lors de la réception

Le champ "connection parameters" (EC-parameters) (paramètres de connexion, paramètres EC), qui se divise grosso modo comme montré dans le Tableau 36, doit spécifier les paramètres pour la DLC proposée:

a) Les deux premiers octets, dans l'ordre tel que montré dans le Tableau 37 et dans le Tableau 38, spécifient les attributs de base de la DLC, en commençant par le numéro de version du protocole de DL:

Tableau 37 – Paramètres EC: 1er octet

Demande de réponse	Discriminateur de réutilisation d'adresse de DLCEP éditeur			Diversité de chemins	Version de protocole de DL		
R	NNN			Q	VVV		
7	6	5	4	3	2	1	0

1) Un sous-champ (R) "Reply-Request" (demande de réponse) de 1 bit, spécifiant si une réponse est demandée (=1) ou non (=0);

NOTE Ce champ est fourni pour assurer une action correcte indépendamment de tout état du DLCEP destinataire.

2) Un sous-champ "publisher-DLCEP-address reuse-discriminator " (discriminateur de réutilisation d'adresse de DLCEP éditeur) de 3 bits (NNN), désignant l'ensemble courant de paramètres EC associés à une adresse de DLCEP PUBLISHER

   i) codé comme étant zéro, lorsque la classe de DLCEP (CC) de l'expéditeur n'est pas PUBLISHER;

   ii) assigné par la DLE, lorsque la classe de DLCEP (CC) de l'expéditeur est PUBLISHER

NOTE La DLE assigne une valeur à ce sous-champ chaque fois qu'elle reçoit une demande d'initier un nouveau DLCEP PUBLISHER issue soit d'un utilisateur de DLS local dans une primitive "request" de DL-CONNECT, soit d'un utilisateur de DLS distant dans une DLPDU EC. Le choix d'une valeur peut être aléatoire, ou peut être basé sur la connaissance de valeurs récemment utilisées que la DLE évite.

Ce sous-champ est utilisé pour discerner entre deux DLC différentes en utilisant la même adresse de DLCEP éditeur. Ceci pourrait arriver lorsque la première DLC avait été déconnectée et la DLE d'éditeur utilisait la même adresse de DLCEP pour établir la seconde DLC, qui est différente de la première DLC. Si un abonné à la première DLC n'a pas reçu la DLPDU DC pour cette DLC, il pourrait utiliser ce champ de la DLPDU EC qui vient juste d'être reçue pour déterminer que la DLPDU EC reçue est destinée à une DLC différente. Par conséquent, s'il convient que le DLCEP éditeur envoie une DLPDU EC pour la même DLC, il ne doit pas changer ce sous-champ.

3) un sous-champ "Path-diversity" (diversité de chemins) de DLL d'un bit (Q), codé comme

   0: any-path (n'importe quel chemin);

   1: this-path – (ce chemin) utiliser le chemin sur lequel cette DLPDU EC a été reçue;

NOTE Ce sous-champ fournit un moyen par lequel une DLE peut restreindre toutes les communications en un DLCEP à un chemin de DL spécifique, procurant ainsi un moyen de soumettre à essai le chemin de DL spécifique. Le moyen par lequel la valeur THIS-PATH est sélectionnée et par lequel le chemin réel utilisé est choisi, relève d'une initiative locale à une DLE.

4) un numéro de version de protocole DL à 3 bits (VVV), dont la valeur est spécifiée au début de l'Article 7;

Tableau 38 – Paramètres EC: 2ème octet

Classe de DLCEP d'expéditeur		Priorité de DL		Taille d'adresse de DL		Authentification de DLPDU	
CC		PP		SS		X	
7	6	5	4	3	2	1	0

5) un sous-champ "ender's DLCEP-class" (classe de DLCEP d'expéditeur (CC)) de deux bits, codé comme:

   00: réservé pour une utilisation normalisée ultérieure;

   01: DLCEP HOMOLOGUE;

   10: DLCEP EDITEUR;

   11: DLCEP ABONNE;

6) un sous-champ " DLL priority" (priorité de DLL (PP) de deux bits, codant la priorité de DLL comme (voir 5.2.1.3):

00: réservé pour une utilisation normalisée ultérieure;

01: URGENT (haute) priorité;

10: NORMAL (moyenne) priorité;

11: TIME-AVAILABLE (basse) priorité:

7) un sous-champ "DLL address size" (taille d'adresse de DLL (SS)) de deux bits, codant la taille de l'adresse de DLL proposée (voir 5.2.1.1) comme:

00: VERY-SHORT – les adresses sont omises lorsque cela est possible;

01: SHORT;

10: LONG;

11: réservé pour une utilisation normalisée ultérieure;

8) un sous-champ "DLPDU-authentication" (authentification de DLPDU (XX)) de deux bits, codant l'authentification de DLPDU requise (voir 4.2.4) as:

00: ORDINARY;

01: réservé pour une utilisation normalisée ultérieure;

10: SOURCE;

11: MAXIMAL;

b) Les quatre octets suivants, dans l'ordre tel que montré dans le Tableau 39 et le Tableau 40, spécifient respectivement le retard maximal de confirmation, en unités de 1 ms, pour les primitives "request" des services DL-CONNECT, DL-RESET et DL-SUBSCRIBER-QUERY de l'expéditeur et les primitives "request" de DL-DATA. La valeur d'utilisateur de DLS UNLIMITED doit être codée comme étant la valeur $FFFF_{16}$ (des uns partout).

Tableau 39 – Paramètres EC: 3ème et 4ème octets

Retard maximal de confirmation expéditeur pour les demandes de DL-CONNECT, DL-RESET et DL-SUBSCRIBER-QUERY															
MCD_CRS															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

et

Tableau 40 – Paramètres EC: 5ème et 6ème octets

Retard maximal de confirmation expéditeur pour les demandes de DL-DATA															
MCD_D															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

NOTE Chaque retard maximal de confirmation maximum de deux octets est envoyé avec l'octet de poids fort le premier.

c) Les quatre octets suivants, dans l'ordre tel que montré du Tableau 41 au Tableau 43, spécifient les attributs proposés pour le sens d'envoi de flux de données de l'expéditeur

Tableau 41 – Paramètres EC: 7ème octet

Attributs de DLC d'envoi de l'expéditeur							
Caractéristiques de livraison de données de DLCEP		Activité résiduelle	File d'attente / tampon	Taille maximale de fenêtre			
TT_S		A_S	B_S	$WWWW_S$			
7	6	5	4	3	2	1	0

1) un sous-champ "sending DLCEP data-delivery features " (caractéristiques de livraison de données de DLCEP d'envoi (TT_S)) de deux bits, codé comme:

   00: DLC UNORDERED;

   01: DLC ORDERED;

   10: DLC DISORDERED;

   11: DLC CLASSICAL;

2) un sous-champ "residual activity" (activité résiduelle (A_S)) d'un seul bit, indiquant si, oui ou non, l'activité résiduelle dans le sens spécifié (expéditeur vers destinataire) sera fournie lorsqu'il n'y a pas de DLSDU non acquittées, codé comme un Boolean;

   NOTE L'activité résiduelle est requise lorsque l'authentification de DLPDU MAXIMAL est spécifiée, ou à l'instigation de l'utilisateur de DLS ou de la gestion de DL. D'autres méthodes de demande d'activité résiduelle sont pour étude ultérieure.

3) un sous-champ "buffer/queue" (tampon/file (B_S)) d'un seul octet, indiquant si la source dans le sens spécifié (expéditeur vers destinataire) est un tampon (=1) ou une file d'attente (=0);

4) un sous-champ "window-size" (taille de fenêtre ($WWWW_S$)) de quatre bits, indiquant le nombre maximal de DLSDU précédemment envoyées qui seront disponibles pour retransmission, et codé comme zéro lorsque la transmission dans le sens en question n'est pas permise sur la DLC;

Tableau 42 – Paramètres EC: 8ème octet

Attributs de DLC d'envoi de l'expéditeur (suite)							
Format de paramètres de DLC de base			échange bidirectionnel de données	reserved (réservé)	Opportunité incluse	Format de marqueur temporel	
FFF_S			E_S	0	G_S	HH_S	
7	6	5	4	3	2	1	0

5) un sous-champ "format" (FFF_S) de trois bits, donnant le format désiré de la partie "paramètres de DLC de base" des "Status-Data-parameters" (paramètres de données de statut) (paramètres SD) dans le sens spécifié, codé comme nombre entier de trois bits:

   0: format A – [7.4.2.1A)];
   1: format B – [7.4.2.1B)];
   2: format C – [7.4.2.1C)];
   3: réservé pour une utilisation normalisée ultérieure;
   4: format D – [7.4.2.1D)];
   5: format E – [7.4.2.1E)];
   6: format F – [7.4.2.1F)];
   7: format G – [7.4.2.1G)].

6) un sous-champ "2-way data exchange" (échange bidirectionnel de données (E_S)) d'un seul bit, indiquant si les données de DLSDU peuvent être envoyées dans le sens expéditeur vers destinataire par les DLPDU ED, codé comme un Boolean;

7) un sous-champ d'un bit réservé pour utilisation dans des normes ultérieures, codé comme zéro;

8) un sous-champ "Timeliness-included " (opportunité incluse (G_S) d'un seul bit, indiquant la présence d’un paramètre d'opportunité (voir 7.4.2.1B)3) et 7.4.2.1C)5)) associé à des DLSDU émises dans des DLPDU émises, codé comme un Boolean;

9) un sous-champ "time-stamp-le format" (le format de marqueur temporel (H_S)) de deux bits, indiquant si un marqueur de temps de DL (voir 7.4.2.2) est inclus dans les paramètres SD qui accompagnent les DLSDU émises, codé comme:

   00: format J – [7.4.2.2J)], qui est un champ vide ne comportant pas de marqueur temporel;

   01: format K – [7.4.2.2K)], qui est un champ de deux octets avec une période de marqueur temporel supérieure à 2 s;

   10: format L – [7.4.2.2L)], qui est un champ de trois octets avec une période de marqueur temporel supérieure à 8 min ;

   11: format M – [7.4.2.2M)], qui est un champ de six octets avec une période de marqueur temporel supérieure à 100 années;

Tableau 43 – Paramètres EC: 9ème et 10ème octets

Attributs de DLC d'envoi de l'expéditeur (suite)															
Maximum DLSDU size (taille maximale de DLSDU)															
M...M_S															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

10) un sous-champ de deux octets (M...M_S), spécifiant la DLSDU de taille maximale qui peut être envoyée sur la DLC.

NOTE 1 La taille maximale de deux octets est émise l'octet de poids fort le premier.

NOTE 2 La plage de chaque paramètre est dépendante de la priorité de la DLC, et peut s'étendre entre zéro et 16 fois la longueur maximale de données spécifiée en 5.2.4 pour la priorité correspondante, inclusivement.

d) Les quatre octets suivants, identiques quant à la forme à ceux spécifiés en c) et mis dans l'ordre tel que montré du Tableau 44 au Tableau 46, spécifient les attributs proposés pour le sens de réception de flux de données de l'expéditeur:

Tableau 44 – Paramètres EC: 11ème octet

Attributs de DLC de réception de l'expéditeur							
Caractéristiques de livraison de données de DLCEP		Activité résiduelle	File d'attente / tampon	Taille maximale de fenêtre			
TT_R		A_R	B_R	$WWWW_R$			
7	6	5	4	3	2	1	0

1) un sous-champ "DLCEP data delivery features" (caractéristiques de livraison de données de DLCEP (TT_R)) de réception de deux bits), codé tel que spécifié dans c)1);

2) un sous-champ "residual activity" (activité résiduelle (A_R)) d'un seul bit, indiquant si, oui ou non, l’activité résiduelle dans le sens spécifié (destinataire vers expéditeur) est requise, codé tel que spécifié en c)2);

NOTE Voir la note après c)2).

3) un sous-champ "buffer/queue" (tampon/file (B_R)) d'un seul octet, indiquant si le collecteur dans le sens spécifié (destinataire vers expéditeur) est un tampon ou une file d'attente, codé tel que spécifié en c)3);

4) un sous-champ "window-size" (taille de fenêtre ($WWWW_R$)) de quatre bits, indiquant le nombre maximal de DLSDU non acquittées qui peuvent être reçues utilement;

Tableau 45 – Paramètres EC: 12ème octet

Attributs de DLC de réception de l'expéditeur (suite)							
Format de paramètres de DLC de base			échange bidirectionnel de données	reserved (réservé)	Opportunité incluse	Format de marqueur temporel	
FFF_R			E_R	0	G_R	HH_R	
7	6	5	4	3	2	1	0

5) un sous-champ "format" (FFF_R)) de trois bits, donnant le format désiré de la partie "paramètres de DLC de base" des paramètres SD dans le sens spécifié, codé tel que spécifié en c)5).

6) un sous-champ "2-way data exchange" (échange bidirectionnel de données (E_R)) d'un seul bit, indiquant si les données de DLSDU peuvent être envoyées dans le sens destinataire vers expéditeur par les DLPDU ED, codé comme un Boolean;

7) un sous-champ d'un bit réservé pour utilisation dans des normes ultérieures, codé comme zéro;

8) un sous-champ "Timeliness-included " (opportunité incluse (G_R)) d'un seul bit, indiquant la présence dans les DLPDU reçues d'un paramètre d'opportunité (voir 7.4.2.1b)3) et 7.4.2.1c)5)) associé à des DLSDU reçues, codé comme un Boolean;

9) un sous-champ "time-stamp-format" (format de marqueur temporel (HH_R)) de deux bits, indiquant si un marqueur de temps de DL est inclus dans les paramètres SD qui accompagnent les DLSDU reçues, codé tel que spécifié en c)9);

Tableau 46 – Paramètres EC: 13ème et 14ème octets

Attributs de DLC de réception de l'expéditeur (suite)															
Maximum DLSDU size (taille maximale de DLSDU)															
$M\ldots M_R$															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

10) un sous-champ de deux octets ($M\ldots M_R$), spécifiant la DLSDU de taille maximale qui peut être reçue sur la DLC

NOTE Voir les notes suivant c)10).

7.2 Structure et codage des DC-PARAMETERS (paramètres DC)

Une DLPDU DISCONNECT CONNECTION (DC) est utilisée pour déconnecter une DLC existante ou proposée.

Le champ "Disconnect-Connection-parameters" (DC-parameters, c'est-à-dire paramètres "déconnecter la connexion" (paramètres DC)) doit spécifier l'action et la cause de prise en charge de DLC souhaitées, accompagnées de tous les éventuels paramètres opérationnels associés. Ce champ de paramètre est de deux octets, mis dans l'ordre tel que montré du Tableau 47 au Tableau 49.

Tableau 47 – Paramètres DC et paramètres RC: 1er octet

Demande de réponse	reserved (réservé)				Version de protocole de DL		
R	0000				VVV		
7	6	5	4	3	2	1	0

a) un sous-champ d'un seul octet, codé comme dans le Tableau 47, consistant en

1) un sous-champ (R) d'un seul bit, spécifiant si une réponse est demandée (=1) ou non (=0);

   NOTE Ce champ est fourni pour assurer une action correcte indépendamment de tout état du DLCEP destinataire.

2) un sous-champ de quatre bits réservé pour utilisation dans des normes ultérieures et codé comme zéro;

3) un numéro de version de protocole DL à 3 bits (VVV), dont la valeur est spécifiée au début de l'Article 7;

Tableau 48 – Paramètres DC et paramètres RC: 2ème octet

Cause pour la DLPDU							
Cause							
7	6	5	4	3	2	1	0

b) un sous-champ d'un seul octet, spécifiant la cause pour l'action de prise en charge de DLC demandée, basé sur le 8.6.2 de la CEI 61158-3-1, et codé en hexadécimal tel que spécifié dans le Tableau 49. Tous les codes de cause non utilisés dans le Tableau 49 situés dans la plage hexadécimale de 40 à 7F sont réservés à une utilisation dans des normes ultérieures; les autres codes dans la plage 00 à 3F peuvent être utilisés comme souhaité, et peuvent être interprétés comme étant une cause non spécifiée (par la présente norme).

7.3 Structure et codage des RC-PARAMETERS (paramètres RC)

Une DLPDU RESET CONNECTION (RC) est utilisée pour réinitialiser ou déconnecter une DLC existante ou proposée.

Le champ "Reset-Connection-parameters" (RC-parameters, c'est-à-dire paramètres "Réinitialiser la connexion" (paramètres RC)) doit spécifier l'action et la cause de prise en charge de DLC souhaitées, accompagnées de tous les éventuels paramètres opérationnels associés.

Ce champ de paramètre est de quatre octets, mis dans l'ordre tel que montré du Tableau 47 au Tableau 52.

a) Un champ d'un seul octet, codé tel que spécifié en 7.2a).

Tableau 49 – Causes de déconnexion

Code	Cause de déconnexion	Classe de raison	Mapping à la CEI 61158-3-1
00	déconnexion à l'initiative d'un utilisateur – état normal	à l'initiative d'un utilisateur	8.6.2.5b)1)
02	déconnexion à l'initiative d'un utilisateur – état anormal	déconnexion	8.6.2.5b)2)
1E	déconnexion ou rejet de connexion à l'initiative d'un utilisateur – cause non spécifiée	(00..1F)	8.6.2.5b)7)
20	rejet de connexion à l'initiative d'un utilisateur – connexion non autorisée, état permanent	à l'initiative d'un utilisateur	8.6.2.5b)4)
21	rejet de connexion à l'initiative d'un utilisateur – QoS inacceptable, état permanent	connexion	8.6.2.5b)4)
22	rejet de connexion à l'initiative d'un utilisateur – cause non liée à la QoS, état permanent	rejet	8.6.2.5b)5)
24	rejet de connexion à l'initiative d'un utilisateur – état transitoire	(20..3F)	8.6.2.5b)6)
40	déconnexion à l'initiative d'un fournisseur – appariement de DLCEP incorrect, état permanent		8.6.2.5a)1)
41	déconnexion à l'initiative d'un fournisseur – mauvais discriminateur de réutilisation d'adresse de DLCEP éditeur, état permanent		8.6.2.5a)1)
42	déconnexion à l'initiative d'un fournisseur – autre état permanent	à l'initiative d'un fournisseur	8.6.2.5a)1)
43	déconnexion à l'initiative d'un fournisseur – mauvais format ou paramètres de DLPDU, état permanent	déconnexion	8.6.2.5a)1)
44	déconnexion à l'initiative d'un fournisseur – mauvaise taille de DLSDU, état permanent	(40..5F)	8.6.2.5a)1)
45	déconnexion à l'initiative d'un fournisseur – état transitoire		8.6.2.5a)2)
46	déconnexion à l'initiative d'un fournisseur – temporisation		8.6.2.5a)3)
5E	déconnexion ou rejet de connexion à l'initiative d'un fournisseur – cause non spécifiée		8.6.2.5a)12)
60	rejet de connexion à l'initiative d'un fournisseur – adresse de DL(SAP) inconnue		8.6.2.5a)5)
62	rejet de connexion à l'initiative d'un fournisseur – DLSAP injoignable, état permanent	à l'initiative d'un fournisseur	8.6.2.5a)6)
64	rejet de connexion à l'initiative d'un fournisseur – DLSAP injoignable, état transitoire	connexion	8.6.2.5a)7)
65	rejet de connexion à l'initiative d'un fournisseur – état de DLCEP incohérent, état permanent	rejet	8.6.2.5a)8)
66	rejet de connexion à l'initiative d'un fournisseur – QoS indisponible, état permanent	(60..7D)	8.6.2.5a)8)
68	rejet de connexion à l'initiative d'un fournisseur – QoS indisponible, état transitoire		8.6.2.5a)9)
7E	déconnexion ou rejet de connexion, origine inconnue – cause non spécifiée	origine inconnue ou type (7E..7F)	8.6.2.5c)

b) Un sous-champ d'un seul octet, spécifiant la cause pour l'action de prise en charge de DLC demandée, basé sur 8.7.3.5 et 8.7.3.6 de la CEI 61158-3-1, et codé en hexadécimal tel que spécifié dans le Tableau 50. Tous les codes de cause non utilisés dans le Tableau 50 situés dans la plage hexadécimale de C0 à FF sont réservés à une utilisation dans des normes ultérieures; les autres codes dans la plage 80 à BF peuvent être utilisés comme souhaité, et peuvent être interprétés comme étant une cause non spécifiée (par la présente norme).

Tableau 50 – Causes de réinitialisation

Code	Cause pour la réinitialisation	Classe de raison	Mapping à la CEI 61158-3-1
80	réinitialisation à l'initiative d'un utilisateur – resynchronisation après temporisation utilisateur	réinitialisation à l'initiative d'un utilisateur (80..BF)	8.7.3.6b)1)
82	réinitialisation à l'initiative d'un utilisateur – resynchronisation après des incohérences d'état d'utilisateur détectées par un utilisateur		8.7.3.6b)2)
9E	réinitialisation à l'initiative d'un utilisateur – cause non spécifiée		8.7.3.6b)3)
C0	réinitialisation à l'initiative d'un fournisseur – resynchronisation après activation d'un DLCEP établi par une gestion de DL	réinitialisation à l'initiative d'un fournisseur (C0..FD)	8.7.3.6a)1)
C2	réinitialisation à l'initiative d'un fournisseur – resynchronisation après temporisation		8.7.3.6a)2)
C4	réinitialisation à l'initiative d'un fournisseur – resynchronisation après le nombre maximal de demandes ou de tentatives de retransmission;		8.7.3.6a)3)
C6	réinitialisation à l'initiative d'un fournisseur – resynchronisation après erreur de numéro de séquence détectée		8.7.3.6a)4)
C8	réinitialisation à l'initiative d'un fournisseur – resynchronisation après autres incohérences d'état de DLCEP détectées		8.7.3.6a)5)
FC	réinitialisation à l'initiative d'un fournisseur – cause non spécifiée		8.7.3.6a)7)
FE	réinitialisation, origine inconnue – cause non spécifiée	origine inconnue (FE..FF)	8.7.3.6a)1)

Tableau 51 – Paramètres RC: 3ème octet

Numéro de module précédant la prochaine DLSDU devant être envoyée, le cas échéant							
toute valeur			NDS mod 2^5				
7	6	5	4	3	2	1	0

c) Un sous-champ d'un seul octet, spécifiant, dans ses bits d'ordre inférieur, les cinq bits d'ordre inférieur du numéro de séquence NDS précédant le numéro de séquence de la prochaine DLSDU devant être envoyée. Les trois bits d'ordre supérieur de cet octet peuvent avoir n'importe quelle valeur (par exemple, les prochains bits d'ordre supérieur de NDS) et doivent être ignorés à la réception (voir le Tableau 51).

Tableau 52 – Paramètres RC: 4ème octet

Nombre modulus de la dernière DLSDU complète reçue, le cas échéant							
toute valeur			NDR mod 2^5				
7	6	5	4	3	2	1	0

d) Un sous-champ d'un seul octet, spécifiant, dans ses bits d'ordre inférieur, les cinq bits d'ordre inférieur du numéro de séquence NDR de la dernière DLSDU reçue, le cas échéant. Les trois bits d'ordre supérieur de cet octet peuvent avoir n'importe quelle valeur (par exemple, les prochains bits d'ordre supérieur de NDR) et doivent être ignorés à la réception (voir le Tableau 52).

7.4 Structure et codage des SD-Parameters (paramètres SD)

7.4.1 Paramètres SD dans les DLPDU adressées à une adresse de DL(SAP)

Une DLPDU Compel Acknowledgment (CA) adressée à une adresse de DL(SAP) est utilisée comme première phase du service de transfert d'unitdata confirmé à distance pour transférer (ou retransférer) une quantité limitée de données utilisateur transparentes d'un utilisateur de DLS vers un autre utilisateur de DLS et pour faire demander le statut du transfert à la DLE destinataire.

Une DLPDU Compel Data (CD) adressée à une adresse de DL(SAP) est utilisée comme première phase du service d'échange d'unitdata pour demander le transfert (ou le retransfert) immédiat d'une quantité limitée de données utilisateur transparentes d'un utilisateur de DLS vers un autre utilisateur de DLS. Une DLPDU CD contient toujours un champ "paramètres de SD" non vide.

Une DLPDU Exchange Data (ED) adressée à une adresse de DL(SAP) est utilisée comme première phase du service d'échange d'unitdata pour transférer (ou retransférer) une quantité limitée de données utilisateur transparentes d'un utilisateur de DLS vers un autre utilisateur de DLS et pour demander le transfert (ou le retransfert) immédiat d'une quantité limitée de données utilisateur transparentes d'un second utilisateur de DLS vers le premier utilisateur de DLS. Une DLPDU ED contient toujours un champ "paramètres de SD" non vide.

Une DLPDU Data (DT) adressée à une adresse de DL(SAP) est utilisée

1) pour transférer une quantité limitée de données utilisateur transparentes d'un utilisateur de DLS demandeur vers un ou plusieurs autres utilisateurs DLS sans établir ou ultérieurement libérer une DLC; ou
2) comme deuxième phase d'un service d'échange d'unitdata ou de service de transfert d'unitdata confirmé à distance pour acquitter le transfert de ces données sans établir ou ultérieurement libérer une DLC;

Les DLPDU correspondant à 1) contiennent toujours un champ "paramètres SD" vide; les DLPDU correspondant à 2) contiennent toujours un champ "paramètres SD" non vide.

NOTE Ces DLPDU sont également utilisées dans le contexte d'une DLC pour fournir des services DL-Data.

Les différents formats de paramètres SD et leur applicabilité potentielle aux DLPDU CA, CD, DT et ED adressées implicitement ou explicitement à une adresse de DLSAP, montrés dans le Tableau 53, sont:

Tableau 53 – Structure des DLPDU CA, CD, DT et ED en mode sans connexion

DLPDU en mode sans connexion					
Format	**Contrôle de trame**	**Adresse de DLSAP d**	**Adresse de DLSAP s**	**Paramètres**	**Données utilisateur**
1	CA	[HL.]N.S	[HL.]N.S	format R	DLSDU
2	CD	[HL.]N.S	[HL.]N.S	format R	—
3	ED	[HL.]N.S	[HL.]N.S	format R	DLSDU
4	DT 1	[HL.]N.S	[HL.]N.S	format P	DLSDU
5	DT 2	[HL.]N.S	—	format U	o-DLSDU
6	DT 3	PSA implicite	—	format U	—

Format	**Paramètres SD**

	Paramètres autres que de DLC
P	— (null) —
R	ZZ | PP | N(LTI)
U	xxxx | N(LTI)

a) Le format **null** (vide), format P, d'une longueur de zéro octet, est utilisé pour le service de transfert d'unitdata. Il est toujours utilisé dans des DLPDU DT qui ont deux adresses explicites (voir 6.7.1 formats 1L et 1S).

b) Le format **initiator** (initiateur), le format R (voir Tableau 54), d'une longueur d'un octet, est utilisé pour la première phase d'une transaction à deux phases requise pour les services unitdata-exchange (échange d'unitdata) et remotely-confirmed unitdata-transfer (transfert d'unitdata confirmé à distance). Il est toujours utilisé dans des DLPDU CA, CD et ED, qui ont toutes toujours deux adresses explicites (voir en 6.4.1, 6.5.1 et 6.6.1 les formats 1L et 1S).

Tableau 54 – Paramètres SD de format short (court) pour les initiateurs de transaction en mode sans connexion

reserved (réservé)		**Priorité et présence de DLSDU**		**Indice de transaction de l'initiateur**			
ZZ		PP		N(LTI)			
7	6	5	4	3	2	1	0

L'octet comprend

1) un sous-champ de deux bits, réservé pour une utilisation dans des normes ultérieures, qui doit être zéro;

2) un sous-champ "DLSDU-priority-and-presence" (priorité et présence de DLPDU) de deux bits, spécifiant la présence et la priorité de DLL de la DLSDU d'accompagnement (5.2.1.3), codé comme

   00: aucune DLSDU d'accompagnement;

   01: DLSDU d'accompagnement de (haute) priorité URGENT;

   10: DLSDU d'accompagnement de (moyenne) priorité NORMAL;

   11: DLSDU d'accompagnement de (basse) priorité TIME-AVAILABLE;

3) un sous-champ "transaction-index" (indice de transaction) de quatre bits, spécifiant l'indice de transaction assigné au cours de la formation et de la transmission des DLPDU CA, CD ou ED telles que spécifiées en 6.4.3, 6.5.3 et 6.6.3.

c) Le format **responder** (répondeur), le format U (voir Tableau 55), d'une longueur d'un octet, est utilisé pour la seconde phase d'une transaction à deux phases requise pour les services unitdata-exchange (échange d'unitdata) et remotely-confirmed unitdata-transfer (transfert d'unitdata confirmé à distance). Il est toujours utilisé dans les DLPDU DT qui n'ont pas d'adresse source explicite (voir en 6.7.1 les formats 2L, 2S et 4):

Tableau 55 – Paramètres SD de format short (court) pour les répondeurs en mode sans connexion

Status				Indice de transaction de l'initiateur			
xxxx				N(LTI)			
7	6	5	4	3	2	1	0

1) un sous-champ "status" (statut) de quatre bits, codé tel que spécifié dans le Tableau 56;
2) un sous-champ "indice de transaction" de quatre bits, identique à b)3).

Tableau 56 – Statut Reply (réponse) pour les DLPDU DT d'acquittement d'unitdata et de réponse d'échange d'unitdata

Nom court	Définition	Codage hexadécimal	Mapping à la CEI 61158-3-1
OK	succès – aucune donnée de réponse n'est disponible au niveau du répondeur ou le service ne fournit pas cette information	0	9.5.1.2.5a) 9.5.2.2.5a)
OK_U	succès – des données de réponse ayant la priorité URGENT sont disponibles au niveau du répondeur	1	9.5.2.2.5a)
OK_N	succès – des données de réponse ayant la priorité NORMAL sont disponibles au niveau du répondeur	2	9.5.2.2.5a)
OK_TA	succès – des données de réponse ayant la priorité TIME-AVAILABLE sont disponibles au niveau du répondeur	3	9.5.2.2.5a)
RR	échec – limitation de ressources dans le répondeur – aucune donnée de réponse n'est disponible au niveau du répondeur ou le service ne fournit pas cette information (note 1)	4	9.5.1.2.5j) 9.5.2.2.5m)
RR_U	échec – limitation de ressources dans le répondeur – des données de réponse ayant la priorité URGENT sont disponibles au niveau du répondeur (note 1)	5	9.5.2.2.5c)
RR_N	échec – limitation de ressources dans le répondeur – des données de réponse ayant la priorité NORMAL sont disponibles au niveau du répondeur (note 1)	6	9.5.2.2.5c)
RR_TA	échec – limitation de ressources dans le répondeur – des données de réponse ayant la priorité TIME-AVAILABLE sont disponibles au niveau du répondeur (note 1)	7	9.5.2.2.5c)
RF	échec – défaut dans le répondeur	8	9.5.1.2.5k) 9.5.2.2.5e)
RI	échec —le rôle de DL(SAP) du répondeur est incompatible avec cette DLPDU (note 2)	9	9.5.1.2.5g) 9.5.2.2.5h)
RA	échec – réponse restreinte à une différente adresse de DLSAP d'homologue (note 3)	A	9.5.2.2.5g)
—	réservé pour compatibilité avec le statut de réponse de la DLPDU SR – non disponible pour une autre utilisation	B	
BF	échec – défaut dans le pont intermédiaire	C	9.5.2.2.5f)
BR	échec – limitation de ressource dans le pont intermédiaire	D	9.5.1.2.5n) 9.5.2.2.5f)
BOK	réservé pour succès provisoire – le pont intermédiaire transmet la transaction	E	
DR	succès provisoire – réponse différée; le poste final a besoin de plus de temps pour préparer la réponse	F	

NOTE 1 Ce statut peut apparaître lorsqu'un répondeur adressé ne peut pas mettre en tampon la DLSDU reçue (possiblement en raison d'un état de file d'attente pleine). Dans ce cas, le répondeur n'est pas autorisé à envoyer une DLSDU en réponse.

NOTE 2 Ce statut ne peut être généré que lorsque le rôle de DL(SAP) associé à l'adresse de DLSAP de destination reçue est BASIC ou INITIATOR.

NOTE 3 Ce statut ne peut être généré que lorsque le rôle de DL(SAP) associé à l'adresse de DLSAP de destination reçue est CONSTRAINED RESPONDER (répondeur contraint).

7.4.2 Paramètres SD dans des DLPDU adressées à un DLCEP

Une DLPDU COMPEL ACKNOWLEDGMENT (CA) ou EXCHANGE DATA (ED) adressée à un DLCEP est utilisée pour transférer (ou retransférer) une quantité limitée de données utilisateur transparentes d'un utilisateur de DLS vers un ou plusieurs autres utilisateurs de DLS et pour demander le statut du DLCEP distant.

Une DLPDU DATA (CD) ou EXCHANGE DATA (ED) adressée à un DLCEP est utilisée pour demander le transfert (ou le retransfert) immédiat d'une quantité limitée de données utilisateur transparentes d'un utilisateur de DLS vers un ou plusieurs autres utilisateurs de DLS. Une DLPDU CD peut contenir un champ "paramètres SD". une DLPDU CA ou ED contient toujours un champ "paramètres de SD".

A DLPDU DATA (DT) adressée à un DLCEP est utilisée pour une quantité limitée de données utilisateur transparentes d'un utilisateur de DLS demandeur vers un ou plusieurs autres utilisateurs de DLS au sein du contexte d'une DLC; ou pour acquitter le transfert de telles données au sein du contexte d'une DLC. Ces DLPDU contiennent toujours un champ "paramètres SD" (qui peut être vide).

Les quatre types de DLPDU aident à la synchronisation des DLCEP et de leurs utilisateurs de DLS

NOTE Ces DLPDU sont également utilisées hors du contexte d'une DLC pour fournir des services DL-UNITDATA

a) Lorsqu'une DLPDU CD est envoyée par une DLE LAS, ou une telle DLPDU est transmise par un pont, la DLPDU ne doit pas contenir d'adresse de DLSAP source explicite et son champ de paramètre (paramètres SD) doit être vide.
b) Lorsqu'une DLPDU CA, CD, DT ou ED est envoyée à partir d'un DLCEP ou est transmise par un pont, et la DLPDU ne contient pas de données utilisateur, alors le champ paramètre négocié doit être présent et doit comporter deux sous-parties:
   1) la première doit avoir une structure et une longueur fixes [formats A à G] telles que déterminées au cours du processus d'établissement de DLC (voir 7.1c)5) ou 7.1d)5));
   2) la seconde soit
      – doit avoir la structure et la longueur fixes [formats J à M] déterminées lors du processus d'établissement de la DLC (voir 7.1c)5) ou 7.1d)5)), soit
      – doit avoir le format vide, à savoir le format J.
c) Dans tous les autres cas, le champ paramètre négocié doit être présent et doit comporter deux sous-parties ayant chacune une structure et une longueur fixes [formats A à G, concaténés aux formats J à M] telles que déterminées au cours du processus d'établissement de la DLC (voir 7.1c)5) ou 7.1d)5)).
d) Toutes les DLE doivent prendre en charge les formats A, C, F, G et J.

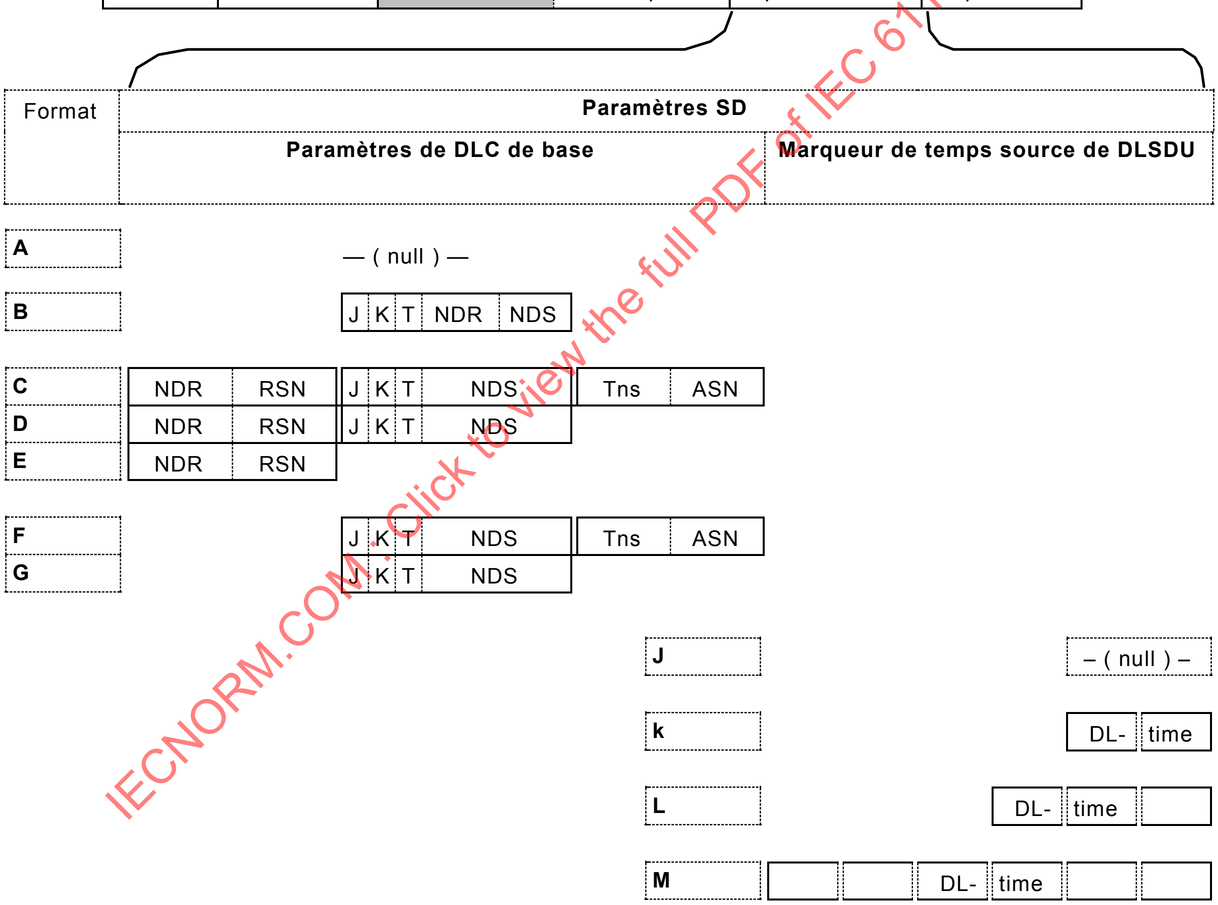
   NOTE La prise en charge des formats A et G est requise pour faciliter la migration à partir de normes nationales existantes.

e) Les DLE qui prennent en charge tant la classe **timeliness** (opportunité) qu'une classe "time -synchronism" (voir 10.6.3 autre que **none** (aucune) doivent prendre en charge les formats K, L et M.

Les divers formats de paramètres SD et leur potentielle applicabilité aux DLPDU CA, CD, DT et ED sont montrés dans le Tableau 57.

Tableau 57 – Structure des DLPDU CA, CD, DT et ED orientées connexion

DLPDU orientées connexion					
Format	**Contrôle de trame**	**adresse de DLCEP d**	**adresse de DLCEP s**	**Paramètres**	**Données utilisateur**
	CA 1	[HL.]N.S	[HL.]N.S	paramètres	o-pDLSDU
2	CA 2	[HL.]N.S	—	paramètres	o-pDLSDU
3	CD 1	[HL.]N.S	[HL.]N.S	paramètres o	—
4	CD 2	[HL.]N.S	—	paramètres o	—
5	ED 1	[HL.]N.S	[HL.]N.S	paramètres	pDLSDU
6	ED 2	[HL.]N.S	—	paramètres	pDLSDU
7	DT 1	[HL.]N.S	[HL.]N.S	paramètres	o-pDLSDU
8	DT 2	[HL.]N.S	—	paramètres	o-pDLSDU
9	DT 3		[HL.]N.S	paramètres	o-pDLSDU
10	DT 4	PSA implicite	—	paramètres	o-pDLSDU
11	DT 5		PDA implicite	paramètres	o-pDLSDU



7.4.2.1 Paramètres acheminant l'état de DLCEP et l'opportunité de DLSDU

Les formats pour la première sous-partie des paramètres SD ont reçu des noms qui reflètent leur utilisation attendue. Ces formats sont:

A) le format **null** (vide), à savoir le format A, ayant une longueur de zéro octet;

B) le format **short** (court), à savoir le format B (voir Tableau 58), ayant une longueur d'un octet, est comme suit:

Tableau 58 – Paramètres SD de format Short (court) pour l'état de DLCEP

Réémettre	Acquitter	Opportunité	Nombre module de DLSDU post-acquittée ou demandée		Nombre module de DLSDU associée ou DLSDU du numéro le plus élevé envoyée		
J	K	T	NDR mod 2^2		NDS mod 2^3		
7	6	5	4	3	2	1	0

1) un sous-champ "retransmission-request (selective-reject)" (demande de retransmission (rejet sélectif)) d'un seul bit, demandant (lorsque J=1) une retransmission de la DLSDU de rang NDR;
2) un sous-champ "acknowledgment" (acquittement) d'un seul bit, reconnaissant (lorsque K=1) que toutes les DLSDU avant celle de rang NDR ont été reçues et rapportées à l'utilisateur de DLS;
3) un sous-champ "timeliness" (opportunité) d'un seul bit, indiquant (lorsque T = 1) que la DLSDU associée avait son origine dans un tampon avec les critères d'opportunité associés, et que ces critères d'opportunité étaient respectés;
4) un sous-champ de deux bits, spécifiant les deux bits d'ordre inférieur du numéro de séquence NDR de la DLSDU demandée et/ou de la DLSDU venant après celle qui est acquittée;
5) un sous-champ de trois bits, spécifiant les trois bits d'ordre inférieur du numéro de séquence NDS de la DLSDU associée (si cette DLPDU contient des données utilisateur) ou de la DLSDU de plus grand numéro qui a été envoyée (si cette DLPDU ne contient pas de données utilisateur);
6) les valeurs implicites pour les champs de 7.4.2.1c) qui sont omises de 7.4.2.1b) sont zéro.

NOTE Les DLSDU envoyés à partir d'un DLCEP ont des numéros de séquence consécutifs qui leur sont assignés, en commençant à un, avant la transmission.

C) Le format **long**, à savoir le format C (voir Tableau 59 à Tableau 61), ayant une longueur de trois octets, est comme suit:

Tableau 59 – Paramètres SD de format Long pour l'état de DLCEP 1^{er} octet

Nombre module de DLSDU post-acquittée ou demandée				Numéro de segment demandé base zéro			
NDR mod 2^4				RSN			
7	6	5	4	3	2	1	0

1) un sous-champ de quatre bits, spécifiant les quatre bits d'ordre inférieur du numéro de séquence NDR de la DLSDU demandée et/ou de la DLSDU venant après celle qui est acquittée;
2) un sous-champ de quatre bits, spécifiant le numéro de segment base zéro du segment de DLSDU demandé, ou zéro lorsque aucun segment n'est demandé;

Tableau 60 – Paramètres SD de format Long pour l'état de DLCEP $2^{ème}$ octet

Réémettre	Acquitter	Opportunité	Nombre module de DLSDU associée ou DLSDU du numéro le plus élevé envoyée				
J	K	T	NDS mod 2^5				
7	6	5	4	3	2	1	0

3) un sous-champ "retransmission-request (selective-reject)" (demande de retransmission (rejet sélectif)) d'un seul bit, demandant (lorsque J=1) une retransmission de la DLSDU de rang NDR;

4) un sous-champ "acknowledgment" (acquittement) d'un seul bit, reconnaissant (lorsque K=1) que toutes les DLSDU avant celle de rang NDR ont été reçues dans leur intégralité et rapportées à l'utilisateur de DLS;

5) un sous-champ "timeliness" (opportunité) d'un seul bit, indiquant (lorsque T = 1) que la DLSDU associée avait son origine dans un tampon avec les critères d'opportunité associés, et que ces critères d'opportunité étaient respectés;

6) un sous-champ de cinq bits, spécifiant les cinq bits d'ordre inférieur du numéro de séquence NDS de la DLSDU associée (si cette DLPDU contient des données utilisateur) ou de la DLSDU de plus grand numéro qui a été envoyée (si cette DLPDU ne contient pas de données utilisateur);

NOTE Si cette DLPDU ne contient aucune donnée utilisateur, toutes les DLSDU demandées précédemment doivent avoir été envoyés dans leur intégralité.

Tableau 61 – Paramètres SD de format Long pour l'état de DLCEP 3ème octet

Nombre total base zéro de segments dans la DLSDU				Numéro de segment associé base zéro			
Tns				ASN			
7	6	5	4	3	2	1	0

7) un sous-champ de quatre bits, spécifiant le nombre total base zéro de segments dans la DLSDU associée, ou zéro lorsque la DLPDU ne contient aucune donnée utilisateur;

8) un sous-champ de quatre bits, spécifiant le numéro de segment base zéro du segment de DLSDU associée ou zéro lorsque la DLPDU ne contient aucune donnée utilisateur;

NOTE Cette représentation a les avantages suivants:

a) Le numéro de segment base zéro, couplé au fait que tous les segments à l'exception du dernier sont de la taille maximum permise par la priorité de la DLPDU, facilite le réassemblage de la DLSDU au cours de la réception.

b) Le nombre total de segments dans la DLSDU réelle peut être utilisé pour allouer un registre de tampon ou de file d'attente, ayant la taille appropriée, à la suite de la première réception de n'importe quel segment de DLSDU.

D) Le format **unsegmented long** (long non segmenté), à savoir le format D, d'une longueur de deux octets, est constitué des deux premiers octets du le format C (voir Tableau 59 et Tableau 60).

E) Le format **subscriber** (abonné), à savoir le format E, d'une longueur d'un seul octet, est constitué du premier octet du le format C (voir Tableau 59), avec des valeurs implicites pour les champs J, K et T du second octet absent. Les valeurs implicites pour les champs J, K, et T selon 7.4.2.1C), qui sont omises de 7.4.2.1E), sont respectivement un (1), zéro (0) et zéro (0).

F) Le format **publisher** (éditeur), à savoir le format F, d'une longueur de deux octets, est constitué des deux derniers octets du le format C (voir Tableau 60 et Tableau 61). Les valeurs pour les champs J et K selon 7.4.2.1C) doivent être zéro (0).

G) Le format **unsegmented publisher** (éditeur non segmenté), à savoir le format G, d'une longueur d'un seul octet, est constitué du second octet du le format C (voir Tableau 60). Les valeurs pour les champs J et K selon 7.4.2.1C) doivent être zéro (0).

Seuls les formats A, C, F et G ont besoin d'être pris en charge; la prise en charge des formats B, D et E est facultative.

7.4.2.2 Paramètres acheminant l'heure de production (time-of-production) de la DLSDU

Les formats pour la seconde sous-partie des paramètres SD sont

J) le format **no-time** (pas d'heure), à savoir le format J, ayant une longueur de zéro octet;

K) le format **2-second** (deux secondes), à savoir le format K, d'une longueur de deux octets, est constitué des troisième et deuxième octets du plus petit ordre du temps de DL de la

DLE émettrice au moment de l'exécution de la "request" de DL-Put, dans cet ordre. Il est adéquat de distinguer la temporisation des événements de moins de 2 s d'écart, et de fournir une résolution potentielle de 2^{-5} ms;

L) le format **8-minute** (huit minutes), à savoir le format L, d'une longueur de trois octets, est constitué des quatrième, troisième et deuxième octets du plus petit ordre 'du temps de DL de la DLE émettrice au moment de l'exécution de la "request" de DL-Put, dans cet ordre. Il est adéquat de distinguer la temporisation des événements de moins de 8 min d'écart, et de fournir une résolution potentielle de 2^{-5} ms;

M) Le format de **full-time** (temps plein), à savoir le format M, d'une longueur de six octets, est constitué des six octets d'ordre supérieur du temps de DL de la DLE émettrice au moment de l'exécution de la demande DL-Put, codé l'octet de poids fort en premier. Il est adéquat de distinguer la temporisation de deux événements quelconques et de fournir une résolution potentielle de 2^{-5} ms.

7.5 Structure et codage des SR-Parameters (paramètres SR)

Il y a deux formats de paramètres SR:

a) Le format **null** (vide), à savoir le format X, d'une longueur de zéro octet, est identique au format P du paramètre SD et est utilisé pour indiquer que le pont d'expédition a été capable de placer en tampon en vue de la transmission une DLPDU CA, CD ou ED reçue, lorsque le jeton de réponse est créé par cette DLPDU reçue. En utilisant ce format, le pont retourne implicitement le statut symbolique "BOK" tel que spécifié dans le Tableau 62.

Tableau 62 – Statut Reply (réponse) pour les DLPDU SR

Nom court	Définition	Codage hexadécimal
RR	échec – limitation de ressource dans le répondeur;	4
RI	échec – la classe DLE du répondeur est incompatible avec cette demande	9
RTR	échec – le transfert de LAS est rejeté	B
BF	échec – défaut dans le pont intermédiaire	C
BR	échec temporaire – limitation de ressource dans le pont intermédiaire	D
BOK	succès provisoire – le pont intermédiaire transmet la transaction	E

NOTE Ce tableau est nécessairement un sous-ensemble du Tableau 56.

b) Le format **error** (erreur), à savoir le format Y, (voir Tableau 63), est basé sur le format U de paramètre SD (voir Tableau 55), et donc sa longueur est d'un seul octet. Le format Y est utilisé

   – pour indiquer qu'un pont pour lequel il convient qu'il ait transmis une DLPDU CA, CD ou ED reçue a été incapable de placer en tampon la DLPDU reçue, lorsque le jeton de réponse a été créé par cette DLPDU reçue (causes BF ou BR);

   – pour rejeter une DLPDU CA, CD ou ED reçue en un DLCEP dans certaines conditions d'erreur;

   – pour rejeter une tentative de transfert du rôle de LAS de la DLE de réponse, lorsque le jeton de réponse a été créé avec une DLPDU TL (causes RR, RI et RNC).

   Les sous-champs du format d'erreur du champ "paramètres SR" sont

   1) un sous-champ "status" (statut) de quatre bits, XXXX, codé tel que spécifié dans le Tableau 62;

      NOTE Les deux seules valeurs possibles pour le statut retourné sont "BF" et "BR".

   2) un sous-champ de quatre bits, ZZZZ, r codé comme zéro et réservé pour utilisation dans des normes ultérieures;

Tableau 63 – Paramètres SR de format Short (court)

Statut				doit être zéro			
xxxx				zzzz			
7	6	5	4	3	2	1	0

7.6 Structure et codage des TD-Parameters (paramètres TD)

Le champ "TD-parameters" (paramètres TD) doit être constitué de quatre sous-champs (voir Tableau 64):

Tableau 64 – Structure des TD-parameters (paramètres TD)

	Paramètres TD et les valeurs de variable émises				
Type de champ	**Liaison produisant le temps de DL**	**Qualité du temps de DL**	**Décalage du temps de DL**	**Temps de DL avant la fin de transmission**	**Réajustement de Temps de DL**
Valeur telle qu'envoyée	V_S(TSL)	V_S(TQ)	V_S(DLTO)	V_S(DLTO) + V_S(LSTO) + $C_S(NT)_1$	P_S(TD) = $C_S(NT)_2$~$C_S(NT)_1$
Nom à la réception	N_S(TSL)	N_S(TQ)	N_S(DLTO)	N_S(DLT)	N_S(DLTA)
Taille de champ	deux octets	un octet	sept octets	sept octets	trois octets

a) L'appellation de liaison de la DLE source de temps de DL – la DLE qui est à l'origine du sens du temps de DL sur la liaison étendue – doit être exprimée en deux octets tels que spécifiés en 4.3.

NOTE Lorsqu'un pont interconnecte deux ou plusieurs liaisons uniques sur une liaison étendue, il s'agit du link-id (identifiant de liaison) de la liaison racine de l'arborescente formée par les ponts qui interconnectent les liaisons sur la liaison étendue. Il s'agit de l'identifiant de liaison locale uniquement pour les DLE de cette liaison racine, ou lorsque aucun pont n'interconnecte la liaison locale avec d'autres liaisons; dans le dernier cas, l'identifiant de liaison peut avoir la valeur zéro.

b) La qualité du temps de DL doit être exprimée en un octet comme suit (voir Tableau 65):

Tableau 65 – Structure et codage pour les mesures de qualité de temps de DL

Classe "time-synchronism" limitante			Nombre de liaisons intermédiaires			Type de source de temps	
TTT			LLL			SS	
7	6	5	4	3	2	1	0

1) un sous-champ de trois bits, TTT, spécifiant de la classe "Time-synchronism" (synchronisme de temps) la moins capable (voir 10.6.3) de toutes les DLE sur le chemin de propagation du temps de DL allant de la source de temps de DL jusqu'à la DLE, inclusivement, codé comme:

000 NONE;
001 1 s;
010 100 ms;
011 10 ms;
100 1 ms;
101 100 µs;
110 10 µs;

111 1 µs;

2) un sous-champ de trois bits, LLL, spécifiant le nombre de liaisons intermédiaires sur le chemin de propagation de temps de DL allant de la DLE source de temps de DL jusqu'à la DLE expéditrice, exprimé comme étant 0 à 7, où

0 indique que le temps de DL provient de la DLE elle-même;

1 indique que le temps de DL provient d'une autre DLE sur la liaison locale;

2 indique que le temps de DL provient d'une DLE située sur une liaison en ayant enlevé un pont,

et ainsi de suite;

3) un sous-champ de deux bits, SS, exprimant la méthode de synchronisation de la DLE source de temps de DL au Temps Universel Coordonné (TUC), la norme de temps mondiale, codé comme

00 un certain temps de nœud de DLE, généré localement et non reçu en provenance d'une source de fournisseur de DLS supplémentaire;

01 le temps local (non TUC) reçu (directement ou indirectement) en provenance d'une source humaine;

10 TUC reçu (directement ou indirectement) en provenance d'une source humaine ou en provenance d'une source électronique peu fiable, ou antérieurement (mais n'étant plus) reçu (directement ou indirectement) en provenance d'une source électronique fiable;

11 TUC continuellement reçu (directement ou indirectement) en provenance d'une source électronique fiable.

NOTE 1 Les sources électroniques fiables comprennent les récepteurs radio pour les transmissions de temps national, les horloges atomiques, et sources similaires qui fournissent une coordination mondiale intrinsèque du temps.

NOTE 2 Les mécanismes de synchronisation d'une DLE qui produit le sens du temps de DL pour la liaison étendue avec une source de temps externe sont nécessairement spécifiques à une mise en œuvre et ne relèvent pas du domaine d'application de la normalisation.

c) Le décalage de temps de DL, V_S(DLTO), de la DLE expéditrice au cours de la formation de la DLPDU doit être exprimé sous la forme d'un nombre entier signé de sept octets codé avec l'octet de poids de fort en premier, où le bit de poids faible représente une granularité de temps de 2^{-13} approximativement telle que montrée dans le Tableau 66.

d) Le temps de DL, $C_S(NT)_1 + V_S(LSTO) + V_S(DLTO)$, de la DLE expéditrice à un moment au cours de la formation de la DLPDU, moins de 1 s avant la transmission, doit être exprimé sous la forme d'un nombre entier non-négatif de sept octets codé avec l'octet de poids fort en premier (voir Tableau 66), où le bit de poids fort est toujours zéro et le bit de poids faible représente une granularité de temps de 2^{-13} ms approximativement.

NOTE La granularité de temps de DL codé n'implique pas que la DLE incrémente le temps de DL à cette granularité, mais plutôt que chaque fois que la DLE incrémente son sens du temps de DL, elle le fait d'une quantité qui amène la fréquence moyenne d'incrémentation à être approximativement de 2^{13} comptes par milliseconde (ce qui correspond à $2^{13} \times 10^3$ comptes par seconde).

Tableau 66 – Poids numérique approché des bits du temps de DL de sept octets

Octet du temps de DL dans l'ordre de transmission	Contenu symbolique	Poids approché du bit d'ordre inférieur de l'octet
1	0YYYYYYY	1,09 année
2	DDDDDDDD	1,55 jour
3	HHHHHMMM	8,74 min
4	MMMSSSSS	2,05 s
5	Smmmmmmm	8,00 ms
6	mmm•µµµµµµ	31,25 µs

Octet du temps de DL dans l'ordre de transmission	Contenu symbolique	Poids approché du bit d'ordre inférieur de l'octet
7	µµµµµnnn	122 ns

e) Le réajustement de temps, $P_S(TD) = C_S(NT)_2 \sim C_S(NT)_1$, est un petit réajustement pour rendre compte des décalages systémiques du sous-champ de temps de DL d'accompagnement d), dus à des considérations bien connues de mise en œuvre au sein du système final réel expéditeur lors de l'envoi d'une DLPDU TD, telles que l'échantillonnage précoce de $C_S(NT)_1$ au cours de la formation et de la transmission de la DLPDU TD, si bien que la somme des valeurs spécifiées par d) et e) est un temps de DL,

$V_S(DLTO) + V_S(LSTO) + C_S(NT)_2$,
auquel la PhIDU END-OF-DATA-AND-ACTIVITY (voir 5.1) sera envoyée à la PhE associée.

Le réajustement du temps doit être exprimé sous la forme d'un nombre entier non signé de trois octets codé l'octet de poids fort le premier (voir Tableau 67), où le bit de poids faible représente une granularité de temps approximativement de 2^{-13} ms.

NOTE 1 Ce champ est inclus afin d'éviter l'exigence que la DLE effectue un réajustement temps réel équivalent sur le temps de DL de sept octets au complet spécifié par c).

NOTE 2 Ce réajustement variera en sens inverse du réel débit instantané de transmission de données de la DLE.

Tableau 67 – Poids numérique approché des bits du temps court de trois octets

Octet du temps court dans l'ordre de transmission	Contenu symbolique	Poids approché du bit d'ordre inférieur de l'octet
1	Smmmmmmm	8,00 ms
2	mmm•µµµµµ	31,25 µs
3	µµµµµnnn	122 ns

NOTE L'objectif du calcul de d) et e) est d'assurer que la somme des sous-champs d) et e) est supérieure au $V_S(NTO)$ de l'expéditeur si bien que

A) si l'expéditeur doit mesurer son temps de propagation aller-retour, $V_S(MD)$ (voir 8.4.1.5), lors de sa communication avec lui-même par l'intermédiaire de la PhE et du support, alors

B) si l'expéditeur doit recevoir sa propre transmission de DLPDU TD par une réception pendant une transmission, alors

C) la somme des sous-champs d) et e) et, telle que calculée par l'expéditeur de la DLPDU TD reçue, est égale à la valeur courante de
$V_S(DLTO) + V_S(LSTO) + C_S(NT)$
au niveau de l'expéditeur à l'instant où la PhE de l'expéditeur rapporte l'achèvement de la réception de la DLPDU TD à partir du support local.

7.7 Structure et codage des RQ-Parameters (paramètres RQ)

Le champ "RQ-parameters" (paramètres RQ) doit être constitué d'un seul sous-champ (voir Tableau 68):

Tableau 68 – Structure des RQ-parameters (paramètres RQ)

Paramètres RQ
C_S(NT)
short-time$_1$
trois octets

a) Le sous-champ paramètre doit être égal aux 24 bits d'ordre inférieur du temps de nœud, C_S(NT), de la DLE expéditrice au moment de la formation de la DLPDU RQ juste avant (ou pendant) la transmission, représentant les trois octets d'ordre inférieur du temps de nœud courant du fournisseur de DLS en unités de 2^{-13} ms.

NOTE Le contenu de ce sous-champ reflète le temps relatif auquel la DLPDU RQ est émise.

Si la classe de synchronisation de temps de la DLE expéditrice (voir 10.6.3) est NONE et la DLE ne maintient même pas un C(NT) estimé, elle peut alors utiliser n'importe quelle valeur pour ce sous-champ.

Le Tableau 67 montre l'ordre de transmission et le poids numérique approché des octets de ce sous-champ. Il est codé sous la forme d'un nombre entier non signé de trois octets envoyé l'octet de poids fort en premier.

7.8 Structure et codage des RR-Parameters (paramètres RR)

Le champ "RR-parameters" (paramètres RR) doit être constitué de quatre sous-champs (voir Tableau 69):

Tableau 69 – Structure des RR-parameters (paramètres RR)

Paramètres RR			
N_S(TQ)$_{7\text{-}5}$	N_1(NT)	N_2(NT)	C_S(NT)
qualité de mesure	short-time$_1$	short-time$_2$	short-time$_3$
un octet	trois octets	trois octets	trois octets

a) Le premier sous-champ paramètre doit être codé tel que spécifié dans le Tableau 70, et doit spécifier

1) la classe "time-synchronism" de la DLE de réponse, TTT, codée comme en 7.6b)1);

2) si, oui ou non, une DLE de la classe "time-synchronism" NONE maintient un C(NT) estimé, E, codé

i) comme étant un Boolean (0=FALSE, 1=TRUE) lorsque TTT a la valeur 000;

ii) comme étant 1 (TRUE) lorsque TTT a une valeur autre que 000;

3) un sous-champ de quatre bits, codé comme étant zéro.

Tableau 70 – Structure et codage pour les mesures de qualité de temps de RR

Classe "time-synchronism" limitante			**La DLE maintient un C(NT) estimé**	**reserved (réservé)**			
TTT			E	0000			
7	6	5	4	3	2	1	0

b) Les deuxième et troisième sous-champs "paramètre" doivent être respectivement égaux aux sous-champs paramètre reçus et aboutés localement de la DLPDU RQ reçue.

c) Le quatrième sous-champ paramètre doit être égal aux 24 bits d'ordre inférieur du temps de nœud, C_S(NT), de la DLE répondeuse au moment de la formation de la DLPDU RR juste avant (ou pendant) la transmission, représentant les trois octets d'ordre inférieur du temps de nœud courant du fournisseur de DLS en unités de 2^{-13} ms.

NOTE Le contenu de ce sous-champ reflète le temps relatif auquel la DLPDU RR est émise. Les calculs de temps de propagation aller-retour résultant sont exacts, indépendants vis-à-vis de tous les éventuels retards internes aux DLE, à condition que la somme des temps de propagation entre les deux DLE soit inférieure à 2 s.

Si la classe de synchronisation de temps de la DLE expéditrice (voir 10.6.3) est NONE et la DLE ne maintient même pas un C(NT) estimé, elle doit coder ce sous-champ comme étant zéro.

Le Tableau 67 montre l'ordre de transmission et le poids numérique approché des octets de chacun des trois derniers sous-champs. Chacun de ces sous-champs est codé sous la forme d'un nombre entier non signé de trois octets avec l'octet de poids fort envoyé en premier.

7.9 Structure et codage des PN-Parameters (paramètres PN)

Le champ "paramètres PN" spécifie les valeurs courantes de ces paramètres de DLE et de PhE nécessaires pour qu’une DLE de réception se configure elle-même et configure sa PhE associée, afin qu'elles puissent répondre à une DLPDU PN ultérieure. Une fois réglées, les valeurs de ces paramètres ne peuvent pas être changées tant que la DLE de réception reste ONLINE, autrement que par la réception d'une autre DLPDU PN telle que spécifiée en 10.1.3.

NOTE Théoriquement, la DLE LAS courante pourrait changer les paramètres de la liaison en envoyant une DLPDU PN à chacune des autres DLE sur la liaison locale, forçant chacune d'elle à être OFFLINE, après quoi elles apprendraient les nouveaux paramètres de liaison dans le processus les faisant redevenir ONLINE. Mais le présent protocole DL ne spécifie pas de moyen de demander cette action de LAS.

Le champ "PN-parameters" (paramètres PN) doit être constitué de quatre sous-champs (voir Tableau 71 à Tableau 76):

Tableau 71 – Structure des PN-parameters (paramètres PN)

Paramètres PN et les valeurs de variable émises						
Dissymétrie maximale de signal entre voies de PhL	**Version**	**Unités d'extension de trous post-transmission de PhL**	**Unités d'extension de préambule de PhL**	**Intervalle de temps**	**Retard maximal de réponse**	**Retard minimal entre des DLPDU**
V(PhIS)	0VVV	V(PhGE)	V(PhPE)	V(ST)	V(MRD)	V(MID)
un quartet	un quartet	un quartet	un quartet	deux octets	un octet	un octet

Le champ "paramètres PN", qui se partitionne comme montré dans le Tableau 71 doit spécifier les paramètres nécessaires pour répondre à la DLPDU PN:

a) Les deux premiers octets, mis dans l'ordre tel que montré dans le Tableau 72 et le Tableau 73, spécifient le numéro de version du protocole de DL et les paramètres PhL de la liaison locale, définis dans le Tableau 4 de la CEI 61158-2, requis pour générer une réponse à une DLPDU PN reçue:

Tableau 72 – Paramètres PN: 1er octet

Dissymétrie maximale de signal entre voies de PhL				**Zéro**	**Version**		
V(PhIS)				0	VVV		
7	6	5	4	3	2	1	0

1) un sous-champ de quatre bits qui spécifie la dissymétrie maximale de signal entre voies de PhL requise, définie en 6.2.2.2 de la CEI 61158-2, pour les PhE de la liaison locale;

2) un sous-champ de 1 bit, codé comme étant zéro.
3) un numéro de version de protocole DL à 3 bits (VVV), dont la valeur est spécifiée au début de l'Article 7;

Tableau 73 – Paramètres PN: 2ème octet

Unités d'extension de trou post-transmission de PhL				Unités d'extension de préambule de PhL			
V(PhGE)				V(PhPE)			
7	6	5	4	3	2	1	0

4) un sous-champ de quatre bits qui spécifie le nombre requis d'unités d'extension de trou post-transmission de PhL, définies en 6.2.2.2 de la CEI 61158-2, pour les PhE de la liaison locale;

NOTE Bien que cette mesure soit appelée “post-transmission-gap-extension” (extension de trou post-transmission) dans la CEI 61158-2, il s'agit d'une mesure de la quantité de non-transmission observable requise entre deux transmissions quelconques sur le support local, telle qu'observée à partir de n'importe point de ce support, et ce, que les deux transmissions émanent de la même PhE source ou de deux PhE sources différentes. Elle pourrait donc tout aussi correctement être appelée "pre-transmission-gap-extension" (extension de trou avant transmission), ou même, plus correctement, "inter-transmission-gap-extension" (extension de trou entre transmissions)

5) un sous-champ de quatre bits qui spécifie le nombre requis d'unités d'extension de préambule de PhL, définies en 6.2.2.2 de la CEI 61158-2, pour les PhE de la liaison locale;

b) Les quatre octets suivants, mis dans l'ordre tel que montré Tableau 74 à Tableau 76, spécifient les paramètres DLL pour la liaison locale requis pour la génération d'une réponse à une DLPDU PN reçue:

Tableau 74 – Paramètres PN: 3ème et 4ème octets

Intervalle de temps															
V(ST)															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Tableau 75 – Paramètres PN: 5ème octet

Retard maximal de réponse							
V(MRD)							
7	6	5	4	3	2	1	0

Tableau 76 – Paramètres PN: 6ème octet

Retard minimal entre des DLPDU							
V(MID)							
7	6	5	4	3	2	1	0

1) Le cinquième sous-champ "paramètre" doit acheminer, dans deux octets émis l'octet de poids fort en premier, la valeur courante de V(ST), intervalle de temps, défini en 4.7.1.1.
2) Le sixième sous-champ "paramètre" doit acheminer, dans un seul octet, la valeur courante de V(MRD), retard maximal de réponse, défini en 4.7.1.3.
3) Le septième sous-champ "paramètre" doit acheminer, dans un seul octet, la valeur courante de V(MID), retard minimal entre des DLPDU, défini en 4.7.1.12.

7.10 Structure et codage des DD-Parameters (paramètres DD)

Le champ "DD-parameters" (Delegation-Duration parameters) (paramètres de durée de délégation) doit être composé d'un seul sous-champ (voir Tableau 77):

Tableau 77 – Structure des DD-parameters (paramètres DD)

Paramètres DD
Durée déléguée ou demandée
V(RD)
deux octets

a) Le sous-champ doit spécifier la durée pendant laquelle le jeton est délégué ou pour laquelle la délégation est demandée. Il est mesuré en durées d'octet, et est codé sous la forme d'un nombre entier non signé de deux octets envoyé l'octet poids fort en premier. Sa plage s'étend de 0 à 65 000.

NOTE La valeur pratique minimale pour utiliser cette durée pour une transmission est 16 environ. La valeur 0 est utilisée seulement au cours de la maintenance de la liste active V(LL) de la liaison locale.

8 Éléments de procédure du service de DL

NOTE 1 Les éléments de procédure dans le présent article correspondent généralement au sous-niveau de DLL le plus élevé, tel que spécifié en 4.1.3.

Dans tout l'Article 8, chaque fois qu'une action conditionnelle est spécifiée et la condition habilitante spécifiée ne se produit pas, l'action correspondante ne se produit pas non plus. L'effet net produit est que les événements qui ne satisfont à aucune des conditions habilitantes spécifiées dans une procédure de service n'auront aucune conséquence en ce qui concerne cette procédure de service spécifique.

Dans tout l'Article 8, la valeur (V(NRC)+1) est utilisée comme une indication indépendante vis-à-vis d'une liaison du nombre maximum de fois pour accomplir une procédure – une seule fois initialement, plus V(NRC) répétitions – avant de conclure qu'une situation d'erreur irrécupérable existe et qu'une action alternative plus drastique est requise. Lorsque la DLE peut établir, à partir des adresses de DL(SAP) ou adresses de DLCEP impliquées, que toutes les DLE impliquées dans une DLC sont locales à une seule liaison, la DLE peut alors substituer en toute cohérence la valeur de la liaison dépendant de la liaison (V(MRC)+1) à la valeur de la liaison indépendante vis-à-vis de la liaison (V(NRC)+1).

NOTE 2 Dans tout l'Article 8, une DLE reçoit une occasion d'émettre, soit par la réception d'une occasion de réponse immédiate appropriée, soit par la réception d'un un jeton approprié.

8.1 Fonctionnement de services de gestion d'adresses de DL(SAP), de tampons et de files d'attente

Les services de gestion d'adresses de DL(SAP), de tampons et de files d'attente sont les services "create" et "delete" "buffer or queue" (créer et supprimer un tampon ou une file d'attente), les services "bind" et " unbind" DL(SAP)-Address (lier et délier une adresse de DL(SAP), le service "put buffer' (placer un tampon" et le service "get buffer or queue" (récupérer un tampon ou une file d'attente".

8.1.1 Réception d'une primitive "request" de DL-CREATE

Lorsque la DLE reçoit une demande de DL-CREATE, elle doit

a) allouer un tampon de la taille de DLSDU spécifiée, ou une file d'attente de la taille de DLSDU et le nombre maximal d'entrées maximum et l'initialiser comme étant vide;

NOTE Les techniques statistiques d'allocation utilisant une zone partagée de stockage à usages multiples sont admissibles.

b) assigner à ce tampon ou à cette file d'attente un identifiant de DL de tampon ou de file d'attente, et tout identifiant d'utilisateur de DLS fourni;

c) retourner cet identifiant à l'utilisateur de DLS, accompagné d'un statut «succès».

En variante, un statut d'échec approprié doit être retourné à l'utilisateur de DLS.

8.1.2 Réception d'une primitive "request" de DL-DELETE

Lorsque la DLE reçoit une demande de DL-DELETE, si le tampon ou la file d'attente spécifié(e)

a) avait été créé(e) par l'utilisateur de DLS et non par une action de la gestion de DL;

b) n'est pas actuellement lié(e) à un DLCEP ou à une adresse de DL(SAP)

la DLE doit supprimer le tampon ou la file d'attente spécifié(e) et retourner un statut de "succès".

Autrement, la DLE doit retourner un statut d'échec approprié.

8.1.3 Réception d'une primitive "request" de DL-BIND

Lorsque la DLE reçoit une demande de DL-BIND, elle doit vérifier

a) que l'adresse de DL(SAP) spécifiée est une adresse de DL de groupe qui n'est pas présentement utilisée par l'utilisateur de DLS demandeur ou une adresse de DLSAP qui n'est pas présentement utilisée par la DLE;

b) que les composantes de liaison et de nœud spécifiées de l'adresse de DL(SAP) sont compatibles avec l'une ou les plusieurs composantes d'adresse de liaison et l'une ou les plusieurs composantes d'adresse de nœud assignées à cette DLE (voir 4.3);

c) que toute liaison explicite spécifiée à des files d'attente de DL ou à des tampons de DL est valide et n'est pas en conflit avec d'autres liaisons existantes à ces files d'attente ou tampons;

d) que tout attribut de QoS statique ou dynamique spécifié est valide et autorisé par la gestion de DL locale.

Un échec doit être rapporté à l'utilisateur de DLS avec un statut approprié Autrement

1) l'adresse de DL(SAP) doit être associée à l'utilisateur de DLS demandeur et à tout identifiant d'utilisateur de DLS fourni;

2) toutes files d'attente et tous les tampons spécifiés explicitement doivent être liés comme spécifié à l'adresse de DL(SAP);

3) les attributs de QoS statiques et dynamiques doivent être liés à l'adresse de DLSAP;

4) l'adresse de DL(SAP) spécifiée doit être activée pour la réception;

5) un identifiant d'adresse de DL(SAP) doit être assigné à l'adresse de DL(SAP) et retourné à l'utilisateur de DLS, accompagné d'un statut de "succès";

6) Si soit

   i) la DLE est une DLE facteur d'utilisation fractionnaire (FDC), soit

   ii) l'adresse de DL(SAP) a une composante "appellation de liaison' (voir 4.3.2.1) dont la valeur spécifie une adresse de DL non locale plate (entre 0001_{16} et $0FFF_{16}$, inclusivement) et la variable V(TL) a une valeur autre que zéro,

   alors la DLE doit envoyer une SPDU de rapport d'adresse de DL telle que spécifiée en 9.3.6.3, avec une cause appropriée de la SPDU de rapport et avec un identifiant de demande de zéro, aux fonctions de prise en charge de DL de tous les ponts sur la liaison locale (voir 4.3.3.2).

8.1.4 Réception d'une primitive "request" de DL-UNBIND

Lorsque la DLE reçoit une demande de DL-UNBIND,

a) si l'adresse de DL(SAP) spécifiée n'est présentement pas liée à l'utilisateur de DLS demandeur, ou si l'adresse de DL(SAP) spécifiée était liée par une action de la gestion de L, la DLE doit ignorer la primitive "request";

b) autrement, la DLE doit

   – déconnecter toute DLCEP associée à cette adresse de DL(SAP);

   – confirmer avec un statut approprié d'erreur toute demande de service non confirmé orienté connexion ou en mode sans connexion qui est en cours à cette adresse de DL(SAP) ou ses DLCEP;

   – délier toutes les files d'attente et tous les tampons de l'adresse de DL spécifiée;

   – dissocier l'adresse de DL spécifiée de l'utilisateur de DLS demandeur.

   Si la DLE n'a pas d'autres utilisateurs de DLS associés à cette adresse de DL (ce qui est toujours le cas pour les adresses de DLSAP), elle doit désactiver la réception de cette adresse de DL, auquel cas si soit

   1) la DLE est une DLE facteur d'utilisation fractionnaire (FDC), soit

   2) l'adresse de DL(SAP) a une composante "appellation de liaison' (voir 4.3.2.1) dont la valeur spécifie une adresse de DL non locale plate (entre 0001_{16} et $0FFF_{16}$, inclusivement) et la variable V(TL) a une valeur autre que zéro,

   alors la DLE doit envoyer une SPDU de rapport d'adresse de DL telle que spécifiée en 9.3.6.3, avec une cause appropriée de la SPDU de rapport et avec un identifiant de demande de zéro, aux fonctions de prise en charge de DL de tous les ponts sur la liaison locale (voir 4.3.3.2).

8.1.5 Réception d'une primitive "request" de DL-PUT

Lorsque la DLE reçoit une demande de DL-PUT pour un tampon

a) qui est associé à l'utilisateur de DLS demandeur;

b) qui n'est pas lié comme un tampon récepteur

   1) à un DLCEP homologue ou abonné, ou

   2) à une adresse de DLSAP dont le rôle DL(SAP) est INITIATOR ou CONSTRAINED RESPONDER ou UNCONSTRAINED RESPONDER;

alors

c) si une DLSDU est présentée et la taille de la DLSDU est inférieure ou égale à la taille de tampon, la DLE doit

   1) rendre le contenu du temps égal à la DLSDU présentée;

   2) indiquer pour chaque DLCEP qui a une liaison d'expédition au tampon que le tampon contient une nouvelle DLSDU;

   3) mettre le statut d'opportunité d'écriture dans un tampon, V_B(TS) (voir 4.7.4.21), à l'opportunité spécifiée par l'utilisateur de DLS, ou à FALSE si l'utilisateur n'a pas spécifié d'opportunité;

   4) si le statut d'opportunité d'écrire dans un tampon est TRUE, mettre au temps de DL courant la variable "time-of-last-buffer-write" (temps de dernière écriture dans le tampon), V_B(TW) (voir 4.7.4.19) associée au tampon;

   5) si le statut d'opportunité d'écrire dans un tampon est TRUE, mettre au temps de DL courant l'heure de production (time-of-production), V_B(TP) (voir 4.7.4.20) associée au tampon;

   6) retourner un statut de "succès";

d) sinon si c) ne s'applique pas et aucune DLSDU n'est présentée (c'est-à-dire que le tampon est mis à vide) et le tampon est lié à une adresse de DLSAP, alors la DLE doit mettre à vide le tampon et retourner un statut de "succès";

e) sinon si ni c) ni d) ne s'appliquent, la DLE doit retourner un statut approprié d'échec.

Tout accès en cours au contenu d'un tampon ou à ses informations d'opportunité associées, qui est incomplet au moment d'une demande de DL-PUT, ne doit pas être affecté par la demande de DL-PUT.

NOTE Cette contrainte assure que chacun des accès à un tampon est logiquement atomique.

8.1.6 Réception d'une primitive "request" de DL-GET

a) Si la DLE reçoit une demande de DL-GET pour un tampon qui est associé à l'utilisateur de DLS demandeur, alors

   1) s'il a été écrit dans le tampon par un DLCEP qui a spécifié une opportunité RESIDENCE ou une opportunité UPDATE ou une opportunité SYNCHRONIZED ou une opportunité TRANSPARENT, alors la DLE doit évaluer les critères d'opportunité associés tels que spécifiés en 8.1.7, en utilisant le temps de DL courant comme temps de lecture de tampon, et doit retourner le résultat comme étant l'attribut d'opportunité de DLE locale de la demande;

   2) autrement, si 1) ne s'applique pas, la DLE doit retourner la valeur FALSE comme étant l'attribut d'opportunité de DLE locale pour la demande;

   3) la DLE doit retourner le statut d'opportunité, $V_B(TS)$ (voir 4.7.4.21), associé à l'écriture dans le tampon comme étant l'attribut "sender-and-remote-DLE-timeliness" (c'est-à-dire: opportunité de DLE expéditrice et distante) (voir 8.2.2.5.3) de la demande.

   NOTE Cet attribut "sender-and-remote timeliness" a toujours la valeur FALSE lorsque le tampon a été écrit par une instance de service d'échange d'unitdata.

   4) Si l'attribut "sender-and-remote-DLE-timeliness" associé au tampon est TRUE et le tampon a été écrit par un DLCEP qui fournit une "DL-time-of-production" (heure de production de DL), alors la DLE doit retourner l'heure de production de DL associée à ce tampon, $V_B(TP)$ (voir 4.7.4.20).

   5) La DLE doit retourner le contenu courant du tampon avec un statut de "succès" si le tampon n'est pas vide et un statut "échec possible – tampon vide" si le tampon est vide.

   6) Si le tampon est un tampon non rétenteur (BUFFER-NR), le tampon doit être mis à vide.

   7) La lecture d'un tampon et de ses informations d'opportunité associées doit être logiquement atomique en ce qui concerne l'écriture dans le tampon.

   NOTE Lorsqu'une mise en œuvre fournit un accès à un tampon pour une période de temps étendue au cours de la lecture ou de l'écriture dans le tampon, alors dans le cas le plus défavorable, cette restriction d'atomicité exige que des copies séparées du contenu du tampon et des informations d'opportunité soient fournies pour chaque lecteur et pour l'unique rédacteur, en plus du tampon réel avec son contenu et son opportunité. Alors, chaque lecteur peut être au milieu d'un accès étendu dans une différente époque du contenu du tampon, et le rédacteur peut être en train d'écrire un tampon provisoire à la réception, qui ne deviendra le tampon courant que si une erreur de Ph ou une erreur de FCS n'est pas détectée avant la fin du processus de réception.

b) Si la DLE reçoit une demande DL-GET pour une file d'attente spécifiée par un utilisateur de DLS qui est

   – associée à l'utilisateur de DLS demandeur;

   – non liée comme file d'attente d'envoi soit à un DLCEP, soit à une adresse de DLSAP;

   – non vide

   alors, la DLE doit

      1) retourner la prochaine DLSDU contenue dans la file d'attente, accompagnée de l'identifiant de DLC appelée associée à cette DLSDU, ou la priorité de DLL et les adresses de DL(SAP) ou identifiants d'adresse DL(SAP) appelante et appelée associé(e)s à cette DLSDU;

2) retourner les attributs d'opportunité de DLE locale et distante ayant la valeur FALSE;
3) retirer cette DLSDU de la file d'attente;
4) retourner un statut de "succès".

Autrement, la DLE doit retourner un statut d'échec approprié.

8.1.7 Calcul de l'opportunité de DL

La DLE doit calculer l'opportunité de DL (DL-timeliness) intrinsèque pour le tampon en se basant sur le type d'accès au tampon – lecture ou écriture – et le type correspondant d'opportunité de DL qui a été spécifié sur la primitive "request" ou "response" de DL-CONNECT locale correspondante, comme suit, où $P_C(NP.\Delta T)$ est la taille de fenêtre temporelle appropriée spécifiée dans cette primitive "request" ou "response";

a) **RESIDENCE**

DL-timeliness ≡ TRUE lorsque $0 \le$ (temps de DL courant – $V_B(TW)$)
et (temps de DL courant – $V_B(TW)$) $\le P_C(NP.\Delta T)$;
DL-timeliness ≡ FALSE autrement.

b) **UPDATE**

DL-timeliness ≡ TRUE lorsque $0 \le (V_B(TW) - V_C(TNA))$
et $(V_B(TW) - V_C(TNA)) \le P_C(NP.\Delta T)$;
DL-timeliness ≡ FALSE autrement.

c) **SYNCHRONIZED**

DL-timeliness ≡ TRUE lorsque $0 \le (V_B(TW) - V_C(TNA))$
et $(V_B(TW) - V_C(TNA)) \le$ (temps de DL courant – $V_C(TNA)$)
et (temps de DL courant – $V_C(TNA)$) $\le P_C(NP.\Delta T)$;
DL-timeliness ≡ FALSE autrement.

d) **TRANSPARENT**

DL-timeliness ≡ TRUE.

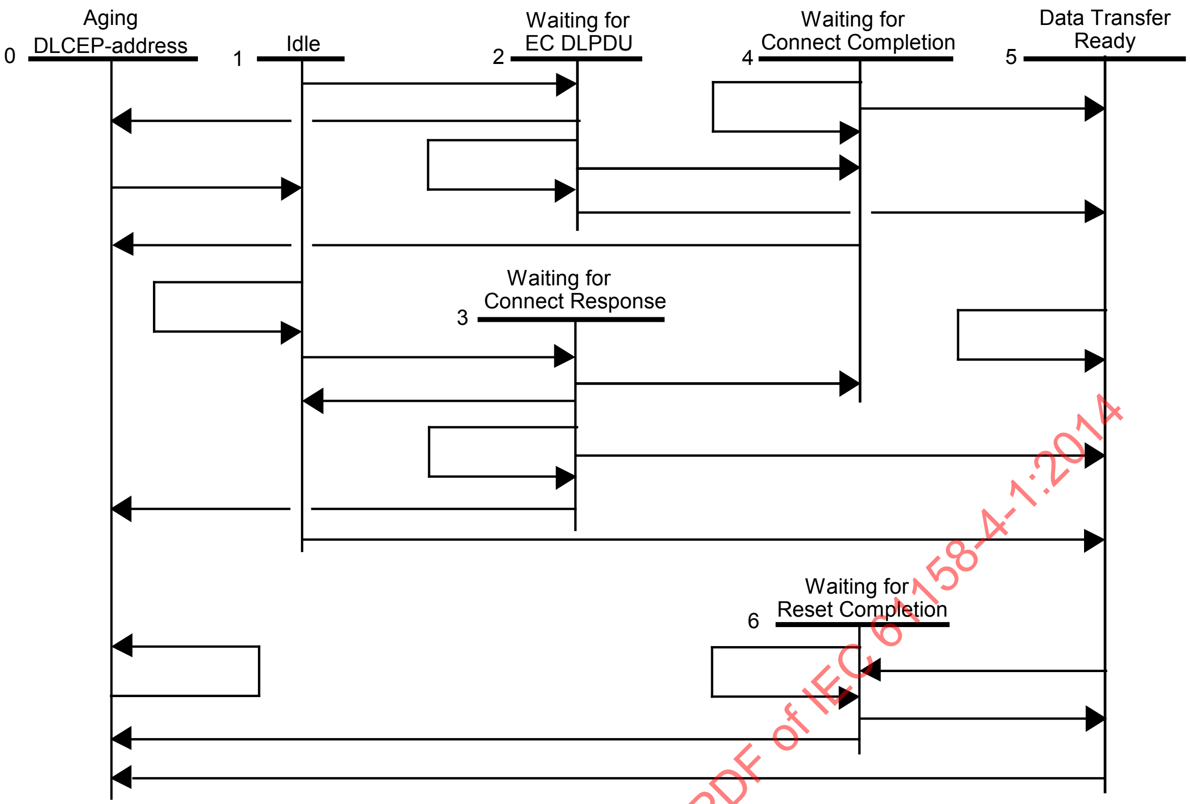
e) **NONE**

DL-timeliness ≡ FALSE.

8.2 Fonctionnement des services en mode connexion

Les services en mode connexion sont les services "DLCEP establishment" (établissement de DLCEP) et "DLCEP release" (libération de DLCEP), le service "DLC data transfer" (transfert de données de DLC) et "DLCEP reset" (réinitialisation de DLC) et le service "DLC subscriber query" (interrogation d'abonné de DLC).

La CEI 61158-3-1, Figure 16, montre un diagramme de transition d'états conceptuel pour les séquences de primitives de services de DLC en un DLCEP. Un diagramme de transition d'états correspondant pour ce protocole de DL est montré à la Figure 11 et à l'Annexe B, où des statuts similaires ont des numéros d'état identiques.



Légende

Anglais	Français
Aging DLCEP-address	Adresse-de DCLEP vieillissante
Idle	Repos
Waiting for EC DLPDU	Attendant une DLPDU EC
Waiting for Connect Completion	Attendant un achèvement de connexion
Data Transfer Ready	Prêt au transfert de donnée
Waiting for Connect Response	Attendant une réponse de connexion
Waiting for Reset Completion	Attendant un achèvement de réinitialisation

Figure 11 – Diagramme de transition d'états pour un DLCEP

8.2.1 Fonctionnement des services d'établissement de DLCEP et de libération de DLCEP

Les primitives du d'établissement de DLCEP sont "request", "indication", "response" et "confirm" de .DL-CONNECT et "indication" de DL-CONNECTION-ESTABLISHED. Les primitives du service de libération DLCEP sont "request" et "indication" de DL-DISCONNECT. L'utilisateur de DLS local associé, ou l'utilisateur de DLS homologue ou éditeur distant du DLCEP ou le fournisseur de DLS, peut invoquer un service de libération de DLCEP à tout moment après que l'établissement du DLCEP a commencé.

Au cours de l'établissement et de la libération du DLCEP, si le DLCEP est un DLCEP homologue ou éditeur et l'adresse de DLCEP utilisée par le DLCEP n'est pas envoyée comme une adresse de DL source dans une DLPDU DC ou EC, et si soit

a) la DLE est une DLE facteur d'utilisation fractionnaire (FDC), soit

b) l'adresse de DLCEP a une composante "appellation de liaison' (voir 4.3.2.1) dont la valeur spécifie une adresse de DL non locale plate (entre 0001_{16} et $0FFF_{16}$, inclusivement) et la variable V(TL) a une valeur autre que zéro, alors la DLE doit envoyer une SPDU de rapport d'adresse de DL telle que spécifiée en 9.3.6.3, incluant l'adresse de DLCEP spécifiée, avec une cause appropriée de la SPDU de rapport et avec un identifiant de demande de zéro, aux fonctions de prise en charge de DL de tous les ponts sur la liaison locale (voir 4.3.3.2).

8.2.1.1 Règles de négociation de DLC

Les aspects visibles à un utilisateur de DLS des règles de négociation de DLC sont spécifiés dans la CEI 61158-3-1. Des règles supplémentaires de négociation qui n'ont pas d'impact sur les aspects visibles à un utilisateur de DLS relatifs à la DLC sont autorisées à être spécifiées en 7.1 de la présente norme. En cas de conflit évident, les règles spécifiées dans ce paragraphe prévalent sur toutes celles qui peuvent être spécifiées en 7.1, qui à leur tour prévalent sur celles qui sont spécifiées dans la CEI 61158-3-1.

NOTE Lorsqu'un utilisateur de DLS abonné qui tente de se joindre à une DLC existante demande des attributs de DLC qui ne sont pas fournis par les attributs de cette DLC existante, l'utilisateur de DLS éditeur peut soit

A) indiquer son incapacité de fournir les attributs demandés en répondant par une primitive "request" de DL-DISCONNECT à l'abonné proposé,

B) créer une nouvelle DLC avec les attributs désirés et connecter l'abonné à cette DLC,

C) soit
modifier la DLC existante afin d'inclure les caractéristiques demandées, lorsque cela est permis par la présente norme.

a) Si l'éditeur, ou l'un des homologues, d'une DLC spécifie un attribut DLPDU-authentication (authentification DLPDU) MAXIMAL, alors

   NOTE L'attribut "DLPDU-authentication" ayant la valeur MAXIMAL est principalement fourni pour un usage dans des systèmes de sécurité. Pour cette raison, elle maximise la quantité des informations d'état échangées dans chaque DLPDU envoyée sur la DLC et empêche l'échange bidirectionnel de données utilisateur dans une seule transaction, l'exécution de programme centralisée et autres activités dans lesquelles plusieurs DLE ont besoin d'avoir des informations d'état cohérentes.

   1) chaque DLPDU envoyée à partir de chaque DLCEP de la DLC doit contenir le nombre maximal autorisé d'adresses explicites;
   2) les paramètres EC de chaque DLPDU EC doivent être contraints comme suit:
      i) le sous-champ "address-size" (taille d'adresse (SS)) doit spécifier LONG;
      ii) le sous-champ "DLPDU-authentication" (authentification de DLPDU (XX)) doit spécifier MAXIMAL;
      iii) le sous-champ "residual-activity" (activité résiduelle (A)) doit spécifier TRUE dans le sens éditeur vers abonné, ou dans tous les sens peer-to-peer d'envoi, du transfert de données;

         NOTE L'activité résiduelle n'est pas interprétable dans le sens abonnés vers éditeur direction.

      iv) les deux sous-champs d'échange bidirectionnel de données (E) doivent spécifier FALSE;
      v) ; le format B de paramètre SD (sous-champ FFF), et les formats de marqueur temporel K et L (sous-champ HH) ne doivent être demandés ou utilisés dans aucun sens sur la DLC;
   3) Si une demande de DL-SCHEDULE-SEQUENCE spécifie une séquence dans laquelle un ou plusieurs éléments de séquence spécifient le DLCEP local ou distant de cette DLC, cette séquence peut être programmée localement ou centralement, mais doit être exécutée localement (voir 8.4.3.1).

b) Si a) ne s'applique pas, alors
   i) si l'éditeur, ou l'un des homologues, d'une DLC spécifie un attribut d'authentification DLPDU SOURCE, alors le sous-champ d'authentification DLPDU (XX) dans les paramètres EC doit spécifier SOURCE, et chaque DLPDU envoyée à partir de chaque DLCEP sur la DLC doit contenir le nombre maximal autorisé d'adresses explicites autorisé;
   ii) si un abonné d'une DLC spécifie un attribut d'authentification de DLPDU MAXIMAL dans une primitive "request" de DL-CONNECT, le sous-champ d'authentification de DLPDU

(XX) dans la DLPDU de paramètres EC qui en résulte doit spécifier MAXIMAL. Si la DLC demandée était déjà établie, alors

A) si cette DLC n'a pas été établie avec un attribut d'authentification DLPDU MAXIMAL, alors la DLE d'édition doit rejeter la demande d'établissement de connexion issue de cet abonné;

B) autrement, lorsque A) ne s'applique pas, alors la DLE d'édition doit tenter d'ajouter cet abonné à la DLC existante;

iii) si un abonné d'une DLC spécifie un attribut d'authentification de DLPDU SOURCE dans une primitive "request" de DL-CONNECT, le sous-champ d'authentification de DLPDU (XX) dans la DLPDU de paramètres EC qui en résulte doit spécifier SOURCE. Si la DLC demandée était déjà établie, alors

A) si cette DLC avait été établie par un attribut d'authentification de DLPDU ORDINARY, alors la DLE de l'éditeur doit changer l'authentification en SOURCE et chaque DLPDU envoyée à partir de chaque DLCEP de la DLC doit contenir par la suite le nombre maximal autorisé d'adresses explicites;

B) autrement, lorsque A) ne s'applique pas, alors la DLE d'édition doit tenter d'ajouter cet abonné à la DLC existante;

iv) sinon si aucun de i) à iii) ne s'applique, alors le sous-champ authentification de DLPDU (XX) dans les paramètres EC doit spécifier ORDINARY, et chaque DLPDU envoyée à partir de chaque DLCEP de la DLC doit contenir le nombre minimal autorisé d'adresses explicites.

c) Le sous-champ de diversité de voie DLL (Q) des paramètres EC doit spécifier ANY-PATH comme valeur par défaut. La négociation de ce sous-champ est de ANY-PATH à THIS-PATH.

NOTE Voir la note en 7.1a)3) pour l'utilisation de la valeur THIS-PATH et le choix du chemin réel dans le cas de THIS-PATH.

d) Le sous-champ "taille d'adresse" des paramètres EC doit être déterminé comme suit.

1) Si requis par a)2)i), ou si n'importe laquelle des adresses de DL de la DLPDU EC n'a seulement qu'une représentation LONG, alors le sous-champ "taille d'adresse" des paramètres EC doit spécifier LONG.

NOTE Une adresse de DL a une représentation SHORT lorsqu'il existe une adresse de DL de deux octets (voir 5.2.1.1).

2) Sinon, lorsque 1) ne s'applique pas, et soit b)i) s'applique, soit n'importe quel autre élément de la DLC est une DLE facteur d'utilisation fractionnaire (FDC), alors le sous-champ "taille d'adresse" des paramètres EC doit spécifier SHORT.

3) Autrement, lorsque 1) et 2) ne s'appliquent pas, le sous-champ "taille d'adresse" des paramètres EC doit spécifier VERY-SHORT.

NOTE La taille d'adresse VERY-SHORT s'applique seulement aux DLPDU DT envoyées à l'aide d'un jeton de réponse (voir en 6.7.1 les formats 4 et 5); Dans tous les autres cas, la taille d'adresse SHORT est réellement utilisée.

e) Le sous-champ "DLCEP-data-delivery-features" (caractéristiques de livraison de données de DLCEP (TT)) des paramètres EC doit spécifier, indépendamment pour chaque sens de la DLC, les caractéristiques de livraison de données fournies, telles que spécifiées dans la CEI 61158-3-1, excepté que la valeur NONE doit être remplacée par UNORDERED avec une taille maximale de fenêtre (WWWW) de zéro et un sous-champ "maximum-DLSDU-size" (taille maximale de DLSDU (M...M)) de zéro dans le sens correspondant, indiquant une DLC simplex.

f) Le sous-champ "activité résiduelle" (A) des paramètres EC doit spécifier TRUE dans un sens "éditeur vers abonné" ou "homologue à homologue" d'envoi de transfert de données lorsque cela est requis par a)2)iii), ou par la gestion de DL, ou par un utilisateur de DLS éditeur ou homologue et doit spécifier FALSE autrement. La négociation de ce sous-champ est de FALSE à TRUE.

g) La négociation de la taille de fenêtre est effectuée indépendamment dans chaque sens de la DLC. La taille de fenêtre maximum actuelle pour un sens donné de transmission doit être la plus petite parmi la taille maximale de fenêtre de l'expéditeur et la taille maximale de fenêtre du destinataire dans ce sens, et le sous-champ "taille de fenêtre maximum "(WWWW) des paramètres EC doit spécifier zéro seulement lorsque le sous-champ "taille maximale de DLSDU "(M...M) dans le même sens est zéro, indiquant une DLC simplex.

h) Le sous-champ "format de paramètre SD" (FFF) des paramètres EC doit spécifier le format négocié pour chaque sens de transmission de données. Les formats initiaux pour les deux sens de transmission de données doivent être choisis satisfaire aux contraintes suivantes.

NOTE 1 Ces contraintes sont imposées par le protocole de DLC et les caractéristiques de service sélectionnées.

1) Si a) s'applique, alors le format B ne doit être choisi pour aucun de ces sens.

2) Si le sous-champ "DLCEP-data-delivery-features" (caractéristiques de livraison de données de DLCEP (TT)) dans l'un des sens spécifie ORDERED, DISORDERED, ou CLASSICAL, alors

   i) si le sous-champ "taille maximale de fenêtre" (WWWW) dans le sens en question spécifie une valeur de quatre (4) ou plus, alors le format B ne doit pas être choisi pour le sens en question;

   ii) si le sous-champ "DLCEP-class" (classe de DLCEP (CC)) spécifie PEER, alors le format choisi pour l'autre sens doit contenir à la fois les sous-champs J et K;

   iii) si le sous-champ "DLCEP-class" (CC) spécifie PUBLISHER, alors le format choisi pour ce sens doit être le format F ou G, et le format choisi pour l'autre sens doit être le format C ou D ou E;

   iv) si le sous-champ "DLCEP-class" (CC) spécifie SUBSCRIBER, alors le format choisi pour ce sens doit être le format C ou D ou E, et le format choisi pour l'autre sens doit être le format F ou G.

   NOTE Le format C est inclus dans iii) et iv) seulement parce que la prise en charge des formats D et E n'est pas obligatoire.

3) Si un sous-champ "DLCEP-data-delivery-features" (TT) spécifie ORDERED, DISORDERED ou CLASSICAL, alors le format choisi pour ce sens doit contenir un sous-champ NDS, et le format choisi pour l'autre sens doit contenir un sous-champ NDR.

4) Si un sous-champ "taille maximale de DLSDU" (M...M) spécifie une valeur plus élevée que la quantité de données utilisateur de DLS qui peut être acheminée par une seule DLPDU de la priorité de la DLC, telle que déterminée par le sous-champ "priorité de DLL" (PP) de la DLC, alors le format choisi pour ce sens doit contenir les sous-champs TNS et ASN, et le format choisi pour l'autre sens doit contenir un sous-champ RSN.

5) Si un sous-champ contenant une opportunité (G) spécifie TRUE, alors le format choisi pour ce sens doit contenir un sous-champ T.

6) Le format choisi doit être le plus court (le moins d'octets de longueur) possible de ceux que la DLE prend en charge et qui satisfait à toutes les contraintes 1) à 5).

NOTE 2 L'ensemble attendu de formats dans le sens d'envoi est comme suit.

a) Formats A, et potentiellement E, issus d'un DLCEP PEER lorsque le sous-champ "DLCEP-data-delivery-features" (TT) du sens d'envoi spécifie UNORDERED, selon le sous-champ "DLCEP-data-delivery-features" (TT) correspondant dans l'autre sens;

b) Formats B, C et D et potentiellement F, issus d'un DLCEP PEER lorsque le sous-champ "DLCEP-data-delivery-features" (TT) du sens d'envoi spécifie ORDERED, DISORDERED, ou CLASSICAL, selon le sous-champ "DLCEP-data-delivery-features" (TT) correspondant dans l'autre sens;

c) Formats A, F et G issus d'un DLCEP PUBLISHER;

d) Formats A, C, D, E et G issus d'un DLCEP SUBSCRIBER, où les formats D et G sont choisis lorsque des DLSDU abonné à éditeur ont des opportunités associées, et le format C est choisi lorsque le format D serait autrement choisi, mais indisponible parce que sa prise en charge n'est pas obligatoire. Pour les formats C, D, et G, la DLE d'édition ignorera les sous-champs NDS et K dans les paramètres SD des DLPDU CA, CD, DT et ED reçues.

i) Le sous-champ "échange bidirectionnel de données" (E) doit spécifier FALSE lorsque les sous-champs "taille maximale de DLSDU" (M...M) sont zéro dans un sens de transmission de données, ou lorsque demandé par une DLE homologue au cours d'une négociation d'une DLC PEER, ou lorsque requis par a)2)iv) ou par la gestion de DL locale ou par la construction de la DLE; et doit spécifier TRUE autrement. La négociation de ce sous-champ est de TRUE à FALSE.

j) Les attributs d'opportunité du DLCEP sont communiqués, mais ne sont pas négociés;
   1) Le sous-champ "timeliness-included" (opportunité incluse (G)) des paramètres EC doit spécifier FALSE lorsque l'opportunité de l'expéditeur spécifiée est NONE, et doit spécifier TRUE autrement.
   2) Le sous-champ "time-stamp-format" (de format de marqueur temporel (HH)) des paramètres EC doit spécifier
      i) format J lorsqu'il n'y a pas d'opportunité d'expéditeur ou l'opportunité d'expéditeur spécifiée est NONE, ou l'heure de production n'est pas demandée;
      ii) format K lorsqu'il y a une opportunité d'expéditeur et l'heure de production est demandée et la classe d'opportunité est RESIDENCE ou SYNCHRONIZED et la taille de fenêtre de temps associée (ΔT) est 1 s ou moins;
      iii) format L lorsqu'il y a une opportunité d'expéditeur et l'heure de production est demandée et la classe d'opportunité est RESIDENCE ou SYNCHRONIZED et la taille de fenêtre de temps associée (ΔT) est supérieure à 1 s;
      iv) format M lorsqu'il y a une opportunité d'expéditeur et l'heure de production est demandée et la classe d'opportunité est UPDATE ou TRANSPARENT.

k) Si un sens de communication de données n'est pas requis pour la DLC, parce que les caractéristiques de livraison de données spécifiées par un utilisateur de DLS pour le sens question spécifiaient NONE, alors, dans le sens en question,
   1) le sous-champ "residual-activity" (A) doit être spécifié comme étant FALSE;
   2) le sous-champ "Queue/Buffer" (file d'attente/tampon (B)) doit être spécifié comme étant QUEUE;
   3) le sous-champ "opportunité" (G) doit être spécifié comme étant FALSE;
   4) le sous-champ "time-stamp-format" (format de marqueur temporel (HH)) doit être spécifié comme étant FORMAT J.

8.2.1.2 Réception d'une primitive "request" de DL-CONNECT

Lorsque la DLE reçoit une primitive "request" de DL-CONNECT issue d'un utilisateur de DLS, la DLE doit effectuer la série suivante d'actions, et si une erreur est détectée au cours du processus, la DLCEP doit être déconnectée comme spécifié en 8.2.1.8.

a) La DLE doit assigner un nouvel identifiant de DLCEP, et l'identifiant d'utilisateur de DLS fourni, au DLCEP qui peut provenir de la demande, et fournir cet identifiant de DLCEP à l'utilisateur de DLS comme le seul paramètre de sortie de cette demande.

b) La DLE doit créer et démarrer un temporisateur de demandes utilisateur T_U(MCD) avec une durée basée sur le retard maximal de confirmation spécifié par un utilisateur pour les primitives des services DL-CONNECT, DL-RESET et DL-SUBSCRIBER-QUERY. Si la valeur spécifiée était autre que UNLIMITED, il convient que la durée de ce temporisateur soit **Error!**; autrement, il convient que la durée soit **Error!**. La gestion de DL peut remplacer ces durées préférentielles.

   NOTE Voir la permission au début de l'Article 8 en ce qui concerne l'utilisation de V(NRC) versus V(MRC).

c) La DLE doit valider l'adresse de DLSAP appelante ou l'identifiant de DLCEP fourni(e) par l'utilisateur de DLS; si invalide, la DLE doit rejeter la demande de DL-CONNECT par une indication de DL-DISCONNECT.

   NOTE La structure et les plages de valeurs d'une adresse de DLSAP sont spécifiées en 4.3.

d) La DLE doit valider l'auto-cohérence de l'ensemble de paramètres QoS demandés, lorsque tous les paramètres statiques et non spécifiés dynamiques prennent les valeurs par défaut associées à l'adresse de DLSAP appelante, et lorsque les réajustements automatiques suivants apportés à cette QoS se produisent.
   1) Lorsqu'un quelconque paramètre est en violation avec une limite imposée par la gestion de DL, ce paramètre doit être mis égal à cette limite, si cela est permis par les

règles de négociation spécifiées en 8.2.1.1, ou la DLE doit rejeter la demande de DL-CONNECT par une indication de DL-DISCONNECT.

2) Si une quelconque taille maximale de DLSDU est égale à zéro ou les caractéristiques de livraison de données de DLCEP d'envoi spécifient NONE, alors les caractéristiques de livraison de données de DLCEP d'envoi correspondantes doivent être mises à UNORDERED.

NOTE Ce cas spécial n'est pas considéré comme étant une violation des règles de négociation en 8.2.1.1.

3) Si une quelconque taille maximale de DLSDU est plus élevée que le maximum qui peut être acheminé par une seule et même DLPDU de la priorité de DLL spécifiée (voir 4.2.3) et les caractéristiques de livraison de données de DLCEP d'envoi dans le sens correspondant sont UNORDERED, ces caractéristiques de livraison de données DLCEP d'envoi doivent être mises à ORDERED.

e) si l'identifiant d'adresse appelante est un identifiant de DLCEP pour un DLCEP existant, alors

1) Si le DLCEP existant est un DLCEP d'éditeur, alors la DLE doit

i) utiliser l'adresse de DLSAP locale du DLCEP spécifié comme adresse de DLSAP appelante pour cette nouvelle demande DL-CONNECT;

ii) mettre chaque paramètre QoS égal au paramètre correspondant du DLCEP spécifié, si cela est permis par les règles de négociation 8.2.1.1, ou la demande de service DL-CONNECT présentée doit être rejetée par une indication de DL-DISCONNECT spécifiant "rejet de connexion – QoS non disponible, état permanent", mettant fin au traitement de la demande de DL-CONNECT;

iii) utiliser le discriminateur de réutilisation d'adresse de DLCEP d'éditeur du DLCEP spécifié dans le paramètre EC;

iv) considérer la nouvelle demande de DL-CONNECT comme étant une demande d'ajout de nouveaux abonnés à la DLC existante et continuer avec j).

2) Si la classe de DLCEP du DLCEP existant est PEER ou SUBSCRIBER, la DLE doit rejeter la demande de DL-CONNECT par une indication de DL-CONNECT.

f) Autrement, si e) ne s'applique pas, la DLE doit déterminer les tailles maximales de fenêtres d'envoi et de réception en fonction des liaisons tampon et file d'attente respectives, comme suit

1) Si les caractéristiques de DLCEP sont NONE, la taille de fenêtre correspondante doit être zéro (0).

2) Autrement, si 1) ne s'applique pas, alors

i) si la liaison était à un BUFFER, alors la taille de fenêtre correspondante doit être égale à un (1);

ii) si la liaison était à une QUEUE-*K*, alors la taille de la fenêtre correspondante doit être la plus petite valeur parmi *K* ou 15;

iii) si la liaison par défaut était utilisée, alors la taille de fenêtre correspondante doit être au moins égale à un (1);

iv) dans tous les cas, la gestion de DL peut contraindre encore plus la taille de la fenêtre.

g) Si l'adresse facultative de DLCEP appelante avait été spécifiée dans la primitive "request", alors la DLE doit affecter cette adresse de DLCEP au DLCEP; si un conflit d'affectation est détecté, alors la DLCEP doit être résilié comme spécifié en 8.2.1.8.

h) Autrement, si g) ne s'applique pas, et si l'adresse appelée n'est pas une adresse de DLCEP présumée être pour un DLCEP d'éditeur, une adresse de DLCEP allouée à la DLE et qui n'est pas en cours d'utilisation doit être affectée au DLCEP; lorsque cela est possible, elle doit être une adresse de DL SHORT de la liaison locale.

Lorsque l'adresse appelée est une adresse de DLCEP présumée être pour un DLCEP d'éditeur, cette affectation est facultative.

i) La DLC doit initialiser les variables V_C(NP), V_C(N), V_C(R), V_C(A), V_C(M), V_C(MS), V_C(H), V_C(HS) et V_C(L) du DLCEP telles que spécifiées en 4.7.4.

j) La DLE doit coder une DLPDU EC telle que spécifiée en 6.1 et 7.1:

1) : Si le paramètre d'adresse de DL appelée spécifie une adresse de DL(SAP) ou une adresse de DLCEP, la DLE doit former une DLPDU EC avec trois adresses, dont les valeurs doivent respectivement être

i) l'adresse de DL(SAP) appelée ou l'adresse de DLCEP;

ii) l'adresse de DLCEP affectée au DLCEP, ou l’adresse de DLSAP appelante si aucune affectation n’avait été faite comme dans h);

iii) l’adresse de DLSAP appelante.

2) Si le paramètre "called-DL(SAP)-address" (adresse de DL(SAP) appelée) spécifie UNKNOWN, alors

i) si l’utilisateur de DLS ne spécifiait pas une adresse de DLCEP appelante, la DLE doit rejeter la demande de DL-CONNECT par une indication de DL-DISCONNECT;

ii) Autrement si i) ne s'applique pas, et le type DLCEP est PUBLISHER, alors la DLE doit former une DLPDU EC avec deux adresses, dont les valeurs doivent respectivement être l’adresse de DLCEP affectée au DLCEP et l'adresse de DLSAP appelante;

iii) sinon si i) ne s'applique pas et la classe de DLCEP est PEER ou SUBSCRIBER, la DLE ne doit pas former et envoyer une DLPDU EC, mais doit attendre la réception d'une DLPDU EC complémentaire issue du PEER ou PUBLISHER distant.

k) Si la classe de DLCEP de la DLE doit être PEER ou SUBSCRIBER, alors

1) si une DLPDU EC a été formée, alors

i) la DLE doit positionner le champ "reply-requested" dans les paramètres EC dans la DLPDU;

ii) la DLE doit placer en file d'attente la DLPDU à la priorité TIME-AVAILABLE telle que spécifiée en 8.4.5.

2) la DLE doit activer la reconnaissance de l’adresse de DLCEP locale de la DLCEP et changer l'état du DLCEP, V_C(ST) en WAITING-FOR-EC-DLPDU.

l) Si la classe de DLCEP d'envoi de DLE doit être PUBLISHER, alors

1) la DLE doit effacer le champ "reply-requested" dans les paramètres EC dans la DLPDU;

2) si l’adresse de DLCEP source n'est pas celle d'un DLCEP existant, la DLE doit affecter une nouvelle valeur au sous-champ "publisher-DLCEP-address reuse-discriminator" (discriminateur de réutilisation d'adresse de DLCEP d'éditeur (NNN)) des paramètres EC (voir 7.1a)2)):

i) si la DLE est capable d'enregistrer le discriminateur de réutilisation d'adresse de DLCEP d'éditeur entre les incarnations de DLCEP, alors il convient qu'il maximise l'intervalle entre réutilisations de la même valeur de discriminateur;

ii) autrement, lorsque i) ne s'applique pas, la DLE doit choisir au hasard la valeur du discriminateur avec une distribution approximativement uniforme;

3) la DLE doit placer en file d'attente la DLPDU à la priorité TIME-AVAILABLE telle que spécifiée en 8.4.5.

4) la DLE doit produire la primitive "confirm" de service DL-CONNECT pour le DLCEP immédiatement après la transmission de la DLPDU EC;

5) la DLE doit annuler le temporisateur de demandes utilisateur T_U(MCD);

6) si e) ne s’applique pas, alors la DLE doit activer la reconnaissance de l’adresse de DLCEP locale du DLCEP et changer le statut du DLCEP, V_C(ST), en DATA-TRANSFER -READY.

8.2.1.3 Réception d'une primitive "response" de DL-CONNECT

Lorsque la DLE reçoit une primitive "response" de DL-CONNECT issue d'un utilisateur de DLS, la DLE doit accomplir la série d'actions suivante; si une erreur est détectée au cours du processus, le DLCEP doit être déconnecté comme spécifié en 8.2.1.8.

a) La DLE doit valider l'identifiant de DLCEP, et l'adresse de DLSAP de réponse ou l'identifiant de DLCEP, fourni par l'utilisateur de DLS, et doit associer au DLCEP l'identifiant utilisateur de DLS fourni.

b) Si le DLCEP identifié n'est pas dans l'état WAITING-FOR-CONNECT-RESPONSE, le DLCEP doit être déconnecté.

c) La DLE doit valider l'auto-cohérence de l'ensemble de paramètres QoS de réponse, lorsque tous les paramètres statiques et non spécifiés non spécifiés prennent leurs valeurs par défaut associées l'adresse de DLSAP de réponse, et lorsque les réajustements automatiques suivants apportés à cette QoS spécifiés en 8.2.1.2d) se produisent. La DLE doit ensuite valider la cohérence de l'ensemble des paramètres QoS résultants avec les paramètres correspondants de la DLPDU EC reçue, et le respect des règles de négociation de paramètre spécifiées en 8.2.1.1.

d) Si l'identifiant d'adresse de réponse contenu dans la réponse de DL-CONNECT était pour un DLCEP, alors la DLE doit considérer la nouvelle réponse de DL-CONNECT comme une demande de fusion de la DLC tentée de la réponse avec une DLC existante, après quoi l'identifiant de DLCEP qui a été spécifié dans l'indication de DL-CONNECT associée ne sera plus valide et toutes les informations d'état de fournisseur de DLS (y compris le discriminateur de réutilisation d'adresse de DLCEP d'éditeur) relatives à la DLC tentée de l'indication doivent être rejetées.

   1) Si la DLC existante est une DLC à plusieurs homologues, alors la DLE doit considérer la réponse DL-CONNECT comme une demande d'ajout d'un nouvel abonné à la DLC existante;

      i) Si le DLCEP spécifié est un DLCEP d'éditeur, alors la DLE doit

         A) utiliser l'adresse de DLCEP locale du DLCEP spécifié et l'adresse de DLSAP comme adresse de DLCEP et adresse de DLSAP de réponse pour cette réponse de DL-CONNECT;

         B) mettre chaque paramètre QoS, et le discriminateur de réutilisation d'adresse de DLCEP d'éditeur, à une valeur égale au paramètre correspondant du DLCEP spécifié, si cela est permis par les règles de négociation en 8.2.1.1;

         C) si requis par la règle en 8.2.1.1d), modifier la taille de l'adresse de la DLC existante de VERY-SHORT en SHORT ou de SHORT en LONG.

      ii) Si aucune violation de règle de négociation n'est détectée, alors la DLE doit

         A) coder une DLPDU EC ne demandant pas de réponse, avec trois adresses telles que spécifiées en 6.1 et 7.1, qui sont, respectivement,

            – la première des deux adresses de DL source issues de la DLPDU reçue ayant abouti à l'indication de DL-CONNECT et à sa réponse de DL-CONNECT consécutive;

            – l'adresse de DLCEP de la DLC existante;

            – l'adresse de DLSAP associée à cette adresse de DLCEP de réponse;

         B) placer en file d'attente la DLPDU à la priorité TIME-AVAILABLE comme spécifié en 8.4.5;

         C) arrêter le temporisateur qui avait été lancé en 8.2.1.4.2b)4)iv).

      iii) Autrement, lorsque i) ne s'applique pas parce que le DLCEP spécifié est un DLCEP d'abonné, ou lorsque ii) ne s'applique pas parce qu'une violation de règle de négociation a été détectée, alors la DLE doit rejeter la demande d'établissement de DLC reçue et mettre fin au traitement de la DLPDU EC DLPDU reçue, comme suit:

      Si l'adresse de DL de destination de la DLPDU CE reçue n'était pas une adresse de DL de groupe, alors

A) La DLE doit coder une DLPDU DC comme spécifié en 6.2 et 7.2, avec une cause de “rejet de connexion – QoS non disponible, état permanent”, et programmer la DLPDU pour transmission à la priorité TIME-AVAILABLE comme spécifié en 8.4.5.

B) La DLPDU DC doit avoir à la fois les adresses de destination et de source (voir en 6.2.1 les formats 1L et 1S), l'adresse de destination doit être identique à la première adresse de DLCEP source de la DLPDU EC reçue, qui a abouti à l’indication de DL-CONNECT et sa réponse de DL-CONNECT consécutive, et l'adresse source doit être identique à l'adresse de DL de destination de cette DLPDU EC reçue.

2) Si la classe de DLCEP du DLCEP existant est PEER, la DLE doit considérer la réponse de DL-CONNECT comme une demande de résolution d'une collision de demande de DL-CONNECT sur une DLC d'homologues qui est dans l'état WAITING-FOR-EC-DLPDU. La DLE doit appliquer les règles de négociation en 8.2.1.1 conjointement aux paramètres de QoS des deux DLCEP et doit refléter les résultats dans la QoS du DLCEP demandé à l'origine.

Si une violation de règle de négociation est détectée, ou si le DLCEP spécifié par l'adresse de réponse n'est plus dans l'état WAITING-FOR-EC-DLPDU, les deux DLCEP doivent être déconnectés comme spécifié en 8.2.1.8, mais avec une cause de "rejet de connexion – QoS non disponible, état permanent”.

Autrement, lorsqu’aucune violation de règle de négociation n’a été détectée, alors la DLE doit

i) doit utiliser l’adresse de DLCEP et l’adresse de DLSAP source de la DLPDU EC reçue, qui ont abouti à l'indication DL-CONNECT et sa réponse de DL-CONNECT consécutive, comme étant les adresse de DLCEP et adresse de DLSAP distantes du DLCEP qui était dans l'état WAITING-FOR-EC-DLPDU;

ii) considérer comme mis le champ "reply-requested" la DLPDU EC reçue, qui a abouti à l'indication de DL-CONNECT et sa réponse de DL-CONNECT consécutive;

iii) amener le DLCEP spécifié comme adresse de réponse à

– envoyer une DLPDU EC à triple adresse spécifiant que la réponse n'est pas demandée;

– arrêter le temporisateur qui avait été lancé en 8.2.1.4.2b)4)iv).

– démarrer un temporisateur comme en 8.2.1.2b), avec une durée égale à la valeur maximale pour le retard maximal de confirmation relatif à DL-CONNECT comme spécifié dans la primitive "response" de DL-CONNECT;

– activer la reconnaissance de l'adresse de DLCEP locale du DLCEP;

– changer l'état du DLCEP spécifié, V_C(ST), en WAITING-FOR-CONNECT-COMPLETION.

e) Si l'identifiant d'adresse de réponse dans la réponse de DL-CONNECT était une adresse de DLSAP, alors

– cette adresse de DLSAP doit être utilisée comme adresse de DLSAP locale;

– la DLE doit déterminer les tailles maximales de fenêtre d'envoi et de réception basées su les liaisons tampon et file d'attente respectives, éventuellement limitées encore plus par la gestion de DL, comme spécifié en 8.2.1.2.

La DLE doit alors déterminer la taille maximale de fenêtre d'envoi réelle comme étant la plus petite de la taille de fenêtre d'envoi locale et la taille de fenêtre de réception de la DLPDU EC reçue, et la taille maximale de fenêtre de réception réelle comme étant la plus petite de la taille de fenêtre de réception locale et la taille de fenêtre d'envoi de la DLPDU EC reçue, comme spécifié en 8.2.1.1. La DLE doit aussi effectuer toutes les autres négociations requises, telles que spécifiées en 8.2.1.1.

f) Si l'adresse facultative de DLCEP avait été spécifiée dans la primitive "response", alors la DLE doit affecter cette adresse de DLCEP au DLCEP; si un quelconque conflit d'affectation est détecté, alors le DLCEP doit être déconnecté comme en 8.2.1.8 avec la cause "déconnexion – appariement de DLCEP incorrect, état permanent". Autrement, une adresse de DLCEP qui n'est pas en cours d'utilisation doit être affectée au DLCEP; lorsque cela est possible, elle doit être une adresse de DL SHORT de la liaison locale.

Lorsque la DLE sert seulement d'abonné dans la DLC, aucune DLPDU EC de réponse n'est autorisée, et donc aucune affectation d'une adresse de DLCEP n'est requise.

NOTE Après que l'établissement d'un DLCEP est achevé, un abonné substitue son adresse de DLSAP appelante là où une adresse de DLCEP expéditrice aurait autrement été requise dans une DLPDU CA, CD, ED ou DT.

g) Si la classe de DLCEP de réponse est SUBSCRIBER, alors la DLE doit
   – accepter comme numéro de séquence de la première DLSDU le prochain numéro de séquence de la DLC qui est reçu;
   – arrêter le temporisateur qui avait été lancé en 8.2.1.4.2b)4)iv).
   – émettre une indication de DL-CONNECTION-ESTABLISHED;
   – activer la reconnaissance de l'adresse de DLCEP (éditeur) distante du DLCEP;
   – changer l'état du DLCEP, V_C(ST), en DATA-TRANSFER-READY.
h) Autrement, si g) ne s'applique pas, la DLE doit
   1) coder une DLPDU EC ne demandant pas de réponse, comportant trois adresses telles que spécifiées en 6.1 et 7.1, où ses adresses sont, respectivement,
      – la première des deux adresses de DL source issues de la DLPDU reçue ayant abouti à l'indication de DL-CONNECT et à sa réponse de DL-CONNECT consécutive;
      – l'adresse de DLCEP qui vient d'être affectée au DLCEP;
      – l'adresse de DLSAP de réponse;
   2) attribuer un discriminateur de réutilisation d'adresse de DLCEP d'éditeur lorsque la classe de DLCEP de réponse est PUBLISHER;
   3) programmer la DLPDU pour la transmission à la priorité TIME-AVAILABLE telle que spécifiée en 8.4.5.

   NOTE Cette procédure amène un DLCEP éditeur à répondre à une DLPDU EC issu d'un DLCEP d'abonné proposé en envoyant directement une DLPDU EC à l'adresse temporaire de DLCEP appelante de ce DLCEP abonné. En variante, la DLE d'éditeur peut créer un DLCEP éditeur indépendant (par l'utilisation d'une demande de DL-CONNECT) après avoir reçu l'indication de DL-CONNECT) provenant de l'abonné proposé, et puis fusionner le DLCEP d'éditeur demandé de l'abonné avec celui qui vient juste d'être établi. Cette dernière approche fait envoyer une DLPDU EC à l'adresse de DL(SAP) appelée de l'éditeur avant que la DLPDU EC de réponse ne soit envoyée à l'abonné demandeur.
i) Si la classe de DLCEP de réponse est PUBLISHER, alors la DLE doit
   – arrêter le temporisateur qui avait été lancé en 8.2.1.4.2b)4)iv).
   – émettre une indication de DL-CONNECTION-ESTABLISHED;
   – activer la reconnaissance de l'adresse de DLCEP locale du DLCEP;
   – changer l'état du DLCEP, V_C(ST), en DATA-TRANSFER-READY.
j) Si la classe de DLCEP de réponse est PEER, alors la DLE doit
   – arrêter le temporisateur qui avait été lancé en 8.2.1.4.2b)4)iv).
   – démarrer un temporisateur comme en 8.2.1.2b), avec une durée égale à la valeur pour le retard maximal de confirmation relatif à DL-CONNECT comme spécifié dans la primitive "response" de DL-CONNECT;
   – activer la reconnaissance de l'adresse de DLCEP locale du DLCEP;
   – changer l'état du DLCEP, V_C(ST), en WAITING-FOR-CONNECT-COMPLETION.

8.2.1.4 Réception d'une DLPDU EC

Lorsque la DLE reçoit une DLPDU EC, la DLE doit déterminer le numéro de version du protocole de DL utilisé, tel que spécifié dans la DLPDU EC reçue, et doit interpréter les autres paramètres-EC de la DLPDU en conséquence.

8.2.1.4.1 Réception d'une DLPDU EC comportant deux adresses

La DLE doit accomplir la série d'actions suivante, et si une erreur est détectée au cours du processus, alors la DLC doit être déconnectée comme spécifié en 8.2.1.8:

a) Si la première adresse (source) de la DLPDU EC reçue est une adresse de DLCEP PUBLISHER pour un DLCEP SUBSCRIBER existant et ce DLCEP SUBSCRIBER est dans l'état WAITING-FOR-EC-DLPDU, alors la DLE doit
   i) valider les paramètres DLC reçues par rapport à ceux qui étaient précédemment demandés pour le DLCEP SUBSCRIBER (et qui n'ont éventuellement pas été envoyés dans une DLPDU EC);
   ii) mettre les paramètres du DLCEP existant à une valeur égale aux paramètres de DLC reçus, si cela est permis par les règles de négociation en 8.2.1.1.

   Si une erreur n'est détectée, alors la DLE doit déconnecter le DLCEP SUBSCRIBER comme spécifié en 8.2.1.8, avec une cause de “rejet de connexion – QoS non disponible, état permanent”.

   Si aucune erreur n'est détectée au cours de cette validation, alors la DLE doit

   iii) mettre le discriminateur de réutilisation d'adresse de DLCEP à la valeur de celui contenu dans la DLPDU EC reçue;
   iv) annuler le temporisateur de demandes utilisateur T_U(MCD);
   v) accepter comme numéro de séquence de la première DLSDU le prochain numéro de séquence de la DLC qui est reçu;
   vi) émettre une primitive "confirm" de DL-CONNECT vers l'utilisateur de DLS;
   vii) changer l'état du DLCEP SUBSCRIBER à l'état DATA-TRANSFER-READY.

b) Lorsque la première adresse (source) de la DLPDU EC reçue est une adresse de DLCEP PUBLISHER pour un DLCEP SUBSCRIBER existant, mais a) ne s'applique pas parce que le DLCEP SUBSCRIBER n'est pas dans l'état WAITING-FOR-EC-DLPDU, alors la DLE doit comparer le discriminateur de réutilisation de l'adresse de DLCEP d'éditeur du DLCEP existant à celui qui est contenu dans la DLPDU EC reçue. Si les deux valeurs sont égales, la DLE doit mettre la taille d'adresse du DLCEP existant à celle contenue dans la DLPDU EC reçue.

c) Autrement, lorsque a) et b) ne s'appliquent pas, le DLCEP SUBSCRIBER existant doit être déconnecté comme spécifié en 8.2.1.8, avec une cause de "déconnexion – mauvais discriminateur de réutilisation d'adresse de DLCEP éditeur, état permanent”.

8.2.1.4.2 Réception d'une DLPDU EC comportant trois adresses

La DLE doit accomplir la série d'actions suivante, et si une erreur est détectée au cours du processus, alors la DLC doit être déconnectée comme spécifié en 8.2.1.8:

a) Si la première adresse contenue dans la DLPDU EC reçue est une adresse de DL de groupe associée à plus d'un des utilisateurs DLS de la DLE, alors la DLE doit traiter chacun de ces utilisateurs DLS comme si cet utilisateur avait individuellement reçue la DLPDU EC. Cependant, aucune DLPDU DC ne doit être envoyée comme réponse directe à la DLPDU EC reçue.

b) Si la première adresse de la DLPDU EC reçue est une adresse de DLSAP, alors
   1) la DLE doit valider l'auto-cohérence de la DLPDU EC reçue, où tous les paramètres statiques et dynamiques non spécifiés prennent les valeurs par défaut associées à cette adresse de DLSAP appelée, et où n'importe quel paramètre en violation d'une

limite imposée par une gestion DL locale doit être mis à la valeur de cette limite, si cela est permis par les règles de négociation en 8.2.1.1, ou le DLCEP doit être déconnecté comme spécifié en 8.2.1.8 avec une cause de "rejet de connexion – QoS non disponible, état permanent";

2) la DLE doit vérifier si, oui ou non, un utilisateur de DLS associé à cette adresse de DLSAP a un DLCEP actif dont l'adresse de DLCEP distante est égale à l'adresse de DLCEP source spécifiée dans la DLPDU EC reçue;

3) si un tel DLCEP actif existe, alors

   i) si

      – le DLCEP est dans l'état waiting-for-connect-completion;

      – la DLPDU EC reçue demande une réponse

   alors une DLPDU EC de retour, adressée à la première adresse de DL source spécifiée dans la DLPDU EC reçue, avec les adresses source égales aux adresses de DLCEP et de DLSAP locales du DLCEP, et spécifiant les paramètres du DLCEP actif, et ne demandant pas de réponse, doit être codée et doit être placée en file d'attente à la priorité TIME-AVAILABLE telle que spécifiée en 8.4.5;

   ii) sinon lorsque i) ne s'applique pas, la DLE doit déconnecter le DLCEP comme spécifié en 8.2.1.8 avec une cause de "rejet de connexion – état de DLCEP incohérent, état permanent";

NOTE Cette reconfirmation de la DLC existante ne provoque pas de changement de l'état du DLCEP répondeur.

4) si aucun DLCEP actif de ce type n'existe, la DLE doit attribuer un nouvel identifiant de DLCEP au DLCEP, et doit appliquer les règles de négociation en 8.2.1.2d). Si une violation des règles de négociation a lieu, la DLE doit déconnecter le DLCEP proposé comme spécifié en 8.2.1.8 avec une raison de "rejet de connexion – QoS non disponible, état permanent". Si aucune violation n'est détectée, la DLE doit, pour chaque utilisateur de DLS associé à l'adresse de DLSAP ou à l'adresse DL de groupe qui était la première adresse de la DLPDU EC reçue,

   i) créer un DLCEP, en initialisant ses variables $V_S(NP)$, $V_S(N)$, $V_S(R)$, $V_S(A)$, $V_S(M)$, $V_S(MS)$, $V_S(H)$, $V_S(HS)$ et $V_S(L)$ comme spécifié en 4.7.4;

   ii) enregistrer l'adresse de DLCEP source et l'adresse de DLSAP source issues de la DLPDU EC reçue comme étant respectivement l'adresse de DLCEP distante et l'adresse de DLSAP distante du DLCEP, et lorsque la classe de DLCEP de l'expéditeur est PUBLISHER, également enregistrer le discriminateur de réutilisation d'adresse de DLCEP éditeur de la DLPDU EC comme étant le discriminateur de réutilisation d'adresse de DLCEP éditeur local du DLCEP;

   iii) rapporter une indication de DL-CONNECT à l'utilisateur de DLS;

   iv) démarrer un temporisateur pour surveiller la réponse de l'utilisateur de DLS à l'indication de DL-CONNECT, comme spécifié en 8.2.1.2b);

   v) changer l'état du DLCEP, $V_C(ST)$, en WAITING-FOR-CONNECT-RESPONSE.

c) Sinon, si la première adresse de la DLPDU EC reçue est une adresse de DLCEP pour un DLCEP existant, et si le DLCEP adressé est dans l'état WAITING-FOR-EC-DLPDU, alors la DLE doit valider les paramètres DLC reçus, et les adresse de DL source de la DLPDU reçue lorsque leurs valeurs attendues sont connues, par rapport à ceux qui ont été envoyés dans une précédente DLPDU EC, et si une erreur est détectée, alors

1) si l'adresse appelée du DLCEP était une adresse de DL de groupe, alors la DLE doit répondre par une DLPDU DC ne demandant pas une réponse, adressée à la première adresse de DL source spécifiée dans la DLPDU EC reçue, avec une adresse source égale à l'adresse de DLCEP locale du DLCEP, et doit autrement ignorer la DLPDU EC reçue.

2) autrement, lorsque l'adresse appelée du DLCEP était une adresse de DLSAP, alors la DLE doit déconnecter le DLCEP comme spécifié en 8.2.1.8 avec une cause de "rejet de connexion – QoS non disponible, état permanent".

Si aucune erreur n'est détectée au cours de la validation de la DLPDU EC reçue, alors

3) Si la classe de DLCEP du DLCEP de réception est PEER et la DLPDU EC reçue demande une réponse, alors

   i) les deux adresses de DL source de la DLPDU EC reçue doivent être notées comme étant l'adresse de DLCEP distante et l'adresse de DLSAP distante du DLCEP;

   ii) une DLPDU EC de retour, adressée à la première adresse de DL source spécifiée dans la DLPDU EC reçue, avec les adresses source égales aux adresses de DLCEP et de DLSAP locales du DLCEP, et spécifiant les paramètres du DLCEP actif, et ne demandant pas de réponse, doit être codée et doit être placée en file d'attente à la priorité TIME-AVAILABLE telle que spécifiée en 8.4.5;

   iii) la DLE doit démarrer un temporisateur comme spécifié en 8.2.1.2b), avec une durée égale à la valeur pour le retard maximal de confirmation relatif à DL-CONNECT comme spécifié dans la primitive "request" de DL-CONNECT;

   iv) la DLE doit changer l'état en WAITING-FOR-CONNECT-COMPLETION.

4) Sinon, lorsque la classe de DLCEP du DLCEP de réception est SUBSCRIBER ou la DLPDU EC reçue ne demande pas de réponse, alors

   i) si la classe de DLCEP du DLCEP de réception est PEER ou SUBSCRIBER, alors les deux adresses de DL source de la DLPDU EC reçue doivent être notées comme étant l'adresse de DLCEP distante et l'adresse de DLSAP distante du DLCEP;

   ii) si la classe de DLCEP du DLCEP de réception PEER, alors

      – une DLPDU DT ne contenant pas de données utilisateur de DLS;

      – avec une adresse de destination égale à la première adresse de DL source spécifiée dans la DLPDU EC reçue;

      – lorsque les attributs du DLCEP exigent que la DLPDU ait une adresse source, avec une adresse source égale à l'adresse de DLCEP locale du DLCEP

      doit être codée et doit être placée en file d'attente à la priorité de DLCEP telle que spécifiée en 8.4.5, pour notifier à la DLE homologue la réception réussie de la DLPDU EC de confirmation

   iii) si la classe de DLCEP du DLCEP de réception est SUBSCRIBER, alors la DLE doit accepter, comme étant le numéro de séquence de la première DLSDU, le prochain numéro de séquence de la DLC qui est reçu;

   iv) la DLE doit émettre une primitive "confirm" de DL-CONNECT, acheminant les attributs de DLCEP négociés, vers l'utilisateur de DLS demandeur;

   v) la DLE doit annuler le temporisateur de demandes utilisateur T_U(MCD) TU(MCD) et changer l'état du DLCEP en DATA-TRANSFER-READY.

d) Sinon, si la première adresse de la DLPDU EC reçue est une adresse de DLCEP pour un DLCEP existant, et si le DLCEP adressé est dans l'état WAITING-FOR-CONNECT-COMPLETION, la DLE doit valider les paramètres DLC reçus par rapport à ceux qui ont été envoyés dans une précédente DLPDU EC, et si une erreur est détectée, la DLE doit déconnecter le DLCEP comme spécifié en 8.2.1.8 avec une cause de "rejet de connexion – QoS non disponible, état permanent".

Si aucune erreur n'est détectée au cours de cette validation, alors

1) si la DLPDU EC reçue demande une réponse, alors

   i) une DLPDU EC de retour, adressée à la première adresse de DL source spécifiée dans la DLPDU EC reçue, avec les adresses source égales aux adresses de DLCEP et de DLSAP locales du DLCEP, et spécifiant les paramètres du DLCEP actif, et ne demandant pas de réponse, doit être codée et doit être placée en file d'attente à la priorité TIME-AVAILABLE telle que spécifiée en 8.4.5;

   ii) la DLE doit redémarrer le temporisateur $_U$(MCD) avec la même période que la fois précédente.

2) sinon, lorsque 1) ne s'applique pas, alors

i) une DLPDU DT ne contenant pas de données utilisateur de DLS,

– avec une adresse de destination égale à la première adresse de DL source spécifiée dans la DLPDU EC reçue;

– si les attributs du DLCEP exigent que la DLPDU ait une adresse source, avec une adresse source égale à l'adresse de DLCEP locale du DLCEP

doit être codée et doit être placée en file d'attente à la priorité de DLCEP telle que spécifiée en 8.4.5, pour notifier à la DLE homologue la réception réussie de la DLPDU EC de confirmation

ii) si l'état avant .WAITING-FOR-CONNECT-COMPLETION était WAITING-FOR-EC-DLPDU, alors la DLE doit émettre une primitive «confirm» de DL-CONNECT, acheminant vers l'utilisateur de DLS demandeur les attributs de DLCEP négociés;

iii) sinon, lorsque ii) ne s'applique pas, alors l'état avant WAITING-FOR-CONNECT-COMPLETION était WAITING-FOR-CONNECT-RESPONSE, et la DLE doit émettre une primitive "indication" de DL-CONNECTION-ESTABLISHED vers l'utilisateur de DLS répondeur;

iv) la DLE doit annuler le temporisateur de demandes utilisateur T_U(MCD) TU(MCD) et changer l'état du DLCEP en DATA-TRANSFER-READY.

e) Sinon si la première adresse de la DLPDU EC reçue est une adresse de DLCEP pour un DLCEP existant, et la DLPDU EC demande une réponse, et si le DLCEP adressé est dans l'état DATA-TRANSFER-READY, alors

1) si le DLCEP existant est un DLCEP d'éditeur, alors la DLE doit

i) mettre chaque paramètre QoS, et le discriminateur de réutilisation d'adresse de DLCEP d'éditeur, à une valeur égale au paramètre correspondant du DLCEP spécifié, si cela est permis par les règles de négociation en 8.2.1.1;

ii) si requis par la règle en 8.2.1.1d), modifier la taille de l'adresse de la DLC existante de VERY-SHORT en SHORT ou de SHORT en LONG.

2) Si aucune violation de règle de négociation n'est détectée, alors la DLE doit

i) coder une DLPDU EC ne demandant pas de réponse, comportant deux adresses telles que spécifiées en 6.1 et 7.1, où ses adresses sont, respectivement,

– l'adresse de DLCEP de la DLC existante;

– l'adresse de DLSAP associée à cette adresse de DLCEP existant;

ii) programmer la DLPDU EC pour la transmission à la priorité TIME-AVAILABLE telle que spécifiée en 8.4.5.

3) Lorsque 2) ne s'applique pas, car une violation de règle de négociation a été détectée, la DLE doit rejeter la demande d'établissement DLC reçue et mettre fin au traitement de la DLPDU EC reçue, comme suit:

i) La DLE doit coder une DLPDU DC telle que spécifiée en 6.2 et 7.2, avec son champ "réponse requise" mis à FALSE, avec la cause de "déconnexion à l'initiative du fournisseur - QoS non disponible, état permanent", et programmer la DLPDU DC pour la transmission à priorité TIME-AVAILABLE telle que spécifiée en 8.4.5.

ii) La DLPDU DC doit avoir à la fois les adresses de destination et de source (voir en 6.2.1 les formats 1L et 1S)), l'adresse de destination doit être identique à la première adresse de DL source de la DLPDU EC reçue, et l'adresse source doit être identique à l'adresse de DL de destination de cette DLPDU EC reçue.

f) Autrement, la DLE doit ignorer la DLPDU EC reçue.

8.2.1.5 Expiration du temporisateur T_U(MCD)

Si le temporisateur T_U(MCD) expire, alors si l'état du DLCEP, V_C(ST), est

a) waiting-for-EC-DLPDU, alors

1) s'il s'agit de la (V(NRC)+1)^{ème} expiration consécutive, alors

NOTE Voir la permission au début de l'Article 8 en ce qui concerne l'utilisation de V(NRC) versus V(MRC).

i) la DLE doit mettre fin au traitement de la demande;

ii) si le retard maximal de confirmation spécifié par l'utilisateur relatif à la primitive "request" de DL-CONNECT spécifiait une valeur autre que UNLIMITED, alors

A) alors la DLE doit initier une indication de DL-DISCONNECT rapportant la cause "rejet de connexion – DLSAP injoignable, état transitoire, origine locale";

B) si l'adresse appelée était soit une adresse de DLSAP, soit une adresse de DLCEP, et la classe de DLCEP du DLCEP est PEER, alors la DLE

– doit coder une DLPDU DC demandant la déconnexion, avec une cause de "cause non spécifiée", à la même adresse de DL que celle à laquelle la DLPDU EC antérieure a été envoyée;

– doit être placée en file d'attente à la priorité TIME-AVAILABLE telle que spécifiée en 8.4.5;

iii) autrement, lorsque le retard maximal de confirmation spécifié par l'utilisateur relatif à la primitive "request" de DL-CONNECT spécifiait une valeur de UNLIMITED, la DLE doit maintenir la demande en file d'attente, et le DLCEP dans son état courant, mais ne doit initier aucune autre action concernant cette demande.

NOTE La DLE peut encore répondre à la réception d'une DLPDU EC qui tente d'établir une connexion avec le DLCEP cité plus haut, en acceptant ou en rejetant la connexion présentée.

2) autrement, si 1) ne s'applique pas, la DLE doit

i) redémarrer le temporisateur T_U(MCD) avec la même période que la fois précédente;

ii) remettre en file d'attente la même DLPDU EC pour retransmission à la priorité. TIME-AVAILABLE telle que spécifiée en 8.4.5;

b) WAITING-FOR-CONNECT-RESPONSE, alors la DLE doit déconnecter le DLCEP comme spécifié en 8.2.1.8, en spécifiant une cause de déconnexion de "déconnexion à l'initiative d'un fournisseur – temporisation";

c) waiting-for-connect-completion, alors

1) s'il s'agit de la (V(NRC)+1)^{ème} expiration consécutive, alors la DLE doit déconnecter le DLCEP comme spécifié en 8.2.1.8, en spécifiant une cause de déconnexion de "déconnexion à l'initiative d'un fournisseur – temporisation";

NOTE Voir la permission au début de l'Article 8 en ce qui concerne l'utilisation de V(NRC) versus V(MRC).

2) autrement, si 1) ne s'applique pas, alors la DLE doit renvoyer la DLPDU EC comportant trois adresses qui avait été envoyée à l'entrée dans l'état WAITING-FOR-CONNECT-COMPLETION et encore une fois redémarrer le temporisateur T_U(MCD) comme spécifié en 8.2.1.2b).

d) DATA-TRANSFER-READY, alors la DLE doit agir comme spécifié en 8.2.2.10;

e) WAITING-FOR-RESET-COMPLETION, alors la DLE doit agir comme spécifié en 8.2.2.18.

8.2.1.6 Réception d'une primitive "request" de DL-DISCONNECT

Lorsque la DLE reçoit à un DLCEP une demande de DL-DISCONNECT issue d'un utilisateur de DLS, alors la DLE

a) doit coder une DLPDU DC comme spécifié en 6.2 et 7.2, demandant la déconnexion et spécifiant la cause donnée par l'utilisateur de DLS et doit programmer la DLPDU pour transmission à la priorité TIME-AVAILABLE telle que spécifiée en 8.4.5, sauf lorsque le DLCEP

1) est dans l'état WAITING-FOR-CONNECT-RESPONSE et l'adresse de DL de destination de la DLPDU EC qui a activé le DLCEP est une adresse de DL de groupe; ou

2) est dans l'état WAITING-FOR-EC-DLPDU et l'adresse de DL(SAP) appelée est une adresse de DL de groupe ou est UNKNOWN; ou

3) est un DLCEP SUBSCRIBER;

Si une DLPDU DC est codée, alors

i) si le DLCEP en cours de déconnexion est un DLCEP PEER, alors la DLPDU DC doit avoir à la fois les adresses de destination et de source (voir en 6.2.1 les formats 1L et 1S), et l'adresse de destination doit être l'adresse de DLCEP distante de la DLC, si elle est connue, ou l'adresse de DL(SAP) appelée de la DLPDU EC initiatrice dans tous les autres cas. Le champ "reply-requested" doit être mis à TRUE dans les paramètres DC de la DLPDU DC initiatrice.

ii) si le DLCEP en cours de déconnexion est un DLCEP PUBLISHER, alors la DLPDU DC doit avoir seulement une adresse source (voir en 6.2.1 les formats 2L et 2S). Le champ "reply-requested" doit être mis à FALSER dans les paramètres DC de la DLPDU DC initiatrice.

iii) l'adresse source de la DLPDU DC doit être une adresse de DLCEP locale, s'il en existe une; ou l'adresse de DLSAP locale de répondeur ou appelante, s'il en existe une, ou l'adresse de DLSAP appelée de la DLPDU EC initiatrice dans tous les autres cas.

b) doit mettre fin au DLCEP, y compris

1) pour chaque demande de DL-DATA en cours (c'est-à-dire, pas encore confirmée):

i) retirer la demande de la file d'attente appropriée de demandes utilisateur de DLCEP, Q_A(UR), et les références à la demande de toutes les files d'attente de la DLE;

ii) initier une primitive "confirm" de DL-DATA avec l'identifiant de demande associé rapportant la cause "échec – réinitialisation ou déconnexion";

iii) supprimer le temporisateur T_U(MCD) associé à la demande.

2) si une demande de DL-RESET est en cours, alors initier une primitive "confirm" de DL-RESET avec l'identifiant de DLCEP de l'utilisateur de DLS rapportant la cause '"échec – déconnexion";

3) supprimer toutes les temporisateurs associés à ce DLCEP.

Après la déconnexion, si une adresse de DLCEP locale associée au DLCEP a été incluse dans n'importe quelle DLPDU envoyée sur la liaison, alors la DLE doit s'assurer que l'adresse de DLCEP n'est pas réutilisée pendant une période de temps excédant la plus grande valeur parmi

1) deux fois la durée de vie maximale de DLPDU dans le réseau, V(NDL);

2) le retard D…D_{CR} de DLCEP, lorsque ce retard n'est pas UNLIMITED.

Dans le diagramme d'états à la Figure 11, cette procédure est modélisée comme une entrée dans un état supplémentaire, Aging-DLCEP-address (ces: adresse de DLCEP vieillissante), qui est quitté à la fin de la période de temps spécifiée ci-dessus.

8.2.1.7 Réception d'une DLPDU DC

Lorsque la DLE reçoit une DLPDU DC, spécifiant qu'il convient que le DLCEP soit déconnecté, alors

a) la DLE doit déterminer le numéro de version du protocole de DL en cours d'utilisation, tel que spécifié dans la DLPDU DC reçue, et doit interpréter les autres paramètres DC de la DLPDU en conséquence;

b) si la DLPDU DC reçue demande une réponse, alors une DLPDU DC de retour, adressée à l'adresse de DL source spécifiée dans la DLPDU DC reçue, et spécifiant une cause de déconnexion de "déconnexion ou rejet de connexion, origine inconnue – cause non spécifiée", et ne demandant pas de réponse, doit être codée et doit être placée en file d'attente à la priorité TIME- telle que spécifiée en 8.4.5;

c) Si la DLPDU DC reçue

1) spécifie seulement une adresse source (voir en 6.2.1 les formats 2L et 2S)) et l'adresse source est une adresse de DLCEP d'une DLC à plusieurs homologues à laquelle la DLE est abonnée; ou
2) spécifie à la fois les adresses de destination et de source (voir en 6.2.1 les formats 1L et 1S), où
   i) l'adresse de destination est une adresse de DL(SAP) et la DLE a un DLCEP, en un DLSAP auquel cette adresse de DL(SAP) est liée, dont l'adresse de DLCEP distante a la même valeur que l'adresse de DL source reçue; ou
   ii) l'adresse de destination est une adresse de DLCEP, et l'adresse de DLCEP distante du DLCEP identifié a la même valeur que l'adresse de DL source reçue; ou
   iii) l'adresse de destination est une adresse de DLCEP, et l'adresse de DLCEP appelée du DLCEP identifié a la même valeur que l'adresse de DL source reçue;

   alors si le DLCEP est connu de l'utilisateur de DLS local, alors
3) la DLE doit rapporter une indication de DL-DISCONNECT à l'utilisateur de DLS local spécifiant à la fois l'origine non locale et la cause de l'indication de DL-DISCONNECT telles que reçues dans la DLPDU DC;
4) la DLE doit mettre fin au DLCEP comme spécifié en 8.2.1.6b);
5) après la déconnexion, la DLE doit s'assurer que toute adresse de DLCEP qui avait été assignée au DLCEP n'est pas réutilisée pendant une période de temps excédant la plus grande valeur parmi
   i) deux fois la durée de vie maximale de DLPDU dans le réseau, V(NDL);
   ii) le retard D...D_{CR} de DLCEP, lorsque ce retard n'est pas UNLIMITED.

8.2.1.8 Déconnexion initiée par une DLE

Lorsque la DLE détermine par elle-même qu'il est nécessaire de déconnecter le DLCEP, alors

a) si le DLCEP est connu de l'utilisateur de DLS local, alors la DLE doit rapporter une indication DL-DISCONNECT à l'utilisateur de DLS local, spécifiant à la fois la cause de l'indication de DL-DISCONNECT et que son origine était locale.

   NOTE Le DLCEP ne sera pas connu de l'utilisateur de DLS local si la déconnexion a lieu pendant le traitement d'une DLPDU EC reçue dont la réception avait juste déclenché la DL pour créer le DLCEP.

b) Si
   1) la classe de DLCEP du DLCEP est PEER ou PUBLISHER;
   2) l'adresse de DL(SAP) appelée de la DLPDU EC qui a activée le DLCEP n'était pas une adresse de DL de groupe,

   alors
   i) la DLE doit coder une DLPDU DC comme spécifié en 6.2 et 7.2 et doit programmer la DLPDU pour transmission à la priorité TIME-AVAILABLE telle que spécifiée en 8.4.5;
   ii) si le DLCEP en cours de déconnexion est un DLCEP PEER, alors la DLPDU DC doit avoir à la fois les adresses de destination et de source (voir en 6.2.1 les formats 1L et 1S), et l'adresse de destination doit être l'adresse de DLCEP distante de la DLC, si elle est connue, ou l'adresse de DL(SAP) appelée de la DLPDU EC initiatrice dans tous les autres cas. Le champ "reply-requested" doit être mis à TRUE dans les paramètres DC de la DLPDU DC initiatrice.
   iii) si le DLCEP en cours de déconnexion est un DLCEP PUBLISHER, alors la DLPDU DC doit avoir seulement une adresse source (voir en 6.2.1 les formats 2L et 2S) et le champ "reply-requested "doit être mis à FALSE dans les paramètres DC de la DLPDU DC initiatrice;
   iv) l'adresse source de la DLPDU DC doit être une adresse de DLCEP locale, s'il en existe une; ou l'adresse de DLSAP locale de répondeur ou appelante, s'il en existe

une, ou l'adresse de DLSAP appelée de la DLPDU EC initiatrice dans tous les autres cas.

c) la DLE doit mettre fin au DLCEP comme spécifié en 8.2.1.6b).

Après la déconnexion, la DLE doit s'assurer que toute adresse de DLCEP qui avait été assignée au DLCEP n'est pas réutilisée pendant une période de temps excédant la plus grande valeur parmi

1) deux fois la durée de vie maximale de DLPDU dans le réseau, V(NDL);
2) le retard $D...D_{CR}$ de DLCEP, lorsque ce retard n'est pas UNLIMITED.

8.2.2 Fonctionnement des services "DLC data transfer" et "DLCEP reset"

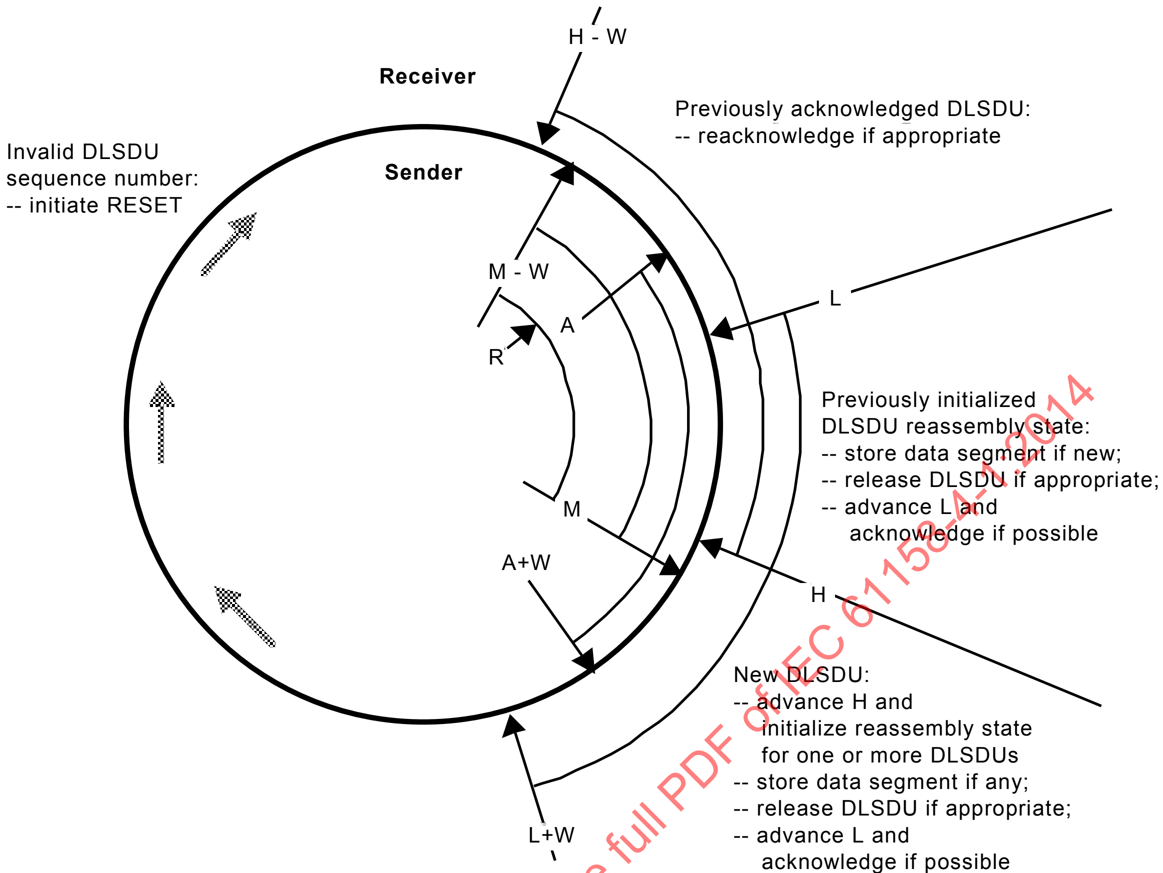
Les primitives du service "DLC data transfer" sont "request", "indication" et "confirm" de DL-DATA, "indication" de DL-BUFFER-SENT et "indication" de DL-BUFFER-RECEIVED. Les primitives des services de réinitialisation de DLCEP sont "request", "indication", "response" et "confirm" de DL-RESET et "indication" de DL-RESET-COMPLETED. Une réinitialisation de DLCEP peut être invoquée, par n'importe quel utilisateur de DLS ou le fournisseur de DLS, à tout moment après l'établissement du DLCEP et avant la libération du DLCEP.

Si les caractéristiques de livraison de données de DLCEP d'envoi dans l'un des sens de la DLC sont ORDERED, DISORDERED, ou CLASSICAL, alors les variables linéaires (conceptuellement illimitées de DLCEP spécifiées en 4.7.4 et à la Figure 7 sont projetées sur un anneau cyclique de '2^N identifiants de séquence (voir Figure 12). Chaque DLSDU est associée à un de ces identifiants avant transmission, et toutes les DLPDU qui acheminent des segments de la DLSDU spécifient également cet identifiant associé. Le partitionnement dynamique de ces identifiants et la catégorisation résultante des DLSDU associées à ces identifiants sont également montrés à Figure 12.

NOTE Les relations entre les variables linéaires (conceptuellement illimités) de DLCEP spécifiées en 4.7.4 et à la Figure 7 sont répétées ici pour aider à la compréhension de la Figure 12.

$$H–W \le M–W \le A \le L \le H \le M \le A+W \le L+W$$

$$M–W \le R \le M$$



Légende

Anglais	Français
invalid DLSDU sequence number; -- initiate RESET	numéro de séquence de DLSDU non valide; -- lancer la réinitialisation
Receiver	Destinataire
Sender	Expéditeur
Previously acknowledged DLSDU: -- reacknowledge if appropriate	DLSDU acquittée précédemment; -- acquitter de nouveau, le cas échéant
Previously initialized DLSDU reassembly state; -- store data segment if new; -- release DLSDU if appropriate; -- advance L and acknowledge if possible	état de réassemblage de DLSDU initialisé précédemment; -- stocker le segment de données s'il est nouveau; -- libérer la DLSDU si cela est approprié; -- avancer L et l'acquitter si possible
New DLSDU: -- advance H and initialize reassembly state for one or more DLSDU -- store data segment if any; -- release DLSDU if appropriate; -- advance L and acknowledge if possible	Nouvelle DLSDU; -- avancer H et initialiser l'état de réassemblage pour une ou plusieurs DLSDU -- stocker le segment de données, le cas échéant; -- libérer la DLSDU si cela est approprié; -- avancer L et l'acquitter si possible

Figure 12 – Projection des variables «numéro de séquence» de DLCEP expéditeur et destinataire de la Figure 7 sur les paramètres numéro de séquence cyclique des DLPDU CA, CD, DT, ED et RC, avec la détermination consécutive des actions requises

Chaque DLPDU avec des paramètres SD non vides peut acheminer à la DLE homologue les deux à cinq bits d'ordre inférieur des $V_C(L)$ et $V_C(M)$ de l'expéditeur, comme les sous-champs (NDR mod 2^R) et (NDS mod 2^S) de paramètres SD.

8.2.2.1 Sélection du format des DLPDU CA, CD, DT et ED

Les formats d'adresse de toutes les DLPDU CA, CD, ED et DT envoyées à partir d'un DLCEP doivent être choisis tels que déterminés au cours du processus d'établissement DLCEP (voir 8.2.1.1) et tels que respectivement spécifiés en 6.4.3, 6.5.3, 6.6.3 et 6.7.3. Le format de paramètre SD de toutes ces DLPDU CA, CD, ED et DT formées par la DLE doit être le même que celui qui est négocié pour le DLCEP d'envoi (voir 7.1c)5), 7.1d)5), et 8.2.1.1).

Toutes les DLPDU CD envoyées depuis une DLE LAS comme partie intégrante de ses activités d'exécution de programme, et non depuis un DLCEP de la DLE LAS, doivent spécifier une adresse de destination explicite ayant la longueur négociée en 8.2.1.1 et doivent omettre l'adresse source et les paramètres SD.

NOTE Un format d'adresse VERY-SHORT est toujours réalisé en utilisant des adresses SHORT dans de n'importe quelles DLPDU CA, CD, DC, ED et RC associées.

8.2.2.2 Réception d'une primitive "request" de DL-DATA

Si la demande est acceptée, comme indiqué par un statut retourné de "succès" pour la demande de DL-DATA, alors, à la suite de l'achèvement de la demande, soit avec succès, soit après l'échec, la DLE doit émettre une primitive "confirm" de DL-DATA avec le même identifiant de demande, tel que spécifié par l'utilisateur de DLS dans la primitive "request" de DL-DATA correspondante, acheminant le statut de la demande vers l'utilisateur de DLS.

Il convient que la source DLCEP spécifiée dans la demande de DL-DATA soit liée à une file d'attente explicite (commandée par l'utilisateur) ou une file d'attente implicite (commandée par la DLE). Si la file d'attente est pleine, ou si la longueur de DLSDU spécifiée, $P_U(L)$, est non valide, ou si l'état de DLCEP $V_C(ST)$ n'est pas DATA-TRANSFER-READY, alors la DLE doit immédiatement retourner la primitive "confirm" de DL-DATA correspondante indiquant la cause de l'échec.

Autrement

a) la DLE doit créer et démarrer un temporisateur de demandes utilisateur $T_U(MCD)$ avec une durée basée sur le retard maximal de confirmation spécifié par l'utilisateur pour la primitive DL-DATA. Si la valeur spécifiée était autre que UNLIMITED, la durée de ce temporisateur doit être égale à ce retard maximal de confirmation spécifié par l'utilisateur; autrement, il convient que la durée soit 60 s. La gestion de DL peut remplacer ces durées préférentielles
b) la DLE doit assigner le prochain numéro de séquence non assigné $N = V_C(N)$ à la demande et à sa DLSDU associée;
c) la DLE doit initialiser la variable $V_{C,N}(SS)$ en fonction de la longueur, $P_N(L)$, de la $N^{ème}$ DLSDU, pour indiquer que tous les segments de la $N^{ème}$ DLSDU, et aucun autre segment de cette DLSDU, ont besoin de transmission;
d) la DLE doit abouter la demande à la file d'attente de demandes utilisateur de l'adresse de DLCEP, $Q_A(UR)$, comme suit.
   1) Si l'attribut "DL-scheduling-policy" (politique de programmation de DL) de l'adresse de DLCEP source est EXPLICIT, alors la demande doit être placée dans la troisième partition de $Q_A(UR)$, où elle doit attendre une demande de DL-COMPEL-SERVICE pour la libérer pour transmission.

      NOTE La libération d'une demande différée s'applique aux DLSDU, et non aux DLPDU.

   2) Sinon si 1) ne s'applique pas, et
      i) si $N > V_C(A) + P_C(WS)$, et le DLCEP expéditeur est un homologue CLASSICAL ou DISORDERED, ou
      ii) si $N > V_C(R) + P_C(WS)$, et le DLCEP expéditeur est ORDERED et $V_C(R)$ existe, ou est un éditeur ou abonné CLASSICAL ou DISORDERED,

      alors, la demande doit être placée dans la troisième partition de $Q_A(UR)$.

3) Sinon, si 1) et 2) ne s'appliquent pas, la troisième partition de Q_A(UR) QA(UR) est vide, et la demande doit donc être placée dans la deuxième partition de Q_A(UR), et la DLE doit abouter à la file d'attente de services non programmés de la DLE, Q(US), une référence à Q_A(UR) de la même priorité que la demande qui vient juste d'être aboutée.

NOTE Q(US) n'a jamais besoin d'avoir plus de références à une Q_A(UR) que le nombre des DLSDU attendant une transmission ou une retransmission.

e) La DLE doit incrémenter V_C(N).

8.2.2.3 Transmission d'une DLPDU DT à partir d'un DLCEP

Dans ce qui suit, soit MUD défini comme étant le nombre maximal d'octets de données utilisateur qui peuvent être acheminés dans une seule et même DLPDU DT de priorité de DLCEP telle que spécifiée en 4.2.3.

À la réception d'une occasion de transmission pour une DLSDU placée en file d'attente ou en tampon, la DLE doit former et envoyer une DLPDU DT de la priorité spécifiée; avec des formats de champ "adresse" de DL et champ "paramètres SD" tels que spécifiés en 6.7, 7.4 et 8.2.2.1; avec les adresses de DLCEP distantes (destination) et locales (source) de la DLC, selon le cas; avec la longueur et le contenu du champ "données utilisateur", et le contenu du champ "paramètres SD," comme suit;

NOTE À l'exception du sous-champ "'utilisation finale de jeton" du premier octet (contrôle de trame) de la DLPDU DT; probablement à l'exception des sous-champs NDR, RSN, J, K et T, s'ils sont présents, du format de paramètres SD négocié pour la transmission à partir du DLCEP expéditeur, le reste de la DLPDU DT de données peut être formé lorsque la demande est placée en file d'attente, et peut ne pas être formé dynamiquement au moment de la transmission.

8.2.2.3.1 Formation du champ "données utilisateur" et des sous-champs "paramètres SD" connexes

Les sous-champs T, NDS, TNS, ASN et "truncated-DL-time" (temps de DL tronqué) du champ "SD-parameters" et le champ " données utilisateur" de la DLPDU doivent être formés comme suit.

a) Si

   i) le DLCEP expéditeur est un DLCEP DISORDERED ou CLASSICAL PUBLISHER, ou un DLCEP ORDERED PEER ou PUBLISHER et V_C(R) existe;

   ii) $V_C(M) = V_C(R)+P_C(WS)$ et $V_C(M) +1 < V_C(N)$ et $V_{C,V_C(M)}(SS)$ est vide;

   iii) une primitive "confirm" de DL-DATA a été émise pour la $(V_C(R)+1)^{ème}$ demande de DL-DATA

   alors, la DLE doit

   1) annuler le temporisateur associé à la $(V_C(R)+1)^{ème}$ demande de DL-DATA;

   2) incrémenter V_C(R).

b) Si le DLCEP expéditeur est lié à une file d'attente d'envoi, et est

   i) un DLCEP SUBSCRIBER, ou DLCEP UNORDERED PEER ou PUBLISHER, et il existe un plus petit K tel que $V_C(M) < K \le V_C(N)-1$ et $V_{C,K}(SS)$ est non vide

   NOTE Dans ce cas $V_{C,K}(SS)$ peut contenir un seul élément, car ces DLSDU ne peuvent pas exiger une segmentation.

   ii) un DLCEP DISORDERED ou CLASSICAL PEER, et il existe un K tel que $V_C(A) < K \le \min(V_C(A)+P_C(WS), V_C(N)-1)$ et $V_{C,K}(SS)$ est non vide;

   iii) un DLCEP DISORDERED ou CLASSICAL PUBLISHER, ou un DLCEP ORDERED PEER ou PUBLISHER et V_C(R) existe; et il existe un K tel que $V_C(R) < K \le \min(V_C(R)+P_C(WS), V_C(N)-1)$ et $V_{C,K}(SS)$ est non vide; ou

   iv) un DLCEP ORDERED PEER or PUBLISHER et V_C(R) n'existe pas; et il existe un K tel que $V_C(M) \le K \le \min(V_C(M)+P_C(WS), V_C(N)-1)$ et $V_{C,K}(SS)$ est non vide; ou

   alors la DLE doit former le reste de la DLPDU comme suit.

1) Le sous-champ NDS, s'il est présent, doit acheminer les trois ou cinq bits d'ordre inférieur de la valeur K, selon le cas.

2) Si S est l'indice d'origine zéro de l'élément de plus petit numéro de l'ensemble $V_{C,K}(SS)$, qui est le numéro de segment d'un segment de la $K^{ème}$ DLSDU demandant la transmission, alors le sous-champ ASN, s'il est présent, doit acheminer la valeur de S.

NOTE Il est possible pour une mise en œuvre de ce protocole de DL d'accomplir une segmentation de DLSDU au moment de la transmission de DLPDU. Une telle approche de mise en œuvre pourrait réduire au maximum le traitement des données utilisateur de DLS de la DLE.

3) Le sous-champ TNS, s'il est présent, doit acheminer la valeur P_K(SDUL) -1)/MUD, arrondie par défaut, qui est le numéro d'origine zéro des segments dans la $K^{ème}$ DLSDU.

4) Le champ "données utilisateur "doit être composé des octets
$S \times$ MUD +1 à min((S+1) $\times$ MUD, P_K(SDUL)), inclusivement, de P_K(SDU).

5) Le sous-champ T doit spécifier FALSE.

6) le sous-champ "truncated-DL-time" ne doit pas être présent.

la DLE doit retirer le numéro d'élément S de $V_{C,K}(SS)$.

c) Si le DLCEP expéditeur est lié à un tampon d'envoi, alors

1) la DLE doit incrémenter $V_C(N)$ si le tampon a été écrit depuis la dernière transmission à partir du tampon sur ce DLCEP;

2) la DLE doit laisser K égal à $V_C(N)$-1;

3) si K n'est pas égal à zéro, alors la DLE doit former le reste de la DLPDU comme spécifié de b)1 à b)4 et comme suit.

i) Si la DLSDU n'a pas d'attribut d'opportunité, alors le sous-champ T doit spécifier FALSE.

ii) Si le DLCEP a une classe "DL-timeliness" d'expéditeur autre que NONE, alors la DLE doit

A) calculer l'opportunité du Sème segment de la Kème DLSDU comme spécifié en 8.1.7;

B) exécuter une opération ET logique de ce statut d'opportunité calculé avec le statut d'opportunité associé à l'écriture dans le tampon, $V_B(TS)$ (voir 4.7.4.21);

C) acheminer ce résultat dans le sous-champ T de la DLPDU.

iii) Si présent, le sous-champ "truncated-DL-time" (e temps de DL tronqué) doit acheminer les octets appropriés de l'heure de production associée au tampon.

la DLE doit retirer le numéro d'élément S de $V_{C,K}(SS)$.

d) S'il n'existe pas un tel K comme en b) ou si K est égal à zéro en c), ou si la DLE doit envoyer la DLPDU sans données utilisateur comme en 6.4.4.1c), 6.5.4.1c), et 6.6.4.1c); ou 6.4.4.1d), 6.5.4.1d), et 6.6.4.1d), alors

1) les sous-champs T, TNS, ASN et "temps de DL tronqué" des paramètres SD, s'ils sont présents, doivent être codés comme étant zéro (0); NDS doit être codé sous la forme d'un nombre approprié de bits d'ordre inférieur de $V_C(M)$;

2) le sous-champ "temps de DL tronqué" peut être vide (c'est-à-dire omis et pas émis);

3) le champ "données utilisateur" doit être vide.

8.2.2.3.2 Formation des autres sous-champs de paramètres SD

Les sous-champs J, K, NDR et RSN des paramètres SD doivent être formés comme suit:

a) Si le DLCEP expéditeur est un DLCEP PUBLISHER, alors les sous-champs J, K, NDR et RSN des paramètres SD doivent être codés comme étant (0).

b) Si le DLCEP expéditeur est un DLCEP a PEER ou SUBSCRIBER), alors

1) s'il y a un K le plus petit tel que $V_C(L) < K \leq V_C(H)$ et $V_{C,K}(RRS)$ est non vide, alors si S est l'indice d'origine zéro de l'élément de plus petit numéro de l'ensemble $V_{C,K}(RRS)$, qui est le numéro de segment d'un segment absent de la $K^{ème}$ DLSDU, et soit $K < V_C(H)$, soit $S \leq V_C(HS)$, alors

i) le sous-champ J des paramètres SD, s'il est présent, doit être codé comme étant un (1);

ii) le sous-champ NDR, s'il est présent, doit acheminer les deux ou quatre bits d'ordre inférieur de la valeur K;

iii) le sous-champ RSN, s'il est présent, doit acheminer la valeur de S

iv) le segment S doit être retiré de $V_{C,K}(RRS)$.

2) Autrement, si 1) ne s'applique pas, soit le champ paramètre SD doit être vide, si cela est permis par 8.2.2.1; soit

i) les sous-champs J et RSN doivent être codés comme étant zéro (0), s'ils sont présents;

ii) le sous-champ NDR, s'il est présent, doit acheminer les deux ou quatre bits d'ordre inférieur de $(V_C(L)+1)$.

3) Si les sous-champs K et NDR des paramètres SD sont tous les deux présents, et la valeur du sous-champ NDR est égale à la valeur des bits d'ordre inférieur correspondants de $(V_C(L)+1)$, alors le sous-champ K doit être codé comme étant (1); autrement, le sous-champ K doit être codée comme étant (0).

8.2.2.3.3 Achèvement de transmission

a) Si la DLPDU qui vient juste d'être émise contenait des données utilisateur de DLS, alors la DLE doit mettre à jour $V_C(M)$ et $V_C(MS)$ à partir des variables locales K et S en 8.2.2.3.1 comme suit.

– Si $K > V_C(M)$, ou $K = V_C(M)$ et $S > V_C(MS)$, alors $V_C(M)$ doit être mis égal à K, et $V_C(MS)$ doit être mis égal à S.

b) Si le DLCEP expéditeur

– est un DLCEP homologue dont les caractéristiques de DLCEP d'envoi sont UNORDERED, ou est DLCEP éditeur dont les caractéristiques de DLCEP d'envoi sont UNORDERED, ou est un DLCEP abonné;

– la DLSDU DT a un champ "données utilisateur" non vide qui contient le dernier, ou seul, segment d'une DLSDU;

alors

1) si la source d'envoi est un tampon, alors la DLE doit émettre une primitive "indication" de DL-BUFFER-SENT spécifiant l'identifiant d'utilisateur de DLS s'il est connu, ou l'identifiant de DL autrement, pour le DLCEP; ou

2) si la source d'envoi est une file d'attente et s'il y a une demande de service DL-DATA non confirmé pour cette DLSDU, la DLE doit

i) doit émettre une primitive "confirm" de DL-DATA avec le même identifiant de demande que cette demande de DL-DATA, rapportant "succès";

ii) annuler et supprimer le temporisateur $T_U(MCD)$ associé.

c) Si le DLCEP expéditeur

– est un DLCEP éditeur dont les caractéristiques de DLCEP d'envoi sont ORDERED, DISORDERED ou CLASSICAL, ou est un DLCEP homologue dont les caractéristiques de DLCEP d'envoi sont ORDERED;

– la DLSDU DT a un champ "données utilisateur" non vide qui contient le dernier, ou seul, segment d'une DLSDU;

alors

1) si la source d'envoi est un tampon, alors la DLE doit émettre une primitive "indication" de DL-BUFFER-SENT spécifiant l'identifiant d'utilisateur de DLS, s'il est connu, ou l'identifiant de DL autrement, pour le DLCEP; soit
2) si la source d'envoi est une file d'attente et s'il y a une demande de service DL-DATA non confirmé pour cette DLSDU, alors
   i) la DLE doit émettre une primitive "confirm" de DL-DATA avec le même identifiant de demande que cette demande de DL-DATA, rapportant "succès";
   ii) si $V_C(R)$ existe, et $V_C(M) - P_C(WS) > V_C(R)$, alors la DLE doit mettre $V_C(R)$ égal à $V_C(M) - P_C(WS)$;
   iii) la DLE doit conserver la DLSDU pour une retransmission potentielle aux abonnés
      – jusqu'à l'expiration du temporisateur $T_U(MCD)$ associé; ou
      – jusqu'à ce que la nécessité besoin d'émettre une autre DLSDU avec une augmentation résultante de $V_C(M)$, VC(M), lorsqu'elle est couplée avec la taille de fenêtre négociée $P_C(WS)$, exige le rejet de la DLSDU conservée avec une augmentation résultante de $V_C(R)$.

d) Si ce DLCEP a été spécifié comme un DLCEP de synchronisation pendant l'établissement d'un ou plusieurs autres DLCEP locaux, et si une primitive "indication" de DL-BUFFER-SENT a été émise en b)1) ou en c)1), alors la DLE doit enregistrer le temps de DL de l'accès au réseau, $V_C(TNA)$, pour une utilisation dans les calculs d'opportunité de ce(s) DLCEP de référence.

8.2.2.4 Transmission d'une DLPDU CA, CD ou ED à partir d'un DLCEP

8.2.2.4.1 Transmission d'une DLPDU CA

À réception d'une opportunité de transmission pour forcer une transmission à partir d'un DLCEP distant, lorsque

a) la DLC est
   1) simplex, avec transmission de données utilisateur de DLS seulement à partir du DLCEP local vers le DLCEP distant; ou
   2) duplex, avec transmission de données utilisateur de DLS à partir du DLCEP local vers le DLCEP distant, et à partir de ce DLCEP distant vers le DLCEP local (et éventuellement d'autres DLCEP), et les attributs de DLCEP ne permettent pas l'envoi d'une DLPDU ED à partir du DLCEP vers le DLCEP disant;
b) la DLE expéditrice doit déterminer l'état du DLCEP distant;

alors la DLE doit former et envoyer une DLPDU CA de la priorité spécifiée; avec des formats de champ "adresse" de DL et champ "paramètres SD" tels que spécifiés en 6.5, 7.4 et 8.2.2.2; avec les adresses de DLCEP distantes (destination) et locales (source) de la DLC, selon le cas; et avec un contenu de champ "paramètres SD" et contenu du champ "données utilisateur" tels que spécifiés en 8.2.2.3.

8.2.2.4.2 Transmission d'une DLPDU CD

Le présent paragraphe ne s'applique pas à la DLE LAS quand elle envoie des DLPDU CD comme partie intégrante de son activité programmée et non à partir d'un DLCEP de la DLE LAS; de telles DLPDU sont contraintes comme spécifié en 6.5 et 8.2.2.

À réception d'une opportunité de transmission pour forcer une transmission à partir d'un DLCEP distant, lorsque

a) la DLC est
   1) simplex, avec transmission de données utilisateur de DLS seulement à partir du DLCEP distant vers le DLCEP local (et possiblement vers d'autres DLCEP); ou

2) duplex, avec transmission de données utilisateur de DLS à partir du DLCEP local vers le DLCEP distant, et à partir de ce DLCEP distant vers le DLCEP local (et possiblement vers d'autres DLCEP), et soit

   i) les attributs DLCEP ne permettent pas d'envoyer une DLPDU ED à partir du DLCEP local vers le DLCEP distant, soit

   ii) il n'y a pas de segments DLSDU attendant la transmission vers l'utilisateur de DLS distant comme spécifié par les critères de sélection en 8.4.2.1a);

b) l'exécution locale d'une primitive "request" de DL-COMPEL-SERVICE ou d'une action de service Compel programmée, force la transmission à partir d'un DLCEP homologue ou éditeur distant;

alors la DLE doit former et envoyer une DLPDU CD de la priorité spécifiée; avec des formats de champ "adresse" de DL et champ "paramètres SD" tels que spécifiés en 6.5, 7.4 et 8.2.2.2; avec les adresses de DLCEP distantes (destination) et locales (source) de la DLC, selon le cas; et avec le contenu du champ "paramètres SD", comme suit:

NOTE À l'exception du sous-champ "'utilisation finale de jeton" du premier octet (contrôle de trame) de la DLPDU CD; et probablement à l'exception des sous-champs NDR, RSN, J et K, s'ils sont présents, du format de paramètres SD négocié pour la transmission à partir du DLCEP expéditeur, le reste de la DLPDU CD de données peut être formé lorsque la demande est placée en file d'attente, et peut ne pas être formé dynamiquement au moment de la transmission.

1) les sous-champs T, NDS, TNS, ASN et "temps de DL tronqué" des paramètres SD, s'ils sont présents, doivent être codés comme étant zéro (0);
2) Le sous-champ "temps de DL tronqué" peut être vide (c'est-à-dire omis et pas émis);
3) Le champ "données utilisateur" doit être vide.
4) Les sous-champs J, K, NDR et RSN des paramètres SD doivent être formés comme spécifié en 8.2.2.3.2.

8.2.2.4.3 Transmission d'une DLPDU ED

À réception d'une opportunité de transmission pour forcer une transmission à partir d'un DLCEP distant, lorsque

a) la DLC est duplex, avec transmission de données utilisateur de DLS à partir du DLCEP local vers le DLCEP distant, et à partir de ce DLCEP distant vers le DLCEP local (et possiblement vers d'autres DLCEP);
b) les attributs de DLCEP permettent d'envoyer une DLPDU ED à partir du DLCEP vers le DLCEP distant;
c) un ou plusieurs segments DLSDU attendent la transmission vers l'utilisateur de DLS distant comme spécifié par les critères de sélection en 8.4.2.1a);
d) l'exécution locale d'une primitive "request" de DL-COMPEL-SERVICE ou d'une action de service Compel programmée, force la transmission à partir d'un DLCEP homologue ou éditeur distant;

alors la DLE doit former et envoyer une DLPDU FD de la priorité spécifiée; avec des formats de champ "adresse" de DL et champ "paramètres SD" tels que spécifiés en 6.5, 7.4 et 8.2.2.2; avec les adresses de DLCEP distantes (destination) et locales (source) de la DLC, selon le cas; et avec un contenu de champ "paramètres SD" et un contenu du champ "données utilisateur" tels que spécifiés en 8.2.2.3.

8.2.2.5 Validation et traitement des paramètres SD dans une DLPDU CA, CD, ED ou DT DLPDU reçue en un DLCEP

Si l'état du DLCEP, V_C(ST), est

1) waiting-for-connect-completion, alors

   i) si l'état avant WAITING-FOR-CONNECT-COMPLETION était WAITING-FOR-EC-DLPDU, la DLE doit émettre une primitive "confirm" de DL-CONNECT, acheminant les attributs de

DLCEP négociés, vers l'utilisateur de DLS demandeur et annuler le temporisateur de demandes utilisateur associé T_U(MCD);

iii) sinon, lorsque i) ne s'applique pas, alors l'état avant WAITING-FOR-CONNECT-COMPLETION était WAITING-FOR-CONNECT-RESPONSE, et la DLE doit émettre une primitive "indication" de DL-CONNECTION-ESTABLISHED vers l'utilisateur de DLS destinataire et annuler le temporisateur de demandes utilisateur associé T_U(MCD);

iii) la DLE doit changer l'état du DLCEP, V_C(ST), en DATA TRANSFER READY; et doit appliquer le reste du le présent paragraphe.

2) waiting-for-reset-completion, alors

i) la DLE doit émettre une primitive "indication" de DL-RESET-COMPLETED vers l'utilisateur de DLS destinataire en spécifiant l'identifiant d'utilisateur de DLS pour le DLCEP s'il est connu, ou l'identifiant de DL pour le DLCEP autrement;

ii) la DLE doit annuler le temporisateur de demandes utilisateur associé T_U(MCD);

iii) la DLE doit changer l'état du DLCEP, V_C(ST), en DATA TRANSFER READY; et doit appliquer le reste du le présent paragraphe.

3) autre que WAITING-FOR-CONNECT-COMPLETION, autre que WAITING-FOR-RESET-COMPLETION et autre que DATA-TRANSFER-READY, la DLPDU reçue doit être ignorée par les fonctions de DLC de niveau supérieur.

Autrement, la DLE doit valider et traiter les paramètres SD de la DLPDU reçue selon le format de paramètre SD, P_C(NP.FFF$_R$), négocié pour ce sens (récepteur) de transmission de la DLC. Cette validation et ce traitement doivent être tels que spécifiés dans le reste de 8.2.2.5, avec les considérations dépendant d'un format suivantes, basées sur le format de paramètre SD (A à G) et sur le format de temps de DL tronqué (J à M). La valeur dépendant d'un format pour le module d'envoi MOD_S doit également être utilisée dans les procédures en 8.2.2.6.

Format A) Les paramètres SD d'envoi et de réception de la DLPDU sont implicites et donc toujours valides; les valeurs implicites de RSN, T, TNS, ASN et temps de DL tronqué sont toutes zéro; et toute donnée utilisateur d'accompagnement est une DLSDU complète. 8.2.2.5.2 ne s'applique pas.

Format B) Les paramètres SD d'envoi et de réception de la DLPDU sont explicites; le module d'envoi MOD_S est égal à 2^3; le module de réception MOD_R est égal à 2^2; les valeurs implicites de RSN, TNS, et ASN sont zéro; et toute donnée utilisateur d'accompagnement est une DLSDU complète.

Format C) Les paramètres SD d'envoi et de réception de la DLPDU sont explicites; le module d'envoi MOD_S est égal à 2^5; le module de réception MOD_R est égal à 2^4; et toute donnée utilisateur d'accompagnement ne peut être qu'une DLSDU partielle.

Format D) Les paramètres SD d'envoi et de réception de la DLPDU sont explicites; le module d'envoi MOD_S est égal à 2^5; le module de réception MOD_R est égal à 2^4; les valeurs implicites de TNS, et ASN sont zéro; et toute donnée utilisateur d'accompagnement est une DLSDU complète.

Format E) Les paramètres SD d'envoi de la DLPDU sont "non-existent" (inexistants); les paramètres SD de réception de la DLPDU sont explicites; le module de réception MOD_R est égal à 2^4; les valeurs implicites de J, K et T sont telle que spécifiées en 7.4.2.1E); la valeur implicite de NDS est V_C(H)+1; les valeurs implicites de TNS, et ASN sont zéro; et il ne peut pas y avoir de données utilisateur d'accompagnement.

Format F) Les paramètres SD d'envoi de la DLPDU sont explicites; les paramètres SD de réception de la DLPDU sont "non-existent" (inexistants); le module d'envoi MOD_S est égal à 2^5; la valeur implicite de NDR est V_C(M)+1; la valeur implicite de RSN est zéro; et toute donnée utilisateur d'accompagnement ne peut être qu'une DLSDU partielle.

Format G) Les paramètres SD d'envoi de la DLPDU sont explicites; les paramètres SD de réception de la DLPDU sont "non-existent" (inexistants); le module d'envoi MOD_S est égal à 2^5; la valeur implicite de NDR est $V_C(M)+1$; les valeurs implicites de RSN, TNS, et ASN sont zéro; et toute donnée utilisateur d'accompagnement est une DLSDU complète.

8.2.2.5.1 Validation des sous-champs "NDS", "TNS", "ASN" et "truncated-DL-time" des paramètres SD reçus

Dans ce qui suit, $P_C(NP.WWWW_R)$ est la taille de fenêtre de réception négociée et $P_C(NP.TT_R)$ représente les caractéristiques négociées de livraison de données du DLCEP de réception.

a) Si $P_C(NP.TT_R)$ spécifie UNORDERED, comme cela est toujours le cas avec le format A, alors si le champ "données utilisateur" de la DLPDU reçue est non vide, alors la DLE de réception
   i) doit incrémenter $V_C(H)$, et doit laisser *K* égal à la nouvelle valeur de $V_C(H)$;
   ii) doit créer la variable $V_{C,K}(MRS)$ avec une valeur indiquant que le numéro de segment zéro (0) de la $K^{ème}$ DLSDU est absent;
   iii) doit traiter les données utilisateur reçues comme spécifié en 8.2.2.6.
b) Autrement, lorsque $P_C(NP.TT_R)$ spécifie ORDERED, DISORDERED ou CLASSICAL, et si le DLCEP de réception est un DLCEP abonné, et il s'agit de la première DLPDU DT reçue après que l'état DLCEP a été changé en DATA-TRANSFER-READY, alors la DLE doit mettre les variables $V_C(L)$ et $V_C(H)$ à la valeur du sous-champ $N_R(NDS)$ de la DLPDU DT reçue.

   alors, la DLE doit calculer

$$TEMP = (\ N_R(NDS)\ +\ P_C(NP.WWWW_R)\ -\ V_C(H)\ -1\ )\ \text{modulo}\ MOD_S \tag{22}$$

   1) Si

$$TEMP\ >\ (\ V_C(L)\ +\ 2 \times P_C(NP.WWWW_R)\ -\ V_C(H)\ -1\ )\ \text{modulo}\ MOD_S$$

      alors
      i) si $P_C(NP.TT_R)$ est ORDERED, alors la DLE doit
         A) mettre $V_C(L)$ à la valeur de
            $V_C(L) + ((\ N_R(NDS)\ -\ (P_C(NP.WWWW_R) + V_C(L)))\ \text{modulo}\ MOD_S)$;
         B) annuler tous les temporisateurs $T_{C,N}(RRS)$ qui peuvent exister, où *N* est inférieur ou égal à $V_C(L)$. Il convient de libérer toutes les ressources de DLE consacrées à la réception et au réassemblage des DLSDU ayant des numéros de séquence inférieurs ou égaux à $V_C(L)$;
      ii) Si $P_C(NP.TT_R)$ est CLASSICAL ou DISORDERED, le numéro de séquence de la DLSDU reçue est non valide; les procédures en 8.2.2.6 ne s'appliquent pas; et la DLE doit initier une réinitialisation au DLCEP (voir 8.2.2.19).
   2) Sinon si 1)ii) ne s'applique pas, alors si soit
      – $TEMP\ >\ (\ P_C(NP.WWWW_R) -1\ )$, soit
      – $TEMP\ = (\ P_C(NP.WWWW_R) -1\ )$ et $N_R(ASN)\ >\ V_C(HS)$

      alors
      i) la DLE doit mettre *N* égal à $TEMP - (\ P_C(NP.WWWW_R) -1\ )$;
      ii) si $N > 0$, alors le numéro de séquence de la DLSDU reçue concerne une nouvelle DLSDU, qui n'a pas été précédemment reçue ou inférée; la DLE doit répéter l'étape A) ci-après *N* fois et en suite une fois l'étape B).
         A) La DLE doit incrémenter $V_C(H)$. Soit *K* égal à la valeur juste incrémentée de $V_C(H)$. Alors $V_{C,K}(MRS)$ doit être créé et doit indiquer que tous les segments possibles de la $K^{ème}$ DLSDU, basés sur la taille maximale de

DLSDU négociée, sont absents; et $V_{C,K}$(RRS) doit être créé et doit indiquer que le numéro de segment zéro (0) de la $K^{ème}$ DLSDU est absent.

NOTE 1 En variante, la DLE peut simplement établir $V_{C,K}$(MRS) pour indiquer que tous les (16) segments sont absents; des procédures ultérieures corrigeront le nombre de segments absents à ceux de la DLSDU réelle.

NOTE 2 Cette combinaison de valeurs pour $V_{C,K}$(MRS) et $V_{C,K}$(RRS) assure que tous les segments de la $K^{ème}$ DLSDU seront reçus avant que la DLSDU réassemblée ne soit livrée à l'utilisateur de DLS.

NOTE 3 La DLE répète *N* fois l'étape ci-dessus.

B) si les paramètres SD reçus contiennent un champ TNS explicite, alors la DLE doit modifier $V_{C,K}$(MRS) afin d'indiquer que tous les segments, dont le numéro base zéro est supérieur à la valeur de TNS, ne sont pas absents, lorsque *K* est égal à la nouvelle valeur de V_C(H) après l'étape A).

iii) si *N* = 0, alors la DLE doit mettre *K* égal à la valeur de V_C(H).

iv) pour toutes les valeurs de *N*,

A) la DLE doit mettre la variable V_C(HS) égale à la valeur du champ N_R(ASN). S'il n'y pas de données utilisateur d'accompagnement dans la DLPDU reçue, alors la DLE doit modifier à la fois $V_{C,K}$(MRS) et $V_{C,K}$(RRS) afin indiquer que le segment dont le numéro base zéro est égal à la valeur du champ N_R(ASN) n'est pas absent; et les procédures en 8.2.2.6 doivent aussi être appliquées..

B) s'il existe un $V_{C,K}$(RRS), tel que créé en b)2)ii)A), qui n'est pas vide et qui exige donc une demande de retransmission, et si le DLCEP de réception est un DLCEP CLASSICAL ou DISORDERED, ou facultativement, est un DLCEP ORDERED, alors

– la DLE doit vérifier une référence au DLCEP sur la file d'attente de services non programmés de la DLE, Q(US);

– si une telle référence n'est pas trouvée, alors la DLE doit ajouter une référence au DLCEP dans la file d'attente de services non programmés de la DLE, Q(US), afin d'assurer qu'une autre DLPDU demandant la retransmission du segment absent est envoyée à partir du DLCEP de réception.

3) Sinon si 1) et 2) ne s'appliquent pas, et

$TEMP < (V_C(L) + P_C(NP.WWWW_R) – V_C(H))$ modulo MOD_S

alors le numéro de séquence de la DLSDU reçue est pour une DLSDU précédemment livrée, et sur des DLC d'homologues précédemment acquittées. S'il existe des données utilisateur d'accompagnement dans la DLPDU reçue, et le DLCEP est un DLCEP homologue CLASSICAL ou DISORDERED, alors la DLE doit effectuer une vérification pour détecter une référence au DLCEP sur la file d'attente de services non programmés de la DLE, Q(US), et si elle n'en trouve pas, alors ajouter une référence au DLCEP à la file de services non programmés de la DLE, Q(US), afin d'assurer qu'une autre DLPDU acquittant de nouveau la DLSDU qui vient juste d'être référencée est envoyée à partir du DLCEP de réception

Si la liaison de réception du DLCEP est explicitement ou implicitement à une file d'attente, alors les procédures en 8.2.2.6 ne s'appliquent pas. Si la liaison de réception du DLCEP est à un tampon, alors toute donnée utilisateur de DLS dans la DLPDU doit être rejetée et la réception de la DLPDU en doublon doit être rapportée à l'utilisateur de DLS avec une indication de DL-BUFFER-RECEIVED spécifiant que la DLSDU rapportée est une DLSDU en doublon.

4) Sinon, si 1), 2) et 3) ne s'appliquent pas, alors le numéro de séquence de la DLSDU reçue est pour une DLSDU précédemment reçue ou inférée, mais non encore acquittée, ou livrée, ou les deux.

Soit $K = V_C(H) + TEMP + 1 – P_C(NP.WWWW_R)$.

Si les paramètres SD reçus contiennent un sous-champ TNS explicite, alors $V_{C,K}$(MRS) et $V_{C,K}$(RRS) doivent être modifiés tous les deux pour indiquer que tous les segments dont le numéro base zéro est supérieur à TNS ne sont pas absents. Sil n'existe pas de données utilisateur d'accompagnement dans la DLPDU reçue,) et $V_{C,K}$(MRS) indique que les données utilisateur n'ont pas été reçues précédemment, alors la DLE doit modifier à la fois $V_{C,K}$(MRS) et $V_{C,K}$(RRS) pour indiquer que le segment dont le numéro base zéro est égal à la valeur du champ N_R(ASN) n'est pas absent, les procédures en 8.2.2.6 doivent également être appliquées.

8.2.2.5.2 Validation des sous-champs NDR, RSN, J et K des paramètres SD reçus

Dans ce qui suit, P_C(NP.WWWW$_R$) est la taille de fenêtre de réception négociée et P_C(NP.TT$_R$) représente les caractéristiques négociées de livraison de données du DLCEP de réception.

a) Si le DLCEP est un DLCEP abonné, et les sous-champs NDR, RSN J et K, s'ils sont présents, des paramètres SD reçus ne sont pas tous zéro, alors la DLE doit se déconnecter du DLCEP comme spécifié en 8.2.1.8 avec une cause de “déconnexion à l'initiative d'un fournisseur – mauvais format ou paramètres de DLPDU, état permanent”.

b) Si le DLCEP est un DLCEP homologue CLASSICAL ou DISORDERED, et les sous-champs J et K des paramètres SD reçus ne sont pas zéro tous les deux, alors la DLE doit calculer

$$TEMP = (N_R(NDR) - V_C(A)) \text{ modulo } MOD_R \tag{23}$$

$$N = TEMP + V_C(A) \tag{24}$$

La DLPDU reçue acquitte une DLSDU précédemment émise non acquittée (K=1), ou demande la retransmission d'un segment d'une DLSDU émise précédemment (J=1), ou les deux.

Si K=1, et le DLCEP est un DLCEP homologue CLASSICAL ou DISORDERED, et $V_C(A) < N \le V_C(M) + 1$, alors la DLE doit

i) mettre V_C(A) égal à *N-1*;

ii) émettre, dans l'ordre demandé initialement, une primitive "confirm" de DL-DATA pour chaque primitive "request" de DL-DATA qui avait été acquittée par le NDR reçu;

iii) annuler l'ensemble de temporisateurs de demandes utilisateur associés { T_U(MCD) } pour les primitives "request" de DL-DATA qui viennent juste d'être confirmées;

iv) annuler tous les éventuels temporisateurs de retransmission $T_{C,K}$(SS) associées aux demandes de DL-DATA qui viennent juste d'être confirmées, ou le temporisateur simplifié T_C(SS) associé au DLCEP, et dans ce dernier cas (utilisant T_C(SS)), si $V_C(A) < V_C(M)$, qui implique l'existence de DLSDU non acquittées, alors T_C(SS) doit être redémarré;

v) lorsque cela est possible et permis, déplacer les demandes de DL-DATA de la troisième partition à la deuxième partition de la file d'attente de demandes utilisateur correspondante, Q_A(UR), comme spécifié en 8.2.2.2d);

si le $V_{C,K}$(SS) associé aux demandes de DL-DATA qui viennent d'être confirmées n'était pas vide, alors la DLE peut annuler de telles demandes de retransmission et mettre le $V_{C,K}$(SS) correspondant à vide.

Si J=1, et *N* est supérieur à V_C(A), et soit

- $N < V_C(M)$, soit
- $N = V_C(M)$ et RSN $\le V_C(MS)$,

alors la DLE doit ajouter l'élément de rang RSN à l'ensemble $V_{C,N}$(SS); et si l'ensemble $V_{C,N}$(SS) était précédemment vide, alors la DLE doit

– annuler les temporisateurs de retransmission $T_{C,N}(SS)$ associés à la $N^{ème}$ DLSDU, ou $T_C(SS)$ associé au DLCEP;

– ajouter à la file d'attente de services non programmés de la DLE, Q(US), une référence à $Q_A(UR)$ du DLCEP de réception, afin d'assurer que la DLPDU demandée est envoyée à partir du DLCEP de réception

NOTE Q(US) n'a jamais besoin d'avoir plus de références à une $Q_A(UR)$ que le nombre des DLSDU attendant une transmission ou une retransmission.

c) Si le DLCEP est un DLCEP ORDERED ou un DLCEP éditeur, et $V_C(R)$ existe, et le sous-champ J des paramètres SD reçus n'est pas zéro, alors la DLE doit calculer

$$TEMP = (N_R(NDR) - V_C(R)) \text{ modulo } MOD_R \qquad (25)$$

1) Si $TEMP > (V_C(M) - V_C(R))$, ou
$TEMP = (V_C(M) - V_C(R))$ et RSN > $V_C(MS)$,
alors le résidu de numéro de séquence $N_R(NDR)$ reçu pour une DLSDU acquittée ou redemandé est obsolète ou non valide et doit être ignoré.

NOTE Le désordre des DLPDU émises, qui peut être occasionné dans cette condition, est possible lorsque le chemin de communications entre les DLE d'envoi et de réception comprend des ponts ou chemins de DL redondants ou de secours actifs.

2) Sinon, si 1) ne s'applique pas, alors la DLPDU reçue demande une retransmission d'un segment d'une DLSDU émise précédemment (J=1). Soit *N* égal à *TEMP* + $V_C(R)$. Si J=1 et *N* est supérieur à $V_C(R)$, alors la DLE doit ajouter l'élément de rang RSN à l'ensemble $V_{C,N}(SS)$; si l'ensemble $V_{C,N}(SS)$ était précédemment vide, alors la DLE doit ajouter à la file d'attente de services non programmés de la DLE, Q(US), une référence à $Q_A(UR)$ du DLCEP de réception, afin d'assurer que la DLPDU demandée est envoyée à partir du DLCEP de réception.

NOTE 3 Q(US) n'a jamais besoin d'avoir plus de références à un $Q_A(UR)$ que le nombre des DLSDU attendant une transmission ou une retransmission.

8.2.2.5.3 Traitement des sous-champs T et "truncated DL-time" des paramètres SD reçus

Si la liaison de réception du DLCEP est à un tampon, alors

a) si le DLCEP de réception a une classe d'opportunité de DL d'un expéditeur NONE, alors le statut d'opportunité, $V_B(TS)$ (voir 4.7.4.21), associé à l'écriture dans le tampon doit mis à FALSE;

b) autrement, lorsque a) ne s'applique pas, alors

1) Si la DLPDU reçue achemine le premier segment reçu de la DLSDU, alors

i) le statut d'opportunité associé du tampon, $V_B(TS)$ (voir 4.7.4.21), doit être mis égal au sous-champ T de la DLPDU reçue;

ii) si ce statut d'opportunité est TRUE, et si les paramètres SD incluaient l'heure de production, format K à M, alors l'heure de production du tampon, $V_B(TP)$ (voir 4.7.4.20), doit être inférée comme étant le temps de DL le plus récent dont le résidu dans le format de DLPDU négocié donnerait lieu au résidu de temps de DL acheminé par la DLPDU reçue;

NOTE Cette inférence fera que le temps de DL courant sera inféré lorsque le champ paramètres SD de la DLPDU reçue n'achemine pas des octets de temps de DL.

iii) le temps de DL de la réception de la DLPDU doit être utilisé comme étant le temps d'écriture dans le tampon, $V_B(TW)$ (voir 4.7.4.19).

2) si 1) ne s'applique pas, si bien que la DLPDU reçue acheminait un segment non reçu précédemment appartenant à une DLSDU à plusieurs segments pour laquelle au moins un segment a été reçu précédemment, alors

i) la valeur du sous-champ T de la DLPDU nouvellement reçue doit être soumise à une opération ET-logique dans le statut d'opportunité associé du tampon, $V_B(TS)$ (voir 4.7.4.21);

ii) tout temps de DL acheminé dans le paramètre SD de la DLPDU doit être ignoré;

iii) le temps de DL de la réception de la DLPDU doit être utilisé comme étant le temps d'écriture dans le tampon, V_B(TW) (voir 4.7.4.19).

8.2.2.6 Validation et traitement de données utilisateur reçues dans une DLPDU CA, DT ou ED

Dans ce qui suit, MUD est défini comme étant le nombre maximal d'octets de données utilisateur qui peuvent être acheminés dans une seule et même DLPDU CA, DT ou ED de priorité de DLCEP telle que spécifiée en 4.2.3.

Si une DLPDU CA, DT ou ED reçue a un champ "données utilisateur" non vide suivant son champ "paramètres SD", alors les données contenues constituent le ASN^{ème} segment, des TNS segments, de la $K^{ème}$ DLSDU, où ASN et TNS sont tous deux d'origine zéro, et K a la dernière valeur qui lui a été donnée en 8.2.2.5.1a)i), en 8.2.2.5.1b)2)ii)A), en8.2.2.5.1b)2)iii) ou en 8.2.2.5.1b)4).

La DLE de réception doit vérifier si la quantité (ASN × MUD plus la longueur des données utilisateur reçues) est inférieure ou égale à la taille maximale de DLSDU autorisée, $P_C(NP.M...M_R)$, qui a été négociée pour ce sens (de réception) d'une transmission de DLC. Si ASN est inférieur à TNS, alors la DLE doit aussi vérifier si la longueur des données utilisateur reçues est égale à MUD. Si l'une de ces exigences n'est pas respectée, alors la DLE doit déconnecter le DLCEP comme spécifié en 8.2.1.8, avec une cause de "déconnexion à l'initiative d'un fournisseur – mauvaise taille de DLSDU, état permanent":

Si l'élément numéro ASN est un élément de l'ensemble de variable Vc,k(MRS), alors

a) la DLE de réception doit déterminer si un enregistrement de données reçues a déjà été alloué à la $K^{ème}$ DLSDU, et dans la négative, doit allouer un enregistrement qui peut contenir au moins
min ($P_C(NP.M...M_R)$, TNS × MUD) octets de données utilisateur de DLS et doit associer cet enregistrement à la $K^{ème}$ DLSDU reçue;

b) la DLE de réception doit copier le champ "données utilisateur" de la DLPDU reçue vers cet enregistrement de données reçues, en commençant par l'octet ((ASN × MUD +1) de l'enregistrement de données reçues;

c) si l'ensemble variable $V_{C,K}$(MRS) est maintenant vide, alors la DLE de réception doit tenter de livrer la DLSDU comme spécifié en 8.2.2.7.

NOTE Il est possible pour une mise en œuvre de ce protocole de DL d'accomplir un réassemblage de DLSDU au moment de la réception de DLPDU. Une telle approche de mise en œuvre pourrait réduire au maximum le traitement des données utilisateur de DLS de la DLE.

8.2.2.7 Livraison d'une DLSDU complète qui a été entièrement reçue en un DLCEP

Soit K le numéro de séquence associé à la DLSDU entièrement reçue, tel que déterminé par 8.2.2.5.1 au moment auquel cette réception de DLSDU s'est achevée.

Si K est supérieur à V_C(L), et l'état courant V_C(ST) du DLCEP est DATA TRANSFER READY, alors les caractéristiques de livraison de données de réception du DLCEP et le type liaison de tampon ou file d'attente de réception déterminent la politique de livraison de DLSDU

a) Si la DLC est une DLC CLASSICAL, et K est supérieur à V_C(L) + 1, alors la DLSDU a été reçue dans le désordre et la DLE doit conserver, mais pas livrer, la DLSDU à cet instant.

b) Autrement,

   1) si un tampon de réception est lié au DLCEP, alors la DLE doit agir comme spécifié en 8.2.2.7.1;

   2) si une file d'attente de réception explicite est liée au DLCEP, alors la DLE doit agir comme spécifié en 8.2.2.7.2; ou

3) si aucune file d'attente de réception n'est liée au DLCEP, ce qui est la situation par défaut dans l'OSI, alors la mise en œuvre doit, en toute cohérence, soit
   i) agir comme spécifié en 8.2.2.7.3; soit
   ii) traiter ceci comme une variante du cas b), en utilisant une file d'attente de réception que la mise en œuvre a assignée dans ce but.

   NOTE Cette file d'attente peut être propre au DLCEP, ou peut être partagée avec d'autres DLCEP de priorité similaire en ce DLSAP.

8.2.2.7.1 Livraison à un tampon de réception

a) si le DLCEP de réception a un attribut d'opportunité de récepteur autre que NONE, la DLE doit mettre au temps de DL courant la variable V_B(TW) (voir 4.7.4.19), associée à l'écriture dans le tampon au temps de DL en cours.

   NOTE Il s'agit du temps local de réception, et non de l'heure de production distante.

b) La DLE doit livrer la DLSDU complète comme étant le nouveau contenu du tampon, et doit associer toutes les informations d'opportunité reçues dans la/les DLPDU d'acheminement.

c) La DLE doit rapporter une "indication" de DL-BUFFER-RECEIVED à l'utilisateur de DLS. Si la classe de DLCEP de réception était ORDERED, et la DLSDU reçue est un doublon d'une DLSDU précédemment reçue, alors l'attribut de la DLSDU doublon de la primitive "indication" de DL-BUFFER-RECEIVED doit spécifier TRUE; dans tous les autres cas, il doit spécifier FALSE.

   Si ce DLCEP a été spécifié comme un DLCEP de synchronisation au cours de l'établissement d'un ou plusieurs autres DLCEP locaux, alors la DLE doit enregistrer le temps de DL de l'accès au réseau, V_C(TNA), pour une utilisation dans les calculs d'opportunité de ce(s) DLCEP de référence.

d) Aucun accès en cours au contenu du tampon qui est incomplet au moment d'une livraison de DLSDU ne doit être affecté par la livraison de la DLSDU.

   NOTE Cette contrainte assure que chacun des accès à un tampon est logiquement atomique.

e) la DLE doit mettre V_C(L) à *K*; et doit annuler tous les temporisateurs $T_{C,N}$(RRS) qui peuvent exister, où *N* est inférieur ou égal à *K*. Il convient de libérer toutes les ressources de DLE consacrées à la réception et au réassemblage des DLSDU ayant des numéros de séquence inférieurs ou égaux à *K*;

8.2.2.7.2 Livraison à une file d'attente de réception

La DLE doit tenter d'abouter la DLSDU en entier, accompagnée de l'identification du DLCEP de réception, à la file d'attente de réception.

En cas d'échec, la DLE doit informer la gestion de DL locale de cette situation de file d'attente pleine.

NOTE Cette notification à la gestion de DL peut prendre la forme consistant à incrémenter un compteur des DLSDU rejetées.

En cas de réussite,

a) la DLE doit rapporter une "indication" de DL-DATA à l'utilisateur de DLS,
b) la DLE doit annuler le temporisateur $T_{C,K}$(RRS) s'il existe;
c) si la DLC est une DLC UNORDERED ou ORDERED, alors
   1) la DLE doit mettre V_C(L) à *K*;
   2) la DLE doit annuler tous les temporisateurs $T_{C,N}$(RRS) qui peuvent exister, où *N* est inférieur ou égal à *K*. Il convient de libérer toutes les ressources de DLE consacrées à la réception et au réassemblage des DLSDU ayant des numéros de séquence inférieurs ou égaux à *K*;
d) si la DLC est une DLC DISORDERED, et si *K* est égal à (V_C(L) +1), alors

1) la DLE doit mettre $V_C(L)$ à *K*;
2) si *K* est inférieur à $V_C(H)$, alors la DLE doit incrémenter *K*. Si l'ensemble variable $V_{C,K}(MRS)$ est vide, alors la DLE doit mettre $V_C(L)$ à *K* et doit répéter cette étape;
3) si la DLC est une DLC PEER, alors si la file d'attente de services non programmés d'adresse de DL de la DLE, Q(US), ne contient pas déjà une référence au DLCEP, alors la DLE doit abouter à cette Q(US) une référence au DLCEP, afin d'assurer qu'un acquittement de réception de DLSDU est envoyé à partir du DLCEP de réception;

e) si la DLC est une DLC CLASSICAL, alors
   1) la DLE doit mettre $V_C(L)$ à *K*;
   2) si *K* est inférieur à $V_C(H)$, alors la DLE doit incrémenter *K*. Si l'ensemble variable $V_{C,K}(MRS)$ est vide, alors la DLE doit répéter l'entière procédure de livraison de données (voir début de 8.2.2.7.2)) en utilisant la nouvelle valeur de *K*;
   3) si la DLC est une DLC PEER, alors si la file d'attente de services non programmés d'adresse de DL de la DLE, Q(US), ne contient pas déjà une référence au DLCEP, alors la DLE doit abouter à cette Q(US) une référence au DLCEP, afin d'assurer qu'un acquittement de réception de DLSDU est envoyé à partir du DLCEP de réception.

8.2.2.7.3 Livraison par défaut selon l'OSI

La DLE doit rapporter une indication de DL-DATA à l'utilisateur de DLS, acheminant la DLSDU reçue comme un paramètre; après quoi la DLE doit agir comme spécifié de 8.2.2.7.2b) à 8.2.2.7.2e).

8.2.2.8 réception d'une DLPDU DT adressée à un DLCEP

Dans ce qui suit, MUD est défini comme étant le nombre maximal d'octets de données utilisateur qui peuvent être acheminés dans une seule et même DLPDU DT de priorité de DLCEP telle que spécifiée en 4.2.3.

Lorsque la DLE reçoit une DLPDU DT adressée à un DLCEP de la DLE, la DLE doit accomplir la série d'actions suivante:

a) Si l'état de DLCEP, $V_C(ST)$, est WAITING-FOR-RESET-COMPLETION et la DLE attend seulement la réception d'une DLPDU DT au DLCEP, alors la DLE doit changer l'état du DLCEP en DATA-TRANSFER-READY.
b) La DLE doit valider que
   1) la priorité de la DLPDU DT reçue est telle que prévue;
   2) dans une DLPDU DT reçue adressée à tous les abonnés d'un DLCEP PUBLISHER, longueur de l'adresse de DL de l'éditeur est supérieure ou égale à celle qui est attendue;
   3) dans une DLPDU DT reçue adressée à un DLCEP PUBLISHER, le nombre d'adresses de DL est tel que prévu;
   4) dans une DLPDU DT reçue adressée à un DLCEP PEER
      i) la longueur et le nombre d'adresse(s) de DL sont tels que prévus (seulement LONG, ou seulement SHORT, ou soit SHORT ou VERY-SHORT à la discrétion de l'expéditeur);
      ii) lorsque deux adresses sont attendues, la seconde adresse de DL de la DLPDU est l'adresse de DLCEP de l'homologue distant du DLCEP adressé par la première adresse de DL de la DLPDU DT.

   Si cette validation échoue, alors

   – si le DLCEP est DLCEP PEER, la DLE doit déconnecter le DLCEP de la DLC comme spécifié en 8.2.1.8 avec une cause de "déconnexion à l'initiative d'un fournisseur – mauvais format ou paramètres de DLPDU, état permanent";
   – sinon la DLE doit rejeter la DLPDU DT.

c) Si le DLCEP est un DLCEP PEER ou SUBSCRIBER dont l'attribut d'activité résiduelle négocié est TRUE, alors la DLE doit redémarrer le T_C(RAM) du DLCEP comme spécifié en 8.2.2.14.

NOTE Voir la permission au début de l'Article 8 en ce qui concerne l'utilisation de V(NRC) versus V(MRC).

d) Si le nombre restant d'octets dans la DLPDU est inférieur au nombre d'octets dans le format de paramètres SD négocié pour le sens de transmission "expéditeur vers destinataire" applicable, alors

- si le DLCEP est un DLCEP PEER ou SUBSCRIBER, alors la DLE doit déconnecter le DLCEP de la DLC comme spécifié en 8.2.1.8 avec une cause de "déconnexion à l'initiative d'un fournisseur – mauvais format ou paramètre de DLPDU, état permanent";
- sinon la DLE doit rejeter la DLPDU DT.

Autrement, la DLE doit analyser et traiter les paramètres SD du format applicable à partir de ces octets restants comme spécifié en 8.2.2.5; et si le nombre restant d'octets dans la DLPDU, après les paramètres SD, est supérieur à zéro, alors la DLE doit traiter ces données utilisateur comme spécifié en 8.2.2.6 et éventuellement en 8.2.2.7.

8.2.2.9 Réception d'une DLPDU CA, CD ou ED

Lorsque la DLE reçoit une DLPDU CA, CD ou ED adressée à un DLCEP de la DLE, la DLE doit accomplir la série d'actions suivante:

a) Si le DLCEP est un DLCEP PEER, alors la DLE doit valider que
   1) la longueur et le nombre d'adresses de DL sont tels que prévus;
   2) quand elle est présente et le DLCEP de réception est un DLCEP PEER, la deuxième adresse de DL de la DLPDU est l'adresse de DLCEP de l'homologue distant du DLCEP de réception;
   3) la priorité de DLL du DLCEP est égale à la priorité spécifiée dans la DLPDU reçue.

   Si cette validation échoue, alors la DLE doit déconnecter le DLCEP de la DLC comme spécifié en 8.2.1.8 avec une cause de "déconnexion à l'initiative d'un fournisseur – mauvais format ou paramètres de DLPDU, état permanent".

b) Si le DLCEP est un DLCEP PEER dont l'attribut d'activité résiduelle négocié est TRUE, alors la DLE doit redémarrer le T_C(RAM) du DLCEP avec une durée basée sur le retard maximal de confirmation spécifié par l'utilisateur pour les primitives des services DL-CONNECT, DL-RESET and DL-SUBSCRIBER-QUERY. Si la valeur spécifiée était autre que UNLIMITED, il convient que la durée de ce temporisateur soit ; autrement, il convient que la durée soit $\frac{60\ \text{s}}{V(NRC)+1}$. La gestion de DL peut remplacer ces durées préférentielles.

NOTE Voir la permission au début de l'Article 8 en ce qui concerne l'utilisation de V(NRC) versus V(MRC).

c) Si la DLPDU reçue est une DLPDU CD, alors
   1) si le DLCEP est un DLCEP PEER, et si le nombre restant d'octets dans la DLPDU n'est pas zéro, et n'est pas égal au nombre d'octets dans le format de paramètres SD négocié pour le sens de transmission "expéditeur vers destinataire" applicable, la DLE doit déconnecter le DLCEP de la DLC comme spécifié en 8.2.1.8 avec une cause de "déconnexion à l'initiative d'un fournisseur – mauvais format ou paramètres de DLPDU, état permanent";
   2) autrement, lorsque 1) ne s'applique pas, alors
      i) si le nombre restant d'octets dans la DLPDU est égal au nombre d'octets dans le format de paramètres SD négocié pour le sens de transmission "expéditeur vers destinataire" applicable, alors la DLE doit analyser et traiter le format de paramètres SD applicable à partir de ces octets restants comme spécifié en 8.2.2.5;

ii) lorsque les caractéristiques du DLCEP de réception sont ORDERED, et le DLCEP est lié à un tampon d'envoi, et la DLSDU dans le tampon était trop grande à émettre intégralement dans la réponse immédiate à la DLPDU CD reçue, alors la DLE doit

A) pour le K approprié, modifier la variable $V_{C,K}$(SS), en fonction de la longueur de la $K^{\text{ème}}$ DLSDU, afin d'indiquer que tous les segments de la DLSDU à l'exception du segment zéro (0) requièrent une transmission;

B) abouter à la file d'attente de services non programmés de la DLE, Q(US), une référence à la Q_A(UR) du DLCEP. La référence signale la nécessité d'émettre le contenu du tampon d'envoi tampon qui est lié au DLCEP;

d) Si la DLPDU reçue est une DLPDU CA ou ED, alors

1) si le nombre restant d'octets dans la DLPDU est supérieur ou égal au nombre d'octets dans le format de paramètres SD négocié pour le sens de transmission "expéditeur vers destinataire" applicable, alors, la DLE doit

i) doit analyser et traiter les paramètres SD du format applicable à partir de ces octets restants comme spécifié en 8.2.2.5;

ii) puis traiter tous les éventuels octets restants de la DLSDU comme étant des données utilisateur de DLS reçue comme spécifié en 8.2.2.6 et 8.2.2.7;

2) if 1) ne s'applique pas et le DLCEP est un DLCEP PEER, alors la DLE doit déconnecter le DLCEP de la DLC comme spécifié en 8.2.1.8 avec une cause de "déconnexion à l'initiative d'un fournisseur – mauvais format ou paramètre de DLPDU, état permanent".

8.2.2.10 Démarrage, annulation et expiration du temporisateur T_U(MCD) sur une demande de DL-DATA

Le temporisateur T_U(MCD) doit être démarré lorsque l'utilisateur de DLS émet la demande de DL-DATA correspondante. Il doit être annulé

– en un DLCEP abonné, ou en un DLCEP homologue CLASSICAL ou DISORDERED ou en un DLCEP homologue ou éditeur ORDERED lorsque V_C(R) n'existe pas, lorsque la DLE émet une primitive "confirm" de DL-DATA correspondante;

– en un DLCEP éditeur CLASSICAL ou DISORDERED ou en un DLCEP homologue ou éditeur ORDERED lorsque V_C(R) existe, lorsque la DLE trouve nécessaire d'incrémenter V_C(R) et la valeur qui en résulte de V_C(R) est égal au numéro de séquence assigné à la DLSDU correspondante.

Si le temporisateur T_U(MCD) expire à la suite d'une demande de DL-DATA, alors la DLE doit

a) incrémenter V_C(R);

NOTE Après l'incrémentation, il convient que V_C(R) ait une valeur égale au numéro de séquence de la DLSDU associée à la demande expirée.

b) supprimer la demande de la file d'attente de demandes utilisateur de l'adresse de DLCEP d'envoi, Q_A(UR), et mettre fin au traitement de la demande;

c) maintenir toute statistique de gestion de DL appropriée;

NOTE La détermination du minimum de statistique nécessaire est pour étude ultérieure.

d) si une primitive "confirm" de DL-DATA pour la demande n'a pas encore été émise, alors

1) initier une primitive "confirm" de DL-DATA rapportant "échec à l'initiative d'un fournisseur – temporisation de demande";

2) si le choix des caractéristiques de livraison de données de DLCEP d'envoi est DISORDERED ou CLASSICAL, alors lancer une réinitialisation de la DLC comme spécifié en 8.2.2.19.

8.2.2.11 Démarrage, annulation et expiration du temporisateur $T_{C,K}$(SS)

NOTE Ce temporisateur est utilisé seulement par les DLCEP PEER dont les caractéristiques de livraison de données d'envoi sont DISORDERED ou CLASSICAL.

Le temporisateur $T_{C,K}$(SS) doit être relancé chaque fois qu'une DLPDU contenant tout ou partie de la $DLSDU_K$ est émise et $V_{C,K}$(SS) est vide; il doit être annulé chaque fois que la valeur de $V_{C,K}$(SS) est supérieure ou égale à K ou chaque fois que $V_{C,K}$(SS) devient non vide (voir 8.2.2.5.2b)).

La durée de ce temporisateur doit être basée sur le retard maximal de confirmation spécifié par un utilisateur local pour les primitives de DL-DATA. Si la valeur spécifiée était autre que UNLIMITED, alors il convient que la durée de ce temporisateur soit comprise entre 25 % et 50 % de $\frac{V_c(NP) \times MCD\_D}{V(NRC)+2}$; autrement il convient que la durée soit comprise entre 25 % et 50 % de $\frac{60s}{V(NRC)+2}$. La gestion de DL peut remplacer ces durées préférentielles.

NOTE Voir la permission au début de l'Article 8 en ce qui concerne l'utilisation de V(NRC) versus V(MRC).

Si le temporisateur $T_{C,K}$(SS) expire, alors la DLE doit

a) modifier la variable $V_{C,K}$(SS), en fonction de la longueur de la $K^{ème}$ DLSDU, afin d'indiquer que le dernier segment de la $K^{ème}$ DLSDU nécessite une retransmission

b) abouter à la file d'attente de services non confirmés de la DLE, Q(US), une référence à la Q_A(UR) du DLCEP, afin de programmer une retransmission de la DLSDU non acquittée;

c) maintenir toute statistique de gestion de DL appropriée.

NOTE 3 La détermination du minimum de statistique nécessaire est pour étude ultérieure.

8.2.2.11.1 Utilisation du temporisateur T_C(SS) simplifié

Lorsque la permission en 4.7.4.9.1 est utilisée, les règles suivantes s'appliquent:

- Le temporisateur T_C(SS) doit être relancé, chaque fois qu'une DLPDU contenant tout ou partie de la $DLSDU_K$ est émise et $V_{C,K}$(SS) est vide. Le temporisateur doit être relancé chaque fois qu'il n'est pas en marche et V_C(A) est inférieur à V_C(M); il doit être annulé chaque fois que V_C(A) est égal à V_C(M) ou chaque fois que $V_{C,K}$(SS) devient non vide en raison de la réception pour retransmission (voir 8.2.2.5.2b)).
- La durée de ce temporisateur doit être basée sur le retard maximal de confirmation spécifié par un utilisateur local pour les primitives de service DL-DATA. Si la valeur spécifiée était autre que UNLIMITED, alors il convient que la durée de ce temporisateur soit entre 25 % et 50 % de $\frac{V_c(NP) \times MCD\_D}{V(NRC)+2}$; autrement, il convient que la durée soit entre 25 % et 50 % de $\frac{60s}{V(NRC)+2}$. La gestion de DL peut remplacer ces durées préférentielles.

  NOTE Voir la permission au début de l'Article 8 en ce qui concerne l'utilisation de V(NRC) versus V(MRC).
- Si le temporisateur T_C(SS) expire, alors la DLE doit
  a) modifier la variable $V_{C,K}$(SS), en fonction de la longueur de la $DLSDU_K$ non acquitté ayant le plus petit numéro de séquence, afin d'indiquer que le(s) dernier(s) segment(s) de cette DLSDU nécessite(nt) une retransmission;
  b) abouter à la file d'attente de services non confirmés de la DLE, Q(US), une référence à la Q_A(UR) du DLCEP, afin de programmer une retransmission de la DLSDU non acquittée;
  c) maintenir toute statistique de gestion de DL appropriée.

8.2.2.12 Démarrage, annulation et expiration du temporisateur $T_{C,K}$(RRS)

NOTE Ce temporisateur est utilisé par les DLCEP d'abonné DISORDERED ou CLASSICAL, et en option, par des DLCEP homologue ou éditeur ORDERED.

Le temporisateur $T_{C,K}$(RRS) doit être lancé chaque fois qu'il n'est pas en marche et une DLPDU demandant une retransmission d'un ou plusieurs segments de la $K^{ème}$ DLSDU est émise. Il doit être annulé chaque fois que tous les segments de la $K^{ème}$ DLSDU sont reçus.

La durée de cette temporisateur doit être basée sur le retard maximal de confirmation spécifié par un utilisateur distant pour les primitives de service DL-DATA et acheminée dans une DLPDU EC précédemment reçue en provenance du DLCEP d'envoi. Si la valeur spécifiée était autre que UNLIMITED, alors il convient que la durée de ce temporisateur soit comprise entre 25 % et 50 % de $\frac{V_c(NP) \times MCD\_D}{V(NRC)+2}$; autrement il convient que la durée soit comprise entre 25 % et 50 % de $\frac{60s}{V(NRC)+2}$. La gestion de DL peut remplacer ces durées préférentielles.

NOTE Voir la permission au début de l'Article 8 en ce qui concerne l'utilisation de V(NRC) versus V(MRC).

Si le temporisateur $T_{C,K}$(RSS) expire, alors la DLE doit

a) mettre $V_{C,K}$(RRS) à la valeur alors courante de $V_{C,K}$(MRS), afin d'indiquer que les segments spécifiés de la $K^{ème}$ DLSDU nécessitent une retransmission.
b) effectuer une vérification pour détecter une référence au DLCEP sur la file d'attente de services non programmés de la DLE, Q(US), et si elle n'en trouve pas, alors abouter une référence au DLCEP à la file d'attente de services non programmés, Q(US), afin de programmer une retransmission des segments de DLSDU absents;
c) maintenir toute statistique de gestion de DL appropriée.

NOTE La détermination du minimum de statistique nécessaire est pour étude ultérieure.

8.2.2.13 Démarrage, annulation et expiration du temporisateur T_C(RAS)

NOTE Ce temporisateur est utilisé seulement par les DLE PUBLISHER ou les DLE PEER d'envoi dont les caractéristiques de livraison de données d'envoi sont ORDERED, DISORDERED ou CLASSICAL, et il est seulement requis lorsque l'établissement de la DLC demandait une activité résiduelle sur la DLC dans ce sens (expéditeur vers destinataire) de transfert de données.

Lorsque cela est applicable (voir 4.7.4.16 pour les conditions d'utilisation du temporisateur), le temporisateur T_C(RAS) doit être lancé

a) en un DLCEP homologue DISORDERED ou CLASSICAL, chaque fois qu'il n'est pas en marche et lorsque V_C(A) est égal à (V_C(N) – 1);
b) en un DLCEP PUBLISHER, ou DLCEP d'envoi ORDERED PEER, chaque fois qu'il n'est pas en marche et lorsque V_C(M) est égal (V_C(N) – 1) et $V_{C,V_C(M)}$(SS) est vide.

La durée de ce temporisateur doit être basée sur le retard maximal de confirmation spécifié par un utilisateur pour les primitives "request" ou "response" de DL-CONNECT. Si la valeur spécifiée était autre que UNLIMITED, alors il convient que la durée de ce temporisateur soit comprise entre 70 % et 95 % de $\frac{V_c(NP) \times MCD\_D}{V(NRC)+2}$; autrement il convient que la durée soit comprise entre 70 % et 95 % de $\frac{60s}{V(NRC)+2}$. La gestion de DL peut remplacer ces durées préférentielles.

NOTE Voir la permission au début de l'Article 8 en ce qui concerne l'utilisation de V(NRC) versus V(MRC).

Il doit être annulé chaque fois que V_C(A) n'est pas égal à (V_C(N) – 1). Si le temporisateur T_C(RAS) expire, la DLE doit effectuer une vérification pour détecter une référence au DLCEP sur la file d'attente de services non programmés de la DLE, Q(US), et si elle n'en trouve pas, alors abouter une référence au DLCEP à la file d'attente de services non programmés de la DLE, Q(US), afin de programmer la retransmission vers le(s) DLCEP distant(s).

NOTE La transmission résultante peut consister en une DLPDU DT avec un champ "données utilisateur" vide, ou une DLPDU CA.

8.2.2.14 Démarrage, annulation et expiration du temporisateur T_C(RAM)

NOTE Ce temporisateur est utilisé seulement par les DLE SUBSCRIBLER ou les DLE PEER de réception dont les caractéristiques de livraison de données de réception sont ORDERED, DISORDERED ou CLASSICAL, et il est seulement requis lorsque l'établissement de la DLC demandait une activité résiduelle sur la DLC dans ce sens (destinataire à partir d'expéditeur) de transfert de données.

Lorsque cela est applicable (voir 4.7.4.17 pour les conditions d'utilisation du temporisateur), le temporisateur T_C(RAM) doit fonctionner en continu. Il doit être relancé chaque fois qu'une DLPDU quelconque est reçue sur le DLCEP.

La durée de ce temporisateur doit être basée sur le retard maximal de confirmation spécifié par un utilisateur distant pour les primitives "request" ou "response" de DL-CONNECT et acheminée dans une DLPDU EC précédemment reçue en provenance du DLCEP d'envoi. Si la valeur spécifiée était UNLIMITED, il convient que la durée de ce temporisateur soit 60 s. Autrement, il convient que la durée soit V_C(NP).MCD_CRS. La gestion de DL peut remplacer ces durées préférentielles.

Si le temporisateur T_C(RAM) expire, alors la DLE doit lancer une réinitialisation du DLCEP comme spécifié en 8.2.2.19.

8.2.2.15 Réception d'une primitive "request" de DL-RESET

Lorsque la DLE reçoit à une demande de DL-RESET issue d'un utilisateur de DLS pour un DLCEP dans l'état DATA-TRANSFER-READY alors

a) si la classe de DLCEP n'est pas PEER, la DLE doit émettre une primitive "confim" avec un statut "échec —cause non spécifiée" pour chaque primitive «request» de DL-DATA en cours sur le DLCEP local, et doit incrémenter V_C(R) pour chacune de ces primitives confirmées;
b) la DLE doit
   1) mettre V_C(M) à V_C(N) -1; mettre V_C(MS) et V_C(HS) à zéro;
   2) libérer toutes les ressources DLE consacrées à la réception et au réassemblage des DLSDU;
   3) annuler tous les temporisateurs $T_{C,K}$(SS) et $T_{C,K}$(RRS).
c) si la classe de DLCEP est PEER, alors la DLE doit
   1) noter qu'à la fois
      i) une primitive "confirm" de DL-RESET;
      ii) la réception d'une DLPDU RC ne demandant pas une DLPDU RC de réponse issue la part du DLCEP homologue distant

      seront requises;
   2) coder une DLPDU RC, adressée au DLCEP homologue distant et demandant une DLPDU RC de réponse, avec la cause spécifiée par l'utilisateur de DLS, utilisant V_C(L) +1 pour le numéro de séquence NDR et V_C(M) pour le numéro de séquence NDS;
   3) programmer la DLPDU pour la transmission à la priorité de DLC telle que spécifiée en 8.4.5;
   4) lancer une temporisateur tel que spécifié en 8.2.1.2b);
   5) changer l'état du DLCEP, V_C(ST), en WAITING-FOR-RESET-COMPLETION, tout en attendant une DLPDU RC.
d) si la classe de DLCEP est ^PUBLISHER, alors la DLE doit
   1) coder une DLPDU RC, adressée à tous les DLCEP d'abonné de la DLC, ne demandant pas une DLPDU RC de réponse, avec la cause spécifiée par l'utilisateur de DLS,

utilisant zéro (0) pour le numéro de séquence NDR et V_C(M) pour le numéro de séquence NDS;

2) mettre V_C(A) ou V_C(R), selon le cas, à V_C(N) –1;
3) programmer la DLPDU pour la transmission à la priorité de DLC telle que spécifiée en 8.4.5;
4) rapporter une confirmation de -RESET à l'utilisateur de DLS local avec un paramètre de statut de «succès».

e) Si la classe de DLCEP est SUBSCRIBER, alors la DLE doit rapporter une confirmation de -RESET à l'utilisateur de DLS local avec un paramètre de statut de "succès"

8.2.2.16 Réception d'une primitive "response" de DL-RESET

Lorsque la DLE reçoit une réponse de DL-RESET issue d'un utilisateur de DLS pour un DLCEP actif dont l'état est WAITING-FOR-RESET-COMPLETION, alors

a) si la classe de DLCEP est PEER, et que l'ensemble d'événements au DLCEP pour lequel la DLE est en attente est
   1) seulement une réponse DL-Reset, alors la DLE doit
      i) coder une DLPDU DT (avec ou sans données) issue du DLCEP pour notifier à ses homologues la réception réussie de la DLPDU RC de confirmation;
      ii) programmer la DLPDU pour la transmission à la priorité de DLC telle que spécifiée en 8.4.5;
      iii) vider de nouveau les files d'attente internes du DLCEP;
      iv) changer l'état du DLCEP, V_C(ST), en DATA-TRANSFER-READY.
      v) rapporter une indication de DL-RESET-COMPLETED à l'utilisateur de DLS local;
   2) à la fois
      i) une réponse de DL-Reset;
      ii) la réception d'une DLPDU RC ne demandant pas une réponse issue du DLCEP homologue,

      alors la DLE doit attendre cette DLPDU RC;
   3) à la fois
      i) une réponse de DL-Reset;
      ii) la réception ultérieure d'une DLPDU DT issue du DLCEP homologue

      alors, la DLE doit
      A) coder une DLPDU RC, adressée au DLCEP homologue distant et ne demandant pas une DLPDU RC de réponse, avec une cause "réinitialisation, origine inconnue – cause non spécifiée" utilisant V_C(L) +1 pour le numéro de séquence NDR et V_C(M) pour le numéro de séquence NDS;
      B) programmer la DLPDU pour la transmission à la priorité de DLC telle que spécifiée en 8.4.5;
      C) lancer une temporisateur tel que spécifié en 8.2.1.2b);
      D) noter qu'une indication de DL-RESET-COMPLETED à l'utilisateur de DLS local est toujours requise;
      E) attendre une DLPDU DT.

b) Si la classe de DLCEP de réponse est PUBLISHER ou SUBSCRIBER, alors la DLE doit
   1) changer l'état du DLCEP, V_C(ST), en DATA-TRANSFER-READY.
   2) vider de nouveau les files d'attente internes du DLCEP;
   3) rapporter une indication de DL-RESET-COMPLETED à l'utilisateur de DLS local;

8.2.2.17 Réception d'une DLPDU RC

Lorsque la DLE reçoit une DLPDU RC, la DLE doit déterminer la version du protocole de DL utilisé, tel que spécifié dans la DLPDU RC reçue, et doit interpréter les autres paramètres-RC de la DLPDU en conséquence.

a) Si la classe de DLCEP du DLCEP est PEER, ou SUBSCRIBER, lorsque la DLE reçoit une DLPDU RC spécifiant qu'il convient de réinitialiser le DLCEP, alors la DLE doit agir selon l'état du DLCEP, $V_C(ST)$:

   1) waiting-for-EC-DLPDU ou waiting-for-connect-response ou waiting-for-connect-completion:

      La DLE doit

      i) déconnecter le DLCEP comme spécifié en 8.2.1.8 avec une cause de "rejet de connexion – état de DLCEP incohérent, – état permanent".

   2) data-transfer-ready:

      La DLE doit

      i) rapporter une indication de DL-RESET à l'utilisateur de DLS local avec un paramètre de cause égal à celui contenu dans la DLPDU RC reçue;

      ii) si la classe de DLCEP est PEER, et $V_C(A)$ est inférieur au numéro de séquence sous-entendu par le sous-champ NDR des paramètres RC de la DLPDU RC reçue (qui est le plus petit numéro de séquence ≥ $V_C(A)$ dont le module serait NDR), alors la DLE doit traiter les demandes de DL-DATA en cours sur le DLCEP local comme spécifié en 8.2.2.5.2, avec le NDR tel que reçu dans la DLPDU RC, J=0 et K=1;

      iii) si la classe de DLCEP du DLCEP est PEER, la DLE doit émettre une primitive "confirm" avec un statut de "échec – réinitialisation ou déconnexion" pour chaque primitive «request» de DL-DATA en cours sur le DLCEP local, et doit incrémenter $V_C(R)$ pour chacune de ces primitives confirmées; et mettre set $V_C(A)$ ou $V_C(R)$, selon le cas, à $V_C(N)$ –1;

      iv) mettre $V_C(L)$ et $V_C(H)$ au numéro de séquence sous-entendu par le sous-champ NDS des paramètres RC de la DLPDU RC reçue (qui est le plus petit numéro de séquence ≥ $V_C(H)$ dont le module serait NDS);

      v) mettre $V_C(M)$ à $V_C(N)$ -1; mettre $V_C(MS)$ et $V_C(HS)$ à zéro;

      vi) libérer toutes les ressources DLE consacrées à la réception et au réassemblage des DLSDU;

      vii) annuler tous les temporisateurs $T_{C,K}(SS)$ et $T_{C,K}(RRS)$;

      viii) changer l'état de DLCEP, $V_C(ST)$, en WAITING-FOR-RESET-COMPLETION;

      ix) attendre une réponse de DL-RESET plus, si la classe du DLCEP est PEER, la réception ultérieure d'une DLPDU DT indiquant l'achèvement de réinitialisation d'homologue;

   3) waiting-for-reset-completion:

      i) si la classe de DLCEP est PEER, et $V_C(A)$ est inférieur au numéro de séquence sous-entendu par le sous-champ NDR des paramètres RC de la DLPDU RC reçue (qui est le plus petit numéro de séquence ≥ $V_C(A)$ dont le module serait NDR), alors la DLE doit traiter les demandes de DL-DATA en cours sur le DLCEP local comme spécifié en 8.2.2.5.2, avec le NDR tel que reçu dans la DLPDU RC, J=0 et K=1;

      ii) si la classe de DLCEP du DLCEP est PEER, la DLE doit émettre une primitive "confirm" avec un statut de "échec – réinitialisation ou déconnexion" pour chaque primitive «request» de DL-DATA en cours sur le DLCEP local, et doit incrémenter $V_C(R)$ pour chacune de ces primitives confirmées; et mettre $V_C(A)$ ou $V_C(R)$, selon le cas, à $V_C(N)$ –1;

iv) La DLE doit mettre $V_C(L)$ et $V_C(H)$ au numéro de séquence sous-entendu par le sous-champ NDS des paramètres RC de la DLPDU RC reçue (qui est le plus petit numéro de séquence ≥ $V_C(H)$ dont le module serait NDS);

iv) Si la classe de DLCEP SUBSCRIBER, la DLE doit rejeter la DLPDU RC reçue.

v) Si la classe de DLCEP est PEER, et la DLE attend une primitive "response" de DL-RESET au DLCEP, alors

– si la DLPDU RC reçue spécifie qu'une DLPDU RC de réponse est requise, alors la DLE doit noter qu'une DLPDU RC est requise, sinon la DLE doit noter qu'une DLPDU DT est requise.

vi) Si la classe de DLCEP est PEER, et la DLE n'attend pas une primitive «response» de DL-RESET au DLCEP, et la DLPDU RC reçue a demandé une réponse, alors la DLE doit

A) coder une DLPDU RC, adressée au DLCEP homologue distant et ne demandant pas une DLPDU RC de réponse, avec une cause "réinitialisation, origine inconnue – cause non spécifiée" utilisant $V_C(L)$ +1 pour le numéro de séquence NDR et $V_C(M)$ pour le numéro de séquence NDS;

B) programmer la DLPDU pour la transmission à la priorité de DLC telle que spécifiée en 8.4.5;

C) lancer une temporisateur tel que spécifié en 8.2.1.2b);

D) si la primitive de "reset-completion" (achèvement de réinitialisation) attendue par l'utilisateur de DLS au DLCEP est une "confirm" de DL-RESET, alors la DLE doit noter qu'une DLPDU RC est requise, autrement la DLE doit noter qu'une DLPDU DT est requise.

vii) Si la classe de DLCEP est PEER, et la DLE n'attend pas une primitive «response» de DL-RESET au DLCEP, et la DLPDU RC reçue n'a pas demandé une réponse, alors la DLE doit

A) coder une DLPDU DT (avec ou sans données) issue du DLCEP pour notifier à ses homologues la réception réussie de la DLPDU RC de confirmation;

B) programmer la DLPDU pour la transmission à la priorité de DLC telle que spécifiée en 8.4.5;

C) vider de nouveau les files d'attente internes du DLCEP;

D) changer l'état du DLCEP, $V_C(ST)$, en DATA-TRANSFER-READY.

E) si la primitive d'achèvement de réinitialisation attendue par l'utilisateur de DLS au DLCEP est une "confirm' de DL-RESET, alors la DLE doit rapporter à l'utilisateur de DLS local une "confirm" de DL-RESET, sinon la DLE doit rapporter une "indication" de DL-RESET-COMPLETED à l'utilisateur de DLS local.

viii) Lorsque aucun point de ii) à v) ne s'applique, alors

A) la DLE doit coder une DLPDU DC comme spécifié en 6.2 et 7.2, avec une cause de "déconnexion à l'initiative d'un fournisseur ou rejet de connexion – cause non spécifiée", et programmer la DLPDU DC pour la transmission à la priorité TIME-AVAILABLE telle que spécifiée en 8.4.5;

B) La DLPDU DC doit avoir à la fois les adresses de destination et de source (voir en 6.2.1 les formats 1L et 1S)), l'adresse de destination doit être identique à la première adresse de DLCEP source de la DLPDU RC reçue, et l'adresse source doit être identique à l'adresse de DL de destination de cette DLPDU RC reçue.

4) Dans tous les autres cas, la DLE doit agir comme en a)3)v).

b) Si la classe de DLCEP du DLCEP est PUBLISHER, alors la DLE doit agir comme en a)3)v).

8.2.2.18 Expiration du temporisateur $T_U(MCD)$ sur une "request" ou "indication" de DL-RESET

Si le temporisateur $T_U(MCD)$ expire sur une demande de DL-REST ou une réinitialisation lancée par la DLE, alors

a) s'il s'agit de la (V(NRC)+1)^{ème} expiration consécutive sans retourner à l'état DATA-TRANSFER-READY, alors la DLE doit mettre fin au traitement de réinitialisation de DLCEP et doit déconnecter le DLCEP de la DLC comme spécifié en 8.2.1.8 avec une cause de “déconnexion à l'initiative d'un fournisseur – temporisation, état permanent”;

NOTE Voir la permission au début de l'Article 8 en ce qui concerne l'utilisation de V(NRC) versus V(MRC).

b) autrement, si a) ne s'applique pas, la DLE doit
   1) relancer le temporisateur avec la même période que la fois précédente;
   2) reprogrammer la DLPDU RC (qui a d'abord été envoyée juste après avoir quitté l'état DATA-TRANSFER-READY) pour la retransmission à la priorité de la DLC telle que spécifiée en 8.4.5.

8.2.2.19 Réinitialisation à l'initiative de la DLE

Lorsque la DLE initie une réinitialisation, alors

a) la DLE doit changer l'état du DLCEP, V_C(ST), en WAITING-FOR-RESET-COMPLETION;
b) La DLE doit noter le besoin pour
   1) une réception d'une réponse de DL-RESET issue de l'utilisateur de DLS;
   2) si la classe de DLCEP est PEER, une réception ultérieure d'une DLPDU RC indiquant l'achèvement de la réinitialisation d'homologue;
c) si la classe de DLCEP du DLCEP est PEER, alors la DLE doit lancer un temporisateur de demandes utilisateur T_U(MCD) avec une durée basée sur le retard maximal de confirmation spécifié par un utilisateur pour les primitives des services DL-CONNECT, DL-RESET et DL-SUBSCRIBER-QUERY. Si la valeur spécifiée était autre que UNLIMITED, il convient que la durée de ce temporisateur soit $\frac{V_c(NP)\times MCD\_CRS}{V(NRC)+1}$; autrement, il convient que la durée soit $\frac{60s}{V(NRC)+1}$. La gestion de DL peut remplacer ces durées préférentielles

   NOTE Voir la permission au début de l'Article 8 en ce qui concerne l'utilisation de V(NRC) versus V(MRC).

d) si la classe de DLCEP du DLCEP n'est pas PEER, la DLE doit émettre une primitive "confirm" avec un statut de "échec – réinitialisation ou déconnexion" pour chaque primitive «request» de DL-DATA en cours sur le DLCEP local, doit incrémenter V_C(R) pour chaque primitive confirmé de ce type, et doit mettre V_C(A) ou V_C(R), selon le cas, à V_C(N) –1;
e) la DLE doit
   1) mettre V_C(M) à V_C(N) -1; mettre V_C(MS) et V_C(HS) à zéro;
   2) libérer toutes les ressources DLE consacrées à la réception et au réassemblage des DLSDU;
   3) annuler tous les temporisateurs $T_{C,K}$(SS) et $T_{C,K}$(RRS);
f) la DLE doit émettre une primitive "indication" DL-RESET vers l'utilisateur de DLS local, spécifiant le motif de la réinitialisation;
g) si la classe de DLCEP PEER, alors la DLE doit
   1) coder une DLPDU RC, avec la même cause que celle donnée à l'utilisateur de DLS local et demandant une DLPDU RC de réponse, utilisant V_C(L) +1 pour le numéro de séquence NDR et V_C(M) pour le numéro de séquence NDS;
   2) placer en file d'attente cette DLPDU pour la transmission à la priorité de DLC telle que spécifiée en 8.4.5;
h) si la classe de DLCEP est ^PUBLISHER, alors la DLE doit
   1) coder une DLPDU RC, adressée à tous les DLCEP d'abonné de la DLC, ne demandant pas une DLPDU RC de réponse, avec la même cause que celle donnée à l'utilisateur de DLS, utilisant zéro (0) pour le numéro de séquence NDR et V_C(M) pour le numéro de séquence NDS, et

2) programmer la DLPDU pour la transmission à la priorité de DLC telle que spécifiée en 8.4.5.

8.2.3 Fonctionnement du service "DLC subscriber query"

Les primitives du service d'interrogation d'abonné de DLC sont "request" et "confirm" de DL-SUBSCRIBER-QUERY.

8.2.3.1 Réception d'une primitive "request" de DL-SUBSCRIBER-QUERY

Si la demande est acceptée, comme indiqué par un statut retourné de "demande acceptée" pour la demande de DL-SUBSCRIBER-QUERY, alors à la suite de l'achèvement de la demande, soit avec succès, soit après l'échec, la DLE doit émettre une "confirm" de DL-SUBSCRIBER-QUERY en ce DLCEP, acheminant le statut de la demande à l'utilisateur de DLS.

La demande de DL-SUBSCRIBER-QUERY doit être placée dans la file d'attente de demandes utilisateur d'adresse de DL NODE de la DLE, à savoir Q_N(UR), qui est une file d'attente implicite (commandée par une DLE).

a) Si cette file d'attente est pleine, alors la DLE doit immédiatement retourner la "confirm" de DL-LISTENER-QUERY correspondante indiquant la cause de l'échec – "échec à l'initiative d'un fournisseur – file d'attente pleine".
b) Si a) ne s'applique pas, alors la DLE doit lancer un temporisateur de demandes d'utilisateur T_U(MCD) avec une durée basée sur le retard maximal de confirmation spécifié par un utilisateur pour les primitives des services DL-CONNECT, DL-RESET et DL-SUBSCRIBER-QUERY. Si la valeur spécifiée était autre que UNLIMITED, il convient que la durée de ce temporisateur soit $\frac{V_c(NP) \times MCD\_CRS}{V(NRC)+1}$; autrement, il convient que la durée soit $\frac{60s}{V(NRC)+1}$. La gestion de DL peut remplacer ces durées préférentielles.

   NOTE Voir la permission au début de l'Article 8 en ce qui concerne l'utilisation de V(NRC) versus V(MRC).
c) Si a) ne s'applique pas, la DLE doit
   – abouter la demande à la file d'attente de demandes utilisateur NODE de la DLE, à savoir Q_N(UR), à la priorité TIME-AVAILABLE, où la demande doit être placée dans la deuxième partition;

     NOTE La troisième partition dans la Q_N(UR) de la DLE est toujours vide.
   – abouter à la file d'attente de services non programmés de la DLE, à savoir Q(US), une référence à la Q_N(UR).

8.2.3.2 Transmission d'une SPDU DL-address query

À la suite de la réception d'une occasion de transmission pour la demande de service DL-SUBSCRIBER-QUERY placée en file d'attente, la DLE doit

a) former une SPDU "DL-address query" (interrogation d'adresse de DL) (voir 9.3.6.2) spécifiant l'adresse de DL demandée;
b) inclure cette SPDU comme étant la DLSDU d'une DLPDU DT de format 1S en mode sans connexion de priorité TIME-AVAILABLE avec une adresse de DL de destination SHORT pour les fonctions de prise en charge de DL de toutes les DLE présentes sur la liaison locale, 0100_{16} (voir 4.3.3.2), et une adresse de DL source égale à l'adresse de DL NODE.0, V(TN).0, des fonctions de prise en charge de DL de la DLE d'envoi.

8.2.3.3 Réception d'une SPDU "DL-address query" spécifiant une adresse de DLCEP abonné

Si une DLE reçoit une SPDU/DLPDU "DL-address query" (interrogation d'adresse de DL) pour une adresse de DL et une classe d'adresses de DL qui est active au sein de la DLE, alors la

DLE doit répondre à la priorité TIME-AVAILABLE par une SPDU/DLPDU de réponse d'adresse de DL correspondante, confirmant l'interrogation d'adresse de DL reçue.

8.2.3.4 Réception d'une SPDU DL-address reply

Si la DLE reçoit une SPDU/DLPDU "DL-address reply" (réponse d'adresse de DL) en réponse à la SPDU/DLPDU "interrogation d'adresse de DL" qui a été émise en résultat d'une demande de DL-SUBSCRIBER-QUERY, alors la DLE doit

a) annuler le temporisateur T_U(MCD);

b) initier une "confirm" de DL-SUBSCRIBER-QUERY rapportant "succès – un abonné existe";

c) libérer l'identifiant de demande, le rendant disponible pour une réutilisation ultérieure.

8.2.3.5 Expiration du temporisateur T_U(MCD) à la suite d'une "request" de DL-SUBSCRIBER-QUERY

Si le temporisateur T_U(MCD) expire à la suite d'une demande de DL-SUBSCRIBER-QUERY,

a) s'il s'agit de la (V(NRC)+1)^{ème} expiration consécutive, alors la DLE doit

   NOTE Voir la permission au début de l'Article 8 en ce qui concerne l'utilisation de V(NRC) versus V(MRC).

   1) mettre fin au traitement de la demande;

   2) initier une primitive "confirm" de DL-SUBSCRIBER-QUERY rapportant "échec à l'initiative d'un fournisseur – temporisation de demande";

b) autrement, si a) ne s'applique pas, la DLE doit

   1) relancer le temporisateur avec la même période que la fois précédente;

   2) replacer en file d'attente la SPDU/DLPDU pour une retransmission comme en 8.2.3.2b).

8.3 Fonctionnement des services en mode sans connexion

Les services en mode sans connexion sont le service "connectionless data transfer with local-DLE-confirmation" (transfert de données sans connexion avec une confirmation de DLE locale), le service "connectionless data transfer with remote-DLE-confirmation" (transfert de données sans connexion avec une confirmation de DLE distante), le service "connectionless data exchange" ('échange de données sans connexion) et le service "listener query" (interrogation d'auditeur).

8.3.1 Fonctionnement du service "transfert de données en mode sans connexion avec confirmation de DLE locale"

Les primitives du service "connectionless data transfer with local-DLE-confirmation" sont les primitives "request", "indication" et "confirm" de DL-UNITDATA.

8.3.1.1 Réception d'une primitive "request" de DL-UNITDATA ne spécifiant pas de confirmation de DLE distante

Lorsque la DLE reçoit une primitive «request» de DL-UNITDATA ne spécifiant pas de confirmation de DLE distante, elle doit associer l'identifiant de demande spécifié par l'utilisateur de DLS à la demande. Si la demande est rejetée, alors la DLE doit émettre une "confirm" de DL-UNITDATA avec le même identifiant de demande, acheminant le statut de la demande à l'utilisateur de DLS. Si la demande est acceptée, alors à l'achèvement de la transmission demandée, soit avec succès, soit après l'échec, la DLE doit émettre une "confirm" de DL-UNITDATA avec le même identifiant de demande, acheminant le statut de la demande à l'utilisateur de DLS.

a) Si le rôle de DL(SAP) de cette adresse de DL(SAP) source est BASIC, alors l'adresse de DL(SAP) appelante spécifiée dans la demande de DL-UNITDATA est liée comme étant une source à la priorité spécifiée à une file d'attente explicite (commandée par un utilisateur) ou implicite (commandée par une DLE).

b) Si le rôle de DL(SAP) de cette adresse de DL(SAP) source n'est pas BASIC, ou si cette file d'attente est pleine, ou si la DLSDU spécifiée a une longueur non valide, alors la DLE doit retourner immédiatement la cause de l'échec (dans une primitive «confirm» de DL-UNITDATA) comme étant le statut de la primitive "request" de DL-UNITDATA.

c) Si l'adresse de DL(SAP) appelée spécifiée dans la demande de DL-UNITDATA n'est pas une adresse de DL(SAP) dont le rôle de DL(SAP) est BASIC ou GROUP, alors la demande est erronée. La DLE a la permission, mais n'est pas tenue, de détecter une telle erreur (par exemple, en détectant que l'adresse appelée n'est pas une adresse de DL(SAP)). Si une telle erreur est détectée, alors la DLE doit immédiatement retourner la cause de l'échec (dans une primitive "confirm" de DL-UNITDATA) comme étant le statut de la primitive "request" de DL-UNITDATA.

d) Autrement, si ni b) ni c) ne s'applique, alors

   1) la DLE doit créer et démarrer un temporisateur de demandes d'utilisateur T_U(MCD) avec une durée basée sur le retard maximal de confirmation spécifié par un utilisateur pour la primitive "request" de DL-DATA. Si la valeur spécifiée était autre que UNLIMITED, il convient que la durée de ce temporisateur soit égale à ce retard maximal de confirmation spécifié par un utilisateur, P_U(MCD); autrement, il convient que la durée soit 60 s. La gestion de DL peut remplacer ces durées préférentielles.

   2) la DLE doit abouter la demande à la file d'attente de demandes utilisateur de l'adresse de DLSAP appelante, Q_A(UR), comme suit.

      i) Si l'attribut "DL-scheduling-policy" (politique de programmation de DL) de l'adresse de DLSAP appelante est EXPLICIT, alors la demande doit attendre une demande de DL-COMPEL-SERVICE pour la libérer pour transmission et, donc, doit être placée dans la troisième partition de la file d'attente de demandes utilisateur, Q_A(UR).

      ii) Autrement, si i) ne s'applique pas, la demande doit être placée dans la deuxième partition et la DLE doit abouter à la file d'attente de services non programmés de la DLE, Q(US), une référence à la Q_A(UR) de la même priorité que la demande qui vient juste d'être aboutée.

8.3.1.2 Transmission d'une DLSDU DT unitdata

À la suite de la réception d'une occasion de transmission pour la DLSDU placée en file d'attente, la DLE doit

a) retirer la demande de la file d'attente appropriée de demandes utilisateur d'adresse de DLSAP, $_A$(UR);

b) former et envoyer une DLPDU DT de la priorité spécifiée, avec les adresses spécifiées de DL(SAP) appelée et appelante, avec un champ "paramètres SD" vide (format P (voir Q7.4.1a))), et avec un champ "données utilisateur" dont la longueur et le contenu sont égaux à la DLSDU spécifiée;

   NOTE Avec l'exception d'un sous-champ "utilisation finale de jeton" du premier octet de la DLPDU DT, la DT DLPDU peut être formée lorsque la demande est mise en file d'attente, et peut ne pas être formée dynamiquement au moment de la transmission.

c) émettre une "confirm" de DL-UNITDATA avec l'identifiant de demande associé rapportant «succès»;

d) annuler et supprimer le temporisateur T_U(MCD) associé.

8.3.1.3 Réception d'une DLPDU DT, avec une adresse source explicite, adressée à une adresse de DL(SAP)

Lorsque la DLE reçoit une DLPDU DT avec une adresse source explicite adressée à une adresse de DL(SAP) liée à un ou plusieurs DLSAP de la DLE, alors si le rôle de DL(SAP) pour cette adresse de DL(SAP) de destination est

a) INITIATOR ou CONSTRAINED RESPONDER ou UNCONSTRAINED RESPONDER, alors la DLPDU reçue est erronée et la DLE

1) doit informer la gestion DL locale de l'événement, éventuellement en incluant les adresses de DLPDU erronées;
2) doit rejeter la DLPDU.

b) BASIC ou GROUP, alors pour chacune de ces liaisons
   1) la DLE doit tenter d'abouter les données utilisateur reçues comme une DLSDU, accompagnées des adresses de DL(SAP) appelée et appelante et de la priorité de DLL de la DLPDU reçue, à
      – la file d'attente de réception qui était liée au niveau de la priorité reçue, ou
      – la file d'attente OSI implicite si aucun tampon de réception ou aucune file d'attente n'a été liée(e) à ce niveau de priorité reçue;
   2) si 1) a réussi, alors la DLE doit initier une primitive «indication» de DL-UNITDATA en ce DLSAP;
   3) autrement, si 1) n'a pas réussi, alors la DLE doit informer la gestion de DL locale de cette situation de file d'attente pleine.

NOTE Cette notification à la gestion de DL peut prendre la forme consistant à incrémenter un compteur des DLSDU rejetées.

8.3.1.4 Expiration du temporisateur T_U(MCD) à la suite d'une "request" de DL-UNITDATA ne spécifiant pas de confirmation de DLE distante

Si le temporisateur T_U(MCD) expire à la suite d'une demande de DL-UNITDATA ne spécifiant pas confirmation de DLE distante, alors la DLE doit

a) retirer la demande de la file d'attente appropriée de demandes utilisateur d'adresse de DLSAP, Q_A(UR), et la référence à la demande de la file d'attente de services non programmés de la DLE, Q(US);
b) initier une "confirm" de DL-UNITDATA avec l'identifiant de demande associé rapportant "échec – temporisation avant transmission".

8.3.2 Fonctionnement du service "transfert de données en mode sans connexion avec confirmation de DLE distante"

Les primitives du service "connectionless data transfer with remote-DLE-confirmation" sont les primitives "request", "indication" et "confirm" de DL-UNITDATA.

8.3.2.1 Réception d'une primitive "request" de DL-UNITDATA spécifiant une confirmation de DLE distante

Lorsque la DLE reçoit une primitive «request» de DL-UNITDATA spécifiant une confirmation de DLE distante, elle doit associer à la demande l'identifiant de demande spécifié par l'utilisateur de DLS. Alors à l'achèvement de la demande, soit avec succès, soit après l'échec, la DLE doit émettre une "confirm" de DL-UNITDATA avec le même identifiant de demande, acheminant le statut de la demande à l'utilisateur de DLS.

L'adresse de DL(SAP) appelante spécifiée dans la demande de DL-UNITDATA est liée comme source à la priorité spécifiée à une file d'attente soit explicite (contrôlée par l'utilisateur), soit implicite (contrôlée par la DLE). Si le rôle de DL(SAP) de cette adresse de DL(SAP) source n'est pas BASIC, ou si cette file d'attente est pleine, ou si la DLSDU spécifiée a une longueur non valide, alors la DLE doit retourner immédiatement la cause de l'échec (dans une primitive «confirm» de DL-UNITDATA) comme étant le statut de la primitive "request" de DL-UNITDATA.

Si l'adresse de DL(SAP) appelée spécifiée dans la demande de DL-UNITDATA n'est pas une adresse de DLSAP dont le rôle de DL(SAP) est BASIC, alors la demande est erronée. La DLE a la permission, mais n'est pas tenue, de détecter une telle erreur (par exemple, en détectant que l'adresse appelée n'est pas une adresse de DLSAP). Si une telle erreur est détectée au cours du traitement de la demande, alors la DLE doit retourner un statut d'erreur approprié dans une primitive «confirm» de DL-UNITDATA, indiquant la cause de l'échec, et doit terminer le traitement de la demande.

Autrement

a) la DLE doit créer et démarrer un temporisateur de demandes d'utilisateur T_U(MCD) avec une durée basée sur le retard maximal de confirmation spécifié par un utilisateur pour la primitive "request" de DL-DATA. Si la valeur spécifiée était autre que UNLIMITED, il convient que la durée de ce temporisateur soit égale à ce retard maximal de confirmation spécifié par un utilisateur, P_U(MCD); autrement, il convient que la durée soit 60 s. La gestion de DL peut remplacer ces durées préférentielles.

b) la DLE doit abouter la demande à la file d'attente de demandes utilisateur de l'adresse de DLSAP appelante, Q_A(UR), comme suit:

   1) Si l'attribut "DL-scheduling-policy" (politique de programmation de DL) de l'adresse de DLSAP appelante est EXPLICIT, alors la demande doit attendre une demande de DL-COMPEL-SERVICE pour la libérer pour transmission et, donc, doit être placée dans la troisième partition de la file d'attente de demandes utilisateur, Q_A(UR).

   2) Autrement, la demande doit être placée dans la deuxième partition de la file d'attente de demandes utilisateur, Q_A(UR), et la DLE doit abouter à la file d'attente de services non programmés de la DLE, Q(US), une référence à la Q_A(UR) de la même priorité que la demande qui vient juste d'être aboutée.

8.3.2.2 Transmission d'une DLPDU CA unitdata

À la réception d'une occasion de transmission pour la DLSDU placée en file d'attente, la DLE doit former et envoyer une DLPDU CA de la priorité spécifiée,

a) avec les adresses de DLSAP spécifiées appelée et appelante;

b) avec un champ "paramètres SD" de format R (voir 7.4.1b));

c) avec un champ "données utilisateur" dont la longueur et le contenu sont égaux à la DLSDU spécifiée.

NOTE À l'exception du sous-champ "utilisation finale de jeton" du premier octet de la DLPDU CA, et éventuellement du sous-champ "indice de transaction" de l'initiateur, N(LTI), des paramètres SD de la DLPDU CA, (voir 7.4.1b)3)), la DLPDU CA peut être formée lorsque la demande est mise en file d'attente, et peut ne pas être formée dynamiquement au moment de la transmission.

8.3.2.3 Réception d'une DLPDU CA, avec une adresse source explicite, adressée à une adresse de DL(SAP)

Lorsque la DLE reçoit une DLPDU CA avec une adresse source explicite adressée à une adresse de DL(SAP) liée à un ou plusieurs DLSAP de la DLE, alors si le rôle de DL(SAP) pour cette adresse de DL(SAP) de destination est

a) GROUP, alors la DLPDU reçue est erronée, et

   1) la DLE doit informer la gestion de DL locale de l'événement, éventuellement en incluant les adresses de DLPDU erronées;

   2) la DLE doit rejeter la DLPDU CA reçue.

b) INITIATOR ou CONSTRAINED RESPONDER ou UNCONSTRAINED RESPONDER, alors la DLPDU reçue est erronée et

   1) la DLE doit informer la gestion de DL locale de l'événement, éventuellement en incluant les adresses de DLPDU erronées;

   2) la DLE doit former et envoyer, comme réponse immédiate, une DLPDU DT de réponse, telle que spécifiée en 6.4.4.1 et 6.7, dont le statut rapporté indique "échec – rôle de DL(SAP) de répondeur incompatible avec cette DLPDU";

   3) la DLE doit rejeter la DLPDU CA reçue.

c) BASIC, alors

   1) la DLE doit tenter d'abouter les données utilisateur reçues comme une DLSDU, accompagnées des adresses de DLSAP appelée et appelante et de la priorité de DLL de la DLPDU reçue, à la file d'attente de réception qui était explicitement ou implicitement liée au niveau de priorité reçue;

2) si 1) a réussi, alors la DLE doit initier une primitive «indication» de DL-UNITDATA en ce DLSAP et doit indiquer son succès en 4);

3) autrement, si 1) n'a pas réussi, alors la DLE doit indiquer la cause de son échec en 4) et doit notifier à la gestion de DL locale cet échec;

NOTE Cette notification à la gestion de DL peut prendre la forme consistant à incrémenter un compteur des DLSDU rejetées.

4) la DLE doit former et envoyer, comme réponse immédiate, une DLPDU DT de réponse, telle que spécifiée en 6.4.4.1 et 6.7, dont le statut rapporté indique le succès 2) ou l'échec 3).

8.3.2.4 Réception d'une DLPDU DT, avec une adresse source implicite, adressée à une adresse de DL(SAP)

Lorsque la DLE reçoit une DLPDU DT, avec une adresse source implicite, adressée à une adresse de DL(SAP) liée à un ou plusieurs DLSAP de la DLE, alors si le rôle de DL(SAP) pour cette adresse de DL(SAP) de destination est

a) CONSTRAINED RESPONDER ou UNCONSTRAINED RESPONDER ou GROUP, alors la DLPDU reçue est erronée;

   1) la DLE doit informer la gestion de DL locale de l'événement, éventuellement en incluant les adresses de DLPDU erronées;

   2) la DLE doit rejeter la DLPDU DT reçue.

b) INITIATOR, alors la DLE doit agir comme spécifié en 8.3.3.3.

c) BASIC, alors

   1) Si le sous-champ "transaction index" (indice de transaction), N(LTI), des paramètres SD de la DLPDU reçue spécifie un indice de transaction pour une demande incomplète en cours, alors la DLE

      i) doit libérer cet indice de transaction, le rendant disponible pour une utilisation ultérieure;

      ii) doit retirer la demande de la file d'attente appropriée de demandes utilisateur d'adresse de DLSAP, $_{A}$(UR);

      iii) doit annuler et supprimer le temporisateur T_U(MCD) associé à la demande;

      iv) doit émettre une "confirm" de DL-UNITDATA vers l'utilisateur de DLS demandeur, indiquant l'identifiant d'utilisateur de DLS de la demande et rapportant le statut acheminé par la DLPDU DT reçue.

   2) Autrement, si 1) ne s'applique pas, alors la DLE doit notifier à la gestion de DL locale la réception d'une DLPDU inappropriée.

   NOTE Cette notification à la gestion de DL peut prendre la forme consistant à incrémenter un compteur des DLSDU rejetées.

8.3.2.5 Réception d'une DLPDU SR

Lorsque la DLE reçoit une DLPDU SR,

– qui rapporte un statut différent de "BOK - bridge OK (pont correct)" (c'est-à-dire qu'il a une valeur différente de E_{16});

– qui est reçue comme réponse à une DLPDU CA, CD ou ED immédiatement antérieure qui avait été émise par la DLE destinataire et qui avait été adressée à une adresse de DLSAP;

– où le rôle de DL(SAP) pour l'adresse de DLSAP à partir de laquelle la transaction débutait (c'est-à-dire l'adresse de DLSAP source de la DLPDU CA, CD ou ED initiatrice) est BASIC,

alors la DLE destinataire doit

a) libérer l'indice de transaction, N(LTI), utilisé dans la DLPDU CA, CD ou ED immédiatement antérieure, le rendant disponible pour utilisation ultérieure;

b) retirer la demande de la file d'attente appropriée de demandes utilisateur d'adresse de DLSAP, $_A$(UR);

c) annuler et supprimer le temporisateur T_U(MCD) associé à la demande;

d) émettre une "confirm" de DL-UNITDATA vers l'utilisateur de DLS demandeur, indiquant l'identifiant d'utilisateur de DLS de la demande et rapportant le statut acheminé par la DLPDU SR reçue.

8.3.2.6 Expiration du temporisateur T_U(MCD) à la suite d'une "request" de DL-UNITDATA spécifiant une confirmation de DLE distante

Si le temporisateur T_U(MCD) expire à la suite d'une demande de DL-UNITDATA spécifiant une confirmation de DLE distante, alors la DLE

a) doit libérer l'indice de transaction correspondant et le retirer de l'utilisation active, le rendant disponible pour utilisation ultérieure seulement après une période de deux fois la durée de vie maximale de DLPDU dans le réseau, V(NDL), de la liaison étendue;

b) doit retirer la demande de la file d'attente appropriée de demandes utilisateur d'adresse de DLSAP, $_A$(UR);

c) doit supprimer le temporisateur T_U(MCD);

d) doit initier une "confirm" de DL-UNITDATA à l'utilisateur de DLS demandeur, indiquant l'identifiant d'utilisateur de DLS de la demande et rapportant un statut de "échec – temporisation avant transmission".

8.3.3 Fonctionnement du service "échange de données sans connexion"

8.3.3.1 Transmission d'une DLSDU CD ou ED unidata

À la suite de la réception d'une occasion de transmission pour une référence à une file d'attente de demandes utilisateur d'adresse de DL, Q_A(UR), pour une adresse de DLSAP dont le rôle de DL(SAP) est INITIATOR, la DLE doit examiner l'ensemble de liaisons d'envoi à un tampon pour cette adresse de DLSAP par ordre de priorité comme suit:

a) Si
   - il existe une liaison d'envoi, à la priorité URGENT, à un tampon, et
   - le tampon est non vide,

   alors les données utilisateur de DLS contenues dans le tampon URGENT deviennent la DLSDU sélectionnée pour transmission, et la priorité de cette DLSDU est URGENT; soit

b) Si a) ne s'applique pas, et
   - la priorité de transaction spécifiée dans la référence d'invocation est "normal" ou "time-available";
   - il existe une liaison d'envoi à la priorité "normal" à un tampon;
   - le tampon est non vide,

   alors les données utilisateur de DLS contenues dans le tampon NORMAL deviennent la DLSDU sélectionnée pour transmission, et la priorité de cette DLSDU est NORMAL; ou

c) Si ni a) ni b) ne s'applique, et
   - la priorité de transaction spécifiée dans la référence d'invocation est TIME-AVAILABLE;
   - il existe une liaison d'envoi, à la priorité TIME-AVAILABLE, à un tampon;
   - le tampon est non vide,

   alors les données utilisateur de DLS contenues dans le tampon TIME-AVAILABLE deviennent la DLSDU sélectionnée pour transmission, et la priorité de cette DLSDU est TIME-AVAILABLE; ou

d) Si aucun point a) à c) ne s'applique, alors aucune DLSDU n'est sélectionné pour transmission, alors la DLE doit former et envoyer une DLPDU appropriée:

e) Si l'un quelconque de a) à c) s'applique, alors une DLSDU était sélectionnée par cette procédure, et
   1) la DLE doit former et envoyer une DLPDU ED comme spécifié en 6.6.3
      i) avec l'adresse de destination égale à l'adresse de DLSAP distante spécifiée dans la référence d'invocation;
      ii) avec l'adresse source égale à l'adresse de DLSAP associée à la file d'attente de demandes utilisateur, Q_A(UR), spécifiée par cette référence;
      iii) avec une priorité égale à la priorité de transaction spécifiée dans la référence d'invocation;
      iv) avec un champ "paramètres SD" de format R (voir 7.4.1b)), spécifiant la priorité réelle de la DLSDU sélectionnée;
      v) avec un champ "données utilisateur" dont la longueur et le contenu sont égaux à la DLSDU sélectionnée;
   2) la DLE doit attendre une réponse immédiate. Si
      i) la DLSDU qui a été envoyée au répondeur dans la DLPDU ED était obtenue en provenance d'un tampon non rétenteur (BUFFER-NR);
      ii) une DLPDU DT ou SR appropriée est reçue comme réponse immédiate, comme spécifié en 6.5.3, 6.5.4, 6.6.3, 6.6.4, 6.7.3 et 6.8.3;
      iii) l'état explicite ou implicite acheminé par cette réponse est soit «succès», soit «succès provisoire»,

      alors le tampon doit être vidé (mis à vide).

f) Si d) s'applique, alors aucune DLSDU n'avait été sélectionnée par cette procédure, et
   1) la DLE doit former et envoyer une DLPDU CD comme spécifié en 6.5.3
      i) avec l'adresse de destination égale à l'adresse de DLSAP distante spécifiée dans la référence d'invocation;
      ii) avec l'adresse source égale à l'adresse de DLSAP associée à la file d'attente de demandes utilisateur, Q_A(UR), spécifiée par cette référence;
      iii) avec une priorité égale à la priorité de transaction spécifiée dans la référence d'invocation;
      iv) avec un champ "paramètres SD" de format R (voir 7.4.1b)), spécifiant qu'il n'y a aucune DLSDU d'accompagnement;
   2) la DLE doit attendre une réponse immédiate.

8.3.3.2 Réception d'une DLPDU CD ou ED, avec une adresse source explicite, adressée à une adresse de DL(SAP)

Lorsque la DLE reçoit une DLPDU CD ou ED avec une adresse source explicite adressée à une adresse de DL(SAP) liée à un ou plusieurs DLSAP de la DLE, alors si le rôle de DL(SAP) pour cette adresse de DL(SAP) de destination est

a) GROUP, alors la DLPDU reçue est erronée, et
   1) la DLE doit informer la gestion de DL locale de l'événement, éventuellement en incluant les adresses de DLPDU erronées;
   2) la DLE doit rejeter la DLPDU CD ou ED reçue.

b) BASIC ou INITIATOR, alors la DLPDU reçue est erronée, et
   1) la DLE doit informer la gestion de DL locale de l'événement, éventuellement en incluant les adresses de DLPDU erronées;
   2) la DLE doit former et envoyer, en réponse immédiate, une DLPDU DT de réponse, comme spécifié en 6.5.4.1, 6.6.4.1 et 6.7, dont le statut rapporté indique "échec – rôle de DL(SAP) de répondeur incompatible avec cette DLPDU";

3) la DLE doit rejeter la DLPDU CD ou ED reçue.

c) "constrained responder", alors

1) si l'adresse source de la DLPDU CD ou ED reçue est égale à la valeur du paramètre "adresse de DLSAP distante" qui est spécifié dans la plus récente primitive «request» de DL-BIND pour l'adresse de DLSAP répondeur, alors la DLE doit agir comme en d);

2) autrement, lorsque 1) ne s'applique pas, la DLPDU reçue est erronée et la DLE doit agir comme en b), excepté que l'état retourné en b)2) doit être «échec – réponse restreinte à une adresse de DLSAP homologue différente».

d) "unconstrained responder", alors

1) si la DLPDU reçue est une DLPDU ED et achemine donc une DLSDU non vide, alors

i) si

– l'adresse de DLSAP de répondeur a une liaison explicite comme destinataire à la priorité de la DLPDU reçue;

– le liaison est à un tampon;

– la taille du tampon est au moins aussi grande que la taille de la DLSDU,

alors la DLE doit écraser le tampon avec la DLSDU qui vient juste d'être reçue.

ii) Si

– l'adresse de DLSAP de répondeur a une liaison explicite comme destinataire à la priorité de la DLPDU reçue;

– le liaison est à une file d'attente;

– la taille de chaque enregistrement de file d'attente est au moins aussi grande que la taille de la DLSDU;

– la file d'attente n'est pas pleine,

alors la DLE doit abouter à la file d'attente la DLSDU qui vient juste d'être reçue.

iii) si soit i), soit ii) s'applique, alors la DLE doit examiner l'ensemble de liaisons au tampon d'envoi pour cette adresse de DLSAP, par ordre de priorité comme suit:

A) si

– il existe une liaison d'envoi, à la priorité URGENT, au tampon;

– le tampon est non vide,

alors les données utilisateur de DLS contenues dans le tampon URGENT deviennent la DLSDU sélectionnée pour transmission, et la priorité de cette DLSDU est URGENT; soit

B) si A) ne s'applique pas, et

– la priorité de transaction spécifiée dans la DLPDU reçue est NORMAL ou TIME-AVAILABLE;

– il existe une liaison d'envoi, à la priorité NORMAL à un tampon,

– le tampon est non vide,

alors les données utilisateur de DLS contenues dans le tampon NORMAL deviennent la DLSDU sélectionnée pour transmission, et la priorité de cette DLSDU est NORMAL; soit

C) si ni A) ni B) ne s'applique, et

– la priorité de transaction spécifiée dans la DLPDU reçue TIME-AVAILABLE;

– il existe une liaison d'envoi, à la priorité TIME-AVAILABLE, à un tampon;

– le tampon est non vide,

alors les données utilisateur de DLS contenues dans le tampon TIME-AVAILABLE deviennent la DLSDU sélectionnée pour transmission, et la priorité de cette DLSDU est TIME-AVAILABLE; soit

D) si aucun de A) à C) ne s'applique, alors aucune DLSDU n'est sélectionnée pour transmission.

iv) Si soit i), soit ii) s'applique, alors iii) s'applique, et

A) la DLE doit former et envoyer, en réponse immédiate, une DLPDU DT appropriée comme spécifié 6.7;

B) si une DLSDU a été envoyée à l'initiateur dans une DLPDU DT de réponse, et cette DLSDU a été obtenue à partir d'un tampon non rétenteur (BUFFER-NR), alors le tampon doit être vidé;

C) la DLE doit émettre une "indication" de DL-UNITDATA-EXCHANGE vers l'utilisateur de DLS répondeur, rapportant

– l'identifiant d'utilisateur de DLS/identifiant de DL de l'adresse de DLSAP de répondeur;

– la priorité des données utilisateur de DLS acheminées par la DLPDU ED reçue ou qu'aucune de telles données utilisateur de DLS n'a été reçue (dans une DLPDU CD);

– la priorité des données utilisateur de DLS acheminées dans la DLPDU DT de réponse, ou qu'aucune de telles données utilisateur de DLS n'a été envoyée;

– un statut de "succès".

v) Si ni i) ni ii) ne s'applique, alors la DLSDU reçue a été rejetée, et

A) la DLE doit former et envoyer, en réponse immédiate, une DLPDU DT appropriée comme spécifié 6.7;

B) la DLE doit émettre une "indication" de DL-UNITDATA-EXCHANGE vers l'utilisateur de DLS répondeur, rapportant

– l'identifiant d'utilisateur de DLS/identifiant de DL de l'adresse de DLSAP de répondeur;

– la priorité des données utilisateur de DLS acheminées par la DLPDU ED reçue qui a été rejetée;

– un statut de "échec".

2) Si la DLPDU reçue est une DLPDU CD et n'acheminait donc pas une DLSDU, alors

i) la DLE doit examiner l'ensemble de liaisons à un tampon d'envoi pour cette adresse de DLSAP, par ordre de priorité comme en d)1)iii);

ii) la DLE doit agir comme en d)1)iv)A) et d)1)iv)B);

iii) si

– une DLSDU était sélectionnée en ii), ou

– le paramètre "indicate-null-UNITDATA-EXCHANGE-transactions" qui avait été spécifié dans la plus récente primitive «request» de DL-BIND pour l'adresse de DLSAP de répondeur a la valeur TRUE,

alors la DLE doit agir comme en d)1)iv)C), excepté que le statut rapporté doit être «échec – aucune donnée utilisateur de DLS échangée».

8.3.3.3 Réception d'une DLPDU DT, avec une adresse source implicite, adressée à une adresse de DL(SAP)

Lorsque la DLE reçoit une DLPDU DT, avec une adresse source implicite, adressée à une adresse de DL(SAP) liée à un ou plusieurs DLSAP de la DLE, alors si le rôle de DL(SAP) pour cette adresse de DL(SAP) de destination est

a) BASIC ou CONSTRAINED RESPONDER ou UNCONSTRAINED RESPONDER ou GROUP, alors la DLE doit agir comme spécifié en 8.3.2.4.

b) INITIATOR, et si le sous-champ "transaction index" (indice de transaction), N(LTI), des paramètres SD de la DLPDU reçue spécifie un indice de transaction pour une demande incomplète en cours, alors

1) la DLE doit libérer cet indice de transaction, le rendant disponible pour une utilisation ultérieure;

2) la DLE doit annuler et supprimer le temporisateur T_U(MCD) TU(MCD) associé à la transaction d'échange d'unitdata maintenant terminée;

3) si la référence d'invocation était dans la file d'attente de services non programmés de la DLE, Q(US), ou a été dynamiquement aboutée à une séquence programmée spécifiée, alors la DLE doit retirer cette référence de la file d'attente ou de cet aboutement de séquence;

4) si la DLPDU DT reçue contient des données utilisateur de DLS et achemine donc une DLSDU non vide, alors

   i) si

   – l'adresse de DLSAP d'initiateur a une liaison explicite comme destinataire à la priorité de la DLPDU reçue;

   – le liaison est à un tampon;

   – la taille du tampon est au moins aussi grande que la taille de la DLSDU,

   la DLE doit écraser le tampon avec la DLSDU qui vient juste d'être reçue;

   ii) si

   – l'adresse de DLSAP d'initiateur a une liaison explicite comme destinataire à la priorité de la DLPDU reçue;

   – le liaison est à une file d'attente;

   – la taille de chaque enregistrement de file d'attente est au moins aussi grande que la taille de la DLSDU;

   – la file d'attente n'est pas pleine,

   la DLE doit abouter à la file d'attente la DLSDU qui vient juste d'être reçue;

   iii) si soit i), soit ii) s'applique, alors

   A) si une DLSDU qui a été envoyée au répondeur dans une DLPDU ED, et cette DLSDU a été obtenue à partir d'un tampon non rétenteur (BUFFER-NR), alors le tampon doit être vidé;

   B) la DLE doit émettre une "indication" de DL-UNITDATA-EXCHANGE vers l'utilisateur de DLS initiateur, rapportant

   – l'identifiant d'utilisateur de DLS/identifiant de DL de l'adresse de DLSAP d'initiateur;

   – la priorité des données utilisateur de DLS acheminées par la DLPDU ED précédemment envoyée ou qu'aucune de telles données utilisateur de DLS n'a été envoyée (dans une DLPDU CD);

   – la priorité des données utilisateur de DLS acheminées par la DLPDU reçue;

   – un statut de "succès";

   iv) si ni i) ni ii) ne s'applique, la DLE doit émettre une "indication" de DL-UNITDATA-EXCHANGE vers l'utilisateur de DLS initiateur, rapportant

   – l'identifiant d'utilisateur de DLS/identifiant de DL de l'adresse de DLSAP d'initiateur;

   – la priorité des données utilisateur de DLS acheminées par la DLPDU ED précédemment envoyée ou qu'aucune de telles données utilisateur de DLS n'a été envoyée (dans une DLPDU CD);

   – la priorité des données utilisateur de DLS acheminées par la DLPDU reçue;

   – un statut de "échec – limitation de ressource dans l'initiateur";

5) si 4) ne s'applique pas, et la DLPDU envoyée à la DLE de réponse était une DLPDU ED, acheminant donc une DLSDU non vide, alors

   i) si la DLSDU qui a été envoyée au répondeur dans la DLPDU ED a été obtenue à partir d'un tampon non rétenteur (BUFFER-NR), alors le tampon doit être vidé;

   ii) la DLE doit émettre une "indication" de DL-UNITDATA-EXCHANGE vers l'utilisateur de DLS initiateur, rapportant

      – l'identifiant d'utilisateur de DLS/identifiant de DL de l'adresse de DLSAP d'initiateur;

      – la priorité des données utilisateur de DLS acheminées par la DLPDU ED précédemment envoyée ou qu'aucune de telles données utilisateur de DLS n'a été envoyée (dans une DLPDU CD);

      – le statut acheminé par la DLPDU DT reçue;

6) si ni 4) ni 5) ne s'applique et la DLPDU envoyée à la DLE répondeuse était une DLPDU CD, alors aucune DLSDU n'a été échangée. Dans ce cas, si le paramètre "indicate-null-UNITDATA-EXCHANGE-transactions" qui était spécifié dans la plus récente primitive «request» de DL-BIND pour l'adresse de DLSAP d'initiateur avait la valeur TRUE, alors la DLE doit émettre une "indication" de DL-UNITDATA-EXCHANGE vers l'utilisateur de DLS initiateur, rapportant l'identifiant d'utilisateur de DLS/identifiant de DL de l'adresse de DLSAP d'initiateur et un statut de "échec – aucune donnée utilisateur de DLS échangée";

c) INITIATOR, et si le sous-champ "indice de transaction", N(LTI), des paramètres SD de la DLPDU reçue ne spécifie pas un indice de transaction pour une demande incomplète en cours, alors la DLE doit notifier à la gestion de DL locale la réception d'une DLPDU inappropriée.

NOTE Cette notification à la gestion de DL peut prendre la forme consistant à incrémenter un compteur des DLSDU rejetées.

8.3.3.4 Réception d'une DLPDU SR

Lorsque la DLE reçoit une DLPDU SR,

– qui rapporte un statut différent de "BOK - bridge OK (pont correct)" (c'est-à-dire qu'il a une valeur différente de E_{16});

– qui était reçue comme réponse à une DLPDU CA, CD ou ED immédiatement antérieure qui avait été émise par la DLE destinataire et qui avait été adressée à une adresse de DLSAP;

– où le rôle de DL(SAP) pour l'adresse de DLSAP à partir de laquelle la transaction débutait (c'est-à-dire l'adresse de DLSAP source de la DLPDU CA, CD ou ED initiatrice) est INITIATOR,

alors

a) la DLE répondeuse doit libérer l'indice de transaction, N(LTI), utilisé dans la DLPDU CA, CD ou ED immédiatement antérieure, le rendant disponible pour utilisation ultérieure;

b) la DLE destinataire doit annuler et supprimer le temporisateur T_U(MCD) associé à la demande;

c) si la référence d'invocation était dans la file d'attente de services non programmés de la DLE, Q(US), ou a été dynamiquement aboutée à une séquence programmée spécifiée, alors la DLE doit retirer cette référence de la file d'attente ou de cet aboutement de séquence;

d) si le paramètre "indicate-null-UNITDATA-EXCHANGE-transactions" qui était spécifié dans la plus récente primitive «request» de DL-BIND pour l'adresse de DLSAP d'initiateur avait la valeur TRUE, alors la DLE doit émettre une "indication" de DL-UNITDATA-EXCHANGE vers l'utilisateur de DLS initiateur, rapportant l'identifiant d'utilisateur de DLS/identifiant de DL de l'adresse de DLSAP d'initiateur et un statut de "échec – aucune donnée utilisateur de DLS échangée";

8.3.3.5 Expiration du temporisateur T_U(MCD) sur une instance incomplète du service d'échange d'unitdata

Si le temporisateur T_U(MCD) expire sur une instance du service d'échange d'unitdata, alors la DLE

a) doit libérer l'indice de transaction correspondant et le retirer de l'utilisation active, le rendant disponible pour utilisation ultérieure seulement après une période de deux fois la durée de vie maximale de DLPDU, V(MDL), de la liaison étendue;

b) doit supprimer le temporisateur T_U(MCD);

c) doit émettre une "indication" de DL-UNITDATA-EXCHANGE vers l'utilisateur de DLS initiateur, spécifiant l'identifiant d'utilisateur de DLS/identifiant de DL de l'adresse de DLSAP d'initiateur et rapportant le statut "échec à l'initiative d'un fournisseur – temporisation de demande".

8.3.4 Fonctionnement du service "listener query"

Les primitives du service d'interrogation d'auditeur sont "request" et "confirm" de DL-LISTENER-QUERY.

8.3.4.1 Réception d'une primitive "request" de DL-LISTENER-QUERY

Lorsque la DLE reçoit une primitive «request» de DL-LISTENER-QUERY, elle doit associer à la demande l'identifiant de demande spécifié par l'utilisateur de DLS. Si la demande est acceptée, comme indiqué par un statut retourné de "demande acceptée" pour la demande de DL-LISTENER-QUERY, alors à la suite de l'achèvement de la demande, soit avec succès, soit après l'échec, la DLE doit émettre une "confirm" de DL-LISTENER-QUERY avec le même identifiant de demande, acheminant le statut de la demande à l'utilisateur de DLS.

La demande de DL-LISTENER-QUERY doit être placée dans la file d'attente de demandes utilisateur d'adresse de DL NODE de la DLE, à savoir Q_N(UR), qui est une file d'attente implicite (commandée par une DLE).

a) Si cette file d'attente est pleine, alors la DLE doit immédiatement retourner la "confirm" de DL-LISTENER-QUERY correspondante indiquant la cause de l'échec – "échec à l'initiative d'un fournisseur – file d'attente pleine".

b) Si a) ne s'applique pas, alors la DLE doit lancer un temporisateur de demandes d'utilisateur T_U(MCD) avec une durée basée sur le retard maximal de confirmation spécifié par un utilisateur pour la primitive "request" de DL-LISTENER-QUERY. Si la valeur spécifiée était autre que UNLIMITED, il convient que la durée de ce temporisateur soit $\frac{P_U(\text{MCD})}{V(\text{NRC})+1}$; autrement, il convient que la durée soit $\frac{60\text{s}}{V(\text{NRC})+1}$. La gestion de DL peut remplacer ces durées préférentielles.

   NOTE Voir la permission au début de l'Article 8 en ce qui concerne l'utilisation de V(NRC) versus V(MRC).

c) Si a) ne s'applique pas, la DLE doit
   – abouter la demande à la file d'attente de demandes utilisateur NODE de la DLE, à savoir Q_N(UR), à la priorité TIME-AVAILABLE, où la demande doit être placée dans la deuxième partition;

     NOTE La troisième partition dans la Q_N(UR) de la DLE est toujours vide.

   – abouter à la file d'attente de services non programmés de la DLE, à savoir Q(US), une référence à la Q_N(UR).

8.3.4.2 Transmission d'une SPDU "DL-address query"

À la réception d'une occasion de transmission pour la demande de DL-LISTENER-QUERY mise en file d'attente, la DLE doit former et envoyer une SPDU "DL-address query" (interrogation d'adresse de DL) comme spécifié en 8.2.3.2.

8.3.4.3 Réception d'une SPDU "DL-address query" spécifiant une adresse de DL de groupe

Si une DLE reçoit une SPDU d'interrogation d'adresse de DL (acheminé par une DLPDU DT) pour une adresse de DL et une classe d'adresses de DL qui est active dans la DLE, alors la DLE doit

a) former une DLPDU DT de réponse contenant une SPDU "DL-address reply" (réponse d'adresse de DL) correspondante, confirmant l'interrogation d'adresse de DL reçue;
b) abouter la DLPDU DT à la file d'attente de demandes utilisateur NODE de la DLE, Q_N(UR), à la priorité applicable;
c) abouter à la file d'attente de services non programmés de la DLE, à savoir Q(US), une référence à la Q_N(UR).

8.3.4.4 Réception d'une SPDU DL-address reply

Si la DLE reçoit une SPDU "réponse d'adresse de DL" (acheminée par une DLPDU DT) en réponse évidente à la SPDU/DLPDU "interrogation d'adresse de DL" qui a été émise en résultat d'une demande de DL-LISTENER-QUERY, alors la DLE doit

a) annuler le temporisateur T_U(MCD);
b) initier une "confirm" de DL-LISTENER-QUERY avec l'identifiant de demande correspondant rapportant “succès – un auditeur existe”.

8.3.4.5 Expiration du temporisateur T_U(MCD) à la suite d'une "request" de DL-LISTENER-QUERY

Si le temporisateur T_U(MCD) expire à la suite d'une demande de DL-LISTENER-QUERY, alors

a) s'il s'agit de la (V(NRC)+1)^{ème} expiration consécutive, alors la DLE doit

   NOTE Voir la permission au début de l'Article 8 en ce qui concerne l'utilisation de V(NRC) versus V(MRC).

   1) mettre fin au traitement de la demande;
   2) initier une "confirm" de DL-LISTENER-QUERY avec l'identifiant de demande correspondant rapportant “échec à l'initiative d'un fournisseur – temporisation de demande”.
b) Autrement, si a) ne s'applique pas, la DLE doit
   1) relancer le temporisateur avec la même période que la fois précédente;
   2) replacer en file d'attente la même DLPDU DT pour une retransmission comme spécifié en 8.3.4.3b).

8.4 Fonctionnement des services de conseils de programmation

Les services de conseils de programmation sont les services "DL-time" (temps de DL), le service "Compel service" (forcer un service), le service "Sequence scheduling" (programmation de séquence) et le service "Sequence subsetting" (création de sous-ensembles de séquence).

8.4.1 Fonctionnement du service "DL-time"

La seule primitive du service "DL-time" est la "request" de DL-TIME.

8.4.1.1 Réception d'une primitive "request" de DL-TIME

Lorsque la DLE reçoit une demande de DL-TIME, elle doit répondre avec un enregistrement de cinq composantes tel que montré dans le Tableau 78, dont les valeurs sont

a) la valeur courante du link-id (identifiant de liaison) de la source de temps, V(TSL);
b) la qualité du temps de DL courant, V(TQ), tel que défini en 7.6b);

c) la valeur courante du décalage de temps de DL local (local DL-time-offset), V(DLTO);

d) la valeur courante du décalage de temps de programmation de liaison locale (local-link-scheduling-time-offset), V(LSTO);

e) la valeur courante du temps de nœud local (local node-time), C(NT).

L'utilisateur de DLS peut alors sommer les trois dernières valeurs pour déterminer le temps de DL courant du nœud ou utiliser les valeurs séparément, selon le cas.

Tableau 78 – Composantes du temps de DL retourné

Indice d'octet	Contenu de champ
1 2	V(TSL)
3	V(TQ)
4 5 6 7 8 9 10	V(DLTO)
11 12 13 14 15 16 17	V(LSTO)
18 19 20 21 22 23 24	C(NT)

Si la classe de synchronisation de temps de la DLE (voir 10.6.3) est NONE et la DLE ne maintient même pas un C(NT) estimé, alors elle peut répondre par les valeurs 0…0 pour V(TQ), V(DLTO), V(LSTO) et C(NT), et par $FFFF_{16}$ pour V(TSL); ce dernier, qui est un identifiant de liaison non valide, indique à l'utilisateur de DLS l'incapacité de la DLE.

8.4.1.2 Transmission d'une DLPDU TD

Après avoir envoyé une DLPDU TD, la DLE LAS doit relancer son temporisateur local T(TDP), pour expirer à un temps inférieur ou égal à 0,95 × V(TDP) dans le futur (voir 4.7.1.26 et 4.7.5.18).

8.4.1.3 Réception d'une DLPDU TD

Si la classe de synchronisation du temps des la DLE (voir 10.6.3) est NONE, alors

a) Si la DLE qui n'est pas la DLE LAS prend en charge les variables, les temporisateurs et les compteurs définis de 4.7.1.19 à 4.7.1.27, alors la DLE doit mettre V(TSL) à la valeur N_S(TSL), mettre V(DLTO) à N_S(DLTO) et mettre V(LSTO) à la valeur

$$\left(N_S(DLT) + N_S(DLTA) - N_S(DLTO) - N_R(NT) + \frac{V(MD)}{2} \right) \text{ modulo } 2^{56}. \quad (26)$$

b) Autrement, lorsque a) ne s'applique pas, alors la DLE doit ignorer la réception de la DLPDU TD.

Lorsqu'une DLE autre que la DLE LAS, avec une classe de synchronisation de temps autre que NONE, reçoit une DLPDU TD, alors

c) la DLE doit relancer son temporisateur local T(TDP), pour expirer à un temps lui permettant, s'il ne reçoit pas de DLPDU TD jusqu'à l'expiration de ce temporisateur, de toujours pouvoir maintenir une synchronisation de temps dans les limites imposées par la classe de synchronisation de temps de la liaison locale (voir 4.7.1.25). La valeur de ce temporisateur doit être supérieure au minimum spécifié pour la classe de synchronisation de temps en 10.6.3, mais la valeur exacte peut être déterminée localement par la DLE.

d) la DLE doit noter que la DLE n'a pas à envoyer un CT.

NOTE La variation des périodes d'expiration des temporisateurs T(TDP) parmi les DLE, et l'annulation à la suite d'une réception d'une DLPDU TD de telles demandes pour la transmission d'une DLPDU CT servent toutes les deux à réduire l'éventualité qu'un seul événement de bruit à l'échelle de la liaison, bloquant la réception correcte d'une DLPDU TD, donne lieu à plusieurs transmissions de DLPDU CT.

e) la DLE doit mettre à jour

   – son sens maintenu localement de l'identifiant de liaison de cette liaison, au sein de la liaison étendue, qui est à la source de temps, V(TSL);
   – son sens maintenu localement du nœud de LAS local, V(LN);
   – son sens maintenu localement de la qualité de temps de DL, V(TQ);
   – son DL-time-offset (décalage du temps de DL), V(DLTO);
   – son local-link-scheduling-time-offset (décalage de programmation programmeur de temps de la liaison locale), V(LSTO);
   – sa fréquence de temporisateur de nœud,

   en comparant les variables locales V(TQ), V(TSL) et V(LN) aux champs correspondants de la DLPDU TD reçue – N_S(TQ) et N_S(TSL) et l'adresse de DL NODE source de la DLPDU – pour détecter les changements soit dans la source de référence pour le temps soit dans le chemin de distribution de temps.

   1) Si l'adresse de DL source de NODE de la DLPDU TD diffère du V(LN) de la DLE de réception, alors

      i) la DLE doit mettre V(LN) à la valeur de l'adresse de DL NODE source issue de la DLPDU reçue;

      ii) la DLE doit informer la gestion de DL locale qu'un changement de source de temps sur la liaison locale a eu lieu;

      iii) si la DLE exige une mesure du temps de propagation aller-retour entre elle-même et le LAS local, afin de maintenir l'exactitude requise pour sa classe de synchronisation de temps, alors la DLE doit invalider V(MD), la mesure de temps de propagation aller-retour antérieure avec le LAS courant.

   2) Si la DLE exige une mesure du temps de propagation aller-retour entre elle-même et le LAS local, et si la valeur courante pour ce temps de propagation aller-retour, V(MD), est notoirement non valide, alors la DLE doit former une DLPDU RQ adressée à V(LN).0, et abouter la DLPDU comme spécifié en 8.4.5, à la priorité NORMAL, à la file d'attente de demandes utilisateur d'adresse de DL NODE de la DLE, à savoir Q_N(UR).

      NOTE Il ne s'agit pas de la file d'attente normalement utilisée avec les procédures en 8.4.5.

   3) Si la mesure du temps de propagation aller-retour au LAS courant, V(MD), est valide, alors la DLE doit mettre son V(TQ) à partir de la valeur de qualité temps du nœud (node-time-quality), N_S(TQ), dans la DLPDU reçue, après réajustement du sous-champ "number-of-intervening-links" (nombre de liaisons intermédiaires) pour la liaison intermédiaire supplémentaire entre les DLE d'envoi et de réception (en incrémentant N_S(TQ).LLL NS(TQ).LLL lors de la formation de V(TQ) pour rendre compte de la DLE de réception).

      NOTE Donc une DLE avec sa propre source de temps, qui ne sert pas aussi de source de temps de DL sur la liaison locale, doit ignorer sa propre source de temps et se synchroniser aux autres DLE sur la liaison locale.

4) Si la valeur liaison de source de temps (time-source-link value), N_S(TSL), dans la DLPDU reçue diffère du V(TSL) de la DLE de réception, et si la mesure de temps de propagation aller-retour avec le LAS courant, V(MD), est valide, alors la DLE doit
   i) mettre V(TSL) à la valeur de N_S(TSL);
   ii) mettre à la valeur de N_S(TSL) toutes les autres références locales à la liaison de source de temps, telles que l'identifiant de liaison de la source de distribution de temps des activités programmées périodiques en 10.3.7);
   iii) informer la gestion de DL locale qu'un changement de source de temps sur la liaison étendue a eu lieu;
5) Si le décalage de temps de DL, N_S(DLTO), dans la DLPDU reçue diffère du V(DLTO) de la DLE de réception, alors la DLE doit
   i) mettre V(DLTO) à la valeur de N_S(DLTO);
   ii) réajuster toutes les autres références locales basées sur le temps de DL, telle la base de temps de DL de programmation périodique DL (T0) en 10.3.7), par la quantité dont que V(DLTO) vient juste de changer, pour refléter la nouvelle base de temps de DL;
   iii) informer la gestion de DL locale qu'un changement de V(DLTO) a eu lieu.
6) Si la mesure du temps de propagation aller-retour au LAS courant, V(MD), est valide, alors la DLE doit calculer

$$TEMP = (N_S(DLT) + N_S(DLTA) - N_S(DLTO) - V(LSTO) - N_R(NT)) \text{ modulo } 2^{56} \quad (27)$$

NOTE L'équation ci-dessus est équivalente dans sa forme aux calculs en a).

Dans les calculs suivants, la DLE doit utiliser la valeur de *K* spécifiée dans le Tableau 79.

Tableau 79 – Calcul de la synchronisation du temps

Classe Time Synchronization (synchronisation du temps)	Valeur of *K* à utiliser en e)4)
1 µs	4
10 µs	41
100 µs	410
1 ms	4 096
10 ms	40 960
100 ms	409 600
1 s	4 096 000

   i) Si abs(*TEMP*), la valeur absolue de *TEMP*, est supérieure à *K*, et si ceci n'a pas été le cas pour les deux DLPDU TD reçues précédemment, alors la DLE doit
      A) mettre V(LSTO) à la valeur de (*TEMP* + V(LSTO)) modulo 2^{56};
      B) informer la gestion de DL locale qu'un changement discontinu de V(LSTO) a eu lieu.
   ii) Si abs(*TEMP*), la valeur absolue de *TEMP*, est inférieure ou égale à *K*, ou si elle était supérieure à *K* pour les deux DLPDU TD reçues précédemment, alors la DLE doit utiliser la valeur calculée, *TEMP*, avec des valeurs calculées antérieures selon le cas, pour réajuster la fréquence d'incrémentation du C(NT) de la DLE.
      A) pour réduire à zéro la précession à long terme de V(LSTO), par rapport à N_S(DLT) + N_S(DLTA) – N_S(DLTO),
      B) pour conserver la fréquence de comptage à long terme de C(NT) afin d'incrémenter approximativement de 8 192 000 ($2^{13} \times 10^3$) comptes par seconde.

Ce réajustement de fréquence doit être maintenu jusqu'à ce qu'un nouveau réajustement soit calculé.

NOTE Les détails de l'algorithme de filtrage qui déterminent ce réajustement de la fréquence sont intentionnellement laissés à la discrétion du réalisateur.

8.4.1.3.1 Actions complémentaires requises d'un pont

Si la DLPDU TD a été reçue en un port actif dans l'état de transmission (voir Article 9) qui est le port racine de la DLE pont (c'est-à-dire le port actif vers la racine de l'arborescence du pont), la DLE doit transmettre l'événement de réception de la DLPDU au niveau opérationnel du pont (voir 4.1.2). Le niveau opérationnel du pont doit partager le sens du temps reçu avec

– les autres ports actifs du pont qui servent actuellement de LAS sur leurs propres liaisons locales;
– tous les ports de pont qui ne sont pas dans les états de transmission ou d'écoute;
– tout utilisateur de DLS local.

Ce qui suit s'applique également.

a) Si le pont a un port quelconque autre que son port racine qui agit comme LAS, alors le pont doit
   1) mettre à jour le décalage de temps de DL, V(DLTO), pour chacun de ces autres ports, afin que la somme de C(NT), V(LSTO) et V(DLTO) pour ce port autre que racine soit égale à la somme de C(NT), V(LSTO) et V(DLTO) pour le port racine;

      NOTE Ce décalage de temps de DL, V(DLTO), et le décalage de temps de programmation de la liaison locale, V(LSTO), sont spécifiques à un port (comme le sont la plupart des variables en 4.7.5), et sont donc maintenus indépendamment pour chaque port.
   2) si 1) a causé un changement dans le décalage de temps de DL, V(DLTO), alors ajuster la base de temps de DL de programmation périodique, T0, en 10.3.7) par la même quantité montant dont V(DLTO) vient juste de changer, pour refléter la nouvelle base de temps de DL;
   3) s'il y a eu changement dans la liaison de source de temps, V(TSL), alors mettre à la nouvelle valeur de V(TSL) l'identifiant de liaison de la source de distribution de temps d'activités programmées périodiques en 10.3.7).
b) Le niveau opérationnel du pont peut programmer l'envoi de DLPDU TD complémentaires sur ces autres ports qui agissent comme LAS.
c) Le contenu des sous-champs de temps de ces DLPDU postérieures doit être fondé sur les valeurs alors courantes pour les variables de temps de nœud intérieurement maintenues du port d'envoi, C(NT), V(LSTO) et V(DLTO), plutôt que sur la valeur des paramètres de temps de DL de la DLPDU TD qui vient juste d'être reçue.
d) Le contenu des sous-champs "source de temps" et "qualité de temps" – V(TSL) et N(TQ) – de ces DLPDU postérieures doit être basé sur les champs correspondants issus de la DLPDU TD qui vient juste d'être reçue, excepté que le N(TQ) envoyé doit refléter à la fois
   – tout impact de la classe d'opportunité de la DLE sur la résolution du temps transmis;
   – le fait qu'il existe un pont supplémentaire dans le chemin de transmission.

8.4.1.4 Réception d'une DLPDU RQ

Lorsque la DLE reçoit une DLPDU RQ, elle doit former et envoyer, en réponse immédiate, une DLPDU ROUND-TRIP-DELAY REPLY (RR)

a) dont l'adresse de DL de destination est égale à l'adresse source de la DLPDU RQ reçue;
b) dont l'adresse source est formée en concaténant l'identifiant de nœud de l'expéditeur, V_S(TN), et un octet de zéro;
c) dont les deux premiers sous-champs "paramètre" sont respectivement égaux aux sous-champs "paramètre" de la DLPDU RQ reçue et à laquelle l'aboutement est effectué.

8.4.1.5 Réception d'une DLPDU RR

Lorsque la DLE reçoit une DLPDU RR, si la DLE maintient le C(NT), alors elle doit, dans la limite d'une période raisonnable n'excédant pas 60 s, calculer le temps de propagation aller-retour, RTD, en fonction des quatre sous-champs "paramètre" de la DLPDU RR reçue et à laquelle l'aboutement est effectué, comme étant:

$$\text{RTD} = (\text{short-time}_4 - \text{short-time}_3 + \text{short-time}_2 - \text{short-time}_1)\ \text{modulo}\ 2^{24} \tag{28}$$

où les champs sont numérotés dans leur ordre de transmission (ou d'aboutement) au sein de la DLPDU. Si la DLE distante dans la mesure était la DLE LAS courante, V(LN), alors le temps de propagation calculé doit être sauvegardé comme étant le temps de propagation mesuré

$$\text{V(MD)} = \text{RTD} \tag{29}$$

qui, de ce fait, devient valide pour une utilisation dans les calculs de réajustement de fréquence de C(NT), comme spécifié dans l'Équation (15) (voir 8.4.1.3).

Les résultats de plusieurs mesures aller-retour avec le même nœud distant peuvent être combinés pour réduire l'erreur moyenne dans V(MD).

8.4.1.6 Expiration du temporisateur T(TDP)

Si le temporisateur T(TDP) expire, alors

a) si la DLE est le LAS de la liaison locale, alors la DLE doit noter l'exigence d'émettre une DLPDU TD à la première occasion;
b) autrement, la DLE doit noter l'exigence d'émettre une DLPDU CT à la première occasion;

8.4.2 Fonctionnement du service "Compel-service"

La seule primitive du service compel-service est "request" de DL-COMPEL-SERVICE.

8.4.2.1 Réception d'une primitive "request" de DL-COMPEL-SERVICE

Lorsque la DLE reçoit une "request" de DL-COMPEL-SERVICE, il doit classer la demande et entreprendre l'action appropriée correspondante. Si la demande porte sur

a) une adresse de DLSAP locale (à la DLE) dont le rôle de DL(SAP) est BASIC, ou un DLCEP homologue ou éditeur ou abonné, pour lequel la politique de programmation de DL (DL-scheduling-policy) est EXPLICIT, alors
   1) si l'adresse de DL est liée à une file d'attente d'envoi, et le nombre des DLSDU dans la troisième partition de cette file d'attente de demandes utilisateur d'adresse de DL, Q_A(UR), qui attendent une primitive "request" de DL-COMPEL-SERVICE est autre que zéro, alors la DLE doit
      i) réajuster la file d'attente de demandes utilisateur d'adresse de DL locale, Q_A(UR), en réduisant de un ce nombre (des DLSDU dans la troisième partition);
      ii) réassigner la demande de la plus haute priorité, de cette troisième partition vers la deuxième partition, à moins qu'un tel déplacement ne provoque le dépassement de la taille de fenêtre d'envoi du DLCEP adressé;

      NOTE 1 La restriction en ii) ci-dessus s'applique seulement aux DLCEP homologues et éditeurs, et jamais aux adresses de DLSAP.

      NOTE 2 Les mises en œuvre sont autorisées à différer la réassignation réelle jusqu'à l'apparition d'une occasion d'émettre pour la DLSDU réassignée. Un tel différé peut permettre à une DLPDU de plus haute priorité demandée postérieurement d'être réassignée en lieu et place de la DLSDU qui aurait été réassignée au moment où était faite la "request" de DL-COMPEL-SERVICE. Toutefois, une mise en œuvre ne peut en aucun cas accumuler un "quota de demandes DL-COMPELled (forcées par la DL) " qui se prolonge au-delà de la prochaine fois que la troisième partition de la file d'attente Q_A(UR) devient vide.

iii) former une référence à cette Q_A(UR) d'adresse de DL à la priorité spécifiée dans la "request" de DL-COMPEL-SERVICE, où la référence indique le besoin d'envoyer une DLSDU de la file d'attente d'envoi identifiée en 1), et d'abouter la référence soit à

A) la séquence programmée spécifiée, si la demande de DL-COMPEL-SERVICE spécifiait un identifiant de séquence, soit

B) la file d'attente de services non programmés de la DLE, Q(US), si aucun identifiant de séquence n'a été spécifié dans la demande;

iv) retourner un statut immédiat de "succès".

Autrement, lorsque aucune réaffectation n'a eu lieu, la DLE doit retourner un statut immédiat de "échec de l'utilisateur – aucune DLSDU à libérer".

2) Si l'adresse de DL est liée à un tampon d'envoi, alors la DLE doit

i) modifier la variable $V_{C,K}$(SS), pour le *K* approprié correspondant à la DLSDU courante associée au tampon, afin d'indiquer que tous les segments de la DLSDU exigent une transmission;

ii) former une référence à cette Q_A(UR) du DLCEP homologue ou éditeur local, à la priorité de DLCEP, où la référence indique le besoin d'envoyer une DLSDU du tampon d'envoi identifié en 2), et d'abouter la référence soit à

A) la séquence programmée spécifiée, si la demande de DL-COMPEL-SERVICE spécifiait un identifiant de séquence, soit

B) la file d'attente de services non programmés de la DLE, Q(US), si aucun identifiant de séquence n'a été spécifié dans la demande;

iv) retourner un statut immédiat de "succès".

b) le DLCEP homologue ou éditeur distant respectif d'un DLCEP abonné ou homologue local, alors la DLE doit

1) former une référence à la Q_A(UR) du DLCEP homologue ou éditeur local spécifié, à la priorité du DLCEP, où la référence indique le besoin de forcer la transmission d'une DLSDU à partir du DLCEP homologue ou éditeur de correspondant distant identifié en b), et d'abouter la référence soit à

i) la séquence programmée spécifiée, si la demande de DL-COMPEL-SERVICE spécifiait un identifiant de séquence, soit

ii) la file d'attente de services non programmés de la DLE, Q(US), si aucun identifiant de séquence n'a été spécifié dans la demande;

NOTE La référence ci-dessus est discernable des références qui conduisent à une libération locale de DLSDU.

2) retourner un statut immédiat de "succès";

c) une adresse de DLSAP locale (à la DLE) dont le rôle DL(SAP) est INITIATOR, alors la DLE doit

1) former une référence à cette Q_A(UR) d'adresse de DL à la priorité spécifiée dans la "request" de DL-COMPEL-SERVICE, où la référence indique le besoin de forcer une instance du service "unitdata-exchange" (échange d'unitdata) avec une adresse de DLSAP distante spécifiée et d'abouter la référence soit à

i) la séquence programmée spécifiée, si la demande de DL-COMPEL-SERVICE spécifiait un identifiant de séquence, soit

ii) la file d'attente de services non programmés de la DLE, Q(US), si aucun identifiant de séquence n'a été spécifié dans la demande;

2) retourner un statut immédiat de "succès".

d) Une autre adresse de DL, alors la DLE doit retourner un statut immédiat de "échec de l'utilisateur – adresse de DL non valide".

8.4.3 Fonctionnement du service Sequence scheduling

Les primitives de service de programmation de séquence sont "request" et "confirm" de DL-SCHEDULE-SEQUENCE, et "request" et "confirm "de DL-CANCEL-SCHEDULE.

L’aptitude d’une DLE à répondre positivement à une demande DL-SCHEDULE-SEQUENCE est déterminée à la fois par les propres aptitudes de la DLE à servir une demande dynamique (c’est-à-dire qui n'est pas préconfigurée par la gestion de DL) et par l’aptitude de la DLE LAS courante à prendre en charge ce processus de programmation. La conformité revendiquée à l’Article 9 fournit un certain nombre d'informations relatives aux capacités de la DLE.

8.4.3.1 Réception d'une primitive "request" de DL-SCHEDULE-SEQUENCE

Lorsque la DLE reçoit une demande de DL-SCHEDULE-SEQUENCE, elle doit assigner un identifiant de programme à cette demande et retourner cet identifiant comme partie intégrante de la primitive «request» de DL-SCHEDULE-SEQUENCE.

À l'achèvement de la demande de programmation, soit avec succès, soit après l’échec, la DLE doit émettre une "confirm" de DL-SCHEDULE-SEQUENCE avec le même identifiant de programme, acheminant le statut de la demande de programmation à l’utilisateur de DLS. Si la demande de programmation a échoué, alors a DLE doit libérer l’identifiant de programme, le rendant disponible pour une utilisation ultérieure.

NOTE 1 La "confirm" de DL-SCHEDULE-SEQUENCE indique l’achèvement de la *programmation* de la séquence associée soit localement, soit en conjonction avec le LAS local; elle n’indique pas l’achèvement de l’*exécution* de cette séquence. Une mise en œuvre qui diffère la transmission des primitives "confirm" de DL-SCHEDULE-SEQUENCE jusqu’à un moment après l’achèvement de leurs séquences programmées est erronée.

La séquence proprement dite est définie en termes de composantes de séquence de primitive spécifiées à l’Article 9 et en 10.5.3.2.2 de la CEI 61158-3-1.

NOTE 2 Les représentations spécifiques de ces composantes de séquence de primitive au sein d'un système final sont une question de vue locale, ne relevant pas du domaine d'application de la normalisation de systèmes ouverts. Cependant, l'Article 9 ne comprend pas une représentation normalisée de certaines de ces composantes lorsqu’elles sont communiquées entre des systèmes finals et une DLE LAS, et des mises en œuvre peuvent choisir d’utiliser localement la même représentation. Les spécifications formelles d'interface de programmation de DL, qui incluraient les détails d’un tel codage, pourraient normaliser cette représentation locale ou une représentation locale similaire.

La DLE doit déterminer la classe d’opération de programmation demandée, en se basant sur la définition de séquence et le type de programme demandé.

Si le paramètre Schedule Type (type de programme) a la valeur ONE-TIME ou PERIODIC, alors

a) si la séquence est composée d’un élément unique, et le paramètre Schedule Type a la valeur ONE-TIME, et le paramètre Desired Starting Time spécifie IMMEDIATE, et le paramètre sequence-priority spécifie une priorité inférieure ou égale à la priorité intrinsèque de l'élément unique, alors la DLE doit abouter une référence à la séquence demandée à la Sequence Priority de la file d’attente de services non programmés de la DLE, Q(US);

b) sinon si a) ne s’applique pas et la séquence consiste en un ou plusieurs éléments, tous étant des références à des DLCEP distants de la DLE, dont aucun n’est pour des DLC spécifiant une authentification de DLPDU EXTRA, alors la DLE doit

   1) former une SPDU équivalent à la séquence, de la même priorité que la séquence;

   2) inclure cette SPDU comme étant la DLSDU d'une DLPDU DT sans connexion de priorité TIME-AVAILABLE avec une adresse de DL de destination SHORT du LAS local, 0400_{16} (voir 4.3.3.2), et une adresse de DL source égale à l'adresse de DL NODE.0, V(TN).0, des fonctions de prise en charge de DL de la DLE d'envoi;

   3) abouter la DLPDU DT à la file d’attente de demandes utilisateur NODE de la DLE, Q_N(UR), à la priorité applicable;

4) abouter à la file d'attente de services non programmés de la DLE, à savoir Q(US), une référence à la Q_N(UR).

c) sinon si a) et b) ne s'appliquent pas, alors la DLE doit

1) utiliser l'adresse de DLSEP fournie par l'utilisateur de DLS, ou allouer une adresse de DLSEP s'il n'en était pas fourni une;

2) associer la séquence demandée à l'adresse de DLSEP, afin que la réception d'une DLPDU ES spécifiant que l'adresse de DLSEP entraînera l'exécution de la séquence associée;

3) calculer la durée ininterrompue maximale de la demande;

4) former une SPDU de demande de programmation avec une priorité égale à la Sequence-Priority de la demande utilisateur de DLS; où la SPDU demande que le jeton soit délégué à la séquence à l'adresse de DLSEP spécifiée, pour la durée calculée;

5) inclure cette SPDU dans une DLPDU DT et placer en file d'attente cette DLPDU pour la transmission comme spécifié de b)2) à b)4).

Si le paramètre de Schedule Type a la valeur REPETITIVE, alors

d) la DLE doit abouter à la file d'attente de services non programmés de la DLE, Q(US), une référence à la séquence demandée, à la Sequence Priority.

Il est permis, mais déconseillé, qu'une séquence conforme aux exigences de b) soit exécutée localement comme en a), c) ou d). Il est également permis, mais déconseillé, qu'une séquence conforme aux exigences en a) ou c) soit envoyée comme SPDU comme en b) à un LAS pour exécution distante; cette permission exige que la syntaxe de SPDU (voir Article 11) soit capable d'acheminer les informations nécessaires.

NOTE L'application du retard de confirmation de demande de DLS à une primitive «request» de DL-SCHEDULE-SEQUENCE est pour étude ultérieure.

8.4.3.2 Réception d'une primitive "request" de DL-CANCEL-SCHEDULE

Lorsque la DLE reçoit une demande de DL-CANCEL-SCHEDULE, elle doit déterminer la classe de séquence programmée annulée:

a) Si la séquence est programmée localement (voir 8.4.3.1a) ou d)), alors la DLE doit

1) retirer la référence à l'horaire de la file d'attente de services non programmés de la DLE, Q(US);

2) émettre une "confirm" de DL-CANCEL-SCHEDULE avec l'identifiant du programme annulé;

3) libérer l'identifiant de programme, le rendant disponible pour une réutilisation ultérieure.

b) Si la séquence est programmée centralement, mais exécutée localement (voir 8.4.3.1c)), alors la DLE doit

1) retirer la référence au programme de l'adresse de DLSEP associée au programme;

2) former une SPDU "cancel-schedule " (d'annulation de programme) (voir 9.3.4.4);

3) inclure cette SPDU comme étant la DLSDU d'une DLPDU DT sans connexion de priorité TIME-AVAILABLE avec une adresse de DL de destination SHORT du LAS local, 0400_{16} (voir 4.3.3.2), et une adresse de DL source égale à l'adresse de DL NODE.0, V(TN).0, des fonctions de prise en charge de DL de la DLE d'envoi;

4) abouter la DLPDU DT à la file d'attente de demandes utilisateur NODE de la DLE, Q_N(UR), à la priorité applicable;

5) abouter à la file d'attente de services non programmés de la DLE, à savoir Q(US), une référence à la Q_N(UR).

c) Si la séquence est programmée centralement et exécutée centralement (voir 8.4.3.1b)), alors la DLE doit former et envoyer une SPDU "cancel-schedule" comme spécifié de b)2) à b)5).

8.4.3.3 Réception d'une SPDU "a schedule-cancelled"

Si la DLE reçoit une SPDU "schedule-cancelled" (programme annulé) en réponse à une SPDU "cancel-schedule" (annuler programme), la DLE doit.

a) émettre une "confirm" de DL-CANCEL-SCHEDULE avec l'identifiant du programme annulé;
b) libérer l'identifiant de programme et l'adresse de DLSEP associée, le cas échéant, la/les rendant disponible(s) pour réutilisation ultérieure.

8.4.3.4 Annulation de programme à l'initiative du LAS

Si la DLE reçoit une SPDU schedule-cancelled issue du LAS, alors la DLE doit déterminer la classe de la séquence programmée annulée:

a) Si la séquence est programmée centralement, mais exécutée localement (voir 8.4.3.1c)), alors la DLE doit
   1) retirer la référence au programme de l'adresse de DLSEP associée au programme;
   2) émettre une "confirm" de DL-CANCEL-SCHEDULE avec l'identifiant du programme annulé;
   3) libérer l'identifiant de programme et l'adresse de DLSEP associée, les rendant disponibles pour réutilisation ultérieure.
b) Si la séquence est programmée centralement et exécutée centralement (voir 8.4.3.1b)), alors la DLE doit
   1) émettre une "confirm" de DL-CANCEL-SCHEDULE avec l'identifiant du programme annulé;
   2) libérer l'identifiant de programme, le rendant disponible pour une réutilisation ultérieure.

8.4.4 Fonctionnement du service Sequence subsetting

Les primitives du service sequence subsetting sont "request" et "confirm" de DL-SUBSET-SEQUENCE.

8.4.4.1 Réception d'une primitive "request" de DL-SUBSET-SEQUENCE

Lorsque la DLE reçoit une séquence de demande de DL-SUBSET-SEQUENCE,

a) si la séquence spécifiée n'est pas définie ou n'est pas définissable en sous-ensemble, alors la DLE doit retourner un statut d'erreur approprié et rejeter la demande;
b) autrement, lorsque a) ne s'applique pas, alors la DLE doit définir en sous-ensemble la séquence conformément à la demande et émettre une primitive "confirm" de DL-SUBSET-SEQUENCE avec l'identifiant de demande fourni par la "request" de DL-SUBSET-SEQUENCE.

8.4.5 Programmation implicite des demandes utilisateur de DLS

Lorsque la DLE reçoit une primitive "request" ou "response" d'utilisateur de DLS, et la politique de programmation de DL applicable (voir 4.2.6) est IMPLICIT, alors la DLE doit exécuter cette demande ou cette réponse dès que possible, en fonction de la priorité, implicite ou explicite, de la demande ou de la réponse. Si la demande ou réponse exige la transmission d'une ou plusieurs DLPDU, alors la DLE doit

a) sélectionner la file d'attente appropriée, Q_A(UR), QA(UR), à laquelle il convient d'abouter la demande:
   1) pour les procédures en mode sans connexion (voir 8.3), l'adresse de DLSAP d'envoi associée;

2) pour les procédures orientées connexion (voir 8.2),
   i) l'adresse de DLCEP locale associée pour les DLPDU DT lorsque la classe de DLCEP du DLCEP d'envoi est PEER ou PUBLISHER, ou
   ii) l'adresse de DLSAP (appelante ou de répondeur) d'envoi associée pour les DLPDU EC, DC et RC, ou
   iii) soit i), soit ii) pour les DLPDU DT lorsque la classe de DLCEP du DLCEP d'envoi est SUBSCRIBER;
3) pour les autres procédures (voir 8.4), la file d'attente désignée par la procédure;

b) abouter la demande, à la priorité appropriée, à cette Q_A(UR), et réajuster cette file d'attente, s'il y a lieu, afin de ne pas augmenter le nombre total d'éléments de file d'attente qui sont présents dans la troisième partition de la file d'attente, le cas échéant;
c) abouter à la file d'attente de services programmés de la DLE une référence à cette Q_A(UR), à la priorité appropriée.

9 Sous-protocole de prise en charge de DL

9.1 Généralités

Le présent paragraphe

a) définit les codages pour les PDU de prise en charge (les SPDU) de niveau supérieure (voir 4.1.3) requises pour prendre en charge le fonctionnement de LAS et le sous-protocole de programmation;
b) définit les composantes d'une séquence définie (voir la CEI 61158-3-1, 10.5.3.2.2);
c) définit les codages pour les PDU de prise en charge (les SPDU) de niveau supérieur (voir 4.1.3) requises pour prendre en charge diverses interrogations de configuration et de capacité utilisées par
   – les éléments de procédure "request" de DL-SUBSCRIBER-QUERY et "request" de DL-LISTENER-QUERY;
   – les ponts pour la maintenance de base de données de transmission;
   – les éléments spécialisés de surveillance, d'analyse et de gestion de DLL;
d) spécifie les éléments de procédure pour le traitement des SPDU reçues.

9.2 Vue d'ensemble du fonctionnement de LAS

Un LAS doit accomplir les fonctions suivantes:

a) Maintenance de liaison – le LAS
   1) détecte la présence d'une DLE nouvellement ajoutée sur la liaison locale, après quoi il active cette DLE;
   2) détecte qu'une DLE précédemment participante n'est plus active sur la liaison locale;
   3) maintient une liste de toutes les DLE qui sont opérationnelles sur la liaison locale;
   4) diffuse la liste des DLE opérationnelles à toutes les autres DLE "link-master" (maîtres de liaison) sur la liaison locale.
b) Distribution du temps – le LAS agit comme base de temps pour la liaison locale et envoie des DLPDU de distribution du temps à toutes les autres DLE sur la liaison locale afin de fournir un sens commun de temps de DL pour la liaison locale.
c) Circulation de jeton – le LAS envoie une DLPDU PT à chaque DLE active sur la liaison locale. Par répétition de cette opération, la DLE LAS fournit un jeton délégué qui "circule" successivement, habituellement dans l'ordre des adresses de DL NODE, vers toutes les DLE actives sur la liaison locale.
d) Exécution de programme – le LAS envoie des DLPDU sur la liaison locale selon ce que dicte un programme qui spécifie le temps de DL initial et la période de répétition pour

chacune de ces transmissions. Un programme initial (éventuellement vide) est fourni par la gestion DL, et peut être modifié ultérieurement par le LAS actif.

e) Transfert de LAS – le LAS transfère son rôle à une autre liaison DLE maîtresse de liaison, si cela est demandé par cette maîtresse de liaison ou par la gestion de DL.

Un LAS peut aussi accomplir les fonctions suivantes:

f) Construction de programme – le LAS peut traiter des demandes de programme issues d'autres DLE sur la liaison locale et modifier le programme existant, le cas échéant, pour satisfaire à ces demandes. La construction de programme doit prendre en compte le maximum-scheduled-traffic (le trafic programmé maximal), V(MST), (voir 4.7.5.5) et le maximum-scheduling-overhead (surdébit maximal de programmation), V(MSO), (voir 4.7.5.6).

g) Transfert de programme – le LAS peut envoyer l'état courant du programme de la liaison vers une autre DLE sur la liaison locale, soit comme mesure de secours, soit précédant un transfert intentionnel du rôle LAS. Le présent paragraphe spécifie les SPDU exigées pour transférer le programme préconstruit de la DLE LAS vers une autre DLE LM sur la liaison locale. Le format de ces SPDU ne prend pas en charge la mise à jour d'un tel programme transféré. Les procédures pour le transfert de programme sont spécifiées à l'Article 9.

Les procédures pour le fonctionnement du LAS et le fonctionnement de la prise en charge de DL sont spécifiées en 10.2; les procédures pour les demandes de construction de programme sont spécifiées en 8.4.3; les procédures pour la distribution de temps sont spécifiées en 8.4.1; les procédures pour la circulation de jeton sont spécifiées en 6.15; les procédures pour les demandes de construction de programme local sont spécifiées en 8.4.3. Les SPDU pour transférer des demandes de construction de programme à la DLE LAS sont spécifiées dans le présent paragraphe. Les procédures pour la construction de programme dépendent de la mise en œuvre et ne sont donc pas spécifiées.

Le présent paragraphe spécifie aussi des SPDU pour prendre en charge les ponts, les analyseurs de protocole et autres outils de diagnostic, (voir 9.3.2.7, 9.3.2.8, 9.3.2.9, 9.3.2.10, 9.3.6.4 et 9.3.6.5). Les procédures pour envoyer et recevoir ces SPDU dépendent de la mise en œuvre et ne sont donc pas spécifiées.

9.3 Définition de sous-protocole de prise en charge de DL

Le sous-protocole de prise en charge de DL définit les codages des Unités de données de protocole de prise en charge (SPDU) pour prendre en charge les besoins du fonctionnement de LAS, y compris la programmation et autres fonctions de DLE. D'autres fonctions de prise en charge DLE définies présentement incluent des mécanismes de partage d'informations relatives aux adresses et à des configurations limitées venant à l'appui

– des demandes de service DL-SUBSCRIBER-QUERY (voir 8.2.3) et de service DL-LISTENER-QUERY (voir 8.3.4);
– de la maintenance de base de données de filtrage au sein de ponts (voir Article 9));
– d'outils de prise en charge (analyseur de liaison) d'analyse de réseau.

Toute DLPDU envoyée vers ou par les fonctions de prise en charge de DL au sein d'une DLE, y compris une DLPDU adressée à une adresse de DL NODE, qui a un champ "données utilisateur" non vide, doit contenir comme "données utilisateur" une SPDU unique dont le codage et l'interprétation sont tels que décrits dans le présent paragraphe. Cette exigence inclut toute DLPDU adressée à une adresse de DLSAP désignant une fonctionnalité de LAS, telle qu'une adresse de DL de liaison locale 0400_{16}. Elle inclut également toute DLPDU PR ou TL, l'une comme l'autre ayant toujours un champ "données utilisateur".

9.3.1 Définitions communes

9.3.1.1 En-tête de SPDU

Le premier octet de chaque SPDU doit spécifier un en-tête commun à toutes les SPDU, tel que spécifié dans le Tableau 80. Il doit contenir:

a) une classification de la SPDU;

b) pour certaines SPDU, la version de protocole, dont la valeur est spécifiée à l'Article 7.

Tableau 80 – 1er octet de SPDU: Classe de SPDU, et version de protocole ou sous-classe

Classe de SPDU					Version de protocole ou sous-classe		
CCCCC ≤ 11011					VVV		
CCCCC ≥ 11100					SSS		
7	6	5	4	3	2	1	0

Lorsque la classe de SPDU est comprise dans la plage 00 à $1B_{16}$, alors la version de protocole, VVV, dont la valeur est spécifiée au début de l'Article 7, doit être présente et codée comme en 7.1a)4).

NOTE 1 La plage de valeurs qui en résulte pour l'octet s'étend de 00 à DF_{16}.

Lorsque la classe de SPDU est comprise dans la plage $1C_{16}$ à $1F_{16}$, alors le champ "sous-classe", SSS, doit être présent.

NOTE 2 La plage de valeurs qui en résulte pour l'octet s'étend de $E0_{16}$ à FF_{16}.

Certaines des SPDU ne contiennent que l'octet d'en-tête. Les SPDU se répartissent entre cinq classes:

a) les SPDU prenant en charge l'initialisation normale de liaison, l'activation de DLE, et autres fonctions de maintenance de liaison,

b) les SPDU prenant en charge le transfert de LAS,

c) les SPDU prenant en charge la construction de programme,

d) les SPDU prenant en charge le transfert de programme,

e) les SPDU prenant en charge les caractéristiques autres que LAS du protocole DL, telles que les demandes de DL-SUBSCRIBER-QUERY et DL-LISTENER-QUERY, les ponts, la gestion de DL distante et les analyseurs de protocole DL, la gestion de DL distante, et les analyseurs de protocole DL à usage spécial.

9.3.1.2 Boolean

Tous les éléments Boolean (booléens) partagent un codage commun:

0 FALSE (c'est-à-dire: faux)

1 TRUE. (c'est-à-dire: vrai)

9.3.2 SPDU Link-maintenance (maintenance de liaison)

9.3.2.1 SPDU Probe-response (sondage de réponse)

La SPDU de sondage de réponse est envoyée dans une DLPDU Probe Response (Sonder la réponse (PR)) (voir 6.14.1). La SPDU de sondage de réponse doit être formatée comme spécifié dans le Tableau 81. Les valeurs multi-octets doivent être codées avec l'octet de poids fort de la valeur codé dans l'octet d'indice le plus faible du champ multi-octet:

Tableau 81 – Probe-response SPDU

Indice d'octet	Contenu de champ							
1	0000 0VVV							
2	Versions de protocole de DL prises en charge							
3	N	F	CC		V(MICD)			
4								
5 **6**	V(MRD) × V(ST)							
7	V(MID)							
8	V(RID)							
	7	6	5	4	3	2	1	0

a) L'octet d'en-tête de la SPDU, octet 1, doit être codé comme 0000 0VVV, qui l'identifie comme étant une SPDU DL-probe-response (sondage de réponse de DL).

Tableau 82 – Versions de protocole de DL prises en charge

Versions de protocole de DL prises en charge							
V7	V6	V5	V4	V3	V2	V1	V0
7	6	5	4	3	2	1	0

b) L'octet 2 doit spécifier la/les version(s) du protocole de DL pris en charge, comme représenté dans le Tableau 82, où chaque valeur V_K doit être codée comme étant un Boolean indiquant si, oui ou non, la version de protocole *K* est prise en charge.

Tableau 83 – SPDU PR: 3ème et 4ème octets

Circulation de jeton non nécessaire	Fonctionnement de Facteur d'utilisation fractionnaire	Classe fonctionnelle de DLE attendue		Maximum-inactivity-to-claim-LAS-delay (bits de poids fort)			
N	F	CC		V(MICD) / 256			
15	14	13	12	11	10	9	8

Maximum-inactivity-to-claim-LAS-delay (bits de poids faible)							
V(MICD) modulo 256							
7	6	5	4	3	2	1	0

c) Les octets 3 et 4 doivent spécifier, comme décrit dans le Tableau 83:

1) l'absence de besoin de circulation de jeton de la DLE sans une demande explicite, codée comme un Booléen, N: 0 (non, la circulation de jeton est nécessaire) ou 1 (oui, la circulation de jeton n'est pas nécessaire);

2) si, oui ou non, la DLE fonctionnera comme une DLE FDC qui peut être censée ne pas réagir à certaines interrogations de maintenance de liaison de liste active, et si, oui ou non, il convient d'inclure cette DLE dans la liste de non-réponses attendues, V(ENRL) (voir 4.7.5.4), ce qui est codé comme un Boolean, F;

3) la classe fonctionnelle attendue du fonctionnement de DLE: basic, link master ou bridge (voir 4.6), pour permettre au LAS courant de déterminer s'il convient

d'interroger davantage les capacités de la nouvelle DLE et potentiellement lancer un transfert du rôle LAS vers la nouvelle DLE, codée comme:

00 non rapportée;

NOTE Cette variante existe exclusivement pour des besoins de rétrocompatibilité avec les éditions antérieures de la CEI 61158-4.

01 Basic;

10 Link Master;

11 Bridge;

NOTE Les trois dernières valeurs ont le même codage que spécifié en 9.3.2.8f)1).

4) si la DLE de réponse est un link master (maître de liaison), alors le paramètre maximum-inactivity-to-claim-LAS-delay, V(MICD), requis de cette DLE défini en 4.7.5.19, sinon zéro.

NOTE Si cette valeur est supérieure à la valeur configurée de V(MICD) de la liaison, alors la DLE de réponse n'est pas capable de fonctionner comme LAS sur la liaison comme spécifié à l'Article 9.

d) Les octets 5 et 6 doivent spécifier la valeur requise du paramètre maximum-response-delay-in-octets de la DLE de réponse, à savoir V(MRD) × V(ST), définie en 4.7.1.3 et 4.7.1.1. Si cette valeur, qui reflète la capacité de l'appareil, est supérieure à la valeur configurée du paramètre maximum-response-delay-in-octets de la liaison, à savoir V(MRD) × V(ST), alors le nœud de réponse n'est pas capable de fonctionner sur la liaison locale.

e) L'octet 7 doit spécifier la valeur requise du paramètre minimum-inter-DLPDU-delay de la DLE de réponse, à savoir V(MID) défini en 4.7.1.12. Si cette valeur, qui reflète la capacité de l'appareil, est supérieure à la valeur configurée de V(MID) de la liaison, alors la DLE de réponse n'est pas capable de fonctionner sur la liaison locale.

f) L'octet 8 doit spécifier la valeur de la variable de l'identifiant aléatoire de la DLE d'envoi, à savoir V(RID) (voir 4.7.1.18), qui est randomisée de nouveau à la réception d'une DLPDU PN adressée à la DLE.

9.3.2.2 SPDU Node-activation (activation de nœud)

La SPDU d'activation de nœud spécifie les paramètres de configuration de DL pour la liaison de la DLE de réception. Son objectif principal est de permettre une initialisation précoce des services DL et ainsi permettre à des piles de protocoles de couche supérieure, y compris toute pile de protocoles de gestion de l'OSI, de devenir pleinement opérationnelle.

La SPDU d'activation de nœud doit être formatée comme spécifié dans le Tableau 84. Les valeurs multi-octets doivent être codées avec l'octet de poids fort de la valeur codé dans l'octet d'indice le plus faible du champ multi-octet:

Tableau 84 – SPDU Node-activation

Indice d'octet	Contenu de champ							
1	0000 1VVV							
2	valeur pour V(MRC)				valeur pour V(NRC)			
3	valeur pour V(PhLO)							
4	réservé					valeur pour V(TSC)		
5 **6**	valeur pour V(MEP)							
7 **8**	valeur pour V(TL)							
9 **10**	valeur pour V(NDL)							
11	V(RID)							
	7	6	5	4	3	2	1	0

a) L'octet d'en-tête de la SPDU, octet 1, doit être codé comme 0000 1VVV, qui l'identifie comme étant une SPDU d'activation de nœud**.**

b) L'octet 2 doit spécifier en deux quartets la valeur de la variable "maximum-retry-count" de la DLE de réception, à savoir V(MRC), telle que spécifiée en 4.7.1.5, et la valeur de la variable network-retry-count de la DLE de réception, à savoir V(NRC), telle que spécifiée en 4.7.1.6. Les deux valeurs sont des propriétés de la liaison locale.

c) L'octet 3 doit spécifier la valeur de "per-DLPDU-PhL-overhead", telle que spécifiée en 4.7.1.2. Lorsque la valeur pour cette variable n'est pas configurée, une valeur correspondant à la propre PhE associée de la DLE LAS, telle que configurée alors, doit être utilisée.

Tableau 85 – SPDU Node-activation 4ème octet

reserved (réservé)					V(TSC)		
7	6	5	4	3	2	1	0

d) L'octet 4 doit spécifier, comme montré dans le Tableau 85:

   1) réservé pour utilisation future, codé comme 00000;

   2) la valeur devant être utilisée par la DLE de réception comme étant la variable "time-synchronization-class" de la liaison locale, à savoir V(TSC), telle que spécifiée en 4.7.1.25, en utilisant le codage spécifié en 7.6 b)1).

e) Les octets 5 et 6 doivent spécifier la valeur de la variable "DL-MAC-address-embedding-prefix" de la DLE de réception, à savoir V(MEP), telle que spécifiée en 4.7.1.10.

f) Les octets 7 et 8 doivent spécifier la valeur de la variable "this-link" de la DLE de réception, à savoir V(TL), telle que spécifiée en 4.7.1.9. Lorsque la valeur appropriée pour cette variable est inconnue, la DLE LAS doit utiliser une valeur de zéro.

g) Les octets 9 et 10 doivent spécifier la valeur de la variable "network-DLPDU-lifetime" de la DLE de réception, à savoir V(NDL), telle que spécifiée en 4.7.1.7.

h) L'octet 11 doit spécifier la valeur de la variable "random identifier" de la DLE adressée, à savoir V(RID) (4.7.1.18), telle qu'enregistrée par le LAS à partir d'une SPDU probe-response reçue précédemment.

9.3.2.3 SPDU LAS-data-base-status

La SPDU LAS-data-base-status est envoyée par le LAS à toutes les DLE LM sur la liaison locale afin d'éditer le dernier numéro de version pour le programme actif et le dernier numéro de révision pour la liste active de la liaison locale, V(LL) (voir 4.7.5.2). Elle est également envoyée dans chaque DLPDU TL. La SPDU LAS-data-base-status doit être codée comme spécifié dans le Tableau 86. Les valeurs multi-octets doivent être codées avec l'octet de poids fort de la valeur codé dans l'octet d'indice le plus faible du champ multi-octet:

Tableau 86 – SPDU LAS-data-base-status

Indice d'octet	Contenu de champ
1	1111 0000
2	type de programme
3 **4**	numéro de version de programme
5 **6**	numéro de version de liste active

a) L'octet d'en-tête de la SPDU, octet 1, doit être codé comme 1111 0000, spécifiant une SPDU LAS-data-base-status (statut de base de données de LAS).

Tableau 87 – SPDU LAS-data-base-status: 2ème octet

Type de programme							
T	réservé					D	S
7	6	5	4	3	2	1	0

b) L'octet 2 doit spécifier, comme montré dans le Tableau 87:
   1) la capacité du LAS à transférer son programme, T, codée comme un Boolean: 0 (non, le LAS n'est pas capable) ou 1 (oui, le LAS est capable de transférer son programme à d'autres DLE LM);
   2) réservé pour utilisation future, codé comme 00000;
   3) si, oui ou non, la totalité ou une partie du programme actif a été dynamiquement construite par le LAS, D, codé comme un Boolean;
   4) si, oui ou non, la totalité ou une partie du programme actif a été statiquement construite par la gestion de DL, S, codé comme un Boolean;

   NOTE Si le LAS a un programme actif, les deux sous-champs D et S ne peuvent pas être codés comme étant 0.

c) Les octets 3 et 4 doivent spécifier le numéro de version du programme couramment actif. S'il n'y a pas de programme actif, alors sa valeur doit être zéro; autrement, elle doit avoir une valeur autre que zéro.

d) Les octets 5 et 6 doivent spécifier le numéro de révision de la liste active courante. Cette valeur est toujours différente de zéro et elle est incrémentée (modulo 2^{16}) chaque fois que le LAS détecte un changement dans la liste active.

9.3.2.4 SPDU Live-list-change

La SPDU Live-list-change est envoyée par le LAS à toutes les DLE LM sur la liaison locale, chaque fois que le LAS a détecté un changement dans l'état d'une ou plusieurs DLE. Cette SPDU doit être codée comme spécifié dans le Tableau 88. Les valeurs multi-octets doivent être codées avec l'octet de poids fort de la valeur codé dans l'octet d'indice le plus faible du champ multi-octet.

Tableau 88 – SPDU Live-list- change

Indice d'octet	Contenu de champ
1	1111 0001
2 **3**	numéro de version de liste active
4 **5**	statut de DLE
…	statut de DLE

a) L'octet d'en-tête de la SPDU, octet 1, doit être codé comme 1111 0001, spécifiant une SPDU Live-list-change (changement de liste active).

b) Les octets 2 et 3 doivent spécifier le numéro de révision de la liste active. Il est incrémenté chaque fois que le LAS détecte un changement dans la liste active et prépare la SPDU live-list, mais il n'est pas incrémenté si le changement est ajouté à une SPDU qui n'a pas encore été éditée par le LAS. La valeur dans cette SPDU est le numéro de révision qui s'applique à la liste active immédiatement après les changements spécifiés dans cette SPDU. Lorsque ce numéro atteint sa valeur maximale, la prochaine fois qu'il est incrémenté, il sera incrémenté à 1 (ainsi ce numéro est toujours différent de zéro).

c) Le reste de la SPDU est une matrice d'éléments de deux octets spécifiant le statut de DLE et structurée telle que montrée dans le Tableau 89:

   1) Le premier octet de chaque élément doit spécifier l'adresse de DL NODE de la DLE qui est affectée par le changement.

   2) Le second octet de chaque élément doit spécifier l'état de la DLE, codé comme:

      i) N, l'absence de besoin de circulation de jeton de la DLE, telle que retournée par cette DLE dans une SPDU PR envoyée précédemment (voir 9.3.2.1c)1));

      ii) F, si, oui ou non, la DLE est une DLE FDC, telle que retournée par cette DLE dans une SPDU PR envoyée précédemment (voir 9.3.2.1c)2));

      ii) réservé pour utilisation future, codé comme 0000;

      iv) SS, le dernier statut observé de cette DLE, codé comme

         01: absente,

         10: présente, mais supposée maintenant endormie, qui est seulement possible pour les DLE FDC, ou

         11: présente et éveillée.

Tableau 89 – Structure du statut de DLE

<table>
<tr><th>Sous-indice d'octet</th><th colspan="8">Contenu de sous-champ</th></tr>
<tr><td>1</td><td colspan="8">Adresse de DL NODE de la DLE décrite</td></tr>
<tr><td>2</td><td>N</td><td>F</td><td colspan="4">réservé</td><td colspan="2">SS</td></tr>
<tr><td></td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
</table>

La longueur de cette SPDU peut approcher 128 octets.

9.3.2.5 SPDU Live-list-request

Cette SPDU demande que la DLE LAS courante édite la liste active complète. La SPDU Live-list-request doit être codée avec le format spécifié dans le Tableau 80, comme suit:

a) L'octet d'en-tête de la SPDU, octet 1, doit être codé comme 1111 0010, spécifiant une SPDU Live-list-request (demande de liste active).

9.3.2.6 SPDU Live-list-detail

La SPDU Live-list-detail SPDU est envoyée par le LAS à toutes les DLE LM sur la liaison locale. Cette SPDU doit être codée comme spécifié dans le Tableau 90. Les valeurs multi-octets doivent être codées avec l'octet de poids fort de la valeur codé dans l'octet d'indice le plus faible du champ multi-octet.

Tableau 90 – SPDU Live-list-detail

Indice d'octet	Contenu de champ
1	1111 0011
2	Adresse de DL NODE de la DLE LM de LAS
3 4	numéro de version de liste active
5	V(FUN)
6	V(NUN)
7 …	Statut de vivacité
… …	Type de DLE

a) L'octet d'en-tête de la SPDU, octet 1, doit être codé comme 1111 0011, spécifiant une SPDU Live-list-detail (détail de liste active).

b) L'octet 2 doit spécifier l'adresse de DL NODE de la DLE link-master qui fournit actuellement la fonctionnalité de LAS.

c) Les octets 3 et 4 doivent spécifier le numéro de révision de la liste active courante.

d) L'octet 5 doit spécifier la valeur courant du paramètre first-unpolled-node (premier nœud non interrogé), à savoir V(FUN) (voir 4.7.5.15).

e) L'octet 6 doit spécifier la valeur courante du paramètre number-of-consecutive-unpolled-nodes (nombre de nœuds non sondés consécutifs), à savoir V(NUN), (voir 4.7.5.16).

f) L'octet 7 et les suivants doivent spécifier l'état de vivacité d'un nombre de DLE pouvant atteindre 240, un bit par DLE, dans l'ordre décroissant des adresses de DL NODE, codé comme une matrice de Boolean: 0 (la DLE ne communique pas et est présumée "morte"), 1 (la DLE communique et est donc "vivante").

   Les DLE qui ne sont pas censées être connectées à la liaison, telles que spécifiées par les variables V(FUN) et V(NUN), ne sont pas incluses dans ce statut. La matrice de Boolean doit être ordonnée afin que le bit 7 de l'octet 7 spécifie le statut de toute DLE associée à la plus haute adresse (FF_{16}), le bit 6 doit spécifier le statut de toute DLE associée à la plus haute adresse (FE_{16}) suivante, et ainsi de suite. Les bits non utilisés du dernier octet doivent être mis à 0.

g) Ce champ spécifie le statut de FDC pour un nombre de DLE pouvant atteindre 240 au maximum, un bit par DLE, dans l'ordre décroissant des adresses NODE, codé comme une matrice de Boolean: 0 (LA DLE n'est pas une DLE FDC) ou 1 (LA DLE est une DLE FDC). Les DLE qui ne sont pas censées être connectées à la liaison, telles que spécifiées par les variables V(FUN) et V(NUN), ne sont pas incluses dans ce statut. La matrice de Boolean doit être ordonnée afin que le bit 7 du premier octet de ce champ spécifie le statut de toute DLE associée à la plus haute adresse (FF_{16}), le bit 6 doit spécifier le statut de toute DLE associée à la plus haute adresse (FE_{16}) suivante, et ainsi de suite. Le type de DLE pour toute DLE avec un statut de vivacité de 0, tel qu'indiqué par le Boolean correspondant codé dans f), doit être codé comme 0.

   Les octets de queue avec la valeur zéro doivent être omis de ce champ et de la SPDU. Par conséquent, lorsque la liaison locale n'a pas de DLE FDC, alors ce champ tout entier doit être omis de la SPDU.

9.3.2.7 SPDU DL-conformance-query

La SPDU DL-conformance-query demande à la DLE de réception d'envoyer à DLE demandeuse une SPDU de conformité de DL spécifiant les classes de conformité de DL de la DLE répondeuse. Ses buts sont

1) permettre une nouvelle gestion de DL d'une DLE LAS pour évaluer l'impact des DLE qui sont déjà connectées à la liaison locale, facilitant la récupération du rôle LAS après l'échec de la DLE LAS antérieure, et pour faciliter le transfert potentiel du rôle LAS vers une DLE plus appropriée;
2) permettre une gestion de DL distante pour interroger les DLE sur la liaison étendue, en fournissant une interrogation (et une réponse) qui peut être transmise par l'intermédiaire d'un pont;
3) faciliter une initialisation d'analyseur de protocole de DL après raccordement à une liaison locale en fonctionnement.

La SPDU DL-conformance-query doit être codée avec le format spécifié dans le Tableau 80, comme suit:

a) L'octet d'en-tête de la SPDU, octet 1, doit être codé comme 1110 1100, spécifiant une SPDU DL-conformance-query (interrogation de conformité de DL).

9.3.2.8 SPDU DL-conformance-reply

La SPDU DL-conformance-reply doit être formatée comme spécifié dans le Tableau 91. Les valeurs multi-octets doivent être codées avec l'octet de poids fort de la valeur codé dans l'octet d'indice le plus faible du champ multi-octet.

Tableau 91 – SPDU DL-conformance-reply

Indice d'octet	Contenu de champ
1	0010 0VVV
2	Versions de protocole de DL prises en charge
3 ... 6	Codage de conformité de DL

a) L'octet d'en-tête de la SPDU, octet 1, doit être codé comme 0010 0VVV, qui l'identifie comme étant une SPDU DL-conformance-reply (réponse de conformité de DL).

Tableau 92 – Versions de protocole de DL prises en charge

Versions de protocole de DL prises en charge							
V_7	V_6	V_5	V_4	V_3	V_2	V_1	V_0
7	6	5	4	3	2	1	0

b) L'octet 2 doit spécifier la/les version(s) du protocole de DL pris en charge, comme représenté dans le Tableau 92, où chaque valeur V_K doit être codée comme étant un Boolean indiquant si, oui ou non, la version de protocole *K* est prise en charge.

Tableau 93 – Codage de conformité de DL (partie 1)

Caractéristiques de livraison de données étendues sans connexion prises en charge		**Priorités de DL prises en charge**	**Prise en charge de marqueur temporel d'opportunité**	**Types d'opportunité pris en charge**			
transfert d'unitdata confirmé de DLE distante	échange d'unitdata	priorité multiple	Prise en charge de DLSDU marquée du temps de DL	TRANSPARENT (TRANSPARENTE)	RESIDENCE (RESIDENCE)	UPDATE (MISE A JOUR)	SYNCHRONIZED (SYNCHRONISEE)
C	X	M	D	T	R	U	S
7	6	5	4	3	2	1	0

c) L'octet 3 doit spécifier, comme montré dans le Tableau 93:

1) la prise en charge de la DLE pour les caractéristiques de livraison de données étendues en mode sans connexion, codée comme un ensemble de deux Boolean;
2) la prise en charge de la DLE pour des priorités multiples DL, codée comme un Boolean;
3) la prise en charge de la DLE pour des marqueurs temporels de tampon sur des DLC, codée comme un Boolean;
4) les types pris en charge de la DLE pour le calcul d'opportunité de DLC, codés comme un ensemble de Boolean.

Tableau 94 – Codage de conformité de DL (partie 2)

Prise en charge dans le sens abonnés vers éditeur	**Caractéristiques de livraison de données de DLCEP prises en charge**		**Rapport maximal DLSDU:données de DLPDU pris en charge**				
X	FF		RRRRR				
7	6	5	4	3	2	1	0

d) L'octet 4 doit spécifier, comme montré dans le Tableau 94:

1) la prise en charge de la DLE pour le sens "abonnés vers éditeur" de transfert de DLSDU d'homologues multiples, X, telle que spécifiée à l'Article 9 et en 11.9.2.3, qui doit être codée comme un Boolean;
2) les caractéristiques de livraison de données de DLCEP prises en charge de la DLE, FF, telles que spécifiées à l'Article 9 et en 11.9.2.2, qui doivent être codées comme étant un de moins que le numéro d'élément pertinent en 11.9.2.2, donc avec une valeur de 0 à 3;
3) le rapport maximal DLSDU:données de DLPDU pris en charge de la DLE pour les DLCEP ORDERED DLCEP, à savoir RRRRR, tel que spécifié à l'Article 9 et en 11.9.2.4, qui doit être codé comme la valeur applicable maximale prise en charge issue de 11.9.2.4, avec une valeur de 0 à 16.