

DIN EN 61158-3-1

ICS 35.100.20; 35.160

Ersatz für
DIN EN 61158-3-1:2008-09
Siehe Anwendungsbeginn

**Industrielle Kommunikationsnetze –
Feldbusse –
Teil 3-1: Dienstfestlegungen des Data Link Layer (Sicherheitsschicht) –
Typ 1-Elemente
(IEC 61158-3-1:2014);
Englische Fassung EN 61158-3-1:2014, nur auf CD-ROM**

Industrial communication networks –
Fieldbus specifications –
Part 3-1: Data-link layer service definition – Type 1 elements
(IEC 61158-3-1:2014);
English version EN 61158-3-1:2014, only on CD-ROM

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
(CEI 61158-3-1:2014);
Version anglaise EN 61158-3-1:2014, seulement en CD-ROM

Gesamtumfang 135 Seiten

DKE Deutsche Kommission Elektrotechnik Elektronik Informationstechnik im DIN und VDE

Anwendungsbeginn

Anwendungsbeginn für die von CENELEC am 2014-09-17 angenommene Europäische Norm als DIN-Norm ist 2015-07-01.

Für DIN EN 61158-3-1:2008-09 besteht eine Übergangsfrist bis 2017-09-17.

Nationales Vorwort

Vorausgegangener Norm-Entwurf: E DIN EN 61158-300:2013-08.

Für dieses Dokument ist das nationale Arbeitsgremium K 956 „Feldbus“ der DKE Deutsche Kommission Elektrotechnik Elektronik Informationstechnik in DIN und VDE (www.dke.de) zuständig.

Die enthaltene IEC-Publikation wurde vom SC 65C „Industrial networks“ erarbeitet.

Das IEC-Komitee hat entschieden, dass der Inhalt dieser Publikation bis zu dem Datum (stability date) unverändert bleiben soll, das auf der IEC-Website unter „<http://webstore.iec.ch>“ zu dieser Publikation angegeben ist. Zu diesem Zeitpunkt wird entsprechend der Entscheidung des Komitees die Publikation

- bestätigt,
- zurückgezogen,
- durch eine Folgeausgabe ersetzt oder
- geändert.

Für den Fall einer undatierten Verweisung im normativen Text (Verweisung auf eine Norm ohne Angabe des Ausgabedatums und ohne Hinweis auf eine Abschnittsnummer, eine Tabelle, ein Bild usw.) bezieht sich die Verweisung auf die jeweils neueste gültige Ausgabe der in Bezug genommenen Norm.

Für den Fall einer datierten Verweisung im normativen Text bezieht sich die Verweisung immer auf die in Bezug genommene Ausgabe der Norm.

Der Zusammenhang der zitierten Normen mit den entsprechenden Deutschen Normen ergibt sich, soweit ein Zusammenhang besteht, grundsätzlich über die Nummer der entsprechenden IEC-Publikation. Beispiel: IEC 60068 ist als EN 60068 als Europäische Norm durch CENELEC übernommen und als DIN EN 60068 ins Deutsche Normenwerk aufgenommen.

Das Präsidium des DIN hat mit Präsidialbeschluss 1/2004 festgelegt, dass DIN-Normen, deren Inhalt sich auf internationale Arbeitsergebnisse der Informationsverarbeitung gründet, unter bestimmten Bedingungen allein in englischer Sprache veröffentlicht werden dürfen. Diese Bedingungen sind für die vorliegende Norm erfüllt.

Da sich die Benutzer der vorliegenden Norm der englischen Sprache als Fachsprache bedienen, wird die Englische Fassung der EN 61158-3-1 veröffentlicht. Zu deren Abschnitt 1, der den Anwendungsbereich festlegt, und Abschnitt 3, der die Begriffe festlegt, wurde eine Übersetzung angefertigt und als Nationaler Anhang NA der vorliegenden Norm hinzugefügt. Für einige der verwendeten Begriffe existieren keine gebräuchlichen deutschen Benennungen, da sich die deutschen Anwender in der Regel ebenfalls der englischen Benennungen bedienen. Diese Norm steht nicht in unmittelbarem Zusammenhang mit Rechtsvorschriften und ist nicht als Sicherheitsnorm anzusehen.

Einführung

Ein Feldbus ist ein digitaler serieller Datenbus für die Kommunikation zwischen Geräten der industriellen Automatisierungstechnik wie beispielsweise Messeinrichtungen, Stellgliedern, Reglern und speicherprogrammierbaren Steuerungen. Die Normen der Reihe DIN EN 61158 legen eine Anzahl von Feldbus-Protokolltypen fest. Die vorliegende Norm legt die Dienste fest, die durch die Data-link Layer der einzelnen Typen zur Verfügung gestellt werden.

Änderungen

Gegenüber DIN EN 61158-3-1:2008-09 wurden folgende Änderungen vorgenommen:

- a) Elemente;
- b) redaktionelle Überarbeitung.

Frühere Ausgaben

DIN EN 50170/A1: 2002-11
DIN EN 61158-3: 2005-05
DIN EN 61158-3-1: 2008-09

Nationaler Anhang NA (informativ)

1 Anwendungsbereich^{N1)}

1.1 Allgemeines

Dieser Teil der IEC 61158 liefert allgemeine Elemente für die Kommunikation von zeitkritisch basiertem Datentransfer zwischen Geräten in einer Automatisierungsumgebung. Der Begriff "zeitkritisch" (en: time-critical) wird verwendet um die Präsenz eines Zeitfensters darzustellen, in dem eine oder mehrere festgelegte Aktivitäten mit einem vorherbestimmten Grad an Gewissheit erledigt werden. Das Scheitern der Erledigung der festgelegten Aktivität innerhalb des Zeitfensters birgt das Risiko eines Ausfalls der dienstanfordernden Anwendung und damit verbundenen Risiken für Geräte, Anlagen und möglicherweise Menschenleben.

Diese Norm definiert auf eine abstrakte Art und Weise den von außen sichtbaren Dienst, der vom Typ 1 des Feldbus Data-Link-Layer bereitgestellt wird, in Form von:

- a) Aktivitäten und Ereignissen der Dienstprimitiven,
- b) den mit den Aktivitäten und Ereignissen der Dienstprimitiven verbundenen Parametern sowie die Form, die sie einnehmen und
- c) der Wechselbeziehung zwischen diesen Aktivitäten und Ereignissen und ihren gültigen Abläufen.

Die Zielsetzung dieser Norm ist es, Dienste zu definieren, die bereitgestellt werden für

- den Typ 1 Feldbus Application-Layer an der Grenze zwischen dem Application Layer und dem Data-Link-Layer des Feldbus-Referenzmodells,
- dem Systemmanagement an der Grenze zwischen dem Data-Link-Layer und dem Systemmanagement des Feldbus-Referenzmodells.

1.2 Spezifikationen

Das hauptsächliche Ziel dieser Norm ist es, für zeitkritische Kommunikation geeignete Eigenschaften der konzeptionellen Data-Link-Layer-Dienste festzulegen und folglich die Ergänzung des OSI-Basisreferenzmodell mit der Anleitung der Entwicklung der Data-Link-Protokolle für die zeitkritische Kommunikation. Ein zweites Ziel ist das Bereitstellen von Migrationspfade von bisherigen bestehenden industriellen Kommunikationsprotokollen.

Diese Spezifikation darf als Basis für formelle DL-Programmierschnittstellen verwendet werden. Nichtsdestotrotz ist es keine formelle Programmierschnittstelle und jede derartige Schnittstelle hat Implementierungsanforderungen zu erfüllen, die nicht von dieser Spezifikation abgedeckt werden und Folgendes beinhalten:

- a) die Größen und Oktett-Folgen von verschiedenen multi-oktett Dienstparametern,
- b) die Korrelation von gepaarten Anfrage- und Indikationsprimitiven oder Anweisungs- und Antwortprimitiven.

1.3 Konformität

Diese Norm legt keine individuellen Implementierungen oder Produkte fest, noch beschränkt sie die Implementierungen von Data-Link-Einheiten innerhalb eines industriellen Automatisierungssystems.

Es gibt keine Konformität von Geräten zu dieser Data-Link-Layer Dienstdefinitionsnorm. Stattdessen wird Konformität durch die Implementierung des entsprechenden Data-Link-Protokolls erreicht, welches in dieser Norm beschriebene Typ 7 der Data-Link-Layer Dienste erfüllt.

3.3 Begriffe zu den Diensten der Sicherungsschicht

3.3.1

Bridge, DL-Router

(en: bridge, DL-router)

DL-Relais-Entität, die selektive Store-and-Forward- sowie Routing-Funktionen ausführt,

^{N1)} Die Nummerierung aus IEC 61158-1 wurde beibehalten.

- a) um zwei oder mehrere einzelne DL-Subnetzwerke (Verbindungszweige) zu einem einzigen DL-Subnetzwerk (dem erweiterten Verbindungszweig) zu verbinden und,
- b) um eine Möglichkeit bereitzustellen, über die zwei Endsysteme miteinander kommunizieren können, wenn mindestens eines der Endsysteme regelmäßig wiederkehrend nicht auf das verbundene DL-Subnetzwerk reagiert,

und die auch für die Zeitsynchronisation zwischen den Verbindungszweigen, zu denen weitergeleitet wird, sorgt

3.3.2

DLCEP-Adresse

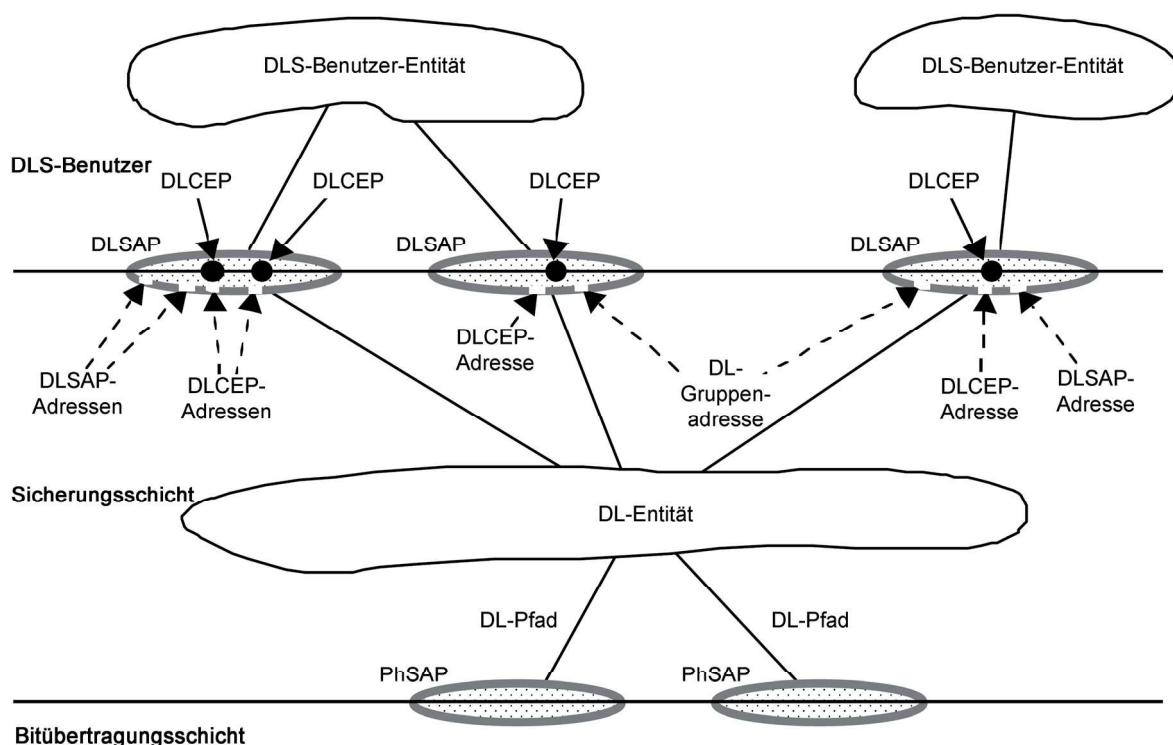
(en: DLCEP-address)

DL-Adresse, die entweder

- a) einen DL-Verbindungsendpunkt von einem Peer oder,
- b) einen DL-Verbindungsendpunkt eines Multi-Peer Publishers und implizit den entsprechenden Satz von Verbindungsendpunkten der DL-Subscriber kennzeichnet,

wobei jeder DL-Verbindungsendpunkt innerhalb eines eindeutigen DLSAP vorhanden und einer entsprechenden eindeutigen DLSAP-Adresse zugeordnet ist

ANMERKUNG 1 zum Begriff: Hierbei handelt es sich um eine Erweiterung der Verwendung von DL-Adressen gegenüber der in ISO/IEC 7498-3 festgelegten Verwendung (siehe Bild 1).



ANMERKUNG 1: DLSAPs und PhSAPs werden als Ellipsen dargestellt, die sich über die Grenze zweier benachbarter Schichten erstrecken.

ANMERKUNG 2: DL-Adressen werden als gekennzeichnete, kleine Lücken (Zugangspunkte) im DLL-Teil eines DLSAP dargestellt. Eine DLCEP-Adresse kennzeichnet auch einen bestimmten Punkt des Informationsflusses (seinen DLCEP) innerhalb des DLSAP.

ANMERKUNG 3: Eine einzelne DL-Entität kann mehrere DLSAP-Adressen und DL-Gruppenadressen, die einem einzelnen DLSAP zugeordnet sind, besitzen.

ANMERKUNG 4: In Bild 1 sind auch die Beziehungen zwischen DL-Pfaden und PhSAPs dargestellt.

Bild 1 – Beziehungen von DLSAPs, DLSAP-Adressen, DLCEPs, DLCEP-Adressen, DLSEPs-Adressen und DL-Gruppenadressen

3.3.3

DL-Segment, Verbindungszweig, lokaler Verbindungszweig

(en: DL-segment, link, local link)

einzelnes DL-Subnetzwerk, in dem alle verbundenen DLEs direkt, d. h. ohne zwischengeschaltete DL-Weitergabe, kommunizieren können, solange alle an einem Kommunikationsvorgang teilnehmenden DLEs während der Dauer(n) der Kommunikationsversuche gleichzeitig auf das DL-Subnetzwerk reagieren können

3.3.4

DLSAP

(en: DLSAP)

eindeutiger Punkt, an dem eine einzelne DL-Entität einer einzelnen Entität auf höherer Schicht DL-Dienste bereitstellt

ANMERKUNG 1 zum Begriff: Diese von ISO/IEC 7498-1 abgeleitete Definition wird an dieser Stelle wiederholt, um den entscheidenden Unterschied zwischen DLSAPs und deren DL-Adressen besser verstehen zu können.

3.3.5

DL(SAP)-Adresse

(en: DL(SAP)-address)

entweder eine individuelle DLSAP-Adresse, die einen einzelnen DLSAP eines einzelnen DLS-Benutzers kennzeichnet, oder eine DL-Gruppenadresse, die möglicherweise mehrere DLSAPs jeweils eines einzelnen DLS-Benutzers kennzeichnet

ANMERKUNG 1 zum Begriff: Diese Terminologie wird gewählt, da ISO/IEC 7498-3 die Verwendung der Benennung „DLSAP-Adresse“ zur Bezeichnung von mehr als einem einzelnen DLSAP bei einem einzelnen DLS-Benutzer nicht zulässt.

3.3.6

(individuelle) DLSAP-Adresse

(en: (individual) DLSAP-address)

DL-Adresse, die nur einen einzigen DLSAP innerhalb des erweiterten Verbindungszweigs kennzeichnet

ANMERKUNG 1 zum Begriff: Eine einzelne DL-Entität kann mehrere DLSAP-Adressen, die einem einzelnen DLSAP zugeordnet sind, besitzen.

3.3.7

DLSEP-Adresse

(en: DLSEP-address)

DL-Adresse, die einen DL-Planungsendpunkt innerhalb einer DLE kennzeichnet

ANMERKUNG 1 zum Begriff: Hierbei handelt es sich um eine Erweiterung der Verwendung von DL-Adressen gegenüber der in ISO/IEC 7498-3 festgelegten Verwendung. (Siehe Bild 1.)

3.3.8

erweiterter Verbindungszweig

(en: extended link)

DL-Subnetzwerk, das aus der maximalen Menge von Verbindungszweigen besteht, die über DL-Relais-Entitäten geschaltet werden können, wobei ein einziger DL-Namensraum (DL-Adressraum) gemeinsam genutzt wird; in diesem Subnetzwerk können alle verbundenen DL-Entitäten miteinander entweder direkt oder über eine oder mehrere der zwischengeschalteten DL-Relais-Entitäten kommunizieren

ANMERKUNG 1 zum Begriff: Ein erweiterter Verbindungszweig besteht möglicherweise aus nur einem einzelnen Verbindungszweig.

3.3.9

Rahmen

(en: frame)

umgangssprachliches Synonym für DLPDU

3.3.10**DL-Gruppenadresse**

(en: group DL-address)

DL-Adresse, die möglicherweise mehr als einen DLSAP innerhalb des erweiterten Verbindungszweigs kennzeichnet

ANMERKUNG 1 zum Begriff: Eine einzelne DL-Entität kann mehrere DL-Gruppenadressen, die einem einzelnen DLSAP zugeordnet sind, besitzen. Eine einzelne DL-Entität kann auch eine einzelne DL-Gruppenadresse, die mehr als einem DLSAP zugeordnet ist, besitzen.

3.3.11**Initiator**

(en: initiator)

DLE-Rolle, in der eine DLE eine DLPDU an eine Peer-Beantworter-DLE sendet, welche im Rahmen derselben Transaktion unmittelbar eine Antwort-DLPDU an den Initiator (und möglicherweise an andere DLEs) zurücksendet

ANMERKUNG 1 zum Begriff: In einigen früheren nationalen Normen wurde diese Rolle als „Master“-Rolle bezeichnet.

3.3.12**Multi-Peer-DLC**

(en: multi-peer DLC)

zentralisierte DL-Verbindung mit mehreren Endpunkten, die eine DL-Duplex-Übertragung zwischen einem einzelnen definierten DLS-Benutzer, auch bekannt als Publisher oder veröffentlichender DLS-Benutzer, und einer Menge von nicht definierten Peer-DLS-Benutzern, die in ihrer Gesamtheit als Subscriber oder abonnierende DLS-Benutzer bekannt sind, anbietet, wobei der veröffentlichende DLS-Benutzer Daten an die abonnierenden DLS-Benutzer als Gruppe (jedoch nicht einzeln) senden kann und die abonnierenden DLS-Benutzer Daten an den veröffentlichenden DLS-Benutzer (jedoch sich nicht gegenseitig) senden können

ANMERKUNG 1 zum Begriff: Eine Multi-Peer-DLC stellt immer einen asymmetrischen Dienst bereit. Sie kann auch nur einen Simplex-Dienst entweder vom Publisher zu den Subscribern oder von den Subscribern zum Publisher bereitstellen. In diesem letzten Fall sind die Bezeichnungen als Publisher und Subscriber unzutreffend.

ANMERKUNG 2 zum Begriff: Der veröffentlichende DLS-Benutzer muss unter Umständen eine Steuerung seiner Veröffentlichungsrate implementieren, da ein abonnierender DLS-Benutzer weder eine Flusssteuerung noch eine Geschwindigkeitssteuerung auf seine veröffentlichende Peer-Entität anwenden kann. Ähnliche Aspekte treffen auf abonnierende DLS-Benutzer in Bezug auf deren Sende-DLSDUs an die veröffentlichenden DLS-Benutzer zu.

3.3.13**Knoten**

(en: node)

einzelne DL-Entität, wie sie auf einem lokalen Verbindungszweig existiert

3.3.14**Peer-DLC**

(en: peer DLC)

DL-Verbindung zwischen einzelnen Punkten, die eine DL-Duplex-Übertragung zwischen zwei Peer-DLS-Benutzern anbietet, wobei jeder von ihnen ein sendender DLS-Benutzer sein kann und jeder von ihnen als empfangender DLS-Benutzer eine Flusssteuerung auf seinen sendenden Peer anwenden kann

ANMERKUNG 1 zum Begriff: Eine Peer-DLC kann entweder einen symmetrischen Dienst oder einen asymmetrischen Dienst bereitstellen. Eine Peer-DLC kann auch nur einen DL-Simplex-Dienst anbieten.

3.3.15**empfangender DLS-Benutzer**

(en: receiving DLS-user)

Benutzer der DL-Dienste, der als Empfänger von DL-Benutzerdaten agiert

ANMERKUNG 1 zum Begriff: Ein Benutzer der DL-Dienste kann gleichzeitig sendender und empfangender DLS-Benutzer sein.

3.3.16

Beantworter

(en: responder)

DLE-Rolle, in der eine DLE im Rahmen einer einzigen Transaktion eine DLPDU als unmittelbare Antwort auf eine DLPDU sendet, die von einer Peer-Initiator-DLE empfangen wurde

ANMERKUNG 1 zum Begriff: In einigen früheren nationalen Normen wurde diese Rolle als „Slave“-Rolle bezeichnet.

3.3.17

sendender DLS-Benutzer

(en: sending DLS-user)

Benutzer der DL-Dienste, der als Quelle von DL-Benutzerdaten agiert

3.3.18

Aktualität, DL-Aktualität

(en: timeliness, DL-timeliness)

Attribut eines Datums zur Bewertung der zeitlichen Nähe dieses Datums; insbesondere von Bedeutung in Systemen mit Abtastwerten, bei denen Entscheidungen unter Umständen auf der Grundlage der Aktualität (oder fehlenden Aktualität) von Abtastwerten getroffen werden müssen

ANMERKUNG 1 zum Begriff: Die Aktualität ist im Allgemeinen ein Benutzerattribut, das durch die verschiedenen Schichten des Datentransportsystems negativ beeinflusst werden kann. Das heißt, ein Datum, das von einem anfordernden Benutzer rechtzeitig an ein Datenkommunikations-Subsystem zur Übertragung übergeben wurde, kann aufgrund von Verzögerungen im Kommunikationssystem verspätet übertragen werden.

ANMERKUNG 2 zum Begriff: Die DL-Aktualität ist ein Attribut für ein Datum eines DLS-Benutzers, das den Zeitpunkt einer Interaktion zwischen DLS-Benutzer und DLE festlegt und dieses Datum mit Bezug auf eine oder mehrere andere (frühere) Interaktionen zwischen DLS-Benutzer und DLE schreibt oder ausliest.

ANMERKUNG 3 zum Begriff: Diese Konzepte unterstützen auch die Migration von früheren nationalen Normen.

3.3.19

Transaktion

(en: transaction)

einzelne DLPDU oder eine Folge von zwei unmittelbar aufeinanderfolgenden zusammengehörigen DLPDUs, die das Ergebnis einer einzelnen Anforderung eines DLS-Benutzers ist

ANMERKUNG 1 zum Begriff: Die DLE, die die erste DLPDU der Transaktion sendet, wird als „Initiator“ bezeichnet; die DLE, die gegebenenfalls die zweite DLPDU der Transaktion sendet, wird als „Beantworter“ bezeichnet.

ANMERKUNG 2 zum Begriff: Eine DL-Entität kann innerhalb derselben Transaktion sowohl Initiator als auch Beantworter sein.

English Version

**Industrial communication networks - Fieldbus specifications -
Part 3-1: Data-link layer service definition - Type 1 elements
(IEC 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
(CEI 61158-3-1:2014)

Industrielle Kommunikationsnetze - Feldbusse - Teil 3-1:
Dienstfestlegungen des Data Link Layer
(Sicherheitsschicht) - Typ 1-Elemente
(IEC 61158-3-1:2014)

This European Standard was approved by CENELEC on 2014-09-17. CENELEC members are bound to comply with the CEN/CENELEC Internal Regulations which stipulate the conditions for giving this European Standard the status of a national standard without any alteration.

Up-to-date lists and bibliographical references concerning such national standards may be obtained on application to the CEN-CENELEC Management Centre or to any CENELEC member.

This European Standard exists in three official versions (English, French, German). A version in any other language made by translation under the responsibility of a CENELEC member into its own language and notified to the CEN-CENELEC Management Centre has the same status as the official versions.

CENELEC members are the national electrotechnical committees of Austria, Belgium, Bulgaria, Croatia, Cyprus, the Czech Republic, Denmark, Estonia, Finland, Former Yugoslav Republic of Macedonia, France, Germany, Greece, Hungary, Iceland, Ireland, Italy, Latvia, Lithuania, Luxembourg, Malta, the Netherlands, Norway, Poland, Portugal, Romania, Slovakia, Slovenia, Spain, Sweden, Switzerland, Turkey and the United Kingdom.



European Committee for Electrotechnical Standardization
Comité Européen de Normalisation Electrotechnique
Europäisches Komitee für Elektrotechnische Normung

CEN-CENELEC Management Centre: Avenue Marnix 17, B-1000 Brussels

Foreword

The text of document 65C/759/FDIS, future edition 2 of IEC 61158-3-1, prepared by SC 65C "Industrial networks" of IEC/TC 65 "Industrial-process measurement, control and automation" was submitted to the IEC-CENELEC parallel vote and approved by CENELEC as EN 61158-3-1:2014.

The following dates are fixed:

- latest date by which the document has to be implemented at national level by publication of an identical national standard or by endorsement (dop) 2015-06-17
- latest date by which the national standards conflicting with the document have to be withdrawn (dow) 2017-09-17

This document supersedes EN 61158-3-1:2008.

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. CENELEC [and/or CEN] shall not be held responsible for identifying any or all such patent rights.

This document has been prepared under a mandate given to CENELEC by the European Commission and the European Free Trade Association.

Endorsement notice

The text of the International Standard IEC 61158-3-1:2014 was approved by CENELEC as a European Standard without any modification.

In the official version, for Bibliography, the following notes have to be added for the standards indicated:

IEC 61158-1	NOTE	Harmonized as EN 61158-1.
IEC 61158-2	NOTE	Harmonized as EN 61158-2.
IEC 61158-4-1	NOTE	Harmonized as EN 61158-4-1.
IEC 61158-5-9	NOTE	Harmonized as EN 61158-5-9.
IEC 61158-6-9	NOTE	Harmonized as EN 61158-6-9.
IEC 61784-1	NOTE	Harmonized as EN 61784-1.
IEC 61784-2	NOTE	Harmonized as EN 61784-2.

CONTENTS

0	INTRODUCTION	8
0.1	General	8
0.2	Nomenclature for references within this standard	8
1	Scope	9
1.1	General	9
1.2	Specifications	9
1.3	Conformance	9
2	Normative references	10
3	Terms, definitions, symbols, abbreviations and conventions	10
3.1	Reference model terms and definitions.....	10
3.2	Service convention terms and definitions.....	12
3.3	Data-link service terms and definitions	12
3.4	Common symbols and abbreviations	16
3.5	Common conventions	16
4	Overview of the data-link layer service	18
4.1	General	18
4.2	Types and classes of data-link layer service.....	21
4.3	Quality-of-service (QoS) attributes common to multiple types of data-link layer service	21
5	DL(SAP)-address, queue and buffer management data-link layer service.....	27
5.1	Facilities of the DL(SAP)-address, queue and buffer management data-link layer service	27
5.2	Model of the DL(SAP)-address, queue and buffer management data-link layer service	27
5.3	Sequence of primitives at one DLSAP	27
5.4	DL(SAP)-address, queue and buffer management facilities.....	29
6	Connection-mode data-link layer service	43
6.1	Facilities of the connection-mode data-link layer service.....	43
6.2	Model of the connection-mode data-link layer service	44
6.3	Quality of connection-mode service.....	51
6.4	Sequence of primitives	57
6.5	Connection establishment phase.....	67
6.6	Connection release phase.....	74
6.7	Data transfer phase	80
7	Connectionless-mode data-link layer service	92
7.1	Facilities of the connectionless-mode data-link layer service.....	92
7.2	Model of the connectionless-mode data-link layer service	92
7.3	Quality of connectionless-mode service.....	94
7.4	Sequence of primitives	94
7.5	Connectionless-mode functions.....	97
8	Time and scheduling guidance data-link layer service.....	108
8.1	Facilities and classes of the time and scheduling guidance data-link layer service.....	108
8.2	Model of the time and scheduling guidance data-link layer service	109
8.3	Quality of scheduling guidance service.....	109
8.4	Sequence of primitives at one DLE	109
8.5	Scheduling guidance functions.....	110

9	DL-management service	121
9.1	Scope and inheritance	121
9.2	Facilities of the DL-management service	121
9.3	Model of the DL-management service	121
9.4	Constraints on sequence of primitives	121
9.5	Set	122
9.6	Get	123
9.7	Action	123
9.8	Event	124
	Bibliography	126
	Annex ZA (normative) Normative references to international publications with their corresponding European publications	127
	Figure 1 – Relationships of DLSAPs, DLSAP-addresses, DLCEPs, DLCEP-addresses, DLSEP-addresses and group DL-addresses	13
	Figure 2 – Example of paths, links, bridges, and the extended link	19
	Figure 3 – Types of DL-timeliness In terms of elapsed DL-time and events at the assessing DLCEP	25
	Figure 4 – Sequence of primitives for the DL(SAP)-address, queue and buffer management DLS	29
	Figure 5 – Supported methods of data management for transmission and delivery	30
	Figure 6 – Peer-to-peer and multi-peer DLCs and their DLCEPs	44
	Figure 7 – OSI abstract queue model of a peer DLC between a pair of DLS-users	45
	Figure 8 – OSI abstract queue model of a multi-peer DLC between a publishing DLS-user and a set of subscribing DLS-users	49
	Figure 9 – Summary of DL-connection-mode service primitive time-sequence diagrams for peer DLCs (portion 1)	60
	Figure 10 – Summary of DL-connection-mode service primitive time-sequence diagrams for peer DLCs (portion 2)	61
	Figure 11 – Summary of DL-connection-mode service primitive time-sequence diagrams for publishers of a multi-peer DLC (portion 1)	62
	Figure 12 – Summary of DL-connection-mode service primitive time-sequence diagrams for publishers of a multi-peer DLC (portion 2)	63
	Figure 13 – Summary of additional DL-connection-mode service primitive time-sequence diagrams for a multi-peer DLC subscriber where the diagrams differ from the corresponding ones for a publisher (portion 1)	64
	Figure 14 – Summary of additional DL-connection-mode service primitive time-sequence diagrams for a multi-peer DLC subscriber where the diagrams differ from the corresponding ones for a publisher (portion 2)	65
	Figure 15 – State transition diagram for sequences of DL-connection-mode service primitives at a DLCEP	66
	Figure 16 – Peer DLC/DLCEP establishment initiated by a single DLS-user	72
	Figure 17 – Multi-peer DLC/DLCEP establishment initiated by the publishing DLS-user	73
	Figure 18 – Multi-peer DLC/DLCEP establishment initiated by a subscribing DLS-user	73
	Figure 19 – Multi-peer DLC/DLCEP establishment using known DLCEP addresses initiated first by the publishing DLS-user	73
	Figure 20 – Multi-peer DLC/DLCEP establishment using known DLCEP addresses initiated first by one or more subscribing DLS-users	73

Figure 21 – Peer DLC/DLCEP establishment initiated simultaneously by both peer DLS-users, resulting in a merged DLC	74
Figure 22 – Multi-peer DLC/DLCEP establishment initiated simultaneously by both publishing and subscribing DLS-users, resulting in a merged DLC.....	74
Figure 23 – Peer DLS-user invocation	77
Figure 24 – Publishing DLS-user invocation	77
Figure 25 – Subscribing DLS-user invocation	77
Figure 26 – Simultaneous invocation by both DLS-users.....	77
Figure 27 – Peer DLS-provider invocation	77
Figure 28 – Publishing DLS-provider invocation.....	77
Figure 29 – Subscribing DLS-provider invocation.....	77
Figure 30 – Simultaneous peer DLS-user and DLS-provider invocations.....	77
Figure 31 – Simultaneous publishing DLS-user and DLS-provider invocations.....	78
Figure 32 – Simultaneous subscribing DLS-user and DLS-provider invocations	78
Figure 33 – Sequence of primitives in a peer DLS-user rejection of a DLC/DLCEP establishment attempt.....	78
Figure 34 – Sequence of primitives in a publishing DLS-user rejection of a DLC/DLCEP establishment attempt.....	78
Figure 35 – Sequence of primitives in a subscribing DLS-user rejection of a DLC/DLCEP establishment attempt.....	78
Figure 36 – Sequence of primitives in a DLS-provider rejection of a DLC/DLCEP establishment attempt.....	79
Figure 37 – Sequence of primitives in a DLS-user cancellation of a DLC/DLCEP establishment attempt: both primitives are destroyed in the queue	79
Figure 38 – Sequence of primitives in a DLS-user cancellation of a DLC/DLCEP establishment attempt: DL-DISCONNECT indication arrives before DL-CONNECT response is sent.....	79
Figure 39 – Sequence of primitives in a DLS-user cancellation of a DLC/DLCEP establishment attempt: peer DL-DISCONNECT indication arrives after DL-CONNECT response is sent.....	79
Figure 40 – Sequence of primitives in a DLS-user cancellation of a DLC/DLCEP establishment attempt: publisher's DL-DISCONNECT indication arrives after DL-CONNECT response is sent	80
Figure 41 – Sequence of primitives in a DLS-user cancellation of a DLC/DLCEP establishment attempt: subscriber's DL-DISCONNECT request arrives after DL-CONNECT request has been communicated to the publisher	80
Figure 42 – Sequence of primitives for a CLASSICAL or DISORDERED peer-to-peer queue-to-queue data transfer.....	82
Figure 43 – Sequence of primitives for an ORDERED or UNORDERED peer-to-peer, or an UNORDERED subscriber-to-publisher queue-to-queue data transfer	83
Figure 44 – Sequence of primitives for a publisher-to-subscribers queue-to-queue data transfer.....	83
Figure 45 – Sequence of primitives for a failed queue-to-queue data transfer.....	83
Figure 46 – Sequence of primitives for an ORDERED or UNORDERED peer to peer, or an UNORDERED subscriber to publisher, buffer to buffer data transfer	84
Figure 47 – Sequence of primitives for a publisher to subscribers buffer to buffer data transfer.....	85
Figure 48 – Sequence of primitives for an ORDERED or UNORDERED peer to peer, or an UNORDERED subscriber to publisher, buffer to queue data transfer.....	85

Figure 49 – Sequence of primitives for a publisher to subscribers buffer to queue data transfer	85
Figure 50 – Sequence of primitives in a peer DLS-user initiated Reset.....	88
Figure 51 – Sequence of primitives in a publishing DLS-user initiated Reset	89
Figure 52 – Sequence of primitives in a subscribing DLS-user initiated Reset	89
Figure 53 – Sequence of primitives in a simultaneous peer DLS-users initiated Reset	89
Figure 54 – Sequence of primitives in a simultaneous multi-peer DLS-users initiated Reset..	89
Figure 55 – Sequence of primitives in a peer DLS-provider initiated Reset	89
Figure 56 – Sequence of primitives in a publishing DLS-provider initiated Reset	89
Figure 57 – Sequence of primitives in a subscribing DLS-provider initiated Reset	90
Figure 58 – Sequence of primitives in a simultaneous peer DLS-user and DLS-provider initiated Reset	90
Figure 59 – Sequence of primitives in a simultaneous publishing DLS-user and DLS-provider initiated Reset	90
Figure 60 – Sequence of primitives in a simultaneous subscribing DLS-user and DLS-provider initiated Reset	90
Figure 61 – Sequence of primitives for Subscriber Query.....	91
Figure 62 – Model for a data-link layer connectionless-mode unitdata transmission or unitdata exchange	93
Figure 63 – Summary of DL-connectionless-mode service primitive time-sequence diagrams	96
Figure 64 – State transition diagram for sequences of connectionless-mode primitives at one DLSAP	97
Figure 65 – Sequence of primitives for a successful locally-acknowledged connectionless-mode unitdata transfer.....	100
Figure 66 – Sequence of primitives for a successful remotely-acknowledged connectionless-mode unitdata transfer.....	101
Figure 67 – Sequence of primitives for an unsuccessful connectionless-mode unitdata transfer	101
Figure 68 – Sequence of primitives for connectionless-mode unitdata exchange	106
Figure 69 – Sequence of primitives for connectionless-mode listener query	107
Figure 70 – Summary of time and scheduling-guidance service primitive time sequence diagrams	110
Figure 71 – Sequence of primitives for DL-time	112
Figure 72 – Sequence of primitives for the Compel-Service service	114
Figure 73 – Sequence of primitives for the sequence scheduling services	118
Figure 74 – Sequence of primitives for the DLM action service	121
Table 1 – Summary of DL(SAP)-address, queue and buffer management primitives and parameters	28
Table 2 – DL-buffer-and-queue-management create primitive and parameters	30
Table 3 – DL-buffer-and-queue-management delete primitive and parameters	33
Table 4 – DL(SAP)-address-management bind primitive and parameters	34
Table 5 – DL(SAP)-role constraints on DLSAPs, DLCEPs and other DLS Primitives	35
Table 6 – DL(SAP)-address-management unbind primitive and parameters.....	38
Table 7 – DL-buffer-management put primitive and parameters	39

Table 8 – DL-buffer-and-queue-management get primitive and parameters	41
Table 9 – Relationships between abstract queue model objects	47
Table 10 – Attributes and class requirements of DLCEP data delivery features	53
Table 11 – Summary of DL-connection-mode primitives and parameters (portion 1)	58
Table 12 – Summary of DL-connection-mode primitives and parameters (portion 2)	59
Table 13 – DLC / DLCEP establishment primitives and parameters (portion 1)	68
Table 14 – DLC / DLCEP establishment primitives and parameters (portion 2)	69
Table 15 – DLC / DLCEP release primitives and parameters	75
Table 16 – Queue data transfer primitive and parameters	80
Table 17 – Buffer sent primitive and parameter	83
Table 18 – Buffer received primitive and parameter	84
Table 19 – DLC/DLCEP reset primitives and parameters (portion 1)	86
Table 20 – DLC/DLCEP reset primitives and parameters (portion 2)	86
Table 21 – Subscriber query primitives and parameters	91
Table 22 – Summary of DL-connectionless-mode primitives and parameters	95
Table 23 – DL-connectionless-mode unitdata transfer primitives and parameters	98
Table 24 – DL-connectionless-mode unitdata exchange primitive and parameters	102
Table 25 – Listener query primitives and parameters	107
Table 26 – Summary of DL-scheduling-guidance primitives and parameters	109
Table 27 – DL-time primitive and parameters	111
Table 28 – DL-scheduling-guidance Compel-service primitive and parameters	112
Table 29 – DL-scheduling-guidance Schedule Sequence primitives and parameters	115
Table 30 – DL-scheduling-guidance Cancel Schedule primitives and parameters	119
Table 31 – DL-scheduling-guidance Subset Sequence primitives and parameters	120
Table 32 – Summary of DL-management primitives and parameters	122
Table 33 – DLM-Set primitive and parameters	122
Table 34 – DLM-Get primitive and parameters	123
Table 35 – DLM-Action primitive and parameters	124
Table 36 – DLM-Event primitive and parameters	125

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.

Throughout the set of fieldbus standards, the term “service” refers to the abstract capability provided by one layer of the OSI Basic Reference Model to the layer immediately above. Thus, the data-link layer service defined in this standard is a conceptual architectural service, independent of administrative and implementation divisions.

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 61158-3, 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 3-1: Data-link layer service definition – Type 1 elements

1 Scope

1.1 General

This part of IEC 61158 provides common elements for basic time-critical messaging communications between devices in an automation environment. The term “time-critical” is used to represent the presence of a time-window, within which one or more specified actions are required to be completed with some defined level of certainty. Failure to complete specified actions within the time window risks failure of the applications requesting the actions, with attendant risk to equipment, plant and possibly human life.

This standard defines in an abstract way the externally visible service provided by the Type 1 fieldbus data-link layer in terms of

- a) the primitive actions and events of the service;
- b) the parameters associated with each primitive action and event, and the form which they take; and
- c) the interrelationship between these actions and events, and their valid sequences.

The purpose of this standard is to define the services provided to

- the Type 1 fieldbus application layer at the boundary between the application and data-link layers of the fieldbus reference model;
- systems management at the boundary between the data-link layer and systems management of the fieldbus reference model.

1.2 Specifications

The principal objective of this standard is to specify the characteristics of conceptual data-link layer services suitable for time-critical communications, and thus supplement the OSI Basic Reference Model in guiding the development of data-link protocols for time-critical communications. A secondary objective is to provide migration paths from previously existing industrial communications protocols.

This specification may be used as the basis for formal DL-Programming-Interfaces. Nevertheless, it is not a formal programming interface, and any such interface will need to address implementation issues not covered by this specification, including

- a) the sizes and octet ordering of various multi-octet service parameters;
- b) the correlation of paired request and confirm, or indication and response, primitives.

1.3 Conformance

This standard does not specify individual implementations or products, nor does it constrain the implementations of data-link entities within industrial automation systems.

There is no conformance of equipment to this data-link layer service definition standard. Instead, conformance is achieved through implementation of the corresponding data-link protocol that fulfills the Type 7 data-link layer services defined in this standard.

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.

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 10731:1994, *Information technology – Open Systems Interconnection – Conventions for the definition of OSI services*

3 Terms, definitions, symbols, abbreviations and conventions

For the purposes of this document, the following terms, definitions, symbols, abbreviations and conventions 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	DL-address	[7498-3]
3.1.2	DL-address-mapping	[7498-1]
3.1.3	called-DL-address	[7498-3]
3.1.4	calling-DL-address	[7498-3]
3.1.5	centralized multi-end-point-connection	[7498-1]
3.1.6	DL-connection	[7498-1]
3.1.7	DL-connection-end-point	[7498-1]
3.1.8	DL-connection-end-point-identifier	[7498-1]
3.1.9	DL-connection-mode transmission	[7498-1]
3.1.10	DL-connectionless-mode transmission	[7498-1]
3.1.11	correspondent (N)-entities correspondent DL-entities (N=2) correspondent Ph-entities (N=1)	[7498-1]
3.1.12	DL-duplex-transmission	[7498-1]

3.1.13	(N)-entity DL-entity (N=2) Ph-entity (N=1)	[7498-1]
3.1.14	DL-facility	[7498-1]
3.1.15	flow control	[7498-1]
3.1.16	(N)-layer DL-layer (N=2) Ph-layer (N=1)	[7498-1]
3.1.17	layer-management	[7498-1]
3.1.18	DL-local-view	[7498-3]
3.1.19	DL-name	[7498-3]
3.1.20	naming-(addressing)-domain	[7498-3]
3.1.21	peer-entities	[7498-1]
3.1.22	primitive name	[7498-3]
3.1.23	DL-protocol	[7498-1]
3.1.24	DL-protocol-connection-identifier	[7498-1]
3.1.25	DL-protocol-data-unit	[7498-1]
3.1.26	DL-relay	[7498-1]
3.1.27	reset	[7498-1]
3.1.28	responding-DL-address	[7498-3]
3.1.29	routing	[7498-1]
3.1.30	segmenting	[7498-1]
3.1.31	(N)-service DL-service (N=2) Ph-service (N=1)	[7498-1]
3.1.32	(N)-service-access-point DL-service-access-point (N=2) Ph-service-access-point (N=1)	[7498-1]
3.1.33	DL-service-access-point-address	[7498-3]
3.1.34	DL-service-connection-identifier	[7498-1]
3.1.35	DL-service-data-unit	[7498-1]
3.1.36	DL-simplex-transmission	[7498-1]
3.1.37	DL-subsystem	[7498-1]
3.1.38	systems-management	[7498-1]
3.1.39	DLS-user-data	[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 DLS-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 Data-link service terms and definitions

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

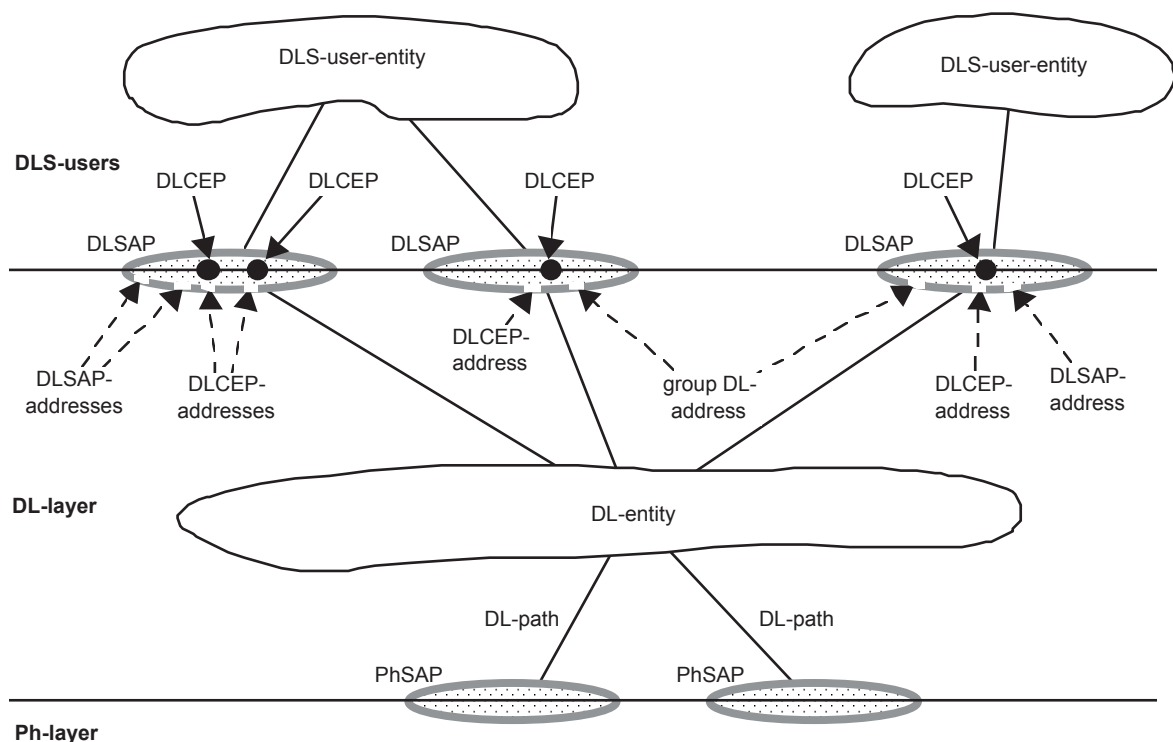
DLCEP-address

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).



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 can have multiple DLSAP-addresses and group DL-addresses associated with a single DLSAP.

NOTE 4 Figure 1 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

3.3.3

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

3.3.4

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.

3.3.5

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.

3.3.6

(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.

3.3.7

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. (See Figure 1.)

3.3.8

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 can be composed of just a single link.

3.3.9

frame

denigrated synonym for DLPDU

3.3.10

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.

3.3.11

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.

3.3.12

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.

3.3.13

node

single DL-entity as it appears on one local link

3.3.14

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.

3.3.15

receiving DLS-user

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

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

3.3.16

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.

3.3.17

sending DLS-user

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

3.3.18

timeliness

DL-timeliness

attribute of a datum which provides an assessment of the temporal currency of that datum, 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 1 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 2 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 (earlier) DLS-user/DLE interactions.

Note 3 to entry: These concepts also support migration from previous national standards.

3.3.19

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 Common symbols and abbreviations

NOTE Many symbols and abbreviations are common to more than one protocol Type; they are not necessarily used by all protocol Types.

3.4.1	DL-	Data-link layer (as a prefix)
3.4.2	DLC	DL-connection
3.4.3	DLCEP	DL-connection-end-point
3.4.4	DLE	DL-entity (the local active instance of the data-link layer)
3.4.5	DLL	DL-layer
3.4.6	DLPCI	DL-protocol-control-information
3.4.7	DLPDU	DL-protocol-data-unit
3.4.8	DLM	DL-management
3.4.9	DLME	DL-management Entity (the local active instance of DL-management)
3.4.10	DLMS	DL-management service
3.4.11	DLS	DL-service
3.4.12	DLSAP	DL-service-access-point
3.4.13	DLSEP	DL-schedule-end-point
3.4.14	DLSDU	DL-service-data-unit
3.4.15	FIFO	First-in first-out (queuing method)
3.4.16	OSI	Open systems interconnection
3.4.17	Ph-	Physical layer (as a prefix)
3.4.18	PhE	Ph-entity (the local active instance of the Physical layer)
3.4.19	PhL	Ph-layer
3.4.20	QoS	Quality-of-service

3.5 Common conventions

3.5.1 Basic conventions

This standard uses the descriptive conventions given in ISO/IEC 10731.

The service model, service primitives, and time-sequence diagrams used are entirely abstract descriptions; they do not represent a specification for implementation.

Service primitives, used to represent service user/service provider interactions (see ISO/IEC 10731), convey parameters that indicate information available in the user/provider interaction.

This standard uses a tabular format to describe the component parameters of the DLS primitives. The parameters that apply to each group of DLS primitives are set out in tables throughout the remainder of this standard. Each table consists of up to six columns, containing the name of the service parameter, and a column each for those primitives and parameter-transfer directions used by the DLS:

- the request primitive's input parameters;
- the request primitive's output parameters;
- the indication primitive's output parameters;
- the response primitive's input parameters; and
- the confirm primitive's output parameters.

NOTE The request, indication, response and confirm primitives are also known as requestor.submit, acceptor.deliver, acceptor.submit, and requestor.deliver primitives, respectively (see ISO/IEC 10731).

One parameter (or part of it) is listed in each row of each table. Under the appropriate service primitive columns, a code is used to specify the type of usage of the parameter on the primitive and parameter direction specified in the column:

- M — parameter is mandatory for the primitive;
- U — parameter is a user option and may or may not be provided depending on the dynamic usage of the DLS-user. When not provided, a default value for the parameter is assumed;
- C — parameter is conditional upon other parameters or upon the environment of the DLS-user;
- CU — parameter is a conditional user option, and may or may not be permitted depending upon other parameters or upon the environment of the DLS-user. When permitted, it may or may not be provided depending on the dynamic usage of the DLS-user. When permitted and not provided, a default value for the parameter is assumed.
- (blank) — parameter is never present.

Some entries are further qualified by items in brackets. These may be

a) a parameter-specific constraint

- (=) indicates that the parameter is semantically equivalent to the parameter in the service primitive to its immediate left in the table;
- (≤) indicates that the set of parameter values has an implicit order and that the parameter's value is less than or equal to that of the parameter in the service primitive to its immediate left in the table (that is, left by one or two columns);
- (≥) indicates that the set of parameter values has an implicit order and that the parameter's value is greater than, or equal to, that of the parameter in the service primitive to its immediate left in the table (that is, left by one or two columns).

b) an indication that some note applies to the entry

- (n) indicates that the following note n contains additional information pertaining to the parameter and its use.

In any particular interface, not all parameters need be explicitly stated. Some may be implicitly associated with the DLSAP at which the primitive is issued.

In the diagrams which illustrate these interfaces, dashed lines indicate cause-and-effect or time-sequence relationships, and wavy lines indicate that events are roughly contemporaneous.

3.5.2 Identifiers

Many of the DLS primitives specify one or more identifier parameters that are drawn from either a local DL-identifier space or a local DLS-user-identifier space. The existence and use of such identifiers in an implementation of the services specified in this standard is a purely local issue. Nevertheless, these identifiers are specified explicitly in these primitives to provide a descriptive means

- a) of cancelling (aborting) an outstanding request primitive before receiving its corresponding confirm primitive;
- b) for referring within a request or response primitive to persistent DL-objects, such as buffers and queues, which were created as the result of a previous DLS primitive; and
- c) for referring within an indication or confirm primitive to persistent DL-objects which were created as the result of a previous DLS primitive.

Adherence to the OSI principle of architectural layering necessitates the presumption of distinct non-intersecting identifier spaces for the DLS-provider and each separate DLS-user, because they may have non-overlapping local views. Consequently, DLS-user identifiers are required for a) and b); while DL-identifiers are required for c).

4 Overview of the data-link layer service

4.1 General

The DLS provides for the transparent and reliable transfer of data between DLS-users. It makes the way that supporting communications resources are utilized invisible to these DLS-users.

In particular, the DLS provides for the following:

- a) Independence from the underlying physical layer. The DLS relieves DLS-users from all direct concerns regarding which configuration is available (for example, direct connection, or indirect through one or more bridges) and which physical facilities are used (for example, which of a set of diverse physical paths).
- b) Transparency of transferred information. The DLS provides for the transparent transfer of DLS-user-data. It does not restrict the content, format or coding of the information, nor does it ever need to interpret the structure or meaning of that information. It may, however, restrict the amount of information that can be transferred as an indivisible unit.

NOTE 1 A DLS-user may segment arbitrary-length data into limited-length DLSDUs before making DLS requests, and may reassemble received DLSDUs into those larger data units.

- c) Reliable transfer of data. The DLS relieves the DLS-user from concerns regarding insertion or corruption of data, or, if requested, loss, duplication or misordering of data, which may occur. In some cases of unrecovered errors in the data-link Layer, duplication or loss of DLSDUs may occur. In cases where protection against misordering of data is not employed, misordering can occur.

NOTE 2 Detection of duplicate, lost or misordered DLSDUs may be performed by DLS-users.

- d) Quality-of-service (QoS) selection. The DLS provides DLS-users with a means to request and to agree upon a quality of service for the transfer of data. QoS is specified by means of QoS parameters representing aspects such as mode of operation, transit delay, accuracy and reliability.
- e) Addressing. The DLS allows the DLS-user to identify itself and to specify the DLSAPs between which a DLC is to be established. DL-addresses have only regional significance within a specific DL-segment. Extended DL-addresses have only regional significance within a specific DL-subsystem over a set of bridged DL-segments. Therefore, it is not appropriate to define a global addressing structure.

NOTE 3 The DLS is required to differentiate between the individual systems that are physically or logically connected to a multipoint data-link and to differentiate between connections. For commonality with other service definitions, this

mechanism is referred to as addressing and the objects used to differentiate between systems are referred to as addresses. In a formal sense, this is an extension of the use of addresses beyond that specified in ISO/IEC 7498-3.

- f) Scheduling. The DLS allows the set of DLS-users to provide some guidance on the internal scheduling of the distributed DLS-provider. This guidance supports the time-critical aspects of the DLS, by permitting the DLS-user some degree of management over when opportunities for communication will be granted to various DLEs for various DLSAP-addresses and DLCEPs.
- g) Common time sense. The DLS can provide the DLS-user with a sense of time that is common to all DLS-users on the extended link.
- h) Queues and buffers. The DLS can provide the sending or receiving DLS-user with either a FIFO queue or a retentive buffer or a non-retentive buffer (that is, one which becomes empty after being read), where each queue item or buffer can hold a single DLSDU.

4.1.1 Overview of DL-subnetwork structuring

A DL-subnetwork consists of a set of DL-segments (links) interconnected by DL-relay entities (bridges) which provide DL-layer-internal synchronization, coordination and routing services to the set of interconnected DLEs.

A DL-segment consists of a set of DLEs, all of which are connected directly (that is, without intervening DL-relay entities) to a single shared logical communications channel, known as a *link*.

A link (logical communications channel) consists of one or more physically independent, logically parallel, cooperatively scheduled, real communications channels, which are known as *paths*.

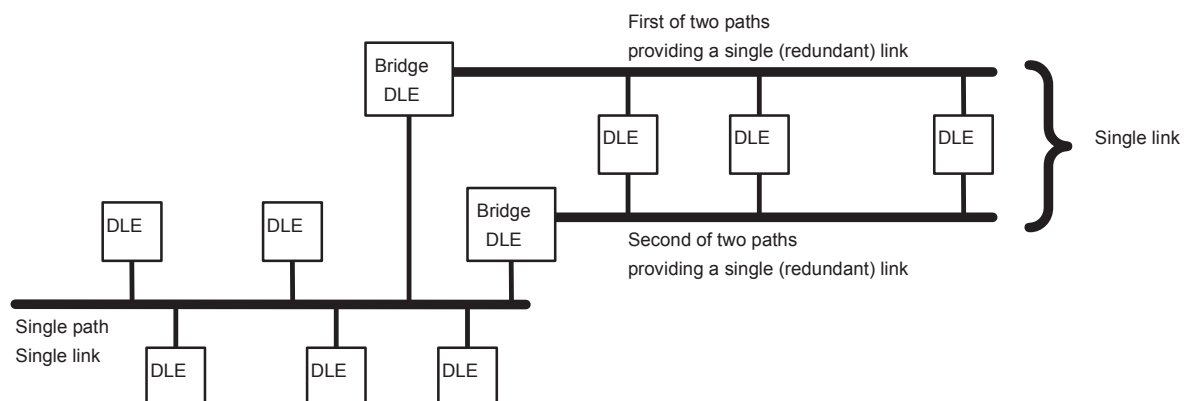


Figure 2 – Example of paths, links, bridges, and the extended link

A single shared PhL-provider enables communications among the DLEs on a given path. A link is made up of conceptually parallel paths. An example is shown in Figure 2.

NOTE 1 A link consisting of more than one path is an instance of DL-redundancy. This is distinct from Ph-redundancy, which is necessarily hidden from the DLL and all DLEs due to the principles of layering (ISO/IEC 7498-1).

NOTE 2 In a logical sense, DLEs are connected to links and bridges interconnect links. Yet in a physical sense, DLEs are connected to paths and bridges interconnect paths. DL-communication-services are independent of the specific path employed, and the DLS-user has no cognizance of any path multiplicity.

4.1.2 Overview of DL-naming (addressing)

DL-names, known conventionally as DL-addresses, are identifiers from a defined identifier space — the DL-address-space — which serve to name objects within the scope of the data-link layer. Examples of such objects are data-link layer-service-access-points (DLSAPs), data-link-connection-end-points (DLCEPs), and data-link-entities (DLEs).

The DL-address-space from which DL-addresses are drawn may be partitioned into subspaces of DL-addresses:

- a) to cluster addresses of the same generic function, such as
 - 1) DLSAP-addresses naming specific DLSAPs;
 - 2) DL-addresses naming groups of DLSAPs;
 - 3) DL-addresses designating one or more DLCEPs; and
 - 4) DL-addresses designating DLEs;
- b) to cluster addresses for administrative purposes, such as addresses that are
 - 1) known to be local to, and allocable by, a single DLE;
 - 2) known to be local to a single link (DL-segment) but not to have a known specific DLE-locality; or
 - 3) known to have no implicit DL-locality;
- c) to cluster addresses for routing purposes, such as addresses that are known to be local to a single DL-segment or a single DLE.

4.1.2.1 Functional partitioning of the DL-address-space

The space of DL-addresses may be partitioned functionally as follows:

- a) DLE-specific DLSAP-addresses;
- b) other DLSAP-addresses;
NOTE These addresses can designate at most one DLSAP within a single DLE within the entire set of DL-interconnected DLEs.
- c) group DL-addresses designating a group of DLSAPs;
NOTE These addresses are sometimes (incorrectly) referred to as group-DLSAP-addresses.
- d) DLE-specific DLCEP-addresses;
- e) other DLCEP-addresses;
- f) specific aspects of a specific DLE; or
- g) specific aspects of a group of DLEs.

4.1.2.2 Administrative partitioning of the DL-address-space

The space of DL-addresses may be partitioned administratively as follows:

- a) DLE-specific DL-addresses, which are known to refer to objects within the scope of a specific DLE, and which are allocable by that DLE;
- b) DL-segment (link) specific but DLE-independent DL-addresses, which are known to refer to objects within the scope of a specific DL-segment, and which are allocable locally by the DL-address-space administrator for that DL-segment; or
- c) DL-segment-independent DL-addresses, which are known to refer to objects within the DL-connected set of DL-segments, and which are allocable by the DL-address-space administrator for the connected set of DL-segments.

NOTE A DL-address-space administrator can always allocate a set of addresses to a subordinate administrator for its sub-administration. For example, the DL-administrator for the entire set of DL-connected DL-segments can allocate a contiguous block of unassigned DL-addresses to the DL-administrator for a specific DL-segment, or to the local administrator within a single DLE.

4.1.2.3 Routing-related partitioning of the DL-address-space

The space of DL-addresses may be partitioned to assist DL-routing activities as follows:

- a) DLE-specific DL-addresses;
- b) DL-segment (link) specific but DLE-independent DL-addresses; or
- c) DL-segment-independent DL-addresses.

4.2 Types and classes of data-link layer service

There are four types of DLS:

- a) a DL(SAP)-address, queue and buffer management service (defined in Clause 5);
- b) a connection-mode data transfer service with four classes of service (defined in Clause 6);
- c) a connectionless-mode data transfer service with three classes of service (defined in Clause 7);
and
- d) a time and transaction scheduling service (defined in Clause 8).

All four types of DLS are always provided; the DLS-user may choose those most appropriate for use. Within the DLS types, the DLS-user is limited to those classes of service supported by the selected DL-protocol implementation.

NOTE Classes of service are defined in detail in the clause which describes a specific type of DLS.

A DL-management service (DLMS) is defined in Clause 9.

4.3 Quality-of-service (QoS) attributes common to multiple types of data-link layer service

A DLS-user may select, directly or indirectly, many aspects of the various data-link services. The term quality-of-service (QoS) refers to those aspects that are under the direct control of the DLS-provider. QoS can only be properly determined when DLS-user behavior does not constrain or impede the performance of the DLS.

Static QoS attributes are selected once for an entire type of DLS. *Dynamic* QoS attributes are selected independently at each DLS invocation. *Semi-static* QoS attributes are static attributes for one type of DLS, and serve as defaults for corresponding dynamic attributes in another type of DLS.

Most QoS attributes have default values which can be set by DL-management and then overridden on a per-DLSAP-address basis by the DLS-user.

NOTE The existence of multiple levels of default QoS attribute values and of means of setting those default values can simplify use of the DLS. Some implementations of this DLS may provide additional levels of default QoS beyond those specified in this standard.

Four QoS attributes of the DL-data transfer services apply conceptually to both connection-mode and connectionless operation. The DLS-user may specify values for these attributes when binding a DLSAP-address to the DLS-user's DLSAP; any unspecified attributes will assume the default values set by DL-management:

- a) Two of these four attributes are considered dynamic; their DLSAP-address-related values serve merely as defaults for each appropriate DLS invocation and can be overridden on an instance by instance basis.
- b) The third and fourth attributes are semi-static; they are static for connectionless DLS, but dynamic for all DLCEP-establishment requests and responses, where its value serves merely as a default for each appropriate DLS invocation and can be overridden on an instance by instance basis.

A fifth attribute applies only to connection-mode operation and is dynamic for each DLCEP.

4.3.1 DLL priority (dynamic QoS attribute)

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. This maximum is determined by the DL-protocol specification.

The DL-protocol should support three DLL priority levels, each of which should be capable of conveying a specified amount of DLS-user data per appropriate 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* — capability of conveying up to 64 DLS-user octets per DLPDU;
- b) *NORMAL* — capability of conveying up to 128 DLS-user octets per DLPDU; and
- c) *TIME-AVAILABLE* — capability of conveying up to 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 DLCEP establishment may negotiate *URGENT* to *NORMAL* or *TIME-AVAILABLE*, or *NORMAL* to *TIME-AVAILABLE*.

The default QoS value can be set by DL-management; when not so set its value is *TIME-AVAILABLE*.

4.3.2 DLL maximum confirm delay (dynamic QoS attribute)

Each DLCEP establishment request, and each response, specifies upper bounds on the maximum time duration permitted for the completion of each related instance of a sequence of connection-oriented DLS primitives:

- a) the common maximum time for completion of
 - a related sequence of DL-CONNECT primitives;
 - a related sequence of DL-RESET primitives;
 - a related sequence of DL-SUBSCRIBER-QUERY primitives; and
- b) the maximum time for completion of a related sequence of DL-DATA primitives.

Each connectionless service request specifies an upper bound on the maximum time duration permitted for the completion of each related instance of a sequence of connectionless DLS primitives:

- c) the maximum time for completion of a related sequence of locally-confirmed DL-UNITDATA primitives; and
- d) the common maximum time for completion of
 - a related sequence of remotely-confirmed DL-UNITDATA primitives;
 - a related sequence of DL-LISTENER-QUERY primitives; or
 - an instance of the DL-UNITDATA-EXCHANGE service.

Each parameter either has the value *UNLIMITED* or specifies an upper bound, 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

- 1) between the issuing of the specified request primitives and the issuing of the corresponding confirm primitives; and
- 2) 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.

The default QoS values can be set by DL-management; when not so set the value for each of these QoS parameters is UNLIMITED.

4.3.3 DLPDU authentication (semi-static QoS attribute)

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 1 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; and

NOTE 2 This continuing background transmission is known as *residual activity*.

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

NOTE 3 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 should include the minimum permitted amount of addressing information;
- 2) SOURCE — each DLPDU should include a source DL-address where possible;
- 3) MAXIMAL — each DL-address should 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.

The default QoS value can be set by DL-management; when not so set its value is ORDINARY. DLCEP establishment may negotiate ORDINARY to SOURCE to MAXIMAL.

4.3.4 DL-scheduling-policy (semi-static QoS attribute)

This attribute is static for connectionless services, but is dynamic for connection-mode services.

For each DLSAP-address, and each DLCEP, the DLS-user can override the normal (implicitly-scheduled) DLL policy of providing the requested DLS 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 cancelled by an involved DLS-user. A DL-COMPEL-SERVICE request, specifying the affected DLSAP-address or DLCEP, permits the continued execution of just a single deferred in-process request or response DLS-primitive. Only DL-services that provide DLS-user intercommunication are affected by this attribute.

NOTE 1 DLC support services such as DL-CONNECT, DL-RESET and DL-DISCONNECT, and intra-DLS-provider services such as DL-SUBSCRIBER-QUERY and DL-LISTENER-QUERY, are not 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;

NOTE 2 The choice IMPLICIT is incompatible with a DLCEP which is bound as sender to a buffer, because writing to a buffer does not trigger transmission. Thus the only usable choice for a sending buffer is EXPLICIT.

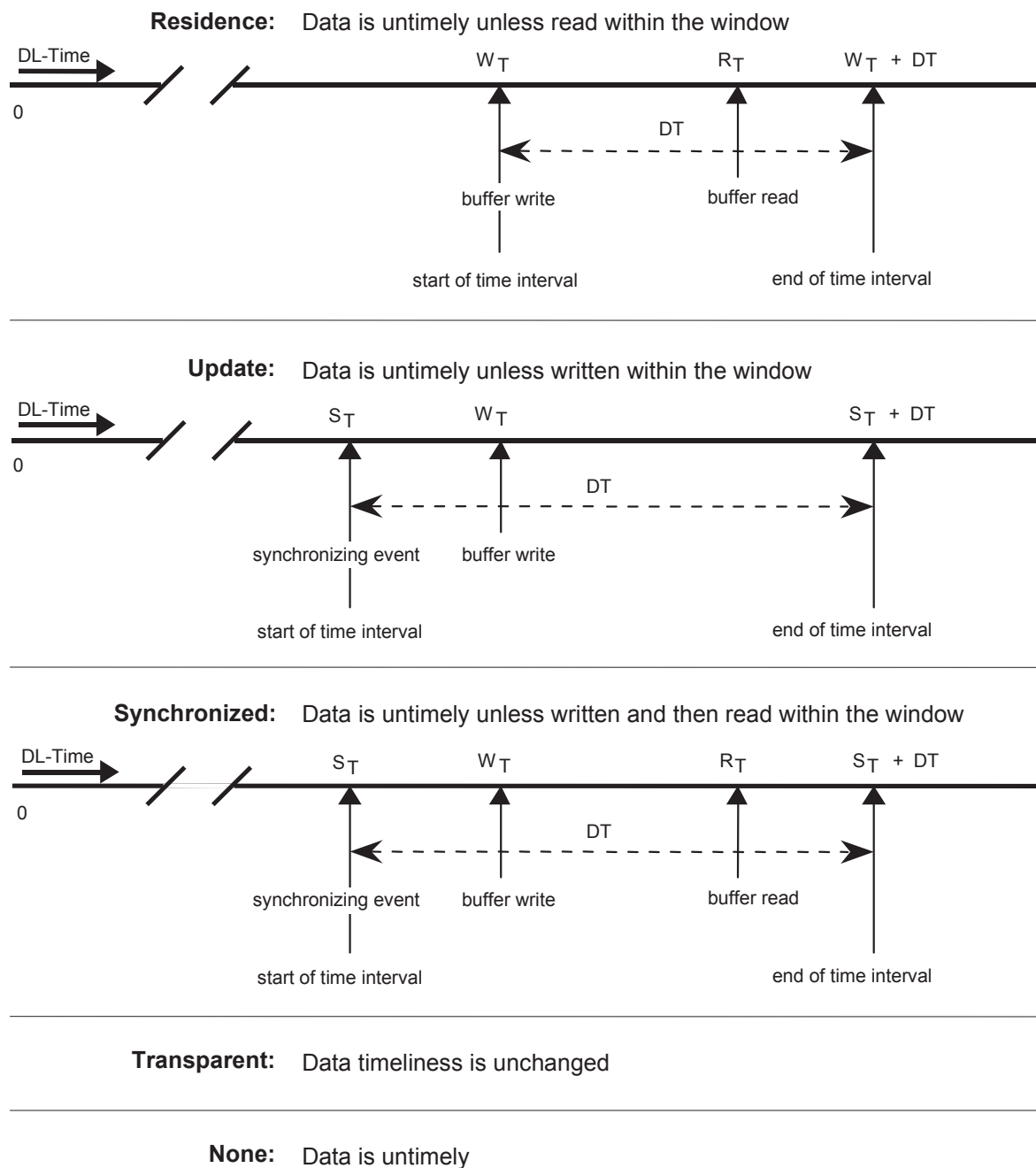
- b) EXPLICIT — any required data or unitdata communications with peer DLS-user(s) from this DLSAP-address, or from this DLCEP, will occur only when the deferral is explicitly cancelled by an involved DLS-user.

NOTE 3 Possible use of previously scheduled communications opportunities makes it possible for this deferral and subsequent release to result in earlier communications with the peer DLS-users than that provided by the IMPLICIT alternative.

The default QoS value cannot be set by DL-management; its value is always IMPLICIT.

4.3.5 DL-timeliness (dynamic DLCEP QoS attributes)

This attribute applies only to retentive 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.



Legend

- ← DL-event: DLS-user/DLE interaction, DL-buffer read, DL-buffer write, DL-timeout
- ← - - - - - → window size interval DT

Figure 3 – Types of DL-timeliness
In terms of elapsed DL-time and events at the assessing DLCEP

Each DLCEP establishment request, and each response, can specify DL-timeliness criteria which are to apply to information sent from, or received into, retentive 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 Figure 3:

- 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); and
 - 2) the moment when the buffer is read (by a DL-GET request primitive, or by transmission from the buffer at a DLCEP);

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

NOTE 1 This type of timeliness was called *asynchronous* in prior national standards.

- 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); and
 - 2) the moment when the buffer is written (by a DL-PUT request primitive, or by reception into the buffer at a DLCEP);

$$\text{DL-timeliness} \equiv 0 \leq (W_T - S_T) < \Delta T \quad (2)$$

NOTE 2 A type of timeliness closely related to this one was called *punctual* in prior national standards.

- 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); and
 - 3) the moment when the buffer is read (by a DL-GET request primitive, or by transmission from the buffer at a DLCEP);

$$\text{DL-timeliness} \equiv 0 \leq (W_T - S_T) \leq (R_T - S_T) < \Delta T \quad (3)$$

NOTE 3 This type of timeliness was called *synchronous* in prior national standards.

- 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 is 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 is FALSE.

The DL-time when the original buffer is written by a DL-PUT request primitive also can be conveyed to DLS-users which read a copy of the buffer. This DL-time is not available when the buffer timeliness is FALSE.

NOTE 4 DL-time is described in Clause 8.

5 DL(SAP)-address, queue and buffer management data-link layer service

5.1 Facilities of the DL(SAP)-address, queue and buffer management data-link layer service

The DLS provides the following facilities to the DLS-user:

- a) a means for creating and deleting a retentive buffer, or a non-retentive buffer, or a FIFO queue of specified depth, for use
 - 1) in communicating DLS-user-data between a DLS-user and the DLS-provider;
 - 2) in redistributing received DLS-user data without continuing DLS-user intervention; and
 - 3) in facilitating DLS-user supervision of the timing of DLSDU-conveyance to peer DLS-users;
- b) a means for associating an individual DLSAP-address or group DL-address, referred to collectively as a DL(SAP)-address, with, and disassociating a DL(SAP)-address from, the DLSAP at which the request is made.

Default values for some quality-of-service (QoS) attributes for connection-mode and connectionless data transfer services using the specified DL(SAP)-address can be specified when the association is made.

Additionally, the DLS-user may specify that previously created buffers or FIFO queues be used for each potential direction and priority of connectionless data transfer at the specified DL(SAP)-address.

- c) A means by which DLSDUs of limited size are written to or read from a buffer, or read from a FIFO queue.

5.2 Model of the DL(SAP)-address, queue and buffer management data-link layer service

This standard uses the abstract model for a layer service defined in ISO/IEC 10731:1994, Clause 5. The model defines interactions between the DLS-user and the DLS-provider that take place at a DLSAP. Information is passed between the DLS-user and the DLS-provider by DLS primitives that convey parameters.

The DL(SAP)-address, queue and buffer management primitives are used to provide a local service between a DLS-user and the local DLE. Remote DLEs and remote DLS-users are not directly involved, so there is no need for the other primitives of ISO/IEC 10731. Therefore the DL(SAP)-address, queue and buffer management services are provided by request (requestor.submit) primitives with input and output parameters.

5.3 Sequence of primitives at one DLSAP

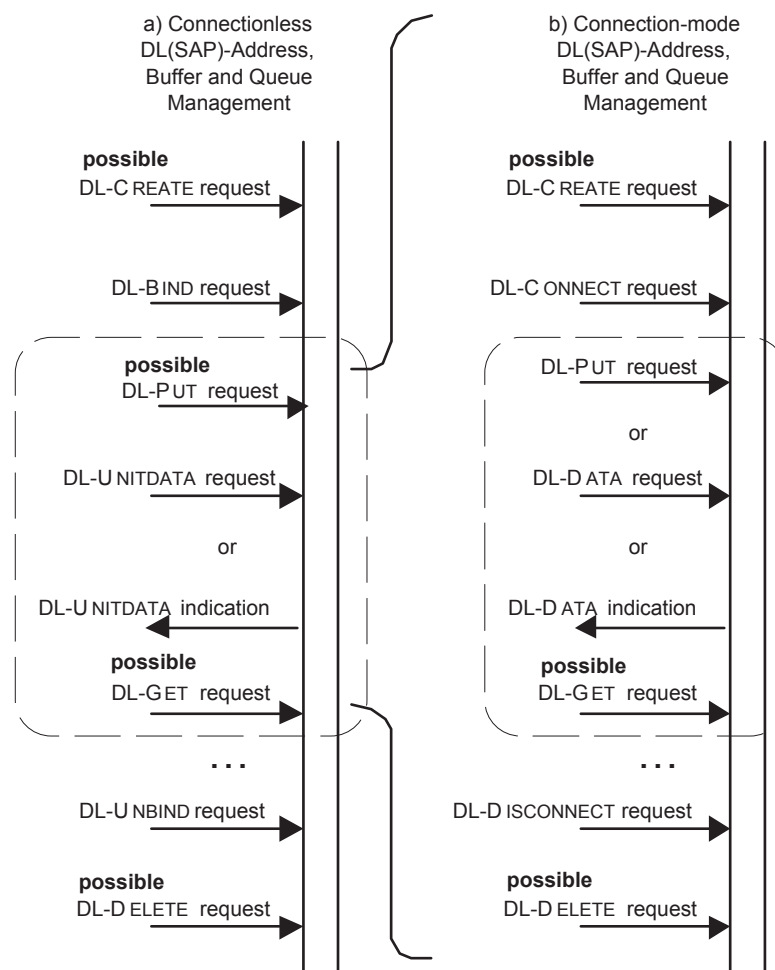
5.3.1 Constraints on sequence of primitives

Subclause 5.3.1 defines the constraints on the sequence in which the primitives defined in 5.4 may occur. The constraints determine the order in which primitives occur, but do not fully specify when they may occur.

The DL(SAP)-address, queue and buffer management primitives and their parameters are summarized in Table 1. The major relationships among the primitives at a single DLE are shown in Figure 4.

Table 1 – Summary of DL(SAP)-address, queue and buffer management primitives and parameters

Service	Primitive	Parameter
Buffer or queue creation	DL-CREATE request	(<i>in</i> Buffer-or-queue DLS-user-identifier, Queuing policy, Maximum DLSDU size, <i>out</i> Status, Buffer-or-queue DL-identifier)
Buffer or queue deletion	DL-DELETE request	(<i>in</i> Buffer-or-queue DL-identifier, <i>out</i> Status)
DL(SAP)-address activation	DL-BIND request	(<i>in</i> DL(SAP)-address DLS-user-identifier, DL(SAP)-address, DL(SAP)-role, Receiving-buffer-or-queue-bindings, Sending-buffer-or-queue-bindings, Default-QoS-as-sender, <i>out</i> Status, DL(SAP)-address DL-identifier)
DL(SAP)-address deactivation	DL-UNBIND request	(<i>in</i> DL(SAP)-address DL-identifier)
Update buffer	DL-PUT request	(<i>in</i> Buffer DL-identifier, DLS-user-data, DLS-user-data-timeliness, <i>out</i> Status)
Copy buffer or dequeue	DL-GET request	(<i>in</i> Buffer-or-queue DL-identifier, <i>out</i> Status, Reported-service-identification-class, Reported-service-identification, DLS-user-data, DLS-user-data-timeliness)
NOTE DL-identifiers in parameters are local and assigned by the DLS-provider and used by the DLS-user to designate a specific DL(SAP)-address, DLCEP, schedule, buffer-or-queue to the DLS-provider at the DLS interface. DLS-user-identifiers in parameters are local and assigned by the DLS-user and used by the DLS-provider to designate a specific DL(SAP)-address, DLCEP, schedule, buffer-or-queue to the DLS-user at the DLS interface.		



NOTE 1 Primitives within the outlined areas can be repeated many times between instances of the primitives in the enclosing areas.

NOTE 2 The entire right-hand part of the figure is another alternative to the outlined area of the left-hand part of the figure, and also can be repeated many times between instances of the primitives in the left-hand enclosing area.

NOTE 3 DL-PUT and DL-GET request primitives both can be used earlier and later than shown.

Figure 4 – Sequence of primitives for the DL(SAP)-address, queue and buffer management DLS

5.4 DL(SAP)-address, queue and buffer management facilities

DL(SAP)-address management facilities bind a DL(SAP)-address to, and unbind a previously bound DL(SAP)-address from, the DLSAP at which the primitive is invoked. Such a binding is required while communicating using the specified DL(SAP)-address.

Queue and buffer management facilities permit, but do not require, a DLS-user to use retentive or non-retentive buffers or specified depth FIFO queues when employing the DLS-provider's data communications facilities (see Figure 5). Since these buffers and queues are managed by the DLS-provider, they support DLS-user interactions and data transfer and scheduling paradigms not available with DLS-user-based queuing.

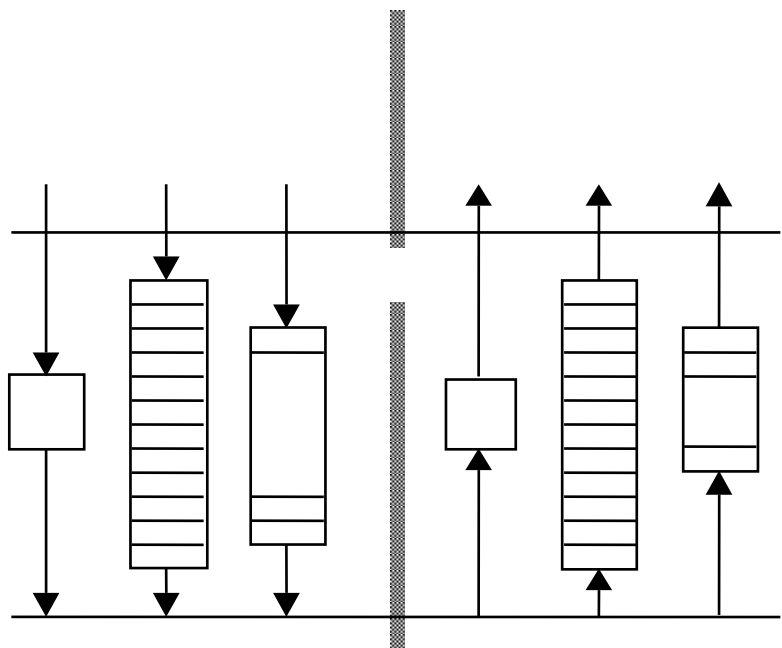


Figure 5 – Supported methods of data management for transmission and delivery

5.4.1 Create

5.4.1.1 Function

The create buffer or queue DLS primitive can be used to create a retentive buffer or non-retentive buffer or limited-depth FIFO queue for later constrained association with a DLSAP — either through a DL(SAP)-address or through a DLCEP. The resulting buffer or FIFO queue initially will be empty.

NOTE This facility may also be provided by local DL-management actions which are beyond the scope of this standard.

5.4.1.2 Types of parameters

Table 2 indicates the primitive and parameters of the create buffer or queue DLS.

Table 2 – DL-buffer-and-queue-management create primitive and parameters

DL-CREATE Parameter name	Request	
	input	output
Buffer-or-queue DLS-user-identifier	M	
Queuing policy	M	
Maximum queue depth	C	
Maximum DLSDU size	M	
Status		M
Buffer-or-queue DL-identifier		C

5.4.1.2.1 Buffer-or-queue DLS-user-identifier

This parameter specifies a means of referring to the buffer or queue in output parameters of other local DLS primitives which convey the name of the buffer or queue from the local DLE to the local DLS-user.

The naming-domain of the buffer-or-queue DLS-user-identifier is the DLS-user’s local-view.

5.4.1.2.2 Queuing policy

This parameter specifies whether to create either

- a) **BUFFER-R** — a retentive buffer, whose contents are not affected by being read, which can be overwritten (as a single atomic action) by either the DLS-provider or DLS-user, and which can be used only with the connectionless unitdata-exchange and connection-oriented data services; or
- b) **BUFFER-NR** — a non-retentive buffer, which is set empty after being read, which can be overwritten (as a single atomic action) by either the DLS-provider or DLS-user, and which can be used only with the connectionless unitdata-exchange service; or
- c) **QUEUE** — a FIFO queue of maximum depth K which contains between zero and K DLSDUs, which will reject attempts to remove DLSDUs when empty and to append DLSDUs when full, and which can be used with the connectionless unitdata-transfer and connection-oriented data services, and for DLSDU-receipt with the connectionless unitdata-exchange service.

The values of the Queuing Policy are BUFFER-R, BUFFER-NR and QUEUE.

NOTE 1 Buffer and queue bindings result from DL-BIND request, DL-CONNECT request and DL-CONNECT response primitives.

NOTE 2 An explicit queue needs to have one output binding and at least one input binding to be useful. Multiple input bindings provide a means of coalescing inputs from many sources into a single queue. Multiple output bindings are not permitted, because a queue-not-empty condition typically causes transmission to be attempted at the DLCEP, or from the DLSAP-address, to which the queue is bound.

NOTE 3 A retentive buffer (BUFFER-R) needs to have one input binding and at least one output binding to be useful. Multiple input bindings are not permitted, because they would permit any data source to overwrite the buffer at any time, with no indication to the buffer users of which source was involved. Multiple output bindings are useful, to share the contents of the buffer among multiple users or to support differing-rate redistribution of cached data within bridges.

NOTE 4 A non-retentive buffer (BUFFER-NR) needs to have one input binding and one output binding to be useful. Multiple input bindings are not useful for the reason stated in 2. Multiple output bindings are not useful; their existence would lead to non-intuitive semantics for the buffer.

NOTE 5 Buffers can be overwritten at any time, with complete loss of the prior contents except to those users which had begun to read the prior contents of the buffer (via DL-GET request primitives or sending DLCEPs) before the overwriting begins. Therefore buffers are well suited for caching of distributed data, where it is desired to retain the most recent value of received information, and are poorly suited for general messaging.

5.4.1.2.2.1 Maximum queue depth

This parameter is present when the Queuing Policy parameter has the value QUEUE. When present, this parameter specifies K , the maximum number of items in the associated queue. The supported values for this parameter should include the values one (1), two (2), three (3), and four (4).

NOTE Implementations of this DLS may extend the range of this parameter to include

- a) values greater than four (4); and
- b) the value zero (0), creating a null queue.

5.4.1.2.3 Maximum DLSDU size

This parameter specifies an upper bound on the size (in octets) of DLSDUs that can be put into the buffer or queue. The maximum size permitted for such DLSDUs may be constrained by a companion DL-protocol specification and by DL-management

NOTE This parameter does not preclude implementations from using a fixed, small set of record sizes when allocating buffers or entries in a queue.

5.4.1.2.4 Status

This parameter allows the DLS-user to determine whether the requested DLS was provided successfully, or failed for the reason specified. The value conveyed in this parameter is as follows:

- a) "success";

- b) “failure — insufficient resources”;
- c) “failure — parameter violates management constraint”;
- d) “failure — number of requests violates management constraint”; or
- e) “failure — reason unspecified”.

NOTE Addition to, or refinement of, this list of values to convey more specific diagnostic and management information is permitted in a DL-protocol standard that provides services as specified in this standard.

If the status is “success”, then the created buffer or queue is subject to the following constraints:

- 1) If a BUFFER-NR or BUFFER-R was created, then it can be written explicitly either
 - by one priority of a receiving DLSAP-address whose DL(SAP)-role is INITIATOR, CONSTRAINED RESPONDER or UNCONSTRAINED RESPONDER;
 - by a receiving DLCEP; or
 - by a DLS-user through a DL-PUT request primitive, if no other binding exists to write the buffer.

It is not permitted to bind such a buffer so that it is written from two distinct sources.

- 2) If a BUFFER-NR or QUEUE was created, then it can be read explicitly either
 - by one priority of a sending DLSAP-address whose DL(SAP)-role is BASIC;
 - by a sending DLCEP; or
 - by a DLS-user through a DL-GET request primitive, if no other binding exists to read the buffer.

It is not permitted to bind such a buffer or queue so that it is read by two distinct sinks.

5.4.1.2.5 Buffer-or-queue DL-identifier

This parameter is present when the Status parameter indicates that the DL-CREATE request primitive was successful. The buffer-or-queue DL-identifier parameter gives the local DLS-user a means of referring to the buffer or queue in input parameters of other local DLS primitives which convey the name of the buffer or queue from the local DLS-user to the local DLE.

The naming-domain of the buffer-or-queue DL-identifier is the DL-local-view.

5.4.1.3 Sequence of primitives

The sequence of primitives in creating, later using, and eventually deleting a buffer or queue is defined in the time sequence diagram of Figure 4.

5.4.2 Delete

5.4.2.1 Function

The delete buffer or queue DLS primitive can be used to delete a buffer or queue created by an earlier create buffer or queue DLS primitive.

NOTE 1 This primitive can only be used to delete a buffer or queue that was created by a prior DL-CREATE request primitive; it cannot be used to delete a buffer or queue that was created by prior local DL-management actions.

NOTE 2 This facility may also be provided by local DL-management actions, which are beyond the scope of this standard.

5.4.2.2 Types of parameters

Table 3 indicates the primitive and parameters of the delete buffer or queue DLS.

Table 3 – DL-buffer-and-queue-management delete primitive and parameters

DL-DELETE Parameter name	Request	
	input	output
Buffer-or-queue DL-identifier	M	
Status		M

5.4.2.2.1 Buffer-or-queue DL-identifier

This parameter specifies the local DL-identifier returned by a successful prior DL-CREATE request primitive whose buffer or queue has not yet been deleted. The DLS-provider will release the local DL-identifier and associated DLS-provider resources.

The DLS-user may not delete a buffer or queue that is still associated with a DLSAP.

NOTE Such associations can occur only as a result of a DL-BIND request, or DL-CONNECT request or response, or of DL-management action.

5.4.2.2.2 Status

This parameter allows the DLS-user to determine whether the requested DLS was provided successfully, or failed for the reason specified. The value conveyed in this parameter is as follows:

- a) "success";
- b) "failure — resource in use";
- c) "failure — management-controlled resource"; or
- d) "failure — reason unspecified".

NOTE Addition to, or refinement of, this list of values to convey more specific diagnostic and management information is permitted in a DL-protocol standard that provides services as specified in this standard.

5.4.2.3 Sequence of primitives

The sequence of primitives in creating and later deleting a buffer or queue is defined in the time sequence diagram of Figure 4.

5.4.3 Bind

5.4.3.1 Function

The bind DL(SAP)-address DLS primitive is used

- a) to associate a DL(SAP)-address with the DLSAP at which the primitive is invoked;
- b) to establish the DL(SAP)'s role, if any, when participating in the DL-Unitdata and DL-Unitdata-Exchange connectionless services at that DL(SAP)-address;
- c) to associate up to six previously created retentive buffers or non-retentive buffers or limited-depth FIFO queues with the various priorities and directions of potential data transfer at the specified DL(SAP)-address; and
- d) to specify default values for some quality-of-service (QoS) attributes for connection-mode and connectionless data transfer services using the specified DL(SAP)-address.

NOTE This facility may also be provided by local DL-management actions, which are beyond the scope of this standard.

5.4.3.2 Types of parameters

Table 4 indicates the primitive and parameters of the bind DL(SAP)-address DLS.