

BS EN 61158-5-11:2008



BSI Standards Publication

# Industrial communication networks — Fieldbus specifications —

Part 5-11: Application layer service definition — Type 11 elements

**bsi.**

...making excellence a habit.  
**AVA** Documents™  
Document Sharing Hub

### National foreword

This British Standard is the UK implementation of EN 61158-5-11:2008. It is identical with IEC 61158-5-11:2007. Together with all of the other sections of BS EN 61158-5, it supersedes BS EN 61158-5:2004 which is withdrawn.

The UK participation in its preparation was entrusted to Technical Committee AMT/7, *Industrial communications - Process measurement and control, including Fieldbus*

A list of organizations represented on this committee can be obtained on request to its secretary.

This publication does not purport to include all the necessary provisions of a contract. Users are responsible for its correct application.

**Compliance with a British Standard cannot confer immunity from legal obligations.**

This British Standard was published under the authority of the Standards Policy and Strategy Committee on 31 January 2009.

© BSI 2009

ISBN 978 0 580 61596 2

ICS 25.040.40; 35.100.70

### Amendments issued since publication

| Amd. No. | Date | Text affected |
|----------|------|---------------|
|----------|------|---------------|

**Industrial communication networks -  
Fieldbus specifications -  
Part 5-11: Application layer service definition -  
Type 11 elements  
(IEC 61158-5-11:2007)**

Réseaux de communication industriels -  
Spécifications des bus de terrain -  
Partie 5-11: Définition des services  
des couches d'application -  
Éléments de type 11  
(CEI 61158-5-11:2007)

Industrielle Kommunikationsnetze -  
Feldbusse -  
Teil 5-11: Dienstfestlegungen  
des Application Layer  
(Anwendungsschicht) -  
Typ 11-Elemente  
(IEC 61158-5-11:2007)

This European Standard was approved by CENELEC on 2008-02-01. 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 Central Secretariat 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 Central Secretariat has the same status as the official versions.

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

**CENELEC**

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

**Central Secretariat: rue de Stassart 35, B - 1050 Brussels**

## Foreword

The text of document 65C/475/FDIS, future edition 1 of IEC 61158-5-11, 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 was approved by CENELEC as EN 61158-5-11 on 2008-02-01.

This and the other parts of the EN 61158-5 series supersede EN 61158-5:2004.

With respect to EN 61158-5:2004 the following changes were made:

- deletion of Type 6 fieldbus for lack of market relevance;
- addition of new fieldbus types;
- partition into multiple parts numbered 5-2, 5-3, ..., 5-20.

The following dates were fixed:

- latest date by which the EN has to be implemented  
at national level by publication of an identical  
national standard or by endorsement (dop) 2008-11-01
- latest date by which the national standards conflicting  
with the EN have to be withdrawn (dow) 2011-02-01

NOTE Use of some of the associated protocol types is restricted by their intellectual-property-right holders. In all cases, the commitment to limited release of intellectual-property-rights made by the holders of those rights permits a particular data-link layer protocol type to be used with physical layer and application layer protocols in type combinations as specified explicitly in the EN 61784 series. Use of the various protocol types in other combinations may require permission from their respective intellectual-property-right holders.

Annex ZA has been added by CENELEC.

---

## Endorsement notice

The text of the International Standard IEC 61158-5-11:2007 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 61158-4-11 | NOTE Harmonized as EN 61158-4-11:2008 (not modified). |
| IEC 61158-6-11 | NOTE Harmonized as EN 61158-6-11:2008 (not modified). |
| IEC 61784-2    | NOTE Harmonized as EN 61784-2:2008 (not modified).    |

## Annex ZA (normative)

### Normative references to international publications with their corresponding European publications

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

NOTE When an international publication has been modified by common modifications, indicated by (mod), the relevant EN/HD applies.

| <u>Publication</u> | <u>Year</u>     | <u>Title</u>                                                                                                                       | <u>EN/HD</u>      | <u>Year</u>        |
|--------------------|-----------------|------------------------------------------------------------------------------------------------------------------------------------|-------------------|--------------------|
| IEC 60559          | - <sup>1)</sup> | Binary floating-point arithmetic for microprocessor systems                                                                        | HD 592 S1         | 1991 <sup>2)</sup> |
| IEC 61131-1        | - <sup>1)</sup> | Programmable controllers - Part 1: General information                                                                             | EN 61131-1        | 2003 <sup>2)</sup> |
| IEC 61131-3        | - <sup>1)</sup> | Programmable controllers - Part 3: Programming languages                                                                           | EN 61131-3        | 2003 <sup>2)</sup> |
| IEC/TR 61158-1     | 2007            | Industrial communication networks - Fieldbus specifications - Part 1: Overview and guidance for the IEC 61158 and IEC 61784 series | -                 |                    |
| IEC 61158-3-11     | - <sup>1)</sup> | Industrial communication networks - Fieldbus specifications - Part 3-11: Data-link layer service definition - Type 11 elements     | EN 61158-3-11     | 2008 <sup>2)</sup> |
| ISO/IEC 646        | - <sup>1)</sup> | Information technology - ISO 7-bit coded character set for information interchange                                                 | -                 | -                  |
| ISO/IEC 7498-1     | - <sup>1)</sup> | Information technology - Open Systems Interconnection - Basic Reference Model: The Basic Model                                     | EN ISO/IEC 7498-1 | 1995 <sup>2)</sup> |
| ISO/IEC 7498-3     | - <sup>1)</sup> | Information technology - Open Systems Interconnection - Basic Reference Model: Naming and addressing                               | -                 | -                  |
| ISO/IEC 8822       | - <sup>1)</sup> | Information technology - Open Systems Interconnection - Presentation service definition                                            | -                 | -                  |
| ISO/IEC 8824       | - <sup>1)</sup> | Information technology - Open Systems Interconnection - Specification of Abstract Syntax Notation One (ASN.1)                      | -                 | -                  |
| ISO/IEC 8859-1     | - <sup>1)</sup> | Information technology - 8-bit single-byte coded graphic character sets - Part 1: Latin alphabet No.1                              | -                 | -                  |
| ISO/IEC 9545       | - <sup>1)</sup> | Information technology - Open Systems Interconnection - Application Layer structure                                                | -                 | -                  |

<sup>1)</sup> Undated reference.

<sup>2)</sup> Valid edition at date of issue.

| <u>Publication</u> | <u>Year</u>     | <u>Title</u>                                                                                                                    | <u>EN/HD</u> | <u>Year</u> |
|--------------------|-----------------|---------------------------------------------------------------------------------------------------------------------------------|--------------|-------------|
| ISO/IEC 10646-1    | - <sup>1)</sup> | Information technology - Universal Multiple-Octet Coded Character Set (UCS) - Part 1: Architecture and Basic Multilingual Plane | -            | -           |
| ISO/IEC 10731      | - <sup>1)</sup> | Information technology - Open Systems Interconnection - Basic reference model - Conventions for the definition of OSI services  | -            | -           |

## CONTENTS

|                                                                  |    |
|------------------------------------------------------------------|----|
| INTRODUCTION.....                                                | 6  |
| 1 Scope.....                                                     | 7  |
| 1.1 Overview.....                                                | 7  |
| 1.2 Specifications.....                                          | 8  |
| 1.3 Conformance.....                                             | 8  |
| 2 Normative references .....                                     | 8  |
| 3 Terms and definitions, abbreviations, and conventions .....    | 10 |
| 3.1 ISO/IEC 7498-1 terms .....                                   | 10 |
| 3.2 ISO/IEC 8822 terms .....                                     | 10 |
| 3.3 ISO/IEC 9545 terms .....                                     | 10 |
| 3.4 ISO/IEC 8824 terms .....                                     | 10 |
| 3.5 Fieldbus data-link layer terms.....                          | 10 |
| 3.6 Fieldbus application layer type-specific definitions .....   | 11 |
| 3.7 Abbreviations and symbols.....                               | 21 |
| 3.8 Conventions .....                                            | 23 |
| 3.9 Nomenclature for references within this standard .....       | 26 |
| 4 Concepts.....                                                  | 26 |
| 4.1 Common concepts.....                                         | 26 |
| 4.2 Type specific concepts .....                                 | 26 |
| 5 Data type ASE.....                                             | 31 |
| 5.1 General.....                                                 | 31 |
| 5.2 Formal definition of data type objects .....                 | 34 |
| 5.3 FAL defined data types.....                                  | 36 |
| 5.4 Data type ASE service specification .....                    | 72 |
| 6 Communication model specification .....                        | 73 |
| 6.1 ASEs.....                                                    | 73 |
| 6.2 ARs.....                                                     | 81 |
| 6.3 Summary of FAL classes .....                                 | 82 |
| 6.4 Permitted FAL services by AREP role.....                     | 83 |
| Bibliography.....                                                | 84 |
| Figure 1 – RTE-TCnet communication profile.....                  | 27 |
| Figure 2 – Application example by using the CM.....              | 28 |
| Figure 3 – Global common-memory concept over the RTE-TCnet ..... | 29 |
| Figure 4 – Relationship of Common Memory and AREP .....          | 30 |
| Figure 5 – Structure of Type 11 AL ASE .....                     | 31 |
| Figure 6 – Data type class hierarchy example.....                | 32 |
| Figure 7 – Common memory publisher/subscriber model .....        | 76 |
| Table 1 – PERSISTDEF .....                                       | 44 |
| Table 2 – VARTYPE .....                                          | 45 |
| Table 3 – ITEMQUALITYDEF.....                                    | 46 |
| Table 4 – STATEDEF .....                                         | 50 |
| Table 5 – GROUPEXCEPTIONDEF .....                                | 50 |

|                                                   |    |
|---------------------------------------------------|----|
| Table 6 – ACCESSRIGHTSDEF .....                   | 50 |
| Table 7 – HRESULT .....                           | 51 |
| Table 8 – UUID .....                              | 58 |
| Table 9 – Data type names for value .....         | 70 |
| Table 10 – UUID .....                             | 72 |
| Table 11 – Update memory service parameters ..... | 74 |
| Table 12 – Memory-status service parameters ..... | 75 |
| Table 13 – AR-Unconfirmed Send .....              | 79 |
| Table 14 – AR-get buffered message service .....  | 80 |
| Table 15 – AR-Status service .....                | 80 |
| Table 16 – FAL class summary .....                | 82 |
| Table 17 – Services by AREP role .....            | 83 |

## INTRODUCTION

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/TR 61158-1.

The application service is provided by the application protocol making use of the services available from the data-link or other immediately lower layer. This standard defines the application service characteristics that fieldbus applications and/or system management may exploit.

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 application layer service defined in this standard is a conceptual architectural service, independent of administrative and implementation divisions.

## INDUSTRIAL COMMUNICATION NETWORKS – FIELDBUS SPECIFICATIONS –

### Part 5-11: Application layer service definition – Type 11 elements

#### 1 Scope

##### 1.1 Overview

The fieldbus Application Layer (FAL) provides user programs with a means to access the Fieldbus communication environment. In this respect, the FAL can be viewed as a “window between corresponding application programs.”

This part of IEC 61158 provides common elements for basic time-critical and non-time-critical messaging communications between application programs in an automation environment and material specific to Type 11 fieldbus. 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 part of IEC 61158 defines in an abstract way the externally visible service provided by the different Types of fieldbus Application Layer in terms of

- a) an abstract model for defining application resources (objects) capable of being manipulated by users via the use of the FAL service,
- b) the primitive actions and events of the service;
- c) the parameters associated with each primitive action and event, and the form which they take; and
- d) the interrelationship between these actions and events, and their valid sequences.

The purpose of this part of IEC 61158 is to define the services provided to

- 1) the FAL user at the boundary between the user and the Application Layer of the Fieldbus Reference Model, and
- 2) Systems Management at the boundary between the Application Layer and Systems Management of the Fieldbus Reference Model.

This part of IEC 61158 specifies the structure and services of the IEC fieldbus Application Layer, in conformance with the OSI Basic Reference Model (ISO/IEC 7498) and the OSI Application Layer Structure (ISO/IEC 9545).

FAL services and protocols are provided by FAL application-entities (AE) contained within the application processes. The FAL AE is composed of a set of object-oriented Application Service Elements (ASEs) and a Layer Management Entity (LME) that manages the AE. The ASEs provide communication services that operate on a set of related application process object (APO) classes. One of the FAL ASEs is a management ASE that provides a common set of services for the management of the instances of FAL classes.

Although these services specify, from the perspective of applications, how request and responses are issued and delivered, they do not include a specification of what the requesting and responding applications are to do with them. That is, the behavioral aspects of the applications are not specified; only a definition of what requests and responses they can send/receive is specified. This permits greater flexibility to the FAL users in standardizing

such object behavior. In addition to these services, some supporting services are also defined in this standard to provide access to the FAL to control certain aspects of its operation.

## 1.2 Specifications

The principal objective of this part of IEC 61158 is to specify the characteristics of conceptual application layer services suitable for time-critical communications, and thus supplement the OSI Basic Reference Model in guiding the development of application layer protocols for time-critical communications.

A secondary objective is to provide migration paths from previously-existing industrial communications protocols. It is this latter objective which gives rise to the diversity of services standardized as the various Types of IEC 61158, and the corresponding protocols standardized in IEC 61158-6.

This specification may be used as the basis for formal Application 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, and
- b) the correlation of paired request and confirm, or indication and response, primitives.

## 1.3 Conformance

This part of IEC 61158 do not specify individual implementations or products, nor do they constrain the implementations of application layer entities within industrial automation systems.

There is no conformance of equipment to this application layer service definition standard. Instead, conformance is achieved through implementation of conforming application layer protocols that fulfil any given Type of application layer services as defined in this part of IEC 61158.

## 2 Normative references

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

IEC 60559, *Binary floating-point arithmetic for microprocessor systems*

IEC 61131-1, *Programmable controllers – Part 1: General information*

IEC 61131-3, *Programmable controllers – Part 3: Programming languages*

IEC/TR 61158-1 (Ed.2.0), *Industrial communication networks – Fieldbus specifications – Part 1: Overview and guidance for the IEC 61158 and IEC 61784 series*

IEC 61158-3-11, *Industrial communication networks – Fieldbus specifications - Part 3-11: Data-link layer service definition – Type 11 elements*

ISO/IEC 646, *Information technology – ISO 7-bit coded character set for information interchange*

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

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

ISO/IEC 8822, *Information technology – Open Systems Interconnection – Presentation service definition*

ISO/IEC 8824, *Information Technology – Abstract Syntax notation One (ASN-1): Specification of basic notation*

ISO/IEC 8859-1, *Information technology – 8-bit single-byte coded graphic character sets – Part 1: Latin alphabet No. 1*

ISO/IEC 9545, *Information technology – Open Systems Interconnection – Application Layer structure*

ISO/IEC 10646-1, *Information technology – Universal Multiple-Octet Coded Character Set (UCS) – Architecture and Basic Multilingual Plane*

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

### 3 Terms and definitions, abbreviations, and conventions

For the purposes of this document, the following terms as defined in these publications apply:

#### 3.1 ISO/IEC 7498-1 terms

- a) application entity
- b) application process
- c) application protocol data unit
- d) application service element
- e) application entity invocation
- f) application process invocation
- g) application transaction
- h) real open system
- i) transfer syntax

#### 3.2 ISO/IEC 8822 terms

For the purposes of this document, the following terms as defined in ISO/IEC 8822 apply:

- a) abstract syntax
- b) presentation context

#### 3.3 ISO/IEC 9545 terms

For the purposes of this document, the following terms as defined in ISO/IEC 9545 apply:

- a) application-association
- b) application-context
- c) application context name
- d) application-entity-invocation
- e) application-entity-type
- f) application-process-invocation
- g) application-process-type
- h) application-service-element
- i) application control service element

#### 3.4 ISO/IEC 8824 terms

For the purposes of this document, the following terms as defined in ISO/IEC 8824 apply:

- a) object identifier
- b) type

#### 3.5 Fieldbus data-link layer terms

For the purposes of this document, the following terms as defined in IEC 61158-3-11 apply.

- a) DLCEP
- b) DLC
- c) DLPDU
- d) DLSDU

- e) DLSAP
- f) link
- g) network address
- h) node
- i) scheduled
- j) unscheduled

### **3.6 Fieldbus application layer type-specific definitions**

For the purposes of this part of IEC 61158, the following terms and definitions apply.

#### **3.6.1**

##### **access protection**

limitation of the usage of an application object to one client

#### **3.6.2**

##### **active connection control object**

instance of a certain FAL class that abstracts the interconnection facility (as Consumer and Provider) of an automation device

#### **3.6.3**

##### **address assignment table**

mapping of the client's internal I/O-Data object storage to the decentralized input and output data objects

#### **3.6.4**

##### **allocate**

take a resource from a common area and assign that resource for the exclusive use of a specific entity

#### **3.6.5**

##### **application**

function or data structure for which data is consumed or produced

#### **3.6.6**

##### **application layer interoperability**

capability of application entities to perform coordinated and cooperative operations using the services of the FAL

#### **3.6.7**

##### **application objects**

multiple object classes that manage and provide a run time exchange of messages across the network and within the network device

#### **3.6.8**

##### **application process**

part of a distributed application on a network, which is located on one device and unambiguously addressed

#### **3.6.9**

##### **application process identifier**

distinguishes multiple application processes used in a device

**3.6.10****application process object**

component of an application process that is identifiable and accessible through an FAL application relationship

NOTE Application process object definitions are composed of a set of values for the attributes of their class (see the definition for Application Process Object Class Definition). Application process object definitions may be accessed remotely using the services of the FAL Object Management ASE. FAL Object Management services can be used to load or update object definitions, to read object definitions, and to dynamically create and delete application objects and their corresponding definitions.

**3.6.11****application process object class**

a class of application process objects defined in terms of the set of their network-accessible attributes and services

**3.6.12****application relationship**

cooperative association between two or more application-entity-invocations for the purpose of exchange of information and coordination of their joint operation. This relationship is activated either by the exchange of application-protocol-data-units or as a result of preconfiguration activities

**3.6.13****application relationship application service element**

application-service-element that provides the exclusive means for establishing and terminating all application relationships

**3.6.14****application relationship endpoint**

context and behavior of an application relationship as seen and maintained by one of the application processes involved in the application relationship

NOTE Each application process involved in the application relationship maintains its own application relationship endpoint.

**3.6.15****attribute**

description of an externally visible characteristic or feature of an object

NOTE The attributes of an object contain information about variable portions of an object. Typically, they provide status information or govern the operation of an object. Attributes may also affect the behaviour of an object. Attributes are divided into class attributes and instance attributes.

**3.6.16****behaviour**

indication of how an object responds to particular events

**3.6.17****bit-no**

designates the number of a bit in a bitstring or an octet

**3.6.18****channel**

single physical or logical link of an input or output application object of a server to the process

**3.6.19****channel related diagnosis**

information concerning a specific element of an input or output application object, provided for maintenance purposes

EXAMPLE: validity of data

**3.6.20****class**

a set of objects, all of which represent the same kind of system component

NOTE A class is a generalisation of an object; a template for defining variables and methods. All objects in a class are identical in form and behaviour, but usually contain different data in their attributes.

**3.6.21****class attributes**

attribute that is shared by all objects within the same class

**3.6.22****class code**

unique identifier assigned to each object class

**3.6.23****class specific service**

service defined by a particular object class to perform a required function which is not performed by a common service

NOTE A class specific object is unique to the object class which defines it.

**3.6.24****client**

- a) object which uses the services of another (server) object to perform a task
- b) initiator of a message to which a server reacts

**3.6.25****common memory**

virtual common memory over the network for the Type 11 fieldbus, which is shared with the nodes participating in the Type 11 fieldbus and is primarily used for the real-time communications by the TCC data service

**3.6.26****communication objects**

components that manage and provide a run time exchange of messages across the network

EXAMPLES: Connection Manager object, Unconnected Message Manager (UCMM) object, and Message Router object

**3.6.27****configuration check**

comparison of the expected I/O-Data object structuring of the client with the real I/O-Data object structuring to the server in the start-up phase

**3.6.28****configuration data base**

interconnection information maintained by the ACCO ASE

**3.6.29****configuration fault**

an unacceptable difference between the expected I/O-Data object structuring and the real I/O-Data object structuring, as detected by the server

**3.6.30****configuration identifier**

representation of a portion of I/O Data of a single input- and/or output-module of a server

**3.6.31****connection**

logical binding between application objects that may be within the same or different devices

NOTE 1 Connections may be either point-to-point or multipoint.

NOTE 2 The logical link between sink and source of attributes and services at different custom interfaces of RT-Auto ASES is referred to as interconnection. There is a distinction between data and event interconnections. The logical link and the data flow between sink and source of automation data items is referred to as data interconnection. The logical link and the data flow between sink (method) and source (event) of operational services is referred to as event interconnection.

### **3.6.32**

#### **connection channel**

description of a connection between a sink and a source of data items

### **3.6.33**

#### **connection ID (CID)**

identifier assigned to a transmission that is associated with a particular connection between producers and consumers, providing a name for a specific piece of application information

### **3.6.34**

#### **connection path**

an octet stream that defines the application object to which a connection instance applies

### **3.6.35**

#### **connection point**

buffer which is represented as a subinstance of an Assembly object

### **3.6.36**

#### **consume**

act of receiving data from a producer

### **3.6.37**

#### **consumer**

node or sink that is receiving data from a producer

### **3.6.38**

#### **consuming application**

application that consumes data

### **3.6.39**

#### **consumerID**

unambiguous identifier within the scope of the ACCO assigned by the consumer to recognize the internal data of a configured interconnection sink

### **3.6.40**

#### **control commands**

action invocations transferred from client to server to clear outputs, freeze inputs and/or synchronize outputs

### **3.6.41**

#### **conveyance path**

unidirectional flow of APDUs across an application relationship

### **3.6.42**

#### **cyclic**

repetitive in a regular manner

### **3.6.43**

#### **data consistency**

means for coherent transmission and access of the input- or output-data object between and within client and server

**3.6.44****data marshalling**

the encoding of parameters of the FAL service primitives with respect to their interface definition

<Type 10> NOTE This is part of the abstract ORPC model.

**3.6.45****dedicated AR**

AR used directly by the FAL User

NOTE On Dedicated ARs, only the FAL Header and the user data are transferred.

**3.6.46****default DL-address**

value 126 as an initial value for DL-address, which has to be changed (e.g. by assignment of an DL-address via the fieldbus) before operation with a DP-master (class 1)

**3.6.47****device**

physical hardware connected to the link

NOTE A device may contain more than one node.

**3.6.48****device profile**

a collection of device dependent information and functionality providing consistency between similar devices of the same device type

**3.6.49****diagnosis information**

all data available at the server for maintenance purposes

**3.6.50****diagnosis information collection**

system diagnosis information that is assembled at the client side

**3.6.51****dynamic AR**

AR that requires the use of the AR establishment procedures to place it into an established state

**3.6.52****end node**

producing or consuming node

**3.6.53****endpoint**

one of the communicating entities involved in a connection

**3.6.54****engineering**

abstract term that characterizes the client application or device responsible for configuring an automation system via interconnecting data items

**3.6.55****error**

discrepancy between a computed, observed or measured value or condition and the specified or theoretically correct value or condition

**3.6.56****error class**

general grouping for related error definitions and corresponding error codes

**3.6.57****error code**

identification of a specific type of error within an error class

**3.6.58****event**

an instance of a change of conditions

**3.6.59****FAL subnet**

subnetworks composed of one or more data link segments, identified by a subset of the network address

NOTE FAL subnets are permitted to contain bridges but not routers.

**3.6.60****FIFO variable**

a Variable Object class, composed of a set of homogeneously typed elements, where the first written element is the first element that can be read

NOTE On the fieldbus only one, complete element can be transferred as a result of one service invocation.

**3.6.61****frame**

denigrated synonym for DLPDU

**3.6.62****group**

- a) <general> a general term for a collection of objects. Specific uses:
- b) <addressing> when describing an address, an address that identifies more than one entity

**3.6.63****interface**

- (a) shared boundary between two functional units, defined by functional characteristics, signal characteristics, or other characteristics as appropriate
- (b) collection of FAL class attributes and services that represents a specific view on the FAL class

**3.6.64****interface definition language**

syntax and semantics of describing service parameters in a formal way

NOTE This description is the input for the ORPC model, especially for the ORPC wire protocol.

**3.6.65****interface pointer**

key attribute that unambiguously addresses an object interface instance

**3.6.66****invocation**

act of using a service or other resource of an application process

NOTE Each invocation represents a separate thread of control that may be described by its context. Once the service completes, or use of the resource is released, the invocation ceases to exist. For service invocations, a service that has been initiated but not yet completed is referred to as an outstanding service invocation.

**3.6.67****I/O data**

object designated to be transferred cyclically for the purpose of processing

**3.6.68****identifier related diagnosis**

information dedicated to modules for maintenance purpose

**3.6.69****index**

address of an object within an application process

**3.6.70****instance**

the actual physical occurrence of an object within a class that identifies one of many objects within the same object class

EXAMPLE California is an instance of the object class state.

NOTE The terms object, instance, and object instance are used to refer to a specific instance.

**3.6.71****instance attributes**

attribute that is unique to an object instance and not shared by the object class

**3.6.72****instantiated**

object that has been created in a device

**3.6.73****logical device**

a certain FAL class that abstracts a software component or a firmware component as an autonomous self-contained facility of an automation device

**3.6.74****manufacturer ID**

identification of each product manufacturer by a unique number

**3.6.75****management information**

network-accessible information that supports managing the operation of the fieldbus system, including the application layer

NOTE Managing includes functions such as controlling, monitoring, and diagnosing.

**3.6.76****master parameter set**

the configuration and parameterization data of all DP-slaves that are assigned to the corresponding DP-master and the bus parameters

**3.6.77****member**

piece of an attribute that is structured as an element of an array

**3.6.78****message router**

object within a node that distributes messaging requests to appropriate application objects

**3.6.79****method**

<object> a synonym for an operational service which is provided by the server ASE and invoked by a client

**3.6.80****module**

- a) <general> hardware or logical component of a physical device
- b) <Type 3> addressable unit inside the DP-slave

**3.6.81****multipoint connection**

connection from one node to many

NOTE Multipoint connections allow messages from a single producer to be received by many consumer nodes.

**3.6.82****network**

a set of nodes connected by some type of communication medium, including any intervening repeaters, bridges, routers and lower-layer gateways

**3.6.83****object**

abstract representation of a particular component within a device, usually a collection of related data (in the form of variables) and methods (procedures) for operating on that data that have clearly defined interface and behaviour

**3.6.84****object remote procedure call**

model for object oriented or component based remote method invocation

**3.6.85****object specific service**

service unique to the object class which defines it

**3.6.86****originator**

client responsible for establishing a connection path to the target

**3.6.87****peer**

role of an AR endpoint in which it is capable of acting as both client and server

**3.6.88****physical device**

<general> an automation or other network device

<Type10> a certain FAL class that abstracts the hardware facilities of an automation device

**3.6.89****point-to-point connection**

connection that exists between exactly two application objects

**3.6.90****pre-defined AR endpoint**

AR endpoint that is defined locally within a device without use of the create service

NOTE Pre-defined ARs that are not pre-established are established before being used

**3.6.91****pre-established AR endpoint**

AR endpoint that is placed in an established state during configuration of the AEs that control its endpoints

**3.6.92****process data**

object(s) which are already pre-processed and transferred acyclically for the purpose of information or further processing

**3.6.93****produce**

act of sending data to be received by a consumer

**3.6.94****producer**

node that is responsible for sending data

**3.6.95****property**

a general term for descriptive information about an object

**3.6.96****provider**

source of a data connection

**3.6.97****providerID**

an unambiguous identifier within the scope of the ACCO assigned by the provider to recognize the internal data of a configured interconnection source

**3.6.98****publisher**

role of an AR endpoint that transmits APDUs onto the fieldbus for consumption by one or more subscribers

NOTE A publisher may not be aware of the identity or the number of subscribers and it may publish its APDUs using a dedicated AR.

**3.6.99****publishing manager**

role of an AR endpoint in which it issues one or more confirmed service request APDUs to a publisher to request the publisher to publish a specified object. Two types of publishing managers are defined by this standard, pull publishing managers and push publishing managers, each of which is defined separately

**3.6.100****pull publisher**

type of publisher that publishes an object in response to a request received from its pull publishing manager

**3.6.101****pull publishing manager**

type of publishing manager that requests that a specified object be published in a corresponding response APDU

**3.6.102****push publisher**

type of publisher that publishes an object in an unconfirmed service request APDU

**3.6.103****push publishing manager**

type of publishing manager that requests that a specified object be published using an unconfirmed service

**3.6.104****pull subscriber**

type of subscriber that recognizes received confirmed service response APDUs as published object data

**3.6.105****push subscriber**

type of subscriber that recognizes received unconfirmed service request APDUs as published object data

**3.6.106****quality code aware**

attribute of the RT-Auto class that indicates that an RT-Auto object uses a status code for its data items

**3.6.107****quality code**

additional status information of a data item

**3.6.108****quality code unaware**

opposite of quality code aware

**3.6.109****real configuration**

input and output data structure of the DP-slave, including definition of data consistency

**3.6.110****resource**

a processing or information capability of a subsystem

**3.6.111****route endpoint**

object container containing Variable Objects of a variable class

**3.6.112****RT-auto**

an FAL class that abstracts the automation function as a process-related component of an automation device

**3.6.113****runtime object model**

objects that exist in a device together with their interfaces and methods that are accessible

**3.6.114****serial number**

<Type 2> a unique 32-bit integer value assigned by each manufacturer to every device having Type 2 communication capabilities

NOTE The Manufacturer ID and serial number jointly form a unique identifier for each device.

**3.6.115****server**

- a) role of an AREP in which it returns a confirmed service response APDU to the client that initiated the request
- b) object which provides services to another (client) object

**3.6.116****service**

operation or function than an object and/or object class performs upon request from another object and/or object class

**3.6.117****slot**

address of a module within a DP-slave

**3.6.118****subscriber**

role of an AREP in which it receives APDUs produced by a publisher

**3.6.119****sync**

function at the DP-slaves for simultaneous data transfer between the output data object and the process

**3.6.120****target**

end-node to which a connection is established

**3.6.121****unconnected message manager (UCMM)**

component within a node that transmits and receives unconnected explicit messages and sends them directly to the Message Router object

**3.6.122****unconnected service**

messaging service which does not rely on the set up of a connection between devices before allowing information exchanges

**3.7 Abbreviations and symbols**

|       |                                                    |
|-------|----------------------------------------------------|
| ACCO  | Active connection control object                   |
| AE    | Application entity                                 |
| AL    | Application layer                                  |
| ALME  | Application layer management entity                |
| ALP   | Application layer protocol                         |
| APO   | Application object                                 |
| AP    | Application process                                |
| APDU  | Application protocol data unit                     |
| API   | Application process identifier                     |
| AR    | Application relationship                           |
| AREP  | Application relationship endpoint                  |
| ASCII | American Standard Code for Information Interchange |
| ASE   | Application service element                        |

|        |                                                   |
|--------|---------------------------------------------------|
| CID    | Connection ID                                     |
| CIM    | Computer integrated manufacturing                 |
| CIP    | Control and information protocol                  |
| CM     | Common memory                                     |
| CM_API | Actual packet interval                            |
| CM_RPI | Requested packet interval                         |
| Cnf    | Confirmation                                      |
| COR    | Connection originator                             |
| CR     | Communication relationship                        |
| CREP   | Communication relationship endpoint               |
| DL-    | (as a prefix) data-link-                          |
| DLC    | Data-link connection                              |
| DLCEP  | Data-link connection endpoint                     |
| DLL    | Data-link layer                                   |
| DLM    | Data-link-management                              |
| DLSAP  | Data-link service access point                    |
| DLSDU  | DL-service-data-unit                              |
| DNS    | Domain name service                               |
| DP     | Decentralised peripherals                         |
| FAL    | Fieldbus application layer                        |
| FIFO   | First-in First-out                                |
| HMI    | Human-machine interface                           |
| ID     | Identifier                                        |
| IDL    | Interface definition language                     |
| IEC    | International Electrotechnical Commission         |
| Ind    | Indication                                        |
| IP     | Internet protocol                                 |
| ISO    | International Organization for Standardization    |
| LDev   | Logical device                                    |
| LME    | Layer management entity                           |
| O2T    | Originator to target (connection characteristics) |
| O⇒T    | Originator to target (connection characteristics) |
| ORPC   | Object remote procedure call                      |
| OSI    | Open Systems Interconnect                         |
| PDev   | Physical device                                   |
| PDU    | Protocol data unit                                |
| PL     | Physical layer                                    |
| QoS    | Quality of service                                |
| QC     | Quality code                                      |
| REP    | Route endpoint                                    |
| Req    | Request                                           |
| Rsp    | Response                                          |

|       |                                                             |
|-------|-------------------------------------------------------------|
| RT    | Runtime                                                     |
| SAP   | Service access point                                        |
| SCL   | Security level                                              |
| SDU   | Service data unit                                           |
| SEM   | State event matrix                                          |
| SMIB  | System management information base                          |
| SMK   | System management kernel                                    |
| STD   | State transition diagram, used to describe object behaviour |
| S-VFD | Simple virtual field device                                 |
| T2O   | Target to originator (connection characteristics)           |
| T⇒O   | Target to originator (connection characteristics)           |
| VAO   | Variable object                                             |

### 3.8 Conventions

#### 3.8.1 Overview

The FAL is defined as a set of object-oriented ASEs. Each ASE is specified in a separate subclause. Each ASE specification is composed of two parts, its class specification, and its service specification.

The class specification defines the attributes of the class. The attributes are accessible from instances of the class using the Object Management ASE services specified in Clause 5 of this standard. The service specification defines the services that are provided by the ASE.

#### 3.8.2 General conventions

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

#### 3.8.3 Conventions for class definitions

Class definitions are described using templates. Each template consists of a list of attributes for the class. The general form of the template is shown below:

|                      |                        |
|----------------------|------------------------|
| <b>FAL ASE:</b>      | <b>ASE Name</b>        |
| <b>CLASS:</b>        | <b>Class Name</b>      |
| <b>CLASS ID:</b>     | #                      |
| <b>PARENT CLASS:</b> | Parent Class Name      |
| <b>ATTRIBUTES:</b>   |                        |
| 1 (o) Key Attribute: | numeric identifier     |
| 2 (o) Key Attribute: | name                   |
| 3 (m) Attribute:     | attribute name(values) |
| 4 (m) Attribute:     | attribute name(values) |
| 4.1 (s) Attribute:   | attribute name(values) |
| 4.2 (s) Attribute:   | attribute name(values) |
| 4.3 (s) Attribute:   | attribute name(values) |
| 5. (c) Constraint:   | constraint expression  |
| 5.1 (m) Attribute:   | attribute name(values) |
| 5.2 (o) Attribute:   | attribute name(values) |
| 6 (m) Attribute:     | attribute name(values) |
| 6.1 (s) Attribute:   | attribute name(values) |
| 6.2 (s) Attribute:   | attribute name(values) |

**SERVICES:**

- 1 (o) OpsService: service name
  2. (c) Constraint: constraint expression
  - 2.1 (o) OpsService: service name
  - 3 (m) MgtService: service name
- (1) The "FAL ASE:" entry is the name of the FAL ASE that provides the services for the class being specified.
  - (2) The "CLASS:" entry is the name of the class being specified. All objects defined using this template will be an instance of this class. The class may be specified by this standard, or by a user of this standard.
  - (3) The "CLASS ID:" entry is a number that identifies the class being specified. This number is unique within the FAL ASE that will provide the services for this class. When qualified by the identity of its FAL ASE, it unambiguously identifies the class within the scope of the FAL. The value "NULL" indicates that the class cannot be instantiated. Class IDs between 1 and 255 are reserved by this standard to identify standardized classes. They have been assigned to maintain compatibility with existing national standards. CLASS IDs between 256 and 2048 are allocated for identifying user defined classes.
  - (4) The "PARENT CLASS:" entry is the name of the parent class for the class being specified. All attributes defined for the parent class and inherited by it are inherited for the class being defined, and therefore do not have to be redefined in the template for this class.
- NOTE The parent-class "TOP" indicates that the class being defined is an initial class definition. The parent class TOP is used as a starting point from which all other classes are defined. The use of TOP is reserved for classes defined by this standard.
- (5) The "ATTRIBUTES" label indicate that the following entries are attributes defined for the class.
    - a) Each of the attribute entries contains a line number in column 1, a mandatory (m) / optional (o) / conditional (c) / selector (s) indicator in column 2, an attribute type label in column 3, a name or a conditional expression in column 4, and optionally a list of enumerated values in column 5. In the column following the list of values, the default value for the attribute may be specified.
    - b) Objects are normally identified by a numeric identifier or by an object name, or by both. In the class templates, these key attributes are defined under the key attribute.
    - c) The line number defines the sequence and the level of nesting of the line. Each nesting level is identified by period. Nesting is used to specify
      - i) fields of a structured attribute (4.1, 4.2, 4.3),
      - ii) attributes conditional on a constraint statement (5). Attributes may be mandatory (5.1) or optional (5.2) if the constraint is true. Not all optional attributes require constraint statements as does the attribute defined in (5.2).
      - iii) the selection fields of a choice type attribute (6.1 and 6.2).
  - (6) The "SERVICES" label indicates that the following entries are services defined for the class.
    - a) An (m) in column 2 indicates that the service is mandatory for the class, while an (o) indicates that it is optional. A (c) in this column indicates that the service is conditional. When all services defined for a class are defined as optional, at least one has to be selected when an instance of the class is defined.
    - b) The label "OpsService" designates an operational service (1).