

DIN EN 61158-5-8

ICS 35.100.70; 35.160

Ersatzvermerk
siehe unten

**Industrielle Kommunikationsnetze –
Feldbusse –
Teil 5-8: Dienstfestlegungen des Application Layer
(Anwendungsschicht) – Typ 8-Elemente (IEC 61158-5-8:2007);
Englische Fassung EN 61158-5-8:2007, nur auf CD-ROM**

Industrial communication networks –
Fieldbus specifications –
Part 5-8: Application layer service definition – Type 8 elements (IEC 61158-5-8:2007);
English version EN 61158-5-8:2007, only on CD-ROM

Réseaux de communication industriels –
Spécifications des bus de terrain –
Partie 5-8: Définition des services des couches d'application –
Éléments de type 8 (CEI 61158-5-8:2007);
Version anglaise EN 61158-5-8:2007, seulement en CD-ROM

Ersatzvermerk

Mit DIN EN 61158-5-2:2008-09, DIN EN 61158-5-3:2008-09, DIN EN 61158-5-4:2008-09,
DIN EN 61158-5-5:2008-09, DIN EN 61158-5-7:2008-09, DIN EN 61158-5-9:2008-09,
DIN EN 61158-5-10:2008-09, DIN EN 61158-5-11:2008-09, DIN EN 61158-5-12:2008-09,
DIN EN 61158-5-13:2008-09, DIN EN 61158-5-14:2008-09, DIN EN 61158-5-15:2008-09,
DIN EN 61158-5-16:2008-09, DIN EN 61158-5-17:2008-09, DIN EN 61158-5-18:2008-09,
DIN EN 61158-5-19:2008-09 und DIN EN 61158-5-20:2008-09 Ersatz für DIN EN 61158-5:2005-05
Siehe jedoch Beginn der Gültigkeit

Gesamtumfang 97 Seiten

DKE Deutsche Kommission Elektrotechnik Elektronik Informationstechnik im DIN und VDE

Beginn der Gültigkeit

Die von CENELEC am 2008-02-01 angenommene EN 61158-5-8 gilt als DIN-Norm ab 2008-09-01.

Daneben darf DIN EN 61158-5:2005-05 noch bis 2011-02-01 angewendet werden.

Nationales Vorwort

Mit DIN EN 61158-5-2:2008-09, DIN EN 61158-5-3:2008-09, DIN EN 61158-5-4:2008-09, DIN EN 61158-5-5:2008-09, DIN EN 61158-5-7:2008-09, DIN EN 61158-5-9:2008-09, DIN EN 61158-5-10:2008-09, DIN EN 61158-5-11:2008-09, DIN EN 61158-5-12:2008-09, DIN EN 61158-5-13:2008-09, DIN EN 61158-5-14:2008-09, DIN EN 61158-5-15:2008-09, DIN EN 61158-5-16:2008-09, DIN EN 61158-5-17:2008-09, DIN EN 61158-5-18:2008-09, DIN EN 61158-5-19:2008-09 und DIN EN 61158-5-20:2008-09 Ersatz für DIN EN 61158-5:2005-05.

Vorausgegangener Norm-Entwurf: E DIN EN 61158-500:2006-11.

Für diese Norm ist das nationale Arbeitsgremium K 956 „Feldbus“ der DKE Deutsche Kommission Elektrotechnik Elektronik Informationstechnik im 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 (maintenance result 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 Europäische Norm EN 60068 durch CENELEC übernommen und als DIN EN 60068 ins Deutsche Normenwerk aufgenommen.

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 von einem Feldbus-Application Layer des Typs 8 (INTERBUS) zur Verfügung gestellt werden.

Änderungen

Gegenüber DIN EN 61158-5:2005-05 wurden folgende Änderungen vorgenommen:

- a) Streichung des Typ 6-Feldbusses wegen zu geringer Marktbedeutung;
- b) Hinzufügung von weiteren Feldbustypen;
- c) Aufteilung in mehrere Teile, jeweils als 5-2, 5-3, ..., 5-20 benummert.

Frühere Ausgaben

DIN V 19245-1:1988-01
DIN 19245-1:1991-04
DIN 19245-2:1991-04
DIN EN 50170-1:1997-07
DIN EN 50170-2:1997-07
DIN EN 50170-3:1997-07
DIN EN 50254:1999-07
DIN EN 50170/A1:2002-11
DIN EN 50170/A2:2000-08
DIN EN 50170/A3:2002-11
DIN EN 61158-5:2005-05

— Leerseite —

Deutsche Fassung

Industrielle Kommunikationsnetze –
Feldbusse –
Teil 5-8: Dienstfestlegungen des Application Layer (Anwendungsschicht) –
Typ 8-Elemente
(IEC 61158-5-8:2007)

Industrial communication networks –
Fieldbus specifications –
Part 5-8: Application layer service definition –
Type 8 elements
(IEC 61158-5-8:2007)

Réseaux de communication industriels –
Spécifications des bus de terrain –
Partie 5-8: Définition des services des couches
d'application – Eléments de type 8
(CEI 61158-5-8:2007)

Diese Europäische Norm wurde von CENELEC am 2008-02-01 angenommen. Die CENELEC-Mitglieder sind gehalten, die CEN/CENELEC-Geschäftsordnung zu erfüllen, in der die Bedingungen festgelegt sind, unter denen dieser Europäischen Norm ohne jede Änderung der Status einer nationalen Norm zu geben ist.

Auf dem letzten Stand befindliche Listen dieser nationalen Normen mit ihren bibliographischen Angaben sind beim Zentralsekretariat oder bei jedem CENELEC-Mitglied auf Anfrage erhältlich.

Diese Europäische Norm besteht in drei offiziellen Fassungen (Deutsch, Englisch, Französisch). Eine Fassung in einer anderen Sprache, die von einem CENELEC-Mitglied in eigener Verantwortung durch Übersetzung in seine Landessprache gemacht und dem Zentralsekretariat mitgeteilt worden ist, hat den gleichen Status wie die offiziellen Fassungen.

CENELEC-Mitglieder sind die nationalen elektrotechnischen Komitees von Belgien, Bulgarien, Dänemark, Deutschland, Estland, Finnland, Frankreich, Griechenland, Irland, Island, Italien, Lettland, Litauen, Luxemburg, Malta, den Niederlanden, Norwegen, Österreich, Polen, Portugal, Rumänien, Schweden, der Schweiz, der Slowakei, Slowenien, Spanien, der Tschechischen Republik, Ungarn, dem Vereinigten Königreich und Zypern.

CENELEC

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

Zentralsekretariat: rue de Stassart 35, B-1050 Brüssel

© 2007 CENELEC – Alle Rechte der Verwertung, gleich in welcher Form und in welchem Verfahren, sind weltweit den Mitgliedern von CENELEC vorbehalten.

Ref. Nr. EN 61158-5-8:2008 D

Vorwort

Der Text des Schriftstücks 65C/475/FDIS, zukünftige 1. Ausgabe von IEC 61158-5-8, ausgearbeitet von dem SC 65C „Industrial networks“ des IEC/TC 65 „Industrial-process measurement, control and automation“, wurde der IEC-CENELEC Parallelen Abstimmung unterworfen und von CENELEC am 2008-02-01 als EN 61158-5-8 angenommen. Dieser und die weiteren Teile der Normenreihe EN 61158-5 ersetzen EN 61158-5:2004.

Gegenüber EN 61158-5:2004 wurden die folgenden Änderungen durchgeführt:

- Streichung des Typ 6-Feldbusses wegen zu geringer Marktbedeutung;
- Hinzufügung von weiteren Feldbustypen;
- Aufteilung in mehrere Teile, jeweils als 5-2, 5-3, ..., 5-20 benummert.

Nachstehende Daten wurden festgelegt:

- spätestes Datum, zu dem die EN auf nationaler Ebene durch Veröffentlichung einer identischen nationalen Norm oder durch Anerkennung übernommen werden muss (dop): 2008-11-01
- spätestes Datum, zu dem nationale Normen, die der EN entgegenstehen, zurückgezogen werden müssen (dow): 2011-02-01

ANMERKUNG Die Anwendung einiger der aufgeführten Protokolltypen wird durch die Inhaber der Rechte an geistigem Eigentum hieran eingeschränkt. Die Zusage zur begrenzten Freigabe ihrer Rechte an geistigem Eigentum durch die Inhaber erlaubt in jedem Fall die Anwendung eines bestimmten Data Link Layer-Protokolls zusammen mit einem Physical Layer- und einem Application Layer-Protokoll in Typzusammenstellungen, die in der Normenreihe EN 61784 angegeben sind. Die Anwendung der verschiedenen Protokolltypen in anderen Zusammenstellungen erfordert möglicherweise die Erlaubnis des jeweiligen Inhabers der Rechte an dem entsprechenden geistigen Eigentum.

Der Anhang ZA wurde von CENELEC hinzugefügt.

Anerkennungsnotiz

Der Text der Internationalen Norm IEC 61158-5-8:2007 wurde von CENELEC ohne irgendeine Abänderung als Europäische Norm angenommen.

In der offiziellen Fassung sind unter „Literaturhinweise“ zu den aufgelisteten Normen die nachstehenden Anmerkungen einzutragen:

- IEC 61158-2 ANMERKUNG Harmonisiert als EN 61158-2:2008 (nicht modifiziert).
- IEC 61158-3-8 ANMERKUNG Harmonisiert als EN 61158-3-8:2008 (nicht modifiziert).
- IEC 61158-4-8 ANMERKUNG Harmonisiert als EN 61158-4-8:2008 (nicht modifiziert).
- IEC 61158-6-8 ANMERKUNG Harmonisiert als EN 61158-6-8:2008 (nicht modifiziert).
- IEC 61784-1 ANMERKUNG Harmonisiert als EN 61784-1:2008 (nicht modifiziert).
- IEC 61784-2 ANMERKUNG Harmonisiert als EN 61784-2:2008 (nicht modifiziert).

CONTENTS

INTRODUCTION.....	5
1 Scope	6
1.1 Overview	6
1.2 Specifications	7
1.3 Conformance	7
2 Normative references	7
3 Terms and definitions	8
3.1 ISO/IEC 7498-1 terms.....	8
3.2 ISO/IEC 8822 terms.....	8
3.3 ISO/IEC 9545 terms.....	8
3.4 ISO/IEC 8824 terms.....	8
3.5 Fieldbus data-link layer terms	8
3.6 Fieldbus application layer specific definitions	9
3.7 Abbreviations and symbols	15
3.8 Conventions	15
4 Concepts.....	19
4.1 Overview	19
4.2 Architectural relationships.....	19
4.3 Fieldbus Application Layer structure.....	21
4.4 FAL naming and addressing.....	31
4.5 Architecture summary	32
4.6 FAL service procedures	32
4.7 Notional FAL service procedures [INFORMATIVE].....	32
4.8 Common FAL attributes	33
4.9 Common FAL service parameters	34
4.10 APDU size.....	34
5 Data type ASE.....	35
5.1 General.....	35
5.2 Formal definition of data type objects.....	37
5.3 FAL defined data types	37
5.4 Data type ASE service specification	41
6 Communication model specification	41
6.1 ASEs.....	41
6.2 ARs.....	84
6.3 Summary of FAL classes	90
6.4 Permitted FAL services by AREP role	90
Bibliography	92
Annex ZA (normative) Normative references to international publications with their corresponding European publications.....	93
Figure 1 – Relationship to the OSI basic reference model.....	19
Figure 2 – Architectural positioning of the Type 8 FAL	20
Figure 3 – Client/server interactions.....	22
Figure 4 – Push model interactions	23

Figure 5 – APOs services conveyed by the FAL	24
Figure 6 – Application entity structure	26
Figure 7 – Example FAL ASEs	27
Figure 8 – FAL management of objects	28
Figure 9 – ASE service conveyance	29
Figure 10 – Defined and established AREPs	31
Figure 11 – FAL architectural components	32
Figure 12 – Data type class hierarchy example.....	35
Figure 13 – The AR ASE conveys APDUs between APs	57
Figure 14 – 1-to-1 AR establishment	64
Table 1 – Get attributes service parameters	42
Table 2 – Identify	49
Table 3 – Get status	50
Table 4 – Initiate.....	52
Table 5 – Terminate	54
Table 6 – Reject	56
Table 7 – Conveyance of service primitives by AREP role	58
Table 8 – Valid combinations of AREP roles involved in an AR	58
Table 9 – AR-unconfirmed send	62
Table 10 – AR-establish service.....	63
Table 11 – Valid combinations of AREP classes to be related	64
Table 12 – AR-abort	65
Table 13 – AR-Data-Send-Acknowledge service parameters.....	66
Table 14 – Read service parameters	71
Table 15 – Write service parameters	73
Table 16 – Information report service	74
Table 17 – Start service parameters.....	78
Table 18 – Stop service parameters	79
Table 19 – Resume service parameters	80
Table 20 – Reset service parameters	81
Table 21 – State transitions for a function invocation object.....	83
Table 22 – FAL class summary	90
Table 23 – Services by AREP role	91

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 5-8: Application layer service definition – Type 8 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 standard 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 8 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 standard defines in an abstract way the externally visible service provided by the Type 8 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 standard 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 standard specifies the structure and services of the Type 8 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 standard 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 subparts of 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 standard does not specify individual implementations or products, nor does it 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 fulfill the Type 8 application layer services as defined in this standard.

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*

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

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

3 Terms and definitions

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

3.1 ISO/IEC 7498-1 terms

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

- a) application entity
- b) application process
- c) application protocol data unit
- d) application process invocation
- e) application transaction
- f) real open system
- g) 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 61158-3-8 and IEC 61158-4-8, apply.

- a) DLCEP
- b) DLSAP
- c) link

d) node

3.6 Fieldbus application layer specific definitions

3.6.1

access protection

limitation of the usage of an application object to one client

3.6.2

allocate

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

3.6.3

application

function or data structure for which data is consumed or produced

3.6.4

application layer interoperability

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

3.6.5

application objects

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

3.6.6

application process

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

3.6.7

application process identifier

distinguishes multiple application processes used in a device

3.6.8

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.

3.6.9

application process object class

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

3.6.10

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

3.6.11

application relationship application service element

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

3.6.12

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.

3.6.13

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.

3.6.14

behaviour

indication of how an object responds to particular events

3.6.15

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.

3.6.16

class attribute

attribute that is shared by all objects within the same class

3.6.17

class code

unique identifier assigned to each object class

3.6.18

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.

3.6.19

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

3.6.20

control commands

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

3.6.21

conveyance path

unidirectional flow of APDUs across an application relationship

3.6.22

cyclic

repetitive in a regular manner

3.6.23

data consistency

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

3.6.24

dedicated AR

AR used directly by the FAL User

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

3.6.25

device

physical hardware connected to the link

NOTE A device may contain more than one node.

3.6.26

dynamic AR

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

3.6.27

end node

producing or consuming node

3.6.28

endpoint

one of the communicating entities involved in a connection

3.6.29

engineering

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

3.6.30

error

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

3.6.31

error class

general grouping for related error definitions and corresponding error codes

3.6.32

error code

identification of a specific type of error within an error class

3.6.33

event

an instance of a change of conditions

3.6.34

group

(a) collection of objects

(b) address that identifies more than one entity

3.6.35
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

3.6.36
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.

3.6.37
index

address of an object within an application process

3.6.38
instance

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

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

3.6.39
instance attributes

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

3.6.40
instantiated

object that has been created in a device

3.6.41
manufacturer ID

identification of each product manufacturer by a unique number

3.6.42
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.

3.6.43
member

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

3.6.44
method

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

3.6.45
network

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

3.6.46

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

3.6.47

object specific service

service unique to the object class which defines it

3.6.48

peer

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

3.6.49

point-to-point connection

connection that exists between exactly two application objects

3.6.50

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

3.6.51

pre-established AR endpoint

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

3.6.52

property

descriptive information about an object

3.6.53

provider

source of a data connection

3.6.54

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.

3.6.55

push publisher

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

3.6.56

push subscriber

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

3.6.57

resource

a processing or information capability of a subsystem

3.6.58

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

3.6.59

service

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

3.6.60

subscriber

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

3.7 Abbreviations and symbols

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
ASE	Application service element
CIM	Computer integrated manufacturing
Cnf	Confirmation
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
FAL	Fieldbus application layer
FIFO	First in first out
ID	Identifier
Ind	Indication
LME	Layer management entity
OSI	Open Systems Interconnect
PDU	Protocol data unit
Req	Request
Rsp	Response
SAP	Service access point

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:

FAL ASE:	ASE Name
CLASS:	Class Name
CLASS ID:	#
PARENT CLASS:	Parent Class Name
ATTRIBUTES:	
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).
 - c) The label "MgtService" designates an management service (2).
 - d) The line number defines the sequence and the level of nesting of the line. Each nesting level is identified by period. Nesting within the list of services is used to specify services conditional on a constraint statement.

3.8.4 Conventions for service definitions

3.8.4.1 General

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

3.8.4.2 Service parameters

Service primitives are used to represent service user/service provider interactions (ISO/IEC 10731). They convey parameters which indicate information available in the user/provider interaction. In any particular interface, not all parameters need be explicitly stated.

The service specifications of this standard uses a tabular format to describe the component parameters of the ASE service primitives. The parameters which apply to each group of service primitives are set out in tables. Each table consists of up to five columns for the

- 1) parameter name,
- 2) request primitive,
- 3) indication primitive,
- 4) response primitive, and
- 5) confirm primitive.

One parameter (or component 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 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 dynamic usage of the service 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 service user.
- (blank) parameter is never present.
- S parameter is a selected item.

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.
- 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.

3.8.4.3 Service procedures

The procedures are defined in terms of

- the interactions between application entities through the exchange of fieldbus Application Protocol Data Units, and
- the interactions between an application layer service provider and an application layer service user in the same system through the invocation of application layer service primitives.

These procedures are applicable to instances of communication between systems which support time-constrained communications services within the fieldbus Application Layer.

4 Concepts

4.1 Overview

This subclause describes fundamentals of the FAL. Detailed descriptive information about each of the FAL ASEs can be found in the “Overview” subclause of each of the Communication Model specifications.

4.2 Architectural relationships

4.2.1 Relationship to the Application Layer of the OSI basic reference model

The functions of the FAL have been described according to OSI layering principles. However, its architectural relationship to the lower layers is different, as shown in Figure 1.

- The FAL includes OSI functions together with extensions to cover time-critical requirements. The OSI Application Layer Structure standard (ISO/IEC 9545) was used as a basis for specifying the FAL.
- The FAL directly uses the services of the underlying layer. The underlying layer may be the data link layer or any layer in between. When using the underlying layer, the FAL may provide functions normally associated with the OSI Middle Layers for proper mapping onto the underlying layer.

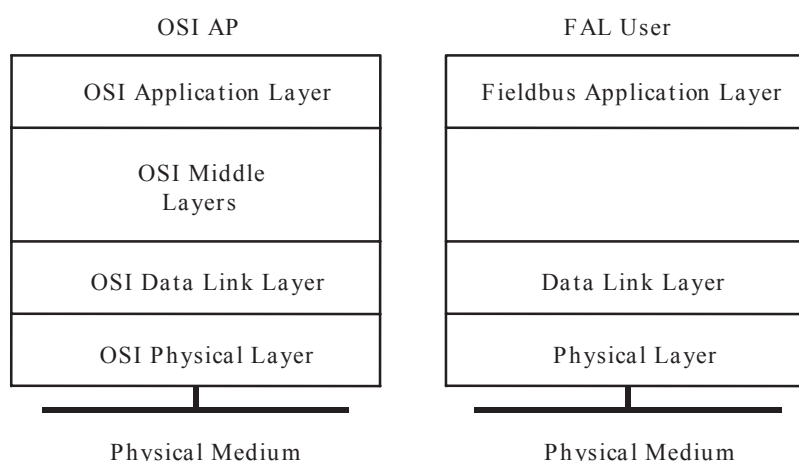


Figure 1 – Relationship to the OSI basic reference model

4.2.2 Relationships to other fieldbus entities

4.2.2.1 General

The Type 8 Fieldbus Application Layer (FAL) architectural relationships, as illustrated in Figure 2, have been designed to support the interoperability needs of time-critical systems distributed within the fieldbus environment.

Within this environment, the FAL provides communications services to time-critical and non-time-critical applications located in Type 8 devices.

In addition, the FAL directly uses the data-link layer to transfer its application layer protocol data units. It does this using a set of data transfer services and a set of supporting services used to control the operational aspects of the data-link layer.

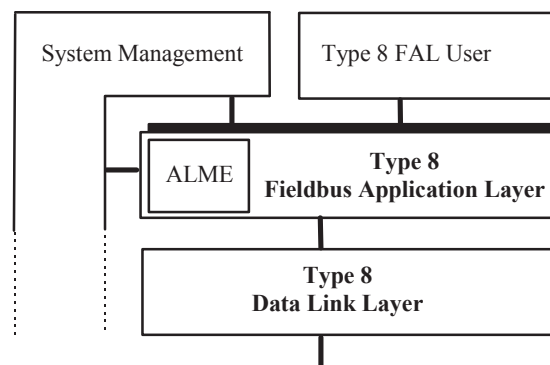


Figure 2 – Architectural positioning of the Type 8 FAL

4.2.3 Mapping of functional capabilities

Two communication mechanisms are supported:

- cyclic transmission of data in a high efficient manner using a publisher subscriber model,
- non cyclic transmission of data using a client server communication model

AREPs, which act as push publisher or push subscriber, are used for cyclic transmission.

AREPs, which act as client and server (peer), are used for non cyclic data transmission. A specific transfer syntax is used for their APDUs. The non cyclic data transmission mechanism is described by a subset of those ASEs and services specified in 6.1.

4.2.3.1 Use of the Type 8 data-link layer

The fieldbus Application Layer (FAL) provides network access to fieldbus APs. It interfaces directly to the fieldbus data-link layer for transfer of its APDUs.

The data-link layer provides various types of services to the FAL for the transfer of data between Data-link endpoints (e.g. DLSAPs, DLCEPs).

4.2.3.2 Support for Type 8 applications

TYPE 8 applications are represented to the network as application processes (APs). APs are the components of a distributed system that may be individually identified and addressed.

Each AP contains an FAL application entity (AE) that provides network access for the AP. That is, each AP communicates with other APs through its AE. In this sense, the AE provides a window of visibility into the AP.

APs contain identifiable components that are also visible across the network. These components are represented to the network as Application Process Objects (APO). They may be identified by one or more key attributes. They are located at the address of the application process that contains them.

The services used to access them are provided by APO-specific application service elements (ASEs) contained within the FAL. These ASEs are designed to support user, function block, and management applications.

4.2.3.3 Support for system management

The FAL services can be used to support various management operations, including management of fieldbus systems, applications, and the fieldbus network.

4.2.3.4 Access to FAL layer management entities

One layer management entity (LME) may be present in each FAL entity on the network. FALMEs provide access to the FAL for system management purposes.

The set of data accessible by the System Manager is referred to as the System Management Information Base (SMIB). Each fieldbus Application Layer Management Entity (FALME) provides the FAL portion of the SMIB. How the SMIB is implemented is beyond the scope of this standard.

4.3 Fieldbus Application Layer structure

4.3.1 Overview

The structure of the FAL is a refinement of the OSI Application Layer Structure (ISO/IEC 9545). As a result, the organization of this subclause is similar to that of ISO/IEC 9545. Certain concepts presented here have been refined from ISO/IEC 9545 for the TYPE 8 environment.

The FAL differs from the other layers of OSI in two principal aspects:

- OSI defines a single type of application layer communications channel, the association, to connect APs to each other. The FAL defines the Application Relationship (AR), of which there are several types, to permit application processes (APs) to communicate with each other.
- The FAL uses the DLL to transfer its PDUs and not the presentation layer. Therefore, there is no explicit presentation context available to the FAL. Between the same pair (or set) of data link service access points the FAL protocol may not be used concurrently with other application layer protocols.

4.3.2 Fundamental concepts

The operation of time critical real open systems is modeled in terms of interactions between time-critical APs. The FAL permits these APs to pass commands and data between them.

Cooperation between APs requires that they share sufficient information to interact and carry out processing activities in a coordinated manner. Their activities may be restricted to a single fieldbus segment, or they may span multiple segments. The FAL has been designed using a modular architecture to support the messaging requirements of these applications.

Cooperation between APs also sometimes requires that they share a common sense of time. The FAL or the data-link layer (part 3 and part 4) may provide for the distribution of time to all devices. They also may define local device services that can be used by APs to access the distributed time.

The remainder of this subclause describes each of the modular components of the architecture and their relationships with each other. The components of the FAL are modeled as objects, each of which provides a set of FAL communication services for use by applications. The FAL objects and their relationships are described below. The detailed specifications of FAL objects and their services are provided in the following clauses of this standard. Part 6 specifies the protocols necessary to convey these object services between applications.

4.3.3 Application Processes

4.3.3.1 Definition of the AP

In the TYPE 8 environment, an application may be partitioned into a set of components and distributed across a number of devices on the network. Each of these components is referred to as a fieldbus Application Process (AP). A fieldbus AP is a variation of an Application Process as defined in ISO OSI Reference Model (ISO/IEC 7498). Fieldbus APs may be unambiguously addressed by at least one individual data-link layer service access point address. Unambiguously addressed, in this context, means that no other AP may simultaneously be located by the same address. This definition

does not prohibit an AP from being located by more than one individual or group data link service access point address.

4.3.3.2 Communication services

APs communicate with each other using confirmed and unconfirmed services (ISO/IEC 10731). The services defined in this standard for the FAL specify the semantics of the services as seen by the requesting and responding APs. The syntax of the messages used to convey the service requests and responses is defined in part 6. The AP behavior associated with the services is specified by the AP.

Confirmed services are used to define request/response exchanges between APs.

Unconfirmed services, in contrast, are used to define the unidirectional transfer of messages from one AP to one or more remote APs. From a communications perspective, there is no relationship between separate invocations of unconfirmed services as there is between the request and response of a confirmed service.

4.3.3.3 AP interactions

4.3.3.3.1 General

APs may interact with other APs as necessary to achieve their functional objectives. No constraints are imposed by this standard on the organization of these interactions or the possible relationships that may exist between them.

For example, interactions may be based on request/response messages sent directly between APs, or on data/events sent by one AP for use by others. These two models of interactions between APs are referred to as client/server and publisher/subscriber interactions.

The services supported by an interaction model are conveyed by application relationship endpoints (AREPs) associated with the communicating APs. The role that the AREP plays in the interaction (e.g. client, server, peer, publisher, subscriber) is defined as an attribute of the AREP.

4.3.3.3.2 Client/server interactions

Client/server interactions are characterized by a bi-directional data flow between a client AP and one or more server APs. Figure 3 illustrates the interaction between a single client and a single server. In this type of interaction, the client may issue a confirmed or unconfirmed request to the server to perform some task. If the service is confirmed then the server will always return a response. If the service is unconfirmed, the server may return a response using an unconfirmed service defined for this purpose.

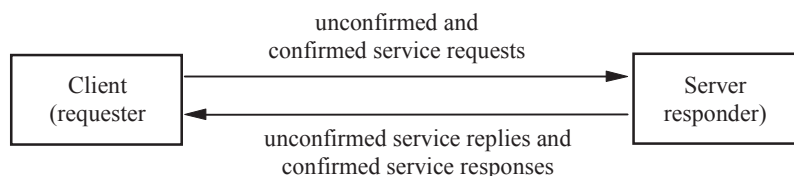


Figure 3 – Client/server interactions

4.3.3.3.3 Publisher/subscriber interactions

4.3.3.3.3.1 General

Publisher/subscriber interactions, on the other hand, involve a single publisher AP, and one subscriber APs. This type of interaction has been defined to support one model of interaction

between APs, the "push" model. In the model, the setup of the publishing AP is performed by management and is outside the scope of this standard.

4.3.3.3.2 Push model interactions

In the "push" model, two services may be used, one confirmed and one unconfirmed. The confirmed service is used by the subscriber to request to join the publishing. The response to this request is returned to the subscriber, following the client/server model of interaction. This exchange is only necessary when the subscriber and the publisher are located in different APs.

The unconfirmed service used in the Push Model is used by the publisher to distribute its information to subscribers. In this case, the publisher is responsible for invoking the correct unconfirmed service at the appropriate time and for supplying the appropriate information. In this fashion, it is configured to "push" its data onto the network.

Subscribers for the Push Model receive the published unconfirmed services distributed by publishers. Figure 4 illustrates the concept of the Push Model.

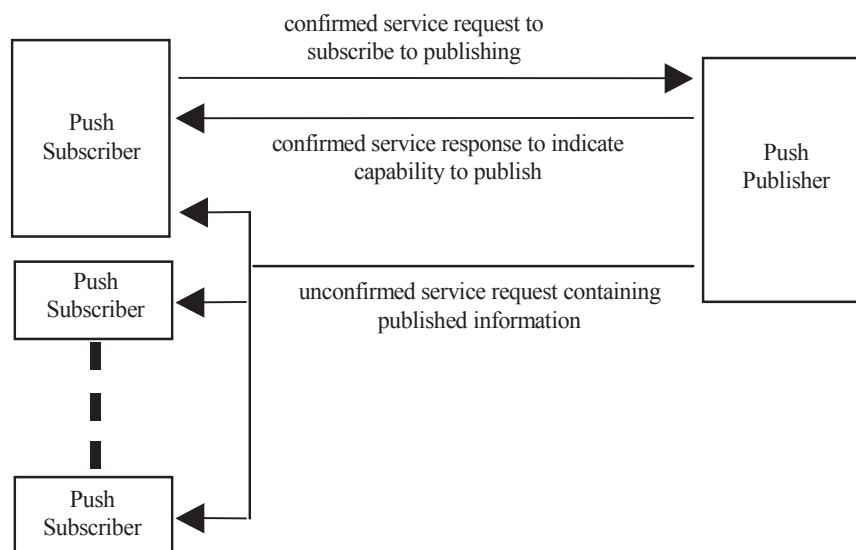


Figure 4 – Push model interactions

4.3.3.4 AP structure

The internals of APs may be represented by one, or more Application Process Objects (APOs) and accessed through one or more Application Entities (AEs). AEs provide the communication capabilities of the AP. For each AP, there is one and only one FAL AE. APOs are the network representation of application specific capabilities (user application process objects) of an AP that are accessible through its FAL AE.

4.3.3.5 AP class

An AP class is a definition of the attributes and services of an AP. The standard class definition for APs is defined in this standard. User defined classes also may be specified. Class identifiers (described in 3.8.2) are assigned from a set reserved for this purpose.

4.3.3.6 AP type

As described above in the previous subclauses, APs are defined by instantiating an AP class. Each AP definition is composed of the attributes and services selected for the AP from those defined by

its AP class. In addition, an AP definition contains values for one or more of the attributes selected for it. When two APs share the same definition, that definition is referred to as an AP type. Thus, an AP type is a generic specification of an AP that may be used to define one or more APs.

4.3.4 Application process objects

4.3.4.1 Definition of APO

An application process object (APO) is a network representation of a specific aspect of an AP. Each APO represents a specific set of information and processing capabilities of an AP that are accessible through services of the FAL. APOs are used to represent these capabilities to other APs in a fieldbus system.

From the perspective of the FAL, an APO is modeled as a network accessible object contained within an AP or within another APO (APOs may contain other APOs). APOs provide the network definition for objects contained within an AP that are remotely accessible. The definition of an APO includes an identification of the FAL services that can be used by remote APs for remote access. The FAL services, as shown in Figure 5, are provided by the FAL communications entity of the AP, known as the FAL Applications Entity (FAL AE).

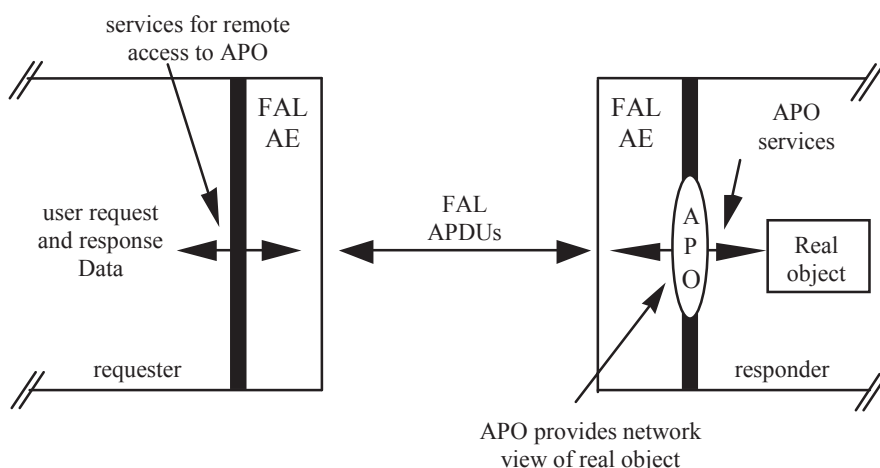


Figure 5 – APOs services conveyed by the FAL

In Figure 5, remote APs acting as clients may access the real object by sending requests through the APO that represents the real object. Local aspects of the AP convert between the network view (the APO) of the real object and the internal AP view of the real object.

To support the publisher/subscriber model of interaction, information about the real object can be published through its APO. Remote APs acting as subscribers see the APO view of the published information instead of having to know any of the real object specific details.

4.3.4.2 APO classes

An APO Class is a generic specification for a set of APOs, each of which is described by the same set of attributes and accessed using the same set of services.

APO Classes provide the mechanism for standardizing network visible aspects of APs. Each standard APO Class definition specifies a particular set of network accessible AP attributes and services. IEC 61158-6-8 specifies the syntax and the procedures used by the FAL protocol to provide remote access to the attributes and services of an APO Class.

Standard APO Classes are specified by this standard for the purpose of standardizing remote access to APs. User defined classes may also be specified.

User defined classes are defined as subclasses of standardized APO Classes or of other user-defined classes. They may be defined by identifying new attributes or by indicating that optional attributes for the parent class are mandatory for the subclass. The conventions for defining classes defined in 3.8.2 may be used for this purpose. The method for registering or otherwise making these new class definitions available for public use is beyond the scope of this standard.

4.3.4.3 APOs as instances of APO classes

APO Classes are defined in this standard using templates. These templates are used not only to define APO Classes, but also to specify the instances of a class.

Each APO defined for an AP is an instance of an APO Class. Each APO provides the network view of a real object contained in an AP. An APO is defined by

- (1) selecting the attributes from its APO Class template that are to be accessible from the real object;
- (2) assigning values to one or more attributes indicated as key in the template. Key attributes are used to identify the APO;
- (3) assigning values to zero, one, or more non-key attributes for the APO. Non-key attributes are used to characterize the APO;
- (4) selecting the services from the template that may be used by remote APs to access the real object.

Subclause 3.8.2 of this standard specifies the conventions for class templates. These conventions provide for the definition of mandatory, optional, and conditional attributes and services.

Mandatory attributes and services are required to be present in all APOs of the class. Optional attributes and services may be selected, on an APO by APO basis, for inclusion in an APO. Conditional attributes and services are defined with an accompanying constraint statement. Constraint statements specify the conditions that indicate whether or not the attribute is to be present in an APO.

4.3.4.4 APO types

APO types provide the mechanism for defining standard APOs.

As described above in the previous subclauses, APOs are defined by instantiating an APO class. Each APO definition is composed of the attributes and services selected for the APO from those defined by its APO class. In addition, an APO definition contain values for one or more of the attributes selected for it. When two APOs share the same definition, except for the key attribute settings, that definition is referred to as an APO type. Thus, an APO type is a generic specification of an APO that may used to define one or more APOs.

4.3.5 Application entities

4.3.5.1 Definition of FAL AE

An application entity provides the communication capabilities for a single AP. An FAL AE provides a set of services and the supporting protocols to enable communications between APs in a fieldbus environment. The services provided by FAL AEs are grouped into Application Service Elements (ASE), such that the FAL services provided to an AP are defined by the ASEs its FAL AE contains. Figure 6 illustrates this concept.

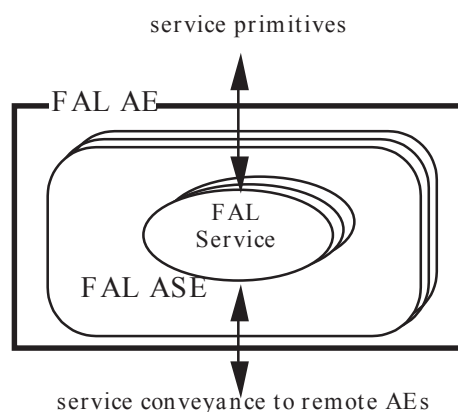


Figure 6 – Application entity structure

4.3.5.2 AE type

Application entities that provide the same set of ASEs are of the same AE-type. Two AEs that share a common set of ASEs are capable of communicating with each other.

4.3.6 Application service elements

4.3.6.1 General

An application service element (ASE), as defined in ISO/IEC 9545, is a set of application functions that provide a capability for the interworking of application-entity-invocations for a specific purpose. ASEs provide a set of services for conveying requests and responses to and from application processes and their objects. AEs, as defined above, are represented by a collection of ASE invocations within the AE.

4.3.6.2 FAL services

FAL Services convey functional requests/responses between APs. Each FAL service is defined to convey requests and responses for access to a real object modeled as an FAL accessible object.

The FAL defines both confirmed and unconfirmed services. Confirmed service requests are sent to the AP containing the real object. An invocation of a confirmed service request may be identified by a user supplied Invoke ID. This Invoke ID is returned in the response by the AP containing the real object. When present, it may be used by the user to associate the response with the appropriate request.

NOTE This user-specified Invoke ID is distinct from any Instance ID that the requesting AP and its FAL AE may itself use to associate request with confirm, indication with response, or command APDU with reply APDU. See 1.2.

Unconfirmed services may be sent from the AP containing the real object to send information about the object. They also may be sent to the AP containing the real object to access the real object. Both types of unconfirmed services may be defined for the FAL.

4.3.6.3 Definition of FAL ASEs

4.3.6.3.1 General

A modular approach has been taken in the definition of FAL ASEs. The ASEs defined for the FAL are also object-oriented. In general, ASEs provide a set of services designed for one specific object class or for a related set of classes. Common object management ASEs, when present, provide a common set of management services applicable to all classes of objects.