



BSI Standards Publication

Industrial communication networks — Fieldbus specifications

Part 5-26: Application layer service definition — Type 26 elements

National foreword

This British Standard is the UK implementation of EN IEC 61158-5-26:2023. It is identical to IEC 61158-5-26:2023. It supersedes BS EN IEC 61158-5-26:2019, which is withdrawn.

The UK participation in its preparation was entrusted to Technical Committee GEL/65, Measurement and control.

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

Contractual and legal considerations

This publication has been prepared in good faith, however no representation, warranty, assurance or undertaking (express or implied) is or will be made, and no responsibility or liability is or will be accepted by BSI in relation to the adequacy, accuracy, completeness or reasonableness of this publication. All and any such responsibility and liability is expressly disclaimed to the full extent permitted by the law.

This publication is provided as is, and is to be used at the recipient's own risk.

The recipient is advised to consider seeking professional guidance with respect to its use of this publication.

This publication is not intended to constitute a contract. Users are responsible for its correct application.

© The British Standards Institution 2023
Published by BSI Standards Limited 2023

ISBN 978 0 539 13668 5

ICS 25.040.40; 35.100.70; 35.110

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 May 2023.

Amendments/corrigenda issued since publication

Date	Text affected
------	---------------

English Version

Industrial communication networks - Fieldbus specifications -
Part 5-26: Application layer service definition - Type 26 elements
(IEC 61158-5-26:2023)

Réseaux de communication industriels - Spécifications des
bus de terrain - Partie 5-26: Définition des services de la
couche application - Eléments de type 26
(IEC 61158-5-26:2023)

Industrielle Kommunikationsnetze - Feldbusse - Teil 5-26:
Application layer service definition (Anwendungsschicht) -
Type 26-Elemente
(IEC 61158-5-26:2023)

This European Standard was approved by CENELEC on 2023-04-20. CENELEC members are bound to comply with the CEN/CENELEC Internal Regulations which stipulate the conditions for giving this European Standard the status of a national standard without any alteration.

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

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

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



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

CEN-CENELEC Management Centre: Rue de la Science 23, B-1040 Brussels

European foreword

The text of document 65C/1203/FDIS, future edition 2 of IEC 61158-5-26, prepared by SC 65C "Industrial networks" of IEC/TC 65 "Industrial-process measurement, control and automation" was submitted to the IEC-CENELEC parallel vote and approved by CENELEC as EN IEC 61158-5-26:2023.

The following dates are fixed:

- latest date by which the document has to be implemented at national level by publication of an identical national standard or by endorsement (dop) 2024-01-20
- latest date by which the national standards conflicting with the document have to be withdrawn (dow) 2026-04-20

This document supersedes EN IEC 61158-5-26:2019 and all of its amendments and corrigenda (if any).

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

Any feedback and questions on this document should be directed to the users' national committee. A complete listing of these bodies can be found on the CENELEC website.

Endorsement notice

The text of the International Standard IEC 61158-5-26:2023 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 standard indicated:

IEC 61784-1 (series) NOTE Approved as EN IEC 61784-1 (series)¹

IEC 61784-2 (series) NOTE Approved as EN IEC 61784-2 (series)²

¹ To be published. Stage at time of publication : FprEN 61784-1-X:2023.

² To be published. Stage at time of publication : FprEN 61784-2-X:2023.

Annex ZA (normative)

Normative references to international publications with their corresponding European publications

The following documents are referred to in the text in such a way that some or all of their content constitutes requirements 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 1 Where an International Publication has been modified by common modifications, indicated by (mod), the relevant EN/HD applies.

NOTE 2 Up-to-date information on the latest versions of the European Standards listed in this annex is available here: www.cencenelec.eu.

<u>Publication</u>	<u>Year</u>	<u>Title</u>	<u>EN/HD</u>	<u>Year</u>
IEC 61131-3	-	Programmable controllers - Part 3: Programming languages	EN 61131-3	-
IEC 61158-1	2023	Industrial communication networks - Fieldbus specifications - Part 1: Overview and guidance for the IEC 61158 and IEC 61784 series	EN 61158-1	2023
IEC 61158-6-26	2023	Industrial communication networks - Fieldbus specifications - Part 6-26: Application layer protocol specification - Type 26 elements	EN IEC 61158-6-26 — ³	
IEC 61784-2-21	2023	Industrial networks - Profiles - Part 2-21: Additional real-time fieldbus profiles based on ISO/IEC/IEEE 8802-3 - CPF 21	-	-
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: The basic model	-	-
ISO/IEC/IEEE 8802-3	-	Standard for Ethernet	-	-
ISO/IEC 8822	-	Information technology - Open Systems Interconnection - Presentation service definition	-	-
ISO/IEC 8824-1	-	Information technology - Abstract Syntax Notation One (ASN.1) - Part 1: Specification of basic notation	-	-
ISO/IEC 9545	-	Information technology - Open Systems Interconnection - Application layer structure	-	-

³ To be published. Stage at time of publication: FprEN IEC 61158-6-26:2023.

EN IEC 61158-5-26:2023 (E)

ISO/IEC 10731	-	Information technology - Open Systems Interconnection - Basic Reference Model - Conventions for the definition of OSI services	-	-
IEC 60559	-	Floating-Point arithmetic	HD 592 S1	-
IETF RFC 768	1980	User Datagram Protocol	-	-
IETF RFC 791	1981	Internet Protocol	-	-
IETF RFC 792	1981	Internet Control Message Protocol	-	-
IETF RFC 793	1981	Transmission Control Protocol	-	-
IETF RFC 796	2081	Address mappings	-	-
IETF RFC 826	1982	An Ethernet Address Resolution Protocol: Or Converting Network Protocol Addresses to 48.bit Ethernet Address for Transmission on Ethernet Hardware	-	-
IETF RFC 894	1984	Standard for the Transmission of IP Datagrams over Ethernet	-	-
IETF RFC 919	1984	Broadcasting Internet Datagrams	-	-
IETF RFC 922	1984	Broadcasting Internet datagrams in the presence of subnets	-	-
IETF RFC 950	1985	Internet Standard Subnetting Procedure	-	-

CONTENTS

FOREWORD.....	5
INTRODUCTION.....	7
1 Scope.....	8
1.1 General.....	8
1.2 Specifications	9
1.3 Conformance	9
2 Normative references	9
3 Terms, definitions, symbols, abbreviated terms and conventions	11
3.1 Terms and definitions from other ISO/IEC standards.....	11
3.1.1 Terms and definitions from ISO/IEC 7498-1	11
3.1.2 Terms and definitions from ISO/IEC 8822	11
3.1.3 Terms and definitions from ISO/IEC 9545	11
3.1.4 Terms and definitions from ISO/IEC 8824-1	12
3.2 Fieldbus application layer Type 26-specific definitions	12
3.3 Abbreviated terms and symbols	18
3.4 Conventions.....	20
3.4.1 Overview	20
3.4.2 General conventions	20
3.4.3 Conventions for class definitions	21
3.4.4 Conventions for service definitions	22
4 Concepts	23
5 Data type ASE.....	23
5.1 Overview.....	23
5.2 Formal definition of data type objects.....	24
5.2.1 Data type class definitions	24
5.2.2 Attributes	25
5.3 FAL defined data types	26
5.3.1 Fixed length types	26
5.3.2 String types	32
5.4 Data type ASE service specification.....	33
6 Communication model specification	33
6.1 General.....	33
6.2 Protocol stack for Type 26 fieldbus	33
6.3 Overview of Type 26 communication model	34
6.4 Cyclic data communication service with Common-memory	35
6.4.1 Overview	35
6.4.2 Common-memory: allocation to each node.....	36
6.4.3 Data sharing among nodes with the CM.....	37
6.4.4 CM data type	39
6.5 ASEs	39
6.5.1 Overview of Type 26 ASEs	39
6.5.2 Type 26 specific conventions for FAL service common parameters	40
6.5.3 Cyclic-data ASE.....	41
6.5.4 Message data ASE	47
6.5.5 Load measurement ASE	77
6.5.6 Network management ASE	82

6.5.7	General purpose command server ASE	97
6.5.8	AR ASE	100
6.5.9	FAL ASE summary	113
Bibliography.....		116
Figure 1	– Protocol stack for Type 26 fieldbus	34
Figure 2	– Unconfirmed Push-Publisher/Subscriber type interaction.....	35
Figure 3	– Unconfirmed/Confirmed Client/Server type interaction.....	35
Figure 4	– Common memory allocation	37
Figure 5	– Data sharing with the CM	38
Figure 6	– Node #01 for reception only	38
Figure 7	– Node #01 without the CM	39
Figure 8	– Data sharing among nodes with and without CM3	39
Figure 9	– The structure of ASEs for Type 26 FAL	40
Figure 10	– Virtual-address-space for Byte block	52
Figure 11	– Virtual-address-space for Word block	54
Figure 12	– AR ASE internal architecture	101
Figure 13	– Structure of IP address	111
Table 1	– Write service parameters.....	44
Table 2	– Send-CM service parameters	45
Table 3	– Read service parameters.....	45
Table 4	– Update memory service parameters	46
Table 5	– Get- buffer service parameters	47
Table 6	– Byte block read service parameters.....	52
Table 7	– Byte block write service parameters	53
Table 8	– Word block read service parameters	54
Table 9	– Word block write service parameters	55
Table 10	– Network parameter read service parameters	56
Table 11	– Extended network parameter read service parameters	57
Table 12	– Network parameter write service parameters	58
Table 13	– Extended network parameter write service parameters	59
Table 14	– Stop command service parameters.....	60
Table 15	– Operation command service parameters.....	60
Table 16	– profile read service parameters	61
Table 17	– Transparent message service parameters	63
Table 18	– Log data read service parameters	64
Table 19	– Log data items.....	64
Table 20	– Log data clear service parameters.....	69
Table 21	– Message return service parameters.....	70
Table 22	– Vendor specific message service parameters	71
Table 23	– Set remote node configuration parameter service parameters	72
Table 24	– Data elements and Node configuration parameters.....	72

Table 25 – Service parameters of Read remote participating node management information parameter service.....	73
Table 26 – Participating node management information parameters.....	74
Table 27 – Read remote node management information parameter service parameters	75
Table 28 – Node management information parameters	76
Table 29 – Read remote node setting information parameter service parameters.....	77
Table 30 – Node setting information parameters	77
Table 31 – Start TK-holding-time measurement service parameters.....	79
Table 32 – Terminate TK-holding-time measurement service parameters.....	80
Table 33 – Token-holding-time measurement result	81
Table 34 – Start GP_Comm sender log service parameters	82
Table 35 – Terminate GP_Comm sender log service parameters	82
Table 36 – GP_Comm sender log measurement result.....	83
Table 37 – Service parameters for Set configuration parameter	87
Table 38 – Configuration parameters	87
Table 39 – Read node management information parameter service parameters.....	88
Table 40 – Node management information parameters	88
Table 41 – Service parameters for Read participating node mgt. information parameter	89
Table 42 – Participating node management information parameters.....	90
Table 43 – Service parameters for Read network management information parameter.....	90
Table 44 – Network management information parameters	91
Table 45 – Service parameters for Read message sequence number management information.....	91
Table 46 – Read message sequence number management information parameters	92
Table 47 – Read node status service parameters.....	92
Table 48 – Read node status parameters.....	93
Table 49 – Upper layer operating condition matrix	93
Table 50 – Reset node service parameters	94
Table 51 – Set network address service parameters	94
Table 52 – Register service parameters.....	95
Table 53 – Event service parameters	95
Table 54 – Activate/Deactivate measurement service parameters	96
Table 55 – Get log data service parameters.....	97
Table 56 – Send command service parameters.....	100
Table 57 – CT send service parameters.....	104
Table 58 – MT send service parameters	106
Table 59 – CS send service parameters.....	107
Table 60 – Notify state change service parameters.....	108
Table 61 – Control measurement service parameters.....	109
Table 62 – DLSAP assignments.....	110
Table 63 – DLS Primitives and parameters	112
Table 64 – Lower layer T-profile and the required standards.....	113
Table 65 – Summary of FAL ASEs	113

INTERNATIONAL ELECTROTECHNICAL COMMISSION

**INDUSTRIAL COMMUNICATION NETWORKS –
FIELDBUS SPECIFICATIONS –****Part 5-26: Application layer service definition –
Type 26 elements****FOREWORD**

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

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

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

IEC 61158-5-26 has been prepared by subcommittee 65C: Industrial networks, of IEC technical committee 65: Industrial-process measurement, control and automation. It is an International Standard.

This second edition cancels and replaces the first edition published in 2019. This edition constitutes a technical revision.

This edition includes the following significant technical changes with respect to the previous edition:

- a) expand Common-memory-area as a new Common-memory-area-3 (CM3);
- b) add new services with expansion of Common-memory-area:
 - Extended-cyclic-data transfer service;
 - Extended-participation-request service;
 - Extended-network-parameter-read service;
 - Extended-network-parameter-write service.

The text of this International Standard is based on the following documents:

Draft	Report on voting
65C/1203/FDIS	65C/1244/RVD

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

The language used for the development of this International Standard is English.

This document was drafted in accordance with ISO/IEC Directives, Part 2, and developed in accordance with ISO/IEC Directives, Part 1 and ISO/IEC Directives, IEC Supplement, available at www.iec.ch/members_experts/refdocs. The main document types developed by IEC are described in greater detail at www.iec.ch/publications.

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

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

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

INTRODUCTION

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

The application service is provided by the application protocol making use of the services available from the data-link or other immediately lower layer. This document defines the application service characteristics that fieldbus applications and/or system management can 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 document is a conceptual architectural service, independent of administrative and implementation divisions.

INDUSTRIAL COMMUNICATION NETWORKS – FIELDBUS SPECIFICATIONS –

Part 5-26: Application layer service definition – Type 26 elements

1 Scope

1.1 General

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 26 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 document defines in an abstract way the externally visible service provided by the Type26 fieldbus application layer in terms of:

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

The purpose of this document is to define the services provided to:

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

This document specifies the structure and services of the Type 26 fieldbus application layer, in conformance with the OSI Basic Reference Model (see ISO/IEC 7498-1) and the OSI Application Layer Structure (see 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 document to provide access to the FAL to control certain aspects of its operation.

1.2 Specifications

The principal objective of this document 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 document can 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 document 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 26 application layer services as defined in this document.

2 Normative references

The following documents are referred to in the text in such a way that some or all of their content constitutes requirements 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 All parts of the IEC 61158 series, as well as the IEC 61784-1 series and the IEC 61784-2 series are maintained simultaneously. Cross-references to these documents within the text therefore refer to the editions as dated in this list of normative references.

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

IEC 61158-1:2023, *Industrial communication networks – Fieldbus specifications – Part 1: Overview and guidance for the IEC 61158 and IEC 61784 series*

IEC 61158-6-26:2023, *Industrial communication networks – Fieldbus specifications – Part 6-26: Application layer protocol specification – Type 26 elements*

IEC 61784-2-21:2023, *Industrial networks – Profiles – Part 2-21: Additional real-time fieldbus profiles based on ISO/IEC/IEEE 8802-3 – CPF 21*

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/IEEE 8802-3, *Telecommunications and exchange between information technology systems – Requirements for local and metropolitan area networks – Part 3: Standard for Ethernet*

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

ISO/IEC 8824-1, *Information Technology – Abstract Syntax Notation One (ASN-1) – Part 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*

ISO/IEC 60559, *Floating-Point arithmetic*

IETF RFC 768, J. Postel, *User Datagram Protocol*, August 1980, available at <https://www.rfc-editor.org/info/rfc768> [viewed 2022-02-18]

IETF RFC 791, J. Postel, *Internet Protocol*, September 1981, available at <https://www.rfc-editor.org/info/rfc791> [viewed 2022-02-18]

IETF RFC 792, J. Postel, *Internet Control Message Protocol*, September 1981, available at <https://www.rfc-editor.org/info/rfc792> [viewed 2022-02-18]

IETF RFC 793, J. Postel, *Transmission Control Protocol*, September 1981, available at <https://www.rfc-editor.org/info/rfc793> [viewed 2022-02-18]

IETF RFC 796, J. Postel, *Address mappings*, September 1981, available at <https://www.rfc-editor.org/info/rfc796> [viewed 2022-02-18]

IETF RFC 826, D. Plummer, *An Ethernet Address Resolution Protocol: Or Converting Network Protocol Addresses to 48.bit Ethernet Address for Transmission on Ethernet Hardware*, November 1982, available at <https://www.rfc-editor.org/info/rfc826> [viewed 2022-02-18]

IETF RFC 894, C. Hornig, *A Standard for the Transmission of IP Datagrams over Ethernet*, April 1984, available at <https://www.rfc-editor.org/info/rfc894> [viewed 2022-02-18]

IETF RFC 919, J.C. Mogul, *Broadcasting Internet Datagrams*, October 1984, available at <https://www.rfc-editor.org/info/rfc919> [viewed 2022-02-18]

IETF RFC 922, J.C. Mogul, *Broadcasting Internet datagrams in the presence of subnets*, October 1984, available at <https://www.rfc-editor.org/info/rfc922> [viewed 2022-02-18]

IETF RFC 950, J.C. Mogul and J. Postel, *Internet Standard Subnetting Procedure*, August 1985, available at <https://www.rfc-editor.org/info/rfc950> [viewed 2022-02-18]

3 Terms, definitions, symbols, abbreviated terms and conventions

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

ISO and IEC maintain terminological databases for use in standardization at the following addresses:

- IEC Electropedia: available at <https://www.electropedia.org/>
- ISO Online browsing platform: available at <https://www.iso.org/obp>

3.1 Terms and definitions from other ISO/IEC standards

3.1.1 Terms and definitions from ISO/IEC 7498-1

For the purposes of this document, the following terms and definitions given in ISO/IEC 7498-1 apply.

- 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

3.1.2 Terms and definitions from ISO/IEC 8822

For the purposes of this document, the following terms and definitions given in ISO/IEC 8822 apply.

- a) abstract syntax
- b) presentation context

3.1.3 Terms and definitions from ISO/IEC 9545

For the purposes of this document, the following terms and definitions given 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

3.1.4 Terms and definitions from ISO/IEC 8824-1

For the purposes of this document, the following terms and definitions given in ISO/IEC 8824-1 apply.

- a) object identifier
- b) type
- c) value
- d) simple type
- e) structured type
- f) tag
- g) Boolean type
- h) true
- i) false
- j) integer type
- k) bitstring type
- l) octetstring type

3.2 Fieldbus application layer Type 26-specific definitions

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

3.2.1

application

function or data structure for which data is consumed or produced

3.2.2

application objects

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

3.2.3

application process object

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

Note 1 to entry: 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 objects can 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.

3.2.4

application process object class

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

3.2.5

application relationship

cooperative association between two or more application-entity-invocations for the purpose of exchange of information and coordination of their joint operation

Note 1 to entry: This relationship is activated either by the exchange of application-protocol-data-units or as a result of preconfiguration activities.

3.2.6**application relationship ASE**

ASE used for establishing and terminating application relationship

3.2.7**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 1 to entry: Each application process involved in the application relationship maintains its own application relationship endpoint.

3.2.8**attribute**

description of an externally visible characteristic or feature of an object

Note 1 to entry: 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 can also affect the behavior of an object. Attributes are divided into class attributes and instance attributes.

3.2.9**behavior**

indication of how an object responds to particular events

3.2.10**channel**

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

3.2.11**class**

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

Note 1 to entry: A class is a generalization of an object; a template for defining variables and methods. All objects in a class are identical in form and behavior, but usually contain different data in their attributes.

3.2.12**class attributes**

attribute that is shared by all objects within the same class

3.2.13**client**

<object view> object which uses the services of another (server) object to perform a task

3.2.14**client**

<communication view> initiator of a message to which a server reacts

3.2.15**common-memory**

virtual memory accessible with logically unique address used for the cyclic-data transmission

Note 1 to entry: It is composed of the memory-area-1, -2 and -3 of the memory size 512 words, 8 192 words and 1 048 576 words respectively and is shared with the Type 26 nodes in a Type 26 fieldbus network.

3.2.16**connection**

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

Note 1 to entry: Connections can be either point-to-point or multipoint.

3.2.17

consumer

node or sink that is receiving data from a producer

3.2.18

control commands

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

3.2.19

cyclic

repetitive in a regular manner

3.2.20

cyclic-data frame

frame used for the Type 26 cyclic-data transmission

3.2.21

cyclic-data transmission

data transmission with which a Type 26 node sends out data on a pre-assigned area for the node on the common-memory, in broadcasting to all other Type 26 nodes every time the node obtains a token

3.2.22

dedicated AR

AR used directly by the FAL User

Note 1 to entry: On dedicated ARs, only the FAL Header and the user data are transferred.

3.2.23

device

physical hardware connected to the link

Note 1 to entry: A device can contain one or more than one node.

3.2.24

device profile

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

3.2.25

end node

producing or consuming node

3.2.26

endpoint

one of the communicating entities involved in a connection

3.2.27

error

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

3.2.28

event

instance of a change of conditions

3.2.29**FA-link protocol**

communication protocol for Type 26 fieldbus

Note 1 to entry: It defines the token management, the cyclic data transmission and the message data transmission, with which each Type 26 node performs data transmission on an equal basis.

3.2.30**FAL subnet**

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

3.2.31**frame**

synonym for DLPDU

3.2.32**invocation**

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

Note 1 to entry: Each invocation represents a separate thread of control that can 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.

3.2.33**index**

address of an object within an application process

3.2.34**instance**

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 1 to entry: The terms object, instance, and object instance are used to refer to a specific instance.

3.2.35**instance attributes**

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

3.2.36**instantiated**

object that has been created in a device

3.2.37**logical device**

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

3.2.38**management information**

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

Note 1 to entry: Managing includes functions such as controlling, monitoring, and diagnosing.

3.2.39**message-data frame**

frame used for the Type 26 message-data transmission

3.2.40

message-data transmission

sporadic data transmission with which a Type 26 node sends out the message data of request, response, information report, notification upon event happened at the node and so forth, to other Type 26 nodes

3.2.41

method

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

3.2.42

module

hardware or logical component of a physical device

3.2.43

multipoint connection

connection from one node to many

Note 1 to entry: Multipoint connections allow messages from a single producer to be received by many consumer nodes.

3.2.44

network

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

3.2.45

network load measurement function

provision for monitoring and measurement of the communication load over a Type 26 fieldbus with a Type 26 node

3.2.46

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 behavior

3.2.47

participation-request-frame

frame used for a Type 26 node notifying of its participation-request in a Type 26 fieldbus network

3.2.48

peer

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

3.2.49

physical device

automation or other network device

3.2.50

point-to-point connection

connection that exists between exactly two application objects

3.2.51

property

general term for descriptive information about an object

3.2.52**protocol type**

identification of communication protocol

3.2.53**publisher**

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

Note 1 to entry: A publisher cannot be aware of the identity or the number of subscribers and it can publish its APDUs using a dedicated AR.

3.2.54**pull publisher**

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

3.2.55**pull publishing manager**

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

3.2.56**push publisher**

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

3.2.57**push publishing manager**

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

3.2.58**push subscriber**

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

3.2.59**resource**

processing or information capability of a subsystem

3.2.60**sequence number**

consecutive number used to determine whether a receiving message-data from other nodes has been previously received and is redundant

Note 1 to entry: The sequence number of a node is set to 0 (zero) at the time of the network-startup, and it is incremented by 1 (one) every time a message-data transmission is completed. The sequence number of the node is transmitted to other nodes by the message-data transmission.

3.2.61**server**

<communication view> role of an AREP in which it returns a confirmed service response APDU to the client that initiated the request

3.2.62**server**

<object view> object which provides services to another (client) object

3.2.63**service**

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

3.2.64**subscriber**

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

3.2.65**token**

right to send out data on the transmission medium, which is passed successively from node to node for Type 26 fieldbus network

3.2.66**token-frame**

frame used for hand-over the token between Type 26 nodes

3.2.67**transaction code**

code used to discriminate and identify each data transmission frame, and which represents the transaction of a receiving frame

3.2.68**trigger-frame**

frame used in the network-startup state for a Type 26 node notifying of its entrance into a new participation-request over a Type 26 fieldbus network

3.2.69**valid-linking state**

state in which a token is consecutively circulated among nodes over Type 26 fieldbus network

3.2.70**version of sequence number**

version number of the sequence number in a receiving message-data from other nodes, used to determine whether the receiving message-data is valid and is not overdue

Note 1 to entry: The value is set at the time when a Type 26 node completes to participate in a Type 26 fieldbus network, and is maintained until the node drops out of the Type 26 fieldbus network.

3.2.71**virtual-address-space**

virtual memory space with pseudo logical address, used for the message-transmission

3.3 Abbreviated terms and symbols

AE	Application entity
AL	Application layer
ALME	Application layer management entity
ALP	Application layer protocol
ALS	Application layer service
AP	Application process
APO	Application process object
APDU	Application protocol data unit
AR	Application relationship
AREP	Application relationship endpoint

ASCII	American Standard Code for Information Interchange
ASE	Application service element
BCT	Broadcast transmission
C_AD1	Data-head-address on common-memory-area-1
C_AD2	Data-head-address on common-memory-area-2
C_AD3	Data-head-address on common-memory-area-3
CIM	Computer integrated manufacturing
CM	Common-memory
CM1	Common-memory-area-1
CM2	Common-memory-area-2
CM3	Common-memory-area-3
Cnf	Confirmation
C_SZ1	Data-size on common-memory-area-1
C_SZ2	Data-size on common-memory-area-2
C_SZ3	Data-size on common-memory-area-3
DA	Destination-address
DL-	(as a prefix) data-link-
DLC	Data-link connection
DLCEP	Data-link connection endpoint
DLL	Data-link layer
DLS	Data-link service
DLSAP	Data-link service access point
DLSDU	DL-service-data-unit
DNA	Destination-node-number
DNS	Domain name service
FAL	Fieldbus application layer
HMI	Human-machine interface
H_TYPE	Header-type
ID	Identifier
IEC	International Electrotechnical Commission
Ind	Indication
IP	Internet protocol
ISO	International Organization for Standardization
LME	Layer management entity
LSB	Least Significant Bit
M_ADD	Message-offset-address in the virtual-address-space
MAC	Media Access Control
MAJ_VER	FA-link protocol version: major-version
M_CTL	Message-control
MFT	Allowable-minimum-frame-interval-time
MIN_VER	FA-link protocol version: minor-version
MSB	Most Significant Bit
MSN	Manufacturer's model name of a Type 26 node

M_SZ	Message-data-size in the virtual-address-space
M_RLT	Message result
NDN	Node name of Type 26 node
OSI	Open Systems Interconnect
PDU	Protocol data unit
PhL	Ph-layer
PTP	Point-to-Point
P_TYPE	Protocol-type
RCT	Allowable-refresh-cycle-time
Req	Request
RMT	Refresh-cycle-measurement-time
Rsp	Response
RT	Real Time
SA	Source-address
SAP	Service access point
SDU	Service data unit
SEQ	Sequence-number
SNA	Source-node-number
TCD	Transaction code
TCP	Transmission Control Protocol
TW	Token-watchdog
UDP	User Datagram protocol
ULS	Upper layer status
VDN	Vendor code of the node
V_SEQ	Version of sequence number

3.4 Conventions

3.4.1 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 ASE services specified in 6.5 of this document. The service specification defines the services that are provided by the ASE.

3.4.2 General conventions

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

3.4.3 Conventions for class definitions

Class definitions are described using templates. Each template consists of a list of attributes and services 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

- a) The "FAL ASE:" entry is the name of the FAL ASE that provides the services for the class being specified.
- b) The "CLASS:" entry is the name of the class being specified. All objects defined using this template are an instance of this class. The class may be specified by this document, or by a user of this document.
- c) The "CLASS ID:" entry is a number that identifies the class being specified. This number is unique within the FAL ASE that provides 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 document to identify standardized classes. They have been assigned to maintain compatibility with existing national standards. CLASS IDs between 256 and 2 048 are allocated for identifying user defined classes.
- d) 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 document.

- e) The "ATTRIBUTES" label indicate that the following entries are attributes defined for the class.
- 1) 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.
 - 2) 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.
 - 3) 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).
- f) The "SERVICES" label indicates that the following entries are services defined for the class.
- 1) 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.
 - 2) The label "OpsService" designates an operational service (1).
 - 3) The label "MgtService" designates a management service (2).
 - 4) 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.

3.4.4 Conventions for service definitions

3.4.4.1 General

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

3.4.4.2 Service parameters

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

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

- a) parameter name,
- b) request primitive (transmitted from the sender),
- c) indication primitive (transmitted to the receiver),
- d) response primitive (transmitted from the receiver), and
- e) confirm primitive (transmitted to the sender).

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

- M parameter is mandatory for the primitive
- U parameter is a User option, and can be provided or not depending on the 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

- 1) 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, and
- 2) 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.

3.4.4.3 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 FAL.

4 Concepts

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 three parts: the class definitions, the services and the protocol specification. The first two are contained in this document. The protocol specification for each of the ASEs is defined in IEC 61158-6-26.

The fundamental concepts and templates used to describe the application layer service in this document follow IEC 61158-1, Clause 1.

The FAL defined in this document identifies a peer-based controller communications network supporting a Common-memory model in 6.4 as well as client/server models.

5 Data type ASE

5.1 Overview

Fieldbus data types specify the machine independent syntax for application data conveyed by FAL services. The FAL supports the definition and transfer of both basic and constructed data types. Basic types are atomic types. Constructed types are types composed of basic types and other constructed types.

IEC 61158-1, 10.1 provides an overview of the data type ASE and the relationship between data types, and is incorporated herein by reference.

Most basic types are defined from a set of ISO/IEC 8824 types (simple types):

- Basic types have a constant length;
- Two variations are defined, one for defining data types whose length is an integral number of octets, and one for defining data types whose length is bit;
- Some ISO/IEC 8824 types have been extended for fieldbus specific use (specific types). They are defined as simple class subtypes.

Constructed types are of strings, arrays and structures:

- A string shall be composed of an ordered set, variable in number, of homogeneously typed fixed-length elements;
- An array shall be composed of an ordered set of homogeneously typed elements. There are no restrictions on the data type of array elements placed, but each element shall be of the same type;
- A structure shall be made of an ordered set of heterogeneously typed elements called fields. Like arrays, this document does not restrict the data type of fields. However, the fields within a structure do not have to be of the same type.

Data types are defined as instances of the data type class. Defining new types shall be accomplished by providing a new numeric id and supplying values for the attributes defined for the data type class.

User data is transferred between applications by the FAL protocol. All encoding and decoding is performed by the FAL user. The rules for encoding user data in FAL protocol data units are data type dependent. These rules are defined in IEC 61158-6-26.

5.2 Formal definition of data type objects

5.2.1 Data type class definitions

The data type class specifies the root of the data type class tree. Its parent class "top" indicates the top of the FAL class tree. The general form of the template to describe the class definitions is shown below:

FAL ASE:			DATA TYPE ASE
CLASS:			DATA TYPE
CLASS ID:			5 (FIXED LENGTH & STRING), 6 (STRUCTURE), 12 (ARRAY)
PARENT CLASS:			TOP
ATTRIBUTES:			
1	(o)	Key Attribute:	Data Type Numeric Identifier
2	(o)	Key Attribute:	Data Type Name
3	(m)	Attribute:	Format (FIXED LENGTH, STRING, STRUCTURE, ARRAY)
4	(c)	Constraint:	Format = FIXED LENGTH STRING
4.1	(m)	Attribute:	Octet Length
5	(c)	Constraint:	Format = STRUCTURE
5.1	(m)	Attribute:	Number of Fields
5.2	(m)	Attribute:	List of Fields
5.2.1	(o)	Attribute:	Field Name
5.2.2	(m)	Attribute:	Field Data Type
6	(c)	Constraint:	Format = ARRAY

6.1	(m)	Attribute:	Number of Array Elements
6.2	(m)	Attribute:	Array Element Data Type

5.2.2 Attributes

Data type numeric identifier

This optional attribute identifies the numeric identifier of the related data type.

Data type name

This optional attribute identifies the name of the related data type.

Format

This mandatory attribute identifies the data type as a fixed-length, string, array, or data structure.

Octet length

This conditional attribute defines the representation of the dimensions of the associated type object. It is present when the value of the format attribute is "FIXED LENGTH" or "STRING". For FIXED LENGTH data types, it represents the length in octets. For STRING data types, it represents the length in octets for a single element of a string.

Number of fields

This conditional attribute defines the number of fields in a structure. It shall be present when the value of the format attribute is "STRUCTURE".

List of fields

This conditional attribute is an ordered list of fields contained in the structure. Each field is specified by its number and its type. Fields shall be numbered sequentially from 0 (zero) in the order in which they occur. Partial access to fields within a structure is supported by identifying the field by number. This attribute shall be present when the value of the format attribute is "STRUCTURE".

Field name

This conditional, optional attribute specifies the name of the field. It can be present when the value of the format attribute is "STRUCTURE".

Field data type

This conditional attribute specifies the data type of the field. It shall be present when the value of the format attribute is "STRUCTURE". This attribute can itself specify a constructed data type either by referencing a constructed data type definition by its numeric id, or by embedding a constructed data type definition here. When embedding a description, the embedded-data-type description shown below shall be used.

Number of array elements

This conditional attribute defines the number of elements for the array type. Array elements shall be indexed starting at "0" through "n-1" where the size of the array is "n" elements. This attribute shall be present when the value of the format attribute is "ARRAY".

Array element data type

This conditional attribute specifies the data type for the elements of an array. All elements of the array shall have the same data type. It shall be present when the value of the format attribute is "ARRAY". This attribute can itself specify a constructed data type either by referencing a constructed data type definition by its numeric id, or by embedding a constructed data type definition here. When embedding a description, the embedded-data-type description shown below shall be used.

Embedded data type description

This attribute is used to recursively define embedded data types within a structure or array. The template below defines its contents. The attributes shown in the template are defined above in the data type class, except for the embedded-data-type attribute, which is a recursive reference to this attribute. It is used to define nested elements.

ATTRIBUTES:

1	(m)	Attribute:	Format (FIXED LENGTH, STRING, STRUCTURE, ARRAY)
2	(c)	Constraint:	Format = FIXED LENGTH STRING
2.1	(m)	Attribute:	Data Type Numeric ID value
2.2	(m)	Attribute:	Octet Length
3	(c)	Constraint:	Format = STRUCTURE
3.1	(m)	Attribute:	Number of Fields
3.2	(m)	Attribute:	List of Fields
3.2.1	(m)	Attribute:	Embedded Data Type Description
4	(c)	Constraint:	Format = ARRAY
4.1	(m)	Attribute:	Number of Array Elements
4.2	(m)	Attribute:	Embedded Data Type Description

5.3 FAL defined data types

5.3.1 Fixed length types

5.3.1.1 Boolean types

5.3.1.1.1 Boolean

FAL ASE:	DATA TYPE ASE
CLASS:	DATA TYPE
CLASS ID:	5
PARENT CLASS:	TOP
ATTRIBUTES:	
1	Data Type Numeric Identifier = 1
2	Data Type Name = Boolean
3	Format = FIXED LENGTH
4.1	Octet Length = 1

This data type expresses a Boolean data type with the values TRUE (non zero) and FALSE (zero).

5.3.1.1.2 BOOL

This IEC 61131-3 type is the same as Boolean.

5.3.1.2 BitString types

5.3.1.2.1 BitString8

FAL ASE:	DATA TYPE ASE
CLASS:	DATA TYPE
CLASS ID:	5
PARENT CLASS:	TOP
ATTRIBUTES:	
1	Data Type Numeric Identifier = 22
2	Data Type Name = BitString8
3	Format = FIXED LENGTH
5.1	Octet Length = 1

This type contains 1 element of type BitString.

5.3.1.2.2 BitString16

FAL ASE:	DATA TYPE ASE
CLASS:	DATA TYPE
CLASS ID:	5
PARENT CLASS:	TOP
ATTRIBUTES:	
1	Data Type Numeric Identifier = 23
2	Data Type Name = BitString16
3	Format = FIXED LENGTH
5.1	Octet Length = 2

5.3.1.2.3 WORD

This IEC 61131-3 type is the same as BitString16.

5.3.1.2.4 BitString32

FAL ASE:	DATA TYPE ASE
CLASS:	DATA TYPE
CLASS ID:	5
PARENT CLASS:	TOP
ATTRIBUTES:	
1	Data Type Numeric Identifier = 24
2	Data Type Name = BitString32
3	Format = FIXED LENGTH
5.1	Octet Length = 4

5.3.1.2.5 DWORD

This IEC 61131-3 type is the same as BitString32.

5.3.1.3 Numeric types

5.3.1.3.1 Floating Point types

5.3.1.3.1.1 Float32

FAL ASE:	DATA TYPE ASE
CLASS:	DATA TYPE
CLASS ID:	5
PARENT CLASS:	TOP
ATTRIBUTES:	
1	Data Type Numeric Identifier = 8
2	Data Type Name = Float32
3	Format = FIXED LENGTH
4.1	Octet Length = 4

This type has a length of four octets. The format for float32 is that defined by ISO/IEC 60559 as single precision.

5.3.1.3.1.2 REAL

This IEC 61131-3 type is the same as Float32.

5.3.1.3.1.3 Float64

FAL ASE:	DATA TYPE ASE
CLASS:	DATA TYPE
CLASS ID:	5
PARENT CLASS:	TOP
ATTRIBUTES:	
1	Data Type Numeric Identifier = 15
2	Data Type Name = Float64
3	Format = FIXED LENGTH
4.1	Octet Length = 8

This type has a length of eight octets. The format for float64 is that defined by ISO/IEC 60559 as double precision.

5.3.1.3.2 Integer types

5.3.1.3.2.1 Integer8

FAL ASE:	DATA TYPE ASE
CLASS:	DATA TYPE
CLASS ID:	5
PARENT CLASS:	TOP
ATTRIBUTES:	
1	Data Type Numeric Identifier = 2
2	Data Type Name = Integer8
3	Format = FIXED LENGTH
4.1	Octet Length = 1

This integer type is a two's complement binary number with a length of one octet.

5.3.1.3.2.2 SINT

This IEC 61131-3 type is the same as Integer8.

5.3.1.3.2.3 char

This data type is the same as Integer8.

5.3.1.3.2.4 Integer16

FAL ASE:		DATA TYPE ASE	
CLASS:		DATA TYPE	
CLASS ID:		5	
PARENT CLASS:		TOP	
ATTRIBUTES:			
1	Data Type Numeric Identifier	=	3
2	Data Type Name	=	Integer16
3	Format	=	FIXED LENGTH
4.1	Octet Length	=	2

This integer type is a two's complement binary number with a length of two octets.

5.3.1.3.2.5 INT

This IEC 61131-3 type is the same as Integer16.

5.3.1.3.2.6 Integer32

FAL ASE:		DATA TYPE ASE	
CLASS:		DATA TYPE	
CLASS ID:		5	
PARENT CLASS:		TOP	
ATTRIBUTES:			
1	Data Type Numeric Identifier	=	4
2	Data Type Name	=	Integer32
3	Format	=	FIXED LENGTH
4.1	Octet Length	=	4

This integer type is a two's complement binary number with a length of four octets.

5.3.1.3.2.7 DINT

This IEC 61131-3 type is the same as Integer32.

5.3.1.3.2.8 Integer64

FAL ASE:	DATA TYPE ASE
CLASS:	DATA TYPE
CLASS ID:	5
PARENT CLASS:	TOP
ATTRIBUTES:	
1	Data Type Numeric Identifier = 55
2	Data Type Name = Integer64
3	Format = FIXED LENGTH
4.1	Octet Length = 8

This integer type is a two's complement binary number with a length of eight octets.

5.3.1.3.2.9 LINT

This IEC 61131-3 type is the same as Integer64.

5.3.1.3.3 Unsigned types

5.3.1.3.3.1 Unsigned8

FAL ASE:	DATA TYPE ASE
CLASS:	DATA TYPE
CLASS ID:	5
PARENT CLASS:	TOP
ATTRIBUTES:	
1	Data Type Numeric Identifier = 5
2	Data Type Name = Unsigned8
3	Format = FIXED LENGTH
4.1	Octet Length = 1

This type is a binary number. The most significant bit of the most significant octet is always used as the most significant bit of the binary number; no sign bit is included. This type has a length of one octet.

5.3.1.3.3.2 USINT

This IEC 61131-3 type is the same as Unsigned8.

5.3.1.3.3.3 Unsigned16

FAL ASE:		DATA TYPE ASE	
CLASS:		DATA TYPE	
CLASS ID:		5	
PARENT CLASS:		TOP	
ATTRIBUTES:			
1	Data Type Numeric Identifier	=	6
2	Data Type Name	=	Unsigned16
3	Format	=	FIXED LENGTH
4.1	Octet Length	=	2

This type is a binary number. The most significant bit of the most significant octet is always used as the most significant bit of the binary number; no sign bit is included. This unsigned type has a length of two octets.

5.3.1.3.3.4 UINT

This IEC 61131-3 type is the same as Unsigned16.

5.3.1.3.3.5 Unsigned32

FAL ASE:		DATA TYPE ASE	
CLASS:		DATA TYPE	
CLASS ID:		5	
PARENT CLASS:		TOP	
ATTRIBUTES:			
1	Data Type Numeric Identifier	=	7
2	Data Type Name	=	Unsigned32
3	Format	=	FIXED LENGTH
4.1	Octet Length	=	4

This type is a binary number. The most significant bit of the most significant octet is always used as the most significant bit of the binary number; no sign bit is included. This unsigned type has a length of four octets.

5.3.1.3.3.6 UDINT

This IEC 61131-3 type is the same as Unsigned32.

5.3.1.3.3.7 Unsigned64

FAL ASE:	DATA TYPE ASE
CLASS:	DATA TYPE
CLASS ID:	5
PARENT CLASS:	TOP
ATTRIBUTES:	
1	Data Type Numeric Identifier = 56
2	Data Type Name = Unsigned64
3	Format = FIXED LENGTH
4.1	Octet Length = 8

This type is a binary number. The most significant bit of the most significant octet is always used as the most significant bit of the binary number; no sign bit is included. This unsigned type has a length of eight octets.

5.3.1.3.3.8 ULINT

This IEC 61131-3 type is the same as Unsigned64.

5.3.2 String types

5.3.2.1 BitString

FAL ASE:	DATA TYPE ASE
CLASS:	DATA TYPE
CLASS ID:	5
PARENT CLASS:	TOP
ATTRIBUTES:	
1	Data Type Numeric Identifier = 14
2	Data Type Name = BitString
3	Format = STRING
5.1	Octet Length = 1 to n

This string type is defined as a series of BitString8 elements.

5.3.2.2 OctetString

FAL ASE:	DATA TYPE ASE
CLASS:	DATA TYPE
CLASS ID:	5
PARENT CLASS:	TOP
ATTRIBUTES:	
1	Data Type Numeric Identifier = 10
2	Data Type Name = OctetString
3	Format = STRING
4.1	Octet Length = 1 to n

An OctetString is an ordered sequence of octets, numbered from 1 to n. Octet 1 of the sequence is referred to as the first octet. IEC 61158-6-26 defines the order of transmission.