

INTERNATIONAL STANDARD

NORME INTERNATIONALE

**Industrial communication networks – Fieldbus specifications –
Part 5-17: Application layer service definition – Type 17 elements**

**Réseaux de communication industriels – Spécifications des bus de terrain –
Partie 5-17: Définition des services de la couche application - Éléments de
Type 17**

IECNORM.COM : Click to view the full PDF of IEC 61158-5-17:2007



THIS PUBLICATION IS COPYRIGHT PROTECTED

Copyright © 2007 IEC, Geneva, Switzerland

All rights reserved. Unless otherwise specified, no part of this publication may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm, without permission in writing from either IEC or IEC's member National Committee in the country of the requester. If you have any questions about IEC copyright or have an enquiry about obtaining additional rights to this publication, please contact the address below or your local IEC member National Committee for further information.

Droits de reproduction réservés. Sauf indication contraire, aucune partie de cette publication ne peut être reproduite ni utilisée sous quelque forme que ce soit et par aucun procédé, électronique ou mécanique, y compris la photocopie et les microfilms, sans l'accord écrit de l'IEC ou du Comité national de l'IEC du pays du demandeur. Si vous avez des questions sur le copyright de l'IEC ou si vous désirez obtenir des droits supplémentaires sur cette publication, utilisez les coordonnées ci-après ou contactez le Comité national de l'IEC de votre pays de résidence.

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

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

About the IEC

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

About IEC publications

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

IEC Catalogue - webstore.iec.ch/catalogue

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

IEC publications search - www.iec.ch/searchpub

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

IEC Just Published - webstore.iec.ch/justpublished

Stay up to date on all new IEC publications. Just Published details all new publications released. Available online and also once a month by email.

Electropedia - www.electropedia.org

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

IEC Glossary - std.iec.ch/glossary

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

IEC Customer Service Centre - webstore.iec.ch/csc

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

A propos de l'IEC

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

A propos des publications IEC

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

Catalogue IEC - webstore.iec.ch/catalogue

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

Recherche de publications IEC - www.iec.ch/searchpub

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

IEC Just Published - webstore.iec.ch/justpublished

Restez informé sur les nouvelles publications IEC. Just Published détaille les nouvelles publications parues. Disponible en ligne et aussi une fois par mois par email.

Electropedia - www.electropedia.org

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

Glossaire IEC - std.iec.ch/glossary

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

Service Clients - webstore.iec.ch/csc

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

INTERNATIONAL STANDARD

NORME INTERNATIONALE

**Industrial communication networks –Fieldbus specifications –
Part 5-17: Application layer service definition – Type 17 elements**

**Réseaux de communication industriels – Spécifications des bus de terrain –
Partie 5-17: Définition des services de la couche application - Éléments de
Type 17**

INTERNATIONAL
ELECTROTECHNICAL
COMMISSION

COMMISSION
ELECTROTECHNIQUE
INTERNATIONALE

PRICE CODE
CODE PRIX

XA

ICS 25.040.40; 35.100.70

ISBN 978-2-8322-1491-6

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

CONTENTS

CONTENTS	2
FOREWORD.....	4
INTRODUCTION.....	6
1 Scope.....	7
1.1 Overview	7
1.2 Specifications	8
1.3 Conformance.....	8
2 Normative references	8
3 Definitions	8
3.1 Terms and definitions	8
3.2 Abbreviations and symbols	14
3.3 Conventions	14
4 Concepts.....	18
4.1 General.....	18
4.2 Relationships between ASEs	18
4.3 FAL ASEs.....	18
4.4 Common FAL service parameters.....	19
5 ASEs.....	19
5.1 Variable ASE.....	19
5.2 Event ASE.....	23
5.3 Load region ASE	25
5.4 Function invocation ASE	28
5.5 Time ASE	30
5.6 Network management ASE.....	33
5.7 Application relationship ASE	37
6 ARs.....	46
6.1 General.....	46
6.2 Point-to-point user-triggered confirmed client/server AREP (PTC-AR)	46
6.3 Point-to-point user-triggered unconfirmed client/server AREP (PTU-AR)	48
6.4 Point-to-point network-scheduled unconfirmed publisher/subscriber AREP (PSU-AR)	49
6.5 Multipoint user-triggered unconfirmed publisher/subscriber AREP (MTU-AR)	50
6.6 Multipoint network-scheduled unconfirmed publisher/subscriber AREP (MSU-AR).....	51
7 Summary of FAL classes	53
8 Permitted FAL services by AREP role.....	53
Bibliography.....	55
Figure 1 – FAL ASEs	18
Figure 2 – The AR ASE conveys APDUs between APs	37
Table 1 – Read service parameters.....	21
Table 2 – Write service parameters.....	22
Table 3 – Information report service parameters	22

Table 4 – Event notification service parameters	24
Table 5 – Event notification recovery service parameters.....	25
Table 6 – Download service parameters	27
Table 7 – Upload service parameters	27
Table 8 – Start service parameters	29
Table 9 – Stop service parameters.....	29
Table 10 – Resume service parameters	30
Table 11 – Get network time service parameters	31
Table 12 – Set network time service parameters.....	32
Table 13 – Tick notification service parameters.....	32
Table 14 – Get network status service parameters.....	34
Table 15 – Get station status service parameters.....	35
Table 16 – Network status change report service parameters	36
Table 17 – Station status change report service parameters	36
Table 18 – Conveyance of service primitives by AREP role.....	38
Table 19 – Valid combinations of AREP roles involved in an AR.....	38
Table 20 – AR-Unconfirmed Send	42
Table 21 – AR-confirmed send.....	43
Table 22 – AR-establish service.....	44
Table 23 – AR-abort	45
Table 24 – FAL class summary	53
Table 25 – FAL services by AR type	54

IECNORM.COM : Click to view the full PDF of IEC 61158-5-17:2007

INTERNATIONAL ELECTROTECHNICAL COMMISSION

**INDUSTRIAL COMMUNICATION NETWORKS –
FIELD BUS SPECIFICATIONS –****Part 5-17: Application layer service definition – Type 17 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 provides no marking procedure to indicate its approval and cannot be rendered responsible for any equipment declared to be in conformity with an IEC Publication.
- 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.

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

International Standard IEC 61158-5-17 has been prepared by subcommittee 65C: Digital communications, of IEC technical committee 65: Industrial-process measurement and control.

This bilingual version (2014-06) corresponds to the English version, published in 2007-12.

This first edition and its companion parts of the IEC 61158-5 subseries cancel and replace IEC 61158-5:2003. This edition of this part constitutes a technical addition. This part and its Type 17 companion parts also cancel and replace IEC/PAS 62405.

This edition of IEC 61158-5 includes the following significant changes from the previous edition:

- a) deletion of the former Type 6 fieldbus for lack of market relevance;
- b) addition of new types of fieldbuses;
- c) partition of part 5 of the third edition into multiple parts numbered -5-2, -5-3, ...

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

FDIS	Report on voting
65C/475/FDIS	65C/486/RVD

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

The French version of this standard has not been voted upon.

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

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

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

NOTE The revision of this standard will be synchronized with the other parts of the IEC 61158 series.

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

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.

IECNORM.COM : Click to view the full PDF of IEC 61158-5-17:2007

INDUSTRIAL COMMUNICATION NETWORKS – FIELDBUS SPECIFICATIONS –

Part 5-17: Application layer service definition – Type 17 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 17 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 different Types of the 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 IEC fieldbus Application Layer, in conformance with the OSI Basic Reference Model (ISO/IEC 7498) and the OSI Application Layer Structure (ISO/IEC 9545).

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

Although these services specify, from the perspective of applications, how request and responses are issued and delivered, they do not include a specification of what the requesting and responding applications are to do with them. That is, the behavioral aspects of the applications are not specified; only a definition of what requests and responses they can send/receive is specified. This permits greater flexibility to the FAL users in standardizing such object behavior. In addition to these services, some supporting services are also defined in this standard to provide access to the FAL to control certain aspects of its operation.

1.2 Specifications

The principal objective of this 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.

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 do they constrain the implementations of application layer entities within industrial automation systems.

There is no conformance of equipment to this application layer service definition standard. Instead, conformance is achieved through implementation of conforming application layer protocols that fulfill the Type 17 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 all other undated references, the latest edition of the referenced document (including any amendments) applies.

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

ISO/IEC 7498 (all parts), *Information technology – Open Systems Interconnection – Basic Reference Model*

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 Definitions

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

3.1 Terms and definitions

3.1.1 ISO/IEC 7498-1 terms

For the purposes of this document, the following terms as defined in ISO/IEC 7498-1 apply:

- a) application entity
- b) application protocol data unit

c) application service element

3.1.2 ISO/IEC 10731 terms

- a) (N)-connection
- b) (N)-entity
- c) (N)-layer
- d) (N)-service
- e) (N)-service-access-point

3.1.3 Other terms and definitions

3.1.3.1

application

function or data structure for which data is consumed or produced

3.1.3.2

application process

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

3.1.3.3

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

3.1.3.4

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.

3.1.3.5

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.

3.1.3.6

behaviour

indication of how an object responds to particular events

3.1.3.7

bridge

intermediate equipment that connects two or more segments using a data-link layer relay function

3.1.3.8

channel

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

3.1.3.9

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.

3.1.3.10

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

3.1.3.11

connection

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

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

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

3.1.3.12

connection point

buffer which is represented as a subinstance of an Assembly object.

3.1.3.13

conveyance path

unidirectional flow of APDUs across an application relationship

3.1.3.14

dedicated AR

AR used directly by the FAL User

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

3.1.3.15

device

physical hardware connected to the link

NOTE A device may contain more than one node.

3.1.3.16

domain

part of the RTE network consisting of one or two subnetwork(s)

NOTE Two subnetworks are required to compose a dual-redundant RTE network, and each end node in the domain is connected to both of the subnetworks.

3.1.3.17

domain master

station which performs diagnosis of routes to all other domains, distribution of network time to nodes inside the domain, acquisition of absolute time from the network time master and notification of status of the domain

3.1.3.18

domain number

numeric identifier which indicates a domain

3.1.3.19

end node

producing or consuming node

3.1.3.20

endpoint

one of the communicating entities involved in a connection

3.1.3.21**error**

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

3.1.3.22**external bridge**

bridge to which neither internal bridges nor RTE stations are connected directly

3.1.3.23**event**

an instance of a change of conditions

3.1.3.24**group**

a) <general> a general term for a collection of objects. Specific uses:

b) <addressing> when describing an address, an address that identifies more than one entity

3.1.3.25**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

3.1.3.26**interface port**

physical connection point of an end node, which has an independent DL-address

3.1.3.27**internal bridge**

bridge to which no routers, external bridges or nodes non-compliant with this specification are connected directly

3.1.3.28**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.

3.1.3.29**junction bridge**

bridge to which at least one router, external bridge or node non-compliant with this specification, and to which at least one internal bridge or RTE station is connected

3.1.3.30**link**

physical communication channel between two nodes

3.1.3.31**method**

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

3.1.3.32

network

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

3.1.3.33

network time master

station which distributes network time to domain masters

3.1.3.34

node

single DL-entity as it appears on one local link

3.1.3.35

non-redundant interface node

node which has a single interface port

3.1.3.36

non-redundant station

station that consists of a single end node

NOTE "non-redundant station" is synonymous with "end node".

3.1.3.37

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

3.1.3.38

originator

client responsible for establishing a connection path to the target

3.1.3.39

path

logical communication channel between two nodes, which consists of one or two link(s)

3.1.3.40

peer

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

3.1.3.41

producer

node that is responsible for sending data

3.1.3.42

provider

source of a data connection

3.1.3.43

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.

3.1.3.44**redundant interface node**

node with two interface ports one of which is connected to a primary network, while the other is connected to a secondary network

3.1.3.45**redundant station**

station that consists of a pair of end nodes

NOTE Each end node of a redundant station has the same station number, but has a different DL-address.

3.1.3.46**resource**

a processing or information capability of a subsystem

3.1.3.47**RTE station**

station compliant with this specification

3.1.3.48**route**

logical communication channel between two communication end nodes

3.1.3.49**router**

intermediate equipment that connects two or more subnetworks using a network layer relay function

3.1.3.50**segment**

communication channel that connects two nodes directly without intervening bridges

3.1.3.51**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

3.1.3.52**service**

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

3.1.3.53**station**

end node or a pair of end nodes that perform a specific application function

3.1.3.54**station number**

numeric identifier which indicates a RTE station

3.1.3.55**subnetwork**

part of a network that does not contain any routers. A subnetwork consists of end nodes, bridges and segments

NOTE Every end node included in a subnetwork has the same IP network address.

3.1.3.56

subscriber

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

3.2 Abbreviations and symbols

3.2.1 ISO/IEC 10731 abbreviations

ASE	application-service-element
OSI	Open Systems Interconnection

3.2.2 Other abbreviations and symbols

AE	Application entity
AL	Application layer
AP	Application process
APDU	Application protocol data unit
APO	Application object
AR	Application relationship
AREP	Application relationship end point
Cnf	Confirmation
DL-	data-link layer (as a prefix)
DLL	DL-layer
DLSAP	DL-service-access-point
FAL	Fieldbus application layer
FIFO	First-in first-out (queuing method)
ID	Identifier
IEC	International Electrotechnical Commission
Ind	Indication
ISO	International Organization for Standardization
MSU-AR	Multipoint network-scheduled unconfirmed publisher/subscriber AREP
MTU-AR	Multipoint user-triggered unconfirmed publisher/subscriber AREP
PDU	Protocol Data Unit
PSU-AR	Point-to-point network-scheduled unconfirmed client/server AREP
PTC-AR	Point-to-point user-triggered confirmed client/server AREP
PTU-AR	Point-to-point user-triggered unconfirmed client/server AREP
Req	Request
Rsp	Response
SAP	Service Access Point

3.3 Conventions

3.3.1 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.

3.3.2 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.

3.3.3 Conventions for service definitions

3.3.3.1 General

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

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

3.3.3.2 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.

NOTE 1 See the note under 3.3.3.3 relative to the non-inclusion of service parameters that are appropriate to a protocol specification or programming interface specification or implementation specification, but not to an abstract service definition.

This standard 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 standard. Each table consists of up to six columns: a column for the name of the service parameter, and a column each for those primitives and parameter-transfer directions used by the service. The possible six columns are:

- 1) the parameter name;
- 2) the request primitive's input parameters;
- 3) the request primitive's output parameters;

NOTE 2 This is a seldom-used capability. Unless otherwise specified, request primitive parameters are input parameters.

- 4) the indication primitive's output parameters;
- 5) the response primitive's input parameters; and

6) the confirm primitive's output parameters.

NOTE 3 The request, indication, response and confirm primitives are also known as requestor.submit, acceptor.deliver, acceptor.submit, and requestor.deliver primitives, respectively (see ISO/IEC 10731).

One parameter (or 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.

3.3.3.3 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.

NOTE The IEC 61158-5 series of standards define sets of abstract services. They are neither protocol specifications nor implementation specifications nor concrete programming interface specifications. Therefore there are restrictions on the extent to which service procedures can be mandated in the parts of IEC 61158-5. Protocol aspects that can vary among different protocol specifications or different implementations that instantiate the same abstract services are unsuitable for inclusion in these service definitions, except at the level of abstraction that is necessarily common to all such expressions.

For example, the means by which service providers pair request and reply PDUs is appropriate for specification in an IEC 61158-6 protocol specification standard but not in an IEC 61158-5 abstract service definition standard. Similarly, local implementation methods by which a service provider or service user pairs request and confirm(ation) primitives, or indication and response primitives, is appropriate for an implementation specification or for a programming interface specification, but not for an abstract service standard or for a protocol standard, except at a level of abstraction that is necessarily common to all embodiments of the specifying standard. In all cases, the abstract definition is not permitted to over-specify the more concrete instantiating realization.

Further information on the conceptual service procedures of an implementation of a protocol that realizes the services of one of the IEC 61158-5 abstract service definitions can be found in IEC/TR 61158-1 (Ed.2.0), 9.6.

4 Concepts

4.1 General

This Fieldbus Application Layer (FAL) provides communication services to time-critical and non-time-critical applications in fieldbus devices.

The common concepts and templates used to describe the application layer service in this standard are detailed in IEC/TR 61158-1, Clause 9.

4.2 Relationships between ASEs

4.3 FAL ASEs

A modular approach was 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.

To support remote access to the AP, the Application Relationship ASE is defined. This application relationship ASE provides services to the AP for defining and establishing communication relationships with other APs, and provides services to other ASEs for conveying their service requests and responses.

Each FAL ASE defines a set of services, APDUs and procedures that operate on the classes that it represents. Only a subset of the ASE services may be provided to meet the needs of an application. Profiles may be used to define such subsets.

APDUs are sent and received between FAL ASEs that support the same services. Figure 1 shows the FAL ASEs from the perspective of communication.

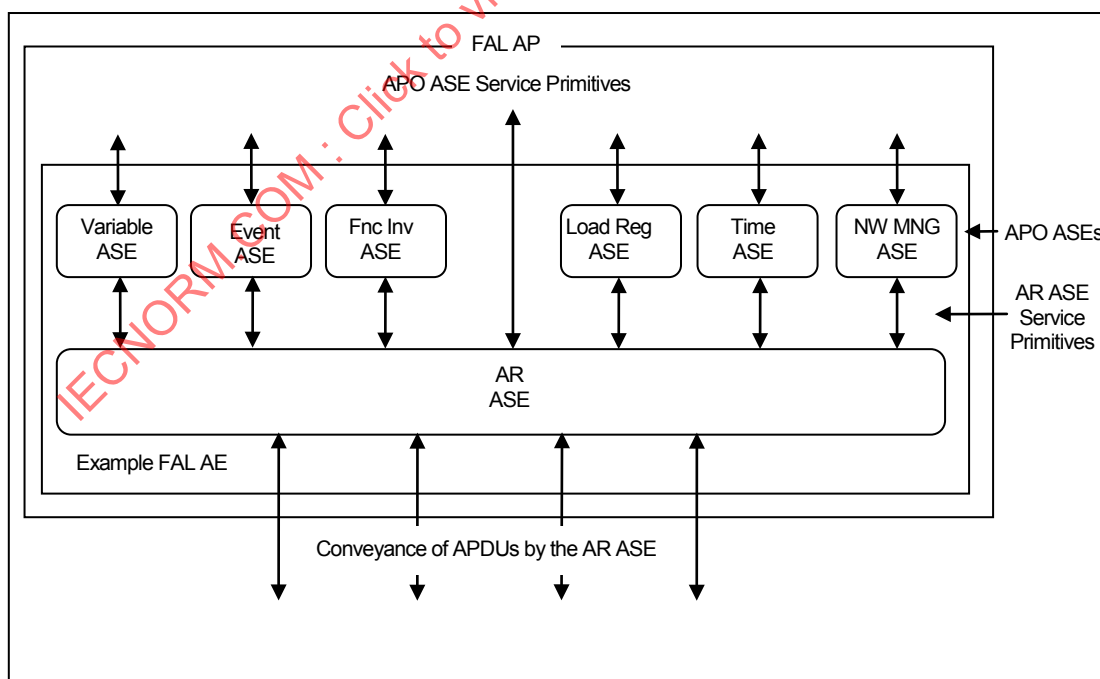


Figure 1 – FAL ASEs

4.4 Common FAL service parameters

Many services have the following parameters. Instead of defining them for each service, the following common definitions are provided:

Argument

This parameter carries the parameters of the service invocation.

AREP

This parameter specifies information sufficient for local identification of the AREP to be used to convey the service. This parameter may use a key attribute of the AREP to identify the application relationship. When an AREP supports multiple contexts simultaneously, the AREP parameter is extended to identify the context as well as the AREP.

Result(+)

This selection type parameter indicates that the service request was successful.

Result(-)

This selection type parameter indicates that the request failed.

Error info

This parameter provides error information for service errors. It is returned in confirmed service response(-) primitives.

5 ASEs

5.1 Variable ASE

5.1.1 Overview

In the fieldbus environment, application processes contain data that can be both read and written by remote applications. The variable ASE defines the network-visible attributes of application data and provides a set of services used to read, write and report their values.

When the appropriate types of application relationship are supported, the variable ASE may be used to support two different access models i.e., the client/server model and the publisher/subscriber model. The client/server model is characterized by a client application sending a read or write request to a server application that responds accordingly. The server's activity is stimulated by clients on the network; if there are no requests, the server generates no responses.

The publisher/subscriber model is different in that it is characterized by a data producer publishing its data onto the network. Subscribers wishing to acquire the published data join the application relationship used to publish it and listen for the data as it is transmitted. Two models are provided to support this publisher/subscriber activity, i.e., the "pull" model and the "push" model.

In the "pull" model, one of the subscribers requests that the publisher publishes a sequence of variable data by issuing a publish request to it. The publisher distributes the variable data periodically according to the remote request by multicasting. The publishing schedule is controlled by the publisher itself.

In the "push" model, the publisher is requested to distribute a sequence of variable data by the local FAL user. The publisher distributes a sequence of variable data by multicasting according to the local request. The publishing schedule is also controlled by the publisher itself.

5.1.2 Variable class specification

5.1.2.1 Formal model

FAL ASE:			VARIABLE ASE
CLASS:			VARIABLE
CLASS ID:			not used
PARENT CLASS:			TOP
ATTRIBUTES:			
1	(m)	Key Attribute:	Numeric Identifier
2	(m)	Attribute:	Data type
3	(0)	Attribute:	Length
4	(o)	Attribute:	Read enable
5	(o)	Attribute:	Write enable
SERVICES:			
1	(o)	OpsService:	Read
2	(o)	OpsService:	Write
3	(o)	OpsService:	Information Report

5.1.2.2 Attributes

Numeric identifier

This key attribute identifies an instance of this object class.

Data type

This attribute is the numeric identifier of the data type.

Length

This optional attribute is the length of the data in octets.

Read enable

The value of this optional attribute indicates whether the data value of the Variable Object can be read via the fieldbus.

Write enable

The value of this optional attribute indicates whether the data value of the Variable Object can be updated via the fieldbus.

5.1.3 Variable List class specification

5.1.3.1 Formal model

FAL ASE:			VARIABLE ASE
CLASS:			VARIABLE LIST
CLASS ID:			not used
PARENT CLASS:			TOP
ATTRIBUTES:			
1	(m)	Key Attribute:	Numeric Identifier
2	(m)	Attribute:	Number of Entries
3	(m)	Attribute:	List Of Variables
SERVICES:			
1	(o)	OpsService:	Read
2	(o)	OpsService:	Write
3	(o)	OpsService:	Information Report

5.1.3.2 Attributes

Numeric identifier

This key attribute identifies an instance of this object class.

Number of entries

This attribute specifies the number of variables in the list.

List of variables

This attribute identifies the variables by the key attributes that are contained in the list.

5.1.4 Variable ASE service specification

5.1.4.1 Supported services

This subclause contains the definition of services that are unique to this ASE. The services defined for this ASE are:

- Read
- Write
- Information Report

5.1.4.2 Read service

5.1.4.2.1 Service overview

This confirmed service may be used to read the value of a variable object or a variable list object. It is not used with the publisher/subscriber model.

5.1.4.2.2 Service primitives

The service parameters for each primitive are shown in Table 1.

Table 1 – Read service parameters

Parameter name	Req	Ind	Rsp	Cnf
Argument				
AREP	M	M		
Variable Specifier	M	M (=)		
Result(+)			S	S (=)
Value			M	M (=)
Result(–)			S	S (=)
Error Info			M	M (=)
NOTE The method by which a confirm primitive is correlated with its corresponding preceding request primitive is a local matter. The method by which a response primitive is correlated with its corresponding preceding indication primitive is a local matter. See 1.2.				

Variable specifier

This parameter specifies a variable object or a variable list object to be read by the key attribute.

Value

This parameter specifies the value read. For each of the variables, this parameter specifies the value of the variable. For variable lists, this parameter specifies the values of each of the variables in the list concatenated together in the order in which they appear in the list.

NOTE If any of the variables in a variable list could not be read, the service fails.

5.1.4.3 Write service

5.1.4.3.1 Service overview

This confirmed service is used to write the value of a variable object or a variable list object. It is not used with the publisher/subscriber model.

5.1.4.3.2 Service primitives

The service parameters for each primitive are shown in Table 2.

Table 2 – Write service parameters

Parameter name	Req	Ind	Rsp	Cnf
Argument				
AREP	M	M		
Variable Specifier	M	M (=)		
Value	M	M (=)		
Result(+)			S	S (=)
Result(-)			S	S (=)
Error Info			M	M (=)
NOTE The method by which a confirm primitive is correlated with its corresponding preceding request primitive is a local matter. The method by which a response primitive is correlated with its corresponding preceding indication primitive is a local matter. See 1.2.				

Variable specifier

This parameter specifies a variable object or a variable list object to be written by the key attribute.

Value

This parameter specifies the value to write. For variables, this parameter specifies the value of the variable. For variable lists, this parameter specifies the values of each of the variables in the list concatenated together in the order that they appear in the list.

NOTE If any variable in a variable list object cannot be updated, none of the variables in the variable list object will be updated and the write will fail.

5.1.4.4 Information report service

5.1.4.4.1 Service overview

This optional service is an unconfirmed service that may be used to report the value of a variable object or a variable list object. This service may be used in the publisher/subscriber push model.

5.1.4.4.2 Service primitives

The service parameters for each primitive are shown in Table 3.

Table 3 – Information report service parameters

Parameter name	Req	Ind
Argument		
AREP	M	M
Variable Specifier	M	M (=)
Value	M	M (=)

Variable specifier

This parameter specifies a variable object or a variable list object to be reported by the key attribute.

Value

This parameter specifies the value to be reported. For variables, this parameter specifies the value of the variable. For variable lists, this parameter specifies the value of each variable in the list concatenated together in the order that they appear in the list.

NOTE If any of the variables in a variable list could not be read, the service fails.

5.2 Event ASE**5.2.1 Overview**

Event objects are used to define messages reporting event occurrences. Event messages contain information that identifies and describes event occurrences.

Notifiers are responsible for collecting event messages from event objects, and distributing one or more event message(s) in a single invocation of the FAL event notification service. The number of event messages that may be submitted in a single service invocation is limited by the maximum APDU size that can be transferred by the AR.

If an application process fails to receive one or more event notifications, a notification recovery service is provided to request the notifier to retransmit the notifications.

In this model, application processes are responsible for providing the functions for event objects and event list objects, and the FAL is responsible for providing communication services designed specifically for them. The application process detects events, builds event messages and aggregates them together. The application process distributes the aggregated event messages using the FAL event notification service.

5.2.2 Event class specification**5.2.2.1 Formal model**

FAL ASE:			EVENT ASE
CLASS:			NOTIFIER
CLASS ID:			not used
PARENT CLASS:			TOP
ATTRIBUTES:			
1	(m)	Key Attribute:	Numeric Identifier
2	(m)	Attribute:	AREP
3	(o)	Attribute:	Last Notification Sequence Number
4	(o)	Attribute:	List Of Events
SERVICES:			
1	(o)	OpsService:	Event Notification
2	(o)	OpsService:	Notification Recovery

5.2.2.2 Attributes**5.2.2.2.1 Numeric identifier**

This key attribute identifies an instance of this object class.

5.2.2.2.2 AREP

This attribute identifies the AREP configured to convey event notifications. This AREP is also used for reporting the event notifications generated by an event recovery request.

5.2.2.2.3 Last notification sequence number

The conditional attribute specifies the last sequence number used. It is incremented for each event notification service invocation.

5.2.2.2.4 List of events

This optional attribute identifies the events that are configured.

5.2.3 Event ASE service specification

5.2.3.1 Supported services

This subclause contains the definition of services that are unique to this ASE. The services defined for this ASE are:

- Event Notification
- Notification Recovery

5.2.3.2 Event notification service

5.2.3.2.1 Service overview

This unconfirmed service is used by the notifier of an FAL AP to notify other APs that one or more events have occurred.

5.2.3.2.2 Service primitives

The service parameters for each primitive are shown in Table 4.

Table 4 – Event notification service parameters

Parameter name	Req	Ind
Argument		
AREP	M	M
NotifierID	M	M (=)
Sequence Number	U	U (=)
Notification Time	U	U (=)
List of Event Messages	M	M (=)
Event Key Attribute	M	M (=)
Event Data type	C	C (=)
Event Detection Time	U	U (=)
Event Data	U	U (=)

NotifierID

This parameter identifies the notifier issuing the event notification.

Sequence number

This optional parameter is the sequence number for the event notification. It may be used for notification recovery purposes.

Notification time

This optional parameter is the time of the event notification.

List of event messages

This parameter specifies the list of event messages that are to be reported. It may contain messages from one or more event objects. The contents of each message are specified by its event object and should be consistent with that specified for the notifier object.

Event key attribute

This parameter identifies each of the specific events being acknowledged by this service.

Event data type

This conditional parameter indicates the data type of each of the event data parameters. This parameter may be present only if the event data parameter is present. If the event data parameter is present, this parameter should be present.

Event detection time

This optional parameter reports the time of the event detection. This parameter is present only if it is defined for the specified event object and is supported by the specified notifier.

Event data

This optional parameter specifies user data to be included in an event message in addition to that used to identify the event occurrence. This parameter is present only if it is defined for the specified event object and is supported by the specified notifier.

5.2.3.3 Notification recovery service**5.2.3.3.1 Service overview**

This unconfirmed service is used to request that a specified number of retained event notifications be returned. Notifications are returned using the event notification service.

5.2.3.3.2 Service primitives

The service parameters for each primitive are shown in Table 5.

Table 5 – Event notification recovery service parameters

Parameter name	Req	Ind
Argument		
AREP	M	M
NotifierID	M	M (=)
Sequence Number	U	U (=)

NotifierID

This parameter identifies the notifier to which this service is directed.

Sequence number

This optional parameter specifies the sequence number of the event notification to be re-sent. If not present, the last notification sent is being requested.

5.3 Load region ASE**5.3.1 Overview**

A load region represents an unstructured memory area whose contents is to be uploaded (read) or downloaded (written). In this context, “unstructured” means that the memory area is represented only as an ordered sequence of octets. No other structure is apparent.

A load region may represent an unnamed volatile memory area, such as that implemented by dynamic computer memory, or a named non-volatile memory object, such as a file. The contents of a load region are referred to as a load image and can contain programs or data. The transfer of a load image to or from a load region is performed using the load process.

This load region model provides services that permit an application process to initiate the downloading or uploading of specified load regions.

5.3.2 Load region class specification

5.3.2.1 Formal model

FAL ASE:			LOAD REGION ASE
CLASS:			LOAD REGION
CLASS ID:			not used
PARENT CLASS:			TOP
ATTRIBUTES:			
1	(m)	Attribute:	Load Region Size
2	(m)	Attribute:	Local Address
3	(o)	Attribute:	Load Image Name
SERVICES:			
1	(m)	OpsService:	Download Services
2	(m)	OpsService:	Upload Services

5.3.2.2 Attributes

Load region size

This attribute specifies the maximum size of the load region in octets.

Local address

This attribute is a locally significant address of the load region.

Load image name

This optional attribute specifies the name of the load image contained in the load region.

5.3.3 Load region ASE service specification

5.3.3.1 Supported services

This subclause contains the definitions of services that are unique to this ASE. The services defined for this ASE are:

- Download
- Upload

5.3.3.2 Download service

5.3.3.2.1 Service overview

This confirmed service is used to download a load image in its request and indication primitives. The response and confirmation primitives are used to convey the success or failure of the download.

5.3.3.2.2 Service primitives

The service parameters for each primitive are shown in Table 6.

Table 6 – Download service parameters

Parameter name	Req	Ind	Rsp	Cnf
Argument				
AREP	M	M		
Load Region	M	M (=)		
Load Data	M	M (=)		
Result(+)			S	S (=)
Result(–)			S	S (=)
Error Info			M	M (=)
NOTE The method by which a confirm primitive is correlated with its corresponding preceding request primitive is a local matter. The method by which a response primitive is correlated with its corresponding preceding indication primitive is a local matter. See 1.2.				

Load region

This parameter specifies the load region into which the image is to be downloaded.

Load data

This parameter specifies the data to be downloaded.

5.3.3.3 Upload service**5.3.3.3.1 Service overview**

This confirmed service is used to upload a load image. Its request and indication primitives convey an upload request. The response and confirmation primitives are used to convey the load image of the load region and the result of loading.

5.3.3.3.2 Service primitives

The service parameters for each primitive are shown in Table 7.

Table 7 – Upload service parameters

Parameter name	Req	Ind	Rsp	Cnf
Argument				
AREP	M	M (=)		
Load Region	M	M (=)		
Result(+)			S	S (=)
Load Data			M	M (=)
Result(–)			S	S (=)
Error Info			M	M (=)
NOTE The method by which a confirm primitive is correlated with its corresponding preceding request primitive is a local matter. The method by which a response primitive is correlated with its corresponding preceding indication primitive is a local matter. See 1.2.				

Load region

This parameter specifies the load region from which the image is to be uploaded.

Load data

This parameter specifies the data uploaded.

5.4 Function invocation ASE

5.4.1 Overview

The function invocation class models the state-oriented function invocation. It may be used to model software processes or user functions the operation of which may be controlled.

5.4.2 Function invocation class specification

5.4.2.1 Formal model

FAL ASE:		FUNCTION INVOCATION ASE
CLASS:		FUNCTION INVOCATION
CLASS ID:		not used
PARENT CLASS:		TOP
ATTRIBUTES:		
1 (m)	Key Attribute:	Identifier
2 (m)	Attribute:	Function Invocation State
SERVICES:		
1 (o)	OpsService:	Start
2 (o)	OpsService:	Stop
3 (o)	OpsService:	Resume

5.4.2.2 Attributes

5.4.2.2.1 Identifier

This key attribute consists of a station identifier and a function identifier.

5.4.2.2.2 Function invocation state

This attribute indicates the current state of the function invocation. An enumerated set of values has been defined.

UNRUNNABLE	This state indicates that the function invocation is not executing and can not be executed.
IDLE	This state indicates that the function invocation is not executing, but is capable of being executed.
RUNNING	This state indicates that the function invocation is executing.
STOPPED	This state indicates that the execution of a function invocation has been suspended.

5.4.3 Function invocation ASE service specification

5.4.3.1 Supported services

This subclause contains the definition of services that are unique to this ASE. The services defined for this ASE are:

- Start
- Stop
- Resume

5.4.3.2 Start service

5.4.3.2.1 Service overview

This confirmed service is used to start a function invocation from the initial condition.

5.4.3.2.2 Service primitives

The service parameters for each primitive are shown in Table 8.

Table 8 – Start service parameters

Parameter name	Req	Ind	Rsp	Cnf
Argument				
AREP	M	M		
FunctionID	M	M (=)		
Result				
Error Info			M	M (=)
NOTE The method by which a confirm primitive is correlated with its corresponding preceding request primitive is a local matter. The method by which a response primitive is correlated with its corresponding preceding indication primitive is a local matter. See 1.2.				

FunctionID

This parameter specifies one of the key attributes of the function invocation object.

5.4.3.3 Stop service

5.4.3.3.1 Service overview

This confirmed service is used to stop a function invocation retaining its context so that it may be resumed.

5.4.3.3.2 Service primitives

The service parameters for each primitive are shown in Table 9.

Table 9 – Stop service parameters

Parameter name	Req	Ind	Rsp	Cnf
Argument				
AREP	M	M (=)		
FunctionID	M	M (=)		
Result				
Error Info			M	M (=)
NOTE The method by which a confirm primitive is correlated with its corresponding preceding request primitive is a local matter. The method by which a response primitive is correlated with its corresponding preceding indication primitive is a local matter. See 1.2.				

FunctionID

This parameter specifies one of the key attributes of the function invocation object.

5.4.3.4 Resume service

5.4.3.4.1 Service overview

This confirmed service is used to request to start a function invocation from the suspended condition.

5.4.3.4.2 Service primitives

The service parameters for this service are shown in Table 10.

Table 10 – Resume service parameters

Parameter name	Req	Ind	Rsp	Cnf
Argument				
AREP	M	M (=)		
FunctionID	M	M (=)		
Result				
Error Info			M	M (=)
NOTE The method by which a confirm primitive is correlated with its corresponding preceding request primitive is a local matter. The method by which a response primitive is correlated with its corresponding preceding indication primitive is a local matter. See 1.2.				

FunctionID

This parameter specifies one of the key attributes of the function invocation object.

5.5 Time ASE

5.5.1 Overview

The Time ASE specifies the clock synchronization mechanism to synchronize the time among network stations. This time can be used to add a time value to alert information and to sequence messages in a time-wise order.

A system consists of one or more time master(s) and several time slaves. Clock synchronization is performed by the communication system. Adjustment and management of the clock are tasks of the application.

The network should have at least one network time master, which may be one of the domain time masters selected automatically or may be a fixed station.

The network time master has the master clock of the network. The master clock may be the local clock of the network time master itself or it may be synchronized with a clock source outside the network.

The domain time master of each domain obtains network time from the network time master and it distributes the network time value to stations in the domain.

5.5.2 Time class specification

5.5.2.1 Formal model

FAL ASE:			Time ASE
CLASS:			Time
CLASS ID:			not used
PARENT CLASS:			TOP
ATTRIBUTES:			
1	(m)	Attribute:	Implicit
2	(m)	Attribute:	Role
3	(o)	Attribute:	Stratum
4	(o)	Attribute:	Poll Interval
5	(o)	Attribute:	Precision
SERVICES:			
1	(m)	OpsService:	Get Network Time
2	(m)	OpsService:	Set Network Time
3	(m)	OpsService:	Tick Notification service

5.5.2.2 Attributes

5.5.2.2.1 Role

This attribute specifies the role of the Time ASE with values defined as follows:

NETWORK TIME MASTER	A network unique end node which responds to time requests from domain masters.
DOMAIN TIME MASTER	A domain unique end node which distributes time information within the domain.
TIME SLAVE	An end node subscribes time information distributed by the domain master.

5.5.2.2.2 Stratum

This attribute indicates the stratum level of the local clock.

5.5.2.2.3 Poll interval

This attribute indicates the maximum interval between successive time messages.

5.5.2.2.4 Precision

This attribute indicates the precision level of the local clock.

5.5.3 Time ASE service specification

5.5.3.1 Supported services

This subclause contains the definitions of services that are unique to this ASE. The services defined for this ASE are:

- Get Network Time
- Set Network Time
- Tick Notification

5.5.3.2 Get network time service

5.5.3.2.1 Service overview

This local service should be used to obtain the network time value that is distributed from the network. If Time ASE is the network time master, the local time value of the master clock is returned.

5.5.3.2.2 Service primitives

The service parameters for each primitive are shown in Table 11.

Table 11 – Get network time service parameters

Parameter name	Req	Cnf
Argument		
Result		
Network Time		M
Status		M
Error info		C
NOTE The method by which a confirm primitive is correlated with its corresponding preceding request primitive is a local matter. See 1.2.		

Network time

This parameter is the value of the network time.

Status

This parameter indicates the states of time synchronization. The possible states are:

- Not synchronized
- Synchronized to the domain time master
- Synchronized to the network time master as the domain time master
- Synchronized to an external time source as a network time master

5.5.3.3 Set network time service

5.5.3.3.1 Service overview

This confirmed service should be used to set the value of network time to the time master.

This service is available under conditions where the network time master is not synchronized to an external time source.

5.5.3.3.2 Service primitives

The service parameters for each primitive are shown in Table 12.

Table 12 – Set network time service parameters

Parameter name	Req	Ind	Rsp	Cnf
Argument Network Time	M	M (=)		
Result Error Info			C	C (=)
NOTE The method by which a confirm primitive is correlated with its corresponding preceding request primitive is a local matter. The method by which a response primitive is correlated with its corresponding preceding indication primitive is a local matter. See 1.2.				

Network time

This parameter is the value of the network time to be set.

5.5.3.4 Tick notification service

5.5.3.4.1 Service overview

This local service should be used to recognize tick timing synchronized to the network time. The tick interval is configurable.

5.5.3.4.2 Service primitives

The service parameters for each primitive are shown in Table 13.

Table 13 – Tick notification service parameters

Parameter name	Ind
Result Tick	M

Tick

This parameter indicates tick timing.

NOTE This service may be implemented by means of a hard-wired interruption.

5.6 Network management ASE**5.6.1 Overview**

End nodes may have two network interfaces that provide network redundancy. These interfaces are referred to as network interface A and network interface B. The former should be connected to the primary network, while the latter should be connected to the secondary network.

Each Network Management ASE constructs two diagnostic message APDUs and sends them concurrently to network interfaces A and B. Each Network Management ASE receives pairs of APDUs from other end nodes on the network that participate in network redundancy. This information is used to determine which network interface (A or B) is to be selected to send other APDUs. This information is also used to determine whether the destination node is reachable in advance of sending an APDU.

This information is preserved in the network status table.

5.6.2 Network management class specification**5.6.2.1 Formal model**

FAL ASE:			Network Management
CLASS:			Network Management
CLASS ID:			not used
PARENT CLASS:			TOP
ATTRIBUTES:			
1	(m)	Attribute:	Number of Network Interfaces
2	(m)	Attribute:	Diagnostic Message Interval
3	(m)	Attribute:	Aging Time
4	(m)	Attribute:	List of Network Interface Status
SERVICES:			
1	(m)	OpsService:	Get Network Status
2	(m)	OpsService:	Get Station Status
3	(m)	OpsService:	Network Status Change Report
4	(m)	OpsService:	Station Status Change Report

5.6.2.2 Attributes**5.6.2.2.1 Number of network interfaces**

This attribute specifies the number of network interfaces on this end node. An end node may have one or two network interfaces.

5.6.2.2.2 Diagnostic message interval

This attribute specifies the time interval between successive invocations of the diagnostic messages in milliseconds.

5.6.2.2.3 Aging time

This attribute specifies the time interval used to remove silent end nodes from the List of Network Interface Status.

5.6.2.2.4 List of network interface status

This attribute is a list of the path status representing the condition of the path to a remote station from which the Network Management ASE may be receiving successive Diagnostic Messages. The status is bi-directional for a path and indicates whether Diagnostic Messages are being received successfully at each end of the path or whether either end of the path is in a fault state.

5.6.3 Network management ASE service specification

5.6.3.1 Supported services

This subclause contains the definition of the services that is unique to this ASE.

The services defined for this ASE are

- Get Network Status
- Get Station Status
- Network Status Change Report
- Station Status Change Report

5.6.3.2 Get network status service

5.6.3.2.1 Service overview

This local service is used to obtain the status of the network.

5.6.3.2.2 Service primitives

The service parameters for each primitive are shown in Table 14.

Table 14 – Get network status service parameters

Parameter name	Req	Cnf
Argument		
Result		
Network Status		M
NOTE The method by which a confirm primitive is correlated with its corresponding preceding request primitive is a local matter. See 1.2.		

Network status

This parameter indicates consistency of the primary and secondary networks. Possible values are

- HEALTHY
- FAILED

5.6.3.3 Get station status service

5.6.3.3.1 Service overview

This local service is used to obtain the status of the specified station.

5.6.3.3.2 Service primitives

The service parameters for each primitive are shown in Table 15.

Table 15 – Get station status service parameters

Parameter name	Req	Cnf
Argument		
Station Identifier	M	
Result		
Station Status		M
Route Status		M
NOTE The method by which a confirm primitive is correlated with its corresponding preceding request primitive is a local matter. See 1.2.		

Station identifier

This parameter indicates the station for which the status is requested, and consists of a domain number and a station number.

Station status

This parameter indicates the status of the station specified by the station identifier, and includes the following information.

a) existence

TRUE	the station exists
FALSE	the station does not exist

b) redundancy

SINGLE	the station is not redundant
DUAL-REDUNDANT	the station is redundant

c) AREP of the end node which is on service

Route status

This parameter indicates the status of route for the station, and includes the following information.

a) status of the route on the primary network

REACHABLE
UNREACHABLE

b) status of the route on the secondary network

REACHABLE
UNREACHABLE

5.6.3.4 Network status change report service**5.6.3.4.1 Service overview**

This local service is used to inform of changes in network status.

5.6.3.4.2 Service primitives

The service parameters for each primitive are shown in Table 16.

Table 16 – Network status change report service parameters

Parameter name	Ind
Result	
Network Status	M

Network status

This parameter indicates consistency of the primary and secondary networks, and has the following possible values:

- HEALTHY
- FAILED

5.6.3.5 Station status change report service

5.6.3.5.1 Service overview

This local service is used to inform of changes in the station status.

5.6.3.5.2 Service primitives

The service parameters for each primitive are shown in Table 17.

Table 17 – Station status change report service parameters

Parameter name	Ind
Result	
Station Identifier	M
Station Status	M
Route Status	M

Station identifier

This parameter indicates the station of which the status has changed.

Station status

This parameter indicates the status of the end node specified in/by the request primitive and includes the following information:

- a) existence

TRUE	the station exists
FALSE	the station does not exist
- b) redundancy

SINGLE	the station is not redundant
DUAL-REDUNDANT	the station is redundant
- c) AREP of the end node which is on service

Route status

This parameter indicates the status of the route for the station, and includes the following information:

- a) status of the route on the primary network

REACHABLE
UNREACHABLE

- b) status of the route on the secondary network

REACHABLE

UNREACHABLE

5.7 Application relationship ASE

5.7.1 Overview

5.7.1.1 General

In a distributed system, application processes communicate with each other by exchanging Application Layer messages across well-defined Application Layer communication channels.

These communication channels are modelled in the Fieldbus Application Layer as application relationships (ARs).

ARs are responsible for conveying messages between applications according to specific communication characteristics required by time-critical systems. Different combinations of these characteristics lead to the definitions of different types of ARs. The characteristics of ARs are defined formally as attributes of AR endpoint classes.

The messages that are conveyed by ARs are FAL service requests and responses. Each is submitted to the AR ASE for transfer by an FAL Application Service Element (ASE) that represents the class of the APO being accessed. Figure 2 illustrates this concept.

Depending on the type of AR, APDUs may be sent to one or more destination application processes connected by the AR. Other characteristics of the AR determine how APDUs are to be transferred. These characteristics are described below.

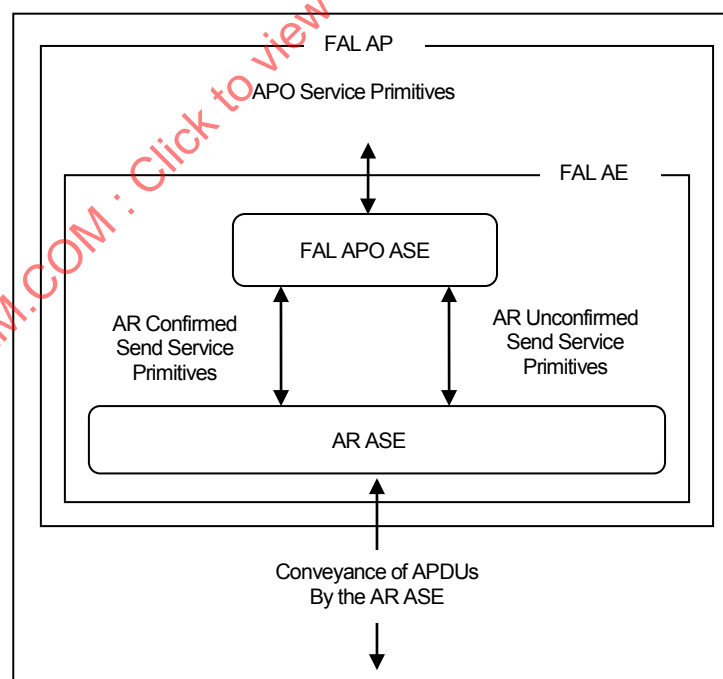


Figure 2 – The AR ASE conveys APDUs between APs

5.7.1.2 Endpoint context

Each AP involved in an AR contains an endpoint of the AR. Each AR endpoint is defined within the AE of the AP. An endpoint definition, when combined with the definitions of the

other endpoints, defines an AR. To ensure communication compatibility among or between endpoints, each endpoint definition contains a set of compatibility-related characteristics.

These characteristics need to be configured appropriately for each endpoint for the AR to operate properly.

Endpoint definitions also contain a set of characteristics that describe the operation of the AR. These characteristics, when combined with those used to specify compatibility, define the context of the endpoint. The endpoint context is used by the AR ASE to manage the operation of the endpoint and the conveyance of APDUs. The characteristics that comprise the endpoint context are described in the next section.

5.7.1.2.1 Endpoint role

The role of an AREP determines the permissible behaviour of an AP at the AREP. An AREP role may be that of a client, server, peer (client and/or server), publisher or subscriber.

Table 18 and Table 19 summarize the characteristics and combinations of each of the AREP roles.

Table 18 – Conveyance of service primitives by AREP role

	Client	Server	Peer	Publisher	Subscriber
Send Service Req	X		X	X	
Recv Service Req		X	X		X
Send Service Rsp		X	X		
Recv Service Rsp	X		X		

Table 19 – Valid combinations of AREP roles involved in an AR

	Client	Server	Peer	Publisher	Subscriber
Client		X	X		
Server	X		X		
Peer	X	X	X		
Publisher					X
Subscriber				X	

5.7.1.2.2 Dedicated AR endpoints

Dedicated AR endpoints provide their services direct to the FAL User. Although their behaviour is the same as that of non-dedicated endpoints, the APDUs they convey contain no service-specific protocol control information other than the AR control field.

FAL Users configured for dedicated ARs construct and transfer their data without using the services of the other FAL ASEs. The format and contents of the user data conveyed over dedicated ARs are known only through configuration.

5.7.1.2.3 Cardinality

The cardinality of an AR specifies, from the point of view of a client or publisher endpoint, how many remote application processes are involved in an AR. Cardinality is never expressed from the viewpoint of a server or a subscriber.

When expressed from the viewpoint of a client or peer endpoint, ARs are always one-to-one. Clients are never capable of issuing a request to multiple servers and waiting for responses from them.

When expressed from the viewpoint of a publisher endpoint, the cardinality indicates that multiple subscribers are supported. These one-to-many ARs provide communications between one application and a group of (one or more) applications. They are often referred to as multi-party and multicast.

In one-to-many ARs, the publisher endpoint identifies the remote endpoints using a group identifier, rather than identifying each one individually as is done with one-to-one ARs. This permits subscribers to join and leave ARs without disrupting the publisher endpoint context because their individual identities are not known to the publisher endpoint. However, the publisher application may be aware of their participation, but that information is not part of the AR endpoint context.

5.7.1.2.4 Conveyance model

The conveyance model defines how APDUs are sent between endpoints of an AR. Three characteristics are used to define these transfers:

- conveyance paths,
- trigger policy, and
- conveyance policy.

a) Conveyance paths

The purpose of AR ASEs is to transfer information between AR endpoints. This information transfer occurs over the conveyance paths of an AR. A conveyance path represents a one-way communication path used by an endpoint for input or output.

To support the role of the application process, the endpoint is configured with either one or two conveyance paths. Endpoints that only send or only receive are configured with either a send or receive conveyance path, respectively, and those that do both are configured with both. ARs with a single conveyance path are called unidirectional, and those with two conveyance paths are called bi-directional.

Unidirectional ARs are capable of conveying service requests only. To convey service responses, a bi-directional AR is necessary. Therefore, unidirectional ARs support the transfer of unconfirmed services in one direction only, while bi-directional ARs support the transfer of unconfirmed and confirmed services initiated by only one endpoint, or by both endpoints.

b) Trigger policy

Trigger policy indicates when APDUs are transmitted over the network by the data-link layer.

The first type is referred to as user-triggered. User-triggered AREPs submit FAL APDUs to the data-link layer for transmission at the earliest opportunity.

The second type is referred to as network-scheduled. Network scheduled AREPs submit FAL APDUs to the data-link layer for transmission according to a schedule configured by management.

c) Conveyance policy

Conveyance policy indicates whether APDUs are transferred according to a buffer model or a queue model. These models describe the method of conveying APDUs from the sender to the receiver.

Buffered ARs contain conveyance paths that have a single buffer at each endpoint. Updates to the source buffer are conveyed to the destination buffer according to the trigger policy of the AR. Updates to either buffer replace its contents with new data. In buffered ARs,

unconveyed or undelivered data that have been replaced are lost. In addition, a data contained in a buffer may be read multiple times without destroying its contents.

Queued ARs contain conveyance paths that are represented as a queue between endpoints. Queued ARs convey data using a FIFO queue. Queued ARs are not overwritten, meaning that new entries are queued until they can be conveyed and delivered.

If a queue is full, a new message will not be placed into the queue.

5.7.1.3 AR establishment

For an AR endpoint to be used by an application process, the corresponding AR must be active. When an AR is activated, it is referred to as “established”.

AR establishment can occur in one of three ways.

First, ARs can be pre-established, meaning that the AE that maintains the endpoint context is created when the AP is connected to the network. In this case, communications among the applications involved in the AR may take place without first having to explicitly establish the AR. Any AR may be defined as being pre-established.

Second, ARs can be pre-defined, but not pre-established. Pre-defined means that the characteristics of the AR are known to each endpoint, but that their contexts have not been synchronized for operation. In this case, the use of the FAL Establish service is required to synchronize them and open them for data transfer.

Third, ARs can require dynamic definition and establishment. In this case, definitions may be created for each of the AREPs using the Create service. After creation, they are established using the Establish service if they were not created as “established”.

5.7.2 Application relationship class specification

5.7.2.1 Formal model

The AR endpoint formal model defines the characteristics common to all AR endpoints. This class is not capable of being instantiated. It is present only for the inheritance of its attributes and services by its subclasses, each specified in a separate subclause of this standard.

All AR endpoint attributes are accessible through system Management.

FAL ASE:			AR ASE
CLASS:			AR ENDPOINT
CLASS ID:			not used
PARENT CLASS:			TOP
ATTRIBUTES:			
1	(m)	Attribute:	FAL Revision
2	(m)	Attribute:	Dedicated (TRUE, FALSE)
3	(m)	Attribute:	Cardinality (one-to-one, one-to-many)
4	(m)	Attribute:	Conveyance policy (Queued, Buffered)
5	(m)	Attribute:	Conveyance path (Unidirectional, Bi-directional)
6	(m)	Attribute:	Trigger policy (User-triggered, Network-scheduled)
7	(o)	Attribute:	Transfer Syntax
SERVICES:			
1	(o)	OpsService:	AR-Unconfirmed Send
2	(o)	OpsService:	AR-Confirmed Send
3	(o)	OpsService:	AR-Establish
4	(o)	OpsService:	AR-Abort

5.7.2.2 Attributes

FAL revision

This specifies the revision level of the FAL protocol used by this endpoint. The revision level is in the AR header of all FAL-PDUs transmitted.

Dedicated

This attribute specifies whether the endpoint is dedicated or not. When TRUE, the services of the AR ASE are accessed directly by the FAL User.

Cardinality

This attribute specifies cardinality of the AR described in 2.7.1.

Conveyance policy

This attribute specifies conveyance policy of the AR described in 2.7.1.

Conveyance path

This attribute specifies conveyance path of the AR described in 2.7.1.

Trigger policy

This attribute specifies trigger policy of the AR described in 2.7.1.

Transfer syntax

This optional attribute identifies the encoding rules to be used on the AR. When not present, the default FAL Transfer Syntax of this standard is used.

5.7.2.3 Services

All services defined for this class are optional. When an instance of the class is defined, at least one of these services should be selected.

AR-unconfirmed send

This optional service is used to send an unconfirmed service.

AR-confirmed send

This optional service is used to send a confirmed service.

AR-establish

This optional service is used to establish (open) an AR.

AR-abort

This optional service is used to abort (abruptly terminate) an AR.

5.7.3 Application relationship ASE service specification

5.7.3.1 Supported services

This subclause contains the definitions of services that are unique to this ASE. The services defined for this ASE are

- AR-Unconfirmed Send
- AR-Confirmed Send
- AR-Establish
- AR-Abort

The services AR-Confirmed Send contain the FAL PDU body as part of the Result parameter in the response and confirmation primitives. The FAL PDU body may contain either the positive response or negative response returned by the FAL User transparently to the AR ASE. Therefore, these services have a single Result parameter instead of the separate Result(+) and Result(-) parameters that are commonly used to convey the positive and negative responses returned by the FAL User.

5.7.3.2 AR-Unconfirmed send service

5.7.3.2.1 Service overview

This service is used to send AR-Unconfirmed Send request APDUs for FAL APO ASEs. The AR-Unconfirmed Send service may be requested at either endpoint of a one-to-one bi-directional AR, at the server endpoint of a one-to-one unidirectional AR, or at the publisher endpoint of a one-to-many AR.

5.7.3.2.2 Service primitives

The service parameters for each primitive are shown in Table 20.

Table 20 – AR-Unconfirmed Send

Parameter name	Req	Ind
Argument		
AREP	M	M
Remote DL Address	M	M (=)
FAL Service Type	M	M (=)
FAL APDU Body	M	M (=)

Remote DL address

This parameter specifies the destination DLSAP address in the request and the source DLSAP address in the indication.

FAL service type

This parameter specifies the type of service being conveyed.

FAL APDU body

This parameter specifies the service-dependent body for the APDU.

5.7.3.2.3 Service procedure

The AR-Unconfirmed Send service is a service that operates through a queue.

The requesting FAL ASE submits an AR-Unconfirmed Send request primitive to its AR ASE. The AR ASE builds an AR-Unconfirmed Send request APDU.

The AR ASE queues the APDU for submission to the lower layer.

If the AREP is user-triggered, the AR ASE immediately requests the lower layer to transfer the APDU. If the AR is network-scheduled, the AR ASE requests the DL to transfer the data at the scheduled time. The data-link mapping indicates how the AR ASE coordinates its requests to transmit the data with the data-link layer.

Upon receipt of the AR-Unconfirmed Send request APDU, the receiving AR ASE delivers an AR-Unconfirmed Send indication primitive to the appropriate FAL ASE as indicated by the FAL Service Type Parameter.

5.7.3.3 AR-confirmed send service

5.7.3.3.1 Service overview

This service is used to send confirmed request and response APDUs for FAL APO ASEs. The AR-Confirmed Send service may be requested at Client and Peer endpoints of one-to-one bi-directional ARs.

5.7.3.3.2 Service primitives

The service parameters for each primitive are shown in Table 21.

Table 21 – AR-confirmed send

Parameter name	Req	Ind	Rsp	Cnf
Argument				
AREP	M	M		
Server DL-Address	M	M (=)		
FAL Service Type	M	M (=)		
FAL APDU Body	M	M (=)		
Result				
Client DL-Address			M	M (=)
FAL Service Type			M	M (=)
FAL APDU Body			M	M (=)
NOTE The method by which a confirm primitive is correlated with its corresponding preceding request primitive is a local matter. The method by which a response primitive is correlated with its corresponding preceding indication primitive is a local matter. See 1.2.				

Server DL-address

This parameter specifies the DL-address of the server.

FAL service type

This parameter specifies the type of service being conveyed.

FAL APDU body

This parameter specifies the service-dependent body for the APDU.

Result

This parameter indicates that the service request succeeded or failed.

Client DL-address

This parameter specifies the DL-address of the client.

5.7.3.3.3 Notional service procedure [INFORMATIVE]

NOTE The following notional explanation is a protocol matter and is outside the scope of this abstract service definition. It is included to indicate one what in which these services could be provided by a real protocol.

The AR-Confirmed Send service is a service that operates through a queue.

The requesting FAL ASE submits an AR-Confirmed Send request primitive to its AR ASE. The AR ASE creates a transaction state machine to control the invocation of the service and assigns an InstanceID to that state machine.

The AR ASE builds an AR-Confirmed Send request APDU that includes the InstanceID and queues the APDU for submission to the lower layer, then the AR ASE immediately requests the lower layer to transfer the APDU.

Upon receipt of the AR-Confirmed Send request APDU, the receiving AR ASE creates a transaction state machine to manage the expected response, assigns an independent (second) InstanceID to that state machine, then delivers an AR-Confirmed Send indication primitive to the appropriate FAL ASE as indicated by the FAL Service Type Parameter, with the (second) InstanceID as an extra implementation parameter.

The responding FAL ASE submits a confirmed send response primitive to its AR ASE, identifying the transaction by the InstanceID presented as part of the response. The AR ASE builds a confirmed send response APDU containing the (first) InstanceID of the original requesting APDU.

The AR ASE queues the APDU for submission to the lower layer. In addition, the AR ASE immediately requests the lower layer to transfer the APDU.

Upon receipt of the confirmed send response APDU, the receiving AR ASE uses the (first) InstanceID contained in the response APDU to associate the response with the appropriate state machine and request. It then delivers an AR-Confirmed Send confirmation primitive to the requesting FAL ASE and cancels the associated transaction state machine.

If the timer expires before the sending AR ASE receives the response APDU, the AR ASE delivers an AR-Confirmed Send confirmation(–) primitive to the requesting FAL ASE and cancels the associated transaction state machine.

5.7.3.4 AR-establish service

5.7.3.4.1 Service overview

This confirmed service operates in a pair-wise manner between two AR endpoints to synchronize their contexts and activate them for the transfer of APDUs.

The endpoint context may be created during or prior to establishment through System Management. Once defined, the Set Attributes and Get Attributes services can be used to update and further coordinate endpoint contexts.

5.7.3.4.2 Service primitives

The service parameters for each primitive are shown in Table 22.

Table 22 – AR-establish service

Parameter name	Req	Cnf
Argument		
AREP	M	
Remote DL-Address	M	
Result		
Error Info		C
NOTE The method by which a confirm primitive is correlated with its corresponding preceding request primitive is a local matter. See 1.2.		

AREP

This parameter specifies sufficient information to locally identify the AREP to be established.

Remote DL-address

This parameter identifies the remote DL-address.

Result

This parameter indicates whether the service request succeeded or failed.

5.7.3.4.3 Service procedure

5.7.3.4.3.1 AR establishment

The AR-Establish service causes local AREPs to be established independently.

Upon receipt of an AR-Establish request service primitive, the calling FAL issues an AR-Establish confirmation service primitive to the calling AP indicating whether or not it was able to establish the AREP. If one of the following cases is found, it was not able to establish the AREP:

- a) the requested AREP is not specified within the FAL, or
- b) resources are not available to establish the requested AREP, or
- c) for AREPs supported by a data-link connection, the data-link layer was not able to establish the data-link connection.

5.7.3.5 AR-abort service

5.7.3.5.1 Service overview

The service is used by the FAL User to terminate an AR abruptly. It is always successful; the receiver of an Abort request always aborts the AR. This service may be used on any open AREP, regardless of whether the AR was pre-established or established dynamically using the establish service.

The AR-Abort service is used to instruct the AR ASE to abruptly terminate all activity on an AREP and place it in the closed state. Receipt of an AR-Abort request primitive causes the AR ASE to close the AREP context immediately. The immediate close of each endpoint context causes all outstanding service requests to be cleared. All subsequent service primitives and APDUs received by the AR ASE for the aborted AR are discarded except those of the AR-Establish service.

The AR ASE may also initiate the Abort service if it detects unrecoverable communication failures. In this case, the AR ASE delivers an AR-Abort indication primitive informing the user of the failure and closes the endpoint context.

5.7.3.5.2 Service primitives

The service parameters for each primitive are shown in Table 23.

Table 23 – AR-abort

Parameter name	Req	Ind
Argument		
AREP	M	M
Originator	M	M (=)
Reason Code	M	M (=)
Additional Detail	U	U (=)

Originator

This parameter identifies the originator of the AR-Abort. Possible values are “DLL”, “FAL”, or “FAL-USER”. The value DLL cannot be used in the request primitive.

Reason code

This parameter indicates the reason for the Abort. It may be supplied by either the provider or the user. One reason is defined: AR ASE error. Other reason code values may be supplied by the data-link layer or by the user.

Additional detail

This optional parameter specifies user data that accompanies the indication. When used, the value submitted in the request primitive is delivered unchanged in the indication primitive.

5.7.3.5.3 Service procedure

The Abort service is a service that operates through a queue.

If the user wishes to abort an AR, it submits an Abort request primitive to its FAL AR ASE. If an AR ASE detects an unrecoverable local failure or a communication failure, it delivers an abort indication primitive to the endpoint user.

The FAL AR ASE also transitions the AREP state to CLOSED.

6 ARs

6.1 General

Application Relationships provided are

- point-to-point user-triggered confirmed client/server AREP (PTC-AR);
- point-to-point user-triggered unconfirmed client/server AREP (PTU-AR);
- point-to-point network-scheduled unconfirmed publisher/subscriber AREP (PSU-AR);
- multipoint user-triggered unconfirmed publisher/subscriber AREP (MTU-AR);
- multipoint network-scheduled unconfirmed publisher/subscriber AREP (MSU-AR).

6.2 Point-to-point user-triggered confirmed client/server AREP (PTC-AR)

6.2.1 Overview

This class is defined to support the on-demand exchange of confirmed services between two application processes. Unconfirmed services are not supported by this type of AR. It uses connectionless-mode data-link services for the exchanges. The data-link layer transfers may be either acknowledged or unacknowledged. The behaviour of this class is described as follows. An AR ASE user wishing to convey a request APDU submits it as an AR ASE Service Data Unit to its AREP and the AREP sending the request APDU queues it to its underlying layer for transfer at the next available opportunity.

The AREP receiving the request APDU from its underlying layer queues it for delivery to its AR ASE user in the order in which it was received.

The AREP receiving the request APDU accepts the corresponding response APDU from its AR ASE user and queues it to the underlying layer for transfer.

The AREP that issued the request APDU receives the response APDU from its underlying layer and queues it for delivery to its AR ASE user in the order in which it was received. It also stops its associated service response timer.

The characteristics of this AREP class are summarized as follows:

Roles	Client
	Server
	Peer
Cardinality	One-to-one
Conveyance paths	Bi-directional

Trigger policy	User-triggered
Conveyance policy	Queued

6.2.2 Formal model

FAL ASE:	AR ASE
CLASS:	PTC-AR
CLASS ID:	not used
PARENT CLASS:	AR ENDPOINT
ATTRIBUTES:	
1 (m) Attribute:	Role (CLIENT, SERVER, PEER)
2 (m) Attribute:	AREP State
3 (m) Attribute:	Maximum Outstanding Requests Calling
4 (m) Attribute:	Maximum Outstanding Requests Called
5 (m) Attribute:	Transmit DL Mapping Reference
6 (m) Attribute:	Receive DL Mapping Reference
SERVICES:	
1 (m) OpsService:	Confirmed Send

6.2.3 Attributes

Role

This attribute specifies the role of the AREP. Possible values are

PEER (CLIENT/SERVER)	Endpoints of this type may perform as either clients or servers, or as both simultaneously.
CLIENT	Endpoints of this type issue confirmed service Request-APDUs to servers and receive confirmed service Response-APDUs.
SERVER	Endpoints of this type receive confirmed service Request-APDUs from clients and issue confirmed service Response-APDUs to them.

AREP state

This attribute specifies the state of the AREP. The values for this attribute are specified in part 3 of this document.

Max outstanding requests calling

This conditional attribute indicates the maximum number of responses that the AREP may be expecting from the remote AREP.

Max outstanding requests called

This conditional attribute indicates the maximum number of responses that the AR endpoint may be expecting from its local user.

Transmit DL mapping reference

This attribute provides a reference to the underlying data-link layer mapping for the transmit conveyance path for this AREP. DL mappings for the data-link layer are specified in part 3 of this document.

Receive DL mapping reference

This attribute provides a reference to the underlying data-link layer mapping for the receive conveyance path for this AREP. DL mappings for the data-link layer are specified in part 3 of this document.

6.2.4 Services

Confirmed send

This optional service is used to send a confirmed service on an AR.

6.3 Point-to-point user-triggered unconfirmed client/server AREP (PTU-AR)

6.3.1 Overview

This class is defined to support the on-demand exchange of unconfirmed services between two application processes. It uses connectionless-mode data-link services for the exchanges. The data-link layer transfers may be either acknowledged or unacknowledged. The behaviour of this class is described as follows.

An AR ASE user wishing to convey a request APDU submits it as an AR ASE Service Data Unit to its AREP. The AREP sending the request APDU queues it to its underlying layer for transfer at the next available opportunity.

The AREP receiving the request APDU from its underlying layer queues it for delivery to its AR ASE user in the order in which it was received.

The characteristics of this AREP class are summarized as follows:

Roles	Client Server Peer
Cardinality	One-to-one
Conveyance paths	Unidirectional
Trigger policy	User-triggered
Conveyance policy	Queued

6.3.2 Formal model

FAL ASE:	AR ASE
CLASS:	Point-to-point user-Triggered Unconfirmed client/server AREP
CLASS ID:	not used
PARENT CLASS:	AR ENDPOINT
ATTRIBUTES:	
1 (m) Attribute:	Role (CLIENT, SERVER, PEER)
2 (m) Attribute:	AREP State
3 (m) Attribute:	Transmit DL Mapping Reference
4 (m) Attribute:	Receive DL Mapping Reference
SERVICES:	
1 (m) OpsService:	Unconfirmed Send

6.3.3 Attributes

Role

This attribute specifies the role of the AREP. Possible values are

PEER (CLIENT/SERVER)	Endpoints of this type may perform as either clients or servers, or as both simultaneously. Endpoints of this type indicate whether they are to be the initiator of the AR establishment process.
CLIENT	Endpoints of this type issue unconfirmed service Request-APDUs to servers.
SERVER	Endpoints of this type receive confirmed and unconfirmed service Request-APDUs from clients.

AREP state

This attribute specifies the state of the AREP. The values for this attribute are specified in IEC61158-6-17.

Transmit DL mapping reference

This attribute provides a reference to the underlying data-link layer mapping for the transmit conveyance path for this AREP. DL mappings for the data-link layer are specified in IEC61158-6-17.

Receive DL mapping reference

This attribute provides a reference to the underlying data-link layer mapping for the receive conveyance path for this AREP. DL mappings for the data-link layer are specified in IEC61158-6-17.

6.3.4 Services**Unconfirmed send**

This optional service is used to send an unconfirmed service on an AR.

6.4 Point-to-point network-scheduled unconfirmed publisher/subscriber AREP (PSU-AR)**6.4.1 Overview**

This class is defined to support the “push” model for scheduled and buffered distribution of unconfirmed services to one application process.

The behaviour of this type of AR can be described as follows.

An AR ASE user wishing to convey a request APDU submits it as an AR ASE Service Data Unit to its AREP for distribution. Sending AREP writes the APDU into the internal buffer, completely replacing the existing contents of the buffer. The AR ASE transfers the buffer contents at the next scheduled transfer opportunity.

If the AREP receives another APDU before the buffer contents are transmitted, the buffer contents will be replaced with the new APDU, and the previous APDU will be lost. When the buffer contents are transmitted, the AR ASE notifies the user of transmission.

At the receiving endpoint, the APDU is received from the network and is written immediately into the buffer, completely overwriting the existing contents of the buffer. The endpoint notifies the user that the APDU has arrived and delivers it to the user according to the local user interface. If the APDU has not been delivered before the next APDU arrives, it will be overwritten by the next APDU and lost.

An FAL user receiving the buffered transmission may request to receive the currently buffered APDU later.

The characteristics of this AREP class are summarized as follows:

Roles	Publisher
	Subscriber
Cardinality	One-to-one
Conveyance paths	Unidirectional
Trigger policy	Network-scheduled
Conveyance policy	Buffered

6.4.2 Formal model

FAL ASE:		AR ASE
CLASS:		Point-to-point network-Scheduled Unconfirmed publisher/subscriber AREP
CLASS ID:		not used
PARENT CLASS:		AR ENDPOINT
ATTRIBUTES:		
1	(m)	Attribute: Role (PUBLISHER, SUBSCRIBER)
2	(m)	Attribute: AREP State
3	(m)	Attribute: Transmit DL Mapping Reference
4	(m)	Attribute: Receive DL Mapping Reference
SERVICES:		
1	(o)	OpsService: Unconfirmed Send

6.4.3 Attributes

Role

This attribute specifies the role of the AREP. Possible values are

- PUBLISHER** Endpoints of this type issue unconfirmed service Request-APDUs to subscribers.
- SUBSCRIBER** Endpoints of this type receive unconfirmed service Request-APDUs issued by the publisher.

AREP state

This attribute specifies the state of the AREP. The values for this attribute are specified in IEC61158-6-17.

Transmit DL mapping reference

This attribute provides a reference to the underlying data-link layer mapping for the transmit conveyance path for this AREP. DL mappings for the data-link layer are specified in IEC61158-6-17.

Receive DL mapping reference

This attribute provides a reference to the underlying data-link layer mapping for the receive conveyance path for this AREP. DL mappings for the data-link layer are specified in IEC61158-6-17.

6.4.4 Services

Unconfirmed send

This optional service is used to send an unconfirmed service on an AR.

6.5 Multipoint user-triggered unconfirmed publisher/subscriber AREP (MTU-AR)

6.5.1 Overview

This class is defined to support the on-demand exchange of unconfirmed services between two or more application processes. This class uses connectionless data-link services for the exchanges. The behaviour of this class is as follows.

An AR ASE user wishing to convey a request APDU submits it as an AR ASE Service Data Unit to its AREP. The AREP sending the request APDU queues it to its underlying layer for transfer at the next available opportunity.

The AREP receiving the request APDU from its underlying layer queues it for delivery to its AR ASE user in the order in which it was received.

The characteristics of this AREP class are summarized as follows:

Roles	Publisher
	Subscriber
Cardinality	One-to-many
Conveyance paths	Unidirectional
Trigger policy	User-triggered
Conveyance policy	Queued

6.5.2 Formal model

FAL ASE:	AR ASE
CLASS:	Multipoint user-Triggered Unconfirmed publisher/subscriber AREP
CLASS ID:	not used
PARENT CLASS:	AR ENDPOINT
ATTRIBUTES:	
1 (m) Attribute:	Role (PUBLISHER, SUBSCRIBER)
2 (m) Attribute:	AREP State
3 (m) Attribute:	Transmit DL Mapping Reference
4 (m) Attribute:	Receive DL Mapping Reference
SERVICES:	
1 (m) OpsService:	Unconfirmed Send

6.5.3 Attributes

Role

This attribute specifies the role of the AREP. Possible values are

PUBLISHER	Endpoints of this type issue unconfirmed service Request-APDUs to subscribers.
SUBSCRIBER	Endpoints of this type receive unconfirmed service Request-APDUs issued by the publisher.

AREP state

This attribute specifies the state of the AREP. The values for this attribute are specified in IEC61158-6-17.

Transmit DL mapping reference

This attribute provides a reference to the underlying data-link layer mapping for the transmit conveyance path for this AREP. DL mappings for the data-link layer are specified in IEC61158-6-17.

Receive DL mapping reference

This attribute provides a reference to the underlying data-link layer mapping for the receive conveyance path for this AREP. DL mappings for the data-link layer are specified in IEC61158-6-17.

6.5.4 Services

Unconfirmed send

This optional service is used to send an unconfirmed service on an AR.

6.6 Multipoint network-scheduled unconfirmed publisher/subscriber AREP (MSU-AR)

6.6.1 Overview

This class is defined to support the “push” model for scheduled and buffered distribution of unconfirmed services to one or more application processes.

The behaviour of this type of AR can be described as follows.

An AR ASE user wishing to convey a request APDU submits it as an AR ASE Service Data Unit to its AREP for distribution. The sending AREP writes the APDU into the internal buffer, completely replacing the existing contents of the buffer.

The AREP transfers the buffer contents at the next scheduled transfer opportunity.

If the AREP receives another APDU before the buffer contents are transmitted, the buffer contents will be replaced with the new APDU, and the previous APDU will be lost. When the buffer contents are transmitted, the AR ASE notifies the user of transmission.

At the receiving endpoint, the APDU is received from the network and is written immediately into the buffer, completely overwriting the existing contents of the buffer. The endpoint notifies the user that the APDU has arrived and delivers it to the user according to the local user interface. If the APDU has not been delivered before the next APDU arrives, it will be overwritten by the next APDU and lost.

An FAL user receiving the buffered transmission may request to receive the currently buffered APDU later.

The characteristics of this AREP class are summarized as follows:

Roles	Publisher Subscriber
Cardinality	One-to-many
Conveyance paths	Unidirectional
Trigger policy	Network-scheduled
Conveyance policy	Buffered

6.6.2 Formal model

FAL ASE:	AR ASE
CLASS:	Multipoint network-Scheduled Unconfirmed publisher/subscriber AREP
CLASS ID:	not used
PARENT CLASS:	AR ENDPOINT
ATTRIBUTES:	
1 (m) Attribute:	Role (PUBLISHER, SUBSCRIBER)
2 (m) Attribute:	AREP State
3 (m) Attribute:	Transmit DL Mapping Reference
4 (m) Attribute:	Receive DL Mapping Reference
SERVICES:	
1 (o) OpsService:	Unconfirmed Send

6.6.3 Attributes

Role

This attribute specifies the role of the AREP. Possible values are

PUBLISHER	Endpoints of this type publish their data issuing unconfirmed service Request-APDUs.
SUBSCRIBER	Endpoints of this type receive unconfirmed service Request-APDUs issued by publishers.

AREP state

This attribute specifies the state of the AREP. The values for this attribute are specified in IEC61158-6-17.

Transmit DL mapping reference

This attribute provides a reference to the underlying data-link layer mapping for the transmit conveyance path for this AREP. DL mappings for the data-link layer are specified in IEC61158-6-17.

Receive DL mapping reference

This attribute provides a reference to the underlying data-link layer mapping for the receive conveyance path for this AREP. DL mappings for the data-link layer are specified in IEC61158-6-17.

6.6.4 Services**Unconfirmed Send**

This optional service is used to send an unconfirmed service on an AR.

7 Summary of FAL classes

This subclause contains a summary of the defined FAL Classes. The Class ID values have been assigned to be compatible with existing standards.

Table 24 provides a summary of the classes.

Table 24 – FAL class summary

FAL ASE	Class
Variable	Variable
	Variable list
Event	Notifier
Load region	Load region
Function invocation	Function invocation
Time	Time
Network management	Network manager
Application relationship	AREP
	PTC-AREP
	PTU-AREP
	PSU-AREP
	MTU-AREP
	MSU-AREP

8 Permitted FAL services by AREP role

Table 25 below defines the valid combinations of services and AR types (which service APDUs) and can be sent or received by AR with the specified type). “Unc” and “Cnf” columns indicate whether the service listed in the left-hand column is unconfirmed or confirmed respectively.

Table 25 – FAL services by AR type

	Unc	Cnf	Client		Server		Publisher		Subscriber	
FAL Services			req	rcv	req	rcv	req	rcv	req	rcv
Variable ASE										
Read		X	X		X					
Write		X	X		X					
Information Report	X				X		X			X
Event ASE										
Event Notification	X						X			X
Notification Recovery	X						X		X	
Load Region ASE										
Download		X	X		X					
Upload		X	X		X					
Function invocation ASE										
Start		X	X		X					
Stop		X	X		X					
Resume		X	X		X					
Time ASE										
Get Network Time		X								
Set Network Time	X						X			X
Tick Notification service	X						X			X
NW Management ASE										
Get Network Status		X								
Get Station Status		X								
NW Status Report	X						X			X
Station Status Report	X						X			X
AR ASE										
PTC-AR		X	X		X					
PTU-AR	X		X		X					
PSU-AR	X		X	X						
MTU-AR	X						X			X
MSU-AR	X						X			X

Bibliography

IEC 61158-6-17, *Industrial communication networks – Fieldbus specifications – Part 6-17: Application layer protocol specification – Type 17 elements*

IEC 61784-1 (Ed.2.0), *Industrial communication networks – Profiles – Part 1: Fieldbus profiles*

IEC 61784-2, *Industrial communication networks – Profiles – Part 2: Additional fieldbus profiles for real-time networks based on ISO/IEC 8802-3*

IECNORM.COM : Click to view the full PDF of IEC 61158-5-17:2007

SOMMAIRE

SOMMAIRE	56
AVANT-PROPOS	58
INTRODUCTION	60
1 Domaine d'application	61
1.1 Vue d'ensemble	61
1.2 Spécifications	62
1.3 Conformité	62
2 Références normatives	62
3 Définitions	63
3.1 Termes et définitions	63
3.2 Abréviations et symboles	69
3.3 Conventions	70
4 Concepts	74
4.1 Généralités	74
4.2 Relations entre les ASE	74
4.3 ASE de FAL	74
4.4 Paramètres communs des services FAL	75
5 ASE	75
5.1 ASE variable	75
5.2 ASE Event	79
5.3 ASE Load region	82
5.4 ASE Function invocation	84
5.5 ASE Time	87
5.6 ASE Network management	90
5.7 ASE Application relationship	93
6 AR	103
6.1 Généralités	103
6.2 Point-to-point user-triggered confirmed client/server AREP (PTC-AR)	104
6.3 Point-to-point user-triggered unconfirmed client/server AREP (PTU-AR)	105
6.4 Point-to-point network-scheduled unconfirmed publisher/subscriber AREP (PSU-AR)	107
6.5 Multipoint user-triggered unconfirmed publisher/subscriber AREP (MTU-AR)	108
6.6 Multipoint network-scheduled unconfirmed publisher/subscriber AREP (MSU-AR)	110
7 Résumé des classes de la FAL	111
8 Services autorisés de FAL par rôle d'AREP	112
Bibliographie	113
Figure 1 – ASE de FAL	75
Figure 2 – L'ASE AR achemine des APDU entre des AP	94
Tableau 1 – Paramètres du service "Read"	77
Tableau 2 – Paramètres du service Write	78

Tableau 3 – Paramètres du service Information report	79
Tableau 4 – Paramètres du service Event notification	81
Tableau 5 – Paramètres du service Event notification recovery.....	82
Tableau 6 – Paramètres du service Download	83
Tableau 7 – Paramètres du service Upload.....	84
Tableau 8 – Paramètres du service Start	85
Tableau 9 – Paramètres du service Stop.....	86
Tableau 10 – Paramètres du service Resume	86
Tableau 11 – Paramètres du service Get network time	88
Tableau 12 – Paramètres du service Set network time.....	89
Tableau 13 – Paramètres du service Tick notification	89
Tableau 14 – Paramètres du service Get network status.....	91
Tableau 15 – Paramètres du service Get station status	91
Tableau 16 – Paramètres du service Network status change report	92
Tableau 17 – Paramètres du service Station status change report	93
Tableau 18 – Acheminement de primitives de services par rôle d'AREP	95
Tableau 19 – Combinaisons valides des rôles d'AREP impliqués dans une AR	95
Tableau 20 – AR-Unconfirmed Send	99
Tableau 21 – AR-confirmed send.....	100
Tableau 22 – Service AR-establish	102
Tableau 23 – AR-abort.....	103
Tableau 24 – Résumé des classes de la FAL	111
Tableau 25 – Services de la FAL par type d'AR	112

COMMISSION ÉLECTROTECHNIQUE INTERNATIONALE

RÉSEAUX DE COMMUNICATION INDUSTRIELS – SPÉCIFICATIONS DES BUS DE TERRAIN –

Partie 5-17: Définition des services de la couche application - Éléments de Type 17

AVANT-PROPOS

- 1) La Commission Électrotechnique Internationale (CEI) est une organisation mondiale de normalisation composée de l'ensemble des comités électrotechniques nationaux (Comités nationaux de la CEI). La CEI a pour objet de favoriser la coopération internationale pour toutes les questions de normalisation dans les domaines de l'électricité et de l'électronique. À cet effet, la CEI – entre autres activités – publie des Normes internationales, des Spécifications techniques, des Rapports techniques, des Spécifications accessibles au public (PAS) et des Guides (ci-après dénommés "Publication(s) de la CEI"). Leur élaboration est confiée à des comités d'études, aux travaux desquels tout Comité national intéressé par le sujet traité peut participer. Les organisations internationales, gouvernementales et non gouvernementales, en liaison avec la CEI, participent également aux travaux. La CEI collabore étroitement avec l'Organisation Internationale de Normalisation (ISO), selon des conditions fixées par accord entre les deux organisations.
- 2) Les décisions ou accords officiels de la CEI concernant les questions techniques représentent, dans la mesure du possible, un accord international sur les sujets étudiés, étant donné que les Comités nationaux de la CEI intéressés sont représentés dans chaque comité d'études.
- 3) Les Publications de la CEI se présentent sous la forme de recommandations internationales et sont agréées comme telles par les Comités nationaux de la CEI. Tous les efforts raisonnables sont entrepris afin que la CEI s'assure de l'exactitude du contenu technique de ses publications; la CEI ne peut pas être tenue responsable de l'éventuelle mauvaise utilisation ou interprétation qui en est faite par un quelconque utilisateur final.
- 4) Dans le but d'encourager l'uniformité internationale, les Comités nationaux de la CEI s'engagent, dans toute la mesure possible, à appliquer de façon transparente les Publications de la CEI dans leurs publications nationales et régionales. Toutes divergences entre toutes Publications de la CEI et toutes publications nationales ou régionales correspondantes doivent être indiquées en termes clairs dans ces dernières.
- 5) La CEI n'a prévu aucune procédure de marquage valant indication d'approbation et n'engage pas sa responsabilité pour les équipements déclarés conformes à une de ses Publications.
- 6) Tous les utilisateurs doivent s'assurer qu'ils sont en possession de la dernière édition de cette publication.
- 7) Aucune responsabilité ne doit être imputée à la CEI, à ses administrateurs, employés, auxiliaires ou mandataires, y compris ses experts particuliers et les membres de ses comités d'études et des Comités nationaux de la CEI, pour tout préjudice causé en cas de dommages corporels et matériels, ou de tout autre dommage de quelque nature que ce soit, directe ou indirecte, ou pour supporter les coûts (y compris les frais de justice) et les dépenses découlant de la publication ou de l'utilisation de cette Publication de la CEI ou de toute autre Publication de la CEI, ou au crédit qui lui est accordé.
- 8) L'attention est attirée sur les références normatives citées dans cette publication. L'utilisation de publications référencées est obligatoire pour une application correcte de la présente publication.
- 9) L'attention est attirée sur le fait que certains des éléments de la présente Publication de la CEI peuvent faire l'objet de droits de propriété intellectuelle ou de droits analogues. La CEI ne saurait être tenue pour responsable de ne pas avoir identifié de tels droits de propriété et de ne pas avoir signalé leur existence.

NOTE L'utilisation de certains des types de protocoles associés est limitée par les détenteurs de leurs droits de propriété intellectuelle. Dans tous les cas, l'engagement à un abandon limité des droits de propriété intellectuelle pris par les détenteurs de ces droits permet d'utiliser un type particulier de protocole de couche liaison de données avec des protocoles de couche physique et de couche application dans des combinaisons de types telles que spécifiées de façon explicite dans la série CEI 61784. L'utilisation des divers types de protocoles dans d'autres combinaisons peut exiger la permission donnée par les détenteurs respectifs de leurs droits de propriété intellectuelle.

La Norme internationale CEI 61158-5-17 a été établie par le sous-comité 65C: Communications numériques, du comité d'études 65 de la CEI: Mesure et commande dans les processus industriels.

La présente première édition et ses parties connexes de la sous-série CEI 61158-5 annulent et remplacent la CEI 61158-5:2003. Cette édition de la présente partie constitue un

supplément technique. La présente partie et ses parties compagnes Type 17 annulent et remplacent la CEI/PAS 62405.

La présente édition de la CEI 61158-5 inclut les modifications majeures suivantes par rapport à l'édition précédente:

- a) suppression du précédent bus de terrain Type 6 en raison du manque de pertinence commerciale;
- b) ajout de nouveaux types de bus de terrain;
- c) division de la Partie 5 de la troisième édition en plusieurs parties numérotées -5-2, -5-3, ...

La présente version bilingue (2014-06) correspond à la version anglaise monolingue publiée en 2007-12.

Le texte anglais de cette norme est issu des documents 65C/475/FDIS et 65C/486/RVD.

Le rapport de vote 65C/486/RVD donne toute information sur le vote ayant abouti à l'approbation de cette norme.

La version française de cette norme n'a pas été soumise au vote.

Cette publication a été rédigée selon les Directives ISO/IEC, Partie 2.

Le comité a décidé que le contenu de cette publication ne sera pas modifié avant la date de maintenance indiquée sur le site web de la CEI sous <http://webstore.CEI.ch> dans les données relatives à la publication recherchée. À cette date, la publication sera:

- reconduite;
- supprimée;
- remplacée par une édition révisée, ou
- amendée.

NOTE La révision de la présente norme sera synchronisée avec les autres parties de la série CEI 61158.

La liste de toutes les parties de la série CEI 61158, publiées sous le titre général *Réseaux de communication industriels – Spécifications de bus de terrain*, est disponible sur le site web de la CEI.

INTRODUCTION

La présente partie de la CEI 61158 fait partie d'une série élaborée pour faciliter l'interconnexion des composants de systèmes d'automatisation. Elle est liée aux autres normes de la série telle que définie par le modèle de référence de bus de terrain "à trois couches" décrit dans la CEI/TR 61158-1.

Le service application est fourni par le protocole d'application utilisant les services disponibles de la liaison de données ou autre couche immédiatement inférieure. La présente norme définit les caractéristiques de services d'application pouvant être exploitées par les applications de bus de terrain et/ou la gestion de système.

Dans toute la série de normes relatives aux bus de terrain, le terme "service" se réfère à la capacité abstraite fournie par une couche du Modèle de référence de base de l'Interconnexion des systèmes ouverts (OSI) à la couche immédiatement supérieure. Ainsi, le service de la couche application défini dans la présente norme est un service architectural conceptuel, indépendant des divisions administratives et de mise en œuvre.

IECNORM.COM : Click to view the full PDF of IEC 61158-5-17:2007

RÉSEAUX DE COMMUNICATION INDUSTRIELS – SPÉCIFICATIONS DES BUS DE TERRAIN –

Partie 5-17: Définition des services de la couche application - Éléments de Type 17

1 Domaine d'application

1.1 Vue d'ensemble

La Couche application de bus de terrain (FAL, Fieldbus Application Layer) fournit aux programmes d'utilisateur un moyen d'accéder à l'environnement de communication du bus de terrain. À cet égard, la FAL peut être considérée comme une "fenêtre entre programmes d'application correspondants".

La présente norme donne les éléments communs visant à assurer les communications de messagerie de base à temps critique et à temps non critique entre des programmes d'application dans un environnement et des équipements d'automatisation spécifiques au bus de terrain de Type 17. Le terme "à temps critique" sert à représenter la présence d'une fenêtre temporelle, dans les limites de laquelle une ou plusieurs actions spécifiées doivent être parachevées avec un certain niveau défini de certitude. Le manquement à parachever les actions spécifiées dans les limites de la fenêtre temporelle risque d'entraîner la défaillance des applications qui demandent ces actions, avec le risque concomitant pour l'équipement, l'installation et éventuellement pour la vie humaine.

La présente norme définit de manière abstraite le service visible de l'extérieur fourni par les différents Types de la Couche Application de bus de terrain en termes:

- a) d'un modèle abstrait pour définir des ressources (objets) d'application capables d'être manipulées par les utilisateurs par l'intermédiaire de l'utilisation du service FAL,
- b) des actions et événements primitifs du service;
- c) des paramètres associés à chaque action primitive et événement primitif, et la forme qu'ils prennent; et
- d) de l'interrelation entre ces actions et événements, et leurs séquences valides.

Le but de la présente norme est de définir les services fournis à:

- 1) l'utilisateur de FAL à la frontière entre l'utilisateur et la Couche Application du modèle de référence de bus de terrain; et
- 2) la Gestion des systèmes au niveau de la frontière entre la Couche Application et la Gestion des systèmes selon le Modèle de référence de bus de terrain.

La présente norme spécifie la structure et les services de la Couche Application des bus de terrain CEI, en conformité avec le Modèle de référence de base de l'OSI (ISO/CEI 7498) et la Structure de la couche application de l'OSI (ISO/CEI 9545).

Les services et protocoles de couche FAL sont fournis par des entités d'application (AE, "Application Entity") de la FAL contenues dans les processus d'application. L'AE de la FAL se compose d'un jeu d'éléments de service application (ASE, "Application Service Element") orientés objet et d'une entité de gestion de couche (LME, "Layer Management Entity") qui gère l'AE. Les ASE fournissent des services de communication qui fonctionnent sur un jeu de classes d'objets de processus d'application (APO, "Application Process Object") connexes. L'un des ASE de la FAL est un ASE de gestion qui fournit un jeu commun de services pour la gestion des instances de classes de la FAL.

Bien que ces services spécifient, du point de vue des applications, la manière dont la demande et les réponses sont émises et délivrées, ils n'incluent pas de spécification concernant ce que les applications qui demandent et répondent doivent en faire. À savoir, les aspects comportementaux des applications ne sont pas spécifiés; seule une définition des demandes et réponses qu'elles peuvent envoyer/recevoir est spécifiée. Cela offre aux utilisateurs de la FAL une plus grande flexibilité pour normaliser le comportement des objets concernés. En plus de ces services, certains services de soutien sont également définis dans la présente norme pour fournir l'accès à la FAL afin de permettre de maîtriser certains aspects de son fonctionnement.

1.2 Spécifications

L'objectif principal de la présente norme est de spécifier les caractéristiques des services conceptuels d'une couche application qui sont adaptés à des communications à temps critique, et donc de compléter le Modèle de référence de base de l'OSI en guidant le développement des protocoles de couche application pour les communications à temps critique.

Un objectif secondaire est de fournir des trajets de migration à partir de protocoles de communication industriels préexistants. C'est ce dernier objectif qui donne naissance à la diversité des services normalisés, tels que les divers Types de la CEI 61158.

La présente spécification peut être utilisée comme base pour les Interfaces de programmation d'applications (API, "Application Programming-Interfaces") formelles. Néanmoins, elle ne constitue pas une interface de programmation formelle et une interface de ce type aura besoin de traiter de questions relatives à la mise en œuvre qui ne sont pas couvertes par la présente spécification, notamment:

- a) les tailles et l'ordonnancement des octets pour les divers paramètres de service à plusieurs octets; et
- b) la corrélation de primitives appariées "request-confirm" (c'est-à-dire: demande et confirmation) ou "indication-réponse" (c'est-à-dire: indication et réponse).

1.3 Conformité

La présente norme ne spécifie ni ne impose de mises en œuvre individuelles ou de produits individuels ni n'impose de contraintes relatives aux mises en œuvre d'entités de couche application au sein des systèmes d'automatisation industriels.

Il n'y a pas de conformité d'équipement à la présente norme de définition des services de couche application. Au contraire, la conformité s'obtient par la mise en œuvre de protocoles de couche application conformes qui satisfont au Type 17 de service de couche application défini dans la présente norme.

2 Références normatives

Les documents de référence suivants sont indispensables pour l'application du présent document. Pour les références datées, seule l'édition citée s'applique. Pour toutes les autres références non datées, la dernière édition du document de référence s'applique (y compris les éventuels amendements).

CEI/TR 61158-1 (Ed.2.0), *Réseaux de communication industriels – Spécifications des bus de terrain – Partie 1: Présentation et lignes directrices des séries CEI 61158 et CEI 61784*

ISO/CEI 7498 (toutes les parties), *Technologies de l'information – Interconnexion de systèmes ouverts (OSI) – Modèle de référence de base*

ISO/CEI 9545, *Technologies de l'information – Interconnexion de systèmes ouverts (OSI) – Structure de la couche application*

ISO/CEI 10731, *Technologies de l'information – Interconnexion de systèmes ouverts (OSI) – Modèle de référence de base – Conventions pour la définition des services OSI*

3 Définitions

Pour les besoins du présent document, les termes et définitions suivants s'appliquent.

3.1 Termes et définitions

3.1.1 Termes de l'ISO/CEI 7498-1

Pour les besoins du présent document, les termes suivants tels que définis dans l'ISO/CEI 7498-1 s'appliquent:

- a) entité d'application
- b) unité de données de protocole application
- c) élément de service application

3.1.2 Termes de l'ISO/CEI 10731

- a) connexion (N)
- b) entité (N)
- c) couche (N)
- d) service (N)
- e) (N)-service-access-point (point d'accès à des services (N))

3.1.3 Autres termes et définitions

3.1.3.1

application

fonction ou structure de données pour laquelle les données sont consommées ou produites

3.1.3.2

processus d'application

partie d'une application distribuée sur un réseau, située sur un appareil et associée à une adresse non ambiguë

3.1.3.3

relation d'applications

association coopérative entre deux ou plusieurs invocations d'entités d'application (application-entity-invocation) à des fins d'échange d'informations et de coordination de leur fonctionnement conjoint

NOTE Cette relation est activée soit par l'échange d'unités de données de protocole d'application, soit suite à des activités de préconfiguration.

3.1.3.4

point d'extrémité de relation d'applications

contexte et comportement d'une relation d'applications tels que vus et maintenus par l'un des processus d'application impliqués dans la relation d'applications

NOTE Chaque processus d'application impliqué dans la relation d'applications maintient son propre point d'extrémité de relation d'applications.

3.1.3.5

attribut

description d'une caractéristique visible en externe ou d'une fonction d'un objet

NOTE Les attributs d'un objet contiennent des informations relatives aux parties variables d'un objet. En règle générale, ils donnent des informations d'état ou régissent le fonctionnement d'un objet. Les attributs peuvent également avoir un impact sur le comportement d'un objet. Les attributs sont divisés en attributs de classe et en attributs d'instance.

3.1.3.6

comportement

indication de la manière dont un objet réagit à des événements particuliers

3.1.3.7

pont

matériel intermédiaire qui relie deux ou plus de deux segments utilisant une fonction relais de couche liaison de données

3.1.3.8

voie

simple liaison physique ou logique d'un objet d'application d'entrée ou de sortie d'un serveur au processus

3.1.3.9

classe

ensemble d'objets, qui représentent tous le même type de composant système

NOTE Une classe est une généralisation d'un objet; un modèle pour définir des variables et des méthodes. Tous les objets dans une classe ont une forme et un comportement identiques, mais contiennent en général des données différentes dans leurs attributs.

3.1.3.10

client

- a) objet qui utilise les services d'un autre objet (serveur) pour accomplir une tâche
- b) initiateur d'un message auquel un serveur réagit

3.1.3.11

connexion

liaison logique entre deux objets d'application qui peuvent se trouver au sein du même appareil ou dans des appareils différents

NOTE 1 Les connexions peuvent être soit point à point, soit multipoint.

NOTE 2 La liaison logique entre puits et source d'attributs et de services en différentes interfaces personnalisées des ASE RT-Auto est appelée "interconnexion". Il existe une distinction entre interconnexions de données et d'événements. La liaison logique et le flux de données entre puits et source d'éléments de données d'automatisation sont appelés "interconnexion de données". La liaison logique et le flux de données entre puits (méthode) et source (événement) de services opérationnels sont appelés "interconnexion d'événements".

3.1.3.12

point de connexion

tampon qui est représenté comme étant une sous-instance d'un objet Assembly (Ensemble)

3.1.3.13

trajet d'acheminement

flux unidirectionnel d'unités APDU à travers une relation d'applications

NOTE L'abréviation « APDU » est dérivée du terme anglais développé correspondant « application protocol data unit ».

3.1.3.14

AR dédiée

AR employée directement par l'utilisateur FAL

NOTE 1 Dans le cas des AR dédiées, seuls l'en-tête FAL et les données utilisateur sont transférés.

NOTE 2 L'abréviation « AR » est dérivée du terme anglais développé correspondant «application relationship». L'abréviation « FAL » est dérivée du terme anglais développé correspondant «Fieldbus application layer».

3.1.3.15

appareil

équipement matériel physique relié à la liaison

NOTE Un appareil peut contenir plus d'un nœud.

3.1.3.16

domaine

partie du réseau RTE composée d'un ou de deux sous-réseau(x)

NOTE 1 Deux sous-réseaux sont nécessaires à la composition d'un réseau RTE à double redondance, chaque nœud d'extrémité du domaine étant connectée à l'un ou l'autre des sous-réseaux.

NOTE 2 L'abréviation « RTE » est dérivée du terme anglais développé correspondant «real-time Ethernet».

3.1.3.17

maître de domaine

station qui accomplit le diagnostic de routes allant vers tous les autres domaines, la distribution du temps réseau aux nœuds à l'intérieur du domaine, l'acquisition du temps absolue auprès du maître d'horloge réseau et la notification du statut du domaine

3.1.3.18

numéro de domaine

identificateur numérique qui indique un domaine

3.1.3.19

nœud d'extrémité

nœud producteur ou consommateur

3.1.3.20

point d'extrémité

l'une des entités en communication impliquées dans une connexion

3.1.3.21

erreur

divergence entre une valeur ou condition calculée, observée ou mesurée et la valeur ou condition spécifiée ou théoriquement correcte

3.1.3.22

pont externe

pont auquel ne sont connectés directement ni les ponts internes ni les stations RTE

NOTE L'abréviation « RTE » est dérivée du terme anglais développé correspondant «real-time Ethernet».

3.1.3.23

événement

instance d'un changement de conditions

3.1.3.24

groupe

a) <généralités> terme général pour un ensemble d'objets. Usages spécifiques:

b) <adressage> lors de la description d'une adresse, adresse qui identifie plus d'une entité

3.1.3.25

interface

a) frontière commune entre deux unités fonctionnelles, définie par des caractéristiques fonctionnelles, des caractéristiques de signal ou d'autres caractéristiques appropriées

- b) ensemble d'attributs de classe FAL et de services qui représente une vue spécifique sur la classe FAL

NOTE L'abréviation « FAL » est dérivée du terme anglais développé correspondant «Fieldbus application layer».

3.1.3.26

port d'interface

point de connexion physique d'un nœud d'extrémité, qui a une adresse DL indépendante

3.1.3.27

pont interne

pont auquel ne sont directement connectés aucun routeur, aucun pont externes, aucun nœud qui ne sont pas conformes à la présente spécification

3.1.3.28

invocation

acte consistant à utiliser un service ou une autre ressource d'un processus d'application

NOTE Chaque invocation représente un processus distinct de commande qui peut être décrit par son contexte. Une fois que le service s'achève ou que l'utilisation de la ressource est libérée, l'invocation cesse d'exister. Pour des invocations de service, un service qui a été lancé, mais n'est pas encore achevé, s'appelle "invocation de service en cours".

3.1.3.29

pont de jonction

pont auquel est connecté au moins un routeur, un pont externe ou nœud qui ne sont pas conforme à la présente spécification et auquel au moins un pont interne ou une station RTE est connecté(e)

NOTE L'abréviation « RTE » est dérivée du terme anglais développé correspondant «real-time Ethernet».é

3.1.3.30

liaison

voie de communication physique entre deux nœuds

3.1.3.31

méthode

<objet> synonyme d'un service opérationnel qui est fourni par l'ASE serveur et invoqué par un client

NOTE L'abréviation « ASE » est dérivée du terme anglais développé correspondant «application service element».

3.1.3.32

réseau

ensemble de nœuds reliés par un support de communication d'un type ou d'un autre, avec d'éventuels répéteurs, ponts, routeurs et passerelles de couche inférieure intermédiaires

3.1.3.33

maître d'horloge réseau

station qui distribue le temps réseau aux maîtres de domaine

3.1.3.34

nœud

entité DLL unique, telle qu'elle apparaît dans une liaison locale

3.1.3.35

nœud d'interface non redondant

nœud qui a un seul port d'interface

3.1.3.36**station non redondante**

station constituée d'un nœud d'extrémité unique

NOTE "station non redondante" est synonyme de "nœud d'extrémité".

3.1.3.37**objet**

représentation abstraite d'un composant particulier au sein d'un appareil, habituellement un ensemble de données connexes (sous la forme de variables) et de méthodes (procédures) pour effectuer des opérations sur les données en question qui ont une interface et un comportement clairement définis

3.1.3.38**auteur**

client chargé d'établir un chemin de connexion vers la cible

3.1.3.39**trajet**

voie de communication logique entre deux nœuds, qui est constitué d'une ou deux liaisons

3.1.3.40**homologue**

rôle d'un point d'extrémité d'AR dans lequel il est capable d'agir à la fois comme client et comme serveur

NOTE L'abréviation « AR » est dérivée du terme anglais développé correspondant « application relationship ».

3.1.3.41**producteur**

nœud qui est chargé d'envoyer des données

3.1.3.42**fournisseur**

source d'une connexion de données

3.1.3.43**éditeur; publicateur**

rôle d'un point d'extrémité d'AR qui émet des APDU sur le bus de terrain en vue de leur consommation par un ou plusieurs abonnés

NOTE 1 Un publicateur peut ne pas connaître l'identité des abonnés ou leur nombre et il peut éditer ses APDU en utilisant une AR dédiée.

NOTE 2 L'abréviation « AR » est dérivée du terme anglais développé correspondant « application relationship ».: L'abréviation « APDU » est dérivée du terme anglais développé correspondant « application protocol data unit ».

3.1.3.44**nœud d'interface redondant**

nœud avec deux ports d'interface, l'un de ceux-ci étant connecté à un réseau primaire alors que l'autre est connecté à un réseau secondaire

3.1.3.45**station redondante**

station qui est constitué d'une paire de nœuds d'extrémité

NOTE Chaque nœud d'extrémité d'une station redondante a le même numéro de station, mais a une adresse DL différente.

3.1.3.46

ressource

capacité de traitement ou d'informations d'un sous-système

3.1.3.47

station RTE

station conforme à la présente spécification

NOTE L'abréviation « RTE » est dérivée du terme anglais développé correspondant «real-time Ethernet».

3.1.3.48

chemin

voie de communication logique entre deux nœuds d'extrémité de communication

3.1.3.49

routeur

matériel intermédiaire qui relie deux ou plus de deux sous-réseaux utilisant une fonction relais de couche réseau

3.1.3.50

segment

voie de communication qui relie deux nœuds directement sans ponts intermédiaires

3.1.3.51

serveur

- a) rôle d'un AREP dans lequel il retourne au client qui a lancé la demande une APDU confirmée de réponse de service
- b) objet qui fournit des services à un autre objet (client)

NOTE L'abréviation « AREP » est dérivée du terme anglais développé correspondant «Application relationship end point». L'abréviation « APDU » est dérivée du terme anglais développé correspondant «Application protocol data unit».

3.1.3.52

service

opération ou fonction qu'un objet et/ou une classe d'objets réalise ou assure à la demande d'un autre objet et/ou d'une autre classe d'objets

3.1.3.53

station

nœud d'extrémité ou paire de nœuds d'extrémité qui accomplit une fonction d'application spécifique

3.1.3.54

numéro de station

identificateur numérique qui indique une station RTE

NOTE L'abréviation « RTE » est dérivée du terme anglais développé correspondant «real-time Ethernet».

3.1.3.55

sous-réseau

partie d'un réseau ne contenant pas de routeur. Un sous-réseau est composé de nœuds d'extrémité, de ponts et de segments

NOTE Les adresses de réseau IP de chaque station d'extrémité incluse dans un sous-réseau sont identiques.

3.1.3.56

abonné

rôle d'un AREP dans lequel il reçoit des unités APDU produites par un publicateur

NOTE L'abréviation « AREP » est dérivée du terme anglais développé correspondant «application relationship end point». L'abréviation « APDU » est dérivée du terme anglais développé correspondant «application protocol data unit».

3.2 Abréviations et symboles

3.2.1 Abréviations de l'ISO/CEI 10731

ASE	Application-Service-Element (Élément de service d'application)
OSI	Open systems interconnection (Interconnexion des systèmes ouverts)

3.2.2 Autres abréviations et symboles

AE	Application entity (Entité d'application)
AL	Application layer (Couche application)
AP	Application process (Processus d'application)
APDU	Application protocol data unit (Unité de données de protocole application)
APO	Application object (Objet d'application)
AR	Application relationship (Relation d'applications)
AREP	Application relationship end point (Point d'extrémité de relation d'applications)
Cnf	Confirmation
DL-	Data-link layer (couche liaison de données) (comme préfixe)
DLL	DL-layer (couche DL)
DLSAP	DL-service-access-point (Point d'accès au service DL)
FAL	Fieldbus application layer (Couche application de bus de terrain)
FIFO	First-in first-out (premier entré, premier sorti (méthode de mise en file d'attente))
ID	Identifier (Identificateur)
CEI	International Electrotechnical Commission (Commission Électrotechnique Internationale)
Ind	Indication
ISO	Organisation internationale de normalisation
MSU-AR	Multipoint network-scheduled unconfirmed publisher/subscriber AREP (AREP "éditeur/abonné de services non confirmés programmé par le réseau en mode multipoint")
MTU-AR	Multipoint user-triggered unconfirmed publisher/subscriber AREP (AREP "éditeur/abonné de services non confirmés déclenché par l'utilisateur en mode multipoint")
PDU	Protocol Data Unit (Unité de données de protocole)
PSU-AR	Point-to-point network-scheduled unconfirmed client/server AREP (AREP "Point-to-point network-scheduled unconfirmed client/server", c'est-à-dire AREP "client/serveur de services non confirmés programmé par le réseau en mode multipoint")
PTC-AR	Point-to-point user-triggered confirmed client/server AREP (AREP "Point-to-point user-triggered confirmed client/server"; c'est-à-dire AREP "client/serveur de services confirmés déclenché par l'utilisateur en mode point à point")
PTU-AR	Point-to-point user-triggered unconfirmed client/server AREP (AREP "Point-to-point user-triggered unconfirmed client/server"; c'est-à-dire AREP "client/serveur de services non confirmés déclenché par l'utilisateur en mode point à point")

Req	Request (Demande)
Rsp	Response (Réponse)
SAP	Service Access Point (Point d'accès au service)

3.3 Conventions

3.3.1 Vue d'ensemble

La FAL se compose d'un ensemble d'ASE orientés objet. Chaque ASE est spécifié dans un paragraphe distinct. Chaque spécification d'ASE est constituée de deux parties, à savoir sa spécification de classe et sa spécification de service.

La spécification de classe définit les attributs de la classe. Les attributs sont accessibles à partir d'instances de la classe en utilisant les services d'ASE de gestion d'objets spécifiés à l'Article 5 de la présente norme. La spécification de service définit les services fournis par l'ASE.

3.3.2 Conventions pour les définitions de classes

Les définitions de classes sont décrites à l'aide de modèles. Chaque modèle est constitué d'une liste d'attributs de la classe. La forme générale du modèle est montrée ci-dessous:

ASE de FAL:	Nom de l'ASE
CLASSE:	Nom de la classe
ID de CLASSE:	#
CLASSE PARENTE:	Nom de la classe parente
ATTRIBUTS:	
1 (o) Attribut-clé:	identificateur numérique
2 (o) Attribut-clé:	nom
3 (m) Attribut:	nom d'attribut(valeurs)
4 (m) Attribut:	nom d'attribut(valeurs)
4.1 (s) Attribut:	nom d'attribut(valeurs)
4.2 (s) Attribut:	nom d'attribut(valeurs)
4.3 (s) Attribut:	nom d'attribut(valeurs)
5. (c) Contrainte:	expression de la contrainte
5.1 (m) Attribut:	nom d'attribut(valeurs)
5.2 (o) Attribut:	nom d'attribut(valeurs)
6 (m) Attribut:	nom d'attribut(valeurs)
6.1 (s) Attribut:	nom d'attribut(valeurs)
6.2 (s) Attribut:	nom d'attribut(valeurs)
SERVICES:	
1 (o) OpsService:	nom de service
2. (c) Contrainte:	expression de la contrainte
2.1 (o) OpsService:	nom de service
3 (m) MgtService:	nom de service

- (1) La rubrique "FAL ASE:" est le nom de l'ASE de la FAL qui fournit les services pour la classe spécifiée.
- (2) La rubrique "CLASSE:" est le nom de la classe spécifiée. Tous les objets définis à l'aide de ce modèle sont une instance de cette classe. La classe peut être spécifiée par la présente norme ou par un utilisateur de la présente norme.
- (3) La rubrique "CLASS ID:" ("ID de classe") est un numéro qui identifie la classe spécifiée. Ce numéro est unique au sein du FAL ASE qui fournit les services pour cette classe. Lorsqu'il est qualifié par l'identité de son FAL ASE, il identifie sans ambiguïté la classe relevant du domaine d'application de la FAL. La valeur "NULL" indique que la classe ne

peut pas être instanciée. Les Class ID (identificateurs de classe) entre 1 et 255 sont réservés par la présente norme pour identifier des classes normalisées. Ils ont été attribués pour maintenir la compatibilité avec les normes nationales existantes. Les CLASS ID entre 256 et 2048 sont alloués pour identifier les classes définies par l'utilisateur.

- (4) La rubrique "PARENT CLASS:" ("CLASSE PARENTE") est le nom de la classe parente pour la classe spécifiée. Tous les attributs définis pour la classe parente et hérités par celle-ci sont hérités pour la classe définie, et ils ne doivent donc pas être redéfinis dans le modèle pour cette classe.

NOTE La classe parente "TOP" indique que la classe définie est une définition de classe initiale. La classe parente "TOP" est utilisée comme point de départ à partir duquel toutes les autres classes sont définies. L'usage de "TOP" est réservé pour les classes définies par la présente norme.

- (5) L'étiquette "ATTRIBUTES" (ATTRIBUTS) indique que les entrées suivantes sont des attributs définis pour la classe.

- a) Chacune des entrées d'attribut contient un numéro de ligne dans la colonne 1, un indicateur obligatoire (m) / facultatif (o) / conditionnel (c) / sélecteur (s) dans la colonne 2, une étiquette de type d'attribut dans la colonne 3, un nom ou une expression conditionnelle dans la colonne 4, et, facultativement, une liste de valeurs énumérées dans la colonne 5. Dans la colonne suivant la liste de valeurs, la valeur par défaut pour l'attribut peut être spécifiée.
- b) Les objets sont normalement identifiés par un identificateur numérique et/ou par un nom d'objet. Dans les modèles de classe, ces attributs clés sont définis sous l'attribut clé.
- c) Le numéro de ligne définit la séquence et le niveau d'imbrication de la ligne. Chaque niveau d'imbrication est identifié par un point. L'imbrication est utilisée pour spécifier
 - i) des champs d'un attribut structuré (4.1, 4.2, 4.3),
 - ii) des attributs conditionnés à un énoncé de contrainte (5). Les attributs peuvent être obligatoires (5.1) ou facultatifs (5.2) si la contrainte est vraie. Tous les attributs facultatifs n'exigent pas des énoncés de contraintes comme le fait l'attribut défini en (5.2).
 - iii) des champs sélection d'un attribut de type choix (6.1 et 6.2).

- (6) L'étiquette "SERVICES" indique que les entrées suivantes sont des services définis pour la classe.

- a) Un (m) dans la colonne 2 indique que le service est obligatoire pour la classe, alors qu'un (o) indique qu'il est facultatif. Un (c) dans cette colonne indique que le service est conditionnel. Lorsque tous les services définis pour une classe le sont comme étant facultatifs, l'un au moins doit être sélectionné quand une instance de la classe est définie.
- b) L'étiquette "OpsService" désigne un service opérationnel (1).
- c) L'étiquette "MgtService" désigne un service de gestion (2).
- d) Le numéro de ligne définit la séquence et le niveau d'imbrication de la ligne. Chaque niveau d'imbrication est identifié par un point. L'imbrication dans la liste de services sert à spécifier des services conditionnés à un énoncé de contrainte.

3.3.3 Conventions pour les définitions des services

3.3.3.1 Généralités

La présente norme emploie les conventions de description énoncées dans l'ISO/CEI 10731.

Le modèle de service, les primitives de service et les schémas de temps-séquence utilisés sont des descriptions totalement abstraites; ils ne constituent pas une spécification pour une mise en œuvre.

3.3.3.2 Paramètres du service

Les primitives de service sont utilisées pour représenter les interactions entre utilisateur de service et fournisseur de service (ISO/CEI 10731). Elles acheminent des paramètres qui indiquent des informations disponibles dans l'interaction entre utilisateur et fournisseur.

NOTE 1 Voir la note en 3.3.3.3 relative à la non-inclusion de paramètres de service qui sont appropriés à une spécification de protocole, à une spécification d'interface de programmation ou à une spécification de mise en œuvre, mais pas à une définition de services abstraits.

La présente norme utilise un format de tableau pour décrire les paramètres de composants des primitives du service. Dans tout le reste de la présente norme, les paramètres qui s'appliquent à chaque groupe de primitives de service sont consignés en tableaux. Chaque tableau comporte jusqu'à six colonnes: une colonne pour le nom du paramètre de service et une colonne chacune pour les primitives et les sens de transfert de paramètres utilisés par le service. Les six colonnes possibles sont:

- 1) le nom de paramètre;
- 2) les paramètres d'entrées de la primitive "request";
- 3) les paramètres de sortie de la primitive "request";

NOTE 2 Il s'agit d'une capacité rarement utilisée. Sauf spécification contraire, les paramètres de primitive de demande sont des paramètres d'entrée.

- 4) les paramètres de sortie de la primitive "indication";
- 5) les paramètres d'entrées de la primitive "response"; et
- 6) les paramètres de sortie de la primitive "confirm".

NOTE 3 Les primitives "request", "indication", "response" et "confirm" sont également respectivement appelées primitives "requestor.submit", "acceptor.deliver", "acceptor.submit" et "requestor.deliver" (voir ISO/CEI 10731).

Un paramètre (ou un composant de celui-ci) est énuméré dans chaque rangée de chaque tableau. Dans les colonnes appropriées de la primitive de service, un code est utilisé pour spécifier le type d'usage du paramètre sur la primitive spécifiée dans la colonne:

- M Le paramètre est obligatoire pour la primitive
- U Le paramètre est une option de l'utilisateur et peut ou peut ne pas être fourni, en fonction de l'usage dynamique de l'utilisateur du service. Lorsqu'il n'est pas fourni, une valeur par défaut est supposée pour le paramètre.
- C Le paramètre est conditionné à d'autres paramètres ou à l'environnement de l'utilisateur du service.
- (blanc ou vide) Le paramètre n'est jamais présent.
- S Le paramètre est un élément sélectionné.

Certaines entrées sont en outre qualifiées par des éléments entre parenthèses. Ceux-ci peuvent être:

- a) une contrainte spécifique à un paramètre:
 "(=)" indique que le paramètre équivaut du point de vue de la sémantique au paramètre dans la primitive de service située immédiatement à sa gauche dans le tableau;
- b) une indication qu'une certaine note s'applique à l'entrée:
 "(n)" indique que la note "n" suivante contient des informations complémentaires relatives au paramètre et à son utilisation.

3.3.3.3 Procédures de service

Les procédures sont définies en termes:

- des interactions entre entités d'application par l'échange d'unités de données de protocole d'application de bus de terrain, et

- des interactions entre un fournisseur de service de couche application et un utilisateur de service de couche application dans le même système par l'invocation de primitives de service de couche application.

Ces procédures sont applicables à des instances de communication entre systèmes qui prennent en charge des services de communications à contrainte de temps au sein de la couche application de bus de terrain.

NOTE La série de normes CEI 61158-5 définit des ensembles de services abstraits. Ce ne sont ni des spécifications de protocole ni des spécifications de mises en œuvre ni des spécifications d'interface de programmation concrète. Par conséquent, il existe des restrictions sur la mesure dans laquelle les procédures de service peuvent être mandatées dans les parties de la CEI 61158-5. Les aspects de protocole qui peuvent varier entre les différentes spécifications de protocole ou les différentes mises en œuvre qui instancient les mêmes services abstraits sont inadaptés à une inclusion dans ces définitions de services, excepté au niveau d'abstraction qui est nécessairement commun à toutes ces expressions.

Par exemple, le moyen par lequel les fournisseurs de services appartiennent des PDU de demande et de réponse est approprié pour spécification dans la norme de spécification de protocole CEI 61158-6, mais pas dans la norme de définition de services abstraits CEI 61158-5. De même, les méthodes de mises en œuvre locales par lesquelles un fournisseur de service ou un utilisateur de service apparie des primitives "request" et "confirm(ation)" ou des primitives "indication" et "response" sont appropriées pour une spécification de mises en œuvre ou pour une spécification d'interface de programmation, mais pas pour une norme de services abstraits ou pour une norme de protocoles, excepté à un niveau d'abstraction qui est nécessairement commun à tous les modes de réalisation de la norme de spécification. Dans tous les cas, la définition abstraite n'est pas permise pour spécifier outre mesure la réalisation d'instantiation plus concrète.

De plus amples informations relatives aux procédures de service conceptuelles d'une mise en œuvre d'un protocole qui réalise les services de l'une des définitions de services abstraits selon la CEI 61158-5 peuvent être consultées dans le paragraphe 9.6 de la CEI/TR 61158-1 (Ed.2.0).

IECNORM.COM : Click to view the full PDF of IEC 61158-5-17:2007

4 Concepts

4.1 Généralités

Cette couche application des bus de terrain (FAL) fournit des services de communication aux Applications à temps critique et à temps non critique dans les appareils de bus de terrain.

Les concepts et modèles communs utilisés pour décrire le service de couche application dans la présente norme sont donnés dans le détail à l'Article 9 de la CEI/TR 61158-1.

4.2 Relations entre les ASE

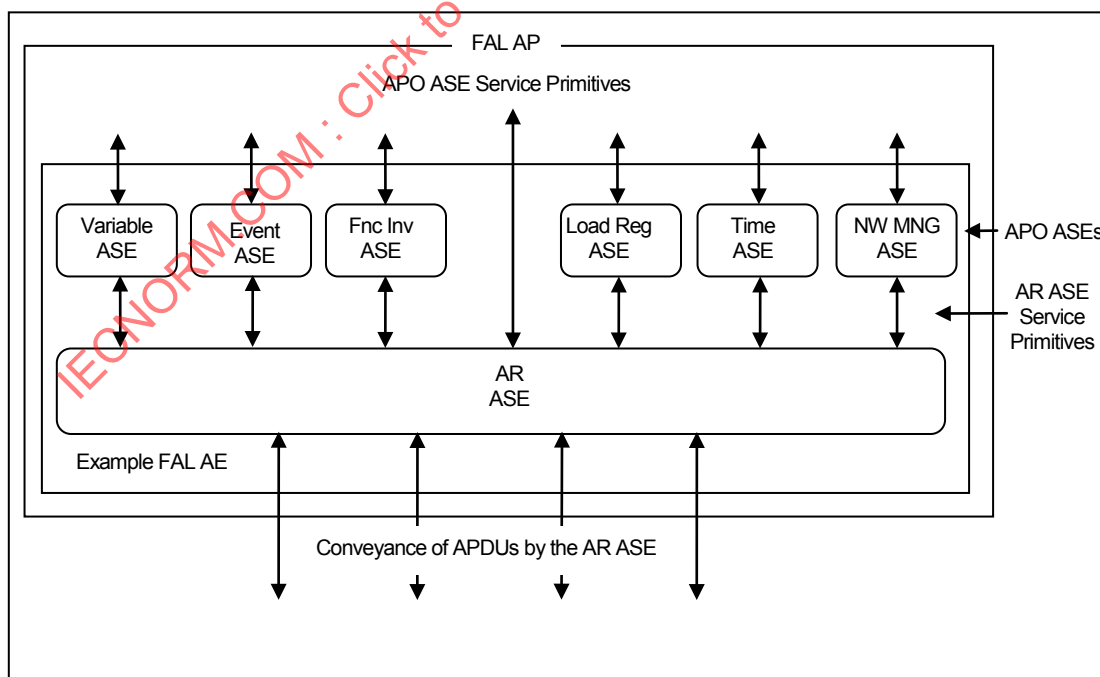
4.3 ASE de FAL

Une approche modulaire a été adoptée pour la définition des ASE de FAL. Les ASE définis pour la FAL sont également orientés objet. En général, les ASE fournissent un ensemble de services conçus pour une classe d'objet spécifique ou pour un ensemble connexe de classes.

L'ASE "Application Relationship" (Relation d'applications) est défini dans le but de prendre en charge l'accès distant à l'AP. Cet ASE "Relation d'applications" fournit des services à l'AP pour définir et établir des relations de communication avec d'autres AP et il fournit également des services à d'autres ASE pour acheminer leurs demandes de services et leurs réponses.

Chaque ASE de FAL définit un ensemble de services, d'unités APDU et de procédures qui effectuent des opérations sur les classes qu'il représente. Seul un sous-ensemble des services de l'ASE peut être fourni pour satisfaire aux besoins d'une application. Des profils peuvent être utilisés pour définir ces sous-ensembles.

Les APDU sont envoyées et reçues entre les ASE de FAL qui prennent en charge le même service. La Figure 1 montre les ASE de FAL dans la perspective de communication.



Légende

Anglais	Français
FAL AP	FAL AP (processus d'application de couche application)

Anglais	Français
	de bus de terrain)
APO ASE Service Primitives	Primitives de services ASE APO
Variable ASE	ASE Variable
Event ASE	ASE Event (Événement)
Fnc Inv ASE	ASE Fnc Inv. (Invocation de fonction)
Load Reg ASE	ASE Load Reg. (Région de chargement)
Time ASE	ASE Time (Temps)
NW MNG ASE	ASE NW MNG. (Gestion de réseau)
APO ASEs	ASE APO (Objet d'application)
AR ASE Service Primitives	Primitives de services ASE AR
AR ASE	ASE AR (Relation d'applications)
Example FAL AE	Exemple d'AE de FAL (entité d'application de FAL)
Conveyance of APDUs by the AR ASE	Acheminement des APDU par l'ASE AR

Figure 1 – ASE de FAL

4.4 Paramètres communs des services FAL

De nombreux services ont les paramètres suivants. Au lieu de les définir pour chaque service, les définitions communes suivantes sont adoptées:

Argument

Ce paramètre contient les paramètres de l'invocation du service.

AREP

Ce paramètre spécifie des informations suffisantes pour l'identification locale de l'AREP qui doit être utilisé pour acheminer le service. Ce paramètre peut utiliser un attribut-clé de l'AREP pour identifier la relation d'applications. Lorsqu'un AREP prend en charge plusieurs contextes simultanément, le paramètre AREP est étendu pour identifier le contexte ainsi que l'AREP.

Result (+)

Ce paramètre de type sélection indique que la demande de service a réussi.

Result(-)

Ce paramètre de type sélection indique que la demande a échoué.

Error info

Ce paramètre fournit des informations d'erreur relatives aux erreurs du service. Il est retourné dans des primitives response(-) de services confirmés.

5 ASE

5.1 ASE variable

5.1.1 Vue d'ensemble

Dans l'environnement de bus de terrain, les processus d'application contiennent des données qui peuvent être tant lues qu'écrites par des applications distantes. L'ASE variable définit les attributs visibles du réseau des données d'application et fournit un ensemble de services utilisés pour lire, écrire, rapporter leurs valeurs.

Lorsque les types appropriés de relation d'applications sont pris en charge, l'ASE variable peut être utilisé pour prendre en charge deux modèles d'accès différents, à savoir le modèle client/serveur et le modèle éditeur/abonné. Le modèle client/serveur est caractérisé par une application client envoyant une demande de lecture ou d'écriture à une application serveur

qui répond en conséquence. L'activité du serveur est stimulée par les clients sur le réseau. S'il n'y a pas de demandes, le serveur ne génère aucune réponse.

Le modèle éditeur/abonné est différent en ce qu'il est caractérisé par un producteur de données éditant ses données sur le réseau. Les abonnés souhaitant acquérir les données éditées rejoignent la relation d'applications utilisée pour les éditer et être à l'écoute des données à mesure qu'elles sont émises. Deux modèles sont fournis pour prendre en charge cette activité éditeur/abonné, à savoir le modèle "pull" et le modèle "push".

Dans le modèle "pull", l'un des abonnés demande que l'éditeur édite une séquence de données variables en produisant une demande d'édition pour celles-ci. L'éditeur distribue périodiquement les données variables en fonction de la demande distante par échange par diffusion multidestinataires ("multicasting"). Le programme d'édition est commandé par le publicateur lui-même.

Dans le modèle "push", il est demandé à l'éditeur de distribuer une séquence de données variables par l'utilisateur FAL local. L'éditeur distribue une séquence de données variables par échange par diffusion multidestinataires ("multicasting") en fonction de la demande locale. Le programme d'édition est également commandé par le publicateur lui-même.

5.1.2 Spécification de la classe Variable

5.1.2.1 Modèle formel

ASE de FAL:		ASE "VARIABLE"
CLASSE:		VARIABLE
ID de CLASSE:		non utilisé
CLASSE PARENTE:		TOP
ATTRIBUTS:		
1	(m) Attribut-clé:	Numeric Identifier
2	(m) Attribut:	Type de données
3	(0) Attribut:	Length
4	(o) Attribut:	Read enable
5	(o) Attribut:	Write enable
SERVICES:		
1	(o) OpsService:	Read
2	(o) OpsService:	Write
3	(o) OpsService:	Information Report

5.1.2.2 Attributs

Numeric identifier

Cet attribut-clé identifie une instance de cette classe d'objet.

Data type

Cet attribut est l'identificateur numérique du type de données.

Length

Cet attribut facultatif est la longueur des données en octets.

Read enable

La valeur de cet attribut facultatif indique si, oui ou non, la valeur des données de l'objet Variable peut être lue par le biais du bus de terrain.

Write enable

La valeur de cet attribut facultatif indique si, oui ou non, la valeur des données de l'objet Variable peut être mise à jour par le biais du bus de terrain.

5.1.3 Spécification de la classe Variable List

5.1.3.1 Modèle formel

ASE de FAL:	ASE "VARIABLE "
--------------------	------------------------

CLASSE:			VARIABLE LIST
ID de CLASSE:			non utilisé
CLASSE PARENTE:			TOP
ATTRIBUTS:			
1	(m)	Attribut-clé::	Numeric Identifier
2	(m)	Attribut:	Number of entries
3	(m)	Attribut:	List Of Variables
SERVICES:			
1	(o)	OpsService:	Read
2	(o)	OpsService:	Write
3	(o)	OpsService:	Information Report

5.1.3.2 Attributs

Numeric identifier

Cet attribut-clé identifie une instance de cette classe d'objet.

Number of entries

Cet attribut spécifie le nombre de variables contenues dans la liste.

List of variables

Cet attribut identifie les variables par les attributs-clés qui sont contenus dans la liste.

5.1.4 Spécification des services de l'ASE Variable

5.1.4.1 Services pris en charge

Le présent paragraphe contient la définition des services qui sont propres à cet ASE. Les services définis pour cet ASE sont:

- Read ("Lecture")
- Write ("Écriture")
- Information Report ("Rapport d'informations")

5.1.4.2 Service "Read"

5.1.4.2.1 Vue d'ensemble du service

Ce service confirmé peut être utilisé pour lire la valeur d'un objet "variable" ou d'un objet "variable list" (liste de variables). Il n'est pas utilisé avec le modèle éditeur/abonné.

5.1.4.2.2 Primitives du service

Les paramètres de service pour chaque primitive sont montrés dans le Tableau 1.

Tableau 1 – Paramètres du service "Read"

Nom du paramètre	Req	Ind	Rsp	Cnf
Argument				
AREP	M	M		
Variable Specifier	M	M (=)		
Result (+)			S	S (=)
Value			M	M (=)
Result(-)			S	S (=)
Error Info			M	M (=)
NOTE La méthode par laquelle une primitive "confirm" est corrélée à sa				

précédente primitive "request" correspondante relève d'une initiative locale.
La méthode par laquelle une primitive "response" est corrélée à sa
précédente primitive "indication" correspondante relève d'une initiative locale.
Voir 1.2.

Variable spécifier

Ce paramètre spécifie un objet variable ou un objet "variable list" qui doit être lu par l'attribut-clé.

Value

Ce paramètre spécifie la valeur lue. Pour chacune des variables, ce paramètre spécifie la valeur de la variable. Pour les listes de variables, ce paramètre spécifie les valeurs de chacune des variables dans la liste concaténées ensemble dans l'ordre de leur apparition dans la liste.

NOTE Si l'une quelconque des variables de la liste de variables ne peut être lue, le service échoue.

5.1.4.3 Service "Write service"

5.1.4.3.1 Vue d'ensemble du service

Ce service confirmé est utilisé pour écrire la valeur d'un objet "variable" ou d'un objet "variable list" (liste de variables). Il n'est pas utilisé avec le modèle éditeur/abonné.

5.1.4.3.2 Primitives du service

Les paramètres de service pour chaque primitive sont montrés dans le Tableau 2.

Tableau 2 – Paramètres du service Write

Nom du paramètre	Req	Ind	Rsp	Cnf
Argument				
AREP	M	M		
Variable Spécifier	M	M (=)		
Value	M	M (=)		
Result (+)			S	S (=)
Result(-)			S	S (=)
Error Info			M	M (=)

NOTE La méthode par laquelle une primitive "confirm" est corrélée à sa
précédente primitive "request" correspondante relève d'une initiative locale.
La méthode par laquelle une primitive "response" est corrélée à sa
précédente primitive "indication" correspondante relève d'une initiative locale.
Voir 1.2.

Variable spécifier

Ce paramètre spécifie un objet variable ou un objet "variable list" qui doit être écrit par l'attribut-clé.

Value

Ce paramètre spécifie la valeur à écrire. Pour les variables, ce paramètre spécifie la valeur de la variable. Pour les listes de variables, ce paramètre spécifie les valeurs de chacune des variables dans la liste concaténées ensemble dans l'ordre de leur apparition dans la liste.

NOTE Si une variable quelconque dans un objet "variable list" ne peut pas être mise à jour, aucune des variables dans l'objet "variable list" ne sera mise à jour et l'écriture échouera.

5.1.4.4 Service Information report

5.1.4.4.1 Vue d'ensemble du service

Ce service facultatif est un service non confirmé qui peut être utilisé pour rapporter la valeur d'un objet variable ou d'un objet "variable". Ce service peut être utilisé dans le modèle éditeur/abonné en mode "push".

5.1.4.4.2 Primitives du service

Les paramètres de service pour chaque primitive sont montrés dans le Tableau 3.

Tableau 3 – Paramètres du service Information report

Nom du paramètre	Req	Ind
Argument		
AREP	M	M
Variable Specifier	M	M (=)
Value	M	M (=)

Variable specifier

Ce paramètre spécifie un objet variable ou un objet "variable list" qui doit être rapporté par l'attribut-clé.

Value

Ce paramètre spécifie la valeur qui doit être rapportée. Pour les variables, ce paramètre spécifie la valeur de la variable. Pour les listes de variables, ce paramètre spécifie la valeur de chaque variable dans la liste concaténée ensemble avec les autres dans l'ordre de leur apparition dans la liste.

NOTE Si l'une quelconque des variables de la liste de variables ne peut être lue, le service échoue.

5.2 ASE Event

5.2.1 Vue d'ensemble

Les objets Event ("Événement") sont utilisés pour définir les messages rapportant des occurrences d'événements. Les messages d'événement contiennent des informations qui identifient et décrivent les occurrences d'événements.

Les notificateurs sont chargés de recueillir les messages d'événements issus des objets "Event" et de distribuer un ou plusieurs messages d'événements en une seule invocation du service "Event notification" (notification d'événement) de la FAL. Le nombre de messages d'événement qui peuvent être présentés dans une seule invocation de service est limité par la taille maximale d'APDU qui peut être transférée par l'AR.

Si un processus d'application échoue à recevoir une ou plusieurs notifications, un service "Notification recovery" (récupération de notification) est fourni pour demander au notificateur d'émettre à nouveau les notifications.

Dans ce modèle, les processus d'application sont chargés de fournir les fonctions pour les objets "Event" et les objets "Event list" (liste d'événements), alors que la FAL est chargée de fournir les services de communication conçus spécifiquement pour eux. Le processus d'application détecte les événements, construit les messages d'événement et les agrège ensemble. Le processus d'application distribue les messages d'événement agrégés en utilisant le service de notification d'événement de la FAL.

5.2.2 Spécification de la classe Event

5.2.2.1 Modèle formel

ASE de FAL:			ASE EVENT
CLASSE:			NOTIFIER
ID de CLASSE:			non utilisé
CLASSE PARENTE:			TOP
ATTRIBUTS:			
1	(m)	Attribut-clé:	Numeric Identifier
2	(m)	Attribut:	AREP
3	(o)	Attribut:	Last Notification Sequence Number
4	(o)	Attribut:	List Of Events
SERVICES:			
1	(o)	OpsService:	Event Notification
2	(o)	OpsService:	Notification Recovery

5.2.2.2 Attributs

5.2.2.2.1 Numeric identifier

Cet attribut-clé identifie une instance de cette classe d'objet.

5.2.2.2.2 AREP

Cet attribut identifie l'AREP configuré pour acheminer les notifications d'événement. Cet AREP est également utilisé pour rapporter les notifications d'événement générées par une demande de récupération d'événement.

5.2.2.2.3 Last notification sequence number

L'attribut conditionnel spécifie le dernier numéro de séquence utilisé. Il est incrémenté à chaque invocation du service Event notification.

5.2.2.2.4 List of events

Cet attribut facultatif identifie les événements qui sont configurés.

5.2.3 Spécification du service de l'ASE Event

5.2.3.1 Services pris en charge

Le présent paragraphe contient la définition des services qui sont propres à cet ASE. Les services définis pour cet ASE sont:

- Event Notification
- Notification Recovery

5.2.3.2 Service Event notification

5.2.3.2.1 Vue d'ensemble du service

Ce service non confirmé est utilisé par le notificateur d'un AP de la FAL pour notifier aux autres AP l'occurrence d'un ou plusieurs événements.

5.2.3.2.2 Primitives du service

Les paramètres de service pour chaque primitive sont montrés dans le Tableau 4.

Tableau 4 – Paramètres du service Event notification

Nom du paramètre	Req	Ind
Argument		
AREP	M	M
NotifierID	M	M (=)
Sequence Number	U	U (=)
Notification Time	U	U (=)
List of Event Messages	M	M (=)
Event Key Attribute	M	M (=)
Event Data type	C	C (=)
Event Detection Time	U	U (=)
Event Data	U	U (=)

NotifierID

Ce paramètre identifie le notificateur émettant la notification d'événement.

Sequence number

Ce paramètre facultatif est le numéro de séquence pour la notification d'événement. Il peut être utilisé à des fins de récupération de notification.

Notification time

Ce paramètre facultatif est l'heure de la notification d'événement.

List of event messages

Ce paramètre spécifie la liste des messages d'événement qui doivent être rapportés. Il peut contenir des messages issus d'un ou plusieurs objets Event. Le contenu de chaque message est spécifié par son événement et il convient qu'il soit compatible avec celui qui est spécifié par l'objet notificateur.

Event key attribute

Ce paramètre identifie chacun des événements spécifiques qui sont en cours d'acquittement par ce service.

Event data type

Ce paramètre conditionnel indique le type de données de chacun des paramètres de données d'événement. Ce paramètre peut être présent seulement si le paramètre Event Data de données d'événement est présent. Si le paramètre Event Data est présent, il convient que ce paramètre soit présent.

Event detection time

Ce paramètre facultatif rapporte l'heure de la détection de l'événement. Ce paramètre peut être présent seulement s'il est défini pour l'objet événement spécifié et est pris en charge par le notificateur spécifié.

Event data

Ce paramètre facultatif spécifie les données utilisateur qui doivent être incluses dans le message d'événement en plus de celles qui sont utilisées pour identifier l'occurrence de l'événement. Ce paramètre peut être présent seulement s'il est défini pour l'objet événement spécifié et est pris en charge par le notificateur spécifié.

5.2.3.3 Service Notification recovery**5.2.3.3.1 Vue d'ensemble du service**

Ce service non confirmé est utilisé pour demander qu'un nombre spécifié de notifications d'événements soit retourné. Les notifications sont retournées en utilisant le service de notification d'événement (Event notification).

5.2.3.3.2 Primitives du service

Les paramètres de service pour chaque primitive sont montrés dans le Tableau 5.

Tableau 5 – Paramètres du service Event notification recovery

Nom du paramètre	Req	Ind
Argument		
AREP	M	M
NotifierID	M	M (=)
Sequence Number	U	U (=)

NotifierID

Ce paramètre identifie le notificateur auquel ce service est dirigé.

Sequence number

Ce paramètre facultatif spécifie le numéro de séquence de la notification d'événement à renvoyer. S'il n'est pas présent, la dernière notification envoyée est demandée.

5.3 ASE Load region

5.3.1 Vue d'ensemble

Une région de chargement (Load Region) représente une zone de mémoire non structurée dont le contenu doit être chargé (lu) ou téléchargé (écrit). Dans ce contexte, "non structuré" signifie que la zone de mémoire est représentée seulement comme étant une séquence ordonnée d'octets. Aucune autre structure n'est apparente.

Une région de chargement peut représenter une zone de mémoire volatile non nommée (telle que celle qui est mise en œuvre par une mémoire d'ordinateur dynamique) ou un objet nommé de mémoire non volatile (tel qu'un fichier). Le contenu d'une région de chargement est appelé "image de chargement" (Load image) et peut contenir des programmes ou des données. Le transfert d'image de chargement vers ou depuis une région de chargement est accompli en utilisant le processus de chargement.

Ce modèle de région de chargement fournit des services qui permettent à un processus d'application de lancer le téléchargement ou le chargement de régions de chargement spécifiées.

5.3.2 Spécification de la classe Load region

5.3.2.1 Modèle formel

ASE de FAL:		ASE LOAD REGION	
CLASSE:		LOAD REGION	
ID de CLASSE:		non utilisé	
CLASSE PARENTE:		TOP	
ATTRIBUTS:			
1	(m)	Attribut:	Load Region Size
2	(m)	Attribut:	Local Address
3	(o)	Attribut:	Load Image Name
SERVICES:			
1	(m)	OpsService:	Services de téléchargement
2	(m)	OpsService:	Services de chargement

5.3.2.2 Attributs

Load region size

Cet attribut spécifie la taille maximale de la région de chargement en octets.

Local address

Cet attribut est une adresse localement significative de la région de chargement.

Load image name

Cet attribut facultatif spécifie le nom de l'image de chargement contenue dans la région de chargement.

5.3.3 Spécification du service de l'ASE Load region**5.3.3.1 Services pris en charge**

Le présent paragraphe contient les définitions des services qui sont propres à cet ASE. Les services définis pour cet ASE sont:

- Download (Téléchargement)
- Upload (Chargement)

5.3.3.2 Service Download**5.3.3.2.1 Vue d'ensemble du service**

Ce service confirmé est utilisé pour télécharger une image de chargement dans ses primitives "request" et "indication". Les primitives "response" et "confirmation" sont utilisées pour acheminer le succès ou l'échec du téléchargement.

5.3.3.2.2 Primitives du service

Les paramètres de service pour chaque primitive sont montrés dans le Tableau 6.

Tableau 6 – Paramètres du service Download

Nom du paramètre	Req	Ind	Rsp	Cnf
Argument				
AREP	M	M		
Load Region	M	M (=)		
Load Data	M	M (=)		
Result (+)			S	S (=)
Result(-)			S	S (=)
Error Info			M	M (=)
NOTE La méthode par laquelle une primitive "confirm" est corrélée à sa précédente primitive "request" correspondante relève d'une initiative locale. La méthode par laquelle une primitive "response" est corrélée à sa précédente primitive "indication" correspondante relève d'une initiative locale. Voir 1.2.				

Load region

Ce paramètre spécifie la région de chargement dans laquelle l'image doit être téléchargée.

Load data

Ce paramètre spécifie les données qui doivent être téléchargées.

5.3.3.3 Service Upload

5.3.3.3.1 Vue d'ensemble du service

Ce service confirmé est utilisé pour charger une image de chargement. Ses primitives "request" et "indication" acheminent une demande de chargement. Les primitives "response" et "confirmation" sont utilisées pour acheminer l'image de chargement de la région de chargement et le résultat du chargement.

5.3.3.3.2 Primitives du service

Les paramètres de service pour chaque primitive sont montrés dans le Tableau 7.

Tableau 7 – Paramètres du service Upload

Nom du paramètre	Req	Ind	Rsp	Cnf
Argument				
AREP	M	M (=)		
Load Region	M	M (=)		
Result (+)			S	S (=)
Load Data			M	M (=)
Result(-)			S	S (=)
Error Info			M	M (=)
NOTE La méthode par laquelle une primitive "confirm" est corrélée à sa précédente primitive "request" correspondante relève d'une initiative locale. La méthode par laquelle une primitive "response" est corrélée à sa précédente primitive "indication" correspondante relève d'une initiative locale. Voir 1.2.				

Load region

Ce paramètre spécifie la région de chargement à partir de laquelle l'image doit être chargée.

Load data

Ce paramètre spécifie les données chargées.

5.4 ASE Function invocation

5.4.1 Vue d'ensemble

La classe "function invocation" (invocation de fonction) modélise l'invocation de fonction orientée état. Elle peut être utilisée pour modéliser des processus logiciels ou des fonctions d'utilisateur dont l'exploitation peut être commandée.

5.4.2 Spécification de la classe Function invocation

5.4.2.1 Modèle formel

ASE de FAL:

CLASSE:

ID de CLASSE:

CLASSE PARENTE:

ATTRIBUTS:

1 (m) Attribut-clé:

2 (m) Attribut:

SERVICES:

1 (o) OpsService: Start

2 (o) OpsService: Stop

3 (o) OpsService: Resume

ASE FUNCTION INVOCATION

FUNCTION INVOCATION

non utilisé

TOP

5.4.2.2 Attributs

5.4.2.2.1 Identifier

Cet attribut-clé se compose d'un identificateur de station et d'un identificateur de fonction.

5.4.2.2.2 Function invocation state

Cet attribut indique l'état actuel de l'invocation de fonction. Un ensemble énuméré de valeurs a été défini.

UNRUNNABLE	Cet état indique que l'invocation de fonction n'est pas en cours d'exécution et ne peut pas être exécutée.
IDLE	Cet état indique que l'invocation de fonction n'est pas en cours d'exécution, mais est capable d'être exécutée.
RUNNING	Cet état indique que l'invocation de fonction est en cours d'exécution.
STOPPED	Cet état indique que l'exécution de l'invocation de fonction a été suspendue.

5.4.3 Spécification du service de l'ASE Function invocation

5.4.3.1 Services pris en charge

Le présent paragraphe contient la définition des services qui sont propres à cet ASE. Les services définis pour cet ASE sont:

- Start
- Stop
- Resume

5.4.3.2 Service Start

5.4.3.2.1 Vue d'ensemble du service

Ce service confirmé est utilisé pour démarrer une invocation de fonction à partir de l'état initial.

5.4.3.2.2 Primitives du service

Les paramètres de service pour chaque primitive sont montrés dans le Tableau 8.

Tableau 8 – Paramètres du service Start

Nom du paramètre	Req	Ind	Rsp	Cnf
Argument				
AREP	M	M		
FunctionID	M	M (=)		
Result				
Error Info			M	M (=)
NOTE La méthode par laquelle une primitive "confirm" est corrélée à sa précédente primitive "request" correspondante relève d'une initiative locale. La méthode par laquelle une primitive "response" est corrélée à sa précédente primitive "indication" correspondante relève d'une initiative locale. Voir 1.2.				

FunctionID

Ce paramètre spécifie l'un des attributs-clés de l'objet function invocation.

5.4.3.3 Service Stop

5.4.3.3.1 Vue d'ensemble du service

Ce service confirmé est utilisé pour arrêter une invocation de fonction en retenant son contexte afin qu'elle puisse être reprise.

5.4.3.3.2 Primitives du service

Les paramètres de service pour chaque primitive sont montrés dans le Tableau 9.

Tableau 9 – Paramètres du service Stop

Nom du paramètre	Req	Ind	Rsp	Cnf
Argument				
AREP	M	M (=)		
FunctionID	M	M (=)		
Result				
Error Info			M	M (=)
NOTE La méthode par laquelle une primitive "confirm" est corrélée à sa précédente primitive "request" correspondante relève d'une initiative locale. La méthode par laquelle une primitive "response" est corrélée à sa précédente primitive "indication" correspondante relève d'une initiative locale. Voir 1.2.				

FunctionID

Ce paramètre spécifie l'un des attributs-clés de l'objet function invocation.

5.4.3.4 Service Resume

5.4.3.4.1 Vue d'ensemble du service

Ce service confirmé est utilisé pour demander de démarrer une invocation de fonction à partir de l'état de suspension.

5.4.3.4.2 Primitives du service

Les paramètres de service pour ce service sont montrés dans le Tableau 10.

Tableau 10 – Paramètres du service Resume

Nom de paramètre	Req	ind	Rsp	Cnf
Argument				
AREP	M	M (=)		
FunctionID	M	M (=)		
Result				
Error Info			M	M (=)

NOTE La méthode par laquelle une primitive "confirm" est corrélée à sa précédente primitive "request" correspondante relève d'une initiative locale. La méthode par laquelle une primitive "response" est corrélée à sa précédente primitive "indication" correspondante relève d'une initiative locale. Voir 1.2.

FunctionID

Ce paramètre spécifie l'un des attributs-clés de l'objet fonction invocation.

5.5 ASE Time

5.5.1 Vue d'ensemble

L'ASE "Time" spécifie le mécanisme de synchronisation d'horloge pour synchroniser le temps parmi les stations du réseau. Ce temps peut être utilisé pour ajouter une valeur temporelle à des informations d'alerte et pour ranger des messages selon un ordre temporel.

Un système consiste en un ou plusieurs maître(s) d'horloge et plusieurs esclaves d'horloge. La synchronisation d'horloge est accomplie par le système de communication. Le réajustement et la gestion de l'horloge sont des tâches de l'application.

Il convient que le réseau ait au moins un maître d'horloge réseau, qui peut être l'un des maîtres d'horloge de domaine sélectionnés automatiquement ou peut être une station fixée.

Le maître d'horloge réseau a l'horloge maîtresse du réseau. L'horloge maîtresse peut être l'horloge locale du maître d'horloge réseau lui-même ou elle peut être synchronisée avec la source d'horloge à l'extérieur du réseau.

Le maître d'horloge de domaine de chaque domaine obtient le temps réseau du maître d'horloge réseau et il distribue le temps réseau aux stations présentes dans le domaine.

5.5.2 Spécification de la classe Time

5.5.2.1 Modèle formel

ASE de FAL:

CLASSE:

ID de CLASSE:

CLASSE PARENTE:

ATTRIBUTS:

1	(m)	Attribut:
2	(m)	Attribut:
3	(o)	Attribut:
4	(o)	Attribut:
5	(o)	Attribut:

SERVICES:

1	(m)	OpsService:
2	(m)	OpsService:
3	(m)	OpsService:

ASE Time

Time

non utilisé

TOP

Implicit

Role

Stratum

Poll Interval

Precision

Get Network Time

Set Network Time

Service Tick Notification

5.5.2.2 Attributs

5.5.2.2.1 Role

Cet attribut spécifie le rôle de l'ASE "Time" avec les valeurs définies comme suit:

NETWORK TIME MASTER	Un nœud d'extrémité unique dans le réseau qui répond aux demandes d'heure issues des maîtres de domaine.
DOMAIN TIME MASTER	Un nœud d'extrémité unique dans le domaine qui distribue des informations relatives à l'heure au sein du domaine.
TIME SLAVE	Un nœud d'extrémité s'abonne aux informations relatives à l'heure distribuées par le maître de domaine.

5.5.2.2.2 Stratum

Cet attribut indique le niveau de strate de l'horloge locale.

5.5.2.2.3 Poll interval

Cet attribut indique l'intervalle maximal entre des messages d'horloge successifs.

5.5.2.2.4 Precision

Cet attribut indique le niveau de précision de l'horloge locale.

5.5.3 Spécification du service de l'ASE Time

5.5.3.1 Services pris en charge

Le présent paragraphe contient les définitions des services qui sont propres à cet ASE. Les services définis pour cet ASE sont:

- Get Network Time
- Set Network Time
- Tick Notification

5.5.3.2 Service Get network time

5.5.3.2.1 Vue d'ensemble du service

Il convient que ce service local soit utilisé pour obtenir la valeur du temps réseau qui est distribuée à partir du réseau. Si l'ASE "Time" est le maître d'horloge réseau, la valeur du temps local du maître d'horloge est retournée.

5.5.3.2.2 Primitives du service

Les paramètres de service pour chaque primitive sont montrés dans le Tableau 11.

Tableau 11 – Paramètres du service Get network time

Nom du paramètre	Req	Cnf
Argument		
Result		
Network Time		M
Status		M
Error info		C
NOTE La méthode par laquelle une primitive "confirm" est corrélée à sa précédente primitive "request" correspondante relève d'une initiative locale. Voir 1.2.		

Network time

Ce paramètre est la valeur du temps réseau.

Status

Ce paramètre indique les états de synchronisation du temps. Les états possibles sont:

- Non synchronisé
- Synchronisé au maître d'horloge de domaine
- Synchronisé au maître d'horloge réseau comme le maître d'horloge de domaine
- Synchronisé à une source de temps externe comme un maître d'horloge réseau

5.5.3.3 Service Set network time

5.5.3.3.1 Vue d'ensemble du service

Il convient que ce service confirmé soit utilisé pour établir la valeur du temps réseau au maître d'horloge.

Ce service est disponible dans les conditions où le maître d'horloge réseau n'est pas synchronisé à une source de temps externe.

5.5.3.3.2 Primitives du service

Les paramètres de service pour chaque primitive sont montrés dans le Tableau 12.

Tableau 12 – Paramètres du service Set network time

Nom du paramètre	Req	Ind	Rsp	Cnf
Argument Network Time	M	M (=)		
Result Error Info			C	C (=)
NOTE La méthode par laquelle une primitive "confirm" est corrélée à sa précédente primitive "request" correspondante relève d'une initiative locale. La méthode par laquelle une primitive "response" est corrélée à sa précédente primitive "indication" correspondante relève d'une initiative locale. Voir 1.2.				

Network time

Ce paramètre est la valeur du temps réseau qui doit être établie.

5.5.3.4 Service Tick notification

5.5.3.4.1 Vue d'ensemble du service

Il convient que ce service local soit utilisé pour reconnaître la temporisation de tops synchronisée au temps réseau. L'intervalle des tops est configurable.

5.5.3.4.2 Primitives du service

Les paramètres de service pour chaque primitive sont montrés dans le Tableau 13.

Tableau 13 – Paramètres du service Tick notification

Nom du paramètre	Ind
Result Tick	M

Tick

Ce paramètre indique la temporisation des tops.

NOTE Ce service peut être mis en œuvre au moyen d'une interruption câblée.

5.6 ASE Network management

5.6.1 Vue d'ensemble

Les nœuds d'extrémité peuvent avoir deux interfaces réseau qui fournissent la redondance de réseau. Ces interfaces sont appelées "interface réseau A" et "interface réseau B". Il convient que la première soit raccordée au réseau primaire et la seconde au réseau secondaire.

Chaque ASE "Network Management" (Gestion de réseau) construit deux APDU de message de diagnostic et les envoie simultanément aux interfaces réseau A et B. Chaque ASE "Network Management" reçoit des paires d'unités APDU issues d'autres nœuds d'extrémité sur le réseau qui participent à la redondance réseau. Ces informations sont utilisées pour déterminer quelle interface réseau (A ou B) doit être sélectionnée pour envoyer d'autres APDU. Ces informations sont également utilisées pour déterminer si le nœud destination est accessible avant l'envoi d'une APDU.

Ces informations sont préservées dans le tableau de statuts de réseau.

5.6.2 Spécification de la classe Network management

5.6.2.1 Modèle formel

ASE de FAL:			Network Management
CLASSE:			Network Management
ID de CLASSE:			non utilisé
CLASSE PARENTE:			TOP
ATTRIBUTS:			
1	(m)	Attribut:	Number of Network Interfaces
2	(m)	Attribut:	Diagnostic Message Interval
3	(m)	Attribut:	Aging Time
4	(m)	Attribut:	List of Network Interface Status
SERVICES:			
1	(m)	OpsService:	Get Network Status
2	(m)	OpsService:	Get Station Status
3	(m)	OpsService:	Network Status Change Report
4	(m)	OpsService:	Station Status Change Report

5.6.2.2 Attributs

5.6.2.2.1 Number of network interfaces

Cet attribut spécifie le nombre d'interfaces réseau sur ce nœud d'extrémité. Un nœud d'extrémité peut avoir une ou deux interfaces réseau.

5.6.2.2.2 Diagnostic message interval

Cet attribut spécifie l'intervalle de temps entre des invocations successives des messages de diagnostic en millisecondes.

5.6.2.2.3 Aging time

Cet attribut spécifie l'intervalle de temps utilisé pour retirer les nœuds d'extrémité silencieux de la List of Network Interface Status (Liste de statuts des interfaces réseau).

5.6.2.2.4 List of network interface status

Cet attribut est une liste du statut de trajet représentant l'état du trajet à une station distante à partir de laquelle l'ASE "Network Management" peut être en train de recevoir des messages de diagnostic (Diagnostic Messages) successifs. Le statut est bidirectionnel pour un trajet et indique si des messages de diagnostic (Diagnostic Messages) sont en cours de réception réussie à chaque extrémité du trajet ou si l'une ou l'autre extrémité du trajet est dans un état de défaut.

5.6.3 Spécification du service de l'ASE Network management

5.6.3.1 Services pris en charge

Le présent paragraphe contient la définition des services qui est propre à cet ASE.

Les services définis pour cet ASE sont

- Get Network Status
- Get Station Status
- Network Status Change Report
- Station Status Change Report

5.6.3.2 Service Get network status

5.6.3.2.1 Vue d'ensemble du service

Ce service local est utilisé pour obtenir le statut du réseau.

5.6.3.2.2 Primitives du service

Les paramètres de service pour chaque primitive sont montrés dans le Tableau 14.

Tableau 14 – Paramètres du service Get network status

Nom du paramètre	Req	Cnf
Argument		
Result		
Network Status		M
NOTE La méthode par laquelle une primitive "confirm" est corrélée à sa précédente primitive "request" correspondante relève d'une initiative locale. Voir 1.2.		

Network status

Ce paramètre indique la cohérence des réseaux primaire et secondaire. Les valeurs possibles sont

- HEALTHY
- FAILED

5.6.3.3 Service Get station status

5.6.3.3.1 Vue d'ensemble du service

Ce service local est utilisé pour obtenir le statut de la station spécifiée.

5.6.3.3.2 Primitives du service

Les paramètres de service pour chaque primitive sont montrés dans le Tableau 15.

Tableau 15 – Paramètres du service Get station status

Nom du paramètre	Req	Cnf
Argument		
Station Identifier	M	

Nom du paramètre	Req	Cnf
Result		
Station Status		M
Route Status		M
NOTE La méthode par laquelle une primitive "confirm" est corrélée à sa précédente primitive "request" correspondante relève d'une initiative locale. Voir 1.2.		

Station identifier

Ce paramètre indique la station pour laquelle le statut est demandé, et se compose d'un numéro de domaine et d'un numéro de station.

Station status

Ce paramètre indique le statut de la station spécifiée par l'identificateur de station et inclut les informations suivantes.

a) existence

TRUE	la station existe
FALSE	la station n'existe pas

b) redundancy

SINGLE	la station n'est pas redondante
DUAL-REDUNDANT	la station est redondante

c) AREP du nœud d'extrémité qui est en service

Route status

Ce paramètre indique l'état du chemin pour la station et inclut les informations suivantes.

a) statut du chemin sur le réseau primaire

REACHABLE
UNREACHABLE

b) statut du chemin sur le réseau secondaire

REACHABLE
UNREACHABLE

5.6.3.4 Service Network status change report

5.6.3.4.1 Vue d'ensemble du service

Ce service local est utilisé pour informer des changements de statut du réseau.

5.6.3.4.2 Primitives du service

Les paramètres de service pour chaque primitive sont montrés dans le Tableau 16.

Tableau 16 – Paramètres du service Network status change report

Nom du paramètre	Ind
Result	
Network Status	M

Network status

Ce paramètre indique la cohérence des réseaux primaire et secondaire et il a les valeurs possibles suivantes: