

INTERNATIONAL STANDARD

NORME INTERNATIONALE

**Field Device Integration (FDI) –
Part 101-1: Profiles – Foundation Fieldbus H1**

**Intégration des appareils de terrain (FDI) –
Partie 101-1: Profils – Foundation Fieldbus H1**



THIS PUBLICATION IS COPYRIGHT PROTECTED

Copyright © 2015 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 It also gives information on projects, replaced and withdrawn publications.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 15 additional languages. Also known as the International Electrotechnical Vocabulary (IEV) online.

IEC Glossary - std.iec.ch/glossary

More than 60 000 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 15 langues additionnelles. Egalement appelé Vocabulaire Electrotechnique International (IEV) en ligne.

Glossaire IEC - std.iec.ch/glossary

Plus de 60 000 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.



IEC 62769-101-1

Edition 1.0 2015-05

INTERNATIONAL STANDARD

NORME INTERNATIONALE

**Field Device Integration (FDI) –
Part 101-1: Profiles – Foundation Fieldbus H1**

**Intégration des appareils de terrain (FDI) –
Partie 101-1: Profils – Foundation Fieldbus H1**

INTERNATIONAL
ELECTROTECHNICAL
COMMISSION

COMMISSION
ELECTROTECHNIQUE
INTERNATIONALE

ICS 25.040.40; 35.100

ISBN 978-2-8322-2634-6

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

FOREWORD.....	4
INTRODUCTION.....	6
1 Scope.....	7
2 Normative references	7
3 Terms, definitions, abbreviated terms and acronyms	8
3.1 Terms and definitions.....	8
3.2 Abbreviated terms and acronyms	8
4 Conventions	8
4.1 EDDL syntax.....	8
4.2 XML syntax	8
4.3 Capitalizations	8
5 Profile for CP 1/1 (FOUNDATION™ H1)	9
5.1 General.....	9
5.2 Catalog profile	9
5.2.1 Protocol support file.....	9
5.2.2 CommunicationProfile definition.....	9
5.2.3 Profile device.....	9
5.2.4 Protocol version information	9
5.3 Associating a Package with a CP 1/1 device	10
5.3.1 Device type identification mapping.....	10
5.3.2 Device type revision mapping	10
5.4 Information Model mapping.....	10
5.4.1 ProtocolType definition	10
5.4.2 DeviceType mapping	11
5.4.3 FunctionalGroup Identification definition	11
5.4.4 BlockType property mapping	11
5.4.5 Mapping to Block ParameterSet.....	12
5.5 Topology elements.....	12
5.5.1 ConnectionPoint definition	12
5.5.2 Communication Device definition	14
5.5.3 Communication service provider definition	15
5.5.4 Network definition	16
5.6 Methods.....	17
5.6.1 Methods for FDI Communication Servers.....	17
5.6.2 Methods for Gateways	24
Annex A (normative) Topology scan schema.....	25
A.1 General.....	25
A.2 FoundationH1AddressT	25
A.3 FoundationH1ConnectionPointT.....	25
A.4 FoundationH1NetworkT	26
A.5 Network	26
A.6 FoundationBlockIdentificationT	27
A.7 FoundationIdentificationT.....	27
Annex B (normative) Transfer service parameters.....	29
B.1 General.....	29

B.2	receiveData	29
B.3	sendData	29
B.4	OperationT	30
B.5	ResponseCodeT	30
B.6	TransferResultDataT	30
B.7	TransferSendDataT	31
Annex C (informative) Communication service arguments for Transfer Method		32
Bibliography		33
Table 1 – Capability File part		9
Table 2 – CommunicationProfile definition		9
Table 3 – Device type catalog mapping		10
Table 4 – ProtocolType Foundation_H1 definition		10
Table 5 – Inherited DeviceType Property mapping		11
Table 6 – Identification Parameters		11
Table 7 – Inherited BlockType property mapping		12
Table 8 – ConnectionPointType ConnectionPoint_Foundation_H1 definition		12
Table 9 – Communication device ParameterSet definition		15
Table 10 – Method Connect arguments		18
Table 11 – Method Disconnect arguments		19
Table 12 – Method Transfer arguments		20
Table 13 – Method GetPublishedData arguments		22
Table 14 – Method SetAddress arguments		23
Table A.15 – Attributes of FoundationH1ConnectionPointT		26
Table A.16 – Elements of FoundationH1ConnectionPointT		26
Table A.17 – Elements of FoundationH1NetworkT		26
Table A.18 – Attributes of FoundationBlockIdentificationT		27
Table A.19 – Attributes of FoundationIdentificationT		28
Table B.1 – Elements of receiveData		29
Table B.2 – Enumerations of OperationT		30
Table B.3 – Attributes of ResponseCodeT		30
Table B.4 – Attributes of TransferResultDataT		31
Table B.5 – Attributes of TransferSendDataT		31

INTERNATIONAL ELECTROTECHNICAL COMMISSION

FIELD DEVICE INTEGRATION (FDI) –

Part 101-1: Profiles – Foundation Fieldbus H1

FOREWORD

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

International Standard IEC 62769-101-1 has been prepared by subcommittee 65E: Devices and integration in enterprise systems, of IEC technical committee 65: Industrial-process measurement, control and automation.

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

CDV	Report on voting
65E/352/CDV	65E/415/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.

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

A list of all parts in the IEC 62769 series, published under the general title *Field Device Integration (FDI)*, can be found on the IEC website.

The committee has decided that the contents of this publication will remain unchanged until the stability date indicated on the IEC website 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.

INTRODUCTION

The International Electrotechnical Commission (IEC) draws attention to the fact that it is claimed that compliance with this document may involve the use of patents concerning

- a) Method for the Supplying and Installation of Device-Specific Functionalities, see Patent Family DE10357276;
- b) Method and device for accessing a functional module of automation system, see Patent Family EP2182418;
- c) Methods and apparatus to reduce memory requirements for process control system software applications, see Patent Family US2013232186;
- d) Extensible Device Object Model, see Patent Family US12/893,680.

IEC takes no position concerning the evidence, validity and scope of this patent right.

The holders of these patent rights have assured the IEC that he/she is willing to negotiate licences either free of charge or under reasonable and non-discriminatory terms and conditions with applicants throughout the world. In this respect, the statement of the holder of this patent right is registered with IEC. Information may be obtained from:

- a) ABB Research Ltd
Claes Ryttoft
Affolterstrasse 4
Zurich, 8050
Switzerland
- b) Phoenix Contact GmbH & Co KG
Intellectual Property, Licenses & Standards
Flachsmarktstrasse 8, 32825 Blomberg
Germany
- c) Fisher Controls International LLC
John Dilger, Emerson Process Management LLLP
301 S. 1st Avenue, Marshalltown, Iowa 50158
USA
- d) Rockwell Automation Technologies, Inc.
1 Allen-Bradley Drive
Mayfield Heights, Ohio 44124
USA

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

ISO (www.iso.org/patents) and IEC (<http://patents.iec.ch>) maintain on-line data bases of patents relevant to their standards. Users are encouraged to consult the data bases for the most up to date information concerning patents.

FIELD DEVICE INTEGRATION (FDI) –

Part 101-1: Profiles – Foundation Fieldbus H1

1 Scope

This part of IEC 62769 specifies an FDI profile of IEC 62769 for IEC 61784-1_CP 1/1 (FOUNDATION™ Fieldbus H1)¹.

2 Normative references

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

IEC 61158-5-9:2014, *Industrial communication networks - Fieldbus specifications - Part 5-9: Application layer service definition - Type 9 elements*

IEC 61784-1, *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*

IEC 61784-3:2010, *Industrial communication networks - Profiles - Part 3: Functional safety fieldbuses - General rules and profile definitions*

IEC 61804 (all parts), *Function blocks (FB) for process control*

IEC 62541-100:2015, *OPC Unified Architecture – Part 100: OPC UA for Devices*

NOTE IEC 62769-1 is technically identical to FDI-2021.

IEC 62769-2, *Field Device Integration (FDI) – Part 2: FDI Client*

NOTE IEC 62769-2 is technically identical to FDI-2022.

IEC 62769-4:2015, *Field Device Integration (FDI) – Part 4: FDI Packages*

NOTE IEC 62769-4 is technically identical to FDI-2024.

IEC 62769-5:2015, *Field Device Integration (FDI) – Part 5: FDI Information Model*

NOTE IEC 62769-5 is technically identical to FDI-2025.

IEC 62769-6, *Field Device Integration (FDI) – Part 6: FDI Technology Mapping*

NOTE IEC 62769-6 is technically identical to FDI-2026.

¹ FOUNDATION™ Fieldbus is the trade name of the non-profit consortium Fieldbus Foundation. This information is given for the convenience of users of this standard and does not constitute an endorsement by IEC of the trademark holder or any of its products. Compliance does not require use of the trade name. Use of the trade name requires permission of the trade name holder.

IEC 62769-7:2015–, *Field Device Integration (FDI) – Part 7: FDI Communication Devices*

NOTE IEC 62769-7 is technically identical to FDI-2027.

3 Terms, definitions, abbreviated terms and acronyms

3.1 Terms and definitions

For the purposes of this document, the terms and definitions given in IEC 61158-5-9, IEC 61784-1, IEC 61784-2, IEC 61784-3, IEC 61804, IEC 62541-100, IEC 62769-2, IEC 62769-4, IEC 62769-5, IEC 62769-6 and IEC 62769-7 apply.

3.2 Abbreviated terms and acronyms

For the purposes of this document, the following abbreviations apply:

CFF	common file format
CP	communication profile (see IEC 61784-1 or IEC 61784-2)
CPF	communication profile family (see IEC 61784-1 or IEC 61784-2)
EDD	Electronic Device Description (see IEC 61804)
FB	function block
IM	Information Model
SMIB	system management information base
VFD	virtual field device

4 Conventions

4.1 EDDL syntax

This part of IEC 62769 specifies content for the EDD component that is part of an FDI Communication Package. EDDL syntax uses the font `Courier New`. EDDL syntax is used for method signature, variable, data structure and component declarations.

4.2 XML syntax

XML syntax examples use the font `Courier New`. The XML syntax is used to describe XML document schema.

EXAMPLE: `<xsd:simpleType name="Example">.`

4.3 Capitalizations

The IEC 62769 series uses capitalized terms to emphasize that these terms have an FDI specific meaning.

Some of these terms use an acronym as a prefix for example

- FDI Client, or
- FDI Server.

Some of these terms are compound terms such as:

- Communication Servers, or
- Profile Package.

Parameter names or attributes are concatenated to a single term, where the original terms start in this term with a capital letter such as:

- ProtocolSupportFile, or
- ProtocolType.

Parameter names or attributes can also be constructed by using an underscore character to concatenate two or more terms such as:

- PROFILE_ID, or
- Profibus_PA_Network.

5 Profile for CP 1/1 (FOUNDATION™ H1)

5.1 General

This profile specifies the protocol specifics needed for FDI Packages describing communication servers, gateways and devices.

5.2 Catalog profile

5.2.1 Protocol support file

5.2.1.1 Capability file

Each CP 1/1 FDI Device Package shall contain a capability file. The capability file part is described in Table 1.

Table 1 – Capability File part

Parameter	Description
Content Type:	txt/plain
Root Namespace:	Not applicable
Source Relationship:	http://fdi-cooperation.com/2010/relationships/attachment-protocol
Filename:	Use file extension .CFF

5.2.2 CommunicationProfile definition

IEC 62769-4 defines a CommunicationProfile enumeration type for the Catalog XML schema. Table 2 defines the CP 1/1 specific values for this enumeration.

Table 2 – CommunicationProfile definition

CommunicationProfile	Description
foundation_h1	CP 1/1 device type with a function block application

5.2.3 Profile device

Not supported in this standard.

5.2.4 Protocol version information

IEC 62769-4 defines an element type named InterfaceT for the Catalog XML Schema. Element type InterfaceT contains an element named Version which is supposed to provide

version information about the applied communication protocol profile. The value follows the IEC 62769-4 defined version information schema defined in element type VersionT.

The major version part of VersionT shall be set to the ITK_VER parameter. The minor and builds parts shall be set to 0.

EXAMPLE For ITK_VER 5, the value for InterfaceT is 5.0.0.

5.3 Associating a Package with a CP 1/1 device

5.3.1 Device type identification mapping

CP 1/1 device types are uniquely identified by the parameters MANUFAC_ID, DEVICE_TYPE and DEV_REV found in the Resource Block. These parameters are used to associate a given device instance to an FDI Device Package. These parameters are mapped to the FDI Device Package Catalog according to Table 3.

Table 3 – Device type catalog mapping

Catalog Element	CP Mapping
Manufacturer element of InterfaceT (IEC 62769-4)	MANUFAC_ID String format "0xdddd" where dddd is the MANUFAC_ID number in hexadecimal format.
DeviceModel element of InterfaceT (IEC 62769-4)	DEVICE_TYPE String format "0xdddd" where dddd is the DEVICE_TYPE number in hexadecimal format.
DeviceRevision element ListOfSupportedDeviceRevisionsT (IEC 62769-4)	DEV_REV String format "x.0.0" where x is the DEV_REV in decimal format (no leading zeros).

5.3.2 Device type revision mapping

Each device type is identified according to 5.3.1. A device may also include a parameter COMPATIBILITY_REV from the Resource Block. This parameter specifies the lowest device version (DEV_REV) that a new device can replace while maintaining compatibility with a prior FDI Device Package.

5.4 Information Model mapping

5.4.1 ProtocolType definition

Table 4 defines the ProtocolType used to identify CP 1/1 network communications.

Table 4 – ProtocolType Foundation_H1 definition

Attribute	Value				
BrowseName	Foundation_H1				
IsAbstract	False				
References	NodeClass	BrowseName	Data Type	TypeDefinition	ModellingRule
Inherits the properties of ProtocolType defined in IEC 62541-100.					

5.4.2 DeviceType mapping

Each device type inherits the properties of the DeviceType. The mapping of the inherited properties from the DeviceType is defined in Table 5.

Table 5 – Inherited DeviceType Property mapping

Property	CP Mapping
SerialNumber	DEV_ID (System Management Information Base)
RevisionCounter	–1 (not defined)
Manufacturer	MANUFAC_ID (Resource Block)
Model	DEV_TYPE (Resource Block)
DeviceManual	entry text string (not supported) ^a
DeviceRevision	DEV_REV (Resource Block)
SoftwareRevision	SOFTWARE_REV (if available, otherwise -1)
HardwareRevision	HARDWARE_REV (if available, otherwise -1)
^a Device manuals are exposed as attachments of the FDI Device Package.	

5.4.3 FunctionalGroup Identification definition

As defined in IEC 62541-100, each device representation in the FDI Server hosted Information Model shall contain a protocol specific FunctionalGroup called Identification. This FunctionalGroup organizes variables found in the Resource Block of the device type instance. The FunctionalGroup Identification for CP 1/1 is defined in Table 6.

Table 6 – Identification Parameters

BrowseName	DataType	Optional/Mandatory
MANUFAC_ID	UInt32	Mandatory
DEV_TYPE	UInt16	Mandatory
DEV_REV	UInt8	Mandatory
HARDWARE_REV	String	Optional
SOFTWARE_REV	String	Optional
COMPATIBILITY_REV	UInt8	Optional
CAPABILITY_LEV	UInt8	Optional
ITK_VER	UInt16	Mandatory
SIF_ITK_VER	UInt16	Optional
FD_VER	UInt16	Optional

5.4.4 BlockType property mapping

CP 1/1 device types are block-oriented according to IEC 62541-100. IEC 62769-5 specifies the mapping of EDDL BLOCK_A elements to block types and instances

The BLOCK_A maps as a subtype of the topology element BlockType and inherits the properties per IEC 62541-100. The mapping of the inherited properties of the BlockType is specified in Table 7.

Table 7 – Inherited BlockType property mapping

Property	CP Mapping (Block ParameterSet)
RevisionCounter	ST_REV
ActualMode	MODE_BLK.ACTUAL
PermittedMode	MODE_BLK.PERMITTED
NormalMode	MODE_BLK.NORMAL
TargetMode	MODE_BLK.TARGET

5.4.5 Mapping to Block ParameterSet

The ParameterSet is relative to each Block. The ParameterSet includes the CHARACTERISTICS records of the block and all the parameters found in the PARAMETERS, LOCAL_PARAMETERS and LIST_ITEMS.

The browse name of the parameters found in the PARAMETERS and LOCAL_PARAMETERS is the member name in the respective lists. For example, ST_REV is the browse name of the Static Revision parameter. LIST_ITEMS do not have member names; therefore the browse name of each LIST in the LIST_ITEMS is the item name of the list.

5.5 Topology elements

5.5.1 ConnectionPoint definition

The ConnectionPoint type ConnectionPoint_Foundation_H1 shall be used to identify CP 1/1 network communication and is defined in Table 8. The ConnectionPoint_Foundation_H1 type is a sub type of the abstract type ConnectionPointType defined in IEC 62541-100.

The Address property shall be the H1 node address.

The OrdinalNumber property reflects the position of the VFD within the SMIB VFD list. For devices exposing multiple FB VFDs the OrdinalNumber property is mandatory to address the FB VFD. For devices with a single FB VFD the OrdinalNumber property can be omitted. Devices exposed as instances of type DeviceType define their connection points as components. Hence Devices with multiple FB VFDs shall contain multiple Connection Points, one per FB VFD.

The SIFConnection property denotes whether a safety instrumented function (SIF) connection is necessary or not according to the functional safety profile (IEC 61784-3:2010, Clause 6). CP 1/1 devices that implement the functional safety profile shall have a connection point as a component that has set this property to true. Devices supporting standard connections and SIF connections shall expose two Connections Points as components.

Table 8 – ConnectionPointType ConnectionPoint_Foundation_H1 definition

Attribute	Value				
BrowseName	ConnnectionPoint_Foundation_H1				
IsAbstract	False				
References	NodeClass	BrowseName	DataType	TypeDefinition	ModellingRule
Inherits the properties of ConnectionPointType defined in IEC 62541-100.					
HasProperty	Variable	Address	Byte	PropertyType	Mandatory
HasProperty	Variable	OrdinalNumber	Int32	PropertyType	Optional
HasProperty	Variable	SIFConnection	Boolean	PropertyType	Optional

The ConnectionPoint type ConnectionPoint_Foundation_H1 shall be described by an EDD element contained in a Communication Device related FDI Package that can drive a CP 1/1 network. Actual ConnectionPoint_Foundation_H1 properties are declared by VARIABLE constructs grouped together in a COLLECTION named Foundation_H1_ConnectionPoint_Properties.

```
COMPONENT ConnectionPoint_Foundation_H1
{
    LABEL "Foundation H1 Connection point";
    CLASSIFICATION NETWORK_CONNECTION_POINT;
    CAN_DELETE FALSE;
    PROTOCOL Foundation_H1;
    CONNECTION_POINT Foundation_H1_ConnectionPoint_Properties;
}
```

```
VARIABLE Address
{
    LABEL "H1 Node address";
    HELP "Address of the H1 Node";
    CLASS DEVICE;
    TYPE UNSIGNED_INTEGER (1)
    {
        MIN_VALUE 16;
        MAX_VALUE 255;
    }
    HANDLING READ & WRITE;
}
```

```
VARIABLE OrdinalNumber
{
    LABEL "OrdinalNumber address property";
    HELP "OrdinalNumber property to address the Function Block
Application";
    CLASS DEVICE;
    TYPE UNSIGNED_INTEGER (4);
    HANDLING READ & WRITE;
}
```

```
VARIABLE SIFConnection
{
    LABEL "SIFConnection address property";
    HELP "Connection point supports SIF Connections";
    CLASS DEVICE;
    TYPE ENUMERATED (1)
    {
        {0, "NO_SIFCONNECTION"} ,
        {1, "SIFCONNECTION"}
    }
    HANDLING READ & WRITE;
}
```

```
COLLECTION Foundation_H1_ConnectionPoint_Properties
{
    LABEL "FF H1 Connection Point data";
    MEMBERS
    {
        CONNECTION_POINT_ADDRESS, Address;
        CONNECTION_POINT_ORDINALNUMBER, OrdinalNumber;
        CONNECTION_POINT_SIFCONNECTION , SIFConnection;
    }
}
```

5.5.2 Communication Device definition

According to IEC 62769-7 each FDI Communication Package shall contain an EDD element describing the device. The following EDDL source code is an example describing an FDI Communication Server.

```
COMPONENT Foundation_H1_Communication_Server
{
    LABEL "Foundation H1 communication server",
    PRODUCT_URI "urn:Fieldbus Foundation:Foundation H1 Communication
Server";
    CAN_DELETE TRUE;
    CLASSIFICATION NETWORK_COMPONENT;
    COMPONENT_RELATIONS
    {
        Foundation_H1_Communication_Device_Setup
    }
}

COMPONENT_RELATION Foundation_H1_Communication_Device_Setup
{
    LABEL "Relation between Device and communication device";
    RELATION_TYPE CHILD_COMPONENT;
    ADDRESSING { LinkId }
    COMPONENTS
    {
        Foundation_H1_Communication_Device{AUTO_CREATE 1;}
    }
    MINIMUM_NUMBER 1;
    MAXIMUM_NUMBER 4;
}

VARIABLE LinkId
{
    LABEL "Link Id address parameter of the Communication device";
    HELP "Link Id address parameter of the Communication device";
    CLASS DEVICE;
    TYPE UNSIGNED_INTEGER (2)
    {
        MIN_VALUE 4096;
        MAX_VALUE 65535;
    }
    HANDLING READ & WRITE;
}
```

Semantics of the EDDL constructs shown with the EDDL source code above are described in IEC 62769-7. The EDDL COMPONENT will be utilized by the FDI server and FDI Communication Server to create an instance of type CommunicationServerType as described in IEC 62769-7.

The LinkId VARIABLE holds the address value for a Communication device instance. In the Information model the LinkId will be represented as an instance of BaseDataVariableType and as a component of the ParameterSet of the communication device.

According to IEC 62769-7 each FDI Communication Package shall contain at least one EDD element describing at least one Communication Device component. The following EDDL source code is an example for a communication device.

```
COMPONENT Foundation_H1_Communication_Device
{
    LABEL "Foundation H1 communication device";
```

```

CAN_DELETE TRUE;
CLASSIFICATION NETWORK_COMPONENT;
COMPONENT_RELATIONS
{
    Foundation_H1_Service_Provider_Relation
}
}

COMPONENT_RELATION Foundation_H1_Service_Provider_Relation
{
    LABEL "Foundation H1 communication service provider";
    RELATION_TYPE CHILD_COMPONENT;
    COMPONENTS
    {
        Foundation_H1_Service_Provider{AUTO_CREATE 1;}
    }
    MINIMUM_NUMBER 1;
    MAXIMUM_NUMBER 16;
}

```

Semantics of the EDDL constructs shown with the EDDL source code above are described in IEC 62769-7. The EDDL COMPONENT will be utilized by the FDI server and FDI Communication Server to create an instance of type ServerCommunicationDeviceType as described in IEC 62769-7.

An instance of ServerCommunicationDeviceType shall contain the following parameter(s) with its ParameterSet. Table 9 shows the definition of Communication device ParameterSet.

Table 9 – Communication device ParameterSet definition

Attribute	Value				
BrowseName	ParameterSet				
References	NodeClass	BrowseName	DataType	TypeDefinition	ModellingRule
See IEC 62541-100:–, 5.2.					
HasTypeDefinition	ObjectType	BaseObjectType			
HasComponent	Variable	LinkId	UInt16	BaseDataVariableType	Mandatory
HasComponent	Variable	<ParameterIdentifier>		BaseDataVariableType	Mandatory-Placeholder

The LinkId parameter is an addressing parameter distinguishing multiple H1 Links. If an FDI Communication Server supports multiple physical H1 Links these are mapped within the Information Model to multiple communication device instances. If the FDI Communication Server supports only one H1 Link it shall define only one communication device within the Information Model. The value of the variable can be set to 0 in this case.

The EDD declaration of the variable LinkId is with the ADDRESSING attribute of the COMPONENT_RELATION of the FDI Communication Server definition (see 5.5.2)

5.5.3 Communication service provider definition

According to IEC 62769-7 each FDI Communication Package shall contain at least one EDD element describing at least one communication service provider component. The following EDDL source code is an example for a CP 1/1 communication service provider component.

The component reference ConnectionPoint_Foundation_H1 corresponds to the related Connection Point definition in 5.5.1.

```

COMPONENT Foundation_H1_Service_Provider
{
    LABEL "Foundation H1 communication service provider";
    CAN_DELETE FALSE;
    CLASSIFICATION NETWORK_COMMUNICATION_SERVICE_PROVIDER;
    COMPONENT_RELATIONS
    {
        Foundation_H1_Connection_Point_Relation
    }
    BYTE_ORDER BIG_ENDIAN;
}

COMPONENT_RELATION
Foundation_H1_Service_Provider_Connection_Point_Relation
{
    LABEL "Relation between communication service provider and
connection point";
    RELATION_TYPE CHILD_COMPONENT;
    ADDRESSING {Address}
    COMPONENTS
    {
        ConnectionPoint_Foundation_H1{ AUTO_CREATE 1;}
    }
    MINIMUM_NUMBER 1;
    MAXIMUM_NUMBER 1;
}

```

Semantics of the EDDL constructs shown with the EDDL source code above are described in IEC 62769-7. The EDDL COMPONENT will be utilized by the FDI server and FDI Communication Server to create an instance of type ServerCommunicationServiceType as described in IEC 62769-7.

5.5.4 Network definition

According to IEC 62769-7 each FDI Communication Package shall contain at least one EDD element describing one Network for each of the protocols that are supported by the Communication Device. The definition supports the network topology engineering.

```

COMPONENT Network_Foundation_H1
{
    LABEL "Foundation H1 Network";
    CAN_DELETE TRUE;
    CLASSIFICATION NETWORK;
    PROTOCOL Foundation_H1
    COMPONENT_RELATIONS
    {
        Foundation_H1_Network_Connection_Point_Relation
    }
}

COMPONENT_RELATION Foundation_H1_Network_Connection_Point_Relation
{
    LABEL "Relation between network and connection point";
    RELATION_TYPE CHILD_COMPONENT;
    ADDRESSING {Address}
    COMPONENTS
    {
        ConnectionPoint_Foundation_H1
    }
    MINIMUM_NUMBER 1;
    MAXIMUM_NUMBER 32;
}

```

Semantics of the EDDL constructs shown with the EDDL source code above are described in IEC 62769-7. The EDDL COMPONENT will be utilized by the FDI server and FDI Communication Server to create an instance of type NetworkType as described in IEC 62541-100.

5.6 Methods

5.6.1 Methods for FDI Communication Servers

5.6.1.1 General

The FDI Communication Server shall implement the services according to the method signatures described in 5.6.1 and according to the Information Model.

5.6.1.2 Connect

Table 10 shows the Method Connect arguments.

Signature:

```
Connect (
    [in]   ByteString      CommunicationRelationId,
    [in]   UInt16          LinkId
    [in]   byte            Address,
    [in]   Int32           OrdinalNumber
    [in]   Boolean         SIFConnection
    [in]   UInt32          ServiceID,
    [out]  UInt32          DelayForNextCall,
    [out]  Int32           ServiceError);
```

Table 10 – Method Connect arguments

Argument	Description
CommunicationRelationId	The argument value contains the nodeId of the Device ConnectionPoint representing the connection between a device and a physical network which is directly connected to the FDI Communication Server hardware. The nodeId allows finding the direct parent-child relation.
LinkId	The argument name shall match with the corresponding BrowseName of the Variable defined as a component of an instance of type CommunicationServerChannelType (refer to 5.5.3). The argument value is passed by the parent instance of a CommunicationServerChannelType. The value may be obtained by the Scan Method or may be directly configured.
Address	The argument name shall match with the corresponding attribute name defined for the ConnectionPoint which is described by a corresponding EDD element. The argument value holds the device's node address.
OrdinalNumber	The argument name shall match with the corresponding attribute name defined for the ConnectionPoint which is described by a corresponding EDD element specified in 5.4.2. The argument value holds the OrdinalNumber. The OrdinalNumber is the position of the VFD within the SMIB VFD list. If a value 0 is passed with this argument the first FB VFD is selected.
SIFConnection	The argument name shall match with the corresponding attribute name defined for the ConnectionPoint which is described by a corresponding EDD element specified in 5.4.2. The argument value denotes whether a SIF Connection is necessary or not.
ServiceId	The service transaction code establishes the relation between the service request and the corresponding response.
DelayForNextCall	The value specifies a delay time in ms to limit the EndConnect invocation cycle that shall not be faster than specified in the argument value.
ServiceError	0: OK / function started asynchronously, result has to be polled with EndConnect 1: OK / execution finished, connection established successfully –1: Connect Failed / canceled by caller –2: Call Failed / unknown service ID –3: Connect Failed / device not found –4: Connect Failed / invalid device node address –5: Connect Failed / invalid device identification –6: Connect Failed / invalid LinkId argument –7: Connect Failed / invalid OrdinalNumber argument
NOTE IEC 62769-7 defines the argument AddressData of the Connect Method as an array of Variant. The address arguments defined with the table are represented as entries of the Variant array in the order they are specified above. IEC 62769-7 defines the argument DeviceInformation as a protocol specific argument list in which the Connect Method stores the resulting data. The DeviceInformation argument is defined as an array of Variant. The DeviceInformation argument is not used.	

5.6.1.3 Disconnect

Table 11 shows the Method Disconnect arguments.

Signature:

```
Disconnect(
    [in]   ByteString      CommunicationRelationId,
    [out]  UInt32          ServiceError)
```

Table 11 – Method Disconnect arguments

Argument	Description
CommunicationRelationId	The argument value contains the nodeId of the Device ConnectionPoint representing the connection between a device and a physical network which is directly connected to the FDI Communication Server hardware. The nodeId allows finding the direct parent-child relation.
ServiceError	1: OK / disconnect finished successfully -1: Disconnect Failed / no existing communication relation -2: Disconnect Failed / invalid communication relation identifier

5.6.1.4 Transfer

Table 12 shows the Method Transfer arguments.

Signature

```
Transfer(
    [in]   ByteString      CommunicationRelationId,
    [in]   String          OPERATION,
    [in]   String          BlockTag,
    [in]   UInt32          INDEX,
    [in]   UInt32          SUB_INDEX,
    [in]   Byte[]          WriteData,
    [out]  Byte[]          ReadData,
    [in]   UInt32          ServiceId,
    [out]  UInt32          DelayForNextCall,
    [out]  Int32           ServiceError);
```

Table 12 – Method Transfer arguments

Argument	Description
CommunicationRelationId	The argument value contains the nodeId of the ConnectionPoint representing the connection between a device and a physical network within the Information Model.
OPERATION	The argument value indicates the data transfer operation. The allowed values are "READ", "WRITE" and "VIEW_READ".
BlockTag	The argument denotes the Block tag of the block instance being addressed. The value can be obtained by the Method Scan.
INDEX	OPERATION indicates "READ" or "WRITE": The argument denotes the relative index of the block parameter being addressed. The relative index can be calculated by iterating the parameters of a block within the FDI Information Model. A value of 0 addresses the block header record (described by the CHARACTERISTICS attribute within the EDD). The first parameter is addressed with the INDEX 1. OPERATION indicates "VIEW_READ": The argument denotes the view identifier in a range from 1 to 4. For instance a value of 1 requests that View_1 shall be read. Multiple View_3 or View_4 objects are identified by the INDEX and the SUB_INDEX argument.
SUB_INDEX	OPERATION indicates "READ" or "WRITE": The argument denotes the subindex of a member of the block parameter being addressed if the block parameter is of type RECORD or ARRAY. To address a simple parameter a value of 0 shall be passed with this argument. To address a specific member of a parameter of type RECORD or ARRAY a 1 relative value shall be passed with this argument. To address the parameter of type RECORD or ARRAY as a whole a value of 0 shall be passed with this argument. OPERATION indicates "VIEW_READ": The argument addresses a View_3 or View_4 if multiple views of that type exist. The argument shall be 0 if there are no multiple views of the type addressed with the INDEX argument. Values of 1 up to the number of views of that type address the specific view.
WriteData	Write data encoded as byte array. Encoding of integers shall follow the rules defined in IEC 62541-6. The argument shall be ignored if OPERATION indicates a read transfer or a view read transfer.
ReadData	With this argument the read data byte stream is returned as byte array. Encoding of integers shall follow the rules defined in IEC 62541-6. The argument shall be ignored if OPERATION indicates a write transfer.
ServiceId	The service transaction code establishes the relation between the service request and the corresponding response.
DelayForNextCall	The value specifies a delay time in ms to limit the EndTransfer invocation cycle that shall not be faster than specified in the argument value.

Argument	Description
ServiceError	0: OK / function started asynchronously, result has to be polled with EndTransfer 1: OK / execution finished, ReceivedData contains the result -1: Transfer Failed / canceled by caller -2: Call Failed / unknown service ID -3: Transfer Failed / no existing communication relation. -4: Transfer Failed / invalid communication relation identifier -5: Transfer Failed / invalid sendData content -6: Transfer Failed / invalid receiveData format -7: Transfer Failed / parameter Check ^a -8: Transfer Failed / exceeds Parameter Limits ^a -9: Transfer Failed / wrong Mode for Request ^a -10: Transfer Failed / write is prohibited by write lock switch or write lock Function Block for SIS devices ^a -11: Transfer Failed / data value is never writeable ^a -12: Transfer Failed / duplicate BlockTag detected -13: Invalid INDEX, SUB_INDEX argument provided with a "VIEW_READ" transfer
The FDI Server maintains an Information Model defined in IEC 62541-100. Hence topology elements representing an FFBlockType are separated from actual block instances. An instance called Blocks of a ConfigurableObjectType is used to implement instantiation rules. Instantiation of blocks is further detailed with IEC 62769-5. According to the rules defined in IEC 62769-5, the FDI Server needs to gather information of the FF Directory object in order to be able to create block instances. This information shall be provided by the Scan Method defined in 5.6.1.7. According to IEC 62769-5, the BlockTag argument denoted above is obtained from the DisplayName attribute of the corresponding Block instance within the FDI Information Model. NOTE 1 IEC 62769-7 defines the argument SendData of the Transfer Method as an array of Variant. The arguments OPERATION, BlockTag, INDEX, SUB_INDEX and WriteData defined in the table are represented as entries of the Variant array in the order they are specified above. NOTE 2 IEC 62769-7 defines the argument ReceiveData of the Transfer Method as an array of Variant. The argument ReadData defined in the table is represented as an entry of the Variant array in the order specified above. NOTE 3 Example (for clarification): A block has two views of type View_4. The first view of type View_4 is addressed with the arguments INDEX = 4 and SUB_INDEX = 1. The second view is addressed with the arguments INDEX = 4 and SUB_INDEX = 2.	
^a A ServiceError value may be returned with a write operation.	

5.6.1.5 GetPublishedData

CP 1/1 alerts represent unsolicited messages as defined in IEC 62769-7. Table 13 shows the Method GetPublishedData arguments.

NOTE CP 1/1 uses the term alerts to refer to alarms and event messages. These are asynchronous, unsolicited messages that deliver state change notifications such as diagnostic conditions. These messages are mapped to the GetPublishData service. CP 1/1 also uses the term publish to refer to synchronous, network scheduled communication for process values. These published messages are not mapped to the GetPublishedData service.

Signature:

```

GetPublishedData (
    [in] ByteString CommunicationRelationId,
    [out] String BlockTag,
    [out] Byte[] AlarmEventData,
    [out] NodeId AlarmEventType
    [out] DateTime TimeStamp
    [in] UInt32 ServiceId,
    [out] UInt32 DelayForNextCall,
    [out] Int32 ServiceError);

```

Table 13 – Method GetPublishedData arguments

Argument	Description
CommunicationRelationId	The argument value contains the nodeId of the ConnectionPoint representing the connection between a device and a physical network within the Information Model.
BlockTag	The output argument denotes the Block tag of the block instance that issued the alarm or event.
AlarmEventData	With this argument the alarm/event data byte stream is returned as a byte array. Encoding of integers shall follow the rules defined in IEC 62541-6.
AlarmEventType	NodeId of the alarm or event type node defined within the FDI Information model to decode the alarm / event data stream. The alarm and event types shall be read from the EDD by the FDI Server when creating the Information Model.
TimeStamp	Denotes the time the alarm or event was detected by the device.
ServiceId	The service transaction code establishes the relation between the service request and the corresponding response
DelayForNextCall	The value specifies a delay time in ms to limit the EndGetPublishedData invocation cycle that shall not be faster than specified in the argument value.
ServiceError	0: OK / function started asynchronously, result has to be polled with EndGetPublishedData 1: OK / execution finished, ReceivedData and TimeStamp contains the result -1: GetPublishedData Failed / canceled by caller -2: Call Failed / unknown service ID -3: GetPublishedData Failed / not supported -4: GetPublishedData Failed / no existing communication relation. -5: GetPublishedData Failed / invalid communication relation identifier -8: GetPublishedData Failed / no alarm/event data published. -9: GetPublishedData Failed / invalid AlarmEventType
The FDI Server maintains an Information Model defined in IEC 62541-100. Hence topology elements representing an FFBlockType are separated from actual block instances. An instance called Blocks of a ConfigurableObjectType is used to implement instantiation rules. Instantiation of blocks is further detailed with IEC 62769-5. According to the rules defined in IEC 62769-5, the FDI Server needs to gather information of the FF Directory object in order to be able to create block instances. This information shall be provided by the Scan Method defined in 5.6.1.7. According to IEC 62769-5, the BlockTag argument denoted above is obtained from the DisplayName attribute of the corresponding Block instance within the FDI Information Model. A ServiceError value may be returned with a write operation. NOTE 1 IEC 62769-7 defines the argument ReceiveData of the GetPublishedData Method as an array of Variant. The arguments BlockTag, AlarmEventData and AlarmEventType defined in the table are represented as entries of the Variant array in the order they are specified above. NOTE 2 IEC 62769-7 defines the argument SendData of the Transfer Method as an array of Variant. The arguments OPERATION, BlockTag, INDEX, SUB_INDEX and WriteData defined in the table are represented as entries of the Variant array in the order they are specified above. NOTE 3 IEC 62769-7 defines the argument ReceiveData of the Transfer Method as an array of Variant. The argument ReadData defined in the table is represented as an entry of the Variant array in the order specified above.	

5.6.1.6 SetAddress

Table 14 shows the Method SetAddress arguments.

Signature

```

SetAddress (
    [in]   String      OPERATION
    [in]   UInt16      LinkId
    [in]   byte        OldAddress,
    [in]   byte        NewAddress,
    [in]   String      NewPDTag
    [in]   UInt32      ServiceId,

```

```

[out] UInt32      DelayForNextCall,
[out] Int32       ServiceError);

```

Table 14 – Method SetAddress arguments

Argument	Description
OPERATION ^a	The argument value indicates the type of addressing operation. The allowed values are "SETASSIGNMENT", "CLEARASSIGNMENT". Argument values given with the arguments below may be ignored depending on the value of the OPERATION argument.
LinkId ^a	The argument name shall match with the corresponding BrowseName of the Variable defined as a component of an instance of type CommunicationServerChannelType (refer to 5.5.3). The argument value is passed by the parent instance of a CommunicationServerChannelType. The value may be obtained by the Scan Method or may be directly configured.
OldAddress ^a	The argument value holds the current address of a device. Allowed values are 16...255.
NewAddress ^b	The argument value holds the new address for a device. Allowed values are 0 and 16...247. The value is 0 if the service is not being used to change the H1 device's address. The argument value is ignored if the OPERATION argument value is "CLEARASSIGNMENT".
NewPDTag ^b	The argument value holds the new PD-Tag to set for the device. The argument value is ignored if the OPERATION argument value is "CLEARASSIGNMENT".
ServiceId	The service transaction code establishes the relation between the service request and the corresponding response
DelayForNextCall	The value specifies a delay time in ms to limit the EndTransfer invocation cycle that shall not be faster than specified in the argument value.
ServiceError	0: OK / function started asynchronously, result has to be polled with EndSetAddress 1: OK / execution finished successfully -1: SetAddress Failed / canceled by caller -2: Call Failed / unknown service ID -3: SetAddress Failed / not initialized -4: SetAddress Failed / not connected to a network -5: SetAddress Failed / no device found responding to oldAddress -6: SetAddress Failed / duplicate address error -7: SetAddress Failed / device did not accept new address -8: SetAddress Failed / invalid oldAddress (in terms of syntax, data type, data format, and so on) -9: SetAddress Failed / invalid newAddress (in terms of syntax, data type, data format, and so on) -10: SetAddress Failed / not possible in status connected
^a IEC 62769-7 defines the argument OldAddress of the SetAddress Method as an array of Variant. The arguments OPERATION, OldAddress and LinkId defined in the table are represented as entries of the Variant array in the order they are specified above. ^b IEC 62769-7 defines the argument NewAddress of the SetAddress Method as an array of Variant. The arguments NewAddress and NewPDTag defined in the table are represented as entries of the Variant array in the order they are specified above.	

5.6.1.7 Scan

The Method signature specified in IEC 62769-7 applies. The corresponding topologyScanResult schema is specified in Annex A.

5.6.1.8 ResetScan

The Method signature specified in IEC 62769-7 applies.

5.6.2 Methods for Gateways

Not supported in this standard.

Annex A (normative)

Topology scan schema

A.1 General

The topology scan result schema specified in Annex A describes the CP 1/1 specific format Method Scan argument `topologyScanResult`. The XML document content and structure shall correspond to the Information Model designed concept to describe a topology in order to enable generic matching between physical devices connected to the network and the FDI Server hosted Information Model.

A.2 FoundationH1AddressT

A simple type that defines the address structure for CP 1/1.

The XML schema for a FoundationH1AddressT type is:

```
<xsd:simpleType name="FoundationH1AddressT">
  <xsd:restriction base="xsd:unsignedByte">
    <xsd:minInclusive value="16"/>
    <xsd:maxInclusive value="255"/>
  </xsd:restriction>
</xsd:simpleType>
```

A.3 FoundationH1ConnectionPointT

A complex type that defines the Connection Point for CP 1/1.

The XML schema for a FoundationH1ConnectionPointT type is:

```
<xsd:complexType name="FoundationH1ConnectionPointT">
  <xsd:sequence>
    <xsd:element name="Identification"
      type="ff:FoundationIdentificationT"/>
    <xsd:element name="BlockScanInstance"
      type="ff:FoundationBlockIdentificationT" minOccurs="0"
      maxOccurs="unbounded"/>
  </xsd:sequence>
  <xsd:attribute name="Address" type="ff:FoundationH1AddressT"
    use="required"/>
  <xsd:attribute name="SIFConnection" type="xsd:boolean"
    use="required"/>
  <xsd:attribute name="OrdinalNumber" type="xsd:unsignedInt"
    use="required"/>
</xsd:complexType>
```

The attributes of a FoundationH1ConnectionPointT type are described in Table A.15.

Table A.15 – Attributes of FoundationH1ConnectionPointT

Attribute	Description
Address	The Attribute value holds the address of the network connected device.
SIFConnection	SIFConnection denotes whether a SIF Connection is necessary or not
OrdinalNumber	The OrdinalNumber property reflects the position of the VFD within the System Management VFD list. Multiple VFDs are mapped to multiple ScanItem elements.

The elements of a FoundationH1ConnectionPointT type are described in Table A.16.

Table A.16 – Elements of FoundationH1ConnectionPointT

Element	Description
Identification	The element data holds the device type identification data. Compared to the Information Model (IEC 62769-5) the ConnectionPoint does not contain or refer to the device type identification data. But in order to support the FDI host system in finding the package that matches the connected device this schema associates the device type identification with the ConnectionPoint.
BlockScanInstance	Block instance information of the scanned device VFD. Used to create Block instances within FDI Server IM. See IEC 62769-5.

A.4 FoundationH1NetworkT

A complex type that defines the network for CP 1/1.

The XML schema for a FoundationH1NetworkT type is:

```
<xsd:complexType name="FoundationH1NetworkT">
  <xsd:sequence>
    <xsd:element name="ConnectionPoint"
      type="ff:FoundationH1ConnectionPointT"
      maxOccurs="unbounded"/>
  </xsd:sequence>
</xsd:complexType>
```

The elements of a FoundationH1NetworkT type are described in Table A.17.

Table A.17 – Elements of FoundationH1NetworkT

Element	Description
ConnectionPoint	CP 1/1 Connection Point.

A.5 Network

The root element that is used to return the scan result of a CP 1/1 network.

The XML schema for a Network element is:

```
<xsd:element name="Network" type="ff:FoundationH1NetworkT"/>
```

A.6 FoundationBlockIdentificationT

A complex type that defines the block instance information of the scanned device.

The XML schema for a FoundationBlockIdentificationT type is:

```
<xsd:complexType name="FoundationBlockIdentificationT">
  <xsd:attribute name="BlockTag" use="required"/>
  <xsd:attribute name="DDItem" use="required"/>
  <xsd:attribute name="DirectoryPosition" use="required"/>
</xsd:complexType>
```

The attributes of a FoundationBlockIdentificationT type are described in Table A.18.

Table A.18 – Attributes of FoundationBlockIdentificationT

Attribute	Description
BlockTag	The BlockTag attribute shall be mapped to the DisplayName of a block instance to be created within the FDI Server IM.
DDItem	This attribute is used to find the correct block type of a block instance to be created within the FDI Server IM. The block type is looked up within the SupportedTypes Folder in the Blocks component of a DeviceType.
DirectoryPosition	This attribute denotes the relative position of the block instance within the Directory object. The first block instance has a value of 0. See block instantiation rules in IEC 62769-5.

A.7 FoundationIdentificationT

A complex type that defines the content corresponds to the FunctionalGroup Identification.

The XML schema for a FoundationIdentificationT type is:

```
<xsd:complexType name="FoundationIdentificationT">
  <xsd:attribute name="MANUFAC_ID" type="xsd:unsignedInt"
    use="required"/>
  <xsd:attribute name="DEV_TYPE" type="xsd:unsignedShort"
    use="required"/>
  <xsd:attribute name="DEV_REV" type="xsd:unsignedShort"
    use="required"/>
  <xsd:attribute name="ITK_VER" type="xsd:unsignedShort"
    use="required"/>
  <xsd:attribute name="HARDWARE_REV" type="xsd:string"
    use="optional"/>
  <xsd:attribute name="SOFTWARE_REV" type="xsd:string"
    use="optional"/>
  <xsd:attribute name="COMPATIBILITY_REV" type="xsd:unsignedInt"
    use="optional"/>
  <xsd:attribute name="CAPABILITY_LEV" type="xsd:unsignedByte"
    use="optional"/>
  <xsd:attribute name="SIF_ITK_VER" type="xsd:unsignedShort"
    use="optional"/>
  <xsd:attribute name="FD_VER" type="xsd:unsignedShort"
    use="optional"/>
</xsd:complexType>
```

The attributes of a FoundationIdentificationT type are described in Table A.19.

Table A.19 – Attributes of FoundationIdentificationT

Attribute	Description
MANUFAC_ID	Manufacturer identification number.
DEV_TYPE	Manufacturer model number associated with the resource.
DEV_REV	Manufacturer revision number associated with the resource.
ITK_VER	ITK Profile Number.
HARDWARE_REV	Manufacturer hardware revision.
SOFTWARE_REV	Manufacturer software revision.
COMPATIBILITY_REV	This parameter is optionally used when replacing field devices. The correct usage of this parameter presumes the COMPATIBILITY_REV value of the replacing device should be equal to or lower than the DEV_REV value of the replaced device.
CAPABILITY_LEV	This parameter may be included in a device to indicate the capability level supported by a device.
SIF_ITK_VER	SIF ITK Profile Number
FD_VER	A parameter equal to the value of the major version of the Field Diagnostics specification that this device was designed for.

Annex B (normative)

Transfer service parameters

B.1 General

Direct Access Services specified in IEC 62769-2 enable the User Interface Plug-in (UIP) to directly exchange data with the device. Direct data exchange means that data exchanged between a device and a UIP may not be reflected in the Information Model. The IEC 62769-6 defined interface `IDirectAccess` corresponds to the IEC 62769-2 specified Direct Access Services. Interface `IDirectAccess` defined functions `BeginTransfer` and `EndTransfer` need to convey protocol specific information. The protocol specifics shall be captured in an XML document.

B.2 receiveData

An element contains data that is returned through `IDirectAccess` function `Transfer` defined argument `receiveData`.

The XML schema for a `receiveData` element is:

```
<xsd:element name="receiveData">
  <xsd:complexType>
    <xsd:complexContent>
      <xsd:extension base="ff:TransferResultDataT">
        <xsd:sequence>
          <xsd:element name="ResponseCode"
            type="ff:ResponseCodeT" minOccurs="0"/>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
</xsd:element>
```

The elements of a `receiveData` element are described in Table B.1.

Table B.1 – Elements of receiveData

Element	Description
ResponseCode	Optional element that holds the return values for a negative service response.

B.3 sendData

An element contains data that is submitted through the `IDirectAccess` function `Transfer` defined argument `sendData`.

The XML schema for a `sendData` element is:

```
<xsd:element name="sendData" type="ff:TransferSendDataT"/>
```

B.4 OperationT

A simple type that defines service operations.

The XML schema for an OperationT enumeration type is:

```
<xsd:simpleType name="OperationT">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="READ"/>
    <xsd:enumeration value="WRITE"/>
    <xsd:enumeration value="GETOD"/>
  </xsd:restriction>
</xsd:simpleType>
```

The enumeration values of a OperationT enumeration type are described in Table B.2.

Table B.2 – Enumerations of OperationT

Enumeration	Description
READ	Read Service according to IEC 61158-5-9:2007, 6.3.5.3.2
WRITE	Write Service according to IEC 61158-5-9:2007, 6.3.5.3.3
GETOD	GetOD (long form) service according to IEC 61158-5-9:2007, 6.3.2.3.2

B.5 ResponseCodeT

A complex type that defines negative response error information.

The XML schema for a ResponseCodeT type is:

```
<xsd:complexType name="ResponseCodeT">
  <xsd:attribute name="ErrorClass" type="xsd:unsignedShort"
    use="required"/>
  <xsd:attribute name="AdditionalCode" type="xsd:short"
    use="optional"/>
  <xsd:attribute name="AdditionalDescription" type="xsd:string"
    use="optional"/>
</xsd:complexType>
```

The attributes of a ResponseCodeT type are described in Table B.3.

Table B.3 – Attributes of ResponseCodeT

Attribute	Description
ErrorClass	Class of error reported by the negative service response.
AdditionalCode	Optional reason code provided by the function block application.
AdditionalDescription	Optional text description of the negative service response.

B.6 TransferResultDataT

A complex type that defines the service parameter data format that shall be applied to Transfer defined recievedData return value.

The XML schema for a TransferResultDataT type is:

```
<xsd:complexType name="TransferResultDataT">
  <xsd:attribute name="DATA" type="xsd:hexBinary" use="optional"/>
</xsd:complexType>
```

The attributes of a TransferResultDataT type are described in Table B.4.

Table B.4 – Attributes of TransferResultDataT

Attribute	Description
DATA	Data received after the service operation. This attribute is required for the GETOD and READ service operations.

B.7 TransferSendDataT

A complex type that defines the service parameter data format that shall be applied to Transfer defined argument sendData.

The XML schema for a TransferSendDataT type is:

```
<xsd:complexType name="TransferSendDataT">
  <xsd:attribute name="OPERATION" type="ff:OperationT"
    use="required"/>
  <xsd:attribute name="OD_INDEX" type="xsd:unsignedInt"
    use="required"/>
  <xsd:attribute name="OD_SUB_INDEX" type="xsd:unsignedInt"
    use="required"/>
  <xsd:attribute name="DATA" type="xsd:hexBinary" use="optional"/>
</xsd:complexType>
```

The attributes of a TransferSendDataT type are described in Table B.5.

Table B.5 – Attributes of TransferSendDataT

Attribute	Description
OPERATION	Data transfer operation
OD_INDEX	Object dictionary index relative to the Function Block application
OD_SUB_INDEX	To address a simple parameter a value of 0 shall be passed with this attribute. To address a specific member of a parameter of type RECORD or ARRAY a 1-relative value shall be passed with this attribute. To address the parameter of type RECORD or ARRAY as a whole a value of 0 shall be passed with this attribute
DATA	Mandatory attribute for data to be transferred in the service WRITE operation. This attribute is not used for the READ or GETOD service operation.

Annex C

(informative)

Communication service arguments for Transfer Method

IEC 62769-3 details that communication service arguments for the Transfer Method (see 5.6.1.4) are obtained from COMMAND elements associated to the VARIABLE element. For variable read or write access the FDI server shall obtain the related COMMAND description and obtain the communication service arguments for the Transfer Method from the attributes of the COMMAND description via name matching.

Since the CPF1 EDD profile does not provide a COMMAND EDD item this approach requires additional considerations. In order to keep the FDI Server as generic as possible the following solution is proposed.

The COMMAND construct is introduced but only at a virtual level. This means there will be no means within the CPF1 EDD profile grammar to define a COMMAND item.

The COMMAND item will have the following attributes:

- INDEX
- SUB_INDEX

The COMMAND item shall be related to the PARAMETER of the block.

The COMMAND item representation will be created automatically (on the fly) for each block parameter by the FDI (EDD) engine when a block is loaded by the engine.

NOTE With today's EDD services the creation of the COMMAND items for PARAMETERS can be accomplished during `ddi_get_item()` for the block when the list of parameters is created.

For parameters of the type RECORD or ARRAY COMMAND DD items shall be created for each member of the RECORD or array. This is to ease operation for the FDI Server.

When the EDD is loaded by the FDI Server the BlockTypes are created within the SupportedTypes Folder in the Blocks component of the DeviceType (see IEC 62769-5 and IEC 62541-100). For each parameter of the BlockType the COMMAND description is provided by the FDI (EDD) engine.

Block instances are created with the result of the Scan Method as described in 5.6.1.7. Block instances are created as (child) components of the Blocks component. According to IEC 62769-5, the DisplayName of the Block instance is the BlockTag.

For variable read or write access the FDI Server shall obtain the communication service arguments INDEX and SUB_INDEX from the COMMAND description of the parameter via name matching. For the communication service argument BlockTag the DisplayName of the block instance shall be used.

Bibliography

IEC 62769-3:2015, *Field Device Integration (FDI) – Part 3: FDI Server*

NOTE IEC 62769-3 is technically identical to FDI-2023

FIELDBUS FOUNDATION. *FOUNDATION Specification Common File Format*. FF-103, Version 1.9. 4 June 2010.

FIELDBUS FOUNDATION. *FOUNDATION Specification Function Block Application Process – Part 1*. FF-890, Version 1.10. 2 Aug. 2012.

FIELDBUS FOUNDATION. *FOUNDATION Specification Function Block Application Process – Part 2*. FF-891, Version 1.10. 2 Aug. 2012.

FIELDBUS FOUNDATION. *FF-880: FOUNDATION Specification System Management*. FF-980, Version 1.6. 1 Dec. 2005.

SOMMAIRE

AVANT-PROPOS	36
INTRODUCTION	38
1 Domaine d'application	39
2 Références normatives	39
3 Termes, définitions, abréviations et acronymes	40
3.1 Termes et définitions	40
3.2 Abréviations et acronymes	40
4 Conventions	40
4.1 Syntaxe EDDL	40
4.2 Syntaxe XML	40
4.3 Majuscules	40
5 Profil pour CP 1/1 (Foundation™ H1)	41
5.1 Généralités	41
5.2 Profil de catalogue	41
5.2.1 Fichier de prise en charge de protocole	41
5.2.2 Définition de CommunicationProfile	41
5.2.3 Appareil de profil	42
5.2.4 Informations relatives à la version de protocole	42
5.3 Association d'un paquetage avec un appareil CP 1/1	42
5.3.1 Mapping d'identification du type d'appareil	42
5.3.2 Mapping de révisions de type d'appareil	42
5.4 Mapping du Modèle d'Information	42
5.4.1 Définition de ProtocolType	42
5.4.2 Mapping de DeviceType	43
5.4.3 Définition du FunctionalGroup "Identification"	43
5.4.4 Mapping des propriétés de BlockType	44
5.4.5 Mapping au ParameterSet du bloc	44
5.5 Eléments de topologie	44
5.5.1 Définition de ConnectionPoint	44
5.5.2 Définition de l'appareil de communication	46
5.5.3 Définition du fournisseur de service de communication	48
5.5.4 Définition du réseau	48
5.6 Méthodes	49
5.6.1 Méthodes pour les Serveurs de communication FDI	49
5.6.2 Méthodes pour les passerelles	56
Annexe A (normative) Schéma de balayage topologique	57
A.1 Généralités	57
A.2 FoundationH1AddressT	57
A.3 FoundationH1ConnectionPointT	57
A.4 FoundationH1NetworkT	58
A.5 Network	58
A.6 FoundationBlockIdentificationT	59
A.7 FoundationIdentificationT	59
Annexe B (normative) Paramètres du service Transfer	61
B.1 Généralités	61

B.2	receiveData	61
B.3	sendData	61
B.4	OperationT	62
B.5	ResponseCodeT	62
B.6	TransferResultDataT	62
B.7	TransferSendDataT	63
Annexe C (informative) Arguments du service de communication pour la méthode Transfer		64
Bibliographie		65
Tableau 1 – Partie fichier de capacité		41
Tableau 2 – Définition de CommunicationProfile		41
Tableau 3 – Mapping du catalogue de type d'appareil		42
Tableau 4 – Définition de ProtocolType Foundation_H1		43
Tableau 5 – Mapping des propriétés héritées de DeviceType		43
Tableau 6 – Paramètres d'Identification		43
Tableau 7 – Mapping des propriétés héritées de BlockType		44
Tableau 8 – Définition de ConnectionPointType ConnectionPoint_Foundation_H1		45
Tableau 9 – Définition du ParameterSet de l'appareil de communication		47
Tableau 10 – Arguments de la méthode Connect		50
Tableau 11 – Arguments de la méthode Disconnect		51
Tableau 12 – Arguments de la méthode Transfer		52
Tableau 13 – Arguments de la méthode GetPublishedData		54
Tableau 14 – Arguments de la méthode SetAddress		55
Tableau A.15 – Attributs de FoundationH1ConnectionPointT		58
Tableau A.16 – Eléments de FoundationH1ConnectionPointT		58
Tableau A.17 – Eléments de FoundationH1NetworkT		58
Tableau A.18 – Attributs de FoundationBlockIdentificationT		59
Tableau A.19 – Attributs de FoundationIdentificationT		60
Tableau B.1 – Eléments de receiveData		61
Tableau B.2 – Enumérations d'OperationT		62
Tableau B.3 – Attributs de ResponseCodeT		62
Tableau B.4 – Attributs de TransferResultDataT		63
Tableau B.5 – Attributs de TransferSendDataT		63

COMMISSION ÉLECTROTECHNIQUE INTERNATIONALE

INTEGRATION DES APPAREILS DE TERRAIN (FDI) –

Partie 101-1: Profils – Foundation Fieldbus H1

AVANT-PROPOS

- 1) La Commission Electrotechnique Internationale (IEC) est une organisation mondiale de normalisation composée de l'ensemble des comités électrotechniques nationaux (Comités nationaux de l'IEC). L'IEC a pour objet de favoriser la coopération internationale pour toutes les questions de normalisation dans les domaines de l'électricité et de l'électronique. A cet effet, l'IEC – 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 l'IEC"). 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 l'IEC, participent également aux travaux. L'IEC 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 l'IEC concernant les questions techniques représentent, dans la mesure du possible, un accord international sur les sujets étudiés, étant donné que les Comités nationaux de l'IEC intéressés sont représentés dans chaque comité d'études.
- 3) Les Publications de l'IEC se présentent sous la forme de recommandations internationales et sont agréées comme telles par les Comités nationaux de l'IEC. Tous les efforts raisonnables sont entrepris afin que l'IEC s'assure de l'exactitude du contenu technique de ses publications; l'IEC 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 l'IEC s'engagent, dans toute la mesure possible, à appliquer de façon transparente les Publications de l'IEC dans leurs publications nationales et régionales. Toutes divergences entre toutes Publications de l'IEC et toutes publications nationales ou régionales correspondantes doivent être indiquées en termes clairs dans ces dernières.
- 5) L'IEC elle-même ne fournit aucune attestation de conformité. Des organismes de certification indépendants fournissent des services d'évaluation de conformité et, dans certains secteurs, accèdent aux marques de conformité de l'IEC. L'IEC n'est responsable d'aucun des services effectués par les organismes de certification indépendants.
- 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 l'IEC, 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 l'IEC, 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 l'IEC ou de toute autre Publication de l'IEC, 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.

La Norme internationale IEC 62769-101-1 a été établie par le sous-comité 65E: Les appareils et leur intégration dans les systèmes de l'entreprise, du comité d'études 65 de l'IEC: Mesure, commande et automation dans les processus industriels.

Le texte de cette norme est issu des documents suivants:

CDV	Rapport de vote
65E/352/CDV	65E/415/RVD

Le rapport de vote indiqué dans le tableau ci-dessus donne toute information sur le vote ayant abouti à l'approbation de cette norme.

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

Une liste de toutes les parties de la série IEC 62769, publiées sous le titre général *Intégration des appareils de terrain (FDI)*, peut être consultée sur le site web de l'IEC.

Le comité a décidé que le contenu de cette publication ne sera pas modifié avant la date de stabilité indiquée sur le site web de l'IEC sous "<http://webstore.iec.ch>" dans les données relatives à la publication recherchée. A cette date, la publication sera

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

INTRODUCTION

La Commission Electrotechnique Internationale (IEC) attire l'attention sur le fait qu'il est déclaré que la conformité avec les dispositions du présent document peut impliquer l'utilisation de brevets intéressant:

- a) la méthode de fourniture et d'installation des fonctionnalités spécifiques aux appareils (cf. famille de brevets DE10357276);
- b) la méthode et l'appareil utilisés pour l'accès à un module fonctionnel du système d'automation (cf. famille de brevets EP2182418);
- c) les méthodes et les appareils utilisés pour diminuer les exigences mémoire relatives aux applications logicielles du système de commande de processus (cf. famille de brevets US2013232186);
- d) modèle d'objet d'appareil extensible (cf. famille de brevets US12/893,680).

L'IEC ne prend pas position quant à la preuve, à la validité et à la portée de ces droits de propriété.

Le détenteur de ces droits de propriété a donné l'assurance à l'IEC qu'il consent à négocier des licences avec des demandeurs du monde entier, soit sans frais soit à des termes et conditions raisonnables et non discriminatoires. A ce propos, la déclaration du détenteur des droits de propriété est enregistrée à l'IEC. Des informations peuvent être demandées à:

- e) ABB Research Ltd
Claes Ryttoft
Affolterstrasse 4
Zurich, 8050
Suisse
- f) Phoenix Contact GmbH & Co KG
Intellectual Property, Licenses & Standards
Flachsmarktstrasse 8, 32825 Blomberg
Allemagne
- g) Fisher Controls International LLC
John Dilger, Emerson Process Management LLLP
301 S. 1st Avenue, Marshalltown, Iowa 50158
Etats-Unis
- h) Rockwell Automation Technologies, Inc.
1 Allen-Bradley Drive
Mayfield Heights, Ohio 44124
Etats-Unis

L'attention est d'autre part attirée sur le fait que certains des éléments du présent document peuvent faire l'objet de droits de propriété autres que ceux qui ont été mentionnés ci-dessus. L'IEC ne saurait être tenue pour responsable de l'identification de ces droits de propriété en tout ou partie.

L'ISO (www.iso.org/patents) et l'IEC (<http://patents.iec.ch>) tiennent à jour des bases de données en ligne sur les brevets relatifs à leurs normes. Les utilisateurs sont encouragés à consulter ces bases de données pour obtenir l'information la plus récente concernant les brevets.

INTEGRATION DES APPAREILS DE TERRAIN (FDI) –

Partie 101-1: Profils – Foundation Fieldbus H1

1 Domaine d'application

La présente partie de l'IEC 62769 spécifie un profil FDI de l'IEC 62769 pour l'IEC 61784-1_CP 1/1 (Foundation™ Fieldbus H1)¹.

2 Références normatives

Les documents suivants sont cités en référence de manière normative, en intégralité ou en partie, dans le présent document et sont indispensables pour son application. Pour les références datées, seule l'édition citée s'applique. Pour les références non datées, la dernière édition du document de référence s'applique (y compris les éventuels amendements).

IEC 61158-5-9:2014, *Réseaux de communication industriels - Spécifications des bus de terrain - Partie 5-9: Définition des services de la couche application - Eléments de type 9*

IEC 61784-1, *Réseaux de communication industriels - Profils - Partie 1: Profils de bus de terrain*

IEC 61784-2, *Réseaux de communication industriels - Profils - Partie 2: Profils de bus de terrain supplémentaires pour les réseaux en temps réel basés sur l'ISO/CEI 8802-3*

IEC 61784-3:2010, *Réseaux de communication industriels - Profils - Partie 3: Bus de terrain de sécurité fonctionnelle - Règles générales et définitions de profils*

IEC 61804 (toutes les parties), *Blocs fonctionnels (FB) pour les procédés industriels*

IEC 62541-100:2015, *OPC Unified Architecture – Part 100: OPC UA for Devices* (disponible en anglais seulement)

NOTE L'IEC 62769-1 est techniquement identique à la FDI-2021.

IEC 62769-2, *Intégration des appareils de terrain (FDI) – Partie 2: Client FDI*

NOTE L'IEC 62769-2 est techniquement identique à la FDI-2022.

IEC 62769-4:2015, *Intégration des appareils de terrain (FDI) – Partie 4: Paquetages FDI*

NOTE L'IEC 62769-4 est techniquement identique à la FDI-2024.

IEC 62769-5:2015, *Intégration des appareils de terrain (FDI) – Partie 5: Modèle d'Information FDI*

NOTE L'IEC 62769-5 est techniquement identique à la FDI-2025.

¹ Foundation™ Fieldbus est l'appellation commerciale du consortium Fieldbus Foundation, une organisation à but non lucratif. Cette information est donnée à l'intention des utilisateurs de la présente norme et ne signifie nullement que l'IEC approuve ou recommande l'emploi exclusif du produit ainsi désigné. La conformité n'exige pas l'utilisation de l'appellation commerciale. L'utilisation de l'appellation commerciale exige l'autorisation du détenteur de l'appellation commerciale.

IEC 62769-6, *Intégration des appareils de terrain (FDI) – Partie 6: Mapping de technologies FDI*

NOTE L'IEC 62769-6 est techniquement identique à la FDI-2026.

IEC 62769-7, *Intégration des appareils de terrain (FDI) – Partie 7: Appareils de communication FDI*

NOTE L'IEC 62769-7 est techniquement identique à la FDI-2027.

3 Termes, définitions, abréviations et acronymes

3.1 Termes et définitions

Pour les besoins du présent document, les termes et définitions de l'IEC 61158-5-9, l'IEC 61784-1, l'IEC 61784-2, l'IEC 61784-3, l'IEC 61804, l'IEC 62541-100, l'IEC 62769-2, l'IEC 62769-4, l'IEC 62769-5, l'IEC 62769-6 et l'IEC 62769-7 s'appliquent.

3.2 Abréviations et acronymes

Pour les besoins du présent document, les abréviations suivantes s'appliquent:

CFF	Common File Format (format de fichier commun)
CP	Communication Profile (profil de communication, voir l'IEC 61784-1 ou l'IEC 61784-2)
CPF	Communication Profile Family (famille de profils de communication, voir l'IEC 61784-1 ou l'IEC 61784-2)
EDD	Electronic Device Description (description d'appareil électronique, voir l'IEC 61804)
FB	Function block (bloc fonctionnel)
IM	Information Model (Modèle d'Information)
SMIB	System Management Information Base (base d'informations de gestion de système)
VFD	Virtual Field Device (appareil de terrain virtuel)

4 Conventions

4.1 Syntaxe EDDL

La présente partie de l'IEC 62769 spécifie le contenu du composant EDD qui fait partie des Paquetages de Communication FDI. La syntaxe EDDL utilise la police *Courier New*. La Syntaxe EDDL est utilisée pour les déclarations des signatures de méthodes, des variables, des structures de données et des composants.

4.2 Syntaxe XML

Les exemples de syntaxe XML utilisent la police *Courier New*. La syntaxe XML est utilisée pour décrire le schéma des documents XML.

Exemple: `<xsd:simpleTypeName="Example">.`

4.3 Majuscules

La série IEC 62769 utilise des termes en majuscules pour souligner que ces termes ont une signification spécifique à la FDI.

Certains de ces termes utilisent un acronyme comme un préfixe par exemple

- FDI Client (Client FDI), ou
- FDI Server (Serveur FDI).

Certains de ces termes sont des termes composés comme:

- Communication Servers (Serveurs de communication), ou
- Profile Package (Paquetage de profil).

Les noms de paramètres ou attributs sont concaténés en un seul terme, où les termes originaux commencent dans ce terme par une lettre majuscule comme:

- ProtocolSupportFile, ou
- ProtocolType.

Les noms de paramètres ou attributs peuvent aussi être construits en utilisant le caractère de soulignement pour concaténer au moins deux termes comme:

- PROFILE_ID, ou
- Profibus_PA_Network.

5 Profil pour CP 1/1 (Foundation™ H1)

5.1 Généralités

Ce profil spécifie les spécificités du protocole dont ont besoin les Paquetages FDI décrivant des serveurs de communication, des passerelles et des appareils.

5.2 Profil de catalogue

5.2.1 Fichier de prise en charge de protocole

5.2.1.1 Fichier de capacité

Chaque Paquetage d'appareil FDI CP 1/1 doit contenir un fichier de capacité. La partie fichier de capacité est décrite dans le Tableau 1.

Tableau 1 – Partie fichier de capacité

Paramètre	Description
Type de contenu:	txt/clair
Espace de noms racine:	Non applicable
Relation source:	http://fdi-cooperation.com/2010/relationships/attachment-protocol
Nom de fichier:	Utiliser l'extension de fichier .CFF

5.2.2 Définition de CommunicationProfile

L'IEC 62769-4 définit un type d'énumération CommunicationProfile pour le schéma XML du catalogue. Le Tableau 2 définit les valeurs spécifiques au CP 1/1 pour cette énumération.

Tableau 2 – Définition de CommunicationProfile

CommunicationProfile	Description
foundation_h1	Type d'appareil CP 1/1 avec une application de bloc fonctionnel

5.2.3 Appareil de profil

Non prises en charge dans la présente norme.

5.2.4 Informations relatives à la version de protocole

L'IEC 62769-4 définit un type d'élément nommé InterfaceT pour le schéma XML Catalog. Le type d'élément InterfaceT contient un élément nommé Version qui est censé fournir des informations de version relatives au profil de protocole de communication appliqué. La valeur suit le schéma d'informations de version définie par l'IEC 62769-4, lequel schéma est défini dans le type d'élément VersionT.

La partie version majeure de VersionT doit être définie sur le paramètre ITK_VER. Les parties version mineure et numéro de build doivent être mises à 0.

EXEMPLE Pour ITK_VER 5, la valeur pour InterfaceT est 5.0.0.

5.3 Association d'un paquetage avec un appareil CP 1/1

5.3.1 Mapping d'identification du type d'appareil

Les types d'appareils CP 1/1 sont identifiés de manière unique par les paramètres MANUFAC_ID, DEVICE_TYPE et DEV_REV disponibles dans le bloc de ressources. Ces paramètres sont utilisés pour associer une instance donnée d'appareil avec un Paquetage d'appareil FDI. Ces paramètres sont mappés au Catalogue de Paquetage d'appareil FDI conformément au Tableau 3.

Tableau 3 – Mapping du catalogue de type d'appareil

Elément du catalogue	Mapping CP
Elément Manufacturer d'InterfaceT (IEC 62769-4)	MANUFAC_ID Format de chaîne "0xdd" où dddd est le numéro MANUFAC_ID au format hexadécimal.
Elément DeviceModel d'InterfaceT (IEC 62769-4)	DEVICE_TYPE Format de chaîne "0xdd" où dddd est le numéro DEVICE_TYPE au format hexadécimal.
Elément DeviceRevision de ListOfSupportedDeviceRevisionsT (IEC 62769-4)	DEV_REV Format de chaîne "x.0.0" où x est le numéro DEV_REV au format décimal (pas de zéros de tête).

5.3.2 Mapping de révisions de type d'appareil

Chaque type d'appareil est identifié selon 5.3.1. Un appareil peut également inclure un paramètre COMPATIBILITY_REV du bloc de ressources. Ce paramètre spécifie la version d'appareil la plus ancienne (DEV_REV) qu'un nouvel appareil peut remplacer tout en maintenant la compatibilité avec un Paquetage d'appareil FDI précédent.

5.4 Mapping du Modèle d'Information

5.4.1 Définition de ProtocolType

Le Tableau 4 définit le ProtocolType utilisé pour identifier des communications réseau CP 1/1.

Tableau 4 – Définition de ProtocolType Foundation_H1

Attribut	Valeur				
BrowseName	Foundation_H1				
IsAbstract	False				
Références	NodeClass	BrowseName	DataType	TypeDefinition	ModellingRule
Hérite des propriétés du ProtocolType définies dans l'IEC 62541-100.					

5.4.2 Mapping de DeviceType

Chaque type d'appareil hérite des propriétés du DeviceType. Le mapping des propriétés héritées du DeviceType est défini dans le Tableau 5.

Tableau 5 – Mapping des propriétés héritées de DeviceType

Propriété	Mapping CP
SerialNumber	DEV_ID (base d'informations de gestion du système)
RevisionCounter	-1 (non défini)
Manufacturer	MANUFAC_ID (bloc de ressources)
Model	DEV_TYPE (bloc de ressources)
DeviceManual	Chaîne textuelle d'entrée (non prise en charge) ^a
DeviceRevision	DEV_REV (bloc de ressources)
SoftwareRevision	SOFTWARE_REV (si disponible, autrement -1)
HardwareRevision	HARDWARE_REV (si disponible, autrement -1)
^a Les manuels d'appareils sont exposés comme pièces jointes du Paquetage d'appareil FDI.	

5.4.3 Définition du FunctionalGroup "Identification"

Comme défini dans l'IEC 62541-100, chaque représentation d'appareil dans le Modèle d'Information hébergé par le Serveur FDI doit contenir un FunctionalGroup spécifique à un protocole appelé Identification. Ce FunctionalGroup organise les variables qui se trouvent dans le bloc de ressources de l'instance de type d'appareil. L'identification du FunctionalGroup pour le CP 1/1 est définie dans le Tableau 6.

Tableau 6 – Paramètres d'Identification

BrowseName	DataType	Facultatif/Obligatoire
MANUFAC_ID	UInt32	Obligatoire
DEV_TYPE	UInt16	Obligatoire
DEV_REV	UInt8	Obligatoire
HARDWARE_REV	String	Facultatif
SOFTWARE_REV	String	Facultatif
COMPATIBILITY_REV	UInt8	Facultatif
CAPABILITY_LEV	UInt8	Facultatif
ITK_VER	UInt16	Obligatoire
SIF_ITK_VER	UInt16	Facultatif
FD_VER	UInt16	Facultatif

5.4.4 Mapping des propriétés de BlockType

Les types d'appareils CP 1/1 sont orientés bloc conformément à l'IEC 62541-100. L'IEC 62769-5 spécifie le mapping des éléments EDDL BLOCK_A aux types et instances de bloc.

Le BLOCK_A est mappé comme un sous-type de l'élément de typologie BlockType et hérite des propriétés indiquées dans l'IEC 62541-100. Le mapping des propriétés héritées de BlockType est spécifié dans le Tableau 7.

Tableau 7 – Mapping des propriétés héritées de BlockType

Propriété	Mapping CP (ParameterSet du bloc)
RevisionCounter	ST_REV
ActualMode	MODE_BLK.ACTUAL
PermittedMode	MODE_BLK.PERMITTED
NormalMode	MODE_BLK.NORMAL
TargetMode	MODE_BLK.TARGET

5.4.5 Mapping au ParameterSet du bloc

Le ParameterSet est relatif à chaque bloc. Le ParameterSet inclut les enregistrements CHARACTERISTICS du bloc et tous les paramètres qui se trouvent dans PARAMETERS, LOCAL_PARAMETERS et LIST_ITEMS.

Le nom de navigation des paramètres qui se trouve dans PARAMETERS et LOCAL_PARAMETERS est le nom du membre dans les listes respectives. Par exemple, ST_REV est le nom de navigation du paramètre de la révision statique. LIST_ITEMS n'a pas les noms des membres; par conséquent, le nom de navigation de chaque LIST dans LIST_ITEMS est le nom d'article de la liste.

5.5 Éléments de topologie

5.5.1 Définition de ConnectionPoint

Le type ConnectionPoint ConnectionPoint_Foundation_H1 doit être utilisé pour identifier la communication de réseau CP 1/1 et est défini dans le Tableau 8. Le type ConnectionPoint_Foundation_H1 est un sous-type du type abstrait ConnectionPointType défini dans IEC 62541-100.

La propriété Address doit être à l'adresse du nœud H1.

La propriété OrdinalNumber reflète la position du VFD dans la liste SMIB VFD. Pour les appareils qui exposent plusieurs FB VFD, la propriété OrdinalNumber est obligatoire pour adresser le FB VFD. Pour les appareils ayant un FB VFD unique, la propriété OrdinalNumber peut être omise. Les appareils exposés comme instance de type DeviceType définissent leurs points de connexion comme des composants. Par conséquent, les appareils ayant des FB VFD multiples doivent contenir des Points de connexion multiples, un par FB VFD.

La propriété SIFConnection dénote si une connexion de fonction instrumentée de sécurité (SIF, *Safety Instrumented Function*) est nécessaire ou non, selon le profil de sécurité fonctionnelle (IEC 61784-3:2010, Article 6). Les appareils CP 1/1 qui mettent en œuvre le profil de sécurité fonctionnelle doivent avoir le point de connexion comme un composant qui a mis cette propriété sur True. Les appareils qui prennent en charge les connexions normalisées et les connexions SIF doivent exposer deux Points de connexion comme des composants.

Tableau 8 – Définition de ConnectionPointType ConnectionPoint_Foundation_H1

Attribut	Valeur				
BrowseName	ConnexionPoint_Foundation_H1				
IsAbstract	False				
Références	NodeClass	BrowseName	Data Type	TypeDefinition	ModellingRule
Hérite des propriétés de ConnectionPointType définies dans l'IEC 62541-100.					
HasProperty	Variable	Address	Byte	PropertyType	Obligatoire
HasProperty	Variable	OrdinalNumber	Int32	PropertyType	Facultatif
HasProperty	Variable	SIFConnection	Boolean	PropertyType	Facultatif

Le type ConnectionPoint de ConnectionPoint_Foundation_H1 doit être décrit par un élément EDD contenu dans un Appareil de communication relié au Paquetage FDI qui peut actionner un réseau CP 1/1. Les propriétés réelles de ConnectionPoint_Foundation_H1 sont déclarées par les constructions VARIABLE regroupées dans une COLLECTION nommée Foundation_H1_ConnectionPoint_Properties.

```

COMPONENT ConnexionPoint_Foundation_H1
{
    LABEL "Foundation H1 Connection point";
    CLASSIFICATION NETWORK_CONNECTION_POINT;
    CAN_DELETE FALSE;
    PROTOCOL Foundation_H1;
    CONNECTION_POINT Foundation_H1_ConnectionPoint_Properties;
}

VARIABLE Address
{
    LABEL "H1 Nodeaddress";
    HELP "Address of the H1 Node";
    CLASS DEVICE;
    TYPE UNSIGNED_INTEGER (1)
    {
        MIN_VALUE 16;
        MAX_VALUE 255;
    }
    HANDLING READ & WRITE;
}

VARIABLE OrdinalNumber
{
    LABEL "OrdinalNumberaddressproperty";
    HELP "OrdinalNumberproperty to address the Function Block
Application";
    CLASS DEVICE;
    TYPE UNSIGNED_INTEGER (4);
    HANDLING READ & WRITE;
}

VARIABLE SIFConnection
{
    LABEL "SIFConnectionaddressproperty";
    HELP "Connection point supports SIF Connections";
    CLASS DEVICE;
    TYPE ENUMERATED (1)
    {
        {0, "NO_SIFCONNECTION"},

```

```

        {1, "SIFCONNECTION"}
    }
    HANDLING READ & WRITE;
}

COLLECTION Foundation_H1_ConnectionPoint_Properties
{
    LABEL "FF H1 Connection Point data";
    MEMBERS
    {
        CONNECTION_POINT_ADDRESS, Address;
        CONNECTION_POINT_ORDINALNUMBER, OrdinalNumber;
        CONNECTION_POINT_SIFCONNECTION, SIFConnection;
    }
}

```

5.5.2 Définition de l'appareil de communication

Conformément à l'IEC 62769-7, chaque Paquetage de communication FDI doit contenir un élément EDD qui décrit l'appareil. Le code source EDDL suivant est un exemple qui décrit un Serveur de communication FDI.

```

COMPONENT Foundation_H1_Communication_Server
{
    LABEL "Foundation H1 communication server",
    PRODUCT_URI "urn:FieldbusFoundation:Foundation H1 Communication
Server";
    CAN_DELETE TRUE;
    CLASSIFICATION NETWORK_COMPONENT;
    COMPONENT_RELATIONS
    {
        Foundation_H1_Communication_Device_Setup
    }
}

COMPONENT_RELATION Foundation_H1_Communication_Device_Setup
{
    LABEL "Relation between Device and communication device";
    RELATION_TYPE CHILD_COMPONENT;
    ADDRESSING { LinkId }
    COMPONENTS
    {
        Foundation_H1_Communication_Device{AUTO_CREATE 1;}
    }
    MINIMUM_NUMBER 1;
    MAXIMUM_NUMBER 4;
}

VARIABLE LinkId
{
    LABEL "Link Id addressparameter of the Communication device";
    HELP "Link Id addressparameter of the Communication device";
    CLASS DEVICE;
    TYPE UNSIGNED_INTEGER (2)
    {
        MIN_VALUE 4096;
        MAX_VALUE 65535;
    }
    HANDLING READ & WRITE;
}

```

La sémantique des constructions EDDL affichées avec le code source EDDL ci-dessus est décrite dans l'IEC 62769-7. Le COMPONENT EDDL sera utilisé par le Serveur FDI et le Serveur de communication FDI pour créer une instance du type CommunicationServerType décrit dans l'IEC 62769-7.

Le LinkId VARIABLE contient la valeur d'adresse d'une instance d'appareil de communication. Dans le Modèle d'Information, le LinkId sera représenté comme une instance de BaseDataVariableType et comme un composant de ParameterSet de l'appareil de communication.

Conformément à l'IEC 62769-7, chaque Paquetage de communication FDI doit contenir au moins un élément EDD qui décrit au moins un composant Appareil de communication. Le code source EDDL suivant est un exemple qui décrit un appareil de communication.

```

COMPONENT Foundation_H1_Communication_Device
{
  LABEL "Foundation H1 communication device";
  CAN_DELETE TRUE;
  CLASSIFICATION NETWORK_COMPONENT;
  COMPONENT_RELATIONS
  {
    Foundation_H1_Service_Provider_Relation
  }
}

COMPONENT_RELATION Foundation_H1_Service_Provider_Relation
{
  LABEL "Foundation H1 communication service provider";
  RELATION_TYPE CHILD_COMPONENT;
  COMPONENTS
  {
    Foundation_H1_Service_Provider{AUTO_CREATE 1;}
  }
  MINIMUM_NUMBER 1;
  MAXIMUM_NUMBER 16;
}

```

La sémantique des constructions EDDL affichées avec le code source EDDL ci-dessus est décrite dans l'IEC 62769-7. Le COMPONENT EDDL sera utilisé par le Serveur FDI et le Serveur de communication FDI pour créer une instance du type ServerCommunicationDeviceType décrit dans l'IEC 62769-7.

Une instance de ServerCommunicationDeviceType doit contenir le ou les paramètres suivants avec son ParameterSet. Le Tableau 9 donne la définition du ParameterSet de l'appareil de communication.

Tableau 9 – Définition du ParameterSet de l'appareil de communication

Attribut	Valeur				
BrowseName	ParameterSet				
Références	NodeClass	BrowseName	Data Type	TypeDefinition	ModellingRule
Voir l'IEC 62541-100:–, 5.2.					
HasTypeDefinition	ObjectType	BaseObjectType			
HasComponent	Variable	LinkId	UInt16	BaseDataVariableType	Obligatoire
HasComponent	Variable	<ParameterIdentifier>		BaseDataVariableType	Obligatoire-Espace réservé

Le paramètre LinkId est un paramètre d'adressage qui permet de distinguer des liaisons H1 multiples. Si un Serveur de communication FDI prend en charge des liaisons H1 physiques multiples, celles-ci sont mappées dans le Modèle d'Information aux multiples instances d'appareil de communication. Si le Serveur de communication FDI ne prend en charge qu'une liaison H1, il doit définir un seul appareil de communication dans le Modèle d'Information. Dans ce cas, la valeur de la variable peut être mise sur 0.

La déclaration EDD de la variable LinkId est avec l'attribut ADDRESSING de la COMPONENT_RELATION de la définition du Serveur de communication FDI (voir 5.5.2).

5.5.3 Définition du fournisseur de service de communication

Conformément à l'IEC 62769-7, chaque Paquetage de communication FDI doit contenir au moins un élément EDD qui décrit au moins un composant fournisseur de service de communication. Le code source EDDL ci-après est un exemple décrivant un composant fournisseur de service communication CP 1/1.

La référence du composant ConnectionPoint_Foundation_H1 correspond à la définition du Point de connexion relatif en 5.5.1.

```
COMPONENT Foundation_H1_Service_Provider
{
    LABEL "Foundation H1 communication service provider";
    CAN_DELETE FALSE;
    CLASSIFICATION NETWORK_COMMUNICATION_SERVICE_PROVIDER;
    COMPONENT_RELATIONS
    {
        Foundation_H1_Connection_Point_Relation
    }
    BYTE_ORDER BIG_ENDIAN;
}

COMPONENT_RELATION
Foundation_H1_Service_Provider_Connection_Point_Relation
{
    LABEL "Relation between communication service provider and
connection point";
    RELATION_TYPE CHILD_COMPONENT;
    ADDRESSING {Address}
    COMPONENTS
    {
        ConnectionPoint_Foundation_H1{ AUTO_CREATE 1;}
    }
    MINIMUM_NUMBER 1;
    MAXIMUM_NUMBER 1;
}
```

La sémantique des constructions EDDL affichées avec le code source EDDL ci-dessus est décrite dans l'IEC 62769-7. Le COMPONENT EDDL sera utilisé par le Serveur FDI et le Serveur de communication FDI pour créer une instance du type ServerCommunicationServiceType décrit dans l'IEC 62769-7.

5.5.4 Définition du réseau

Conformément à l'IEC 62769-7, chaque Paquetage de communication FDI doit contenir au moins un élément EDD qui décrit un réseau pour chaque protocole pris en charge par l'Appareil de communication. La définition prend en charge l'ingénierie topologique du réseau.

```
COMPONENT Network_Foundation_H1
{
```

```

    LABEL "Foundation H1 Network";
    CAN_DELETE TRUE;
    CLASSIFICATION NETWORK;
    PROTOCOL Foundation_H1
    COMPONENT_RELATIONS
    {
        Foundation_H1_Network_Connection_Point_Relation
    }

COMPONENT_RELATION Foundation_H1_Network_Connection_Point_Relation
{
    LABEL "Relation between network and connection point";
    RELATION_TYPE CHILD_COMPONENT;
    ADDRESSING {Address}
    COMPONENTS
    {
        ConnectionPoint_Foundation_H1
    }
    MINIMUM_NUMBER 1;
    MAXIMUM_NUMBER 32;
}

```

La sémantique des constructions EDDL affichées avec le code source EDDL ci-dessus est décrite dans l'IEC 62769-7. Le COMPONENT EDDL sera utilisé par le Serveur FDI et le Serveur de communication FDI pour créer une instance du type NetworkType décrit dans l'IEC 62541-100.

5.6 Méthodes

5.6.1 Méthodes pour les Serveurs de communication FDI

5.6.1.1 Généralités

Le Serveur de communication FDI doit mettre en œuvre les services selon les signatures de méthode décrites en 5.6.1 et conformément au Modèle d'Information.

5.6.1.2 Connect

Le Tableau 10 répertorie les arguments de la méthode Connect.

Signature:

```

Connect (
    [in]   ByteString      CommunicationRelationId,
    [in]   UInt16          LinkId
    [in]   byte            Address,
    [in]   Int32           OrdinalNumber
    [in]   Boolean         SIFConnection
    [in]   UInt32          ServiceID,
    [out]  UInt32           DelayForNextCall,
    [out]  Int32           ServiceError);

```

Tableau 10 – Arguments de la méthode Connect

Argument	Description
CommunicationRelationId	La valeur d'argument contient le nodeId du ConnectionPoint de l'appareil qui représente la connexion entre un appareil et un réseau physique directement connecté au matériel du Serveur de communication FDI. Le nodeId permet de trouver la relation directe parent-enfant.
LinkId	Le nom d'argument doit correspondre au BrowserName correspondant de la variable définie comme un composant d'une instance du type CommunicationServerChannelType (voir 5.5.3). La valeur d'argument est transmise par l'instance parente d'un CommunicationServerChannelType. La valeur peut être obtenue par la méthode Scan ou peut être directement configurée.
Address	Le nom d'argument doit correspondre au nom d'attribut correspondant défini pour le ConnectionPoint qui est décrit par un élément EDD correspondant. La valeur d'argument contient l'adresse de nœud de l'appareil.
OrdinalNumber	Le nom d'argument doit correspondre au nom d'attribut correspondant défini pour le ConnectionPoint qui est décrit par un élément EDD correspondant spécifié en 5.4.2. La valeur d'argument contient l'OrdinalNumber. OrdinalNumber est la position du VFD dans la liste SMIB VFD. Si une valeur 0 est transmise avec cet argument, le premier FB VFD est sélectionné.
SIFConnection	Le nom d'argument doit correspondre au nom d'attribut correspondant défini pour le ConnectionPoint qui est décrit par un élément EDD correspondant spécifié en 5.4.2. La valeur d'argument dénote si une connexion SIF est nécessaire ou non.
ServiceId	Le code de transaction de service établit la relation entre la demande de service et la réponse correspondante.
DelayForNextCall	La valeur spécifie un délai en ms pour limiter le cycle d'invocation EndConnect qui ne doit pas être plus rapide que la valeur spécifiée dans la valeur d'argument.
ServiceError	0: OK/fonction démarrée de façon asynchrone, le résultat doit être interrogé avec EndConnect 1: OK/exécution terminée, connexion établie avec succès -1: Echec de la connexion/annulée par l'appelant -2: Echec de l'appel/ID de service inconnu -3: Echec de la connexion/appareil introuvable -4: Echec de la connexion/adresse de nœud appareil non valide -5: Echec de la connexion/identification d'appareil non valide -6: Echec de la connexion/argument LinkId non valide -7: Echec de la connexion/argument OrdinalNumber non valide
NOTE L'IEC 62769-7 définit l'argument AddressData de la méthode Connect comme une matrice de variantes. Les arguments de l'adresse définis dans le tableau sont représentés comme des entrées de la matrice de variantes suivant l'ordre dans lequel ils sont spécifiés ci-dessus. L'IEC 62769-7 définit l'argument DeviceInformation comme une liste d'arguments spécifiques au protocole dans laquelle la méthode Connect stocke les données résultantes. L'argument DeviceInformation est défini comme une matrice de variantes. L'argument DeviceInformation n'est pas utilisé.	

5.6.1.3 Disconnect

Le Tableau 11 répertorie les arguments de la méthode Disconnect.

Signature:

```
Disconnect (
    [in]   ByteString      CommunicationRelationId,
    [out]  UInt32          ServiceError)
```

Tableau 11 – Arguments de la méthode Disconnect

Argument	Description
CommunicationRelationId	La valeur d'argument contient le nodeld du ConnectionPoint de l'appareil qui représente la connexion entre un appareil et un réseau physique directement connecté au matériel du Serveur de communication FDI. Le nodeld permet de trouver la relation directe parent-enfant.
ServiceError	1: OK/déconnexion effectuée avec succès -1: Echec de la déconnexion/aucune relation de communication existante -2: Echec de la déconnexion/identifiant de relation de communication non valide

5.6.1.4 Transfer

Le Tableau 12 répertorie les arguments de la méthode Transfer.

Signature

```
Transfer (
    [in]   ByteString      CommunicationRelationId,
    [in]   String          OPERATION,
    [in]   String          BlockTag,
    [in]   UInt32          INDEX,
    [in]   UInt32          SUB_INDEX,
    [in]   Byte[]          WriteData,
    [out]  Byte[]          ReadData,
    [in]   UInt32          ServiceId,
    [out]  UInt32          DelayForNextCall,
    [out]  Int32           ServiceError);
```

Tableau 12 – Arguments de la méthode Transfer

Argument	Description
CommunicationRelationId	La valeur d'argument contient le nodeld du ConnectionPoint représentant la connexion entre un appareil et un réseau physique au sein du Modèle d'Information.
OPERATION	La valeur d'argument indique l'opération de transfert de données. Les valeurs permises sont "READ", "WRITE" et "VIEW_READ"
BlockTag	L'argument dénote l'étiquette du bloc de l'instance de bloc en cours d'adressage. La valeur peut être obtenue par la méthode Scan.
INDEX	OPERATION indique "READ" ou "WRITE": L'argument dénote l'index relatif du paramètre de bloc en cours d'adressage. L'index relatif peut être calculé en répétant les paramètres du bloc dans le Modèle d'Information FDI. Une valeur 0 adresse l'enregistrement d'en-tête du bloc (décrite par un attribut CHARACTERISTICS dans l'EDD). Le premier paramètre est adressé dans l'INDEX1. OPERATION indique "VIEW_READ": L'argument dénote l'identifiant de vue dans une plage de 1 à 4. Par exemple, une valeur de 1 demande que View_1 doit être read. Les objets Multiple View_3 ou View_4 sont identifiés par l'argument INDEX et SUB_INDEX.
SUB_INDEX	OPERATION indique "READ" ou "WRITE": L'argument dénote le sous-index d'un membre du paramètre de bloc en cours d'adressage si le paramètre de bloc est de type RECORD ou ARRAY. Pour adresser un paramètre simple, une valeur de 0 doit être transmise avec cet argument. Pour adresser un membre spécifique d'un paramètre de type RECORD ou ARRAY, une valeur relative à 1 doit être transmise avec cet argument. Pour adresser le paramètre de type RECORD ou ARRAY en tant qu'ensemble, une valeur de 0 doit être transmise avec cet argument. OPERATION indique "VIEW_READ": L'argument adresse un View_3 ou View_4 si des vues multiples de ce type existent. L'argument doit être 0 s'il n'y a pas de vues multiples du type adressé avec l'argument INDEX. Les valeurs de 1 jusqu'au nombre de vues de ce type adressent la vue spécifique.
WriteData	Ecrire les données codées comme matrice d'octets. Le codage des entiers doit suivre les règles définies dans l'IEC 62541-6. L'argument doit être ignoré si OPERATION indique un transfert de lecture ou une vue transfert de lecture.
ReadData	Avec cet argument, le flux d'octets des données de lecture est retourné comme matrice d'octets. Le codage des entiers doit suivre les règles définies dans l'IEC 62541-6. L'argument doit être ignoré si OPERATION indique un transfert d'écriture.
ServiceId	Le code de transaction de service établit la relation entre la demande de service et la réponse correspondante.
DelayForNextCall	La valeur spécifie un délai en ms pour limiter le cycle d'invocation EndTransfer qui ne doit pas être plus rapide que la valeur spécifiée dans la valeur d'argument.

Argument	Description
ServiceError	0: OK/fonction démarrée de façon asynchrone, le résultat doit être interrogé avec EndTransfer 1: OK/exécution terminée, ReceivedData contient le résultat -1: Echec du transfert/annulé par l'appelant -2: Echec de l'appel/ID de service inconnu -3: Echec du transfert/aucune relation de communication existante -4: Echec du transfert/identifiant de relation de communication non valide -5: Echec du transfert/contenu sendData non valide -6: Echec du transfert/format receiveData non valide -7: Echec du transfert/vérification de paramètre ^a -8: Echec du transfert/dépasse les limites du paramètre ^a -9: Echec du transfert/mode erroné pour la demande ^a -10: Echec du transfert/l'écriture est interdite par le commutateur de verrouillage de l'écriture ou le bloc fonctionnel de verrouillage de l'écriture pour les appareils SIS ^a -11: Echec du transfert/la valeur de données ne permet jamais l'écriture ^a -12: Echec du transfert/BlockTag en double détecté -13: INDEX non valide, argument SUB_INDEX fourni avec un transfert "VIEW_READ".
Le Serveur FDI gère un Modèle d'Information défini dans l'IEC 62541-100. Par conséquent, les éléments topologiques qui représentent un FFBlockType sont séparés des instances de bloc réelles. Une instance appelée Blocks d'un ConfigurableObjectType est utilisée pour mettre en œuvre les règles d'instanciation. L'instanciation des blocs est abordée avec plus de détails dans l'IEC 62769-5. Conformément aux règles définies dans l'IEC 62769-5, le Serveur FDI a besoin de recueillir les informations de l'objet FF Directory afin de pouvoir créer les instances de blocs. Les informations doivent être fournies par la méthode Scan définie en 5.6.1.7. Conformément à l'IEC 62769-5, l'argument BlockTag mentionné ci-dessus est obtenu à partir de l'attribut DisplayName de l'instance de bloc correspondante dans le Modèle d'Information FDI. NOTE 1 L'IEC 62769-7 définit l'argument SendData de la méthode Transfer comme une matrice de variantes. Les arguments OPERATION, BlockTag, INDEX, SUB_INDEX et WriteData définis dans le tableau sont représentés comme des entrées de la matrice de variantes suivant l'ordre dans lequel ils sont spécifiés ci-dessus. NOTE 2 L'IEC 62769-7 définit l'argument ReceiveData de la méthode Transfer comme une matrice de variantes. L'argument ReadData défini dans le tableau est représenté comme une entrée de la matrice de variantes suivant l'ordre dans lequel il est spécifié ci-dessus. NOTE 3 Exemple (pour clarification): Un bloc a deux vues de type View_4. La première vue du type View_4 est adressée avec les arguments INDEX = 4 et SUB_INDEX = 1. La deuxième vue est adressée avec les arguments INDEX = 4 et SUB_INDEX = 2.	
^a Une valeur ServiceError peut être retournée avec une opération d'écriture.	

5.6.1.5 GetPublishedData

Les alertes CP 1/1 représentent les messages non sollicités, comme définis dans l'IEC 62769-7. Le Tableau 13 répertorie les arguments de la méthode GetPublishedData.

NOTE CP 1/1 utilise le terme "alertes" pour désigner les alarmes et messages d'événements. Il s'agit de messages non sollicités asynchrones qui fournissent des notifications de changement d'état telles que les conditions de diagnostic. Ces messages sont mappés avec le service GetPublishData. CP 1/1 utilise également le terme "publier" pour désigner des communications réseau synchrones programmées pour le traitement des valeurs. Ces messages publiés ne sont pas mappés avec le service GetPublishedData.

Signature:

```

GetPublishedData (
    [in] ByteStringCommunicationRelationId,
    [out] String BlockTag,
    [out] Byte[] AlarmEventData,
    [out] NodeIdAlarmEventType
    [out] DateTimeTimeStamp
    [in] UInt32 ServiceId,
    [out] UInt32 DelayForNextCall,

```

```
[out] Int32 ServiceError);
```

Tableau 13 – Arguments de la méthode GetPublishedData

Argument	Description
CommunicationRelationId	La valeur d'argument contient le nodeld du ConnectionPoint représentant la connexion entre un appareil et un réseau physique au sein du Modèle d'Information.
BlockTag	L'argument de sortie dénote l'étiquette du bloc de l'instance de bloc qui a émis l'alarme ou l'événement.
AlarmEventData	Avec cet argument, le flux de bits de données d'alarme / événement est retourné comme matrice de bits. Le codage des entiers doit suivre les règles définies dans l'IEC 62541-6.
AlarmEventType	Nodeld de l'alarme ou nœud de type d'événement défini dans le Modèle d'Information FDI pour décoder le flux de données d'alarme/événement. Les types d'alarmes et d'événements doivent être lus à partir de l'EDD par le Serveur FDI lors de la création du Modèle d'Information.
TimeStamp	Heure à laquelle l'alarme ou l'événement a été détecté(e) par l'appareil.
ServiceId	Le code de transaction de service établit la relation entre la demande de service et la réponse correspondante.
DelayForNextCall	La valeur spécifie un délai en ms pour limiter le cycle d'invocation EndGetPublishedData qui ne doit pas être plus rapide que la valeur spécifiée dans la valeur d'argument.
ServiceError	0: OK / fonction démarrée de façon asynchrone, le résultat doit être interrogé avec EndGetPublishedData 1: OK/exécution terminée, ReceivedData et TimeStamp contiennent le résultat -1: Echec de GetPublishedData/annulé par l'appelant -2: Echec de l'appel/ID de service inconnu -3: Echec de GetPublishedData/non pris en charge -4: Echec de GetPublishedData/aucune relation de communication existante -5: Echec de GetPublishedData/identifiant de relation de communication non valide -8: Echec de GetPublishedData/aucune donnée d'alarme ou d'événement publiée -9: Echec de GetPublishedData/AlarmEventType non valide

Le Serveur FDI gère un Modèle d'Information défini dans l'IEC 62541-100. Par conséquent, les éléments topologiques qui représentent un FFBlockType sont séparés des instances de bloc réelles. Une instance appelée Blocks d'un ConfigurableObjectType est utilisée pour mettre en œuvre les règles d'instanciation. L'instanciation des blocs est abordée avec plus de détails dans l'IEC 62769-5. Conformément aux règles définies dans l'IEC 62769-5, le Serveur FDI a besoin de recueillir les informations de l'objet FF Directory afin de pouvoir créer les instances de blocs. Ces informations doivent être fournies par la méthode Scan définie en 5.6.1.7. Conformément à l'IEC 62769-5, l'argument BlockTag mentionné ci-dessus est obtenu à partir de l'attribut DisplayName de l'instance de bloc correspondante dans le Modèle d'Information FDI.

Une valeur ServiceError peut être retournée avec une opération d'écriture.

NOTE 1 L'IEC 62769-7 définit l'argument ReceiveData de la méthode GetPublishedData comme une matrice de variantes. Les arguments BlockTag, AlarmEventData et AlarmEventType définis dans le tableau sont représentés comme des entrées de la matrice de variantes suivant l'ordre dans lequel ils sont spécifiés ci-dessus.

NOTE 2 L'IEC 62769-7 définit l'argument SendData de la méthode Transfer comme une matrice de variantes. Les arguments OPERATION, BlockTag, INDEX, SUB_INDEX et WriteData définis dans le tableau sont représentés comme des entrées de la matrice de variantes suivant l'ordre dans lequel ils sont spécifiés ci-dessus.

NOTE 3 L'IEC 62769-7 définit l'argument ReceiveData de la méthode Transfer comme une matrice de variantes. L'argument ReadData défini dans le tableau est représenté comme une entrée de la matrice de variantes suivant l'ordre dans lequel il est spécifié ci-dessus.

5.6.1.6 SetAddress

Le Tableau 14 répertorie les arguments de la méthode SetAddress.

Signature

```
SetAddress (
    [in] String      OPERATION
    [in] UInt16      LinkId
    [in] byte        OldAddress,
    [in] byte        NewAddress,
    [in] String      NewPDTag
    [in] UInt32      ServiceId,
    [out] UInt32      DelayForNextCall,
    [out] Int32       ServiceError);
```

Tableau 14 – Arguments de la méthode SetAddress

Argument	Description
OPERATION ^a	La valeur d'argument indique le type d'opération d'adressage. Les valeurs permises sont "SETASSIGNMENT", "CLEARASSIGNMENT". Les valeurs d'argument données avec les arguments ci-dessous peuvent être ignorées suivant la valeur de l'argument OPERATION.
LinkId ^a	Le nom d'argument doit correspondre au BrowserName correspondant de la variable définie comme un composant d'une instance du type CommunicationServerChannelType (voir 5.5.3). La valeur d'argument est transmise par l'instance parente d'un CommunicationServerChannelType. La valeur peut être obtenue par la méthode Scan ou peut être directement configurée.
OldAddress ^a	La valeur d'argument contient l'adresse actuelle de l'appareil. Les valeurs permises sont 16...255.
NewAddress ^b	La valeur d'argument contient la nouvelle adresse d'un appareil. Les valeurs permises sont 0 et 16...247. La valeur est 0 si le service n'est pas utilisé pour modifier l'adresse de l'appareil H1. La valeur d'argument est ignorée si la valeur d'argument OPERATION est "CLEARASSIGNMENT".
NewPDTag ^b	La valeur d'argument contient le nouveau PD-Tag à définir pour l'appareil. La valeur d'argument est ignorée si la valeur d'argument OPERATION est "CLEARASSIGNMENT".
ServiceId	Le code de transaction de service établit la relation entre la demande de service et la réponse correspondante.
DelayForNextCall	La valeur spécifie un délai en ms pour limiter le cycle d'invocation EndTransfer qui ne doit pas être plus rapide que la valeur spécifiée dans la valeur d'argument.
ServiceError	0: OK/fonction démarrée de façon asynchrone, le résultat doit être interrogé avec EndSetAddress 1: OK/exécution terminée avec succès -1: Echec de SetAddress/annulé par l'appelant -2: Echec de l'appel/ID de service inconnu -3: Echec de SetAddress/non initialisé -4: Echec de SetAddress/non connecté à un réseau -5: Echec de SetAddress/aucun appareil correspondant à oldAddress n'a été trouvé -6: Echec de SetAddress/erreur d'adresse en double -7: Echec de SetAddress/l'appareil n'a pas accepté la nouvelle adresse -8: Echec de SetAddress/oldAddress non valide (en termes de syntaxe, de type de données, de format des données, etc.) -9: Echec de SetAddress/newAddress non valide (en termes de syntaxe, de type de données, de format des données, etc.) -10: Echec de SetAddress/impossible à l'état connecté
^a L'IEC 62769-7 définit l'argument OldAddress de la méthode SetAddress comme une matrice de variantes. Les arguments OPERATION, OldAddress et LinkId définis dans le tableau sont représentés comme des entrées de la matrice de variantes suivant l'ordre dans lequel ils sont spécifiés ci-dessus.	
^b L'IEC 62769-7 définit l'argument NewAddress de la méthode SetAddress comme une matrice de variantes. Les arguments NewAddress et NewPDTag définis dans le tableau sont représentés comme des entrées de la matrice de variantes suivant l'ordre dans lequel ils sont spécifiés ci-dessus.	

5.6.1.7 Scan

La signature de méthode spécifiée dans l'IEC 62769-7 s'applique. Le schéma topologyScanResult correspondant est spécifié à l'Annexe A.

5.6.1.8 ResetScan

La signature de méthode spécifiée dans l'IEC 62769-7 s'applique.

5.6.2 Méthodes pour les passerelles

Non prises en charge dans la présente norme.

Annexe A (normative)

Schéma de balayage topologique

A.1 Généralités

Le schéma du résultat de balayage topologique spécifié à l'Annexe A décrit l'argument `topologyScanResult` de la méthode `Scan` au format spécifique à CP 1/1. Le contenu et la structure du document XML doivent correspondre au concept conçu de Modèle d'Information pour décrire une topologie afin de permettre une concordance générique entre des appareils physiques connectés au réseau et le Modèle d'Information hébergé par le Serveur FDI.

A.2 FoundationH1AddressT

Un type simple qui définit la structure d'adresse pour CP 1/1.

Le schéma XML pour un type `FoundationH1AddressT` est:

```
<xsd:simpleTypeName="FoundationH1AddressT">
  <xsd:restriction base="xsd:unsignedByte">
    <xsd:minInclusive value="16"/>
    <xsd:maxInclusive value="255"/>
  </xsd:restriction>
</xsd:simpleType>
```

A.3 FoundationH1ConnectionPointT

Un type complexe qui définit le Point de connexion pour CP 1/1.

Le schéma XML pour un type `FoundationH1ConnectionPointT` est:

```
<xsd:complexType="FoundationH1ConnectionPointT">
  <xsd:sequence>
    <xsd:elementname="Identification"
      type="ff:FoundationIdentificationT"/>
    <xsd:elementname="BlockScanInstance"
      type="ff:FoundationBlockIdentificationT" minOccurs="0"
      maxOccurs="unbounded"/>
  </xsd:sequence>
  <xsd:attributename="Address" type="ff:FoundationH1AddressT"
    use="required"/>
  <xsd:attributename="SIFConnection" type="xsd:boolean"
    use="required"/>
  <xsd:attributename="OrdinalNumber" type="xsd:unsignedInt"
    use="required"/>
</xsd:complexType>
```

Les attributs d'un type `FoundationH1ConnectionPointT` sont décrits dans le Tableau A.15.

Tableau A.15 – Attributs de FoundationH1ConnectionPointT

Attribut	Description
Address	La valeur d'attribut contient l'adresse de l'appareil connecté au réseau.
SIFConnection	SIFConnection dénote si une connexion SIF est nécessaire ou non.
OrdinalNumber	La propriété OrdinalNumber reflète la position du VFD dans la liste VFD de gestion du système. Les VFD multiples sont mappés aux éléments ScanItem multiples.

Les éléments d'un type FoundationH1ConnectionPointT sont décrits dans le Tableau A.16.

Tableau A.16 – Eléments de FoundationH1ConnectionPointT

Élément	Description
Identification	Les données d'élément contiennent les données d'identification du type d'appareil. Par comparaison avec le Modèle d'Information (IEC 62769-5), le ConnectionPoint ne contient pas les données d'identification du type d'appareil et ne s'y réfère pas. Mais pour aider le système hôte FDI à trouver le paquetage qui correspond à l'appareil connecté, ce schéma associe l'identification du type d'appareil avec le ConnectionPoint.
BlockScanInstance	Informations d'instance de bloc du VFD d'appareil scanné. Utilisées pour créer les instances de bloc dans l'IM du Serveur FDI. Voir l'IEC 62769-5.

A.4 FoundationH1NetworkT

Un type complexe qui définit le réseau pour CP 1/1.

Le schéma XML pour un type FoundationH1NetworkT est:

```
<xsd:complexType name="FoundationH1NetworkT">
  <xsd:sequence>
    <xsd:elementname="ConnectionPoint"
      type="ff:FoundationH1ConnectionPointT"
      maxOccurs="unbounded"/>
  </xsd:sequence>
</xsd:complexType>
```

Les éléments d'un type FoundationH1NetworkT sont décrits dans le Tableau A.17.

Tableau A.17 – Eléments de FoundationH1NetworkT

Élément	Description
ConnectionPoint	Point de connexion CP 1/1

A.5 Network

L'élément racine utilisé pour retourner le résultat du balayage d'un réseau CP 1/1.

Le schéma XML pour un élément Network est:

```
<xsd:elementname="Network" type="ff:FoundationH1NetworkT"/>
```

A.6 FoundationBlockIdentificationT

Un type complexe qui définit les informations de l'instance de bloc de l'appareil scanné.

Le schéma XML pour un type FoundationBlockIdentificationT est:

```
<xsd:complexTypename="FoundationBlockIdentificationT">
  <xsd:attributename="BlockTag" use="required"/>
  <xsd:attributename="DDItem" use="required"/>
  <xsd:attributename="DirectoryPosition" use="required"/>
</xsd:complexType>
```

Les attributs d'un type FoundationBlockIdentificationT sont décrits dans le Tableau A.18.

Tableau A.18 – Attributs de FoundationBlockIdentificationT

Attribut	Description
BlockTag	L'attribut BlockTag doit être mappé au DisplayName d'une instance de bloc qui doit être créée dans l'IM du Serveur FDI.
DDItem	Cet attribut est utilisé pour trouver le type de bloc correct d'une instance de bloc qui doit être créée dans l'IM du Serveur FDI. Le type de bloc est recherché dans le dossier SupportedTypes dans le composant Blocs d'un DeviceType.
DirectoryPosition	Cet attribut dénote la position relative de l'instance de bloc dans l'objet Directory (répertoire). La première instance de bloc a une valeur de 0. Se reporter aux règles d'instanciation de bloc dans l'IEC 62769-5.

A.7 FoundationIdentificationT

Un type complexe qui définit le contenu correspondant à l'identification de FunctionalGroup.

Le schéma XML pour un type FoundationIdentificationT est:

```
<xsd:complexTypename="FoundationIdentificationT">
  <xsd:attributename="MANUFAC_ID" type="xsd:unsignedInt"
    use="required"/>
  <xsd:attributename="DEV_TYPE" type="xsd:unsignedShort"
    use="required"/>
  <xsd:attributename="DEV_REV" type="xsd:unsignedShort"
    use="required"/>
  <xsd:attributename="ITK_VER" type="xsd:unsignedShort"
    use="required"/>
  <xsd:attributename="HARDWARE_REV" type="xsd:string"
    use="optional"/>
  <xsd:attributename="SOFTWARE_REV" type="xsd:string"
    use="optional"/>
  <xsd:attributename="COMPATIBILITY_REV" type="xsd:unsignedInt"
    use="optional"/>
  <xsd:attributename="CAPABILITY_LEV" type="xsd:unsignedByte"
    use="optional"/>
  <xsd:attributename="SIF_ITK_VER" type="xsd:unsignedShort"
    use="optional"/>
  <xsd:attributename="FD_VER" type="xsd:unsignedShort"
    use="optional"/>
</xsd:complexType>
```

Les attributs d'un type FoundationIdentificationT sont décrits dans le Tableau A.19.

Tableau A.19 – Attributs de FoundationIdentificationT

Attribut	Description
MANUFAC_ID	Numéro d'identification du fabricant.
DEV_TYPE	Numéro de modèle du fabricant associé à la ressource.
DEV_REV	Numéro de révision du fabricant associé à la ressource.
ITK_VER	Numéro de profil ITK.
HARDWARE_REV	Révision de matériel du fabricant.
SOFTWARE_REV	Révision de logiciel du fabricant.
COMPATIBILITY_REV	Ce paramètre est utilisé en option lors du remplacement des appareils de terrain. Pour une utilisation correcte de ce paramètre, il convient que la valeur COMPATIBILITY_REV de l'appareil de remplacement soit égale ou inférieure à la valeur DEV_REV de l'appareil remplacé.
CAPABILITY_LEV	Ce paramètre peut être inclus dans un appareil pour indiquer le niveau de capacité pris en charge par un appareil.
SIF_ITK_VER	Numéro de profil SIF ITK.
FD_VER	Un paramètre égal à la valeur de la version majeure de la spécification des diagnostics de terrain pour laquelle cet appareil a été conçu.

Annexe B (normative)

Paramètres du service Transfer

B.1 Généralités

Les services d'accès direct spécifiés dans l'IEC 62769-2 permettent au Plugiciel d'Interface Utilisateur (UIP, *User Interface Plug-in*) d'échanger directement des données avec l'appareil. L'échange direct des données signifie que les données échangées entre un appareil et un UIP peuvent ne pas être reflétées dans le Modèle d'Information. L'interface IDirectAccess définie dans l'IEC 62769-6 correspond aux services d'accès direct définis dans l'IEC 62769-2. Les fonctions BeginTransfer et EndTransfer définies par l'interface IDirectAccess ont besoin d'acheminer des informations spécifiques à un protocole. Les spécificités d'un protocole doivent être saisies dans un document XML.

B.2 receiveData

Il s'agit d'un élément contenant les données retournées par le biais du transfert de l'argument receiveData défini par la fonction IDirectAccess.

Le schéma XML pour un élément reçu est:

```
<xsd:elementname="receiveData">
  <xsd:complexType>
    <xsd:complexContent>
      <xsd:extension base="ff:TransferResultDataT">
        <xsd:sequence>
          <xsd:elementname="ResponseCode" type="ff:ResponseCodeT"
            minOccurs="0"/>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
</xsd:element>
```

Les éléments d'un type receiveData sont décrits dans le Tableau B.1.

Tableau B.1 – Éléments de receiveData

Élément	Description
ResponseCode	Élément facultatif qui contient les valeurs de retour pour une réponse de service négative.

B.3 sendData

Il s'agit d'un élément contenant les données soumises par le biais du transfert de l'argument sendData défini par la fonction IDirectAccess.

Le schéma XML pour un élément sendData est:

```
<xsd:element name="sendData" type="ff:TransferSendDataT"/>
```

B.4 OperationT

Il s'agit d'un type simple qui définit les opérations de service.

Le schéma XML pour un type d'énumération OperationT est:

```
<xsd:simpleType name="OperationT">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="READ"/>
    <xsd:enumeration value="WRITE"/>
    <xsd:enumeration value="GETOD"/>
  </xsd:restriction>
</xsd:simpleType>
```

Les valeurs d'énumération d'un type d'énumération OperationT sont décrites dans le Tableau B.2.

Tableau B.2 – Enumérations d'OperationT

Enumération	Description
READ	Service de lecture conformément à l'IEC 61158-5-9:2007, 6.3.5.3.2
WRITE	Service d'écriture conformément à l'IEC 61158-5-9:2007, 6.3.5.3.3
GETOD	Service GetOD (forme longue) conformément à l'IEC 61158-5-9:2007, 6.3.2.3.2

B.5 ResponseCodeT

Il s'agit d'un type complexe qui définit les informations d'erreur de la réponse négative.

Le schéma XML pour un type ResponseCodeT est:

```
<xsd:complexType name="ResponseCodeT">
  <xsd:attributename="ErrorClass" type="xsd:unsignedShort"
    use="required"/>
  <xsd:attributename="AdditionalCode" type="xsd:short"
    use="optional"/>
  <xsd:attributename="AdditionalDescription" type="xsd:string"
    use="optional"/>
</xsd:complexType>
```

Les attributs d'un type ResponseCodeT sont décrits dans le Tableau B.3.

Tableau B.3 – Attributs de ResponseCodeT

Attribut	Description
ErrorClass	Classe d'erreurs signalées par la réponse de service négative.
AdditionalCode	Code de raison facultatif fourni par l'application de bloc fonctionnel.
AdditionalDescription	Description textuelle facultative de la réponse de service négative.

B.6 TransferResultDataT

Il s'agit d'un type complexe qui définit le format des données du paramètre de service qui doit être appliqué pour transférer la valeur de retour receivedData définie.

Le schéma XML pour un type TransferResultDataT est:

```
<xsd:complexType name="TransferResultDataT">
  <xsd:attributename="DATA" type="xsd:hexBinary" use="optional"/>
</xsd:complexType>
```

Les attributs d'un type TransferResultDataT sont décrits dans le Tableau B.4.

Tableau B.4 – Attributs de TransferResultDataT

Attribut	Description
DATA	Donnée reçue après l'opération de service. Cet attribut est nécessaire pour les opérations de service GETOD et READ.

B.7 TransferSendDataT

Il s'agit d'un type complexe qui définit le format des données du paramètre de service qui doit être appliqué pour transférer l'argument SendData défini.

Le schéma XML pour un type TransferSendDataT est:

```
<xsd:complexType name="TransferSendDataT">
  <xsd:attributename="OPERATION" type="ff:OperationT"
    use="required"/>
  <xsd:attributename="OD_INDEX" type="xsd:unsignedInt"
    use="required"/>
  <xsd:attributename="OD_SUB_INDEX" type="xsd:unsignedInt"
    use="required"/>
  <xsd:attributename="DATA" type="xsd:hexBinary" use="optional"/>
</xsd:complexType>
```

Les attributs d'un type TransferSendDataT sont décrits dans le Tableau B.5.

Tableau B.5 – Attributs de TransferSendDataT

Attribut	Description
OPERATION	Opération de transfert de données.
OD_INDEX	Index du dictionnaire d'objets relatif à l'application du bloc fonctionnel.
OD_SUB_INDEX	Pour adresser un paramètre simple, une valeur de 0 doit être transmise avec cet attribut. Pour adresser un membre spécifique d'un paramètre de type RECORD ou ARRAY, une valeur relative à 1 doit être transmise avec cet attribut. Pour adresser le paramètre de type RECORD ou ARRAY en tant qu'ensemble, une valeur de 0 doit être transmise avec cet attribut.
DATA	Attribut obligatoire pour les données à transférer dans l'opération WRITE du service. Cet attribut n'est pas utilisé pour l'opération du service READ ou GETOD.

Annexe C (informative)

Arguments du service de communication pour la méthode Transfer

L'IEC 62769-3 décrit la manière dont ces arguments du service de communication pour la méthode Transfer (voir 5.6.1.4) sont obtenus des éléments COMMAND associés à l'élément VARIABLE. Pour l'accès de lecture ou d'écriture à la variable, le Serveur FDI doit obtenir la description COMMAND relative et obtenir les arguments du service de communication pour la méthode Transfer des attributs de la description COMMAND par la mise en correspondance des noms.

Etant donné que le profil EDD CPF1 ne fournit pas d'élément EDD COMMAND, cette approche requiert des considérations supplémentaires. Pour garder le Serveur FDI aussi générique que possible, la solution suivante est proposée.

La construction COMMAND est introduite mais à un niveau virtuel uniquement. Cela signifie qu'il n'y aura aucun moyen de définir l'élément COMMAND dans la grammaire du profil EDD CPF1.

L'élément COMMAND aura les attributs suivants:

- INDEX
- SUB_INDEX

L'élément COMMAND doit être relié au PARAMETER du bloc.

La représentation de l'élément COMMAND sera créée automatiquement (à la volée) pour chaque paramètre de bloc par la machine FDI (EDD) quand un bloc est chargé par la machine.

NOTE Avec les services EDD d'aujourd'hui, la création des éléments COMMAND pour des PARAMETER peut être effectuée pendant `ddi_get_item()` pour le bloc lorsque la liste des paramètres a été créée.

Pour les paramètres de type RECORD ou ARRAY, les éléments DD COMMND doivent être créés pour chaque membre de RECORD ou ARRAY. Ceci est destiné à faciliter le fonctionnement du Serveur FDI.

Lorsque l'EDD est chargé par le Serveur FDI, les BlockTypes sont créés dans le dossier SupportedTypes dans le composant Blocks du DeviceType (voir l'IEC 62769-5 et l'IEC 62541-100). Pour chaque paramètre de BlockType, la description COMMAND est fournie par la machine FDI (EDD).

Les instances de bloc sont créées avec le résultat de la méthode Scan, comme décrit en 5.6.1.7. Les instances de bloc sont créées comme composants (enfants) du composant Blocks. Conformément à l'IEC 62769-5, le DisplayName de l'instance de bloc est le BlockTag.

Pour l'accès de lecture ou d'écriture à la variable, le Serveur FDI doit obtenir les arguments du service de communication INDEX et SUB_INDEX de la description COMMAND du paramètre par la mise en correspondance des noms. Pour l'argument du service de communication BlockTag, le DisplayName de l'instance de bloc doit être utilisé.

Bibliographie

IEC 62769-3:2015, *Intégration des appareils de terrain (FDI) – Partie 3: Serveur FDI*

NOTE L'IEC 62769-3 est techniquement identique à la FDI-2023

FIELDBUS FOUNDATION. *Foundation Specification Common File Format*. FF-103, Version 1.9.4, juin 2010 (disponible en anglais seulement)

FIELDBUS FOUNDATION. *Foundation Specification Function Block Application Process – Part 1*. FF-890, Version 1.10.2, août 2012 (disponible en anglais seulement)

FIELDBUS FOUNDATION. *Foundation Specification Function Block Application Process – Part 2*. FF-891, Version 1.10.2, août 2012 (disponible en anglais seulement)

FIELDBUS FOUNDATION. *FF-880: Foundation Specification System Management*. FF-980, Version 1.6. 1 décembre 2005 (disponible en anglais seulement)

INTERNATIONAL
ELECTROTECHNICAL
COMMISSION

3, rue de Varembé
PO Box 131
CH-1211 Geneva 20
Switzerland

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