

CONSOLIDATED VERSION

VERSION CONSOLIDÉE



**Field device tool (FDT) interface specification –
Part 301: Communication profile integration – IEC 61784 CPF 1**

**Spécification des interfaces des outils des dispositifs de terrain (FDT) –
Partie 301: Intégration des profils de communication – IEC 61784 CPF 1**



THIS PUBLICATION IS COPYRIGHT PROTECTED

Copyright © 2016 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 20 000 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

65 000 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 20 000 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

65 000 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.

CONSOLIDATED VERSION

VERSION CONSOLIDÉE



**Field device tool (FDT) interface specification –
Part 301: Communication profile integration – IEC 61784 CPF 1**

**Spécification des interfaces des outils des dispositifs de terrain (FDT) –
Partie 301: Intégration des profils de communication – IEC 61784 CPF 1**

INTERNATIONAL
ELECTROTECHNICAL
COMMISSION

COMMISSION
ELECTROTECHNIQUE
INTERNATIONALE

ICS 25.040.40; 35.100.05; 35.110

ISBN 978-2-8322-3416-7

**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.**

REDLINE VERSION

VERSION REDLINE



**Field device tool (FDT) interface specification –
Part 301: Communication profile integration – IEC 61784 CPF 1**

**Spécification des interfaces des outils des dispositifs de terrain (FDT) –
Partie 301: Intégration des profils de communication – IEC 61784 CPF 1**

CONTENTS

FOREWORD.....	5
INTRODUCTION.....	7
1 Scope	8
2 Normative references	8
3 Terms, definitions, symbols, abbreviated terms and conventions	9
3.1 Terms and definitions	9
3.2 Abbreviated terms	9
3.3 Conventions	9
3.3.1 Data type names and references to data types	9
3.3.2 Vocabulary for requirements.....	9
3.3.3 Use of UML.....	10
4 Fundamentals.....	10
4.1 System and FDT topology	10
4.2 FDT topology for H1 devices	10
4.3 FDT topology for HSE devices	11
4.4 Nested communication.....	13
5 Bus category	14
6 Access to instance and device data	14
6.1 DTM.....	14
6.2 BTM	15
7 Protocol specific behavior	15
7.1 Connection management	15
7.1.1 FMS connection	15
7.1.2 FDT connection.....	16
7.2 Abort.....	17
7.2.1 OnAbort Indication	17
7.2.2 Abort request.....	17
7.3 Relation of FMS requests and FMS responses	17
7.4 Subscription mechanism	19
7.4.1 General	19
7.4.2 Transactions for subscribing H1	19
7.4.3 Transactions for subscribing HSE.....	19
7.4.4 Transactions for subscribing BTM.....	20
8 Protocol specific usage of general data types.....	21
8.1 Address.....	21
8.2 protocolID	21
8.3 applicationDomain	21
8.4 semanticId.....	21
8.4.1 Block specific definitions	21
8.4.2 Fieldbus management definitions	22
8.4.3 Fieldbus specific definitions.....	22
9 Protocol specific data types	23
9.1 DTM.....	23
9.1.1 Topology scan definitions.....	23
9.1.2 Parameter access	23

9.1.3	FF device data types.....	29
9.2	BTM.....	30
9.2.1	General	30
9.2.2	Parameter access - FF specific definitions.....	30
10	Network management data types	44
10.1	General.....	44
10.2	H1 network management definitions.....	44
10.3	HSE network management data types.....	44
11	Communication data types.....	87
11.1	Common data types.....	87
11.2	FF FMS data types	91
11.3	H1 communication data types	97
11.4	HSE communication data types.....	104
11.5	FDT FF standard block communication data types.....	112
12	Channel parameter data types	114
13	Device identification	116
13.1	Protocol specific handling of data type STRING.....	116
13.2	Common device type identification data types	117
13.3	Scan identification data types	123
13.4	Device type identification data types – provided by DTM.....	123
Annex A (informative) Implementation hints.....		125
Annex B (normative) Levels of support		127
Bibliography		130
Figure 1 – Part 301 of the IEC 62453 series		7
Figure 2 – Object relations for H1 Device DTM		10
Figure 3 – Object relations for HSE application with DTMs and BTMs		12
Figure 4 – FMS mapping in the FDT connection		16
Figure 5 – FDT Disconnect service.....		16
Table 1 – Object relations for H1 Device DTM		11
Table 2 – Object relations for HSE application with DTMs and BTMs		13
Table 3 – FF specific protocol identifiers		14
Table 4 – Relation of FMS requests and FMS responses		18
Table 5 – Action object definitions (refer to FF-890):.....		24
Table 6 – Link object definitions		24
Table 7 – Alert object definitions		25
Table 8 – Trend object definitions		26
Table 9 – View definition.....		28
Table 10 – Domain object definitions.....		28
Table 11 – Program invocation object definitions		29
Table 12 – Structured FF device data types.....		30
Table 13 – Parameter mnemonic.....		31
Table 14 – Mnemonic of structured data types.....		39
Table 15 – Simple common data types		40

Table 16 – H1 Fieldbus Management data types.....	44
Table 17 – Simple HSE Fieldbus Management Definitions	44
Table 18 – Structured HSE Network management data types.....	50
Table 19 – Simple common data types	87
Table 20 – Structured common data types.....	88
Table 21 – Simple FF FMS data types.....	91
Table 22 – Structured FF FMS data types	93
Table 23 – Simple H1 communication data types.....	98
Table 24 – Structured H1 communication data types.....	99
Table 25 – Simple HSE communication data types	104
Table 26 – Structured HSE communication data types.....	105
Table 27 – Block communication data types	113
Table 28 – Simple FF channel data types	114
Table 29 – Structured FF channel data types.....	115
Table 30 – FieldbusFoundation H1 table	117
Table 31 – FieldbusFoundation HSE	119
Table 32 – FieldbusFoundation blocks.....	121
Table 33 – Simple Fieldbus Scan definitions.....	123
Table 34 – Device identification data types.....	124
Table 35 – Physical layer identifiers for H1.....	14
Table 36 – DataLink Layer Identifiers	14
Table B.1 – Levels of support.....	128

INTERNATIONAL ELECTROTECHNICAL COMMISSION

FIELD DEVICE TOOL (FDT) INTERFACE SPECIFICATION –

Part 301: Communication profile integration – IEC 61784 CPF 1

FOREWORD

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

DISCLAIMER

This Consolidated version is not an official IEC Standard and has been prepared for user convenience. Only the current versions of the standard and its amendment(s) are to be considered the official documents.

This Consolidated version of IEC 62453-301 bears the edition number 1.1. It consists of the first edition (2009-06) [documents 65E/125/FDIS and 65E/138/RVD] and its amendment 1 (2016-05) [documents 65E/336/CDV and 65E/395A/RVC]. The technical content is identical to the base edition and its amendment.

In this Redline version, a vertical line in the margin shows where the technical content is modified by amendment 1. Additions are in green text, deletions are in strikethrough red text. A separate Final version with all changes accepted is available in this publication.

International Standard IEC 62453-301 has been prepared by subcommittee 65E: Devices and integration in enterprise systems, of IEC technical committee 65: Industrial-process measurement, control and automation.

Each part of the IEC 62453-3xy series is intended to be read in conjunction with IEC 62453-2.

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

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

A list of all parts of the IEC 62453 series, under the general title *Field Device Tool (FDT) interface specification*, can be found on the IEC website.

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

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

IMPORTANT – The 'colour inside' logo on the cover page of this publication indicates that it contains colours which are considered to be useful for the correct understanding of its contents. Users should therefore print this document using a colour printer.

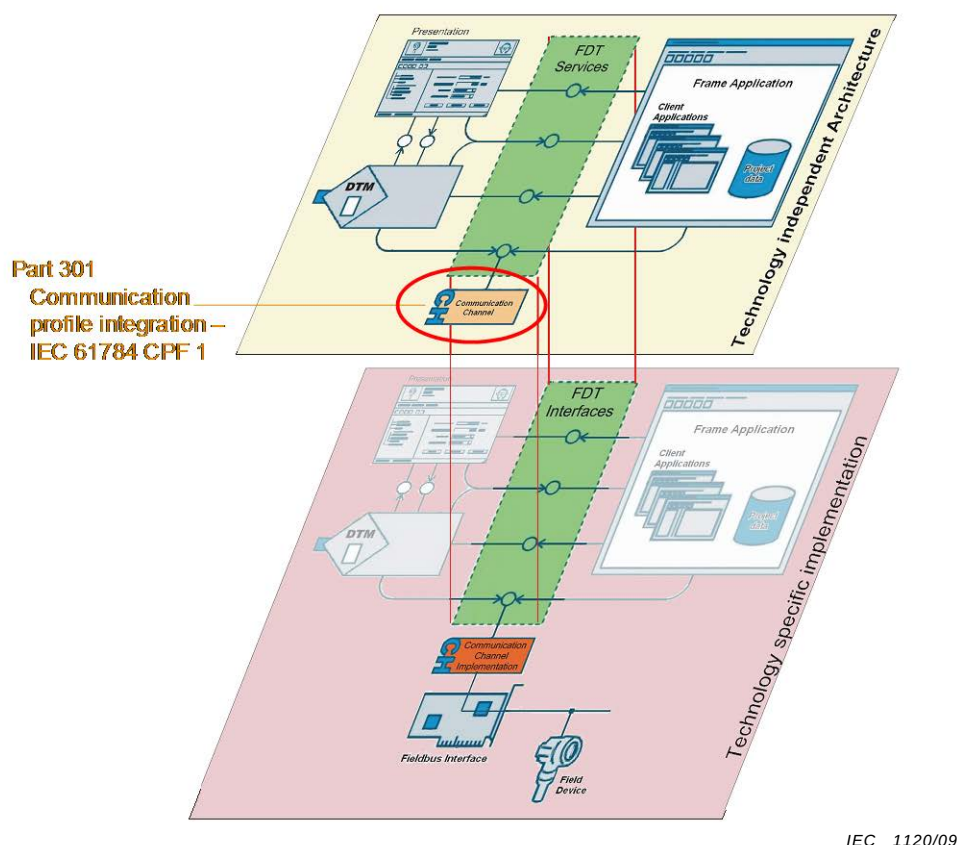
INTRODUCTION

This part of IEC 62453 is an interface specification for developers of FDT (Field Device Tool) components for function control and data access within a client/server architecture. The specification is a result of an analysis and design process to develop standard interfaces to facilitate the development of servers and clients by multiple vendors that need to interoperate seamlessly.

With the integration of fieldbusses into control systems, there are a few other tasks which need to be performed. In addition to fieldbus- and device-specific tools, there is a need to integrate these tools into higher-level system-wide planning- or engineering tools. In particular, for use in extensive and heterogeneous control systems, typically in the area of the process industry, the unambiguous definition of engineering interfaces that are easy to use for all those involved is of great importance.

A device-specific software component, called DTM (Device Type Manager), is supplied by the field device manufacturer with its device. The DTM is integrated into engineering tools via the FDT interfaces defined in this specification. The approach to integration is in general open for all kinds of fieldbusses and thus meets the requirements for integrating different kinds of devices into heterogeneous control systems.

Figure 1 shows how IEC 62453-301 is aligned in the structure of the IEC 62453 series.



IEC 1120/09

Figure 1 – Part 301 of the IEC 62453 series

FIELD DEVICE TOOL (FDT) INTERFACE SPECIFICATION –

Part 301: Communication profile integration – IEC 61784 CPF 1

1 Scope

Communication Profile Family 1 (commonly known as FOUNDATION™ Fieldbus¹) defines communication profiles based on IEC 61158-2, Type 1, IEC 61158-3-1, IEC 61158-4-1, IEC 61158-5-5, IEC 61158-5-9, IEC 61158-6-5, and IEC 61158-6-9. The basic profiles CP 1/1 (FF H1) and CP 1/2 (FF HSE) are defined in IEC 61784-1.

This part of IEC 62453 provides information for integrating the FOUNDATION™ Fieldbus (FF) protocol into the FDT standard (IEC 62453-2).

The standard describes communication definitions, protocol specific extensions and the means for block (e.g. transducer, resource or function blocks) representation.

The new protocol specific definitions are based on FF-specifications for H1 and HSE protocols. Furthermore, the definitions contain information that is needed by systems to configure FF devices.

The scope is limited to FOUNDATION™ Fieldbus device and system specific definitions.

2 Normative references

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

IEC 61158-2, *Industrial communication networks – Fieldbus specifications – Part 2: Physical layer specification and service definition*

IEC 61158-3-1, *Industrial communication networks – Fieldbus specifications – Part 3-1: Data-link layer service definition – Type 1 elements*

IEC 61158-4-1:2007, *Industrial communication networks – Fieldbus specifications – Part 4-1 Data-link layer protocol specification – Type 1 elements*

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

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

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

IEC 61158-6-5, *Industrial communication networks – Fieldbus specifications – Part 6-5: Application layer protocol specification – Type 5 elements*

IEC 61158-6-9, *Industrial communication networks – Fieldbus specifications – Part 6-9: Application layer protocol specification – Type 9 elements*

IEC 61784-1, *Industrial communication networks – Profiles – Part 1: Fieldbus profiles*

IEC 62453-1:2009, *Field Device Tool (FDT) interface specification – Part 1: Overview and guidance*

IEC 62453-2:2009, *Field Device Tool (FDT) interface specification – Part 2: Concepts and detailed description*

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

3 Terms, definitions, symbols, abbreviated terms and conventions

3.1 Terms and definitions

For the purposes of this document, the terms and definitions given in IEC 62453-1 and IEC 62453-2 apply.

3.2 Abbreviated terms

For the purposes of this document, the abbreviations given in IEC 62453-1, IEC 62453-2 and the following apply.

SM	System Management
FDA	Federation Drug Association
FF	FOUNDATION™ Fieldbus
FMS	Fieldbus Message Specification
DTM	Device Type Manager
BTM	Block Type Manager
H1	Low speed version of FF
HSE	High Speed Ethernet

3.3 Conventions

3.3.1 Data type names and references to data types

The conventions for naming and referencing of data types are explained in IEC 62453-2 Clause A.1

3.3.2 Vocabulary for requirements

The following expressions are used when specifying requirements.

Usage of “shall” or “mandatory”	No exceptions allowed.
Usage of “should” or “recommended”	Strong recommendation. It may make sense in special exceptional cases to differ from the described behaviour.
Usage of “can” or “optional”	Function or behaviour may be provided,

depending on defined conditions.

3.3.3 Use of UML

Figures in this standard are using UML notation as defined in Annex A of IEC 62453-1.

4 Fundamentals

4.1 System and FDT topology

This standard provides communication definitions, protocol-specific extensions and means for device and block (e.g. resource, transducer or function blocks) configuration.

The communication definitions provide System Management (SM) and Fieldbus Message Specification (FMS) communication.

Separate definitions are designed to support the different management parameters and structures for H1 and HSE devices.

Protocol-specific definitions can be used to identify FOUNDATION™ Fieldbus devices and their internal structure.

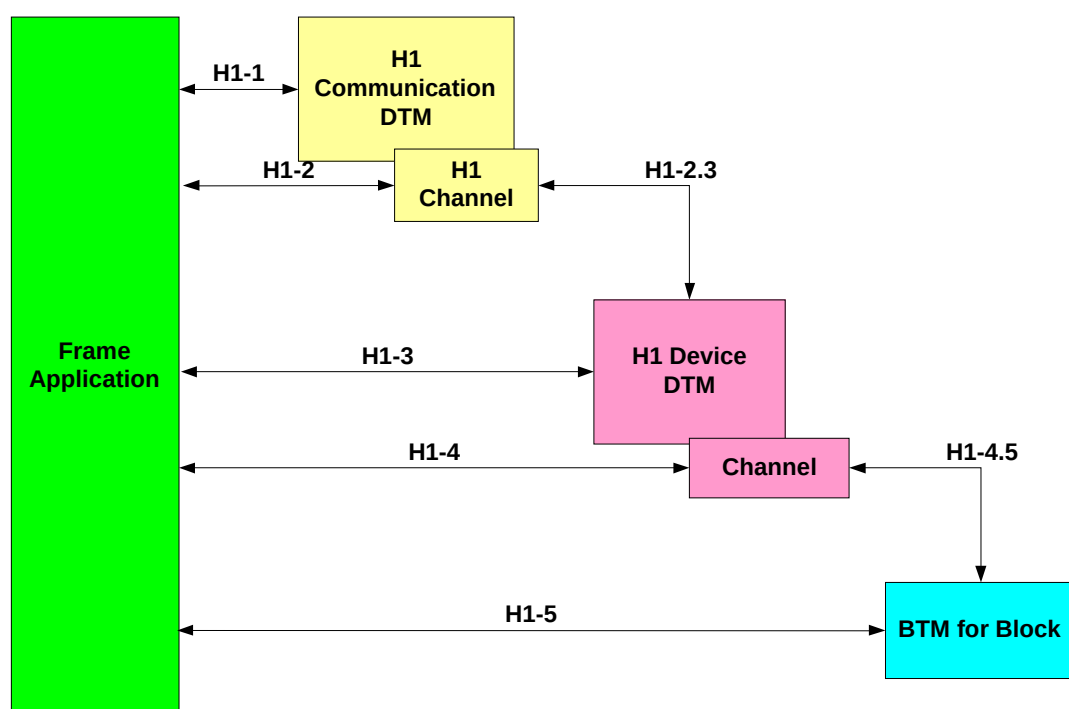
A FOUNDATION™ Fieldbus device is represented by a Device Type Manager (DTM) together with a group of Block Type Managers (BTM). Each BTM represents the functionality of a block functionality in an FF device.

4.2 FDT topology for H1 devices

An FF H1 topology may contain Communication DTM, Device DTM and BTMs.

EXAMPLE

The typical FDT topology for H1 devices is illustrated in Figure 2 and Table 1.



IEC 1121/09

Figure 2 – Object relations for H1 Device DTM

Table 1 – Object relations for H1 Device DTM

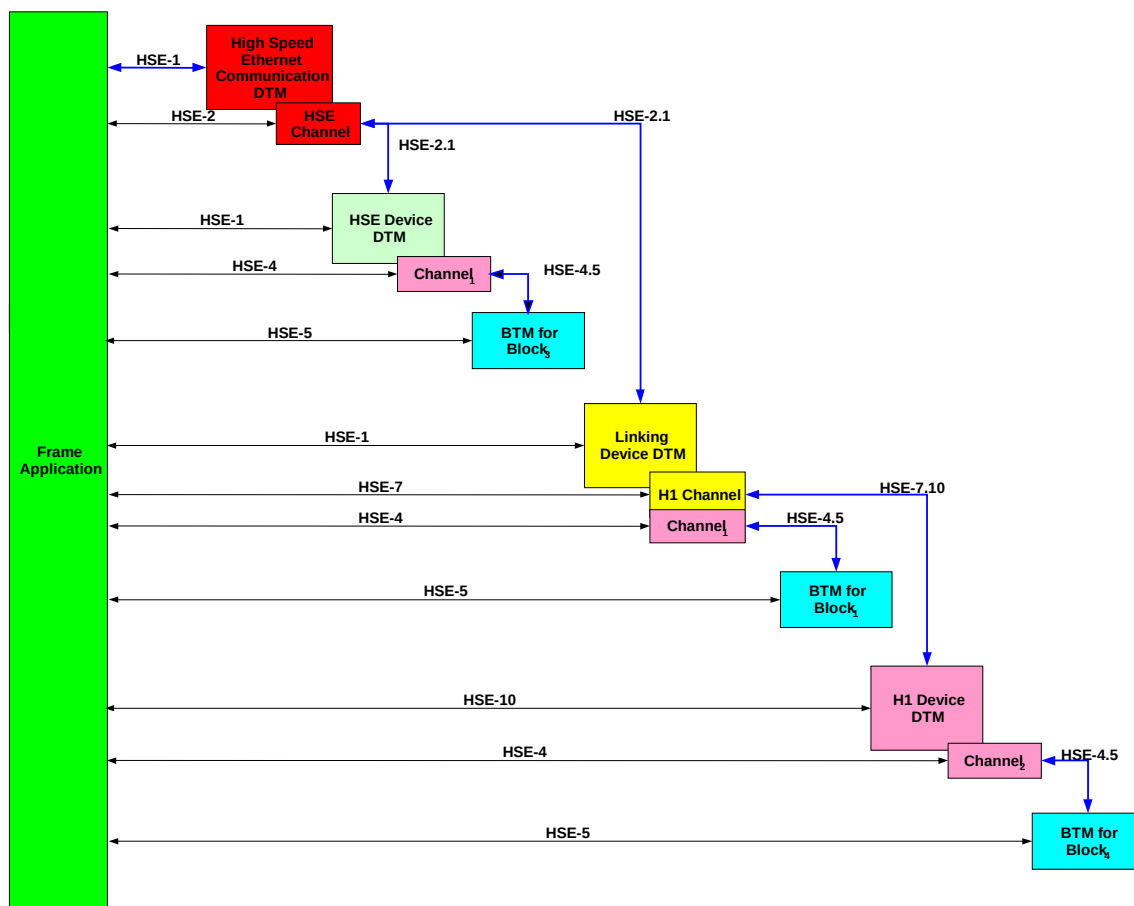
Relation	Type of information	Used definitions
H1-1	Management Parameter Access	H1 Management data types (see 10.2) FF common data types (see 11.1)
H1-2	Network Topology	FF common data types (see 11.1) Identification data types (see 13.4) Scan identification data types (see 13.3)
	Channel Parameter Access	Channel parameter data types (see 12)
H1-3	Management Parameter Access	H1 Management data types (see 10.2) FF common data types (see 11.1)
H1-4	List of instantiated blocks	FF common data types (see 11.1) BTM data types (see IEC 62453-2) FF block data types (see 9.2.2) Identification data types (see 13.4) Scan identification data types (see 13.3)
	Channel Parameter Access	Channel parameter data types (see 12)
H1-5	Block Information	BTM Information data types (see IEC 62453-2) BTM data types (see IEC 62453-2) Identification data types (see 13.4) Identification data types (see 13.4))
	Initialization of BTM	BTM Init data types (see IEC 62453-2) BTM data types (see IEC 62453-2)
	Parameter Access	BTM parameter data types (see IEC 62453-2) BTM data types (see IEC 62453-2)
H1-2.3	Communication	FF FMS data types (see 11.2) H1 communication data types (see 11.3) FF common data types (see 11.1)
H1-4.5	Block communication	FF FMS data types (see 11.2) FF Block communication data types (see 11.5) BTM data types (see IEC 62453-2) FF common data types (see 11.1)

4.3 FDT topology for HSE devices

An FF HSE topology may contain Communication DTM, GatewayDTM, Device DTM and BTMs.

EXAMPLE

The typical FDT topology for a HSE based system is illustrated by Figure 3 and Table 2:



IEC 1122/09

Figure 3 – Object relations for HSE application with DTMs and BTMs

In this illustration, blue lines show the object hierarchy as it is managed in the FDT Frame Application.

Table 2 – Object relations for HSE application with DTMs and BTMs

Relation	Type of information	Used definitions
HSE-1	Management Parameter Access	FF HSE management data types (see 10.3)
HSE-2	Network Topology	FF common data types (see 11.1) Identification data types (see 13.4) Scan identification data types (see 13.3)
	Channel Parameter Access	Channel parameter data types (see Clause 12)
HSE-2.1	Communication	FF FMS data types (see 11.2) FF HSE communication data types (see 11.4) FF common data types (see 11.1)
HSE-4.5	Block communication	FF FMS data types (see 11.2) FF Block communication data types (see 11.5) FF common data types (see 11.1) BTM data types (see IEC 62453-2)
HSE-4	List of instantiated blocks	FF common data types (see 11.1) BTM data types (see IEC 62453-2) Identification data types (see 13.4) Scan identification data types (see 13.3)
	Channel Parameter Access	Channel parameter data types (see Clause 12)
HSE-5	Block Information	BTM Information data types (see IEC 62453-2) BTM data types (see IEC 62453-2) Identification data types (see 13.4)
	Initialization of BTM	BTM Init data types (see IEC 62453-2) BTM data types (see IEC 62453-2)
	Parameter Access	BTM parameter data types (see IEC 62453-2) BTM data types (see IEC 62453-2)
HSE-7	Network Topology	FF common data types (see 11.1) FF HSE communication data types (see 11.4) FF Block communication data types (see 11.5)
	Channel Parameter Access	Channel parameter data types (see Clause 12) H1 Management data types (see 10.2)
	Parameter Access	BTM parameter data types (see IEC 62453-2)
HSE-10	Management Parameter Access	H1 Management data types (see 10.2) FF common data types (see 11.1)
HSE-7.10	Communication	FF FMS data types (see 11.2) H1 communication data types (see 11.3) FF common data types (see 11.1)

4.4 Nested communication

Standard FF blocks may be handled by BTMs implementing the standard behavior. These BTMs may be connected to the corresponding DTM that provides the support of an “FDT FF STANDARD BLOCK” communication protocol.

For device-specific BTMs, a device specific (it may be unique) bus category (CATID) shall be defined for the protocol between DTM and BTM.

The Frame Application can use this bus category to identify the device specific blocks and it can prevent a device-specific block from Device A to be assigned to a Device B that does not support the block.

Different communication protocols defined in this document (H1, HSE, Standard blocks) can use the same communication requests (FMS).

5 Bus category

FF protocols are identified in the protocolId element of the structured data type 'fdt:BusCategory' by the following unique identifiers (Table 3):

Table 3 – FF specific protocol identifiers

Protocol Id	ProtocolId name	Description
036D1691-387B-11D4-86E1-00E0987270B9	'FF H1'	Object supports FF H1 protocol
036D1692-387B-11D4-86E1-00E0987270B9	'FF HSE'	Object supports FF HSE protocol
036D1693-387B-11D4-86E1-00E0987270B9	'FF Standard Block'	For FDT FF standard block protocol

FF protocols are using the following unique identifiers in physicalLayer members within PhysicalLayer data type (Table 35 for H1):

Table 35 – Physical layer identifiers for H1

Identifier value	Name	Description
0D8FB517-1D8D-4455-9CE1-1B4A5DD4A0D2	FF H1	FOUNDATION Fieldbus™ H1 physical layer, as described in "FF 816 - 31,25 kbit/s Physical Layer"

Table 36 defines which DataLinkLayer shall be used in combination with the BusCategory values defined in Table 3.

Table 36 – DataLink Layer Identifiers

Identifier value	Name	Description
63D4C62E-91A9-4904-BCFC-3D3479C2EBAD	FDL	FF - 822 H1 Data Link Protocol and FF - 821 H1 Data Link Services

6 Access to instance and device data

6.1 DTM

The minimum set of provided data shall be:

- for FF devices System Management and Network Management parameters may be provided by the DTM as specified in Clause 10;
- the DTM shall provide support for the data described in 9.1.2;
- for each block in the device all standard block parameters shall be provided by the corresponding BTM object as specified in IEC 62453 2.

It is recommended that all non-standard block parameters are also provided as specified in 9.2.

A Device DTM shall provide access to the following groups of parameters if they are presented to the user:

- link parameters;
- action parameters;
- alert parameters;
- trend parameters;
- domain objects;
- program invocation objects.

6.2 BTM

BTMs built according to the function block specification shall expose all parameters for the corresponding block as defined for the DTM. All standard parameters defined for a block in the FF specification shall be exposed. The manufacturer specific parameters may be added to extend this list. User defined parameter shall be exposed if they exist.

7 Protocol specific behavior

7.1 Connection management

The Connect service establishes an FDT connection. The FDT connection acts as a container for FMS connections, as a container for SM connectionless services and as a container for FDA sessions.

It is necessary to maintain the FMS connection separate from the underlying FDT connection.

7.1.1 FMS connection

FDT implements all FMS and SM services as transactions in the respective protocols. This includes the services for the FMS connection management. FmsInitiate transaction and FmsAbort transaction manage the lifetime of a FMS connection.

In order to open multiple FMS connections on the same FDT connection, a Transaction service request is called with an FmsInitiateRequest element as argument.

The communication reference passed with the FmsInitiateRequest element identifies the FDT connection to be used. The FmsInitiateResponse element returned with Transaction service response provides a communication reference used for all further FMS services on that FMS connection.

This mapping allows multiple FMS connections on one FDT connection.

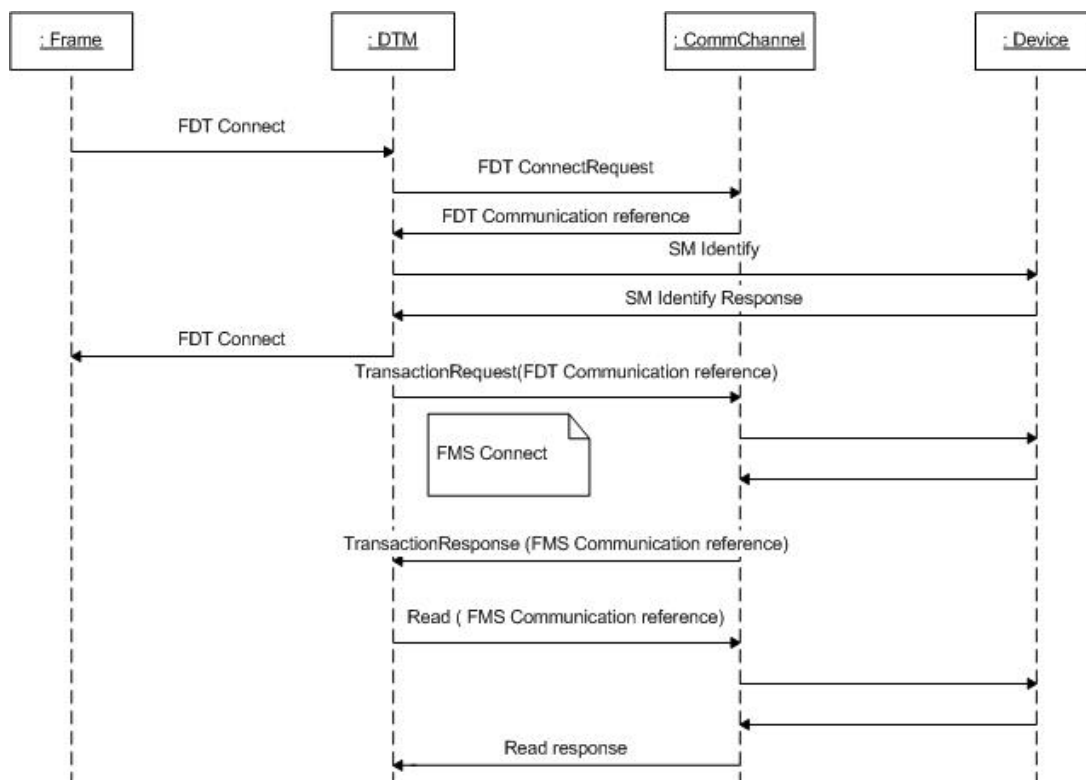


Figure 4 – FMS mapping in the FDT connection

IEC 1123/09

If the FDT connection is closed, the FMS connections for this connection are also closed automatically (by the communication channel) before the FDT Disconnect service is completed.

Any transactions requested after that will fail. No outstanding services will be processed.

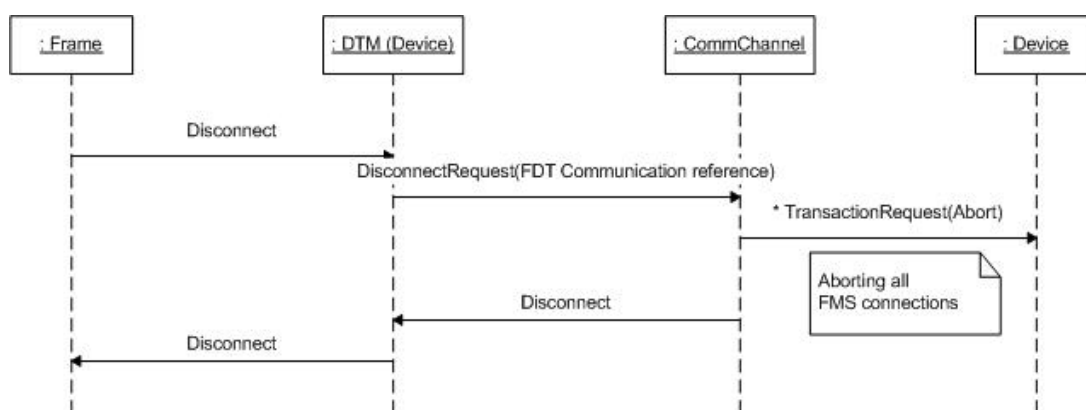


Figure 5 – FDT Disconnect service

IEC 1124/09

FMSAbort service request closes a single FMS connection.

7.1.2 FDT connection

FDT connection is using Connect service request:

- to establish an FDT connection intended for System Management services only. When FDT connections for SM services only are established, the OpenSessionRequest element

is not part of the ConnectRequest. Any request to establish an FMS connection on this type of FDT connection shall be rejected;

- to establish an FDT connection that represents a session as described with the specification FF-588 Field Device Access Agent. FMS connections can be established by using the FDT communication reference returned as a result of FDT connection establishment. Note that multicast SM services are not allowed in this connection.

If an HSE device needs both types of connections it needs to create System management and FDT Session Connections.

When a Frame Application requests the DTM to disconnect, all FMS connections and FDT connections (SM and Session) shall be terminated.

7.2 Abort

7.2.1 OnAbort Indication

At any moment during the lifetime of the (FDT or FMS) Connection an OnAbort event indication can be received by the DTM/BTM from the Communication Channel. There can be two reasons for OnAbort Event:

- FMS Connection Abort;
- FDT Connection Abort.

The Communication reference uniquely identifies the Connection to be aborted. If a FDT Connection Abort is indicated, all FMS connections of the corresponding FDT Connection shall be terminated. No Abort requests are issued to the Communication Channel. All pending requests shall be cancelled.

If an FMS Connection Abort is indicated, only the FMS Connection identified by that Communication Reference is terminated.

7.2.2 Abort request

At any moment during the lifetime of the (FDT or FMS) connections a DTM/BTM can issue an Abort request to the Communication Channel. There can be two types of Communication References in the Abort request:

- FMS Communication Reference;
- FDT Communication Reference.

The Communication reference uniquely identifies the Connection to be aborted.

If an FDT Connection Abort is requested, all related FMS connections in the Communication channel shall be terminated. No separate Abort requests are issued to the Communication Channel for the individual FMS Connections. All pending requests shall be cancelled.

If an FMS Communication Reference is used in the Abort, only the FMS Connection identified by that Communication Reference is terminated.

7.3 Relation of FMS requests and FMS responses

If a DeviceDTM (communication client) issues an FMS request as in Table 4, it shall expect a response as shown in the table.

Table 4 – Relation of FMS requests and FMS responses

Request	Response (s)
FmsInitiateRequest	FmsInitiateResponse FmsInitiateError
FmsAbortRequest	FmsStandardResponse
FmsReadRequest	FmsReadResponse FmsServiceError
FmsWriteRequest	FmsStandardResponse FmsServiceError
FmsStatusRequest	FmsStatusResponse FmsServiceError
FmsIdentifyRequest	FmsIdentifyResponse FmsServiceError
FmsDefineVariableListRequest	FmsDefineVariableListResponse FmsServiceError
FmsDeleteVariableListRequest	FmsStandardResponse FmsServiceError
FmsGetOdRequest	FmsGetOdResponse FmsServiceError
FmsGenericInitiateDownloadSequenceRequest	FmsStandardResponse FmsServiceError
FmsGenericDownloadSegmentRequest	FmsStandardResponse FmsServiceError
FmsGenericTerminateDownloadSequenceRequest	FmsGenericTerminateDownloadSequenceResponse FmsServiceError
fms:FmsInformationReport	fms:FmsInformationReport
fms:CreateProgramInvocation	FmsStandardResponse FmsServiceError
fms>DeleteProgramInvocation	FmsStandardResponse FmsServiceError
fms:Start	FmsStandardResponse FmsServiceError
fms:Stop	FmsStandardResponse FmsServiceError
fms:Resume	FmsStandardResponse FmsServiceError
fms:Reset	FmsStandardResponse FmsServiceError
fms:Kill	FmsStandardResponse FmsServiceError
H1SubscribeRequest	H1SubscribeResponse
H1UnSubscribeRequest	H1UnSubscribeResponse
HSESubscribeRequest	HSESubscribeResponse
HSEUnSubscribeRequest	HSEUnSubscribeResponse
Any Service	fdt:CommunicationError

A DTM or a BTM can receive a standard FDT CommunicationError response instead of FMS Transaction response to indicate a general communication error. Communication clients shall be prepared to handle such a response.

If the FMS connection in the Device DTM is shared between several BTMs, all connections will receive OnAbort event when the Device FMS connection is aborted.

7.4 Subscription mechanism

The FF subscription mechanism is based on the rules of the device initiated data transfer. For a DTM that supports Device Initiated Data Transfer there may be one or multiple subscribers expecting data at the same time.

7.4.1 General

To support the device initiated data transfer for IEC 61784 CPF 1 additional transaction request and transaction responses are defined. Two transaction requests are used to initiate the FMS subscription connection and four transaction responses are associated with the FMS subscriptions. These transactions are detailed below:

- transaction request for subscribing;
- transaction response for subscribing;
- transaction request for unsubscribing;
- transaction response for unsubscribing;
- transaction detailing the FMS Information Report;
- transaction response detailing the FMS EventNotification.

7.4.2 Transactions for subscribing H1

Device DTM/BTM can subscribe to data by calling a H1SubscribeRequest Transaction. It can subscribe for

- data collected in a trend object [H1TrendSubscriptionInfo];
- single device parameter [H1ParameterSubscriptionInfo];
- an alarm or event[H1EventSubscriptionInfo].

Additional information to identify the device is provided in the transaction request. This information may be used by the Communication Provider to speed up the process of subscribing.

The result of the subscription mechanism is provided in the H1SubscribeResponse transaction. The communication reference used is the communication reference of the FDT connection.

The InvokeID of the subscription request will be used to provide the association between the subscription and FmsInformationReport or FmsEventNotification transaction.

The InvokeID received during the subscription will be used for Unsubscribing.

The InvokeID uniquely identifies all transactions related to the created subscription and shall not be used for any other transactions during the lifetime of the subscription.

7.4.3 Transactions for subscribing HSE

The HSE Linking Device DTM can subscribe to data by calling an HSESubscribeRequest Transaction. DTMs can map the subscription from a H1 DTM to a HSE subscription.

The HSE subscription can be used to subscribe for data from

- an H1 device via a HSE Linking device;
- an HSE device.

It can subscribe for

- data in a trend object on an H1 device (HSETrendSubscriptionInfo);
- data in a trend object on an HSE device (LocalTrendSubscriptionInfo);
- single parameter from an H1 device (HSEParameterSubscriptionInfo);
- single parameter from an HSE device (LocalParameterSubscriptionInfo),
- an Event from an H1 device (HSEEventSubscriptionInfo);
- an Event from an HSE device (LocalEventSubscriptionInfo).

Additional information about the device is provided in the transaction request. This information may be used by the Communication Provider to speed up the process of subscribing.

The result of the subscription mechanism is provided in the HSESubscribeResponse transaction.

The communication reference used is the communication reference of the FDT connection.

The InvokeID of the subscription request will be used to provide the association between the subscription and FmsInformationReport or FmsEventNotification transaction. The same InvokeID will identify the subscription during the Unsubscribing.

The InvokeID uniquely identifies all transactions related to the created subscription and shall not be used for any other transactions during the lifetime of the subscription.

7.4.4 Transactions for subscribing BTM

A BTM can subscribe for data by calling a BtmSubscribeRequest Transaction. The parent DTM can map the subscription from a BTM to an H1 or HSE subscription.

BTM can subscribe for

- single parameter (BtmParameterSubscriptionInfo);
- data collected in the Device Trend object (BtmTrendSubscriptionInfo);
- an alarm or event (BtmEventSubscriptionInfo).

The Device DTM shall provide association and configuration of the trend object if a subscription for trend data is requested.

The result of the subscription mechanism is provided in the BtmSubscribeResponse transaction.

The communication reference used is the communication reference of the FDT connection.

The InvokeID of the subscription request will be used to provide the association between the subscription and FmsInformationReport or FmsEventNotification transaction. The same InvokeID will identify the subscription during the Unsubscribing. The InvokeID shall not be used for any other transactions during the lifetime of the subscription.

If more than one BTM is subscribed for the same object, the Device DTM shall propagate the information to all subscribed BTMs.

8 Protocol specific usage of general data types

8.1 Address

For all exposed parameters in DTMs and BTMs the address (defined in Part 2) shall be constructed according to the following model:

VFD:xx.INDEX:yyy[.SUBINDEX:zz],

where

- xx is the VFD tag or a number presenting the index of the VFD (one based);
- yy is a number presenting the parameter index from the beginning of the VFD;
- zz is a number showing the parameter sub-index.

The values of xx, yy and zz shall be integers (some can be 32 bit) and are presented as decimals digits. There should be no leading zeros for the numbers.

The sub-index portion of the address is optional and may be omitted or set to 0.

Providing this information with the device data set is sufficient for accessing the device/block parameters and there is no need for process channel definition for BTMs.

8.2 protocolID

See Clause 5.

8.3 applicationDomain

The applicationDomain attribute is fixed: FDT_FoundationFieldbus.

8.4 semanticId

The semanticId for FF related parameter follows the following rules:

- the semanticId shall be built based on the names defined in the FF specification;
- structured parameters shall be combined with a '.';
- spaces within the profile definition shall be exchanged with an underscore;
- only capital letters should be used.

8.4.1 Block specific definitions

For detailed description, please refer to 9.2.2 for block specific definitions.

EXAMPLES

OUT.STATUS

OUT.VALUE

OUT

8.4.2 Fieldbus management definitions

SemanticID for FF Fieldbus management parameter shall be created according to the following rule:

'FDT.UserDefinedBus' followed by the structure of the data type definitions of the management parameter. Each element shall be divided by a '.'.

EXAMPLE

FDT.UserDefinedBus.ListOfH1NmaVfds.H1NmaVfd.SMIB.SmAgent.T1.t1

defines the semantic ID for the T1 parameter.

8.4.3 Fieldbus specific definitions

For the parameters used for device/block identification (see Clause 12) a set of unique IDs are defined.

- For H1 devices a set of fixed semantic IDs are defined as follows:
 - IdAddress – Node address assigned to the device;
 - IdManufacturer – Device manufacturer ID;
 - IdTypeID – Device Type;
 - IdSoftwareRevision – Device and Device Descriptor's revision;
 - IdTag – Physical Device Tag;
 - IdSerialNumber – Physical device ID. This is unique identifier of the device instance;
- For HSE devices this IDs are defined as follows:
 - IdAddress – Address of the HSE device;
 - IdManufacturer – Device manufacturer ID;
 - IdTypeID – Device Type;
 - IdTag – Physical Device Tag;
 - IdSerialNumber – Physical device ID. This is unique identifier of the device instance;
 - numberOfLinks – link count;
 - linkId[0.. (numberOfLinks-1)] – link ID for the corresponding H1 link;
- For HSE device with Application VFD, the required IDs are:
 - IdManufacturer – Device manufacturer ID;
 - IdTypeID – Device Type;
 - IdSoftwareRevision – Device and Device Descriptor's revision;
- For FF blocks the following fixed semantic IDs are be defined:
 - IdTag – Block Tag as defined in BLOCK.BLOCK_TAG;
 - IdManufacturer – Block manufacturer ID is same as the device manufacturer ID;
 - IdTypeID – Block device type ia same as the device type;
 - IdSoftwareRevision – Block revision is same as the device revision;
 - Profile – Block Profile information as defined in BLOCK.PROFILE;
 - profileRevision – Block Profile revision information as defined in BLOCK.PROFILE_REVISION;
 - blockIndex – the starting index of the block.

9 Protocol specific data types

9.1 DTM

9.1.1 Topology scan definitions

As a result of the scanning, the DTM detects DTMs and BTM objects that are connected to the channel object. As a result of topology scan a DTM can expose information about the FF linking devices, about FF instruments and about the Blocks in the instruments.

9.1.2 Parameter access

9.1.2.1 General

DTM services to access instance data enable Frame Application access to device parameters. The DTM provides its current in-memory representation of its instance data.

DTM services to access device data enable Frame Application access to specific parameters in the device.

A FF device DTM can provide access to the following groups of parameters:

- link parameters;
- action parameters;
- alert parameters;
- trend parameters;
- domain objects;
- program invocation;
- views objects.

It is up to a DTM and depends on the device- and fieldbus-type to extend the list of device parameters.

9.1.2.2 Action structure

The action structure allows an instance of a block or object to be created or deleted in a resource supporting the action object. Through the FB_Action service, a write request may be issued to the action object. If the requested action is supported for the identified function, then the associated objects or blocks (and associated views object) will be created or deleted in the object dictionary.

This data structure consists of an action identifier and associated information concerning the type, function, and occurrence for their related object or block.

The following Table 5 provides the Action structured parameter Mnemonic (based on FOUNDATION™ Fieldbus specifications FF-890) and Action structured parameters mapping to the FDT data types. The mnemonic for the structure and each parameter should be used as name attribute value for parameter definition:

Table 5 – Action object definitions (refer to FF-890):

Structure ²	Parameter mnemonic of member	Fieldbus foundation datatype/structure
ACTION_STRUCTURE	ACTION_STRUCTURE	DS-86
	ACTION_INDEX	-
	ACTION	Unsigned8
	FUNCTION	Unsigned32
	OCCURRENCE	Unsigned16

9.1.2.3 Link structure

Link structure objects provide a mapping between resources and the information exchanged via a communication network. Process data and events to be exchanged between function blocks within a resource or between resources may be defined through link objects. In addition, the communication exchange for support of trends and alerts may be defined using link objects.

Parameters with a class of output and input are connected by function block links.

Links between field devices and interface devices may be defined for access to trend objects, and alert reporting. Link objects identify block parameters, trend objects, events, and the communication characteristics that define how they exchange data. Through link object definition, the alerts and trends may be routed to different devices for processing.

The following Table 6 provides the Link structured parameter mnemonic (based on FOUNDATION™ Fieldbus specifications FF-890) and Link structured parameters mapping to the FDT data types. The mnemonic for the structure and each parameter should be used as the name attribute value for parameter definition:

Table 6 – Link object definitions

Structure	Parameter mnemonic of member	Fieldbus foundation data type/structure
LINK_STRUCTURE	LINK_STRUCTURE	DS-81
LINK_STRUCTURE (HSE)	LINK_STRUCTURE	DS-165
	LINK_TYPE	-
	LOCAL_INDEX	Unsigned32
	VCR_LINK_ID	Unsigned16
	VCR_NUMBER	Unsigned16
	REMOTE_INDEX	Unsigned32
	SERVICE_OPERATION	Unsigned8
	STALE_COUNT_LIMIT	Unsigned8
	H1_SUBSCRIBER_INDEX	Unsigned16

² The names for members are defined by the scheme: name of the structured variable „.“ Name of the Member (see definition of StructuredVariable within FDTDataTypes definition – members of structures are referenced within the definition.). For example TARGET of DS-69 would expand to MODE_BLK.TARGET.

9.1.2.4 Alert structure

Alert structured objects are used to communicate notification messages when alarms or events are detected. An event is an instantaneous occurrence that is significant to scheduling block execution and to the operational view of an application. An alarm is the detection of a block leaving a particular state and when it returns back to that state. The time at which the alert state was detected is included as a time stamp in the alert message. Also, the priority is included to indicate whether this is an advisory or critical alert.

Resources each have an alert notifier responsible for reporting their alert occurrences. Alerts may also be acknowledged. Alert objects are used to report alarm and event occurrences as event notification messages. Acknowledgment indicates that the alert has been processed by an interface device to satisfy operational interface requirements.

Based on the type of alarm and event information which may be reported by blocks contained in a resource, up to three classes of alerts may be defined in the resource:

- analog alert - alert used to report alarms or events whose associated value is the floating point;
- discrete alert - alert used to report alarms or events whose associated value is discrete;
- update alert - alert used to report a change in the static data of the block.

Table 7 provides the alert structured parameter mnemonic (based on specification FF-890) and alert structured parameters mapping to the FDT data types. The mnemonic for the structure and each parameter should be used as name attribute value for parameter definition:

Table 7 – Alert object definitions

Structure mnemonic	PARAMETER MNEMONIC OF MEMBER	Fieldbus foundation datatype/structure
ALERT_ANALOG	ALERT_ANALOG	DS-75
ALERT_ANALOG (HSE)	ALERT_ANALOG	DS-167
	BLOCK_INDEX	Unsigned16/Unsigned32
	ALERT_KEY	Unsigned8
	STANDARD_TYPE	Unsigned8
	MFR_TYPE	Unsigned8
	MESSAGE_TYPE	Unsigned8
	PRIORITY	Unsigned8
	TIME_STAMP	TimeValue
	SUBCODE	Unsigned16
	VALUE	Float
	RELATIVE_INDEX	Unsigned16/Unsigned32
	UNITINDEX	Unsigned16
ALERT_DISCRETE	ALERT_DISCRETE	DS-76
ALERT_DISCRETE(HSE)	ALERT_DISCRETE	DS-168
	BLOCK_INDEX	Unsigned16/Unsigned32
	ALERT_KEY	Unsigned8
	STANDARD_TYPE	Unsigned8
	MFR_TYPE	Unsigned8
	MESSAGE_TYPE	Unsigned8
	PRIORITY	Unsigned8

Structure mnemonic	PARAMETER MNEMONIC OF MEMBER	Fieldbus foundation datatype/structure
	TIME_STAMP	TimeValue
	SUBCODE	Unsigned16
	VALUE	Unsigned8
	RELATIVE_INDEX	Unsigned16/Unsigned32
	UNITINDEX	Unsigned16
ALERT_UPDATE	ALERT_UPDATE	DS-77
ALERT_UPDATE(HSE)	ALERT_UPDATE	DS-169
	BLOCK_INDEX	Unsigned16/Unsigned32
	ALERT_KEY	Unsigned8
	STANDARD_TYPE	Unsigned8
	MFR_TYPE	Unsigned8
	MESSAGE_TYPE	Unsigned8
	PRIORITY	Unsigned8
	TIME_STAMP	TimeValue
	STATIC_REVISION	Unsigned16
	RELATIVE_INDEX	Unsigned16/Unsigned32

9.1.2.5 Trend structure

Trend structure objects support management and control of function blocks by providing visibility into history information for reviewing their behavior. Based on the type of information collected, three classes of trend objects may be defined:

- trend float - trend object used to collect the values and status of floating point input and output parameters;
- trend discrete - trend object used to collect the values and status of discrete input and output parameters;
- trend bit string - trend object used to collect the values and status of bit string input and output parameters.

Trend objects provide for short term history data to be collected and stored within a resource. Such collection may be efficiently accessed by interface or temporary devices which gather this information to provide long term history data. Through link object configuration, some or all trend objects in a resource may be automatically reported to an interface device each time a new set of data is collected. Also, trend objects may be read at any time by an interface device using FMS read services.

Table 8 provides the Trend structured parameter mnemonic (based on specification FF-890) and Trend structured parameters mapping to the FDT data types. The mnemonic for the structure and each parameter should be used as name attribute value for parameter definition:

Table 8 – Trend object definitions

Structure mnemonic	Parameter mnemonic of member	Fieldbus foundation datatype/structure
TREND_FLOAT		DS-78
TREND_FLOAT (HSE)		DS-170
	BLOCK_INDEX	Unsigned16/Unsigned32

Structure mnemonic	Parameter mnemonic of member	Fieldbus foundation datatype/structure
	PARAMETER_RELATIVE_INDEX	Unsigned16
	SAMPLE_TYPE	Unsigned8
	SAMPLE_INTERVAL	Unsigned32
	LAST_UPDATE_TIME	TimeValue
	STATUS[0]	Unsigned8
	SAMPLES[0]	Float
		
	STATUS[15]	Unsigned8
	SAMPLES[15]	Float
TREND_DISCRETE		DS-79
TREND_DISCRETE (HSE)		DS-171
	BLOCK_INDEX	Unsigned16/Unsigned32
	PARAMETER_RELATIVE_INDEX	Unsigned16
	SAMPLE_TYPE	Unsigned8
	SAMPLE_INTERVAL	Unsigned32
	LAST_UPDATE_TIME	TimeValue
	STATUS[0]	Unsigned8
	SAMPLES[0]	Unsigned8
		
	STATUS[15]	Unsigned8
	SAMPLES[15]	Unsigned8
TREND_BIT_STRING		DS-80
TREND_BIT_STRING (HSE)		DS-172
	BLOCK_INDEX	Unsigned16/Unsigned32
	PARAMETER_RELATIVE_INDEX	Unsigned16
	SAMPLE_TYPE	Unsigned8
	SAMPLE_INTERVAL	Unsigned32
	LAST_UPDATE_TIME	TimeValue
	STATUS[0]	Unsigned8
	SAMPLES[0]	BitString
		
	STATUS[15]	Unsigned8
	SAMPLES[15]	BitString

9.1.2.6 View structure

View objects support management and control of function blocks by providing visibility into their configuration and operation. The first parameter of each view object is the static revision code. Accordingly, the remaining parameters included in the view block definition will follow the static revision code sequentially by order of their associated index within the function block. View objects provided for the grouping of dynamic and static data associated with

operation, diagnostic, and configuration. This grouping provides for efficient data transfer and processing.

Table 9 provides the view structured parameter mnemonic (based on specification FF-890) and the view structured parameters mapping to the FDT data types. The mnemonic for the structure and each parameter should be used as name attribute value for parameter definition:

Table 9 – View definition

Structure mnemonic	Parameter mnemonic of member	Fieldbus foundation datatype/structure
VIEW		
	VIEW_INDEX	Unsigned16/Unsigned32
	ELEMENT_1 (1. . . NUMBER OF ELEMENTS)	Unsigned16/Unsigned32

9.1.2.7 Domain object

A domain is a part of memory. It may contain programs or data. The FF specification defines that the domain data shall be of data type “octet string”. The encoding of information in the domain is not defined by FF specification. The maximum number of octets of a domain shall be defined in the domain object. Note that the domain state represents the DTM estimation of the real domain state in the device. Full definition of the domain object member can be found in FF-870 Fieldbus Message Specification.

Table 10 provides the domain structured parameter mnemonic (based on specification FF-870) and domain structured parameters mapping to the FDT data types. The mnemonic for the structure and each parameter should be used as name attribute value for parameter definition:

Table 10 – Domain object definitions

Structure mnemonic	Parameter mnemonic of member	Fieldbus foundation datatype/structure
DOMAIN_DESCRIPTION		-
	INDEX	Unsigned32
	MAX_OCTETS	Unsigned32
	DOMAIN_STATE	Unsigned8
	UPLOAD_STATE	Unsigned8
	COUNTER	Integer8
	DOMAIN_DATA	OctetString

9.1.2.8 Program invocation object

The program invocation model provides services to link domains to a program, to start this program, to stop and to delete it. More than one program invocation may be created in a device.

Domain list associated with the PI object is provided by providing the OD index for each domain in the list. The number of domains can be calculated by counting the elements with name attribute value equal to “domain” in the PI object.

Table 11 provides the PI structured parameter mnemonic (based on specification FF-890) and PI structured parameters mapping to the FDT data types. The mnemonic for the structure and each parameter should be used as name attribute value for parameter definition:

Table 11 – Program invocation object definitions

Structure mnemonic	Parameter mnemonic of member	Fieldbus foundation datatype/structure
PROGRAM_INVOCATION		-
	PI_INDEX	Unsigned32
	REUSABLE	Integer8
	DELETABLE	Integer8
	PI_STATE	Integer8
	DOMAIN	Unsigned32

9.1.3 FF device data types

The following Table 12 specifies FF device data types and it provides description information about the FF devices.

Namespace: fdtffdevice

Table 12 – Structured FF device data types

Data type	Definition			Description
	Elementary data types	Usage	Multiplicity	
BlockList	STRUCT			Provides the list of blocks in the device
	ffblock:BlockInformation	O	[0..*]	
HSEDeviceInformation	STRUCT			HSEDeviceInformation describes High Speed Ethernet compatible devices including device version, redundancy, etc.
	ffhse:stateSMK	M	[1..1]	
	ffhse:repeatTimeHSE	O	[0..1]	
	ffhse:duplicateDetectionState	O	[0..1]	
	fftypes:ip	O	[0..1]	
	fftypes:IP	M	[1..1]	
	ffhse:DeviceInformation	M	[1..1]	
	ffhse:RedundancyInformation	O	[0..1]	
	ffhse:VersionInformation	O	[0..1]	
FoundationFieldbusH1 Device	STRUCT			FoundationFieldbusH1Device includes information describing H1 compatible devices
	fftypes:deviceType	M	[1..1]	
	fdt:tag	M	[1..1]	
	fftypes:devID	M	[1..1]	
	fftypes:VfdIdentification	M	[1..*]	
	fftypes:DataLinkAddress	M	[1..1]	
FoundationFieldbus-HSEDevice	STRUCT			Describes HSE compatible devices including the device address, VFD information, etc.
	fftypes:deviceType	M	[1..1]	
	fdt:tag	M	[1..1]	
	fftypes:devID	M	[1..1]	
	fftypes:VfdIdentification	M	[1..*]	
	HSEDeviceInformation	M	[1..1]	
	fftypes:DataLinkAddress	M	[1..1]	

9.2 BTM

9.2.1 General

In the case of a BTM, the following definitions are used instead of the DTM-specific definitions defined within the FDT specification.

9.2.2 Parameter access - FF specific definitions

Block parameters can be accessed through the following services:

- services to access instance data;

- services to access device data.

BTMs should use standard name attributes for standard block parameters. The following table provides the list of the standard block parameter mnemonic (based on specification FF-890, FF-891, FF-892, FF-893, and FF-894) and their mapping to the FDT data types.

The application domain for FF block parameters shall be “FDT_FFBlock”

The parameter mnemonic should be used as the name attribute value and as a semantic ID for the parameter definition as defined in Table 13:

Table 13 – Parameter mnemonic

Parameter mnemonic	Fieldbus foundation datatype/structure
ACCEPT_ALM	DS-72
ACCEPT_D	Unsigned8
ACCEPT_PRI	Unsigned8
ACK_OPTION	Bit String
ALARM_HYS	Float
ALARM_SUM	DS-74
ALERT_KEY	Unsigned8
ALGORITHM_SEL	Unsigned32
ALM_RATE_DN	Float
ALM_RATE_UP	Float
ARITH_TYPE	Unsigned8
AUTO_CYCLE	Unsigned8
BAL_TIME	Float
BIAS	Float
BIAS_IN_1	Float
BIAS_IN_2	Float
BIAS_IN_3	Float
BKCAL_HYS	Float
BKCAL_IN	DS-65
BKCAL_IN_1	DS-65
BKCAL_IN_2	DS-65
BKCAL_IN_D	DS-66
BKCAL_OUT	DS-65
BKCAL_OUT_D	DS-66
BKCAL_SEL_1	DS-65
BKCAL_SEL_2	DS-65
BKCAL_SEL_3	DS-65
BLOCK	DS-64
BLOCK_ALM	DS-72
BLOCK_ERR	Bit String
BYPASS	Unsigned8
CAS_IN	DS-65
CAS_IN_D	DS-66

Parameter mnemonic	Fieldbus foundation datatype/structure
CFM_ACT1_TIME	Float
CFM_ACT2_TIME	Float
CFM_PASS_TIME	Float
CHANNEL	Unsigned16
CHARACTERISTICS ³	DS-64
CLOCK_PER	Float
CLR_FSTATE	Unsigned8
COMB_TYPE	Unsigned8
COMP_HI_LIM	Float
COMP_LO_LIM	Float
CONFIRM_TIME	Unsigned32
CONTENTS_REV	Unsigned32
CONTROL_OPTS	Bit String
CRACK_TIME	Float
CRACK_TIMER	Float
CURVE_X	Float
CURVE_Y	Float
CYCLE_SEL	Bit String
CYCLE_TYPE	Bit String
DC_STATE	Unsigned8
DD_RESOURCE	Visible String
DD_REV	Unsigned8
DEAD_TIME	DS-65
DELAY_TIME	Float
DELAY_TIMER	Float
DEV_REV	Unsigned8
DEV_TYPE	Unsigned16
DEVICE_OPTS	Bit string
DISABLE_1	DS-66
DISABLE_2	DS-66
DISABLE_3	DS-66
DISABLE_4	DS-66
DISC_ALM	DS-72
DISC_LIM	Unsigned8
DISC_PRI	Unsigned8
DURATION	DURATION_TYPE (10 Floats) ⁴
DV_HI_ALM	DS-71
DV_HI_LIM	Float

³ Represents the block structure of the block as defined in 5.14.1 of the specification FF-890.

⁴ Defined as an array of 10 floats in the Fieldbus Foundation specification FF-892 and will be mapped to a StructuredVariable containing the elements of the array.

Parameter mnemonic	Fieldbus foundation datatype/structure
DV_HI_PRI	Unsigned8
DV_LO_ALM	DS-71
DV_LO_LIM	Float
DV_LO_PRI	Unsigned8
EXPAND_DN	Float
EXPAND_UP	Float
FAIL	Unsigned16
FAIL_ALM	DS-72
FAIL_PRI	Unsigned8
FAULT_STATE	Unsigned8
FCF_LOCATOR	LOCATOR (Unsigned32 Array) 5
FDD_LOCATOR	LOCATOR (Unsigned32 Array)
FEATURE_SEL	Bit String
FEATURES	Bit String
FF_GAIN	Float
FF_SCALE	DS-68
FF_VAL	DS-65
FIELD_VAL	DS-65
FIELD_VAL_D	DS-66
FILE_LOCATOR	LOCATOR (Unsigned32 Array)
FILE_REV	Unsigned32
FOLLOW	DS-66
FREE_SPACE	Float
FREE_TIME	Float
FSTATE_STATUS	Unsigned8
FSTATE_TIME	Float
FSTATE_VAL	Float
FSTATE_VAL_D	Unsigned8
FSTATE_VAL_D1	Unsigned8
FSTATE_VAL_D2	Unsigned8
FSTATE_VAL_D3	Unsigned8
FSTATE_VAL_D4	Unsigned8
FSTATE_VAL_D5	Unsigned8
FSTATE_VAL_D6	Unsigned8
FSTATE_VAL_D7	Unsigned8
FSTATE_VAL_D8	Unsigned8
FSTATE_VAL1	Float
FSTATE_VAL2	Float
FSTATE_VAL3	Float

⁵ Defined as an array of 3 Unsigned32 values in the Fieldbus Foundation specification FF-894 and will be mapped to a StructuredVariable containing the elements of the array.

Parameter mnemonic	Fieldbus foundation datatype/structure
FSTATE_VAL4	Float
FSTATE_VAL5	Float
FSTATE_VAL6	Float
FSTATE_VAL7	Float
FSTATE_VAL8	Float
GAIN	Float
GAIN_IN_1	Float
GAIN_IN_2	Float
GAIN_IN_3	Float
GOOD_LIM	Float
GRANT_DENY	DS-70
HARD_TYPES	Bit String
HI_ALM	DS-71
HI_BIAS	Float
HI_GAIN	Float
HI_HI_ALM	DS-71
HI_HI_BIAS	Float
HI_HI_LIM	Float
HI_HI_LIMX	Float
HI_HI_PRI	Unsigned8
HI_LIM	Float
HI_LIMX	Float
HI_PRI	Unsigned8
HYSTVAL	Float
IGNORE	Bit String
IGNORE_ALM	DS-72
IGNORE_PRI	Unsigned8
IGNORE_TIME	Float
IN	DS-65
IN_1	DS-65
IN_2	DS-65
IN_3	DS-65
IN_4	DS-65
IN_5	DS-65
IN_6	DS-65
IN_7	DS-65
IN_8	DS-65
IN_ARRAY	Float
IN_D	DS-66
IN_D1	DS-66
IN_D2	DS-66
IN_D3	DS-66
IN_D4	DS-66

Parameter mnemonic	Fieldbus foundation datatype/structure
IN_LO	DS-65
INPUT_OPTS	Bit string
INTEG_OPTS	Bit string
INTEG_TYPE	Unsigned8
INTERLOCK_D	DS-66
INVERT_OPTS	Bit string
IO_OPTS	Bit string
ITK_VER	Unsigned16
L_TYPE	Unsigned8
LAG_TIME	Float
LEAD_TIME	Float
LIM_NOTIFY	Unsigned8
LO_ALM	DS-71
LO_BIAS	Float
LO_GAIN	Float
LO_LIM	Float
LO_LIMX	Float
LO_LO_ALM	DS-71
LO_LO_BIAS	Float
LO_LO_LIM	Float
LO_LO_LIMX	Float
LO_LO_PRI	Unsigned8
LO_PRI	Unsigned8
LOCKVAL	Unsigned8
LOW_CUT	Float
MANUFAC_ID	Unsigned32
MAX_NOTIFY	Unsigned8
MEMORY_SIZE	Unsigned16
MIN_CYCLE_T	Unsigned32
MIN_GOOD	Unsigned8
MO_OPTS	Bit string
MODE_BLK	DS-69
N_RESET	Float
N_START	Unsigned16
NV_CYCLE_T	Unsigned32
OP_CMD_INT	Unsigned8
OP_CMD_SPG	Unsigned8
OP_SELECT	DS-66
OUT	DS-65
OUT_1	DS-65
OUT_1_RANGE	DS-68
OUT_2	DS-65
OUT_2_RANGE	DS-68

Parameter mnemonic	Fieldbus foundation datatype/structure
OUT_3	DS-65
OUT_4	DS-65
OUT_5	DS-65
OUT_6	DS-65
OUT_7	DS-65
OUT_8	DS-65
OUT_ALM	DS-66
OUT_ALM_SUM	Unsigned8
OUT_ARRAY	Float
OUT_D	DS-66
OUT_D5	DS-66
OUT_D6	DS-66
OUT_D7	DS-66
OUT_D8	DS-66
OUT_EXP	DS-65
OUT_HI_LIM	Float
OUT_LO_LIM	Float
OUT_PTRIP	DS-66
OUT_RANGE	DS-68
OUT_REM	DS-65
OUT_SCALE	DS-68
OUT_STATE	Unsigned16
OUT_TRIP	DS-66
OUTAGE_LIM	Float
PAUSE	DS-66
PAUSE_CAUSE	Unsigned8
PCT_INCL	Float
PERMISSIVE_D	DS-66
PI_POINTER	Unsigned32
PRE_OUT	DS-65
PRE_OUT_ALM	DS-66
PRE_OUT_D	DS-66
PRE_TRIP	Float
PSP	DS-65
PULSE_VAL1	Float
PULSE_VAL2	Float
PV	DS-65
PV_D	DS-66
PV_FTIME	Float
PV_SCALE	DS-68
PV_STATE	Unsigned16
QUIES_OPT	Unsigned8
RA_FTIME	Float

Parameter mnemonic	Fieldbus foundation datatype/structure
RANGE_HI	Float
RANGE_LO	Float
RATE	Float
RCAS_IN	DS-65
RCAS_IN_D	DS-66
RCAS_OUT	DS-65
RCAS_OUT_D	DS-66
READBACK	DS-65
READBACK_D	DS-66
RESET	Float
RESET_CONFIRM	DS-66
RESET_D	Unsigned8
RESET_IN	DS-66
RESTART	Unsigned8
RESTART_TIME	Float
REV_FLOW1	DS-66
REV_FLOW2	DS-66
ROUT_IN	DS-65
ROUT_OUT	DS-65
RS_STATE	Unsigned8
RTOTAL	Float
SEL_1	DS-65
SEL_2	DS-65
SEL_3	DS-65
SEL_TYPE	Unsigned8
SELECT_TYPE	Unsigned8
SELECTED	DS-66
SET_FSTATE	Unsigned8
SHED_OPT	Unsigned8
SHED_RCAS	Unsigned32
SHED_ROUT	Unsigned32
SHUTDOWN_D	DS-66
SIMULATE	DS-82
SIMULATE_D	DS-83
SP	DS-65
SP_D	DS-66
SP_HI_LIM	Float
SP_LO_LIM	Float
SP_RATE_DN	Float
SP_RATE_UP	Float
SPG_STATE	Unsigned8
SRTOTAL	Float
SSP	Float

Parameter mnemonic	Fieldbus foundation datatype/structure
ST_REV	Unsigned16
START	DS-66
START_TYPE	Unsigned8
START_VAL	START_VAL_TYPE (11 Floats) 6
STATUS_OPTS	Bit string
STEP_POSN	DS-66
STOTAL	Float
STRATEGY	Unsigned16
SWAP_2	Unsigned8
TAG_DESC	Octet string
TEST_RW	DS-85
TIME_POSN	DS-65
TIME_POSN_T	DS-65
TIME_UNIT1	Unsigned8
TIME_UNIT2	Unsigned8
TIME_UNITS	Unsigned8
TIMER_SP	Float
TIMER_TYPE	Unsigned8
TOTAL_SP	Float
TRAVEL_TIMER	Float
TRIP_TIME	Float
TRK_IN_D	DS-66
TRK_SCALE	DS-68
TRK_VAL	DS-65
UNCERT_LIM	Float
UNIT_CONV	Float
UPDATE_EVT	DS-73
WRITE_ALM	DS-72
WRITE_LOCK	Unsigned8
WRITE_PRI	Unsigned8
X_RANGE	DS-68
XD_SCALE	DS-68
XD_STATE	Unsigned16
Y_RANGE	DS-68

Table 14 describes the structure definitions (according to the FF specification) and their mapping to FDT data type definitions:

⁶ Defined as an array of 11 floats within Fieldbus Foundation specification FF-892 and will be mapped to a StructuredVariable containing the elements of the array.

Table 14 – Mnemonic of structured data types

Structure	Parameter mnemonic of member	Fieldbus foundation datatype/structure
DS-64		DS-64
DS-64		DS-166
	BLOCK_TAG	Visible String
	DD_MEMBER	Unsigned32
	DD_ITEM	Unsigned32
	DD_REVIS	Unsigned16
	PROFILE	Unsigned16
	PROFILE_REVISION	Unsigned16
	EXECUTION_TIME	Unsigned32
	EXECUTION_PERIOD	Unsigned32
	NUM_OF_PARAMS	Unsigned16
	NEXT_FB_TO_EXECUTE	Unsigned16/ Unsigned32
	VIEWS_INDEX	Unsigned16/ Unsigned32
	NUMBER_VIEW_3	Unsigned8
	NUMBER_VIEW_4	Unsigned8
DS-65		DS-65
	STATUS	Unsigned8
	VALUE	Float
DS-66		DS-66
	STATUS	Unsigned8
	VALUE	Unsigned8
DS-67		DS-67
	STATUS	Unsigned8
	VALUE	Bit String
DS-68		DS-68
	EU_100	Float
	EU_0	Float
	UNITS_INDEX	Unsigned16
	DECIMAL	Integer8
DS-69		DS-69
	TARGET	Bit String
	ACTUAL	Bit String
	PERMITTED	Bit String
	NORMAL	Bit String
DS-70		DS-70
	GRANT	Bit String
	DENY	Bit String
DS-71		DS-71
	UNACKNOWLEDGED	Unsigned8
	ALARM_STATE	Unsigned8

Structure	Parameter mnemonic of member	Fieldbus foundation datatype/structure
	TIME_STAMP	Time value
	SUB_CODE	Unsigned16
	VALUE	Float
DS-72		DS-72
	UNACKNOWLEDGED	Unsigned8
	ALARM_STATE	Unsigned8
	TIME_STAMP	Time value
	SUB_CODE	Unsigned16
	VALUE	Unsigned8
DS-73		DS-73
	UNACKNOWLEDGED	Unsigned8
	UPDATE_STATE	Unsigned8
	TIME_STAMP	Time value
	STATIC_REVISION	Unsigned16
	RELATIVE_INDEX	Unsigned16
DS-74		DS-74
	CURRENT	Bit String
	UNACKNOWLEDGED	Bit String
	UNREPORTED	Bit String
	DISABLED	Bit String
DS-82		DS-82
	SIMULATE_STATUS	Unsigned8
	SIMULATE_VALUE	Float
	TRANSDUCER_STATUS	Unsigned8
	TRANSDUCER_VALUE	Float
	ENABLE_DISABLE	Unsigned8
DS-83		
	SIMULATE_STATUS	Unsigned8
	SIMULATE_VALUE	Float
	TRANSDUCER_STATUS	Unsigned8
	TRANSDUCER_VALUE	Float
	ENABLE_DISABLE	Unsigned8
DS-84		
	SIMULATE_STATUS	Unsigned8
	SIMULATE_VALUE	BitString
	TRANSDUCER_STATUS	Unsigned8
	TRANSDUCER_VALUE	BitString
	ENABLE_DISABLE	Unsigned8
DS-85		DS-85
	VALUE_1	Boolean
	VALUE_2	Integer8
	VALUE_3	Integer16
	VALUE_4	Integer32



Structure	Parameter mnemonic of member	Fieldbus foundation datatype/structure
	VALUE_5	Unsigned8
	VALUE_6	Unsigned16
	VALUE_7	Unsigned32
	VALUE_8	Float
	VALUE_9	Visible String
	VALUE_10	Octet String
	VALUE_11	Date
	VALUE_12	Time of Day
	VALUE_13	Time Difference
	VALUE_14	Bit String
	VALUE_15	Time Value
DS-141		
	Reserved	OctetString
	Status	Unsigned8
	Value	Boolean
DS-142		
	Reserved	OctetString
	Status	Unsigned8
	Value	Integer8
DS-143		
	Reserved	OctetString
	Status	Unsigned8
	Value	Integer16
DS-144		
	Reserved	OctetString
	Status	Unsigned8
	Value	Integer32
DS-145		
	Reserved	OctetString
	Status	Unsigned8
	Value	Unsigned16
DS-146		
	Reserved	OctetString
	Status	Unsigned8
	Value	Unsigned 32
DS-147		
DS-148		
DS-149		
DS-150		
DS-151		
DS-152		
	Reserved	OctetString
	Status	Unsigned8

Structure	Parameter mnemonic of member	Fieldbus foundation datatype/structure
	Value	VisibleString
DS-153		
	Status	Unsigned8
	Value	Date
DS-154		
	Reserved	OctetString
	Status	Unsigned8
	Value	TimeOfDay
DS-155		
	Reserved	OctetString
	Status	Unsigned8
	Value	TimeDifference
DS-156		
	Reserved	OctetString
	Status	Unsigned8
	Value	Time
DS – 157		
DS – 158		
	Reserved	OctetString
	Status	Unsigned8
	Value	BitString(32)/ BitString(64)
DS – 159		
	Reserved	OctetString
	Status	Unsigned8
	Value_1	Unsigned8
		
	Value_8	Unsigned8
DS – 160		
	Reserved	OctetString
	Status	Unsigned8
	Value_1	Unsigned8
		
	Value_16	Unsigned8
DS – 161		
	Reserved	OctetString
	Status	Unsigned8
	Value_1	Unsigned8
		
	Value_32	Unsigned8
DS – 162		
	Reserved	OctetString
	Status	Unsigned8

Structure	Parameter mnemonic of member	Fieldbus foundation datatype/structure
	Value_1	Float
		
	Value_8	Float
DS – 163		
	Reserved	OctetString
	Status	Unsigned8
	Value_1	Float
		
	Value_16	Float
DS – 164		
	Reserved	OctetString
	Status	Unsigned8
	Value_1	Float
		
	Value_24	Float
DURATION_TYPE	DURATION_1 -	Float
	DURATION_10	Float
START_VAL_TYPE	START_VAL_1	Float
	START_VAL_11	Float
LOCATOR	KEY	Unsigned32
	LENGTH	Unsigned32
	LOCATION_INDEX	Unsigned32

Table 15 describes the simple common data types (according to the FF specification) and their mapping to FDT data type definitions:

Table 15 – Simple common data types

FF data type definition	FDT datatype	Note
Boolean	Int	Definition: == 0 is False != 0 is True
Integer 8	Int	Note that only the first 8 bits (LSBs) will be used
Integer 16	Int	Note that only the first 16 bits (LSBs) will be used
Integer 32	Int	
Unsigned 8	Unsigned	Note that only the first 8 bits (LSBs) will be used
Unsigned 16	Unsigned	Note that only the first 16 bits (LSBs) will be used
Unsigned 32	Unsigned	
Floating Point	Float	
Visible String	ascii	Please see FF-870, 9.3.1.5 for a detailed definition and byte sequence
Octet String	hexString	Please see FF-870, 9.3.1.6 for a detailed definition and byte sequence of Octet string
Date	dateAndTime	The standard dateTime data type will be used to present the value

FF data type definition	FDT datatype	Note
Time of Day	dateAndTime	The standard dateTime data type will be used to present the value
Time Difference	hexString	Please see FF-870, 9.3.1.9 for a detailed definition and byte sequence of Time Difference
Bit String	bitString	NOTE Bit enumerations will be used if possible
Time Value	hexString	Please see FF-870, 9.3.1.11 for a detailed definition and byte sequence of Time Value

10 Network management data types

10.1 General

The management definitions will be used as an inline definition within an instance of the DTMPParameterDefinition at the element 'BusInformation\UserDefinedBus' as an inline definition.

10.2 H1 network management definitions

The H1 Fieldbus Management data types are defined in Table 16 and used with namespace: ffH1Mngmnt:)

Table 16 – H1 Fieldbus Management data types

Name	Description
ListOfH1NmaVfds	List of Network Management VFDs
H1NmaVfd	The Network Management Agent is an application process modeled by the Management VFD. The Management VFD is shared between the NMA and the System Management Kernel, and contains the NMIB and the SMIB
NMIB	Network Management Information Base
SMIB	System Management Information Base

10.3 HSE network management data types

The simple HSE Fieldbus Management definitions are listed in Table 17 and used with namespace: ffhsemngmnt

Table 17 – Simple HSE Fieldbus Management Definitions

Data type	Definition	Description
actualNumberOfVFDs	UDINT	Actual Number of VFDs. FF-803 FS 1.19 SM Section 6.2.1
bufferSize	UDINT	This attribute is defined in the Field Device Access (FDA) Agent Specification [FF-588] as the Buffer Size. DEFAULT_BUFFER_SIZE = 1460 FF-803 FS 1.19 NM Section 5.1.2.
capableTimeSyncClass	UDINT	This variable specifies the time sync class that the SNTP time client in the HSE Presence is capable of supporting. The enumeration of time sync classes is as shown below, defined in IEC 61158-4-1, 7.6, subfield TTT. FF-803 FS 1.19 SM Section 6.2.2.1.

Data type	Definition	Description
daylightTimeDifference	DINT	This variable contains the number of signed ticks to add to Current Time to obtain Daylight time-stamp time. It is used instead of Standard Time Difference when Current Time is inside of the interval defined by Start Daylight Time and End Daylight Time. FF-803 FS 1.19 SM Section 6.2.2.1
discardedForForwardingDelayExceeded	UDINT	
discardedForLackOfBuffers	UDINT	This element counts the number of valid frames that should have been submitted for transmission on one or more outbound interfaces, but were discarded because of a lack of available forwarding buffers. FF-803 FS 1.19 NM Section 5.9.6
dstLinkAddress	UDINT	This attribute contains the 16-bit destination address (HL) of the message that should be forwarded by the H1 Bridge. If the value is 0, then this entry is not configured. FF-803 FS 1.19 NM Section 5.9.5
endpointType	USINT	This attribute is defined in the Field Device Access (FDA) Agent Specification [FF-588] as the Endpoint Type, and indicates the type of the endpoint. FF-803 FS 1.19 NM Section 5.1.2
fasDIIMaxConfirmDelayOnConnect	UINT	This element is defined in the Network Management Specification [FF-801] as FasDIIMaxConfirmDelayOnConnect. FF-803 FS 1.19 NM Section 5.9.4
fasDIIMaxDlsduSize	UINT	This element is defined in the Network Management Specification [FF-801] as FasDIIMaxDlsduSize. FF-803 FS 1.19 NM Section 5.9.4
fasDIISDAP	USINT	This element is defined in the Network Management Specification [FF-801] as FasDIISDAP. . FF-803 FS 1.19 NM Section 5.9.4
fbScheduleDescriptor		This attribute is the OD index of the FB Schedule Descriptor Record. The value of this attribute is FB Schedule OD Index + 2 + offset, where offset represents the zero-based position in the list. That is, the offset of the first record in the list is zero. The offset is no greater than the number of schedules specified in the FB Schedule List Characteristics record minus 1. FF-803 FS 1.19 SM Section 6.2.5.2
fdaAddress	ARRAY OF USINT	This attribute is defined in the Field Device Access (FDA) Agent Specification [FF-588] as the FDA Address. FF-803 FS 1.19 NM Section 5.1.3
fdaGuardBand	UDINT	This attribute is defined in the Field Device Access (FDA) Agent Specification [FF-588] as the Guard Band. FF-803 FS 1.19 NM Section 5.2.1
filteringDatabaseListHeaderOdIndex	UDINT	This is an instance of the ListHeader class representing the Filtering Database header. FF-803 FS 1.19 NM Section 5.9.1
forwardedInbound	UDINT	This element counts the number of valid frames received by the interface that were submitted for transmission on one or more outbound interfaces. FF-803 FS 1.19 NM Section 5.9.6
h1ConfiguratorAddress	UDINT	This element contains the H1 address of the H1 configurator device. FF-803 FS 1.19 NM Section 5.9.3
h1DIOperatFunctionalClassSupported	USINT	This attribute corresponds to the class supported by all H1 interfaces of a linking device. It has the same definition of DIOperatFunctionalClass attribute specified in the Network Management Specification [FF-801]. Its value is zero when the HSE device is not a linking device. FF-803 FS 1.19 NM Section 5.2.1
h1Timeout	UDINT	This attribute defines the amount of time, in milliseconds, that the Linking Device waits before releasing invoke ids that were allocated for FMS and/or SM requests sent on any H1 interface. FF-803 FS 1.19 NM Section 5.2.1
hseSubnetMask	ARRAY OF USINT	This attribute is defined in the Field Device Access (FDA) Agent Specification [FF-588] as the HSE Subnet Mask. FF-803 FS 1.19 NM Section 5.1.3

Data type	Definition	Description
hseSubnetMaskBits	ARRAY OF USINT	This attribute when ANDED with an IP address or HSE Subnet Mask yields the HSE subnet address defined in the Field Device Access (FDA) Agent Specification [FF-588]. It is used for HSE VCRs with the HSE Subnet Mask attribute. FF-803 FS 1.19 NM Section 5.2.1
hseVcrType	UDINT	This attribute is defined in the Field Device Access (FDA) Agent Specification [FF-588] as the HSE VCR Type, and indicates the type of the HSE VCR. FF-803 FS 1.19 NM Section 5.1.3
ignoredInbound	UDINT	This element counts the number of valid frames received by the interface that the Forwarding Process determined did not need forwarding. FF-803 FS 1.19 NM Section 5.9.6
inactivityCloseTime	UDINT	This attribute is defined in the Field Device Access (FDA) Agent Specification [FF-588] as the Inactivity Close Time. DEFAULT_INACTIVITY_CLOSE_TIME = 30 s FF-803 FS 1.19 NM Section 5.1.2
installedInterfaces	ARRAY OF USINT	Installed Interfaces. The entry "Installed" means the hardware is in place and functioning. Each bit in the entry corresponds to an H1 interface with bit k corresponding to interface k. Bit 1 is the MSB. FF-803 FS 1.19 SM Section 6.2.1
interfaceLinkId	UINT	
interfaceNodeId	USINT	
interfaceNumber	USINT	This attribute specifies the index into the InterfaceAddressArray for forwarding. FF-803 FS 1.19 NM Section 5.9.5
interfaceStateArray	ARRAY OF USINT	
interfaceStatisticsSupported	ARRAY OF USINT	The value of this element indicates the interface statistics supported. FF-803 FS 1.19 NM Section 5.9.2
lastSNTPMessage	ARRAY OF USINT	This object holds the first 48 octets of the last SNTP message received. This object contains only dynamic data, which does not affect the calculation of version numbers. FF-803 FS 1.19 SM Section 6.2.2
maxForwardingDelayNormal	UINT	This attribute is defined in the Data Link Protocol Specification: Bridge Operation Addendum [FF-806]. 1/32 ms FF-803 FS 1.19 NM Section 5.9.2
maxForwardingDelayTimeAvailable	UINT	This attribute is defined in the Data Link Protocol Specification: Bridge Operation Addendum [FF-806]. Unit = 1/32 ms. FF-803 FS 1.19 NM Section 5.9.2
maxForwardingDelayUrgent	UINT	This attribute is defined in the Data Link Protocol Specification: Bridge Operation Addendum [FF-806]. Unit = 1/32 ms. FF-803 FS 1.19 NM Section 5.9.2
maxMessageLength	UDINT	This attribute is defined in the Field Device Access (FDA) Agent Specification [FF-588] as the Max Message Length. FF-803 FS 1.19 NM Section 5.1.2
maxNumberOfInterfaces	UDINT	The entry "Max number of interfaces (M)" indicates the maximum number of H1 interfaces that can be installed into the device. The maximum value for this entry is 30. See the Network Management specification [FF-803] on Bridge Characteristics. FF-803 FS 1.19 SM Section 6.2.1
maxNumberOfVFDs	UDINT	Maximal Number of VFDs. FF-803 FS 1.19 SM Section 6.2.1
maxNumEntries	UDINT	This element indicates the maximum number of entries that may be defined in the list (NumConfigured + NumUnconfigured). FF-803 FS 1.19 NM Section 5.1.1
maxNumOfVcrs	UDINT	This element contains the counter for the maximum number of HSE VCRs that have been concurrently related to the Session Endpoint Entry object. This element represents the "high water mark" for the number of VCRs related to this session. This attribute may not be written. FF-803 FS 1.19 NM Section 5.5.2