

Australian Standard™

**Industrial automation systems and
integration—Open systems application
integration framework**

**Part 3: Reference description for
IEC 61158-based control systems**



This Australian Standard was prepared by Committee IT-006, Information Technology for Industrial Automation and Integration. It was approved on behalf of the Council of Standards Australia on 30 March 2004 and published on 3 June 2004.

The following are represented on Committee IT-006:

Association of Consulting Engineers Australia
Australian Electrical and Electronic Manufacturers Association
CSIRO Centre for Planning and Design
CSIRO Manufacturing & Infrastructure Technology
Department of Defence (Australia)
Institute of Instrumentation, Control and Automation Australia
Institution of Engineers Australia
Monash University
RMIT University
The University of Melbourne

Keeping Standards up-to-date

Standards are living documents which reflect progress in science, technology and systems. To maintain their currency, all Standards are periodically reviewed, and new editions are published. Between editions, amendments may be issued. Standards may also be withdrawn. It is important that readers assure themselves they are using a current Standard, which should include any amendments which may have been published since the Standard was purchased.

Detailed information about Standards can be found by visiting the Standards Web Shop at www.standards.com.au and looking up the relevant Standard in the on-line catalogue.

Alternatively, the printed Catalogue provides information current at 1 January each year, and the monthly magazine, *The Global Standard*, has a full listing of revisions and amendments published each month.

Australian Standards™ and other products and services developed by Standards Australia are published and distributed under contract by SAI Global, which operates the Standards Web Shop.

We also welcome suggestions for improvement in our Standards, and especially encourage readers to notify us immediately of any apparent inaccuracies or ambiguities. Contact us via email at mail@standards.org.au, or write to the Chief Executive, Standards Australia International Ltd, GPO Box 5420, Sydney, NSW 2001.

This Standard was issued in draft form for comment as DR 04112.

STANDARDS AUSTRALIA

RECONFIRMATION
OF
AS ISO 15745.3—2004
Industrial automation systems and integration—Open systems application
integration framework
Part 3: Reference description for IEC 61158-based control systems

RECONFIRMATION NOTICE

Technical Committee IT-006 has reviewed the content of this publication and in accordance with Standards Australia procedures for reconfirmation, it has been determined that the publication is still valid and does not require change.

Reconfirmed documents cannot be amended, but may be formally revised and a new edition published.

Certain documents referenced in the publication may have been amended since the original date of publication. Users are advised to ensure that they are using the latest versions of such documents as appropriate, unless advised otherwise in this Reconfirmation Notice.

Approved for reconfirmation in accordance with Standards Australia procedures for reconfirmation on 14 July 2015.

The following are represented on Technical Committee IT-006:

Australia Safety Critical Systems Association
Australian Computer Society
Australian Industry Group
Australian Petroleum Production and Exploration Association
Consult Australia
Engineers Australia
Institute of Chemical Engineers Australia
Institute of Instrumentation, Control & Automation
ISACA
Process Control Society
The University of Queensland
Workplace Health and Safety Queensland

NOTES

Australian Standard™

**Industrial automation systems and
integration—Open systems application
integration framework**

**Part 3: Reference description for
IEC 61158-based control systems**

First published as AS ISO 15745.3—2004.

COPYRIGHT

© Standards Australia International

All rights are reserved. No part of this work may be reproduced or copied in any form or by any means, electronic or mechanical, including photocopying, without the written permission of the publisher.

Published by Standards Australia International Ltd
GPO Box 5420, Sydney, NSW 2001, Australia

ISBN 0 7337 5952 1

PREFACE

This Standard was prepared by the Standards Australia Committee IT-006, Information Technology for Industrial Automation and Integration.

This Standard is identical with, and has been reproduced from, ISO 15745-3:2003, *Industrial automation systems and integration—Open systems application integration framework, Part 3: Reference description for IEC 61158-based control systems*.

The objective of this Standard is to provide the technology specific elements and rules for describing both communication network profiles and the communication related aspects of device profiles specific to IEC 61158-based control systems.

This Standard is Part 3 of AS ISO 15745—2004, *Industrial automation systems and integration—Open systems application integration framework*, which is published in parts as follows:

Part 1: Generic reference description

Part 2: Reference description for ISO 11898-based control systems

Part 3: Reference description for IEC 61158-based control systems (this Standard)

Part 4: Reference description for Ethernet-based control systems

The terms ‘normative’ and ‘informative’ are used to define the application of the annex to which they apply. A normative annex is an integral part of a standard, whereas an informative annex is only for information and guidance.

As this Standard is reproduced from an international standard, the following applies:

- (a) Its number appears on the cover and title page while the international standard number appears only on the cover.
- (b) In the source text ‘this International Standard’ should read ‘this Australian Standard’.
- (c) A full point substitutes for a comma when referring to a decimal marker.

References to International Standards should be replaced by references to Australian or Australian/New Zealand Standards, as follows:

<i>Reference to International Standard</i>		<i>Australian Standard</i>	
ISO		AS	
3166-1	Codes for the representation of names of countries and their subdivisions— Part 1: Country codes	2632.1	Codes for the representation of names of countries and their subdivisions, Part 1: Country codes
ISO		AS ISO	
15745	Industrial automation systems and integration—Open systems application integration framework	15745	Industrial automation systems and integration—Open systems application integration framework
15745-1	Part 1: Generic reference description	15745.1	Part 1: Generic reference description

CONTENTS

Page

Introduction	v
1 Scope	1
2 Normative references	1
3 Terms and definitions.....	2
4 Abbreviated terms.....	3
5 Technology specific elements and rules.....	4
5.1 Integration models and IAS interfaces	4
5.2 Profile templates	4
5.2.1 General.....	4
5.2.2 Contents and syntax.....	4
5.2.3 Header	5
5.3 Technology specific profiles	5
6 Device and communication network profiles for IEC61158-based control systems	6
6.1 ControlNet.....	6
6.1.1 Device profile.....	6
6.1.2 Communication network profile	8
6.2 PROFIBUS.....	10
6.2.1 Device profile.....	10
6.2.2 Communication network profile	10
6.3 P-NET	12
6.3.1 Device profile.....	12
6.3.2 Communication network profile	13
6.4 WorldFIP	15
6.4.1 Device profile.....	15
6.4.2 Communication network profile	18
6.5 INTERBUS.....	25
6.5.1 Device profile.....	25
6.5.2 Communication network profile	32
Annex A (normative) ControlNet profile templates	36
A.1 General.....	36
A.2 Device profile template description	37
A.2.1 Device profile template description – XML based	37
A.2.2 Device profile template description – XML encapsulation of EDS files	55
A.3 Communication network profile template description.....	57
A.3.1 Communication network profile template description – XML based	57
A.3.2 Communication network profile template description – XML encapsulation of EDS files	75
A.4 Electronic Data Sheet (EDS)	76
A.4.1 Common CIP EDS requirements	76
A.4.2 ControlNet specific EDS requirements.....	116
Annex B (normative) PROFIBUS profile templates	121
B.1 General.....	121
B.2 Device profile template description	121
B.2.1 General.....	121
B.2.2 XML schema: GSD_Device_Profile_wrapper.xsd or EDD_Device_Profile_wrapper.xsd	122
B.3 Communication network profile template description.....	124
B.3.1 General.....	124
B.3.2 XML schema: GSD_CommNet_Profile_wrapper.xsd	124
B.4 Generic Station Description (GSD)	125

	<i>Page</i>
B.4.1 General	125
B.4.2 Syntax and format of the GSD files	126
B.5 Semantic of GSD	127
B.5.1 Conventions.....	127
B.5.2 General specifications.....	128
B.5.3 Master-related specifications.....	135
B.5.4 Slave-related specifications.....	142
B.6 Formal description of GSD.....	165
Annex C (normative) P-NET profile templates	179
C.1 Device profile template description	179
C.2 Communication network profile template description.....	181
Annex D (normative) WorldFIP profile templates	184
D.1 Device profile template description	184
D.1.1 Overview	184
D.1.2 DeviceConformityClass.....	184
D.1.3 Device profile template XML schema.....	187
D.2 Communication network profile template description.....	190
D.2.1 Overview	190
D.2.2 Application layers	190
D.2.3 Transport layers; DLConformityClass	193
D.2.4 Network Management	194
D.2.5 Communication network profile template XML schema	196
Annex E (normative) INTERBUS profile templates.....	205
E.1 Device profile template description	205
E.1.1 Overview	205
E.1.2 Basics.....	205
E.1.3 DeviceIdentity object - deviceType object.....	207
E.1.4 DeviceManager object	209
E.1.5 Supplementary element descriptions	214
E.1.6 Device profile template XML schemas.....	217
E.2 Communication network profile template description.....	264
Bibliography	269

INTRODUCTION

The application integration framework (AIF) described in ISO 15745 defines elements and rules that facilitate:

- the systematic organization and representation of the application integration requirements using integration models;
- the development of interface specifications in the form of application interoperability profiles (AIPs) that enable both the selection of suitable resources and the documentation of the "as built" application.

ISO 15745-1 defines the generic elements and rules for describing integration models and AIPs, together with their component profiles - process profiles, information exchange profiles, and resource profiles. The context of ISO 15745 and a structural overview of the constituents of an AIP are given in Figure 1 of ISO 15745-1:2003.

This part of ISO 15745 extends the generic AIF described in ISO 15745-1 by defining the technology specific elements and rules for describing both communication network profiles and the communication related aspects of device profiles specific to control systems based on IEC 61158 (P-NET^{®1}, PROFIBUS², WorldFIP^{®3}, ControlNet^{™4}, and INTERBUS^{®5}). These technologies use profiles of IEC 61158 which are specified in IEC 61784-1. Profiles for ISO/IEC 8802-3-based control systems are outside the scope of this part of ISO 15745 and are specified in ISO 15745-4.

In particular, this part of ISO 15745 describes technology specific profile templates for the device profile and the communication network profile. Within an AIP, a device profile instance or a communication network profile instance is part of the resource profile defined in ISO 15745-1. The device profile and the communication network profile XML instance files are included in a resource profile XML instance using the ProfileHandle_DataType as specified in ISO 15745-1:2003, 7.2.5.

AIFs specified using the elements and rules of ISO 15745-1 can be easily integrated with the component profiles defined using the elements and rules specified in this part.

¹ P-NET is the registered trademark of the International P-NET User Organisation Aps (IPUO). Control of trademark use is given to the non profit organisation IPUO. This information is given for the convenience of users of this International Standard and does not constitute an endorsement by ISO of the trademark holder or any of its products. Compliance to this standard does not require use of the trademark P-NET. Use of the trademark P-NET requires permission of the IPUO..

² PROFIBUS is the trade name of the PROFIBUS Nutzerorganisation e.V. (PNO), control of trade name use is given to the non profit organisation PNO. This information is given for the convenience of users of this International Standard and does not constitute an endorsement by ISO of the trademark holder or any of its products. Compliance to this standard does not require use of the trade name PROFIBUS. Use of the trade name PROFIBUS requires permission of the PNO.

³ WorldFIP[®] is a registered trademark of the WorldFIP Association. Control of trademark use is given to the non profit organisation WorldFIP Association. This information is given for the convenience of users of this International Standard and does not constitute an endorsement by ISO of the trademark holder or any of its products. Compliance to this standard does not require use of the trademark WorldFip. Use of the trademark WorldFIP requires permission of the WorldFIP Association.

⁴ ControlNet[™] is a trade name of ControlNet International, Ltd. This information is given for the convenience of users of ISO 15745 and does not constitute an endorsement by ISO of the trademark holder or any of its products. Compliance to this standard does not require use of the trade name ControlNet[™]. Use of the trade name ControlNet[™] requires permission of ControlNet International, Ltd.

⁵ INTERBUS is a trade name of Phoenix Contact GmbH & Co. KG, control of trade name use is given to the non profit organisation INTERBUS Club. This information is given for the convenience of users of this International Standard and does not constitute an endorsement by ISO of the trademark holder or any of its products. Compliance to this standard does not require use of the trade name INTERBUS. Use of the trade name INTERBUS requires permission of the INTERBUS Club.

NOTES

AUSTRALIAN STANDARD

Industrial automation systems and integration — Open systems application integration framework —**Part 3:
Reference description for IEC 61158-based control systems****1 Scope**

This part of ISO 15745 defines the technology specific elements and rules for describing both communication network profiles and the communication related aspects of device profiles specific to IEC 61158-based control systems. Profiles for ISO/IEC 8802-3-based control systems are outside the scope of this part of ISO 15745.

NOTE Generic elements and rules for describing integration models and application interoperability profiles, together with their component profiles (process profiles, information exchange profiles, and resource profiles) are specified in ISO 15745-1.

This part of ISO 15745 is to be used in conjunction with ISO 15745-1 to describe an application integration framework.

2 Normative references

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

ISO 639-1:2002, *Codes for the representation of names of languages – Part 1: Alpha-2 code*

ISO 639-2:1998, *Codes for the representation of names of languages – Part 2: Alpha-3 code*

ISO 3166-1:1997, *Codes for the representation of names of countries and their subdivisions – Part 1: Country codes*

ISO 9506-1:2000, *Industrial automation systems – Manufacturing Message Specification – Part 1: Service definition*

ISO 15745-1:2003, *Industrial automation and systems integration – Open systems application integration framework – Part 1: Generic reference description*

ISO/IEC 10646-1:2000, *Information technology – Universal Multiple-Octet Coded Character Set (UCS) – Part 1: Architecture and Basic Multilingual Plane*

IEC 61131-3:2003, *Programmable controllers – Part 3: Programming languages*

IEC 61158 (all parts), *Digital data communications for measurement and control – Fieldbus for use in industrial control systems*

IEC 61784-1:2003, *Digital data communications for measurement and control - Part 1: Profile sets for continuous and discrete manufacturing relative to fieldbus use in industrial control systems*

IEC 61804-2, *Function blocks (FB) for process control – Part 2: Specification of FB concept and electronic device description language (EDDL)*⁶

ANSI TIA/EIA-232-F:1997, *Interface Between Data Terminal Equipment and Data Circuit-Terminating Equipment Employing Serial Binary Data Interchange*

ANSI TIA/EIA-485-A:1998, *Electrical Characteristics of Generators and Receivers for Use in Balanced Digital Multipoint Systems*

EN 50170:1996 Volume 3 Part 7-3, *General purpose field communication system – WorldFIP – Network Management*

IEEE Std 754-1985 (R1990), *IEEE Standard for Binary Floating Point Arithmetic*

REC-xml-20001006, *Extensible Markup Language (XML) 1.0 Second Edition – W3C Recommendation 6 October 2000*

REC-xmlschema-1-20010502, *XML Schema Part 1: Structures – W3C Recommendation 02 May 2001*

REC-xmlschema-2-20010502, *XML Schema Part 2: Datatypes – W3C Recommendation 02 May 2001*

RFC 1738:1994, *Uniform Resource Locators (URL) – Internet Engineering Task Force (IETF), Request for Comments (RFC)*

RFC 1759:1995, *Printer MIB – Internet Engineering Task Force (IETF), Request for Comments (RFC)*

UML V1.4, *OMG - Unified Modeling Language Specification (Version 1.4, September 2001)*

3 Terms and definitions

NOTE The UML terminology and notation used in this document is described in Annex A of ISO 15745-1:2003.

For the purposes of this document, the terms and definitions given in ISO 15745-1 apply.

⁶ Edition 1 to be published

4 Abbreviated terms

AIF	Application Integration Framework
AIP	Application Interoperability Profile
AL	Application Layer
ASCII	American Standard Code for Information Interchange (see ISO/IEC 10646)
ASE	Application Service Element (see IEC 61158-5)
CIP™ ⁷	Common Industrial Protocol
CP	Communication Profile
CRC	Cyclic Redundancy Check
DL	Data Link Layer
DP	PROFIBUS DP services and protocol
EDD	Electronic Device Description
EDDL	Electronic Device Description Language
EDS	Electronic Data Sheet
FDCML	Field Device Configuration Markup Language
FIP	Field Industrial Protocol
GSD	Generic Station Description
HMI	Human Machine Interface
I/O	Input and Output
IAS	Industrial Automation Systems
ID	Identifier
kbit/s	1024 bit/s
LSB	Least Significant Bit
MAU	Medium Attachment Unit
Mbit/s	1024*1024 bits/s
MCS	Messaging Common Services
MMS	Manufacturing Messages Specifications (see ISO 9506-1)
MPS	Manufacturing Periodic/aperiodic Services
MS1	Master class 1 (see IEC 61784-1:2003)
MS2	Master class 2 (see IEC 61784-1:2003)

⁷ CIP™ is a trade name of ControlNet International, Ltd. and Open DeviceNet Vendor Association, Inc. This information is given for the convenience of users of ISO 15745 and does not constitute an endorsement by ISO of the trademark holder or any of its products. Compliance to this standard does not require use of the trade name CIP™. Use of the trade name CIP™ requires permission of either ControlNet International, Ltd. or Open DeviceNet Vendor Association, Inc

NC	Numerical Control
NM	Network Management
OSI	Open System Interconnection
PC	Personal Computer
PID	Proportional Integration Differentiation controller
PLC	Programmable Logic Controller
RC	Robot Control
SM_MPS	System Management Manufacturing Periodic Specification
SMS	System Management Specification
SubMMS	Subset of MMS
SWNo	Softwire Number
UML	Unified Modelling Language (see UML V1.4)
VMD	Virtual Managed Device
XML	eXtensible Markup Language (see REC-xml-20001006)

5 Technology specific elements and rules

5.1 Integration models and IAS interfaces

The AIP developer shall develop the integration model using the rules described in ISO 15745-1, and shall ensure that the IEC 61158-based device and communication network profiles (whether representing the interface requirements or those derived from existing devices/communication networks) include the necessary IAS interfaces. The IAS interfaces included in the profile shall be identified in the header section (see ISO 15745-1:2003, 7.2.2).

NOTE IAS interfaces are described in ISO 15745-1:2003, Annex B.

5.2 Profile templates

5.2.1 General

The IEC 61158 technology specific profile templates are derived from the generic profile templates specified in ISO 15745-1:2003, clause 7.

5.2.2 Contents and syntax

ISO 15745 specifies profile templates that are XML schemas (REC-xmlschema-1-20010502 and REC-xmlschema-2-20010502) and use a common general structure. The device and communication network profiles based on these templates typically contain :

- information needed to identify the connected device,
- a description of device data that can be accessed via the network,
- a description of the communication capabilities supported by the device,
- additional vendor-specific information.

However, some IEC 61158 technologies use specific legacy ASCII syntax. Hence, for backward compatibility, template definitions of any technology (Annex A to Annex E) include all or a relevant subset of the following:

- communication network and device profile templates, as defined in ISO 15745-1,
- ISO 15745 template to encapsulate files with legacy ASCII syntax ("wrapper"),
- legacy ASCII syntax.

5.2.3 Header

The profile template header defined in ISO 15745-1:2003, 7.2.2, is used for IEC 61158 technology specific profile templates. Each technology uses one or more names to identify the technology or its particular component(s) (see Table 1). The selected name shall be stored in the ProfileTechnology attribute in the header section.

Table 1 — ProfileTechnology names

ProfileTechnology name	Technology
ControlNet	ControlNet
CIP	ControlNet
EDS	ControlNet
GSD	PROFIBUS
EDDL	PROFIBUS
P-NET	P-NET
WorldFIP	WorldFIP
INTERBUS	INTERBUS
FDCML	INTERBUS

5.3 Technology specific profiles

The technology specific communication network profile structure and communication related aspects of device profile structure based on IEC 61158 fieldbus technologies are described in clause 6. The technologies included are:

- ControlNet (see 6.1)
- PROFIBUS (see 6.2)
- P-NET (see 6.3)
- WorldFIP (see 6.4)
- INTERBUS (see 6.5).

The related profile template definitions are specified in Annex A to Annex E.

6 Device and communication network profiles for IEC 61158-based control systems

6.1 ControlNet

6.1.1 Device profile

6.1.1.1 General

Figure 1 shows the class structure of the ControlNet device profile.

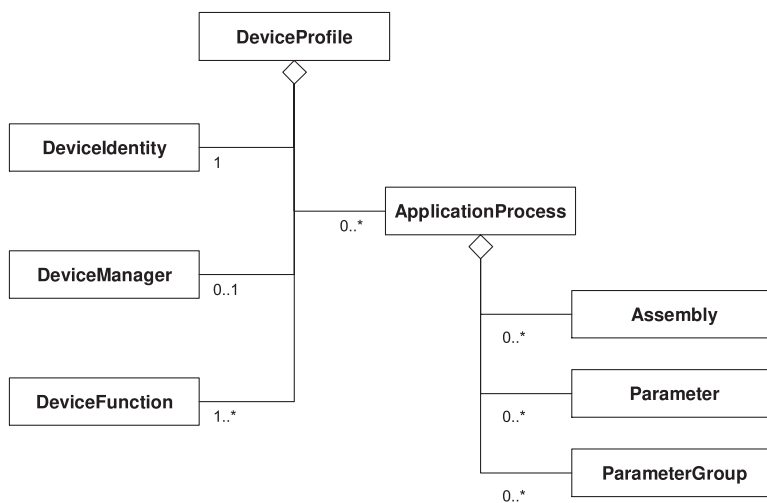


Figure 1 — ControlNet device profile class diagram

The available formats for ControlNet device profiles are described in A.2.

The XML schema representing the ControlNet device profile template is defined in A.2.1.3.3. The file name of this XML schema shall be "CIP_Device_Profile.xsd".

NOTE The ControlNet device profile class diagram shown in Figure 1 defines the main classes. These classes are further decomposed ; details are defined in Annex A.

The XML schema representing the encapsulation of a legacy ControlNet EDS into the ISO 15745 device profile template is defined in A.2.2.2. The file name of this XML schema shall be "EDS_Device_Profile_wrapper.xsd". The legacy EDS ASCII syntax itself is described in A.4.

6.1.1.2 Device identity

The **DeviceIdentity** class contains attributes which uniquely identify the device, and supports services which allow the retrieval of this information from the device.

These attributes provide in particular:

- manufacturer's identification (name and identification code);
- device identification (device type, product name, revision, serial number);
- device classification;
- location of storage of additional information (e.g. icons).

6.1.1.3 Device manager

The DeviceManager class contains attributes and supports services used to monitor and configure the device.

These attributes provide in particular:

- revision of the ControlNet identity object;
- information on device structure (for devices integrated in a modular system).

Services allow:

- device reset;
- retrieval of DeviceManager attributes.

6.1.1.4 Device function

The DeviceFunction class contains attributes and supports services which enable the management (e.g. configuration) of a function of the device.

EXAMPLE Examples of DeviceFunction objects are Overload, Presence Sensing, Analogue Input, and Discrete Output objects.

NOTE The DeviceFunction class is not defined in ISO 15745-3.

6.1.1.5 Application process

Figure 2 shows the class structure of the ApplicationProcess class.

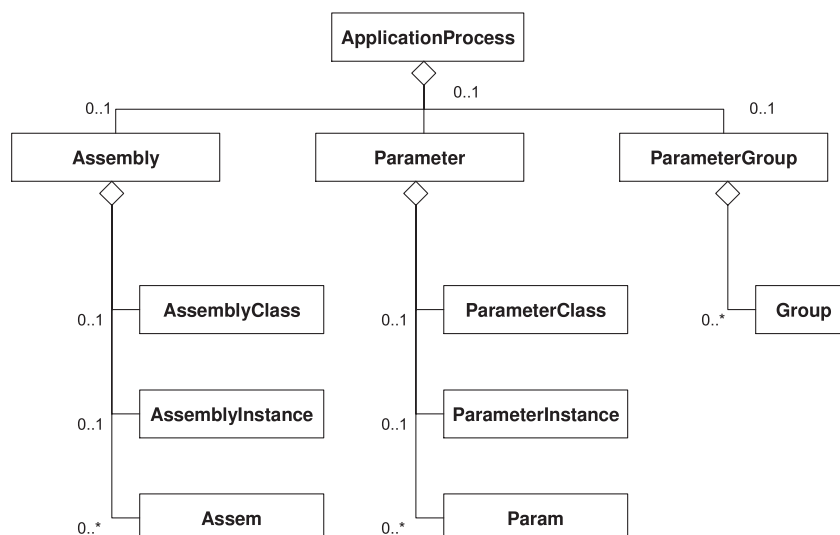


Figure 2 — ControlNet ApplicationProcess class diagram

The Assembly class assembles several application process data items into a single block for optimisation of communications. The Parameter class provides a standardized interface for accessing individual application process data items. The ParameterGroup class specifies groups of related parameters for a specific purpose (e.g. configuration, monitoring).

The Assembly class and the Parameter class support attributes and services both at the class and instance levels.

The Assem, Param and Group classes specify individual instances of the main classes.

NOTE The Assembly class and the Parameter class correspond to the ControlNet Assembly object and Parameter object. The Assembly object is fully specified in IEC 61158-5:2003 and IEC 61158-6:2003 (Type 2).

6.1.2 Communication network profile

6.1.2.1 General

Figure 3 shows the class structure of the ControlNet communication network profile.

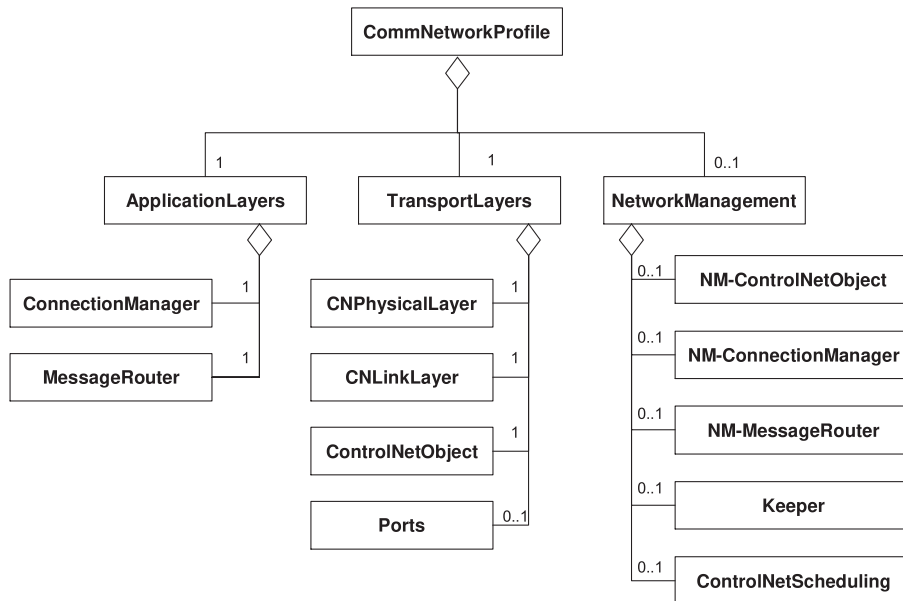


Figure 3 — ControlNet communication network profile class diagram

The available formats for ControlNet communication network profiles are described in A.3.

The XML schema representing the ControlNet communication network profile template is defined in A.3.1.3. The file name of this XML schema shall be "CNet_CommNet_Profile.xsd".

The XML schema representing the encapsulation of a legacy ControlNet EDS into the ISO 15745 communication network profile template is defined in A.3.2.2. The file name of this XML schema shall be "EDS_CommNet_Profile_wrapper.xsd". The legacy EDS ASCII syntax itself is described in A.4.

6.1.2.2 Application layers

The ControlNet ApplicationLayers class represents the combined profiles for the upper 3 OSI layers of the ControlNet communication network integration model.

It is further divided into several classes, as shown in Figure 3:

- ConnectionManager defines the properties associated with connections and connection management;
- MessageRouter defines the properties associated with internal message routing in the device.

NOTE The corresponding Connection Manager object and Message Router object are fully specified in IEC 61158-5:2003 and IEC 61158-6:2003 (Type 2).

6.1.2.3 Transport layers

The ControlNet TransportLayers class represents the combined profiles for the lower 4 OSI layers of the ControlNet communication network integration model.

It is further divided into several classes, as shown in Figure 3:

- CNPhysicalLayer identifies the physical layer characteristics (e.g. connectors, delays);
- CNLinkLayer and ControlNetObject define the properties associated with data link layer configuration and monitoring;
- Ports identifies the device ports which are able to route messages from one link to another link.

NOTE The corresponding ControlNet object is fully specified in IEC 61158-4:2003 (Type 2).

6.1.2.4 Network management

The ControlNet NetworkManagement class represents the network configuration and performance adjustment capabilities of the ControlNet communication network integration model.

It is further divided into several classes, as shown in Figure 3:

- Keeper defines the properties associated with network management;
- ControlNetScheduling defines the properties associated with allocation of scheduled transmission time;
- NM-ConnectionManager , NM-MessageRouter and NM-ControlNetObject define the properties associated with class management of the corresponding objects.

NOTE The corresponding Keeper object and ControlNet Scheduling object are fully specified in IEC 61158-4:2003 (Type 2).

6.2 PROFIBUS

6.2.1 Device profile

Figure 4 shows the class structure of the PROFIBUS device profile.

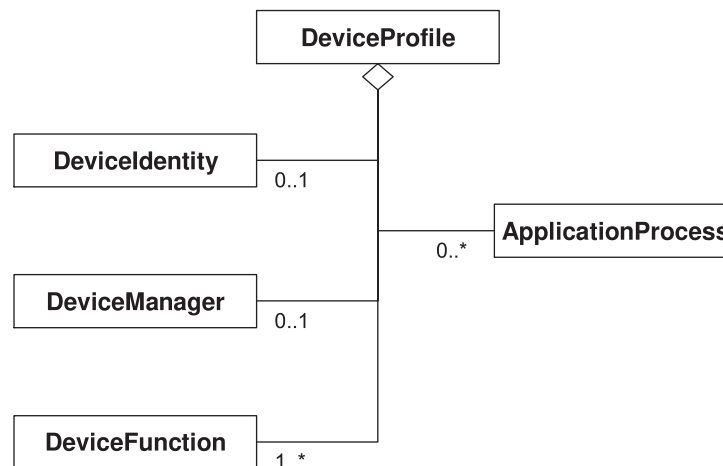


Figure 4 — PROFIBUS device profile class diagram

The class information for DeviceIdentity, DeviceManager, and ApplicationProcess is given by the legacy device profile format EDD and GSD.

The DeviceFunction class contains attributes and supports services which enable the management (e.g. configuration) of a function of the device.

EXAMPLE Examples of DeviceFunction objects are Analogue Input, and Discrete Output objects.

NOTE The DeviceFunction class is not defined in ISO 15745-3.

The available formats for PROFIBUS device profiles are described in B.2.

The XML schema representing the encapsulation of a legacy EDD and GSD of a PROFIBUS device into the ISO 15745 device profile template is defined in B.2. The file name of this XML schema shall be “EDDL_Device_Profile_wrapper.xsd” or “GSD_Device_Profile_wrapper.xsd”. The legacy EDDL ASCII syntax itself is described in IEC 61804-2, using the PROFIBUS profile of IEC 61804-2: Ed.1, F.2.

Simple devices, which do not have the need to have an EDD, shall reference the GSD.

Equivalences of the classes DeviceIdentity, DeviceManager, and ApplicationProcess are defined in IEC 61804-2 or in the case of referencing an GSD in B.4.

6.2.2 Communication network profile

6.2.2.1 General

Figure 5 shows the class structure of the PROFIBUS communication network profile.

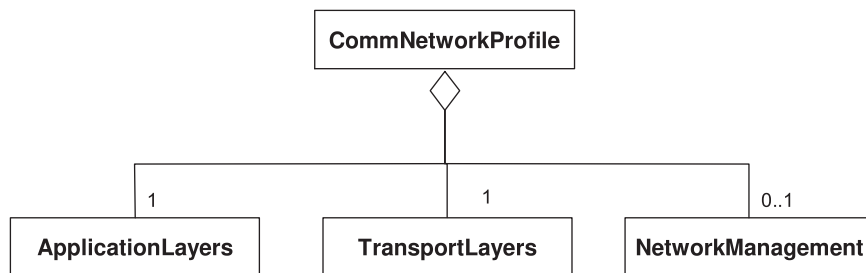


Figure 5 — PROFIBUS communication network profile class diagram

The available formats for PROFIBUS communication network profiles are described in B.3.

The file name of this XML schema shall be “GSD_CommNet_Profile.xsd”.

The XML schema representing the encapsulation of a legacy GSD of PROFIBUS devices into the ISO 15745 communication network profile template is defined in B.3. The file name of this XML schema shall be “GSD_CommNet_Profile_wrapper.xsd”. The legacy GSD ASCII syntax itself is described in B.4 and B.5.

6.2.2.2 Application layers

The PROFIBUS ApplicationLayers class represents the combined profiles for the upper 3 OSI layers of the PROFIBUS communication network integration model, see IEC 61784-1:2003 CP3/1 and CP3/2, especially the subclauses for AL protocol and AL service.

6.2.2.3 Transport layers

The PROFIBUS TransportLayers class represents the combined profiles for the lower 4 OSI layers of the PROFIBUS communication network integration model, see IEC 61784-1:2003 CP3/1 and CP3/2, especially the subclauses for DL protocol and DL service.

6.2.2.4 Network management profile

The NetworkManagement profile class represents the network configuration and performance adjustment capabilities of the PROFIBUS communication network integration model.

6.3 P-NET

6.3.1 Device profile

Variables in P-NET devices are normally organised into Channels. A Channel is a collection of related variables and functions for a single process signal. A Channel can hold up to 16 registers, each having its own SoftWire number (SWNo). The contents of these registers can be of any data type, including complex structures, such as multi-dimensional arrays and databases.

At least one Channel, the ServiceChannel, shall be present in every P-NET device. The ServiceChannel contains information used for making easy service on the device, e.g. a Globally Unique Identifier (GUID), error information, etc. Any other Channels that are included in a device depend on the device type. A device can consist of multiples of the same Channel type and/or a mixture of different Channel types.

EXAMPLE 1 Examples of various standardised Channel types include: Digital Input, Digital Output, Analogue measurement, PID, etc. Additional Channel types are user definable.

A P-NET device profile shall describe all network accessible variables and channels in the device in the ApplicationProcess object. The device profile also includes objects for device management and identification (see Figure 6).

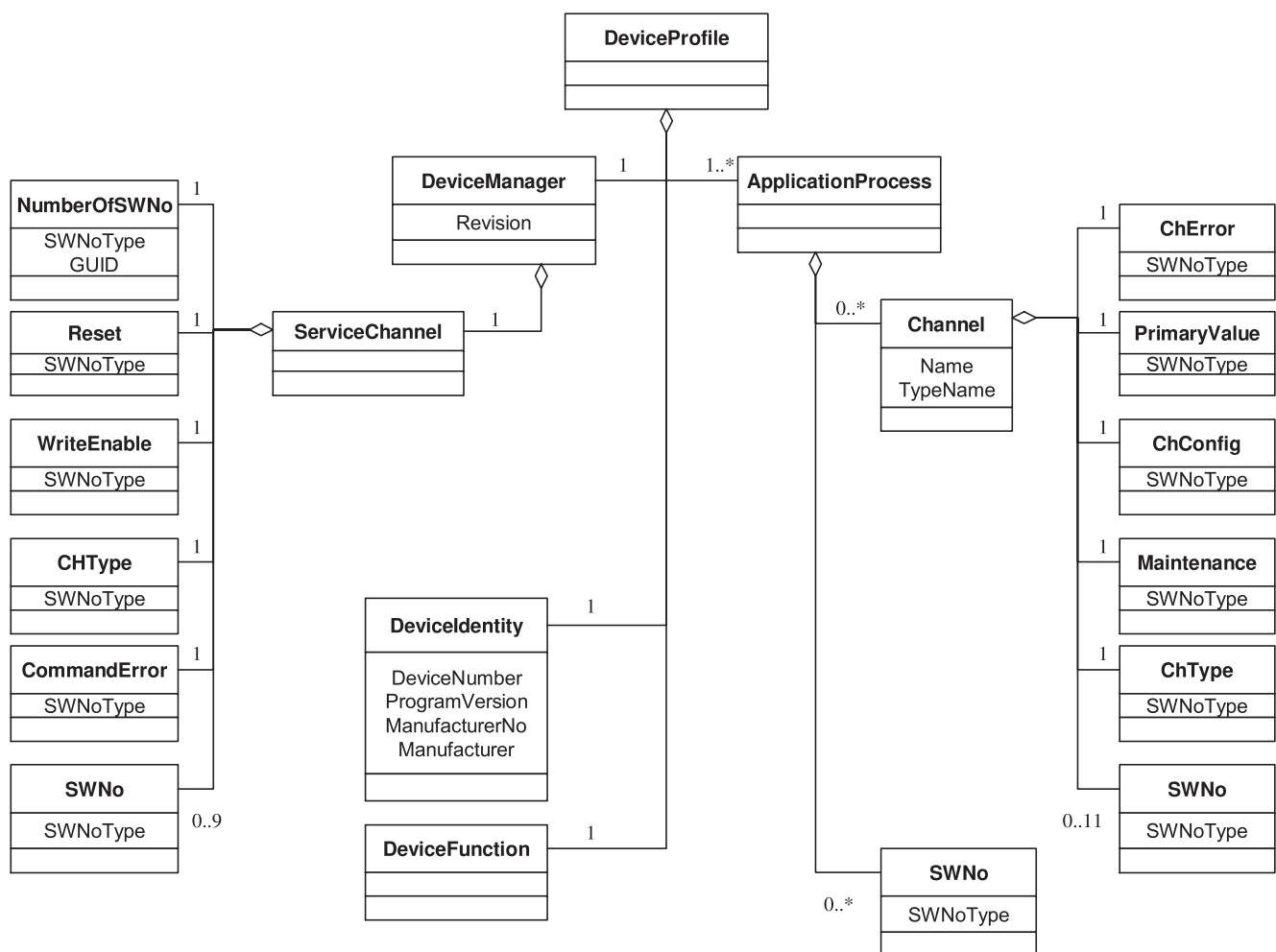


Figure 6 — P-NET device profile class diagram

EXAMPLE 2 Examples of Device Function objects are Flowmeter, Controller, Analogue Input 4-20 mA, Digital I/O objects.

NOTE In P-NET, there are no special services for accessing properties of the DeviceFunction, DeviceManager and DeviceIdentity objects. Instead, these properties can be mapped to the ApplicationProcess objects, and accessed from the network by using the Application Protocol Data Unit (APDU) specified in the P-NET ApplicationLayers class (see 6.3.2.2).

Attributes and sub-classes for the device profile classes are detailed in C.1, which specifies the XML schemas required for device profiles. The file name of the XML schema shall be "P-NetDeviceProfile.XSD".

6.3.2 Communication network profile

6.3.2.1 General

Figure 7 shows the class structure of the P-NET communication network profile.

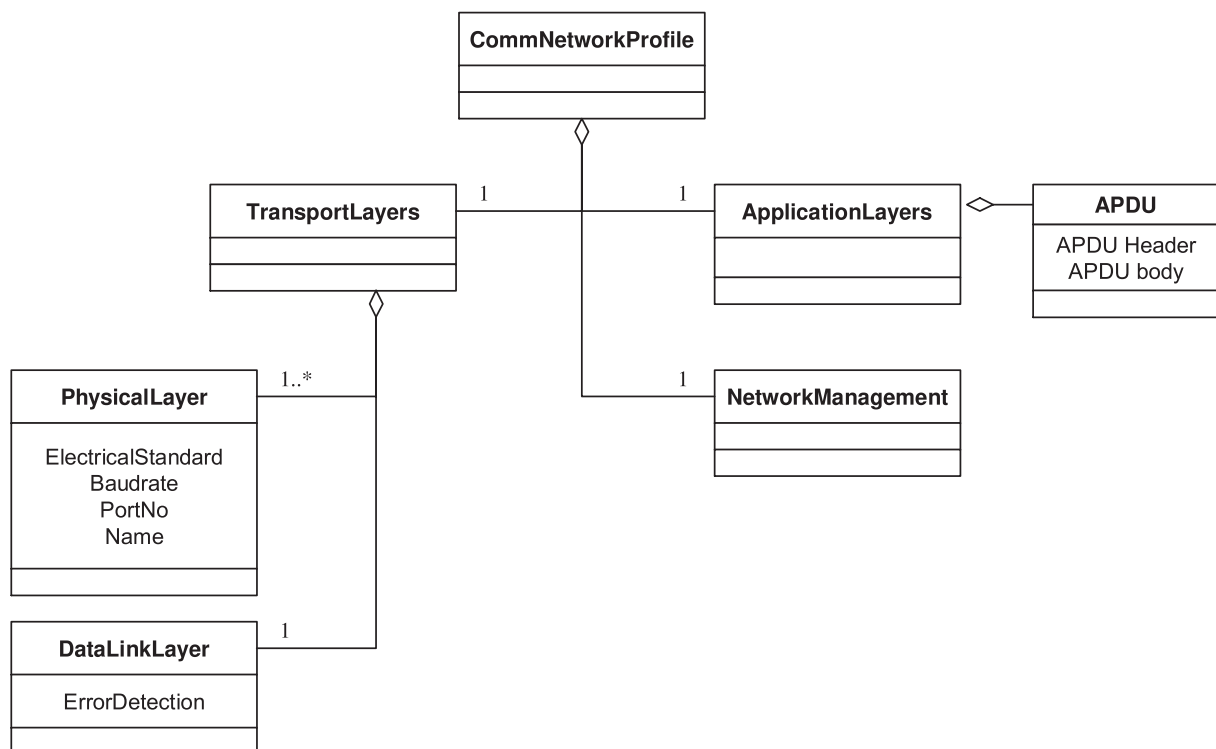


Figure 7 — P-NET communication network profile class diagram

The XML schema representing the P-NET communication network profile is defined in C.2. The file name of the XML schema shall be "P-NetCommNetworkProfile.XSD".

6.3.2.2 Application layers

The application layer of a P-NET device shall always provide the services and protocol elements needed for accessing the variables defined in the P-NET device profile. The services and protocol elements are represented in an APDU.

6.3.2.3 Transport layers

6.3.2.3.1 General

The TransportLayers class shall represent the combined profiles for the lower 4 OSI layers of the communication network integration model. The TransportLayers object is composed of one or more PhysicalLayer objects (one for each physical port) and a DataLinkLayer object.

6.3.2.3.2 PhysicalLayer

The PhysicalLayer object(s) shall specify the supported electrical standard and baud rate(s). Valid electrical standards are RS485 and RS232. Valid baud rates for RS232 are 1200, 2400, 4800, 9600, 19200, and 38400 bits/sec. For RS485 the only valid baud rate is 76800 bits/sec.

6.3.2.3.3 DataLinkLayer

The Data-Link layer of a P-NET device shall always provide the protocol elements needed according to the device class, specified by the NetworkManagement object. Valid device classes are Slave, Simple node or Master.

The DataLinkLayer object shall specify the supported error detection method(s). Valid methods are Normal and Reduced.

6.3.2.4 NetworkManagement

The NetworkManagement class diagram is shown in Figure 8. It shall comprise of at least one object of type Master, Simple Node or Slave for each physical connection point defined in the OSI-U profile.

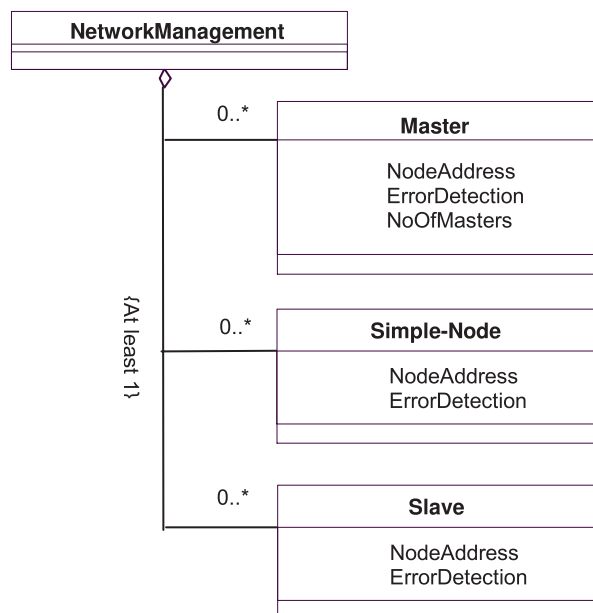


Figure 8 — P-NET NetworkManagement profile class diagram

6.4 WorldFIP

6.4.1 Device profile

6.4.1.1 Main class structure

Figure 9 shows the class structure of the WorldFIP device profile.

The required format for WorldFIP device profiles is described in D.1. The XML schema representing the WorldFIP device profile template is defined in D.1.3. The file name of the XML schema shall be 'WFIPDEVP.XSD'.

NOTE 1 For better readability the WorldFIP device profile class diagram has been divided in four class diagrams.

NOTE 2 All these classes are mapped to the same XML schema defined in D.1.3.

NOTE 3 The WorldFIP device profile class diagrams shown in Figure 9 to Figure 12 define the main classes. Some classes are further decomposed; details are defined in D.1.

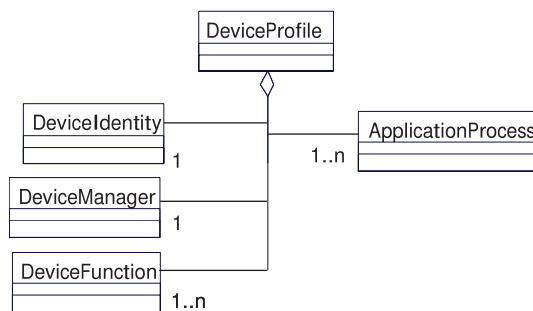


Figure 9 — WorldFIP device profile class diagram

6.4.1.2 Device identity

The **DeviceIdentity** class shall consist of the child classes shown in Figure 10 and specified in Table 2.

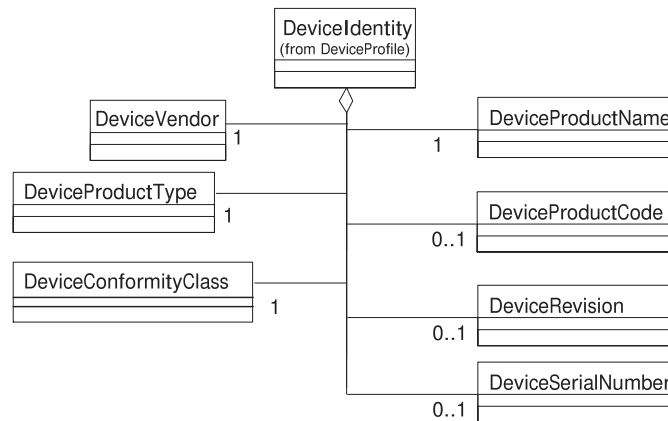


Figure 10 — WorldFIP DeviceIdentity class diagram

Table 2 — Decomposition of DeviceIdentity class

Class	Description	Profile	Type	Instance
DeviceVendor	name of the manufacturer or vendor of the device	X	X	X
DeviceProductType	device type	X	X	X
DeviceConformityClass	class of conformity (see D.1.2)	X	X	X
DeviceProductName	vendor specific name of the product	X	X	X
DeviceProductCode	unique ID, identifying the device type, the format is at the vendor's discretion		X	X
DeviceRevision	revision of the specification to which this device conforms	X	X	X
DeviceSerialNumber	serial number of device instance			X
NOTE The columns Profile, Type and Instance indicate whether a certain child class is suitable for usage in a device profile, device type description or device instance description.				

6.4.1.3 Device manager

6.4.1.3.1 General

Figure 11 shows the WorldFIP representation of the DeviceManager object.

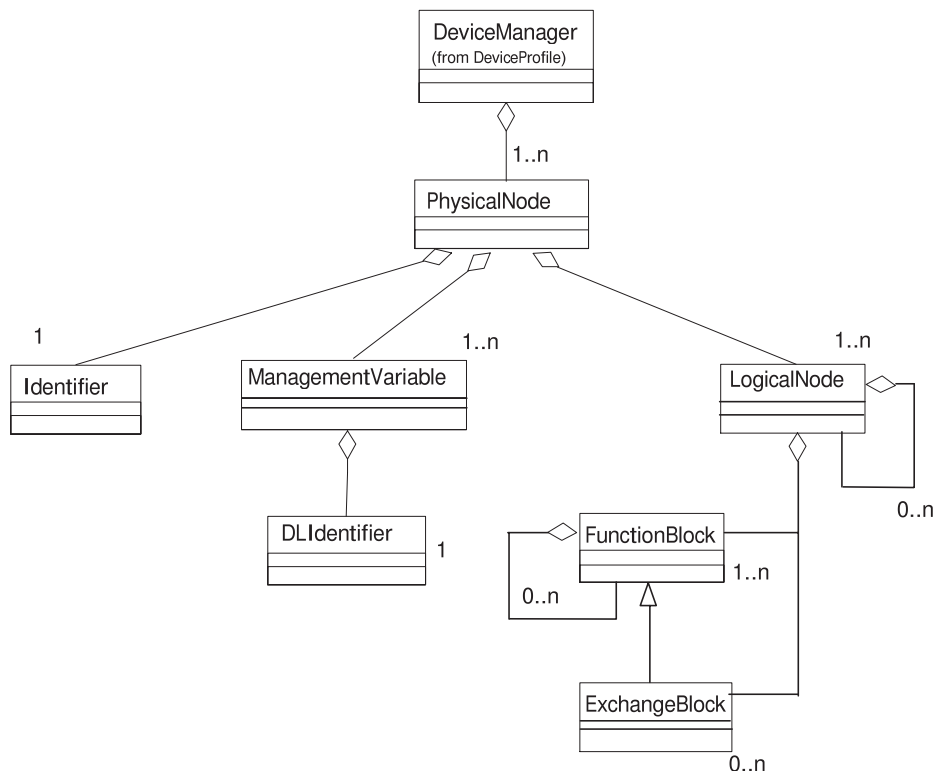


Figure 11 — WorldFIP DeviceManager class diagram

Any device is represented by a physical node which refers to a WorldFIP equipment, participating in application functions and supporting communication functions.

NOTE A device may comprise several physical nodes if it possesses several points to the same network or to different networks (e.g.: gateways, centralization equipment, etc.).

6.4.1.3.2 PhysicalNode

6.4.1.3.2.1 Identifier

Each physical node has a unique identifier.

6.4.1.3.2.2 ManagementVariable

Physical nodes are managed with SM_MPS and implement two or more management variables whose roles and identifiers are fixed, based on the value of DeviceConformityClass (see D.1.2).

6.4.1.3.2.3 LogicalNode

A user application in a physical node is decomposed in a series of logical nodes (at least one), representing an execution context of a processing operation performed in a physical node on behalf of the user application. A user application is described by one or more logical nodes.

A logical node may itself be composed of logical nodes.

Details for logical nodes are given in D.1.2.

6.4.1.3.2.4 FunctionBlock

The application functions of a logical node are composed of a set of user layer objects being represented by function blocks.

A function block allows identification and definition of a simple or complex elementary processing operation.

A function block may itself be composed of function blocks.

Details for function blocks are given in D.1.2.

6.4.1.3.2.5 ExchangeBlock

Exchange blocks are specialised function blocks. Their role is to model the mechanisms of the data exchanges which are necessary between the various remote user entities (i.e. logical nodes) located in different devices (i.e. physical nodes) in order to perform and, if necessary, to synchronize their processing functions.

6.4.1.4 Device function

Figure 12 shows the WorldFIP representation of the DeviceFunction object.

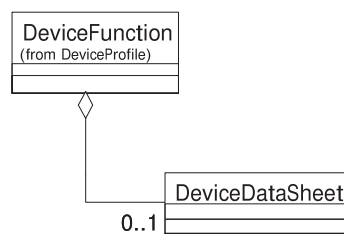


Figure 12 — WorldFIP DeviceFunction class diagram

NOTE The definition of the DeviceDataSheet of the DeviceFunction is outside the scope of this International Standard.

6.4.1.5 Application process

The ApplicationProcess object is composed of function blocks and optional exchange blocks presented in 6.4.1.3.2.4 and 6.4.1.3.2.5.

6.4.2 Communication network profile

6.4.2.1 Main class structure

The required format for WorldFIP communication network profiles is described in D.2. The XML schema representing the WorldFIP communication network profile template is defined in D.2.5. The file name of the XML schema shall be 'WFIPCOMP.XSD'.

NOTE 1 For better readability the WorldFIP communication network profile class diagram has been divided in five class diagrams.

NOTE 2 All these classes are mapped to the same XML schema defined in D.2.5.

NOTE 3 The WorldFIP communication network profile class diagrams shown in Figure 13 to Figure 18 define the main classes. Some classes are further decomposed; details are defined in D.2.

Figure 13 shows the class structure of the WorldFIP communication network profile.

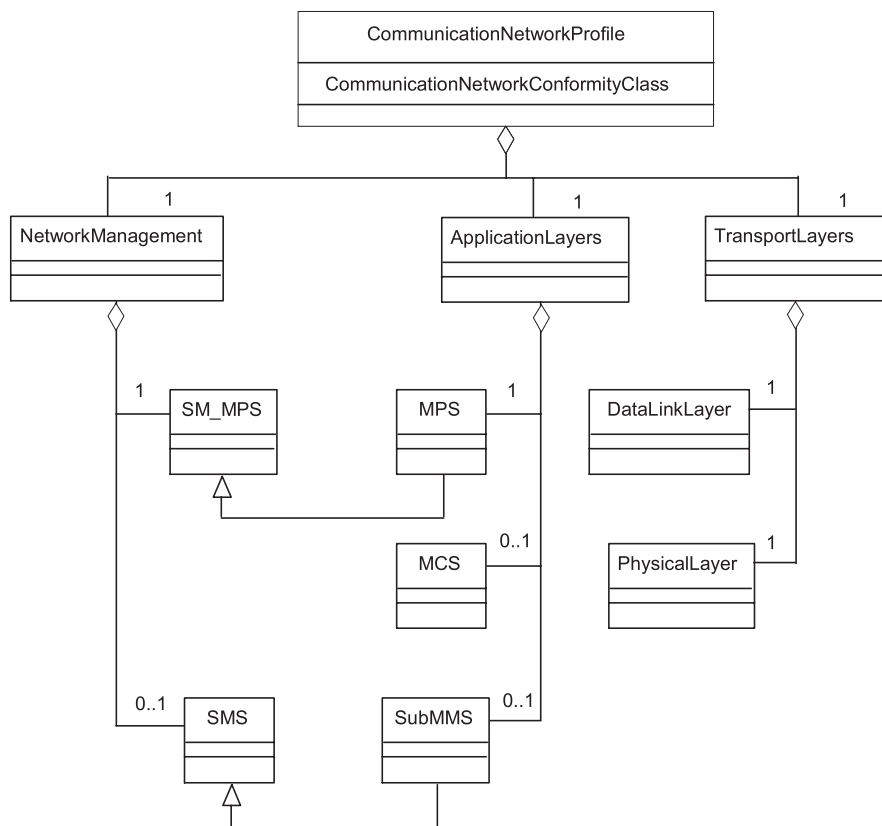


Figure 13 — WorldFIP communication network profile class diagram

The semantics of the CommunicationNetworkConformityClass attribute are detailed in D.2.

6.4.2.2 Application layers

6.4.2.2.1 General

An ApplicationLayers class shall represent the combined profiles for the upper 3 OSI layers of the communication network integration model. It states the supported application service elements and their associated services.

The ASEs (Application Service Elements) defined for WorldFIP can be stated by the following profile objects:

- MPS (Manufacturing Periodic/apperiodic Services);
MPS is a periodic/apperiodic broadcast update of a distributed database. These services are present in every WorldFIP communication network profile.
- MCS (Message Common Services);
MCS is an optional interface layer supporting messaging services.
- SubMMS;
SubMMS is an optional subset of MMS (Manufacturing Messages Specifications) services.

6.4.2.2.2 MPS

6.4.2.2.2.1 General

Figure 14 shows the representation of WorldFIP MPS application layer.

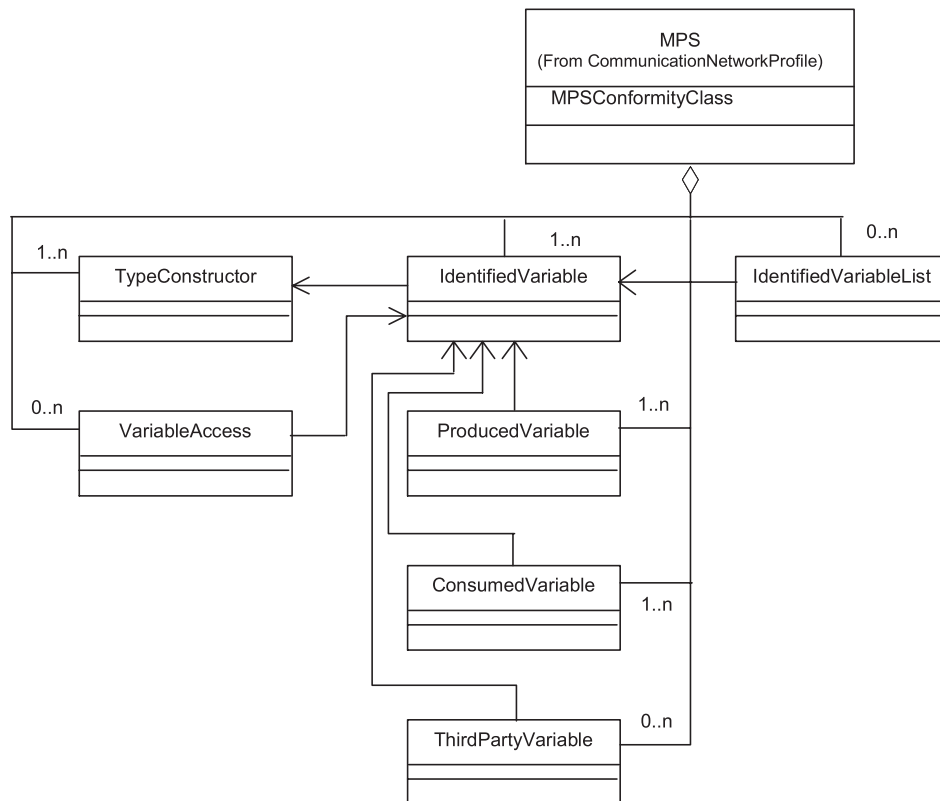


Figure 14 — WorldFIP communication network profile – MPS application layer

The semantics of the MPSConformityClass attribute are detailed in D.2.2.1.2.

6.4.2.2.2.2 IdentifiedVariable

MPS provides periodic and aperiodic exchange of variable between one producer and one or more application entities. All the common attributes of a variable are defined in the IdentifiedVariable class. Details for identified variables are given in the XML schema in D.2.5.

6.4.2.2.2.3 ProducedVariable

Within a distributed application, one unique application entity is declared as the producer of a variable's values. The ProducedVariable class gathers all the attributes relative to a variable within a producer application layer. Details for produced variables are given in the XML schema in D.2.5.

6.4.2.2.2.4 ConsumedVariable

One or several application entities are declared as consumers of a variable's values. The ConsumedVariable class gathers all the attributes relative to a variable within a consumer application layer. Details for consumed variables are given in the XML schema in D.2.5.

6.4.2.2.2.5 ThirdPartyVariable

Some application entities, while being neither producer, nor consumer of a variable's values, may have knowledge of it in order to invoke the updating of its value.

6.4.2.2.2.6 IdentifiedVariableList

The MPS application layer supports definition and handling of variable lists. A variable list is globally defined and instantiated at the application entity level. It is exclusively composed of consumed variables. Details for variable lists are given in the XML schema in D.2.5.

6.4.2.2.2.7 TypeConstructor

The variable type is defined by the TypeConstructor class which is referred by the variable. Figure 15 shows the class structure of the TypeConstructor object.

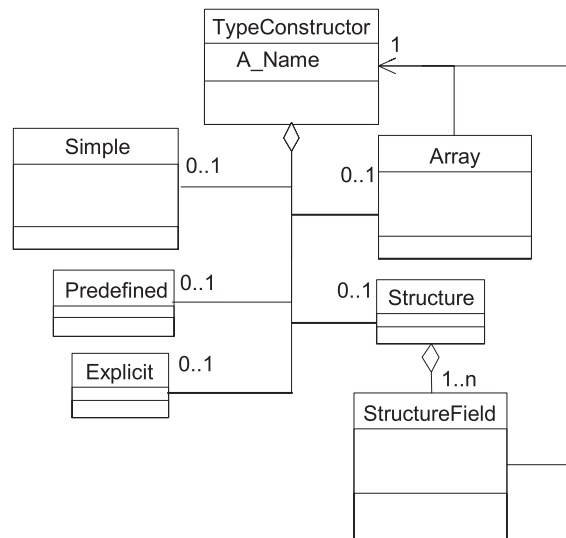


Figure 15 — WorldFIP MPS Variable TypeConstructor

Details for type constructors are given in D.2.2.1.1.

6.4.2.2.2.8 VariableAccess

The VariableAccess refers to a variable and indicates the access mode that is used. Details for variable access are given in the XML schema in D.2.5.

6.4.2.2.3 MCS

Figure 16 shows the representation of the WorldFIP MCS application layer.

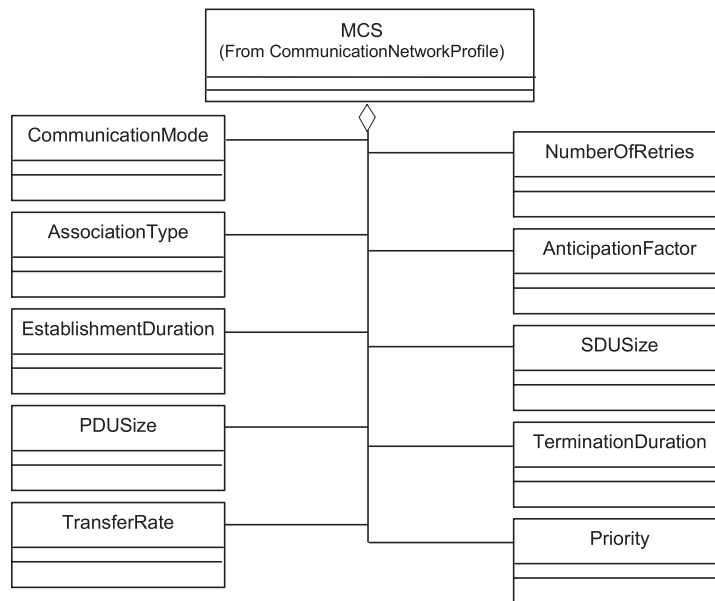


Figure 16 — WorldFIP communication network profile – MCS application layer

The MCS application layer provides services for monitoring application associations and to enable transfer of data units from the MMS application layer in the associated and non associated mode.

6.4.2.2.4 SubMMS

Figure 17 shows the representation of the WorldFIP SubMMS application layer.

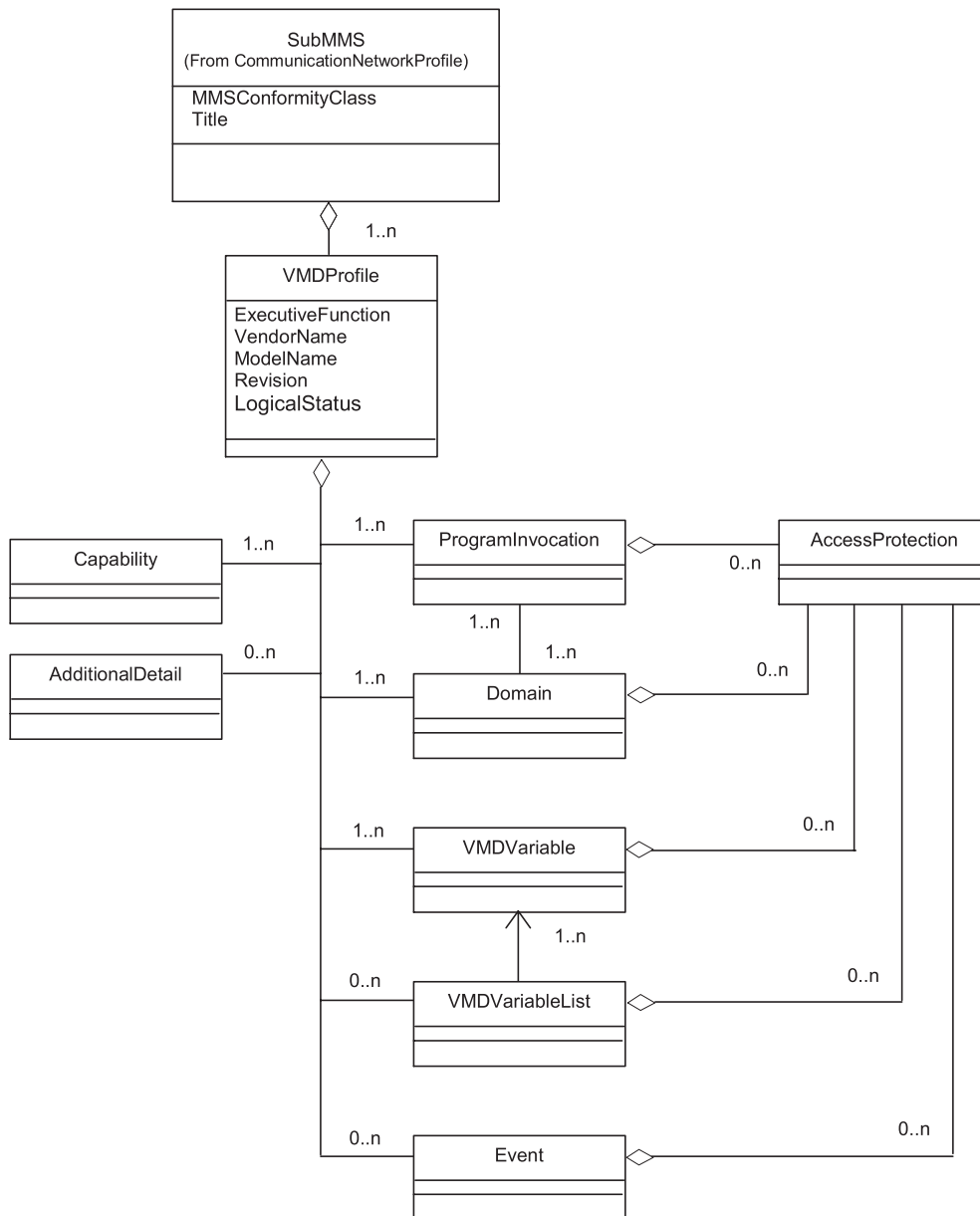


Figure 17 — WorldFIP communication network profile – SubMMS application layer

The semantics of the MMSConformityClass attribute are detailed in D.2.2.3.

6.4.2.3 Transport layers

6.4.2.3.1 General

A TransportLayers object shall represent the combined profiles for the lower 4 OSI layers of the communication network integration model. The TransportLayers object shall be divided into one or more PhysicalLayer objects and a DataLinkLayer object.

6.4.2.3.2 DataLinkLayer

6.4.2.3.2.1 General

Figure 18 shows the representation of the WorldFIP Data Link layer.

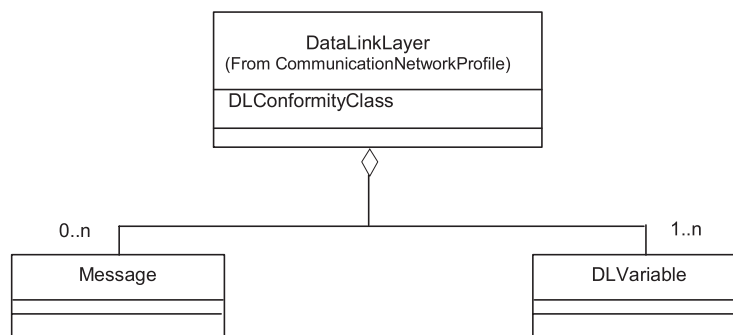


Figure 18 — WorldFIP communication network profile – Data Link layer

The semantics of the DLConformityClass attribute are detailed in D.2.3.

6.4.2.3.2 DLVariable

The WorldFIP Data Link layer supports cyclical variable exchange and explicit request for variable exchange.

Details for variables are given in the XML schema in D.2.5.

6.4.2.3.3 Message

The WorldFIP Data Link layer supports cyclical message transfer and aperiodic message transfer.

Details for messages are given in the XML schema in D.2.5.

6.4.2.3.3 PhysicalLayer

The PhysicalLayer object identifies the MAUType, the interface type and supported baud rates of the physical layer.

6.4.2.4 Network management

The NetworkManagement class shall identify the functionality for the configuration of a particular WorldFIP network. Network management for WorldFIP networks is defined in EN 50170:1996 Volume 3 Part 7-3. Depending on the network management conformity class documented in the NMConformityClass attribute, WorldFIP uses SM_MPS (based on MPS) and optionally SMS (based on MCS and SubMMS).

Network management functions are divided in three categories:

- installation and modification of the configuration,
- configuration consistency check,
- setting into service of the network.

The semantics of the NMConformityClass attribute are detailed in D.2.4.2.

6.5 INTERBUS

6.5.1 Device profile

6.5.1.1 Main class structure

Figure 19 shows the class structure of the INTERBUS device profile.

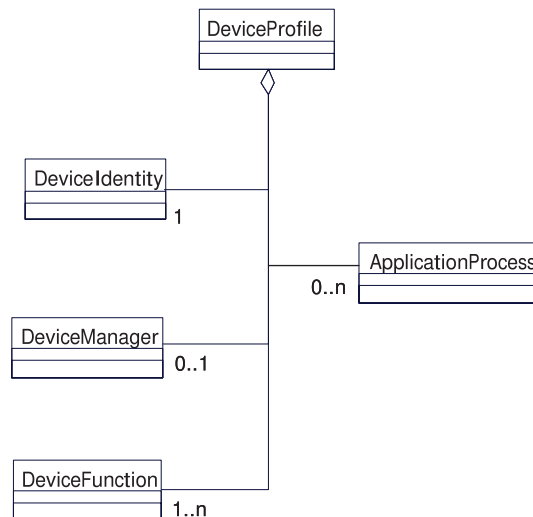


Figure 19 — INTERBUS device profile class diagram

The required format for INTERBUS device profiles is described in E.1. The XML schema representing the INTERBUS device profile template is defined in E.1.6.1. The file name of the XML schema shall be 'FDCML.XSD'.

NOTE 1 For better readability the INTERBUS device profile class diagram has been divided in six class diagrams.

NOTE 2 All these classes are mapped to the same XML schema defined in E.1.6.1.

NOTE 3 The INTERBUS device profile class diagrams shown in Figure 19 to Figure 25 define the main classes. Some classes are further decomposed; details are defined in Annex E.

6.5.1.2 Device identity

The DeviceIdentity class is defined in Figure 20.

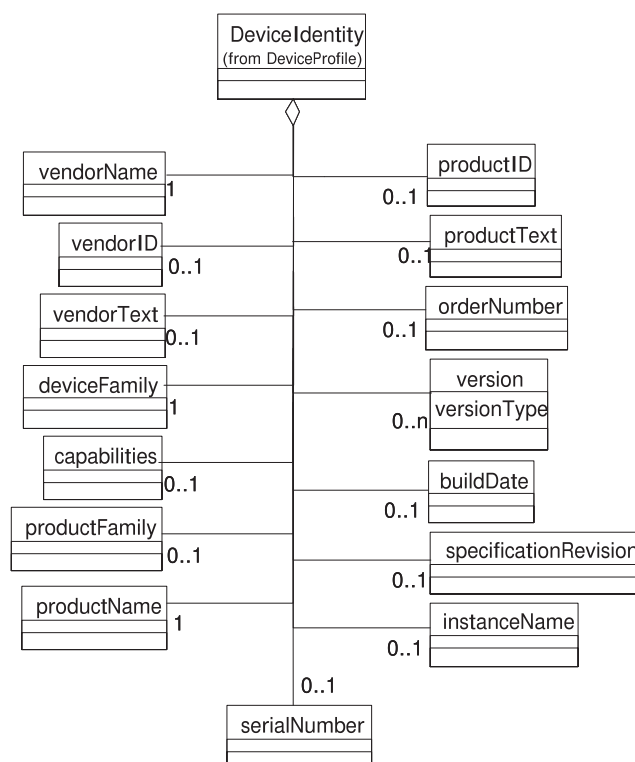


Figure 20 — INTERBUS DeviceIdentity class diagram

The DeviceIdentity class shall consist of the child classes shown in Figure 20 and specified in Table 3.

Table 3 — Decomposition of device identity object class

Class	Description	Profile	Type	Instance
vendorName	name of the manufacturer or vendor of the device	X	X	X
vendorID	IEEE OUI (Organizationally Unique Identifier) (see [3])		X	X
vendorText	can be used to provide further information on the vendor	X	X	X
deviceFamily	INTERBUS specific device type (i.e I/O). For a list of valid device types (see Table E.2)	X	X	X
capabilities	the definition of this class is not defined in ISO 15745-3		X	X
productFamily	vendor specific product family (brand name) of the device		X	X
productName	Vendor specific name of the product	X	X	X
productID	unique ID, identifying the device type, the format is at the vendor's discretion		X	X
productText	can be used to provide further information on the device	X	X	X
orderNumber	Vendor specific order number of the product		X	X
version	vendor specific product version, the versionType attribute allows the distinction of multiple versions (i.e. Hardware, Firmware)		X	X
buildDate	build date of the firmware of software constituting the major functionality of the device		X	X
specificationRevision	revision of the specification to which this device conforms	X	X	X
instanceName	name of device instance			X
serialNumber	serial number of device instance			X
NOTE The columns Profile, Type and Instance indicate whether a certain child class is suitable for usage in a device profile, device type description or device instance description.				

6.5.1.3 Device manager

6.5.1.3.1 General

Figure 21 shows the INTERBUS representation of the device manager object.

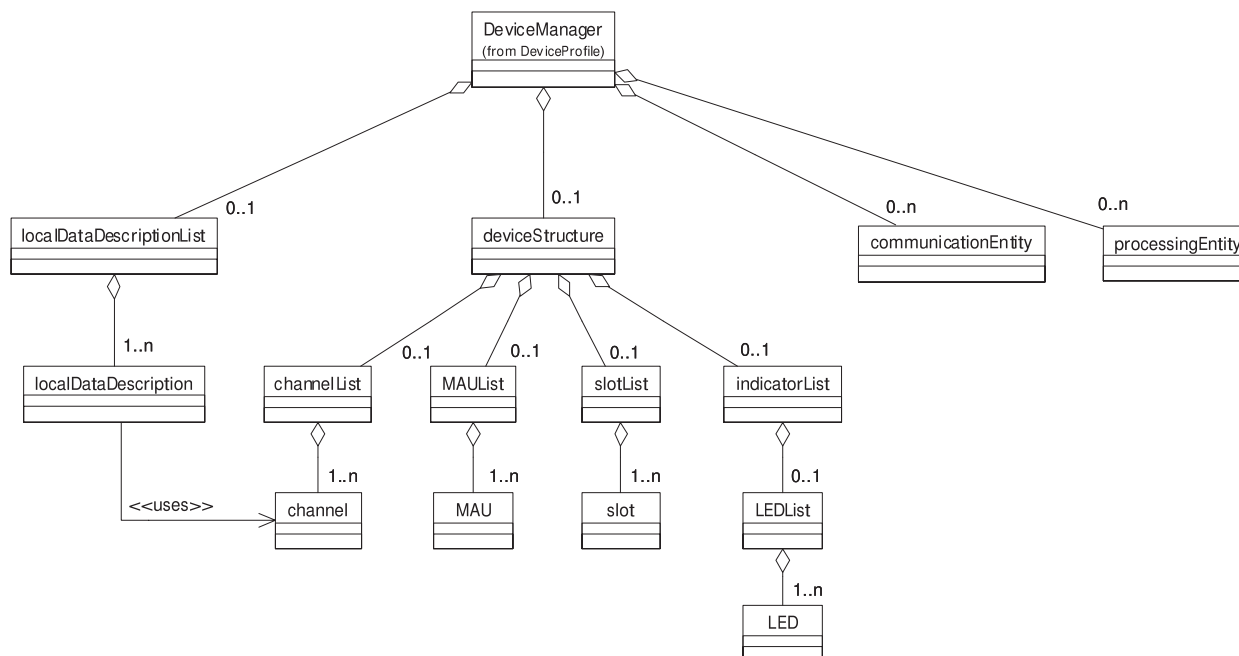


Figure 21 — INTERBUS DeviceManager class diagram

6.5.1.3.2 localDataDescriptionItem, localDataItem, localDataDescription

The localDataDescriptionItem object shall be a collection of localDataItemDescription objects. A localDataItemDescription object shall describe data objects that are used only within the device context.

6.5.1.3.3 deviceStructure

6.5.1.3.3.1 Overview

The deviceStructure object shall be a container of all physical objects of the device. Such an object can be a channel (physical or logical I/O point), a MAU (Medium Attachment Unit), a slot for the connection of additional modules (as part of the device) or a LED (light-emitting diode).

6.5.1.3.3.2 channelList, channel

A channelList shall be a collection of channel objects. Channel objects shall describe physical or logical I/O points of a device.

6.5.1.3.3.3 MAUList, MAU

The MAUList shall be a collection of MAU objects. These objects shall describe the access points to network medias.

6.5.1.3.3.4 slotList, slot

A slotList object shall be a collection of slot objects. A slot object shall contain a reference to an external INTERBUS device profile exchange description.

NOTE Slots are used to describe modular devices or distinct combinations of devices.

6.5.1.3.3.5 indicatorList, LEDList, LED

A LEDList object shall be a collection of LED objects. A LED object shall describe a LED of a device.

NOTE The indicatorList class may be extended in future editions of this International Standard.

6.5.1.3.4 communicationEntity

6.5.1.3.4.1 Overview

Figure 22 shows the definition of the communicationEntity class.

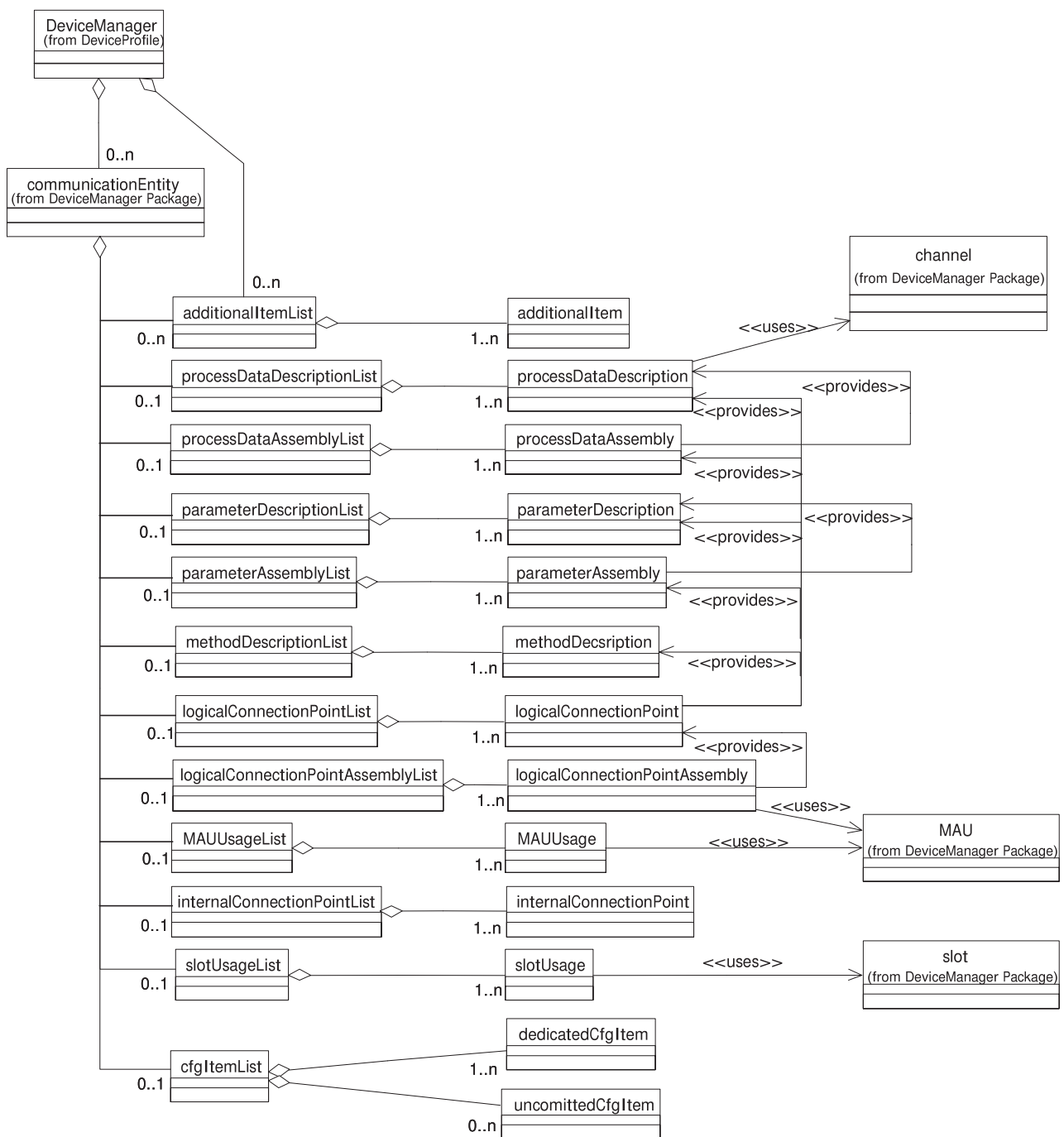


Figure 22 — INTERBUS communicationEntity class diagram

The communicationEntity shall describe an entity of a device, capable of communicating with entities of other devices and shall contain a complete set of predefined configuration items and communication object descriptions. There may be more than one communicationEntity in a device.

EXAMPLE An example for a device with two communication entities is a system coupler which 'combines' an INTERBUS slave and an INTERBUS master (see Figure 23).

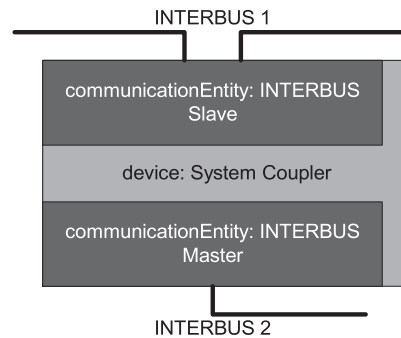


Figure 23 — Example of a device with two communication entities

6.5.1.3.4.2 additionalItemList, additionalItem

The additionalItemList shall be a collection of user defined additionalItem objects. An additionalItem object can be used to describe device properties other than configuration properties or communication objects.

NOTE The definition of the additionalItemType of additional items is outside the scope of this International Standard.

EXAMPLE Device documentation.

6.5.1.3.4.3 processDataDescriptionList, processDataDescription

The processDataDescriptionList shall be a collection of processDataDescription objects. The processDataDescription objects shall be descriptions of process data. A processDataDescription object may have a <<uses>> relation to a channel object.

EXAMPLE A process data item 'analog_out_1' uses a the terminal named '2.1'.

6.5.1.3.4.4 processDataAssemblyList, processDataAssembly

The processDataAssemblyList shall be a collection of processDataAssembly objects. The processDataAssembly objects shall be descriptions of groups of processDataDescription objects. A processDataAssembly object has <<provides>> relations to processDataDescription objects.

6.5.1.3.4.5 parameterDecsriptionList, parameterDecsription

The parameterDescriptionList shall be a collection of parameterDescription objects. The parameterDescription objects are descriptions of the variable and function invocation objects.

6.5.1.3.4.6 parameterAssemblyList, parameterAssembly

The parameterAssemblyList shall be a collection of parameterAssembly objects. The parameterAssembly objects shall be descriptions of groups of parameterDescription objects. A parameterAssembly object has <<provides>> relations to parameterDescription objects.

6.5.1.3.4.7 **methodDescriptionList, methodDescription**

The methodDescriptionList shall be a collection of methodDescription objects. The methodDescription objects are methods which can be invoked by a remote entity.

6.5.1.3.4.8 **logicalConnectionPointList, logicalConnectionPoint**

The logicalConnectionPointList shall be a collection of logicalConnectionPoint objects. A logicalConnectionPoint describes a connection endpoint (see E.1.4.6 for possible attributes).

NOTE It is assumed, that only connections between connection endpoints of the same type are used.

6.5.1.3.4.9 **logicalConnectionPointAssemblyList, logicalConnectionPointAssembly**

The logicalConnectionPointAssemblyList shall be a collection of logicalConnectionPointAssembly objects. A logicalConnectionPointAssembly shall be a description of a group of logicalConnectionPoint objects. A logicalConnectionPointAssembly object has <<provides>> relations to logicalConnectionPoint objects.

6.5.1.3.4.10 **MAUUsageList, MAUUsage**

The MAUUsageList shall be a collection of MAUUsage objects. The MAUUsage objects shall define which MAU objects are used by the communicationEntity.

6.5.1.3.4.11 **internalConnectionPointList, internalConnectionPoint**

The internalConnectionPointList shall be a collection of internalConnectionPoint objects, which define internal connections between multiple communicationEntity and/or resourceEntity objects in the same device.

6.5.1.3.4.12 **slotUsageList, slotUsage**

The slotUsageList object shall be a collection of slotUsage objects. The slotUsage objects shall define, which slots are associated with the communicationEntity.

6.5.1.3.4.13 **cfgItemList (configuration item list)**

The cfgItemList may consist of dedicatedCfgItem objects and uncommittedCfgItem objects.

6.5.1.3.4.14 **dedicatedCfgItem (dedicated configuration item)**

A dedicatedCfgItem shall be a configuration Item with a dedicatedCfgItemType attribute as defined in Table E.4 and Table E.5. A dedicatedCfgItem shall be used to specify the corresponding configuration properties.

6.5.1.3.4.15 **uncommittedCfgItem (uncommitted configuration item)**

An uncommittedCfgItem shall be a configuration item without a dedicatedCfgItemType attribute. An uncommittedCfgItem shall be used to specify configuration properties that cannot be described by a dedicatedCfgItem.

NOTE The definition of uncommitted configuration items is outside the scope of this International Standard.

EXAMPLE A description of a DIP-Switch which changes the ID code of a device.

6.5.1.3.5 **processingEntity**

Figure 24 shows the definition of the processingEntity class.

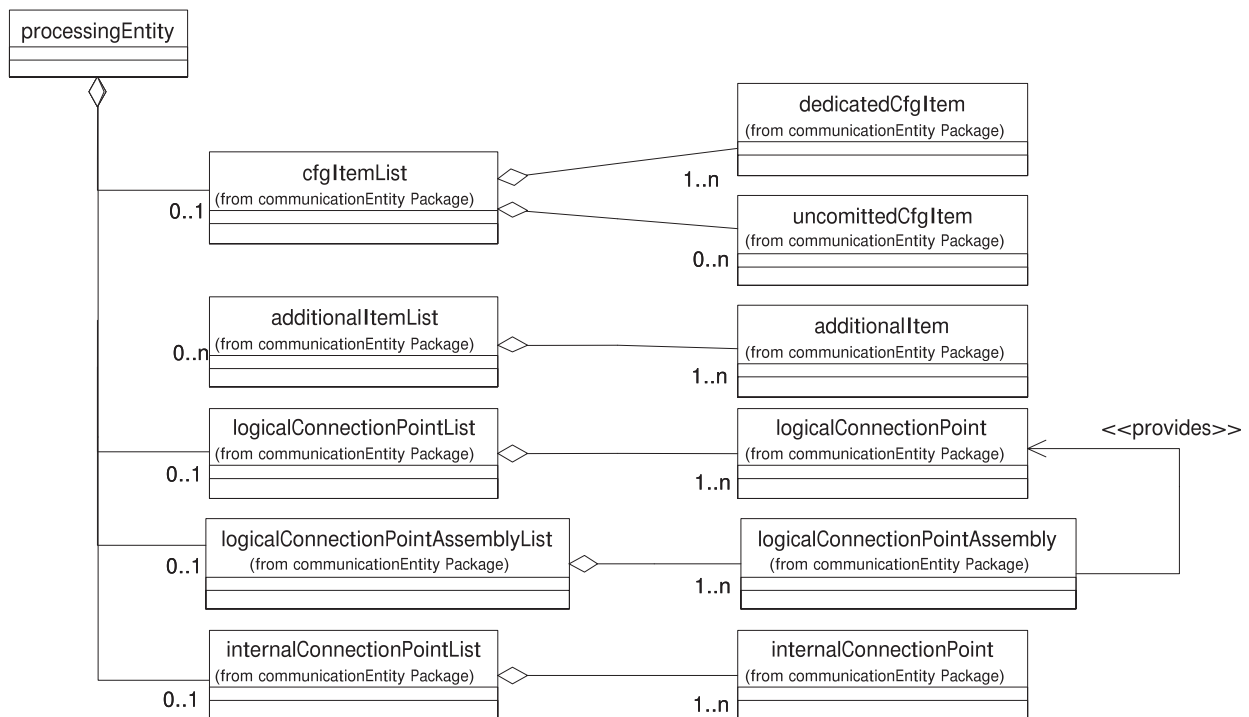


Figure 24 — INTERBUS processingEntity class diagram

A processingEntity shall describe any device entity that is not a communication entity.

NOTE For a description of the child objects see 6.5.1.3.4.

EXAMPLE A resource capable of executing programs.

6.5.1.4 Device function

6.5.1.4.1 Overview

The DeviceFunction is defined in Figure 25.

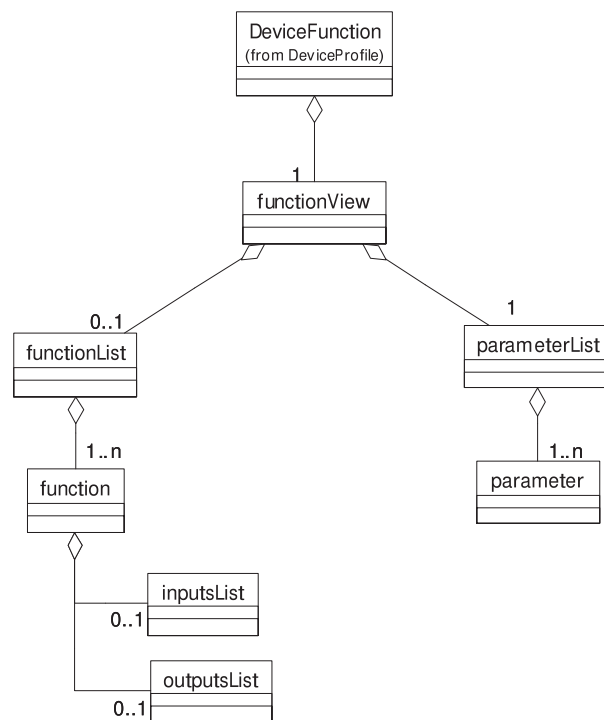


Figure 25 — INTERBUS DeviceFunction class diagram

To allow multiple representations of the device function, an additional XML schema is used to describe the DeviceFunction. The file name of this XML schema shall be "FDCMLISO15745DeviceFunction.XSD". The DeviceFunction XML schema is defined in E.1.6.3.

NOTE The definition of additional XML schemas describing the DeviceFunction is outside the scope of this International Standard.

6.5.1.4.2 parameterList, parameter

The parameterList shall be a collection of parameter objects. A parameter object describes a device parameter from a functional perspective. It is connected with a communication object in the communicationEntity.

6.5.1.4.3 functionList, function, inputsList, outputsList

The functionList shall be a collection of function objects. A function object shall consist of an inputList and an outputList. These lists shall contain a list of references to parameter objects.

6.5.1.5 Application process

The ApplicationProcess may be represented by one or more suitable XML schemas.

NOTE These XML schemas are not defined in ISO 15745-3.

6.5.2 Communication network profile

6.5.2.1 Class structure

Figure 26 shows the class structure of the INTERBUS communication network profile.

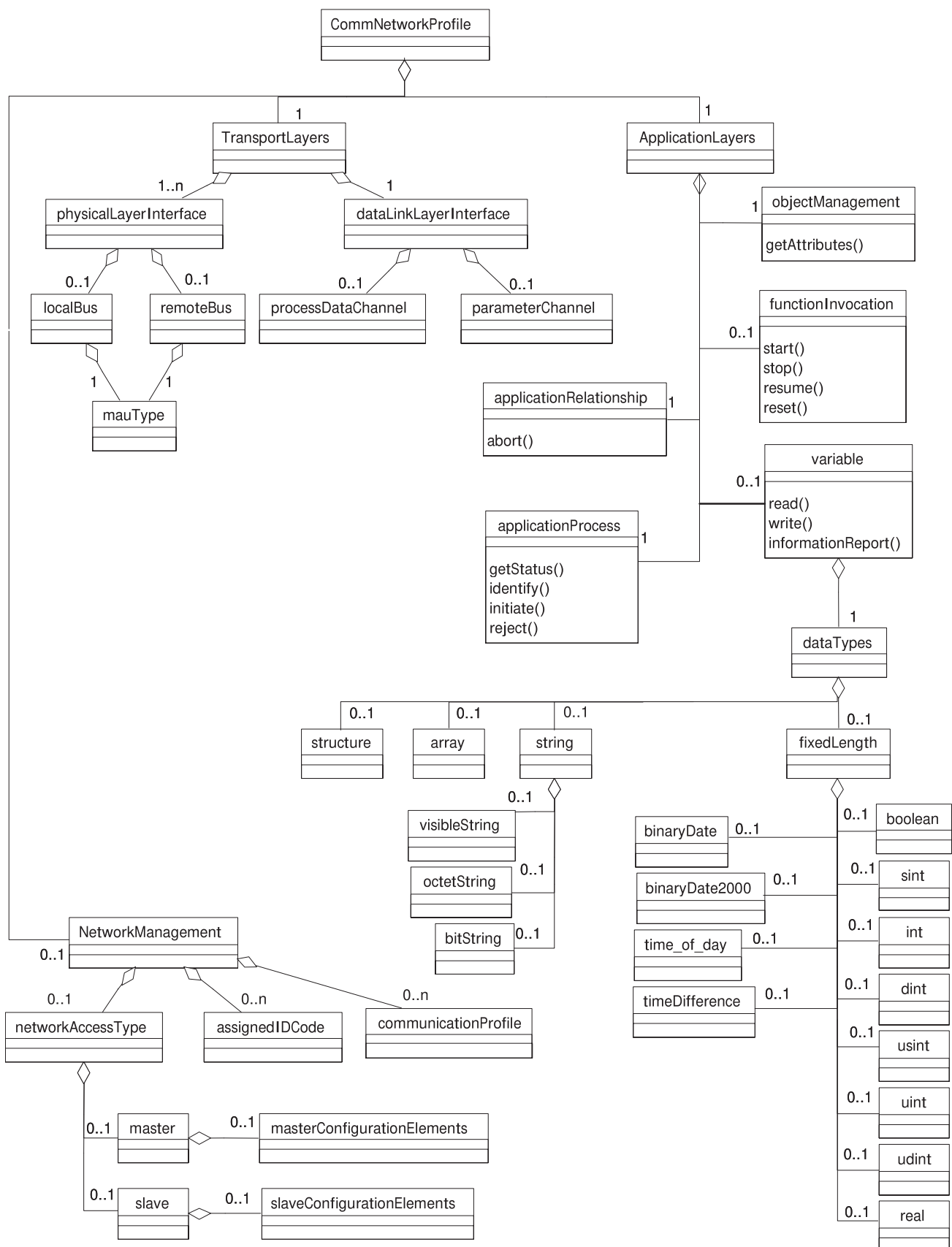


Figure 26 — INTERBUS communication network profile class diagram

The XML schema representing the INTERBUS communication network profile is defined in E.2. The file name of the XML schema shall be "INTERBUSCommNetworkProfile.XSD".

6.5.2.2 Transport layers

6.5.2.2.1 General

A TransportLayers object shall represent the combined profiles for the lower 4 OSI layers of the communication network integration model. The TransportLayers object shall be divided into one or more physicalLayerInterface objects and a dataLinkLayerInterface object.

6.5.2.2.2 physicalLayerInterface

The physicalLayerInterface profile object identifies the mauType, the interface type and supported baud rates of the physical layer interface. Valid interface types are localBus and remoteBus.

6.5.2.2.3 dataLinkLayerInterface

6.5.2.2.3.1 processDataChannel

The processDataChannel profile object shall state the length of the process data channel in bits.

6.5.2.2.3.2 parameterChannel

The parameterChannel profile class shall state the length of the parameter channel in octets.

6.5.2.3 Application layers

An ApplicationLayers class shall represent the combined profiles for the upper 3 OSI layers of the communication network integration model. It states the supported application service elements and their associated services.

The ASEs (Application Service Elements) defined in IEC 61158-5:2003, 13.2 can be stated by the following profile objects:

- applicationProcess
- applicationRelationship
- objectManagement
- functionInvocation
- variable

For the variable ASE, the selectable data types are shown in Figure 26. Data types are defined in IEC 61158-5:2003, clause 5.

6.5.2.4 Network management

6.5.2.4.1 networkAccessType, assignedIDCode

The NetworkManagement class shall identify the functionality for the configuration of a particular INTERBUS network. It contains the ID codes - least significant octet of an INTERBUS device code - assigned to a particular communication network profile, and the networkAccessType object, which contains:

- either the master profile for the INTERBUS master functionality in terms of elements (masterConfigurationElements object) necessary to configure an INTERBUS master;
- or the slave profile for the INTERBUS slave functionality in terms of elements (slaveConfigurationElements object) necessary to configure an INTERBUS slave.

NOTE An INTERBUS slave is a device that accesses the medium only after it has been initiated by the preceding slave or master. This includes remote bus devices, local bus devices and bus coupler.

6.5.2.4.2 communicationProfile

The communicationProfile shall state the usable communication profile identifiers (see 6.5.2.4.3).

6.5.2.4.3 Communication profile identifier

Communication profile identifiers are defined in IEC 61784-1:2003 subclause 10.1. An AIP Designer may specify additional communication profiles, the identifiers for such new communication profiles shall be a three digit number between 680 and 699.

Annex A (normative)

ControlNet profile templates

A.1 General

The upper layers of the ControlNet network are based on the Common Industrial Protocol (CIP). This protocol models all communication and application entities as objects. CIP specific messaging requests services to be performed on corresponding object instances (or their attributes). This scheme provides an explicit access to all configuration, status, and runtime variables data in a node. At the same time, I/O connections allow direct exchange with the I/O database, without intermediate processing. In both cases, all data references within a device are specified using a CIP path, i.e. an octet string stream that defines the application object instance, attribute and/or connection end-point.

Multiple options are available for remote configuration of devices with a CIP communication interface, including:

- device information saved in printed or electronic format;
- dedicated Parameter Objects, which provide a known public interface to individual configuration/parameter data values, and may also embed additional configuration information such as descriptive text, data type, data limits and default;
- dedicated Configuration Assembly, which allows bulk upload and download of configuration data by grouping individual configuration/parameter data values;
- combinations of the above methods.

Configuration tools currently available for CIP-based devices use a specially formatted ASCII file, referred to as the Electronic Data Sheet (EDS), which provides:

- information needed to identify the connected device;
- a description of device data that can be accessed via the network (e.g. configurable parameters);
- a description of the communication capabilities supported by the device (e.g. connections);
- additional vendor-specific information.

The EDS allows a configuration tool to automate the device configuration process. The EDS requirements provide an open, consistent and compatible approach for performing device configuration in the CIP environment.

The EDS information is very similar to the information required in both communication network and device profiles, hence the following subclauses specify format for:

- communication network and device profile templates, as defined in ISO 15745-1;
- encapsulation of legacy EDS files in the ISO 15745 templates (“wrappers”);
- the legacy Electronic Data Sheet, including common semantics information.

NOTE The ControlNet EDS (Electronic Data Sheet) of a given device can be derived from the contents of the corresponding XML device and communication network profile files, using the appropriate style sheets.

A.2 Device profile template description

A.2.1 Device profile template description – XML based

A.2.1.1 General

The device profile XML files shall comply with the device profile XML schema as specified in A.2.1.3.3.

Contents of this XML schema are derived from the device profile class diagrams shown in 6.1.1, and extended with additional elements to allow full description of device requirements or capabilities.

A.2.1.2 Semantics of XML schema elements

A.2.1.2.1 ProfileBody

This main element is associated with a set of attributes which provide additional information about the profile file.

The semantics of these attributes are specified in A.4.1.4.2.

A.2.1.2.2 DeviceIdentity

This element specifies the supported instance attributes and operations of the Identity Object (see IEC 61158-5:2003 and IEC 61158-6:2003 (Type 2)), together with additional information for full device identification. When appropriate, it also indicates the actual values of the instance attributes.

The semantics of the DeviceIdentity_InstanceAttributes sub-elements of the DeviceIdentity element are specified in Table A.1.

Table A.1 — DeviceIdentity_InstanceAttributes elements

XML schema elements	Object Attributes	Semantics
SpecificationConformance	No	String specifying the reference version of the ControlNet specifications
VendCode, ProdType, ProdCode, ProdRevision	Yes	See A.4.1.4.3
VendName, ProdTypeStr, ProdName, Catalog, Icon, ExcludeFromAdapterRackConnection	No	See A.4.1.4.3
Status, SerialNumber	Yes	Not applicable
State, ConfigurationConsistencyValue, HeartbeatInterval	Yes	Not applicable
DeviceClassification	No	See A.4.1.4.4 and A.4.2.2.1

A.2.1.2.3 DeviceManager

This element specifies the supported class attributes and operations of the Identity Object (see IEC 61158-5:2003 and IEC 61158-6:2003 (Type 2)), together with additional information for device management. When appropriate, it also indicates the actual values of the instance attributes.

The semantics of the Modular sub-element of the DeviceManager element are specified in A.4.1.5.2.

A.2.1.2.4 DeviceFunction

The contents of this element are not detailed in this document.

A.2.1.2.5 ApplicationProcess

A.2.1.2.5.1 Assembly

This element specifies the supported class and instance attributes and operations of the Assembly Object (see IEC 61158-5:2003 and IEC 61158-6:2003 (Type 2)), together with a description of the individual instances.

The semantics of the Assem, ProxyAssem and ProxiedAssem sub-elements of the Assembly element are specified in A.4.1.4.8 and A.4.1.5.3.2.

A.2.1.2.5.2 Parameter

This element specifies the supported class and instance attributes and operations of the Parameter Object, together with a description of the individual instances.

The semantics of the Parameter_ClassAttributes sub-element of the Parameter element are specified in A.4.1.4.5.

The semantics of the Param, ProxyParam and ProxiedParam sub-elements of the Parameter element are specified in A.4.1.4.6 and A.4.1.5.3.1.

A.2.1.2.5.3 ParameterGroup

This element specifies groups of related parameters for a specific purpose.

The semantics of the Group sub-element of the ParameterGroup element is specified in A.4.1.4.7.

A.2.1.3 XML schemas

A.2.1.3.1 MasterTemplateTypes.xsd

NOTE This XML schema contains all the styles defined as part of the master template in ISO 15745-1:2003.

```
<?xml version="1.0" encoding="UTF-8" ?>
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema">

<!-- Target namespaces are not specified in this master template -->

<xsd:annotation>
  <xsd:documentation>* HEADER DATA TYPES *</xsd:documentation>
</xsd:annotation>

<xsd:simpleType name="ProfileClassID_DataType">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="AIP" />
    <xsd:enumeration value="Process" />
    <xsd:enumeration value="InformationExchange" />
    <xsd:enumeration value="Resource" />
    <xsd:enumeration value="Device" />
    <xsd:enumeration value="CommunicationNetwork" />
    <xsd:enumeration value="Equipment" />
    <xsd:enumeration value="Human" />
    <xsd:enumeration value="Material" />
  </xsd:restriction>
</xsd:simpleType>

<xsd:complexType name="ISO15745Reference_DataType">
  <xsd:sequence>
    <xsd:element name="ISO15745Part" type="xsd:positiveInteger" />
    <xsd:element name="ISO15745Edition" type="xsd:positiveInteger" />
    <xsd:element name="ProfileTechnology" type="xsd:string" />
  </xsd:sequence>
</xsd:complexType>
```

```

</xsd:sequence>
</xsd:complexType>

<xsd:simpleType name="IASInterface_DataType">
  <xsd:union>
    <xsd:simpleType>
      <xsd:restriction base="xsd:string">
        <xsd:enumeration value="CSI" />
        <xsd:enumeration value="HCI" />
        <xsd:enumeration value="ISI" />
        <xsd:enumeration value="API" />
        <xsd:enumeration value="CMI" />
        <xsd:enumeration value="ESI" />
        <xsd:enumeration value="FSI" />
        <xsd:enumeration value="MTI" />
        <xsd:enumeration value="SEI" />
        <xsd:enumeration value="USI" />
      </xsd:restriction>
    </xsd:simpleType>
    <xsd:simpleType>
      <xsd:restriction base="xsd:string">
        <xsd:length value="4" />
      </xsd:restriction>
    </xsd:simpleType>
  </xsd:union>
</xsd:simpleType>

<xsd:annotation>
  <xsd:documentation>* ISO 15745 DEFINED DATA TYPES *</xsd:documentation>
</xsd:annotation>

<xsd:complexType name="ProfileHandle_DataType">
  <xsd:sequence>
    <xsd:element name="ProfileIdentification" type="xsd:string" />
    <xsd:element name="ProfileRevision" type="xsd:string" />
    <xsd:element name="ProfileLocation" type="xsd:anyURI" minOccurs="0" maxOccurs="1" />
  </xsd:sequence>
</xsd:complexType>

</xsd:schema>

```

A.2.1.3.2 CIPDataTypes.xsd

NOTE This XML schema defines the XML schema items (e.g. data types, element types, attribute groups) used in the other XML schemas.

```

<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <!-- Target namespaces are not specified in this master template -->
  <xsd:annotation>
    <xsd:documentation>* CIP DATA TYPES *</xsd:documentation>
  </xsd:annotation>
  <xsd:simpleType name="dt_USINT">
    <xsd:restriction base="xsd:unsignedByte"/>
  </xsd:simpleType>
  <xsd:simpleType name="dt_UINT">
    <xsd:restriction base="xsd:unsignedShort"/>
  </xsd:simpleType>
  <xsd:simpleType name="dt_UDINT">
    <xsd:restriction base="xsd:unsignedInt"/>
  </xsd:simpleType>
  <xsd:simpleType name="dt_ULINT">
    <xsd:restriction base="xsd:unsignedLong"/>
  </xsd:simpleType>
  <xsd:simpleType name="dt_SINT">
    <xsd:restriction base="xsd:byte"/>
  </xsd:simpleType>
  <xsd:simpleType name="dt_INT">
    <xsd:restriction base="xsd:short"/>
  </xsd:simpleType>
  <xsd:simpleType name="dt_DINT">
    <xsd:restriction base="xsd:int"/>
  </xsd:simpleType>

```

```

<xsd:simpleType name="dt_LINT">
  <xsd:restriction base="xsd:long"/>
</xsd:simpleType>
<xsd:simpleType name="dt_BYTE">
  <xsd:restriction base="xsd:hexBinary">
    <xsd:maxLength value="1"/>
  </xsd:restriction>
</xsd:simpleType>
<xsd:simpleType name="dt_WORD">
  <xsd:restriction base="xsd:hexBinary">
    <xsd:maxLength value="2"/>
  </xsd:restriction>
</xsd:simpleType>
<xsd:simpleType name="dt_DWORD">
  <xsd:restriction base="xsd:hexBinary">
    <xsd:maxLength value="4"/>
  </xsd:restriction>
</xsd:simpleType>
<xsd:simpleType name="dt_LWORD">
  <xsd:restriction base="xsd:hexBinary">
    <xsd:maxLength value="8"/>
  </xsd:restriction>
</xsd:simpleType>
<xsd:simpleType name="dt_REAL">
  <xsd:restriction base="xsd:float"/>
</xsd:simpleType>
<xsd:simpleType name="dt_LREAL">
  <xsd:restriction base="xsd:double"/>
</xsd:simpleType>
<xsd:simpleType name="dt_EDS_Char_Array">
  <xsd:restriction base="xsd:string"/>
</xsd:simpleType>
<xsd:simpleType name="dt_EPATH">
  <xsd:list itemType="et_EPATH_item"/>
</xsd:simpleType>
<xsd:simpleType name="dt_STRINGI">
  <xsd:restriction base="xsd:string"/>
</xsd:simpleType>
<xsd:simpleType name="dt_EDS_Date">
  <xsd:restriction base="xsd:date"/>
</xsd:simpleType>
<xsd:simpleType name="dt_EDS_Time_Of_Day">
  <xsd:restriction base="xsd:time"/>
</xsd:simpleType>
<xsd:simpleType name="dt_EDS_Revision">
  <xsd:restriction base="xsd:string">
    <xsd:pattern value="[0-9]\.[1-9] | [1-9]\.[0-9] | [1-9]\.[1-9]"/>
  </xsd:restriction>
</xsd:simpleType>
<xsd:simpleType name="dt_EDS_URL">
  <xsd:restriction base="xsd:anyURI">
    <xsd:pattern value="http://.*"/>
    <xsd:pattern value="ftp://.*"/>
    <xsd:pattern value=".*"/>
  </xsd:restriction>
</xsd:simpleType>
<xsd:simpleType name="at_AccessType_OptionalGet">
  <xsd:restriction base="xsd:NMTOKEN">
    <xsd:enumeration value="None"/>
    <xsd:enumeration value="Get"/>
  </xsd:restriction>
</xsd:simpleType>
<xsd:simpleType name="at_AccessType_OptionalSet">
  <xsd:restriction base="xsd:NMTOKEN">
    <xsd:enumeration value="None"/>
    <xsd:enumeration value="Get"/>
    <xsd:enumeration value="Set"/>
  </xsd:restriction>
</xsd:simpleType>
<xsd:simpleType name="at_AccessType_Mandatory">
  <xsd:restriction base="xsd:NMTOKEN">
    <xsd:enumeration value="Get"/>
    <xsd:enumeration value="Set"/>
  </xsd:restriction>
</xsd:simpleType>
<xsd:simpleType name="et_VendorSpecificKeyword">

```