

Australian Standard™

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

**Part 2: Reference description for
ISO 11898-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 Standards may also be withdrawn.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 04111.

STANDARDS AUSTRALIA

RECONFIRMATION
OF
AS ISO 15745.2—2004
Industrial automation systems and integration—Open systems application
integration framework
Part 2: Reference description for ISO 11898-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 2: Reference description for
ISO 11898-based control systems**

First published as AS ISO 15745.2—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 5951 3

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 15745-2:2003, *Industrial automation systems and integration—Open systems application integration framework, Part 2: Reference description for ISO 11898-based control systems*.

The objective of this Standard is to provide the specific elements and rules for describing both communication network profiles and the communication related aspects of device profiles specific to ISO 11898-based control systems.

This Standard is Part 2 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 (this Standard)

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

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/IEC		AS/NZS	
62026-3	Low-voltage switchgear and controlgear—Controller-device interfaces (CDIs)—Part 3: DeviceNet	62026.3	Low-voltage switchgear and controlgear—Controller-device interfaces (CDIs)—Part 3: DeviceNet
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.....	iv
1 Scope	1
2 Normative references	1
3 Terms and definitions	2
4 Abbreviated terms.....	2
5 Technology specific elements and rules	3
5.1 Integration models and IAS interfaces	3
5.2 Profile templates	3
5.2.1 General	3
5.2.2 Contents and syntax.....	3
5.2.3 Header.....	3
5.3 Technology specific profiles.....	4
6 Device and communication network profiles for ISO 11898-based control systems	4
6.1 DeviceNet.....	4
6.1.1 Device profile	4
6.1.2 Communication network profile	6
6.2 CANopen.....	7
6.2.1 Device profile	7
6.2.2 Communication network profile	15
Annex A (normative) DeviceNet profile templates.....	17
A.1 General	17
A.2 Device profile template description.....	18
A.2.1 Device profile template description – XML based	18
A.2.2 Device profile template description – XML encapsulation of EDS files.....	35
A.3 Communication network profile template description.....	37
A.3.1 Communication network profile template description – XML based	37
A.3.2 Communication network profile template description – XML encapsulation of EDS files	51
A.4 Electronic Data Sheet (EDS).....	53
A.4.1 Common CIP EDS requirements.....	53
A.4.2 DeviceNet specific EDS requirements.....	85
Annex B (normative) CANopen profile templates.....	98
B.1 Device profile template description.....	98
B.1.1 General	98
B.1.2 Basics	98
B.1.3 DeviceManager object	100
B.1.4 Supplementary element descriptions.....	103
B.1.5 Device profile template XML schemas	105
B.2 Communication network profile template description.....	153
Bibliography.....	163

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 ISO 11898-based control systems (DeviceNet¹, CANopen²).

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) DeviceNetTM is a trade name of 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 DeviceNetTM. Use of the trade name DeviceNetTM requires permission of the Open DeviceNet Vendor Association, Inc.

2) CANopen is a trade name used to describe EN 50325-4. This information is given for the convenience of users of ISO 15745 and does not constitute an endorsement by ISO of the trademark, or any related products. Compliance to this standard does not require use of the trade name CANopen.

AUSTRALIAN STANDARD

Industrial automation systems and integration — Open systems application integration framework —**Part 2:
Reference description for ISO 11898-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 ISO 11898-based control systems.

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

ISO 11898:1993, *Road Vehicles – Interchange of digital information – Controller area network (CAN) for high-speed communication*

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

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 62026-3:2000, *Low-voltage switchgear and controlgear – Controller-device interfaces (CDIs) – Part 3: DeviceNet™*

EN 50325-4 : 2002, *Industrial communications subsystem based on ISO 11898 (CAN) for controller-device interfaces – Part 4 : CANopen*

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.

4 Abbreviated terms

AIF	Application Integration Framework
AIP	Application Interoperability Profile
CAN	Controller Area Network
CIP™ ³	Common Industrial Protocol
EDS	Electronic Data Sheet
IAS	Industrial Automation Systems
OSI	Open System Interconnection
UML	Unified Modelling Language (see UML V1.4)
XML	eXtensible Markup Language (see REC-xml-20001006)

³ 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.

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 ISO 11898-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 ISO 11898-based 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 ISO 11898-based technologies use specific legacy ASCII syntax. Hence, for backward compatibility, template definitions of any technology (Annex A to Annex B) 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 ISO 11898 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
DeviceNet	DeviceNet
CIP	DeviceNet
EDS	DeviceNet
CANopen	CANopen
COFDCML	CANopen

5.3 Technology specific profiles

The technology specific communication network profile structure and communication related aspects of device profile structure based on ISO 11898-based technologies are described in clause 6. The technologies included are:

- DeviceNet (see 6.1);
- CANopen (see 6.2).

The related profile template definitions are specified in Annex A and Annex B.

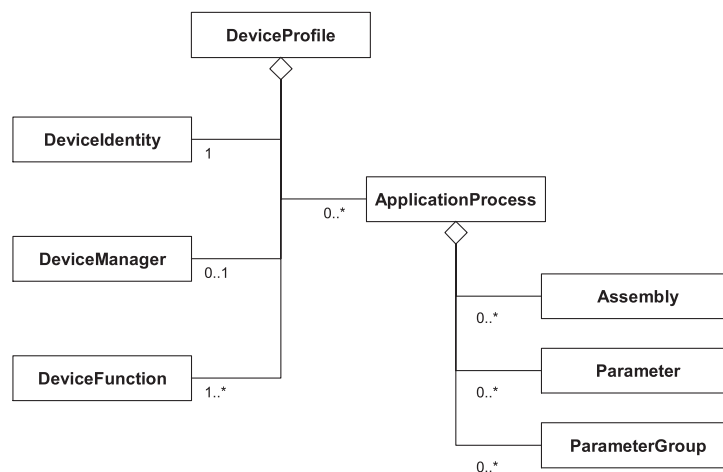
6 Device and communication network profiles for ISO 11898-based control systems

6.1 DeviceNet

6.1.1 Device profile

6.1.1.1 General

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

**Figure 1 — DeviceNet device profile class diagram**

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

The XML schema representing the DeviceNet 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 DeviceNet 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 DeviceNet 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 DeviceNet 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 definition of specific DeviceFunction class is not defined in ISO 15745-2.

6.1.1.5 Application process

Figure 2 shows the class structure of the ApplicationProcess class.

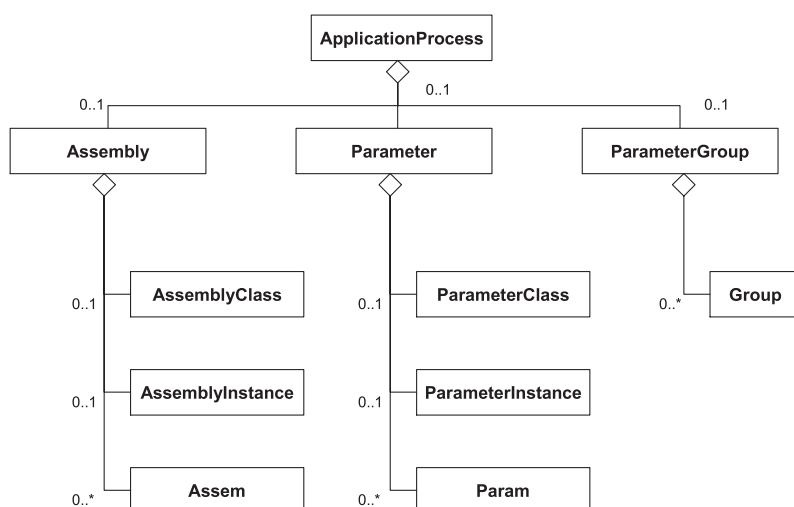


Figure 2 — DeviceNet ApplicationProcess class diagram

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 DeviceNet Assembly object and Parameter objects. 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 DeviceNet communication network profile.

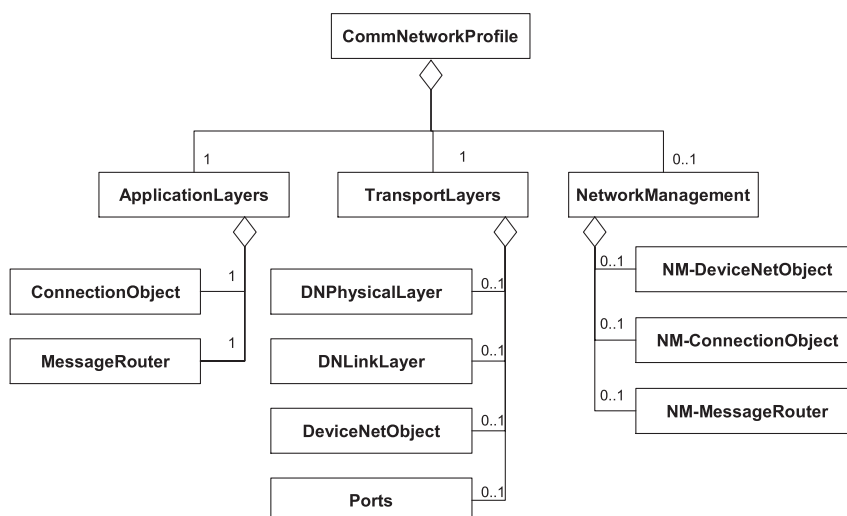


Figure 3 — DeviceNet communication network profile class diagram

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

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

The XML schema representing the encapsulation of a legacy DeviceNet 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 DeviceNet ApplicationLayers class represents the combined profiles for the upper 3 OSI layers of the DeviceNet communication network integration model.

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

- ConnectionObject 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 object and Message Router object are fully specified in IEC 62026-3:2000.

6.1.2.3 Transport layers

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

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

- DNPhysicalLayer identifies the physical layer characteristics (e.g. connectors, baudrates, electrical characteristics);
- DNLinkLayer and DeviceNetObject 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 DeviceNet object is fully specified in IEC 62026-3:2000.

6.1.2.4 Network management

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

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

- NM-DeviceNetObject, NM-ConnectionObject and NM-MessageRouter define the properties associated with class management of the corresponding objects.

6.2 CANopen

6.2.1 Device profile

6.2.1.1 General

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

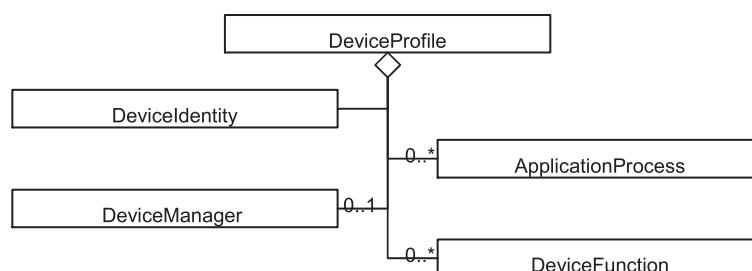


Figure 4 — CANopen device profile class diagram

The required format for CANopen device profiles is described in B.1. The XML schema representing the CANopen device profile template is defined in B.1.5.1. The file name of the XML schema shall be 'COFDCML.xsd'.

NOTE 1 For better readability the CANopen DeviceProfile class diagram has been divided in five class diagrams.

NOTE 2 All these classes are mapped to the same XML schema defined in B.1.5.1.

NOTE 3 The CANopen device profile class diagrams shown in Figure 4 to Figure 10 define the main classes. Some classes are further decomposed; details are defined in Annex B.

6.2.1.2 Device identity

The DeviceIdentity class is defined in Figure 5.

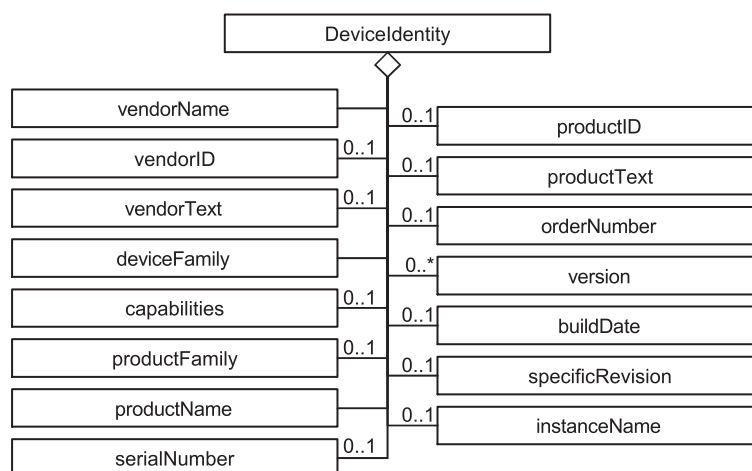


Figure 5 – DeviceIdentity class diagram

The DeviceIdentity class shall consist of the child classes shown in Figure 5 and specified in Table 2.

Table 2 - 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[6])		X	X
vendorText	can be used to provide further information on the vendor	X	X	X
deviceFamily	the definition of this class is is not defined in this standard	X	X	X

Class	Description	Profile	Type	Instance
capabilities	the definition of this class is not defined in this standard		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.				

6.2.1.3 Device manager

6.2.1.3.1 General

Figure 6 shows the CANopen representation of the DeviceManager object.

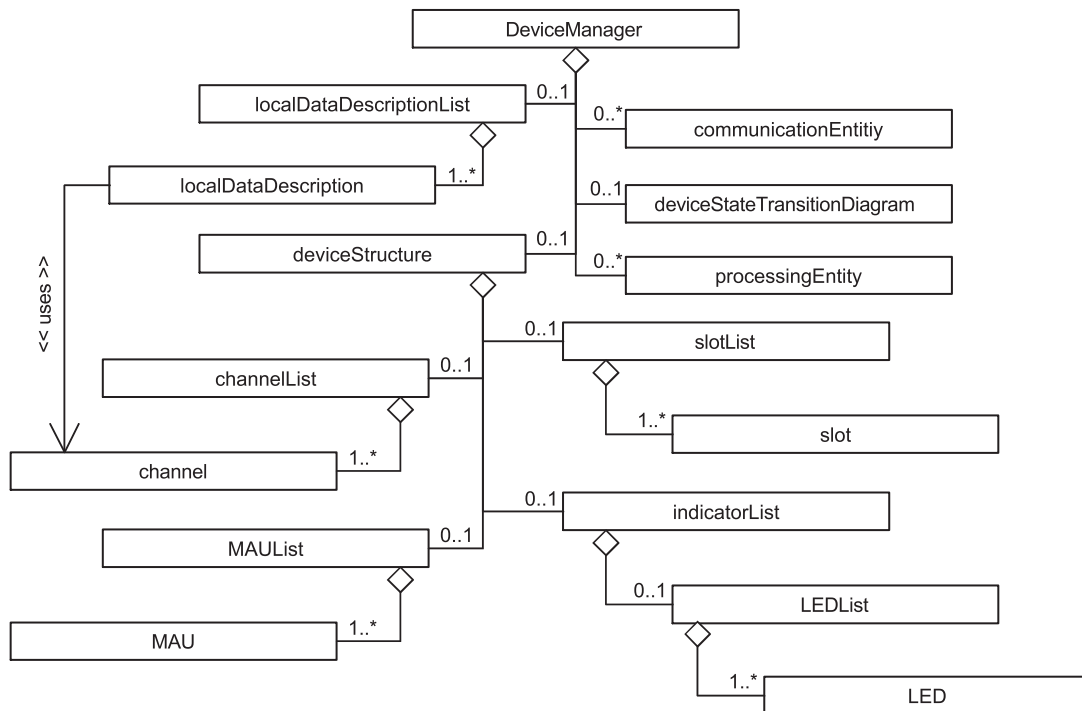


Figure 6 – DeviceManager class diagram

6.2.1.3.2 **localDataDescriptionList, localDataDescription**

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

6.2.1.3.3 **deviceStructure**

6.2.1.3.3.1 **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).

6.2.1.3.3.2 **channelList, channel**

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

6.2.1.3.3.3 **MAUList, MAU**

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

6.2.1.3.3.4 **slotList, slot**

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

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

6.2.1.3.3.5 **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 ISO 15745-2.

6.2.1.3.4 **communicationEntity**

6.2.1.3.4.1 **General**

Figure 7 shows the definition of the communicationEntity class.

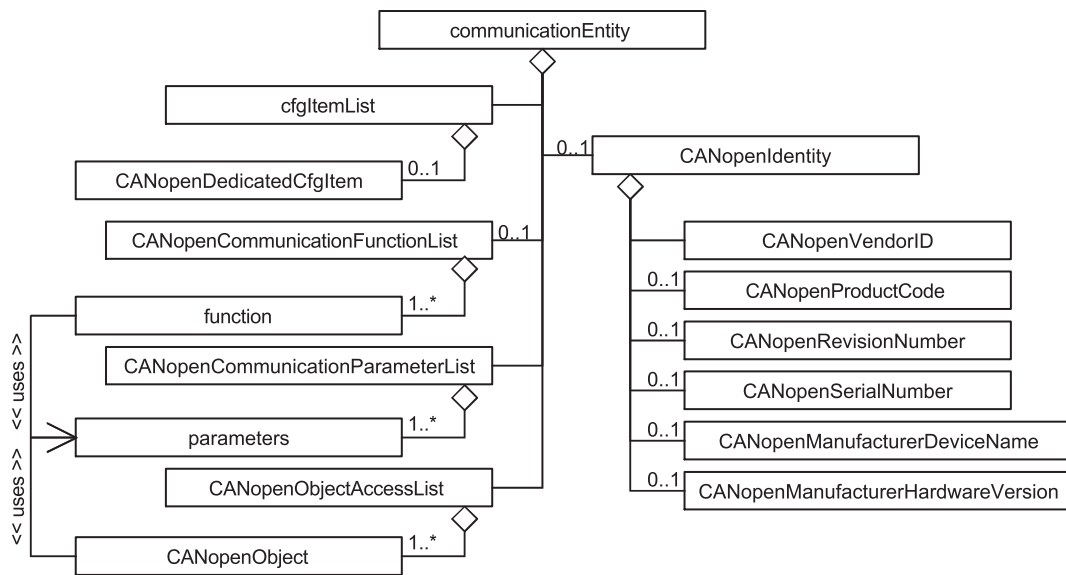


Figure 7 - communicationEntity class diagram

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.

6.2.1.3.4.2 cflgItemList (configuration item list)

The cflgItemList shall consist of a CANopenDedicatedCflgItem object.

6.2.1.3.4.3 CANopenDedicatedCflgItem (CANopen dedicated configuration item)

A CANopenDedicatedCflgItem shall be a collection of configuration items.

NOTE The definition of additional configuration item classes is outside the scope of ISO 15745-2.

6.2.1.3.4.4 CANopenIdentity

The CANopenIdentity class consists of several object that are required to identify a device within a CANopen network. This includes objects for a CANopenVendorID, CANopenProductCode, CANopenRevisionNumber, and CANopenSerialNumber as well as CANopenManufacturerDeviceName, CANopenManufacturerHardwareVersion, and CANopenManufacturerSoftwareVersion.

NOTE The corresponding identity object is specified in EN 50325-4.

6.2.1.3.4.5 CANopenCommunicationFunctionList, function

The CANopenCommunicationFunctionList shall be a collection of function objects. Each function object describes a CANopen functionality from the CANopen communication area by use of the CANopenCommunicationParameterList.

6.2.1.3.4.6 CANopenCommunicationParameterList, parameters

The CANopenCommunicationParameterList shall be a collection of parameter objects. Each parameter object describes a parameter from the CANopen communication area.

6.2.1.3.4.7 CANopenObjectAccessList, CANopenObject

The CANopenObjectAccessList shall be a collection of CANopenObject objects. Each CANopenObject object describes a parameter from the DeviceFunction view or from the CANopenCommunicationParameterList in respect of the CANopen object dictionary.

NOTE The CANopenObjectAccessList is corresponding to the CANopen object dictionary in EN 50325-4.

6.2.1.3.5 processingEntity

6.2.1.3.5.1 General

Figure 8 shows the definition of the processingEntity class.

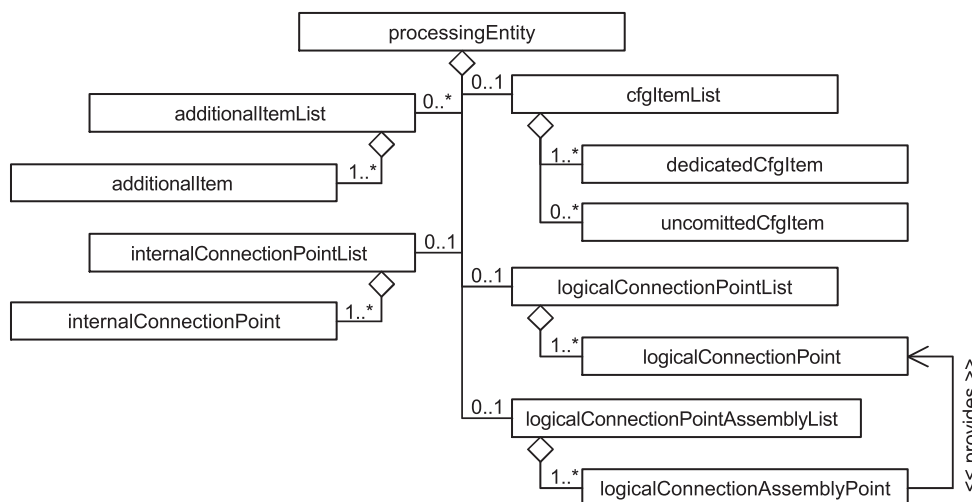


Figure 8 - processingEntity class diagram

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

EXAMPLE A resource capable of executing programs.

6.2.1.3.5.2 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.

6.2.1.3.5.3 logicalConnectionPointList, logicalConnectionPoint

The logicalConnectionPointList shall be a collection of logicalConnectionPoint objects. A logicalConnectionPoint describes a connection.

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

6.2.1.3.5.4 logicalConnectionPointAssemblyList, logicalConnectionPointAssembly

The logicalConnectionPointAssemblyList shall be a collection of logicalConnectionPointAssembly objects. A logicalConnectionPointAssembly shall be a description of a group of logicalConnectionPoint objects.

6.2.1.3.5.5 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.

6.2.1.3.5.6 cfgItemList (configuration item list)

The cfgItemList may consist of dedicatedCfgItem objects and uncommittedCfgItem objects.

6.2.1.3.5.7 dedicatedCfgItem (dedicated configuration item)

A dedicatedCfgItem shall be a configuration Item with a dedicatedCfgItemType. A dedicatedCfgItem shall be used to specify the corresponding configuration properties.

6.2.1.3.5.8 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 ISO 15745-2.

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

6.2.1.4 Device function

6.2.1.4.1 General

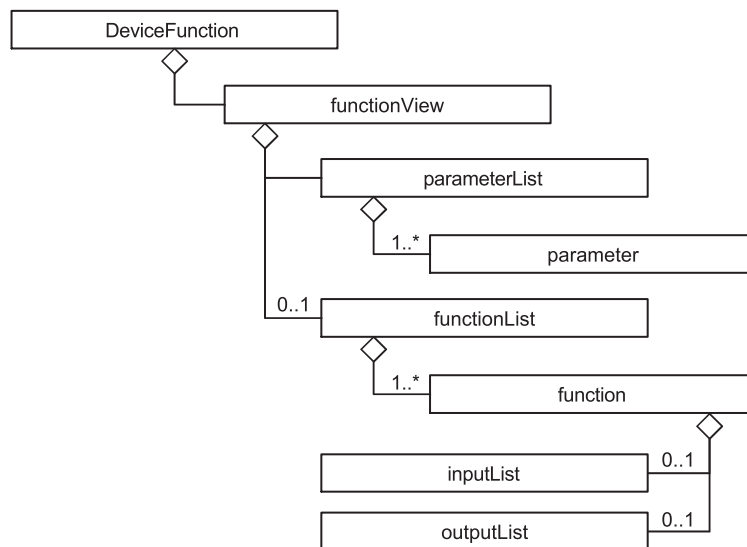


Figure 9 - DeviceFunction class diagram

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 Annex B.

NOTE The definition of additional XML schemas describing the DeviceFunction classes is outside the scope of ISO 15745-2.

6.2.1.4.2 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.

6.2.1.4.3 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.

6.2.1.5 Application process

6.2.1.5.1 General

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

NOTE The definition of these XML schemas are not defined in ISO 15745-2.

6.2.1.5.2 textualDescription

The textualDescription object explains the function of a device in a human readable textual form.

6.2.2 Communication network profile

6.2.2.1 General

Figure 10 shows the class structure of the CANopen communication network profile.

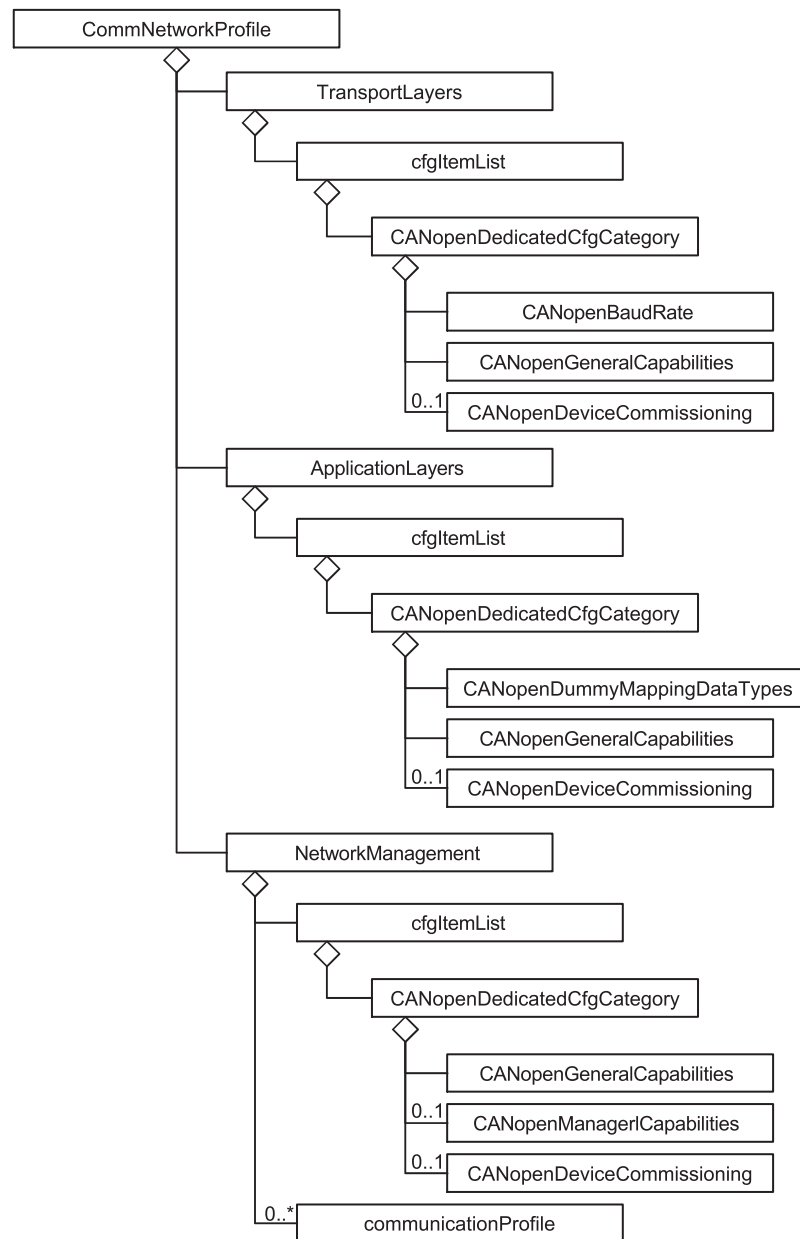


Figure 10 - CANopen communication network profile class diagram

The XML schema representing the CANopen communication network profile is defined in Annex B. The file name of the XML schema shall be "COCommNetworkProfile.xsd".

6.2.2.2 communicationProfile

The communicationProfile shall state the usable communication profile identifiers. Communication profiles and their identifiers are defined in IEC 61784-1:2003 clause 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.

6.2.2.3 Transport layers

6.2.2.3.1 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 contain a cfgItem object.

6.2.2.3.2 cfgItemList, CANopenDedicatedCfgCategory

The cfgItemList shall be a collection of configuration items related to the lower 4 OSI layers of the communication network integration model. This includes a specific category dedicated to CANopen related configuration items, e.g. Baudrates, general capabilities as well as device commissioning.

NOTE This CANopenDedicatedCfgCategory describes the supported services for data transmission (e.g. process data objects and service data objects) as well as the supported baudrates as defined in EN 50325-4.

6.2.2.4 Application layers

6.2.2.4.1 General

An ApplicationLayers object shall represent the combined profiles for the upper 3 OSI layers of the communication network integration model. The ApplicationLayers object shall contain a cfgItem object.

6.2.2.4.2 cfgItemList, CANopenDedicatedCfgCategory

The cfgItemList shall be a collection of configuration items related to the upper 3 OSI layers of the communication network integration model. This includes a specific category dedicated to CANopen related configuration items, e.g. supported data types for dummy mapping, general capabilities as well as device commissioning.

6.2.2.5 Network management

6.2.2.5.1 General

A NetworkManagement object shall represent network management functionality. The NetworkManagement object shall contain a cfgItem object.

6.2.2.5.2 cfgItemList, CANopenDedicatedCfgCategory

The cfgItemList shall be a collection of configuration items related to the network management. This includes a specific category dedicated to CANopen related configuration items, e.g. CANopen manager capabilities, general capabilities as well as device commissioning.

NOTE This CANopenDedicatedCfgCategory specifies the supported network management services as specified in EN 50325-4.

Annex A (normative)

DeviceNet profile templates

A.1 General

The upper layers of the DeviceNet 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 DeviceNet 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 62026-3:2000), 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 DeviceNet 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, 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 62026-3:2000), 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:simpleType name="IASInterface_DataType">
    <xsd:union>
      <xsd:simpleType>
        <xsd:restriction base="xsd:string">
          <xsd:enumeration value="CSI" />
        </xsd:restriction>
      </xsd:simpleType>
    </xsd:union>
  </xsd:simpleType>
</xsd:schema>
```

```

        <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">
  <xsd:restriction base="xsd:string">
    <xsd:pattern value="[1-9][0-9]{0,4}_([A-Z]|[a-z]|[0-9])([A-Z]|[a-z]|[0-9]|_)*"/>
  </xsd:restriction>
</xsd:simpleType>
<xsd:simpleType name="et_EPATH_item">
  <xsd:union>
    <xsd:simpleType>
      <xsd:restriction base="xsd:string">

```

```

        <xsd:pattern value="([0-9]|[a-f]|[A-F]){2}" />
    </xsd:restriction>
</xsd:simpleType>
<xsd:simpleType>
    <xsd:restriction base="xsd:NMTOKEN">
        <xsd:enumeration value="SLOT" />
        <xsd:enumeration value="SLOT_MINUS_ONE" />
        <xsd:enumeration value="SYMBOL_ANSI" />
    </xsd:restriction>
</xsd:simpleType>
<xsd:simpleType>
    <xsd:restriction base="xsd:string">
        <xsd:pattern value="Param[1-9][0-9]{0,4}" />
        <xsd:pattern value="\[Param[1-9][0-9]{0,4}\]" />
        <xsd:pattern value="ProxyParam[1-9][0-9]{0,4}" />
        <xsd:pattern value="\[ProxyParam[1-9][0-9]{0,4}\]" />
    </xsd:restriction>
</xsd:simpleType>
</xsd:union>
</xsd:simpleType>
<xsd:simpleType name="et_ParamReference">
    <xsd:restriction base="xsd:NMTOKEN">
        <xsd:pattern value="Param[1-9][0-9]{0,4}([:][0-9]{1,2})*" />
        <xsd:pattern value="ProxyParam[1-9][0-9]{0,4}([:][0-9]{1,2})*" />
    </xsd:restriction>
</xsd:simpleType>
<xsd:simpleType name="et_AssemReference">
    <xsd:restriction base="xsd:NMTOKEN">
        <xsd:pattern value="Assem[1-9][0-9]{0,4}" />
        <xsd:pattern value="ProxyAssem[1-9][0-9]{0,4}" />
    </xsd:restriction>
</xsd:simpleType>
<xsd:attributeGroup name="ag_FileDescription">
    <xsd:attribute name="DescText" type="dt_EDS_Char_Array" use="required" />
    <xsd:attribute name="CreateDate" type="dt_EDS_Date" use="required" />
    <xsd:attribute name="CreateTime" type="dt_EDS_Time_Of_Day" use="required" />
    <xsd:attribute name="ModDate" type="dt_EDS_Date" use="optional" />
    <xsd:attribute name="ModTime" type="dt_EDS_Time_Of_Day" use="optional" />
    <xsd:attribute name="Revision" type="dt_EDS_Revision" use="required" />
    <xsd:attribute name="HomeURL" type="dt_EDS_URL" use="optional" />
    <xsd:attribute name="SpecificationConformance" type="dt_EDS_Char_Array" use="required" />
</xsd:attributeGroup>
</xsd:schema>

```

A.2.1.3.3 CIP_Device_Profile.xsd

NOTE This XML schema includes the files "MasterTemplateTypes.xsd" (see A.2.1.3.1) and "CIPDataTypes.xsd" (see A.2.1.3.2).

```

<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">
    <!-- Target namespaces are not specified in this master template -->
    <xsd:redefine schemaLocation="MasterTemplateTypes.xsd">
        <xsd:complexType name="ISO15745Reference_DataType">
            <xsd:complexContent>
                <xsd:restriction base="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" fixed="CIP" />
                    </xsd:sequence>
                </xsd:restriction>
            </xsd:complexContent>
        </xsd:complexType>
    </xsd:redefine>
    <xsd:include schemaLocation="CIPDataTypes.xsd" />
    <xsd:element name="ISO15745Profile">
        <xsd:complexType>
            <xsd:sequence>
                <xsd:element ref="ProfileHeader" />
                <xsd:element ref="ProfileBody" />
            </xsd:sequence>
        </xsd:complexType>
    </xsd:element>

```

```

</xsd:element>
<xsd:annotation>
  <xsd:documentation>* HEADER SECTION *</xsd:documentation>
</xsd:annotation>
<xsd:element name="ProfileHeader">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="ProfileIdentification" type="xsd:string"/>
      <xsd:element name="ProfileRevision" type="xsd:string"/>
      <xsd:element name="ProfileName" type="xsd:string"/>
      <xsd:element name="ProfileSource" type="xsd:string"/>
      <xsd:element name="ProfileClassID" type="ProfileClassID_DataType" fixed="Device"/>
      <xsd:element name="ProfileDate" type="xsd:date" minOccurs="0"/>
      <xsd:element name="AdditionalInformation" type="xsd:anyURI" minOccurs="0"/>
      <xsd:element name="ISO15745Reference" type="ISO15745Reference_DataType"/>
      <xsd:element name="IASInterfaceType" type="IASInterface_DataType" minOccurs="0"
maxOccurs="unbounded"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
<xsd:annotation>
  <xsd:documentation>* BODY SECTION *</xsd:documentation>
</xsd:annotation>
<xsd:element name="ProfileBody">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element ref="DeviceIdentity"/>
      <xsd:element ref="DeviceManager" minOccurs="0"/>
      <xsd:element ref="DeviceFunction" maxOccurs="unbounded"/>
      <xsd:element ref="ApplicationProcess" minOccurs="0"/>
      <xsd:element name="ExternalProfileHandle" type="ProfileHandle_DataType" minOccurs="0"/>
    </xsd:sequence>
    <xsd:attributeGroup ref="ag_FileDescription"/>
  </xsd:complexType>
</xsd:element>
<xsd:element name="DeviceIdentity">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="DeviceIdentity_InstanceAttributes">
        <xsd:complexType>
          <xsd:sequence>
            <xsd:element name="VendCode">
              <xsd:complexType>
                <xsd:simpleContent>
                  <xsd:extension base="dt_UINT">
                    <xsd:attribute name="Access_Rule" type="at_AccessType_Mandatory"
use="required" fixed="Get"/>
                  </xsd:extension>
                </xsd:simpleContent>
              </xsd:complexType>
            </xsd:element>
            <xsd:element name="VendName">
              <xsd:complexType>
                <xsd:simpleContent>
                  <xsd:extension base="dt_EDS_Char_Array"/>
                </xsd:simpleContent>
              </xsd:complexType>
            </xsd:element>
            <xsd:element name="SpecificationConformance" type="dt_EDS_Char_Array"
minOccurs="0"/>
          </xsd:sequence>
        </xsd:complexType>
      </xsd:element>
      <xsd:element name="ProdType">
        <xsd:complexType>
          <xsd:simpleContent>
            <xsd:extension base="dt_UINT">
              <xsd:attribute name="Access_Rule" type="at_AccessType_Mandatory"
use="required" fixed="Get"/>
            </xsd:extension>
          </xsd:simpleContent>
        </xsd:complexType>
      </xsd:element>
      <xsd:element name="ProdTypeStr">
        <xsd:complexType>
          <xsd:simpleContent>
            <xsd:extension base="dt_EDS_Char_Array"/>
          </xsd:simpleContent>
        </xsd:complexType>
      </xsd:element>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>

```

```

</xsd:element>
<xsd:element name="ProdCode">
  <xsd:complexType>
    <xsd:simpleContent>
      <xsd:extension base="dt_UINT">
        <xsd:attribute name="Access_Rule" type="at_AccessType_Mandatory"
use="required" fixed="Get"/>
      </xsd:extension>
    </xsd:simpleContent>
  </xsd:complexType>
</xsd:element>
<xsd:element name="ProdRevision">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="MajRev" type="dt_USINT"/>
      <xsd:element name="MinRev" type="dt_USINT"/>
    </xsd:sequence>
    <xsd:attribute name="Access_Rule" type="at_AccessType_Mandatory"
use="required" fixed="Get"/>
  </xsd:complexType>
</xsd:element>
<xsd:element name="Status" minOccurs="0">
  <xsd:complexType>
    <xsd:attribute name="Access_Rule" type="at_AccessType_Mandatory"
use="required" fixed="Get"/>
  </xsd:complexType>
</xsd:element>
<xsd:element name="SerialNumber" minOccurs="0">
  <xsd:complexType>
    <xsd:attribute name="Access_Rule" type="at_AccessType_Mandatory"
use="required" fixed="Get"/>
  </xsd:complexType>
</xsd:element>
<xsd:element name="ProdName">
  <xsd:complexType>
    <xsd:simpleContent>
      <xsd:extension base="xsd:string">
        <xsd:attribute name="Access_Rule" type="at_AccessType_Mandatory"
use="required" fixed="Get"/>
      </xsd:extension>
    </xsd:simpleContent>
  </xsd:complexType>
</xsd:element>
<xsd:element name="State" minOccurs="0">
  <xsd:complexType>
    <xsd:attribute name="Access_Rule" type="at_AccessType_OptionalGet"
use="required"/>
  </xsd:complexType>
</xsd:element>
<xsd:element name="ConfigurationConsistencyValue" minOccurs="0">
  <xsd:complexType>
    <xsd:attribute name="Access_Rule" type="at_AccessType_OptionalGet"
use="required"/>
  </xsd:complexType>
</xsd:element>
<xsd:element name="HeartbeatInterval" minOccurs="0">
  <xsd:complexType>
    <xsd:attribute name="Access_Rule" type="at_AccessType_OptionalSet"
use="required"/>
  </xsd:complexType>
</xsd:element>
<xsd:element name="Catalog" type="xsd:string" minOccurs="0"/>
<xsd:element name="Icon" type="xsd:string" minOccurs="0"/>
<xsd:element name="ExcludeFromAdapterRackConnection" type="xsd:string"
minOccurs="0"/>
<xsd:element name="DeviceClassification" minOccurs="0">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="Class" maxOccurs="unbounded">
        <xsd:complexType>
          <xsd:sequence>
            <xsd:element name="MainClass">
              <xsd:simpleType>
                <xsd:union>
                  <xsd:simpleType>
                    <xsd:restriction base="xsd:NMTOKEN">

```

```

        <xsd:enumeration value="ControlNet"/>
        <xsd:enumeration value="DeviceNet"/>
        <xsd:enumeration value="EtherNetIP"/>
    </xsd:restriction>
</xsd:simpleType>
<xsd:simpleType>
    <xsd:restriction base="et_VendorSpecificKeyword"/>
</xsd:simpleType>
</xsd:union>
</xsd:simpleType>
</xsd:element>
<xsd:element name="SubClass" type="xsd:NMTOKEN" minOccurs="0"
maxOccurs="unbounded"/>

</xsd:sequence>
<xsd:attribute name="id" use="required">
    <xsd:simpleType>
        <xsd:restriction base="xsd:ID">
            <xsd:pattern value="Class[1-9][0-9]{0,4}"/>
        </xsd:restriction>
    </xsd:simpleType>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:any namespace="##any"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="DeviceIdentity_InstanceOperations" minOccurs="0">
    <xsd:complexType>
        <xsd:sequence>
            <xsd:element name="Get_Attribute_All">
                <xsd:complexType>
                    <xsd:attribute ref="SupportedService" fixed="true"/>
                </xsd:complexType>
            </xsd:element>
            <xsd:element name="Reset">
                <xsd:complexType>
                    <xsd:attribute ref="SupportedService" fixed="true"/>
                </xsd:complexType>
            </xsd:element>
            <xsd:element name="Get_Attribute_Single">
                <xsd:complexType>
                    <xsd:attribute ref="SupportedService"/>
                </xsd:complexType>
            </xsd:element>
            <xsd:any namespace="##any"/>
        </xsd:sequence>
    </xsd:complexType>
</xsd:element>
<xsd:any namespace="##any" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="DeviceManager">
    <xsd:complexType>
        <xsd:sequence>
            <xsd:element name="DeviceIdentity_ClassAttributes" minOccurs="0">
                <xsd:complexType>
                    <xsd:sequence>
                        <xsd:element name="ObjectRevision">
                            <xsd:complexType>
                                <xsd:attribute name="Access_Rule" type="at_AccessType_OptionalGet"
use="required"/>
                            </xsd:complexType>
                        </xsd:element>
                        <xsd:element name="MaxInstance">
                            <xsd:complexType>
                                <xsd:attribute name="Access_Rule" type="at_AccessType_OptionalGet"
use="required"/>
                            </xsd:complexType>
                        </xsd:element>
                        <xsd:element name="MaxIDClassAttributes">
                            <xsd:complexType>

```

```

        <xsd:simpleContent>
          <xsd:extension base="xsd:string">
            <xsd:attribute name="Access_Rule" type="at_AccessType_OptionalGet"
use="required"/>
          </xsd:extension>
        </xsd:simpleContent>
      </xsd:complexType>
    </xsd:element>
    <xsd:element name="MaxIDInstanceAttributes">
      <xsd:complexType>
        <xsd:simpleContent>
          <xsd:extension base="xsd:string">
            <xsd:attribute name="Access_Rule" type="at_AccessType_OptionalGet"
use="required"/>
          </xsd:extension>
        </xsd:simpleContent>
      </xsd:complexType>
    </xsd:element>
    <xsd:any namespace="##any" minOccurs="0" maxOccurs="unbounded"/>
  </xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="DeviceIdentity_ClassOperations" minOccurs="0">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="Get_Attribute_All">
        <xsd:complexType>
          <xsd:attribute ref="SupportedService"/>
        </xsd:complexType>
      </xsd:element>
      <xsd:element name="Reset">
        <xsd:complexType>
          <xsd:attribute ref="SupportedService"/>
        </xsd:complexType>
      </xsd:element>
      <xsd:element name="Get_Attribute_Single">
        <xsd:complexType>
          <xsd:attribute ref="SupportedService"/>
        </xsd:complexType>
      </xsd:element>
      <xsd:element name="Find_Next_Object_Instance">
        <xsd:complexType>
          <xsd:attribute ref="SupportedService"/>
        </xsd:complexType>
      </xsd:element>
      <xsd:any namespace="##any" minOccurs="0" maxOccurs="unbounded"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
<xsd:element name="Modular" minOccurs="0">
  <xsd:complexType>
    <xsd:choice>
      <xsd:element name="Chassis">
        <xsd:complexType>
          <xsd:sequence>
            <xsd:element name="DefineSlotsInRack" type="dt_UINT"/>
            <xsd:element name="SlotDisplayRule" type="et_ParamReference"
minOccurs="0"/>
            <xsd:any namespace="##any" minOccurs="0" maxOccurs="unbounded"/>
          </xsd:sequence>
        </xsd:complexType>
      </xsd:element>
      <xsd:element name="Module">
        <xsd:complexType>
          <xsd:sequence>
            <xsd:element name="Width" type="dt_UINT"/>
            <xsd:element name="Rack" maxOccurs="unbounded">
              <xsd:complexType>
                <xsd:sequence>
                  <xsd:element name="VendCode" type="dt_UINT"/>
                  <xsd:element name="ProdType" type="dt_UINT"/>
                  <xsd:element name="ProdCode" type="dt_UINT"/>
                  <xsd:element name="MajRev" type="dt_USINT"/>
                  <xsd:element name="MinRev" type="dt_USINT"/>
                  <xsd:element name="LegalSlot" type="dt_UINT"
maxOccurs="unbounded"/>
                </xsd:sequence>
              </xsd:complexType>
            </xsd:element>
          </xsd:sequence>
        </xsd:complexType>
      </xsd:element>
    </xsd:choice>
  </xsd:complexType>
</xsd:element>

```

```

        <xsd:any namespace="##any" minOccurs="0" maxOccurs="unbounded"/>
      </xsd:sequence>
      <xsd:attribute name="id" use="required">
        <xsd:simpleType>
          <xsd:restriction base="xsd:ID">
            <xsd:pattern value="Rack[1-9][0-9]{0,4}"/>
          </xsd:restriction>
        </xsd:simpleType>
      </xsd:attribute>
    </xsd:complexType>
  </xsd:element>
  <xsd:element name="ExternalID" type="dt_EPATH" minOccurs="0"/>
  <xsd:element name="GenericID" type="dt_EPATH" minOccurs="0"/>
  <xsd:element name="ExternIDExactMatch" minOccurs="0">
    <xsd:simpleType>
      <xsd:restriction base="xsd:NMTOKEN">
        <xsd:enumeration value="Yes"/>
        <xsd:enumeration value="No"/>
      </xsd:restriction>
    </xsd:simpleType>
  </xsd:element>
  <xsd:element name="Query" minOccurs="0">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="Path" type="dt_EPATH"/>
        <xsd:element name="Service" type="dt_USINT"/>
        <xsd:element name="Size">
          <xsd:simpleType>
            <xsd:restriction base="dt_USINT">
              <xsd:minInclusive value="1"/>
              <xsd:maxInclusive value="16"/>
            </xsd:restriction>
          </xsd:simpleType>
        </xsd:element>
        <xsd:element name="ExternalID" type="dt_EPATH"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>
  <xsd:any namespace="##any" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
</xsd:choice>
</xsd:complexType>
</xsd:element>
<xsd:any namespace="##any"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="DeviceFunction">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:any namespace="##any" minOccurs="0" maxOccurs="unbounded"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
<xsd:element name="ApplicationProcess">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="Parameter" minOccurs="0">
        <xsd:complexType>
          <xsd:sequence>
            <xsd:element name="Parameter_Class" minOccurs="0">
              <xsd:complexType>
                <xsd:sequence>
                  <xsd:element name="Parameter_ClassAttributes">
                    <xsd:complexType>
                      <xsd:sequence>
                        <xsd:element name="ObjectRevision" minOccurs="0">
                          <xsd:complexType>
                            <xsd:attribute name="Access_Rule"
type="at_AccessType_OptionalGet" use="required"/>
                          </xsd:complexType>
                        </xsd:element>
                        <xsd:element name="MaxInstance">
                          <xsd:complexType>

```

```

        <xsd:attribute name="Access_Rule"
type="at_AccessType_Mandatory" use="required" fixed="Get"/>
    </xsd:complexType>
</xsd:element>
<xsd:element name="ParameterClassDescriptor">
    <xsd:complexType>
        <xsd:simpleContent>
            <xsd:extension base="xsd:string">
                <xsd:attribute name="Access_Rule"
type="at_AccessType_Mandatory" use="required" fixed="Get"/>
            </xsd:extension>
        </xsd:simpleContent>
    </xsd:complexType>
</xsd:element>
<xsd:element name="ConfigurationAssemblyInstance">
    <xsd:complexType>
        <xsd:simpleContent>
            <xsd:extension base="xsd:string">
                <xsd:attribute name="Access_Rule"
type="at_AccessType_Mandatory" use="required" fixed="Get"/>
            </xsd:extension>
        </xsd:simpleContent>
    </xsd:complexType>
</xsd:element>
<xsd:element name="NativeLanguage" minOccurs="0">
    <xsd:complexType>
        <xsd:simpleContent>
            <xsd:extension base="xsd:string">
                <xsd:attribute name="Access_Rule"
type="at_AccessType_OptionalSet" use="required"/>
            </xsd:extension>
        </xsd:simpleContent>
    </xsd:complexType>
</xsd:element>
<xsd:any namespace="##any" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="Parameter_ClassOperations" minOccurs="0">
    <xsd:complexType>
        <xsd:sequence>
            <xsd:element name="Get_Attribute_All">
                <xsd:complexType>
                    <xsd:attribute ref="SupportedService"/>
                </xsd:complexType>
            </xsd:element>
            <xsd:element name="Reset">
                <xsd:complexType>
                    <xsd:attribute ref="SupportedService"/>
                </xsd:complexType>
            </xsd:element>
            <xsd:element name="Get_Attribute_Single">
                <xsd:complexType>
                    <xsd:attribute ref="SupportedService" fixed="true"/>
                </xsd:complexType>
            </xsd:element>
            <xsd:element name="Set_Attribute_Single">
                <xsd:complexType>
                    <xsd:attribute ref="SupportedService"/>
                </xsd:complexType>
            </xsd:element>
            <xsd:element name="Restore">
                <xsd:complexType>
                    <xsd:attribute ref="SupportedService"/>
                </xsd:complexType>
            </xsd:element>
            <xsd:element name="Save">
                <xsd:complexType>
                    <xsd:attribute ref="SupportedService"/>
                </xsd:complexType>
            </xsd:element>
            <xsd:any namespace="##any" minOccurs="0" maxOccurs="unbounded"/>
        </xsd:sequence>
    </xsd:complexType>
</xsd:element>
</xsd:sequence>

```

```

    </xsd:complexType>
  </xsd:element>
  <xsd:element name="Parameter_Instance" minOccurs="0">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="Parameter_InstanceAttributes" minOccurs="0">
          <xsd:complexType>
            <xsd:sequence>
              <xsd:element name="ParameterValue">
                <xsd:complexType>
                  <xsd:attribute name="Access_Rule"
type="at_AccessType_Mandatory" use="required" fixed="Set"/>
                </xsd:complexType>
              </xsd:element>
              <xsd:element name="LinkPathSize">
                <xsd:complexType>
                  <xsd:attribute name="Access_Rule"
type="at_AccessType_Mandatory" use="required" fixed="Set"/>
                </xsd:complexType>
              </xsd:element>
              <xsd:element name="LinkPath">
                <xsd:complexType>
                  <xsd:attribute name="Access_Rule"
type="at_AccessType_Mandatory" use="required" fixed="Set"/>
                </xsd:complexType>
              </xsd:element>
              <xsd:element name="ParamDescriptor">
                <xsd:complexType>
                  <xsd:attribute name="Access_Rule"
type="at_AccessType_Mandatory" use="required" fixed="Get"/>
                </xsd:complexType>
              </xsd:element>
              <xsd:element name="DataType">
                <xsd:complexType>
                  <xsd:attribute name="Access_Rule"
type="at_AccessType_Mandatory" use="required" fixed="Get"/>
                </xsd:complexType>
              </xsd:element>
              <xsd:element name="DataSize">
                <xsd:complexType>
                  <xsd:attribute name="Access_Rule"
type="at_AccessType_Mandatory" use="required" fixed="Get"/>
                </xsd:complexType>
              </xsd:element>
              <xsd:element name="ParameterName">
                <xsd:complexType>
                  <xsd:attribute name="Access_Rule"
type="at_AccessType_OptionalGet" use="required"/>
                </xsd:complexType>
              </xsd:element>
              <xsd:element name="UnitsString">
                <xsd:complexType>
                  <xsd:attribute name="Access_Rule"
type="at_AccessType_OptionalGet" use="required"/>
                </xsd:complexType>
              </xsd:element>
              <xsd:element name="HelpString">
                <xsd:complexType>
                  <xsd:attribute name="Access_Rule"
type="at_AccessType_OptionalGet" use="required"/>
                </xsd:complexType>
              </xsd:element>
              <xsd:element name="MinimumValue">
                <xsd:complexType>
                  <xsd:attribute name="Access_Rule"
type="at_AccessType_OptionalGet" use="required"/>
                </xsd:complexType>
              </xsd:element>
              <xsd:element name="MaximumValue">
                <xsd:complexType>
                  <xsd:attribute name="Access_Rule"
type="at_AccessType_OptionalGet" use="required"/>
                </xsd:complexType>
              </xsd:element>
              <xsd:element name="DefaultValue">
                <xsd:complexType>

```

```

                                <xsd:attribute name="Access_Rule"
type="at_AccessType_OptionalGet" use="required"/>
                                </xsd:complexType>
                                </xsd:element>
                                <xsd:element name="ScalingMultiplier">
                                <xsd:complexType>
                                <xsd:attribute name="Access_Rule"
type="at_AccessType_OptionalGet" use="required"/>
                                </xsd:complexType>
                                </xsd:element>
                                <xsd:element name="ScalingDivider">
                                <xsd:complexType>
                                <xsd:attribute name="Access_Rule"
type="at_AccessType_OptionalGet" use="required"/>
                                </xsd:complexType>
                                </xsd:element>
                                <xsd:element name="ScalingBase">
                                <xsd:complexType>
                                <xsd:attribute name="Access_Rule"
type="at_AccessType_OptionalGet" use="required"/>
                                </xsd:complexType>
                                </xsd:element>
                                <xsd:element name="ScalingOffset">
                                <xsd:complexType>
                                <xsd:attribute name="Access_Rule"
type="at_AccessType_OptionalGet" use="required"/>
                                </xsd:complexType>
                                </xsd:element>
                                <xsd:element name="MultiplierLink">
                                <xsd:complexType>
                                <xsd:attribute name="Access_Rule"
type="at_AccessType_OptionalGet" use="required"/>
                                </xsd:complexType>
                                </xsd:element>
                                <xsd:element name="DivisorLink">
                                <xsd:complexType>
                                <xsd:attribute name="Access_Rule"
type="at_AccessType_OptionalGet" use="required"/>
                                </xsd:complexType>
                                </xsd:element>
                                <xsd:element name="BaseLink">
                                <xsd:complexType>
                                <xsd:attribute name="Access_Rule"
type="at_AccessType_OptionalGet" use="required"/>
                                </xsd:complexType>
                                </xsd:element>
                                <xsd:element name="OffsetLink">
                                <xsd:complexType>
                                <xsd:attribute name="Access_Rule"
type="at_AccessType_OptionalGet" use="required"/>
                                </xsd:complexType>
                                </xsd:element>
                                <xsd:element name="DecimalPrecision">
                                <xsd:complexType>
                                <xsd:attribute name="Access_Rule"
type="at_AccessType_OptionalGet" use="required"/>
                                </xsd:complexType>
                                </xsd:element>
                                <xsd:any namespace="##any" minOccurs="0" maxOccurs="unbounded"/>
                                </xsd:sequence>
                                </xsd:complexType>
                                </xsd:element>
                                <xsd:element name="Parameter_InstanceOperations" minOccurs="0">
                                <xsd:complexType>
                                <xsd:sequence>
                                <xsd:element name="Get_Attribute_All">
                                <xsd:complexType>
                                <xsd:attribute ref="SupportedService"/>
                                </xsd:complexType>
                                </xsd:element>
                                <xsd:element name="Get_Attribute_Single">
                                <xsd:complexType>
                                <xsd:attribute ref="SupportedService" fixed="true"/>
                                </xsd:complexType>
                                </xsd:element>
                                <xsd:element name="Set_Attribute_Single">

```

```

        <xsd:complexType>
          <xsd:attribute ref="SupportedService" fixed="true"/>
        </xsd:complexType>
      </xsd:element>
      <xsd:element name="Get_Enum_String">
        <xsd:complexType>
          <xsd:attribute ref="SupportedService"/>
        </xsd:complexType>
      </xsd:element>
      <xsd:any namespace="##any" minOccurs="0" maxOccurs="unbounded"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="Param" minOccurs="0" maxOccurs="unbounded">
  <xsd:complexType>
    <xsd:complexContent>
      <xsd:extension base="et_ParamType">
        <xsd:attribute name="id" use="required">
          <xsd:simpleType>
            <xsd:restriction base="xsd:ID">
              <xsd:pattern value="Param[1-9][0-9]{0,4}"/>
            </xsd:restriction>
          </xsd:simpleType>
        </xsd:attribute>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
</xsd:element>
<xsd:element name="ProxyParam" minOccurs="0" maxOccurs="unbounded">
  <xsd:complexType>
    <xsd:complexContent>
      <xsd:extension base="et_ProxyParamType">
        <xsd:attribute name="id" use="required">
          <xsd:simpleType>
            <xsd:restriction base="xsd:ID">
              <xsd:pattern value="ProxyParam[1-9][0-9]{0,4}"/>
            </xsd:restriction>
          </xsd:simpleType>
        </xsd:attribute>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
</xsd:element>
<xsd:element name="ProxiedParam" minOccurs="0" maxOccurs="unbounded">
  <xsd:complexType>
    <xsd:complexContent>
      <xsd:extension base="et_ParamType">
        <xsd:attribute name="id" use="required">
          <xsd:simpleType>
            <xsd:restriction base="xsd:ID">
              <xsd:pattern value="ProxiedParam[1-9][0-9]{0,4}"/>
            </xsd:restriction>
          </xsd:simpleType>
        </xsd:attribute>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
</xsd:element>
  <xsd:any namespace="##any" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="Assembly" minOccurs="0">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="Assembly_Class" minOccurs="0">
        <xsd:complexType>
          <xsd:sequence>
            <xsd:element name="Assembly_ClassAttributes" minOccurs="0">
              <xsd:complexType>
                <xsd:sequence>
                  <xsd:element name="ObjectRevision">
                    <xsd:complexType>

```

```

                                <xsd:attribute name="Access_Rule"
type="at_AccessType_Mandatory" use="required" fixed="Get"/>
                                </xsd:complexType>
                                </xsd:element>
                                <xsd:element name="MaxInstance">
                                    <xsd:complexType>
                                        <xsd:attribute name="Access_Rule"
type="at_AccessType_OptionalGet" use="required"/>
                                        </xsd:complexType>
                                    </xsd:element>
                                    <xsd:any namespace="##any" minOccurs="0" maxOccurs="unbounded"/>
                                </xsd:sequence>
                            </xsd:complexType>
                        </xsd:element>
                        <xsd:element name="Assembly_ClassOperations" minOccurs="0">
                            <xsd:complexType>
                                <xsd:sequence>
                                    <xsd:element name="Create">
                                        <xsd:complexType>
                                            <xsd:attribute ref="SupportedService"/>
                                        </xsd:complexType>
                                    </xsd:element>
                                    <xsd:element name="Delete">
                                        <xsd:complexType>
                                            <xsd:attribute ref="SupportedService"/>
                                        </xsd:complexType>
                                    </xsd:element>
                                    <xsd:element name="Get_Attribute_Single">
                                        <xsd:complexType>
                                            <xsd:attribute ref="SupportedService" fixed="true"/>
                                        </xsd:complexType>
                                    </xsd:element>
                                    <xsd:any namespace="##any" minOccurs="0" maxOccurs="unbounded"/>
                                </xsd:sequence>
                            </xsd:complexType>
                        </xsd:element>
                    </xsd:sequence>
                </xsd:complexType>
            </xsd:element>
            <xsd:element name="Assembly_Instance" minOccurs="0">
                <xsd:complexType>
                    <xsd:sequence>
                        <xsd:element name="Assembly_InstanceAttributes" minOccurs="0">
                            <xsd:complexType>
                                <xsd:sequence>
                                    <xsd:element name="NumberOfMembers">
                                        <xsd:complexType>
                                            <xsd:attribute name="Access_Rule"
type="at_AccessType_OptionalGet" use="required"/>
                                        </xsd:complexType>
                                    </xsd:element>
                                    <xsd:element name="AssemblyMemberList">
                                        <xsd:complexType>
                                            <xsd:attribute name="Access_Rule"
type="at_AccessType_OptionalSet" use="required"/>
                                        </xsd:complexType>
                                    </xsd:element>
                                    <xsd:element name="AssemblyData">
                                        <xsd:complexType>
                                            <xsd:attribute name="Access_Rule"
type="at_AccessType_Mandatory" use="required" fixed="Set"/>
                                        </xsd:complexType>
                                    </xsd:element>
                                    <xsd:any namespace="##any" minOccurs="0" maxOccurs="unbounded"/>
                                </xsd:sequence>
                            </xsd:complexType>
                        </xsd:element>
                        <xsd:element name="Assembly_InstanceOperations" minOccurs="0">
                            <xsd:complexType>
                                <xsd:sequence>
                                    <xsd:element name="Delete">
                                        <xsd:complexType>
                                            <xsd:attribute ref="SupportedService"/>
                                        </xsd:complexType>
                                    </xsd:element>
                                    <xsd:element name="Get_Attribute_Single">

```

```

        <xsd:complexType>
          <xsd:attribute ref="SupportedService" fixed="true"/>
        </xsd:complexType>
      </xsd:element>
      <xsd:element name="Set_Attribute_Single">
        <xsd:complexType>
          <xsd:attribute ref="SupportedService"/>
        </xsd:complexType>
      </xsd:element>
      <xsd:element name="Get_Member">
        <xsd:complexType>
          <xsd:attribute ref="SupportedService"/>
        </xsd:complexType>
      </xsd:element>
      <xsd:element name="Set_Member">
        <xsd:complexType>
          <xsd:attribute ref="SupportedService"/>
        </xsd:complexType>
      </xsd:element>
      <xsd:element name="Insert_Member">
        <xsd:complexType>
          <xsd:attribute ref="SupportedService"/>
        </xsd:complexType>
      </xsd:element>
      <xsd:element name="Remove_Member">
        <xsd:complexType>
          <xsd:attribute ref="SupportedService"/>
        </xsd:complexType>
      </xsd:element>
      <xsd:any namespace="##any" minOccurs="0" maxOccurs="unbounded"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="Assem" minOccurs="0" maxOccurs="unbounded">
  <xsd:complexType>
    <xsd:complexContent>
      <xsd:extension base="et_AssemType">
        <xsd:attribute name="id" use="required">
          <xsd:simpleType>
            <xsd:restriction base="xsd:ID">
              <xsd:pattern value="Assem[1-9][0-9]{0,4}"/>
            </xsd:restriction>
          </xsd:simpleType>
        </xsd:attribute>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
</xsd:element>
<xsd:element name="ProxyAssem" minOccurs="0" maxOccurs="unbounded">
  <xsd:complexType>
    <xsd:complexContent>
      <xsd:extension base="et_AssemType">
        <xsd:attribute name="id" use="required">
          <xsd:simpleType>
            <xsd:restriction base="xsd:ID">
              <xsd:pattern value="ProxyAssem[1-9][0-9]{0,4}"/>
            </xsd:restriction>
          </xsd:simpleType>
        </xsd:attribute>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
</xsd:element>
<xsd:element name="ProxiedAssem" minOccurs="0" maxOccurs="unbounded">
  <xsd:complexType>
    <xsd:complexContent>
      <xsd:extension base="et_AssemType">
        <xsd:attribute name="id" use="required">
          <xsd:simpleType>
            <xsd:restriction base="xsd:ID">
              <xsd:pattern value="ProxiedAssem[1-9][0-9]{0,4}"/>
            </xsd:restriction>
          </xsd:simpleType>
        </xsd:attribute>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
</xsd:element>

```

```

        </xsd:attribute>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
</xsd:element>
<xsd:any namespace="##any" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="ParameterGroup" minOccurs="0">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="Group" minOccurs="0" maxOccurs="unbounded">
        <xsd:complexType>
          <xsd:sequence>
            <xsd:element name="NameString" type="dt_EDS_Char_Array"/>
            <xsd:element name="NumberOfMembers" type="dt_UINT"/>
            <xsd:choice maxOccurs="unbounded">
              <xsd:element name="ParameterRef" type="dt_UINT"/>
              <xsd:element name="VariantRef" type="xsd:NMTOKEN"/>
              <xsd:any namespace="##any" minOccurs="0" maxOccurs="unbounded"/>
            </xsd:choice>
          </xsd:sequence>
          <xsd:attribute name="id" use="required">
            <xsd:simpleType>
              <xsd:restriction base="xsd:ID">
                <xsd:pattern value="Group[1-9][0-9]{0,4}"/>
              </xsd:restriction>
            </xsd:simpleType>
          </xsd:attribute>
        </xsd:complexType>
      </xsd:element>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
<xsd:any namespace="##any" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:attribute name="SupportedService" use="required">
  <xsd:simpleType>
    <xsd:restriction base="xsd:boolean">
      <xsd:pattern value="true|false"/>
    </xsd:restriction>
  </xsd:simpleType>
</xsd:attribute>
<xsd:complexType name="et_ParamType">
  <xsd:sequence>
    <xsd:element name="LinkPathSize" type="dt_USINT" minOccurs="0"/>
    <xsd:element name="LinkPath" type="dt_EPATH" minOccurs="0"/>
    <xsd:element name="ParamDescriptor" type="dt_WORD"/>
    <xsd:element name="DataType">
      <xsd:simpleType>
        <xsd:union memberTypes="dt_USINT dt_EPATH"/>
      </xsd:simpleType>
    </xsd:element>
    <xsd:element name="DataSize" type="dt_USINT"/>
    <xsd:element name="ParameterName" type="dt_EDS_Char_Array"/>
    <xsd:element name="UnitsString" type="dt_EDS_Char_Array"/>
    <xsd:element name="HelpString" type="dt_EDS_Char_Array"/>
    <xsd:element name="MinimumValue" minOccurs="0"/>
    <xsd:element name="MaximumValue" minOccurs="0"/>
    <xsd:element name="DefaultValue" minOccurs="0"/>
    <xsd:element name="ScalingMultiplier" type="dt_UINT" minOccurs="0"/>
    <xsd:element name="ScalingDivider" type="dt_UINT" minOccurs="0"/>
    <xsd:element name="ScalingBase" type="dt_UINT" minOccurs="0"/>
    <xsd:element name="ScalingOffset" type="dt_INT" minOccurs="0"/>
    <xsd:element name="MultiplierLink" type="dt_UINT" minOccurs="0"/>
    <xsd:element name="DivisorLink" type="dt_UINT" minOccurs="0"/>
    <xsd:element name="BaseLink" type="dt_UINT" minOccurs="0"/>
    <xsd:element name="OffsetLink" type="dt_INT" minOccurs="0"/>
    <xsd:element name="DecimalPrecision" type="dt_USINT" minOccurs="0"/>
    <xsd:element name="InternationalParameterName" type="dt_EDS_Char_Array" minOccurs="0"/>
    <xsd:element name="InternationalEngineeringUnits" type="dt_EDS_Char_Array" minOccurs="0"/>
    <xsd:element name="InternationalHelpString" type="dt_EDS_Char_Array" minOccurs="0"/>
    <xsd:element name="Enum" minOccurs="0">

```

```

<xsd:complexType>
  <xsd:sequence maxOccurs="unbounded">
    <xsd:element name="EnumValue" type="dt_LINT"/>
    <xsd:element name="EnumName" type="dt_EDS_Char_Array"/>
  </xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:any namespace="##any" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
<xsd:complexType name="et_ProxyParamType">
  <xsd:complexContent>
    <xsd:extension base="et_ParamType">
      <xsd:sequence>
        <xsd:element name="ProxyParamSizeAdder" minOccurs="0">
          <xsd:complexType>
            <xsd:sequence>
              <xsd:element name="MinimumValue"/>
              <xsd:element name="MaximumValue"/>
              <xsd:element name="DefaultValue"/>
            </xsd:sequence>
          </xsd:complexType>
        </xsd:element>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
<xsd:complexType name="et_AssemType">
  <xsd:sequence>
    <xsd:element name="AssemblyName" type="dt_EDS_Char_Array" minOccurs="0"/>
    <xsd:element name="AssemblyPath" type="dt_EPATH" minOccurs="0"/>
    <xsd:element name="AssemblyDataSize" type="dt_UINT" minOccurs="0"/>
    <xsd:element name="AssemblyDescriptor" type="dt_WORD" minOccurs="0"/>
    <xsd:element name="AssemblyMember" minOccurs="0" maxOccurs="unbounded">
      <xsd:complexType>
        <xsd:choice>
          <xsd:element name="MemberSize" type="dt_UINT"/>
          <xsd:element name="MemberReference" type="et_MemberReferenceType"/>
          <xsd:element name="VariantReference">
            <xsd:complexType/>
          </xsd:element>
        </xsd:choice>
        <xsd:sequence>
          <xsd:element name="MemberSize" type="dt_UINT"/>
          <xsd:element name="MemberReference" type="et_MemberReferenceType"/>
        </xsd:sequence>
        <xsd:sequence>
          <xsd:element name="MemberSize" type="dt_UINT"/>
          <xsd:element name="VariantReference">
            <xsd:complexType/>
          </xsd:element>
        </xsd:sequence>
      </xsd:choice>
    </xsd:choice>
  </xsd:complexType>
  <xsd:element>
    <xsd:any namespace="##any" minOccurs="0" maxOccurs="unbounded"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:simpleType name="et_MemberReferenceType">
  <xsd:union memberTypes="et_AssemReference et_ParamReference dt_UDINT dt_EPATH xsd:NMTOKEN"/>
</xsd:simpleType>
</xsd:schema>

```

A.2.2 Device profile template description – XML encapsulation of EDS files

A.2.2.1 General

The device profile XML files used to encapsulate EDS files shall comply with the device profile XML schema as specified in A.2.2.2.

The semantics of the sub-elements of the ExternalProfileHandle element, used to reference an existing EDS file, are specified in Table A.2. Depending on the value of the attribute WrapperReference, the EDS file will be referenced using either identification elements from the EDS file itself, or from the product described by this EDS.

NOTE 1 Choice of relevant identification elements will depend upon the expected usage of the wrapper file.

Table A.2 — ExternalProfileHandle elements

XML schema elements	WrapperReference = FILEINFO	WrapperReference = DEVICEINFO
ProfileIdentification	EDS File description text ^a	VendorID, Device Type, Product Code ^b
ProfileRevision	EDS Revision ^a	Product Revision ^b
ProfileLocation	EDS HomeURL ^a	Icon File Name ^b
^a See A.4.1.4.2 for more details		
^b See A.4.1.4.3 for more details		

If present, the DeviceIdentity, DeviceManager, DeviceFunction and ApplicationProcess elements should be compatible with the formats specified in A.2.1.3.3.

NOTE 2 This may be used during a transition phase between the legacy EDS format and the full XML format.

A.2.2.2 XML schema : EDS_Device_Profile_wrapper.xsd

NOTE This XML schema includes the file "MasterTemplateTypes.xsd" (see A.2.1.3.1).

```
<?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:redefine schemaLocation="MasterTemplateTypes.xsd">
    <xsd:complexType name="ISO15745Reference_DataType">
      <xsd:complexContent>
        <xsd:restriction base="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" fixed="EDS"/>
          </xsd:sequence>
        </xsd:restriction>
      </xsd:complexContent>
    </xsd:complexType>
  </xsd:redefine>
  <xsd:element name="ISO15745Profile">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="ProfileHeader"/>
        <xsd:element ref="ProfileBody"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>
  <xsd:annotation>
    <xsd:documentation>* HEADER SECTION *</xsd:documentation>
  </xsd:annotation>
  <xsd:element name="ProfileHeader">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="ProfileIdentification" type="xsd:string"/>
        <xsd:element name="ProfileRevision" type="xsd:string"/>
        <xsd:element name="ProfileName" type="xsd:string"/>
        <xsd:element name="ProfileSource" type="xsd:string"/>
        <xsd:element name="ProfileClassID" type="ProfileClassID_DataType" fixed="Device"/>
        <xsd:element name="ProfileDate" type="xsd:date" minOccurs="0"/>
        <xsd:element name="AdditionalInformation" type="xsd:anyURI" minOccurs="0"/>
        <xsd:element name="ISO15745Reference" type="ISO15745Reference_DataType"/>
        <xsd:element name="IASInterfaceType" type="IASInterface_DataType" minOccurs="0"
maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>
  <xsd:annotation>
    <xsd:documentation>* BODY SECTION *</xsd:documentation>
  </xsd:annotation>
  <xsd:element name="ProfileBody">
```

```

<xsd:complexType>
  <xsd:sequence>
    <xsd:element name="DeviceIdentity" minOccurs="0">
      <xsd:complexType>
        <xsd:sequence>
          <xsd:any namespace="##any"/>
        </xsd:sequence>
      </xsd:complexType>
    </xsd:element>
    <xsd:element name="DeviceManager" minOccurs="0">
      <xsd:complexType>
        <xsd:sequence>
          <xsd:any namespace="##any"/>
        </xsd:sequence>
      </xsd:complexType>
    </xsd:element>
    <xsd:element name="DeviceFunction" maxOccurs="unbounded">
      <xsd:complexType>
        <xsd:sequence>
          <xsd:any namespace="##any"/>
        </xsd:sequence>
      </xsd:complexType>
    </xsd:element>
    <xsd:element name="ApplicationProcess" minOccurs="0">
      <xsd:complexType>
        <xsd:sequence>
          <xsd:any namespace="##any"/>
        </xsd:sequence>
      </xsd:complexType>
    </xsd:element>
    <xsd:element name="ExternalProfileHandle">
      <xsd:complexType>
        <xsd:complexContent>
          <xsd:extension base="ProfileHandle_DataType">
            <xsd:attribute name="WrapperReference" use="optional" default="FILEINFO">
              <xsd:simpleType>
                <xsd:restriction base="xsd:NMTOKEN">
                  <xsd:enumeration value="FILEINFO"/>
                  <xsd:enumeration value="DEVICEINFO"/>
                </xsd:restriction>
              </xsd:simpleType>
            </xsd:attribute>
          </xsd:extension>
        </xsd:complexContent>
      </xsd:complexType>
    </xsd:element>
  </xsd:sequence>
</xsd:complexType>
</xsd:element>
</xsd:schema>

```

A.3 Communication network profile template description

A.3.1 Communication network profile template description – XML based

A.3.1.1 General

The communication network profile XML files shall comply with the communication network profile XML schema as specified in A.3.1.3.

Contents of this XML schema are derived from the communication network profile class diagrams shown in 6.1.2, and extended with additional elements to allow full description of communication network requirements or capabilities.

A.3.1.2 Semantics of XML schema elements

A.3.1.2.1 ProfileBody

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

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

A.3.1.2.2 ApplicationLayers

A.3.1.2.2.1 ConnectionObject

This element specifies the supported instance attributes and operations of the Connection Object (see IEC 62026-3:2000), together with a description of the individual connection instances and I/O characteristics.

The ConnectionDescriptions element identifies the instances of the Connection Object which are implemented in the device. The semantics of its Connection sub-element are specified in Table A.3.

Table A.3 — Connection elements

XML schema elements	Semantics
InstanceType	Specifies the type of connection (Explicit Message, Polled I/O, Bit Strobed I/O, Dynamic I/O)
ProductionTrigger	Specifies the type of trigger (Cyclic, Change Of State, Application Triggered)
TransportType	Specifies the type of transport (Server, Client)
TransportClass	Specifies the transport class (0, 2 or 3)
ProducedConnectionSize	Specifies the size of produced data in bytes
ConsumedConnectionSize	Specifies the size of consumed data in bytes

The semantics of the IO_Characteristics element are specified in A.4.2.4.2.

A.3.1.2.2.2 MessageRouter

This element specifies the supported instance attributes and operations of the Message Router Object (see IEC 62026-3:2000).

A.3.1.2.3 TransportLayers

A.3.1.2.3.1 DNPhysicalLayer

This element identifies the physical layer characteristics (e.g. connectors, baudrates, electrical characteristics).

The semantics its sub-elements are specified in Table A.4.

Table A.4 — DNPhysicalLayer elements

XML schema elements	XML schema attribute	Semantics
Connectors	ConnectorType	Specifies type of connectors (Open HardWired, Open Pluggable, Sealed Mini, SealedMicro)
LEDs		Specifies supported LED indicators
	Module	If attribute is present, corresponding LED is present
	Network	
	Combo_ModNet	
	IO	
Baud_Rate_Setting		Specifies supported baud rates
	SwitchType	Specifies supported methods for setting of baud rate selection
	SoftwareSettable	Specifies whether baud rate may be set by software
ElectricalCharacteristics		Sub-element specifies maximum network power consumption in Amps, at 11 Volts DC (worst case)
	IsolatedPhysicalLayer	Specifies whether physical layer is isolated or not

A.3.1.2.3.2 DNetLinkLayer

This element defines some properties associated with data link layer configuration.

The semantics its sub-elements are specified in Table A.5.

Table A.5 — DNetLinkLayer elements

XML schema elements	XML schema attribute	Semantics
Mac-IDSetting		Specifies the default MAC-ID
	SwitchType	Specifies supported methods for setting of MAC-ID
	SoftwareSettable	Specifies whether MAC-ID may be set via software
PredefinedMasterSlaveConnectionSet		Specifies the supported predefined connection set (none, Group2 Client and/or Group2 Server, GroupOnly2 Client, GroupOnly2 Server)
DynamicConnectionSupport		Specifies support of dynamic connections (none, Group1, Group2, Group3)
FragmentedExplicitMessaging		Sub-elements specify parameters for fragmented messaging
	Supported	Specifies whether fragmented messaging is supported

A.3.1.2.3.3 DeviceNetObject

This element specifies the supported instance attributes and operations of the DeviceNet Object (see IEC 62026-3:2000).

A.3.1.2.3.4 Ports

This element identifies the device ports which are able to route messages from one link to another link.

The semantics of the Port sub-element of the Ports element are specified in A.4.1.4.10 and A.4.2.2.2.

A.3.1.2.4 NetworkManagement

A.3.1.2.4.1 NM-DeviceNetObject

This element specifies the supported class attributes and operations of the DeviceNet Object (see IEC 62026-3:2000).

A.3.1.2.4.2 NM-ConnectionObject

This element specifies the supported class attributes and operations of the Connection Object (see IEC 62026-3:2000).

A.3.1.2.4.3 NM-MessageRouter

This element specifies the supported class attributes and operations of the Message Router Object (see IEC 62026-3:2000).

A.3.1.3 XML schema : DNet_CommNet_Profile.xsd

NOTE This XML schema includes the files "MasterTemplateTypes.xsd" (see A.2.1.3.1) and "CIPDataTypes.xsd" (see A.2.1.3.2).

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">
  <!-- Target namespaces are not specified in this master template -->
  <xsd:redefine schemaLocation="MasterTemplateTypes.xsd">
```

```

<xsd:complexType name="ISO15745Reference_DataType">
  <xsd:complexContent>
    <xsd:restriction base="ISO15745Reference_DataType">
      <xsd:sequence>
        <xsd:element name="ISO15745Part" type="xsd:positiveInteger" fixed="2"/>
        <xsd:element name="ISO15745Edition" type="xsd:positiveInteger" fixed="1"/>
        <xsd:element name="ProfileTechnology" type="xsd:string" fixed="DeviceNet"/>
      </xsd:sequence>
    </xsd:restriction>
  </xsd:complexContent>
</xsd:complexType>
</xsd:redefine>
<xsd:include schemaLocation="CIPDataTypes.xsd"/>
<xsd:element name="ISO15745Profile">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element ref="ProfileHeader"/>
      <xsd:element ref="ProfileBody"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
<xsd:annotation>
  <xsd:documentation>* HEADER SECTION *</xsd:documentation>
</xsd:annotation>
<xsd:element name="ProfileHeader">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="ProfileIdentification" type="xsd:string"/>
      <xsd:element name="ProfileRevision" type="xsd:string"/>
      <xsd:element name="ProfileName" type="xsd:string"/>
      <xsd:element name="ProfileSource" type="xsd:string"/>
      <xsd:element name="ProfileClassID" type="ProfileClassID_DataType"
fixed="CommunicationNetwork"/>
      <xsd:element name="ProfileDate" type="xsd:date" minOccurs="0"/>
      <xsd:element name="AdditionalInformation" type="xsd:anyURI" minOccurs="0"/>
      <xsd:element name="ISO15745Reference" type="ISO15745Reference_DataType"/>
      <xsd:element name="IASInterfaceType" type="IASInterface_DataType" fixed="CSI"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
<xsd:annotation>
  <xsd:documentation>* BODY SECTION *</xsd:documentation>
</xsd:annotation>
<xsd:element name="ProfileBody">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element ref="ApplicationLayers"/>
      <xsd:element ref="TransportLayers"/>
      <xsd:element ref="NetworkManagement" minOccurs="0"/>
    </xsd:sequence>
    <xsd:attributeGroup ref="ag_FileDescription"/>
  </xsd:complexType>
</xsd:element>
<xsd:element name="ApplicationLayers">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element ref="ConnectionObject"/>
      <xsd:element ref="MessageRouter"/>
      <xsd:any namespace="##any" minOccurs="0" maxOccurs="unbounded"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
<xsd:element name="TransportLayers">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element ref="DNPhysicalLayer"/>
      <xsd:element ref="DNLinkLayer"/>
      <xsd:element ref="DeviceNetObject"/>
      <xsd:element ref="Ports" minOccurs="0"/>
      <xsd:any namespace="##any" minOccurs="0" maxOccurs="unbounded"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
<xsd:element name="NetworkManagement">
  <xsd:complexType>
    <xsd:sequence>

```

```

        <xsd:element ref="NM-DeviceNetObject"/>
        <xsd:element ref="NM-ConnectionObject" minOccurs="0"/>
        <xsd:element ref="NM-MessageRouter" minOccurs="0"/>
    </xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="ConnectionObject">
    <xsd:complexType>
        <xsd:sequence>
            <xsd:element name="ConnectionObject_InstanceAttributes" minOccurs="0">
                <xsd:complexType>
                    <xsd:sequence>
                        <xsd:element name="State">
                            <xsd:complexType>
                                <xsd:attribute name="Access_Rule" type="at_AccessType_Mandatory"
use="required" fixed="Get"/>
                            </xsd:complexType>
                        </xsd:element>
                        <xsd:element name="InstanceType">
                            <xsd:complexType>
                                <xsd:attribute name="Access_Rule" type="at_AccessType_Mandatory"
use="required" fixed="Get"/>
                            </xsd:complexType>
                        </xsd:element>
                        <xsd:element name="TransportClassTrigger">
                            <xsd:complexType>
                                <xsd:attribute name="Access_Rule" type="at_AccessType_Mandatory"
use="required" fixed="Set"/>
                            </xsd:complexType>
                        </xsd:element>
                        <xsd:element name="ProducedConnectionID">
                            <xsd:complexType>
                                <xsd:attribute name="Access_Rule" type="at_AccessType_Mandatory"
use="required" fixed="Set"/>
                            </xsd:complexType>
                        </xsd:element>
                        <xsd:element name="ConsumedConnectionID">
                            <xsd:complexType>
                                <xsd:attribute name="Access_Rule" type="at_AccessType_Mandatory"
use="required" fixed="Set"/>
                            </xsd:complexType>
                        </xsd:element>
                        <xsd:element name="InitialCommCharacteristics">
                            <xsd:complexType>
                                <xsd:attribute name="Access_Rule" type="at_AccessType_Mandatory"
use="required" fixed="Set"/>
                            </xsd:complexType>
                        </xsd:element>
                        <xsd:element name="ProducedConnectionSize">
                            <xsd:complexType>
                                <xsd:attribute name="Access_Rule" type="at_AccessType_Mandatory"
use="required" fixed="Set"/>
                            </xsd:complexType>
                        </xsd:element>
                        <xsd:element name="ConsumedConnectionSize">
                            <xsd:complexType>
                                <xsd:attribute name="Access_Rule" type="at_AccessType_Mandatory"
use="required" fixed="Set"/>
                            </xsd:complexType>
                        </xsd:element>
                        <xsd:element name="ExpectedPacketRate">
                            <xsd:complexType>
                                <xsd:attribute name="Access_Rule" type="at_AccessType_Mandatory"
use="required" fixed="Set"/>
                            </xsd:complexType>
                        </xsd:element>
                        <xsd:element name="WatchdogTimeoutAction">
                            <xsd:complexType>
                                <xsd:attribute name="Access_Rule" type="at_AccessType_Mandatory"
use="required" fixed="Set"/>
                            </xsd:complexType>
                        </xsd:element>
                        <xsd:element name="ProducedConnectionPathLength">
                            <xsd:complexType>
                                <xsd:attribute name="Access_Rule" type="at_AccessType_Mandatory"
use="required" fixed="Get"/>
                            </xsd:complexType>
                        </xsd:element>
                    </xsd:sequence>
                </xsd:complexType>
            </xsd:element>
        </xsd:sequence>
    </xsd:complexType>
</xsd:element>

```

```

        </xsd:complexType>
      </xsd:element>
      <xsd:element name="ProducedConnectionPath">
        <xsd:complexType>
          <xsd:attribute name="Access_Rule" type="at_AccessType_Mandatory"
use="required" fixed="Set"/>
        </xsd:complexType>
      </xsd:element>
      <xsd:element name="ConsumedConnectionPathLength">
        <xsd:complexType>
          <xsd:attribute name="Access_Rule" type="at_AccessType_Mandatory"
use="required" fixed="Get"/>
        </xsd:complexType>
      </xsd:element>
      <xsd:element name="ConsumedConnectionPath">
        <xsd:complexType>
          <xsd:attribute name="Access_Rule" type="at_AccessType_Mandatory"
use="required" fixed="Set"/>
        </xsd:complexType>
      </xsd:element>
      <xsd:element name="ProductionInhibitTime">
        <xsd:complexType>
          <xsd:attribute name="Access_Rule" type="at_AccessType_OptionalSet"
use="required"/>
        </xsd:complexType>
      </xsd:element>
      <xsd:any namespace="##any" minOccurs="0" maxOccurs="unbounded"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
<xsd:element name="ConnectionObject_InstanceOperations" minOccurs="0">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="Reset">
        <xsd:complexType>
          <xsd:attribute ref="SupportedService"/>
        </xsd:complexType>
      </xsd:element>
      <xsd:element name="Delete">
        <xsd:complexType>
          <xsd:attribute ref="SupportedService"/>
        </xsd:complexType>
      </xsd:element>
      <xsd:element name="Apply_Attributes">
        <xsd:complexType>
          <xsd:attribute ref="SupportedService"/>
        </xsd:complexType>
      </xsd:element>
      <xsd:element name="Get_Attribute_Single">
        <xsd:complexType>
          <xsd:attribute ref="SupportedService" fixed="true"/>
        </xsd:complexType>
      </xsd:element>
      <xsd:element name="Set_Attribute_Single">
        <xsd:complexType>
          <xsd:attribute ref="SupportedService"/>
        </xsd:complexType>
      </xsd:element>
      <xsd:any namespace="##any" minOccurs="0" maxOccurs="unbounded"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
<xsd:element ref="ConnectionDescriptions" minOccurs="0"/>
<xsd:element name="IO_Characteristics" minOccurs="0">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="IO_Info" type="et_IOInfoType"/>
      <xsd:any namespace="##any" minOccurs="0" maxOccurs="unbounded"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
<xsd:any namespace="##any" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="ConnectionDescriptions">

```