ФЕДЕРАЛЬНОЕ АГЕНТСТВО

ПО ТЕХНИЧЕСКОМУ РЕГУЛИРОВАНИЮ И МЕТРОЛОГИИ

НАЦИОНАЛЬНЫЙ СТАНДАРТ РОССИЙСКОЙ ФЕДЕРАЦИИ

ГОСТ Р ИСО 15745-2 — 2010

Системы промышленной автоматизации и интеграция

ПРИКЛАДНАЯ ИНТЕГРАЦИОННАЯ СРЕДА ОТКРЫТЫХ СИСТЕМ

Часть 2

Эталонное описание систем управления на основе стандарта ИСО 11898

ISO 15745-2:2003
Industrial automation systems and integration — Open systems application integration framework — Part 2: Reference description for ISO 11898-based control system
(IDT)

Издание официальное

Москва
Стандартинформ
2013

gost.e-ecolog.ru

Предисловие

Цели и принципы стандартизации в Российской Федерации установлены Федеральным законом от 27 декабря 2002 г. №184-ФЗ «О техническом регулировании», а правила применения национальных стандартов Российской Федерации — ГОСТ Р 1.0—2004 «Стандартизация в Российской Федерации. Основные положения»

Сведения о стандарте

1 ПОДГОТОВЛЕН Научно-техническим центром ИНТЕК на основе собственного аутентичного перевода на русский язык международного стандарта, указанного в пункте 4

2 ВНЕСЕН Техническим комитетом по стандартизации ТК 100 «Стратегический и инновационный менеджмент»

3 УТВЕРЖДЕН И ВВЕДЕН В ДЕЙСТВИЕ Приказом Федерального агентства по техническому регулированию и метрологии от 21 декабря 2010 г. № 865-ст

4 Настоящий стандарт идентичен международному стандарту ИСО 15745-2:2003 «Системы промышленной автоматизации и интеграция. Прикладная интеграционная среда открытых систем. Часть 2. Эталонное описание систем управления на основе стандарта ИСО 11898» (ISO 15745-2:2003 «Industrial automation systems and integration — Open systems application integration framework — Part 2: Reference description for ISO 11898-based control system»).

При применении настоящего стандарта рекомендуется использовать вместо ссылочных международных и европейского регионального стандартов соответствующие им национальные стандарты Российской Федерации, сведения о которых приведены в дополнительном приложении ДА

ВВЕДЕН ВПЕРВЫЕ

Информация об изменениях к настоящему стандарту публикуется в ежегодно издаваемом информационном указателе «Национальные стандарты», а текст изменений и поправок — в ежемесячно издаваемых информационных указателях «Национальные стандарты». В случае пересмотра (замены) или отмены настоящего стандарта соответствующее уведомление будет опубликовано в ежемесячно издаваемом информационном указателе «Национальные стандарты». Соответствующая информация, уведомление и тексты размещаются также в информационной системе общего пользования — на официальном сайте Федерального агентства по техническому регулированию и метрологии в сети Интернет

© Стандартинформ, 2013

Настоящий стандарт не может быть полностью или частично воспроизведен, тиражирован и распространен в качестве официального издания без разрешения Федерального агентства по техническому регулированию и метрологии

Содержание

1 Область применения . . . 1
2 Нормативные ссылки . . . 1
3 Термины и сокращения . . . 2
4 Обозначения и сокращения . . . 2
5 Технологические специальные элементы и правила . . . 3
5.1 Интеграционные модели и интерфейсы IAS . . . 3
5.2 Шаблоны профилей . . . 3
5.2.1 Общие положения . . . 3
5.2.2 Содержимое и синтаксис . . . 3
5.2.3 Заголовок . . . 3
5.3 Технологические специальные профили . . . 3
6 Профили устройств и коммуникационных сетей для систем управления на основе ИСО 11898 . . . 4
6.1 Протокол DeviceNet . . . 4
6.1.1 Профиль устройств . . . 4
6.1.2 Профиль коммуникационных сетей . . . 5
6.2 Протокол CANopen . . . 7
6.2.1 Профиль устройств . . . 7
6.2.2 Профиль коммуникационных сетей . . . 13
Приложение А (обязательное) Шаблоны профилей DeviceNet . . . 15
Приложение В (обязательное) Шаблоны профилей CANopen . . . 94
Приложение ДА (справочное) Сведения о соответствии ссылочных международных и европейского регионального стандартов ссылочным национальным стандартам Российской Федерации . . . 170
Библиография . . . 170

Введение

Прикладная интеграционная среда (AIF) определяет элементы и правила, облегчающие:

- систематическую организацию и представление требований к интеграции прикладных систем с помощью моделей интеграции;

- разработку спецификаций интерфейсов в форме профилей интероперабельности прикладных систем (AIP), что позволяет выбрать как соответствующие ресурсы, так и документацию «встроенных» прикладных систем.

ИСО 15745-1 устанавливает общие элементы и правила описания интеграционных моделей и AIP вместе с их профилями компонент — профили процессов, профили обмена информацией и профили ресурсов. Содержание комплекса стандартов ИСО 15745 и структурный вид составных частей AIP показан на рисунке 1 ИСО 15745-1:2003.

Настоящий стандарт расширяет общую AIF, описанную в ИСО 15745-1, путем определения технологических специальных элементов и правил описания как профилей коммуникационных сетей, так и коммуникационных аспектов профилей устройств (приборов), относящихся к системам управления на основе ИСО 11898 (DeviceNet[1)], CANopen[2)]).

В частности, в настоящем стандарте описаны шаблоны технологического специального профиля для профиля устройств (приборов) и профиля коммуникационной сети. В рамках AIP экземпляр профиля устройств (приборов) или экземпляр профиля коммуникационной сети является частью профиля ресурсов, определенного в ИСО 15745-1. Файлы экземпляров XML профиля устройств (приборов) и профиля коммуникационной сети включаются в экземпляр XML профиля ресурсов с использованием файла Profile-Handle_ DataType, как установлено в ИСО 15745-1:2003, пункт 7.2.5.

AIF, использующие элементы и правила ИСО 15745-1, могут быть легко интегрированы с профилями компонент, определенными с использованием элементов и правил, установленных в настоящем стандарте.

Настоящий стандарт разработан техническим комитетом ИСО/ТК 184 «Системы промышленной автоматизации и интеграция», подкомитетом ПК 5 «Архитектура, системы связи и интегрированные среды».

Комплекс стандартов ИСО 15745 имеет общее название «Системы промышленной автоматизации и интеграция. Прикладная интеграционная среда открытых систем» и включает в себя следующие части:

- часть 1. Общее эталонное описание;
- часть 2. Эталонное описание систем управления на основе стандарта ИСО 11898;
- часть 3. Эталонное описание систем управления на основе стандарта МЭК 61158;
- часть 4. Эталонное описание систем управления на основе стандарта Ethernet.

[1)] DeviceNet™ — торговая марка Open DeviceNet Vendor Association Inc. Эта информация приведена для удобства пользователей ИСО 15745 и не означает одобрения со стороны ИСО владельца торговой марки или любых его продуктов. Соответствие настоящему стандарту не требует использования торговой марки DeviceNet™. Использование торговой марки DeviceNet™ требует разрешения Open DeviceNet Vendor Association Inc.

[2)] CANopen — торговая марка, используемая для описания ЕН 50325-4. Эта информация приведена для удобства пользователей ИСО 15745 и не означает одобрения со стороны ИСО владельца торговой марки или любых его продуктов. Соответствие настоящему стандарту не требует использования торговой марки CANopen.

НАЦИОНАЛЬНЫЙ СТАНДАРТ РОССИЙСКОЙ ФЕДЕРАЦИИ

Системы промышленной автоматизации и интеграция

ПРИКЛАДНАЯ ИНТЕГРАЦИОННАЯ СРЕДА ОТКРЫТЫХ СИСТЕМ

Часть 2

Эталонное описание систем управления на основе стандарта ИСО 11898

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

Дата введения — 2011—09—01

1 Область применения

Настоящий стандарт устанавливает специальные технологические элементы и правила описания как профилей коммуникационных сетей, так и коммуникационных аспектов профилей устройств (приборов), относящихся к системам управления на основе ИСО 11898.

Примечание — Общие элементы и правила описания интеграционных моделей и профилей интероперабельности прикладных систем с профилями их компонент (профили процессов, профили обмена информации и профили ресурсов) установлены в ИСО 15745-1.

Настоящий стандарт следует использовать вместе с ИСО 15745-1.

2 Нормативные ссылки

В настоящем стандарте использованы нормативные ссылки на следующие стандарты, которые необходимо учитывать при использовании настоящего стандарта. В случае ссылок на документы, у которых указана дата утверждения, необходимо пользоваться только указанной редакцией. В случае, когда дата утверждения не приведена, следует пользоваться редакцией ссылочных документов, включая любые поправки и изменения к ним:

ИСО 639-1:2002 Коды для представления названий языков. Часть 1. Двухбуквенный код (ISO 639-1:2002, Codes for the representation of names of languages — Part 1: Alpha-2 code)

ИСО 639-2:1998 Коды для представления названий языков. Часть 2. Трехбуквенный код (ISO 639-2:1998, Codes for the representation of names of languages — Part 2: Alpha-3 code)

ИСО 3166-1:1997 Коды для представления названий стран и единиц их административно-территориального деления. Часть 1. Коды стран (ISO 3166-1:1997, Codes for the representation of names of countries and their subdivisions — Part 1: Country code)

ИСО/МЭК 10646-1:2000 Информационные технологии. Универсальный многооктетный кодированный набор символов (UCS). Часть 1. Архитектура и базовая многоязычная плоскость (ISO/IEC 10646-1:2000, Information technology — Universal Multiple-Octet Coded Character Set (UCS) — Part 1: Architecture and Basic Multilingual Plane)

ИСО 11898:1993 Дорожные транспортные средства. Обмен цифровой информацией. Сеть контроллеров (CAN) для высокоскоростной связи (ISO 11898:1993, Road Vehicles — Interchange of digital information — Controller area network (CAN) for high-speed communication)

Издание официальное

ИСО 15745-1:2003 Системы промышленной автоматизации и интеграция. Прикладная интеграционная среда открытых систем. Часть 1. Общее эталонное описание (ISO 15745-1:2003, Industrial automation and systems integration — Open systems application integration framework — Part 1: Generic reference description)

МЭК 61158 (все части) Передача цифровых данных для измерения и управления. Полевая шина для систем автоматического регулирования и управления технологическими процессами (IEC 61158 (all parts), Digital data communications for measurement and control — Fieldbus for use in industrial control systems)

МЭК 61784-1:2003 Цифровые передачи данных для измерения и управления. Часть 1. Профильные наборы для непрерывного и раздельного производства полевых шин для систем автоматического регулирования и управления технологическими процессами (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)

МЭК 62026-3:2000 Устройства распределительные комплектные низковольтные. Интерфейсы между контроллером и прибором (CDIs). Часть 3. DeviceNETTM (IEC 62026-3:2000, Low-voltage switchgear and controlgear — Controller-device interfaces (CDIs) — Part 3: DeviceNetTM)

ЕН 50325-4:2002 Подсистема промышленной коммуникации на основе ИСО 11898 (CAN) для интерфейсов контроллер-прибор. Часть 4. CANopen (EN 50325-4:2002, Industrial communications subsystem based on ISO 11898 (CAN) for controller-device interfaces — Part 4: CANopen)

IEEE Std 754-1985 (R1990) Стандарт IEEE для двоичной арифметики с плавающей запятой (IEEE Std 754-1985 (R1990), IEEE Standard for Binary Floating-Point Arithmetic)

REC-xml-20001006 Расширяемый язык разметки (XML) 1.0 Второе издание — Рекомендация W3C от 6 октября 2000 г. (REC-xml-20001006, Extensible Markup Language (XML) 1.0 Second Edition — W3C Recommendation 6 October 2000)

REC-xmlschema-1-20010502 Схема XML, Часть 1. Структуры — Рекомедация W3C от 2 мая 2001 (REC-xmlschema-1-20010502, XML Schema. Part 1: Structures — W3C Recommendation 02 May 2001)

REC-xmlschema-2-20010502 Схема XML, Часть 2. Типы данных — Рекомедация W3C от 2 мая 2001 (REC-xmlschema-2-20010502, XML Schema. Part 2: Datatypes — W3C Recommendation 02 May 2001)

RFC 1738:1994 Единый указатель ресурсов (URL) — Целевая группа инженерной поддержки Интернета (IETF), Запрос на комментарий (RFC) (RFC 1738:1994, Uniform Resource Locators (URL) — Internet Engineering Task Force (IETF), Request for Comments (RFC))

RFC 1759:1995 Принтер MIB — Целевая группа инженерной поддержки Интернета (IETF), Запрос на комментарий (RFC) (RFC 1759:1995, Printer MIB — Internet Engineering Task Force (IETF), Request for Comments (RFC))

UML V1.4 OMG — Спецификация для Унифицированного Языка Моделирования (Версия 1.4, сентябрь 2001) (UML V1.4, OMG — Unified Modeling Language Specification (Version 1.4, September 2001))

3 Термины и определения

В настоящем стандарте использованы термины и определения по ИСО 15745-1 (приложение А).

4 Обозначения и сокращения

AIF — прикладная интеграционная среда (Application Integration Framework);
AIP — профиль интероперабельности приложений (Application Interoperability Profile);
CAN — сеть контроллеров (Controller Area Network);
CIP™[3)] — общий промышленный протокол (Common Industrial Protocol);
EDS — электронная информационная таблица; электронная таблица данных (Electronic Data Sheet);
IAS — системы промышленной автоматизации (Industrial Automation Systems);
OSI — взаимодействие открытых систем (Open System Interconnection);
UML — унифицированный язык моделирования (см. UML V1.4) (Unified Modelling Language);
XML — расширяемый язык разметки (см. REC-xml-20001006) (eXtensible Markup Language).

[3)] CIP™ — торговая марка ControlNet International, Ltd., и Open DeviceNet Vendor Association, Inc. Эта информация приведена для удобства использования комплекса стандартов ИСО 15745 и не означает одобрения со стороны ИСО владельца торговой марки или любых его продуктов. Соответствие настоящему стандарту не требует использования торговой марки DeviceNet™. Использование торговой марки CIP™ требует разрешения либо ControlNet International, Ltd., либо Open DeviceNet Vendor Association, Inc.

5 Технологические специальные элементы и правила

5.1 Интеграционные модели и интерфейсы IAS

Разработчик AIP должен создать интеграционную модель с использованием правил, описанных в ИСО 15745-1, а также должен обеспечить, чтобы профили устройства (приборов) и коммуникационных сетей на основе ИСО 11898 (независимо от того, представляют ли они требования к интерфейсам или требования, выводимые из существующих приборов/коммуникационных сетей) включали в себя необходимые интерфейсы IAS. Интерфейсы IAS, включенные в профиль, должны быть идентифицированы в разделе заголовка (см. ИСО 15745-1:2003, пункт 7.2.2).

Примечание — Интерфейсы IAS описаны в ИСО 15745-1:2003 (приложение В).

5.2 Шаблоны профилей

5.2.1Общие положения

Шаблоны технологических специальных профилей на основе ИСО 11898 выводятся из общих шаблонов профилей, установленных в ИСО 15745-1 (раздел 7).

5.2.2 Содержимое и синтаксис

Комплекс стандартов ИСО 15745 устанавливает шаблоны профилей, которые являются XML схемами (REC-xmlschema-1-20010502 и REC-xmlschema-2-20010502) и используют общую основную структуру. Профили устройств (приборов) и коммуникационных сетей, основанные на этих шаблонах, как правило, содержат:

- информацию, необходимую для идентификации подсоединенного прибора;
- описание данных об устройстве (приборе), которые могут быть оценены через сеть;
- описание коммуникационных возможностей, поддерживаемых устройством (прибором);
- дополнительную информацию от продавца.

Однако некоторые технологии, основанные на ИСО 11898, используют специальный унаследованный синтаксис ASCII. Поэтому обратная совместимость, определения шаблонов любой технологии (приложения А, В) включают в себя все или соответствующие поднаборы, а именно:

- шаблоны профилей коммуникационных сетей и устройств (приборов), установленные в ИСО 15745-1;
- шаблон ИСО 15745, инкапсулирующий файлы с унаследованным синтаксисом ASCII («упаковщик»);
- унаследованный синтаксис ASCII.

5.2.3 Заголовок

Заголовок шаблона профилей, установленный в ИСО 15745-1:2003 (пункт 7.2.2), используют для шаблонов технологических специальных профилей ИСО 11898. Каждая технология использует одно или несколько имен для идентификации технологии или ее отдельную компоненту (отдельные компоненты) (см. таблицу 1). Выбранное имя должно храниться в атрибуте ProfileTechnology в разделе заголовка.

Таблица 1 — Имена атрибута ProfileTechnology

Имя атрибута ProfileTechnology	Технология
DeviceNet	DeviceNet
CIP	DeviceNet
EDS	DeviceNet
CANopen	CANopen
COFDCML	CANopen

5.3 Технологические специальные профили

Структура технологических специальных профилей коммуникационных сетей и коммуникационные аспекты структуры профилей устройств (приборов), основанные на ИСО 11898, описаны в разделе 6, включающем в себя следующие технологии:

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

Определения связанных шаблонов профилей даны в приложениях А и В.

6 Профили устройств и коммуникационных сетей для систем управления на основе ИСО 11898

6.1 Протокол DeviceNet

6.1.1 Профиль устройств

6.1.1.1 Общие положения

На рисунке 1 показана диаграмма классов профилей устройств DeviceNet.

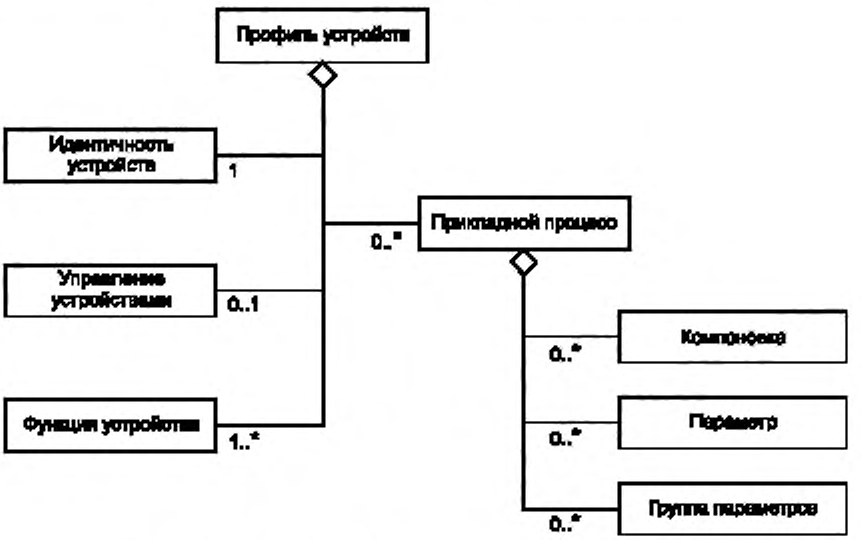


Рисунок 1 — Диаграмма классов профилей устройств DeviceNet

Доступные форматы для профилей устройств DeviceNet описаны в А.2 (приложение А).

XML схема, представляющая шаблон профилей устройств DeviceNet, определена в А.2.1.3.3 (приложение А). Имя XML файла этой схемы должно быть «CIP_Device_Profile.xsd».

Примечание — Диаграмма классов профилей устройств DeviceNet, представленная на рисунке 1, определяет основные классы. Эти классы разбиваются на составные части; подробное описание дано в приложении А.

XML схема, представляющая инкапсуляцию унаследованного DeviceNet EDS в шаблон профилей приборов на основе ИСО 15745, определена в А.2.2.2 (приложение А). Имя XML файла этой схемы должно быть «EDS_Device_Profile_wrapper.xsd». Сам унаследованный синтаксис EDS ASCII описан в А.4 (приложение А).

6.1.1.2 Идентичность устройств

Класс Идентичность устройств (DeviceIdentity) содержит атрибуты, которые однозначно идентифицируют прибор и поддерживают сервисы, позволяющие извлечь эту информацию из устройства.

Эти атрибуты обеспечивают, в частности:

- идентификацию изготовителя (название и идентификационный код);
- идентификацию устройств (тип устройства, название изделия, модификация, серийный номер);
- классификацию устройств;
- место хранения дополнительной информации (например, иконки).

6.1.1.3 Управление устройствами

Класс Управление устройствами (DeviceManager) содержит атрибуты и поддерживает сервисы, используемые для мониторинга и конфигурирования приборов.

Эти атрибуты обеспечивают, в частности:
- модификацию объекта Идентичность DeviceNet;
- информацию по структуре устройства (для устройств, интегрированных в модульную систему).

Сервисы позволяют:
- отключить устройство;
- осуществлять поиск атрибутов DeviceManager.

6.1.1.4 Функция устройств

Класс Функция устройств (DeviceFunction) содержит атрибуты и поддерживает сервисы, позволяющие осуществлять функциональное управление (например, конфигурирование) устройствами.

Пример — Примерами объектов DeviceFunction являются объекты перегрузки, распознавания присутствия объекта, аналоговых входных данных и дискретных выходных данных.

Примечание — Определение специального класса DeviceFunction в ИСО 15745-2 не установлено.

6.1.1.5 Прикладной процесс

На рисунке 2 показана структура класса Прикладной процесс (ApplicationProcess).

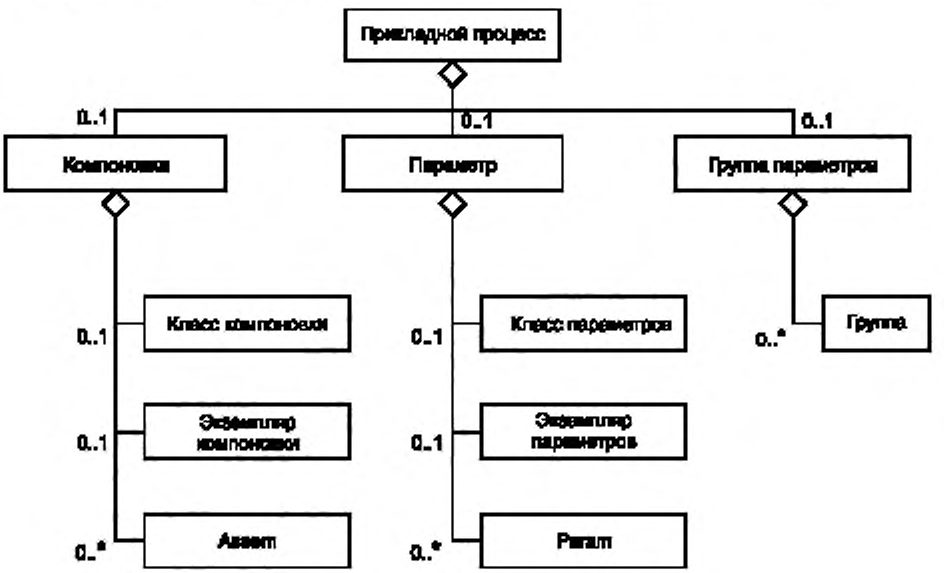


Рисунок 2 — Диаграмма класса ApplicationProcess DeviceNet

Класс Компоновка (Assembly) собирает несколько элементов данных прикладного процесса в один блок для оптимизации коммуникаций. Класс Параметр (Parameter) обеспечивает стандартизованный интерфейс для оценивания отдельных элементов данных прикладного процесса. Класс Группа Параметров (ParameterGroup) устанавливает группы связанных параметров для определенной цели (например, конфигурирование, мониторинг). Класс Компоновка и класс Параметр поддерживают атрибуты и сервисы как на уровне класса, так и на уровне экземпляра.

Классы Assem (Компоновка), Param (Параметр) и Group (Группа) определяют отдельные экземпляры основных классов.

Примечание — Класс Компоновка и класс Параметр соответствуют объектам Компоновка DeviceNet и Параметр. Объект Компоновка полностью определен в МЭК 61158-5:2003 и МЭК 61158-6:2003 (тип 2).

6.1.2 Профиль коммуникационных сетей

6.1.2.1 Общие положения

На рисунке 3 показана структура классов профилей коммуникационных сетей DeviceNet.

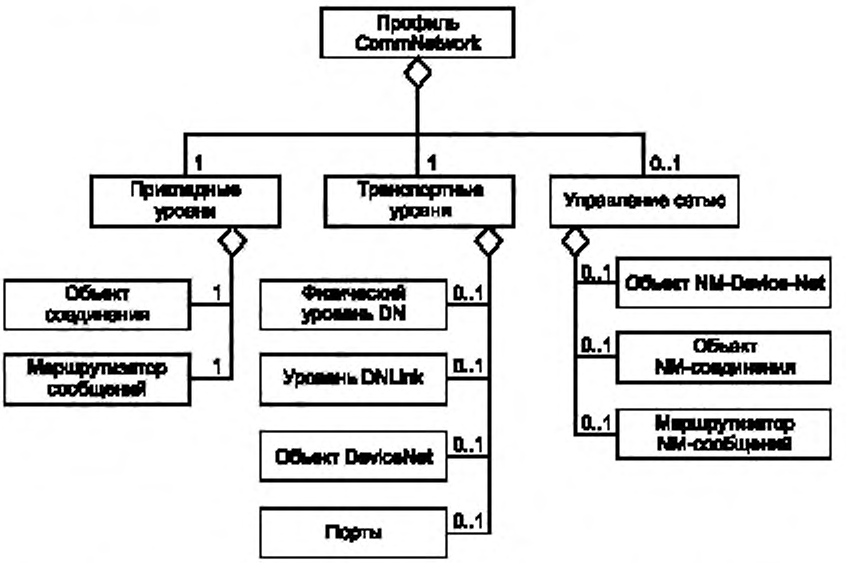


Рисунок 3 — Диаграмма классов профилей коммуникационных сетей DeviceNet

Доступные форматы для профилей коммуникационных сетей DeviceNet описаны в А.3 (приложение А).

XML схема, представляющая шаблон профилей коммуникационных сетей DeviceNet, определена в А.3.1.3 (приложение А). Имя XML файла этой схемы должно быть «DNet_CommNet_Profile.xsd».

XML схема, представляющая инкапсуляцию унаследованного DeviceNet EDS в шаблон профилей коммуникационных сетей на основе ИСО 15745, определена в А.3.2.2 (приложение А). Имя XML файла этой схемы должно быть «EDS_CommNet_Profile_wrapper.xsd». Сам унаследованный синтаксис EDS ASCII описан в А.4 (приложение А).

6.1.2.2 Прикладные уровни

Класс Прикладные уровни (ApplicationLayers) DeviceNet представляет составные профили для трех верхних уровней интегрированной OSI модели коммуникационных сетей DeviceNet.

Далее он разбивается на несколько классов, как показано на рисунке 3:

- объект Соединение (ConnectionObject) определяет характеристики, связанные с соединениями, и управление соединениями;

- объект Маршрутизатор сообщений (MessageRouter) определяет характеристики, связанные с внутренней маршрутизацией сообщений в устройстве.

Примечание — Соответствующие объект Соединение и объект Маршрутизатор Сообщений полностью определены в МЭК 62026-3.

6.1.2.3 Транспортные уровни

Класс Транспортные уровни (TransportLayer) DeviceNet представляет составные профили для четырех нижних уровней интегрированной OSI модели коммуникационных сетей DeviceNet.

Далее он разбивается на несколько классов, как показано на рисунке 3:

- физический Уровень DN (DNPhysicalLayer) идентифицирует характеристики физического уровня (например, соединитель, скорость передачи данных в бодах, электрические характеристики);

- канальный Уровень DN (DNLinkLayer) и объект DeviceNet (DeviceNetObject) определяют характеристики, связанные с конфигурированием канального уровня передачи данных и мониторингом;

- порты идентифицируют порты устройства, которые могут маршрутизировать сообщения от одного канала к другому каналу.

Примечание — Соответствующий объект DeviceNet полностью определен в МЭК 62026-3:2000.

6.1.2.4 Управление сетями

Класс Управление сетями (NetworkManagement) DeviceNet представляет конфигурирование сети и возможности настройки характеристик интеграционной модели коммуникационной сети DeviceNet.

Далее он разбивается на несколько классов, как показано на рисунке 3:

- объект NM-DeviceNet (NM-DeviceNetObject), объект NM-Соединение (NM-ConnectionObject) и Маршрутизатор NM-сообщений (NM-MessageRouter) определяют характеристики, связанные с управлением классов соответствующих объектов.

6.2 Протокол CANopen

6.2.1 Профиль устройств

6.2.1.1 Общие положения

На рисунке 4 показана структура классов профилей устройств CANopen.

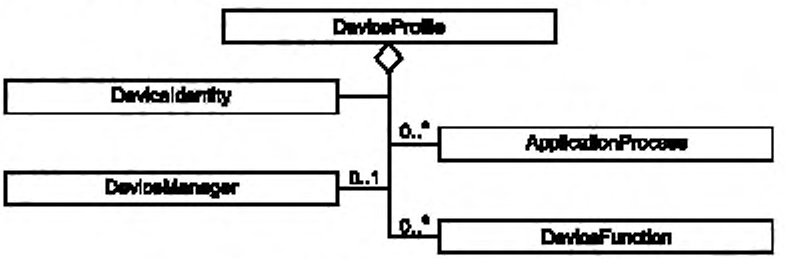


Рисунок 4 — Диаграмма классов профилей устройств CANopen

Требуемый формат профилей устройств CANopen описан в B.1 (приложение B). XML схема, представляющая шаблон профилей устройств CANopen, определена в B.1.5.1 (приложение B). Имя XML файла схемы должно быть 'COFDCML.xsd'.

Примечание 1 — Для лучшей читаемости диаграмма классов профилей устройств (DeviceProfile) CANopen разбита на пять диаграмм классов.

Примечание 2 — Все эти классы отображаются в той же XML схеме, определенной в B.1.5.1 (приложение B).

Примечание 3 — Диаграммы классов профилей устройств CANopen, показанные на рисунках 4—10, определяют основные классы. Далее некоторые классы разбиваются на составные части; подробное описание дано в приложении B.

6.2.1.2 Идентичность устройств

Класс Идентичность устройств (DeviceIdentity) определен на рисунке 5.

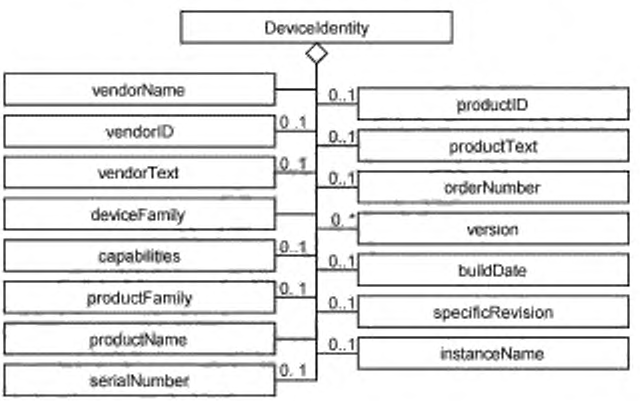


Рисунок 5 — Диаграмма класса Идентичность устройств (DeviceIdentity)

Класс DeviceIdentity должен состоять из дочерних классов, показанных на рисунке 5 и установленных в таблице 2.

Таблица 2 — Разбиение класса объекта Идентичность устройств

Класс	Описание	Профиль	Тип	Экземпляр
vendorName	Название изготовителя или продавца устройства	X	X	X
vendorID	IEEE OUI (Организационно Уникальный Идентификатор) (см.[6])		X	X
vendorText	Может быть использован для предоставления дополнительной информации по продавцу	X	X	X
deviceFamily	Определение этого класса в настоящем стандарте не установлено	X	X	X
capabilities	Определение этого класса в настоящем стандарте не установлено		X	X
productFamily	Специальное семейство продуктов (бренд) продавца устройства		X	X
productName	Специальное название продуктов продавца	X	X	X
productID	Уникальный идентификатор, определяющий тип устройства; формат выбирается по усмотрению продавца		X	X
productText	Может быть использован для предоставления дополнительной информации по устройству	X	X	X
orderNumber	Специальный порядковый номер продукта продавца		X	X
version	Специальная версия продукта продавца, атрибут versionType (Тип версии) позволяет различать множество версий (а именно: аппаратные средства, встроенные программы)		X	X
buildDate	Встроенная дата встроенных программ программного обеспечения, устанавливающего основные функциональные возможности устройства		X	X
specificationRevision	Внесение изменений в спецификацию, которой данное устройство соответствует	X	X	X
instanceName	Имя экземпляра устройства			X
serialNumber	Серийный номер экземпляра устройства			X

Примечание — В графах «Профиль», «Тип» и «Экземпляр» указывают, подходит ли отдельный дочерний класс для использования в профиле устройств, а также для описания типа устройства или описания экземпляра устройства.

6.2.1.3 Управление устройством

6.2.1.3.1 Общие положения

На рисунке 6 показано представление CANopen класса Управление устройством (DeviceManager).

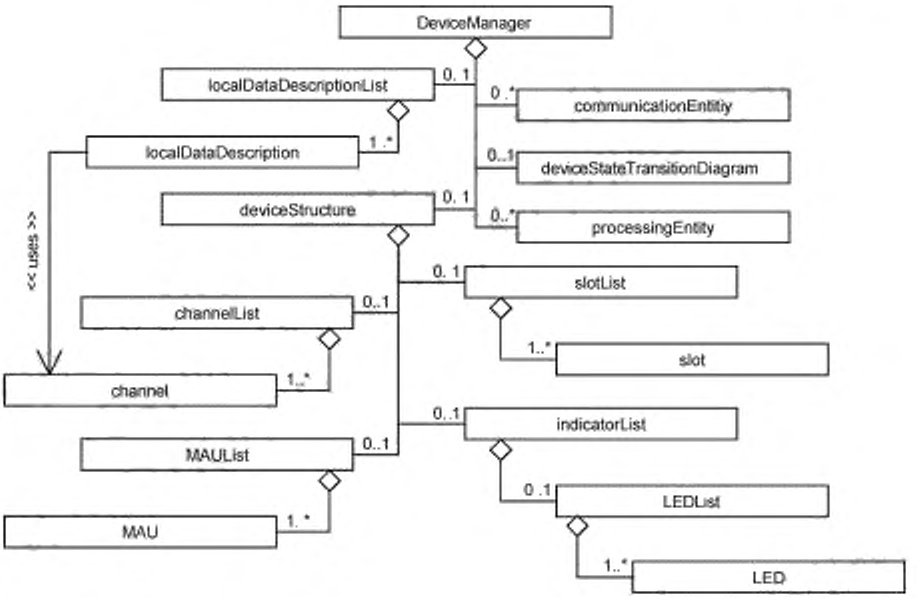


Рисунок 6 — Диаграмма класса Управление устройством (DeviceManager)

6.2.1.3.2 Объекты localDataDescriptionList (Список описаний локальных данных), localDataDescription (Описание локальных данных)

Объект localDataDescriptionList должен быть набором объектов localDataDescription. Объект localDataDescription должен описывать объекты данных, используемых только в контексте устройства.

6.2.1.3.3 Объект deviceStructure

6.2.1.3.3.1 Общее представление

Объект deviceStructure должен быть контейнером всех физических объектов устройства. Такой объект может быть каналом (физической или логической точкой ввода/вывода), MAU (Устройством подключения к среде), слотом для подключения дополнительных модулей (как часть устройства) или LED (светоизлучающим диодом).

6.2.1.3.3.2 Объекты channelList (Список каналов), channel (канал)

Объект channelList должен быть набором объектов channel. Эти объекты должны описывать физические или логические точки ввода/вывода устройства.

6.2.1.3.3.3 Объекты MAUList (Список устройств подключения к среде), MAU (Устройство подключения к среде)

Объект MAUList должен быть набором объектов MAU. Эти объекты должны описывать точки доступа к сетевым средам.

6.2.1.3.3.4 Объекты slotList (Список слотов), slot (слот)

Объект slotList должен быть набором объектов slot. Объект slot должен содержать ссылку на описание внешнего обмена профилями устройства CANopen.

Примечание — Слоты используют для описания модульных устройств или отдельных комбинаций устройств.

6.2.1.3.3.5 Объекты indicatorList (Список индикаторов), LEDList (Список светоизлучающих диодов), LED (светоизлучающий диод)

Объект LEDList должен быть набором объектов LED. Этот объект должен описывать LED устройства.

Примечание — Класс indicatorList может быть расширен в последующих изданиях ИСО 15745-2.

6.2.1.3.4 Класс communicationEntity (Коммуникационная сущность)

6.2.1.3.4.1 Общие положения

На рисунке 7 дано определение класса communicationEntity.

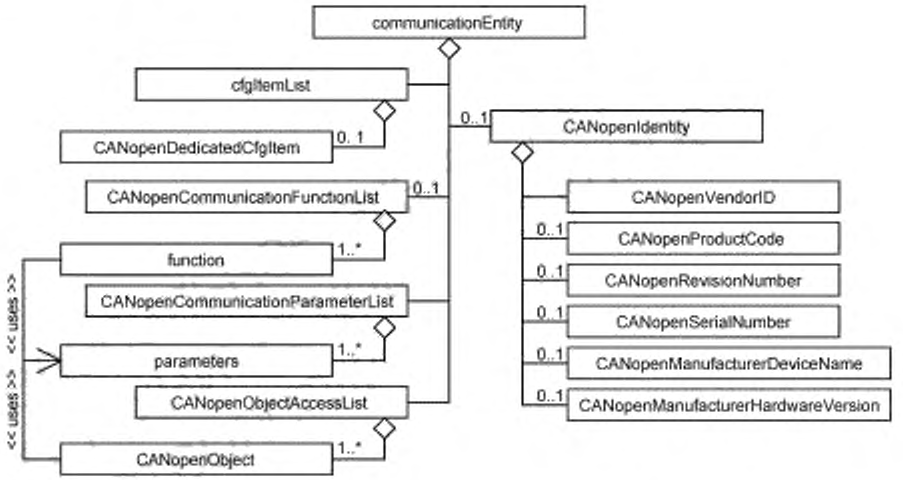


Рисунок 7 — Диаграмма класса communicationEntity

Класс communicationEntity должен описывать объект устройства, способный установить связь с объектами других устройств, и должен содержать полный набор заранее определенных элементов конфигурации и описания коммуникационных объектов. В устройстве могут использоваться несколько communicationEntity.

6.2.1.3.4.2 cfgItemList (Список элементов конфигурации)

cfgItemList должен состоять из объектов CANopenDedicatedCfgItem.

6.2.1.3.4.3 CANopenDedicatedCfgItem (специализированный элемент конфигурации CANopen)

CANopenDedicatedCfgItem должен быть набором элементов конфигурации.

Примечание — Определение дополнительных классов элементов конфигурации в ИСО 15745-2 не рассмотрено.

6.2.1.3.4.4 Класс CANopenIdentity (идентичность CANopen)

Класс CANopenIdentity состоит из нескольких объектов, необходимых для идентификации устройства в сети CANopen. Он включает в себя объекты для CANopenVendorID (идентификатор продавца CANopen), CANopenProductCode (код продуктов CANopen), CANopenRevisionNumber (номер модификации CANopen) и CANopenSerialNumber (серийный номер CANopen), а также CANopenManufacturerDeviceName (название прибора изготовителя CANopen), CANopenManufacturerHardwareVersion (версия аппаратных средств изготовителя CANopen) и CANopenManufacturerSoftwareVersion (версия программных средств изготовителя CANopen).

Примечание — Соответствующий объект идентичности определен в EH 50325-4.

6.2.1.3.4.5 CANopenCommunicationFunctionList (Список коммуникационных функций CANopen), function (функция)

CANopenCommunicationFunctionList должен быть набором объектов function (функция). Каждый такой объект описывает функциональные возможности CANopen, связанные с коммуникационной областью CANopen, путем использования CANopenCommunicationParameterList (Список коммуникационных параметров CANopen).

6.2.1.3.4.6 CANopenCommunicationParameterList (Список коммуникационных параметров CANopen), parameters (параметры)

CANopenCommunicationParameterList должен быть набором объектов Параметр. Каждый такой объект описывает параметр, связанный с коммуникационной областью CANopen.

6.2.1.3.4.7 CANopenObjectAccessList (Список доступа к объектам CANopen), CANopenObject (объект CANopen)

CANopenObjectAccessList должен быть набором объектов CANopenObject. Что касается словаря объектов CANopen, каждый такой объект описывает параметр из вида DeviceFunction или из CANopenCommunicationParameterList.

Примечание — CANopenObjectAccessList соответствует словарю объектов CANopen в EH 50325-4.

6.2.1.3.5 Класс processingEntity (обрабатываемая Сущность)

6.2.1.3.5.1 Общие положения

На рисунке 8 показано определение класса processingEntity.

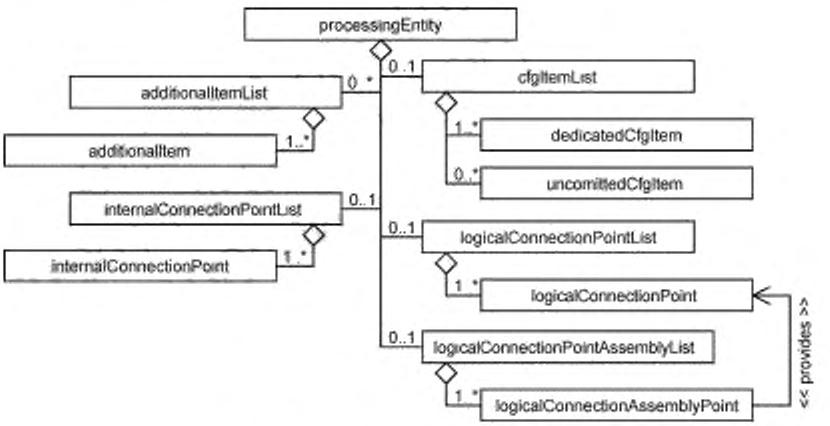


Рисунок 8 — Диаграмма класса processingEntity

Класс processingEntity должен описывать любое устройство, не являющееся коммуникационным объектом.

Пример — Ресурс, способный выполнять программы.

6.2.1.3.5.2 Классы additionalItemList (Список дополнительных элементов), additionalItem (дополнительный Элемент)

Класс additionalItemList должен быть набором объектов дополнительных элементов, определяемых пользователем. Объект additionalItem может быть использован для описания характеристик приборов, кроме характеристик конфигурации или коммуникационных объектов.

Примечание — Определение additionalItemType дополнительных элементов в настоящем стандарте не рассмотрено.

Пример — Документация по прибору.

6.2.1.3.5.3 Классы logicalConnectionPointList (Список логических точек соединений), logicalConnectionPoint (логическая точка соединения)

Класс logicalConnectionPointList должен быть набором объектов логических точек соединений. Этот объект описывает соединение.

Примечание — Предполагается, что используются только соединения между конечными точками соединений одинакового типа.

6.2.1.3.5.4 Классы logicalConnectionPointAssemblyList (Список наборов логических точек соединений), logicalConnectionPointAssembly (набор логических точек соединений)

Класс logicalConnectionPointAssemblyList должен быть набором объектов logicalConnectionPointAssembly. Этот набор должен быть описанием группы объектов logicalConnectionPoint.

6.2.1.3.5.5 Классы internalConnectionPointList (Список внутренних точек соединений), internalConnectionPoint (внутренняя точка соединения)

Класс internalConnectionPointList должен быть набором объектов internalConnectionPoint, определяющим внутренние соединения между различными объектами communicationEntity и/или объектами resourceEntity в одном и том же устройстве.

6.2.1.3.5.6 Класс cfgItemList (Список элементов конфигурации)

Класс cfgItemList может состоять из объектов dedicatedCfgItem и объектов uncomittedCfgItem.

6.2.1.3.5.7 Элемент dedicatedCfgItem (Специализированный элемент конфигурации)

dedicatedCfgItem должен быть элементом конфигурации с атрибутом dedicatedCfgItemType. Такой элемент следует использовать для определения соответствующих характеристик конфигурации.

6.2.1.3.5.8 Элемент uncommittedCfgItem (нескоммутированный элемент конфигурации)

uncommittedCfgItem должен быть элементом конфигурации с атрибутом dedicatedCfgItemType. Такой элемент следует использовать для определения характеристик конфигурации, которые не могут быть описаны dedicatedCfgItem.

Примечание — Определение некоммутированных элементов конфигурации в настоящем стандарте не рассмотрено.

Пример — Описание DIP — переключателей, изменяющих код идентификатора прибора.

6.2.1.4 Функция прибора

6.2.1.4.1 Общие положения

На рисунке 9 приведена диаграмма классов DeviceFunction.

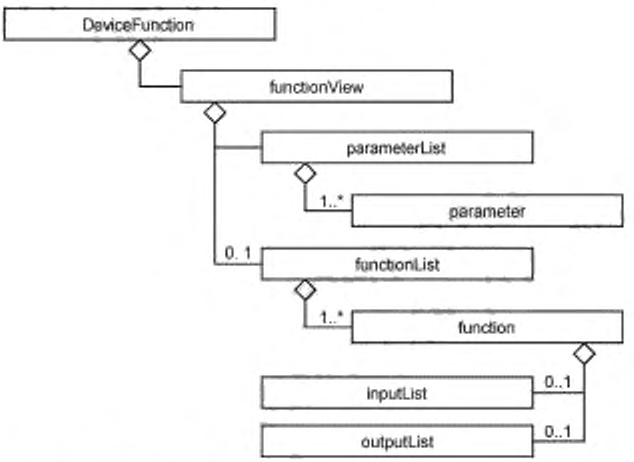


Рисунок 9 — Диаграмма классов DeviceFunction (Функция устройства)

В случае различных представлений функции устройства для описания DeviceFunction используют дополнительную XML схему. Имя XML файла схемы должно быть «FDCMLISO15745DeviceFunction.xsd». XML схема DeviceFunction определена в приложении В.

Примечание — Определение дополнительных XML схем, описывающих классы DeviceFunction, в настоящем стандарте не рассмотрено.

6.2.1.4.2 parameterList (Список параметров), parameter (параметр)

parameterList должен быть набором объектов Параметр. Такой объект описывает параметры устройства на основе функциональной перспективы. Он соединяется с коммуникационным объектом в communicationEntity.

6.2.1.4.3 functionList (Список функций), function (функция), inputsList (Список вводов), outputsList (Список выводов)

functionList должен быть набором объектов function (функция). Такой объект должен состоять из inputsList и outputsList. Эти списки должны содержать список ссылок на объекты Параметр.

6.2.1.5 Прикладной процесс

6.2.1.5.1 Общие положения

Объект ApplicationProcess (Прикладной процесс) может быть представлен одной или несколькими соответствующими XML схемами.

Примечание — Определение этих XML схем в настоящем стандарте не установлено.

6.2.1.5.2 textualDescription (Текстовое описание)

Объект textualDescription объясняет функцию устройства в читаемом текстовом виде.

6.2.2 Профиль коммуникационных сетей

6.2.2.1 Общие положения

На рисунке 10 показана диаграмма классов профиля коммуникационной сети CANopen.

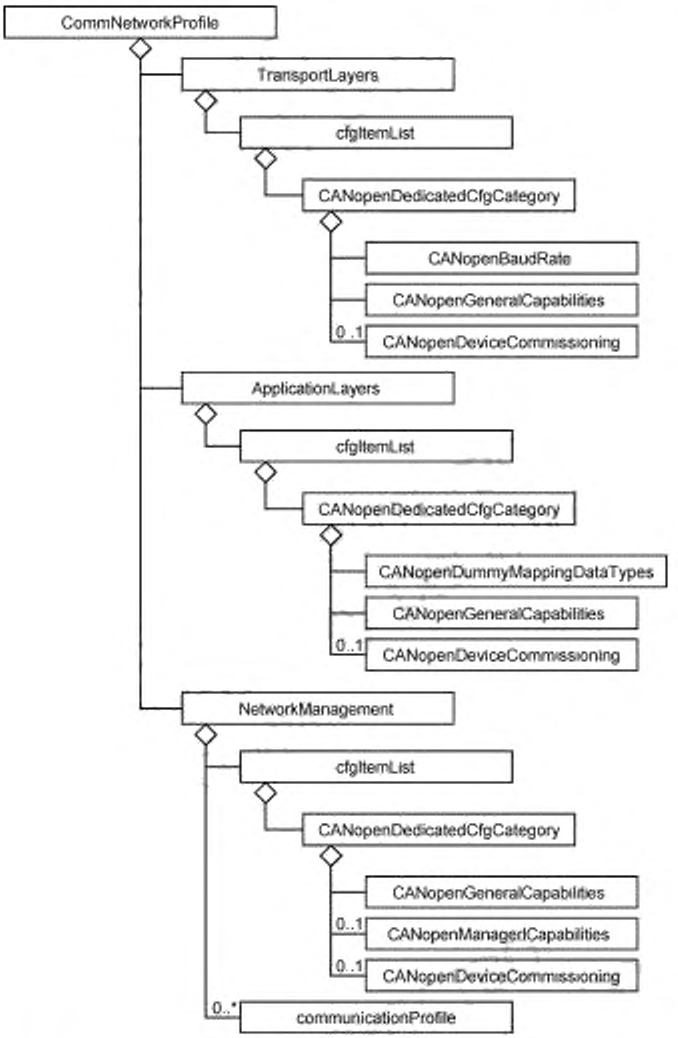


Рисунок 10 — Диаграмма классов профиля коммуникационной сети CANopen

XML схема, представляющая профиль коммуникационной сети CANopen, определена в приложении В. Имя файла XML схемы должно быть «COCommNetworkProfile.xsd».

6.2.2.2 communicationProfile (Коммуникационный профиль)

communicationProfile должен устанавливать подходящие идентификаторы коммуникационных профилей. Эти профили и их идентификаторы определены в МЭК 61784-1 (раздел 10.1). Разработчик AIP может установить дополнительные коммуникационные профили; идентификаторы для таких новых коммуникационных профилей должны быть трехразрядным числом, лежащим между 680 и 699.

6.2.2.3 Транспортные уровни

6.2.2.3.1 Общие положения

Объект TransportLayers (Транспортные уровни) должен представлять комбинированные профили для четырех нижних уровней интегрированной OSI модели коммуникационных сетей. Этот объект должен включать в себя объект cfgItemList (Список элементов конфигурации).

6.2.2.3.2 Объекты cfgItemList (Список элементов конфигурации), CANopenDedicatedCfgCategory (Специализированная категория конфигураций CANopen)

cfgItemList должен быть набором элементов конфигурации, относящимся к четырем нижним уровням интегрированной OSI модели коммуникационных сетей. Набор включает в себя специализированную категорию, предназначенную для связанных элементов конфигурации CANopen, например для скоростей передачи данных в бодах, общих возможностей, а также для ввода прибора в эксплуатацию.

Примечание — CANopenDedicatedCfgCategory описывает поддерживаемые услуги по передаче данных (например, объектов данных технологического процесса или объектов сервисных данных), а также поддерживаемые скорости передачи информации в бодах, как определено в ЕН 60325-4.

6.2.2.4 Прикладные уровни

6.2.2.4.1 Общие положения

Объект ApplicationLayers (Прикладные уровни) должен представлять комбинированные профили для трех уровней интегрированной OSI модели коммуникационных сетей. Объект ApplicationLayers должен содержать объект cfgItemList.

6.2.2.4.2 Объекты cfgItemList (Список элементов конфигурации), CANopenDedicatedCfgCategory (Специализированная категория конфигурации CANopen)

Объект cfgItemList должен быть набором элементов конфигурации, относящихся к трем уровням интегрированной OSI модели коммуникационных сетей. Набор включает в себя специализированную категорию, предназначенную для связанных элементов конфигурации CANopen, например для поддерживаемых типов данных для формального отображения, общих возможностей, а также для ввода прибора в эксплуатацию.

6.2.2.5 Управление сетями

6.2.2.5.1 Общие положения

Объект NetworkManagement (Управление сетями) должен представлять функциональные возможности управления сетями. Этот объект должен содержать объект cfgItemList.

6.2.2.5.2 Объекты cfgItemList (Список элементов конфигурации), CANopenDedicatedCfgCategory (Специализированная категория конфигураций CANopen)

Объект cfgItemList должен быть набором элементов конфигурации, предназначенных для связанных элементов конфигурации CANopen, например для возможностей администратора CANopen, общих возможностей, а также для ввода прибора в эксплуатацию.

Примечание — Объект CANopenDedicatedCfgCategory определяет поддерживаемые сервисы управления сетями, как установлено в ЕН 50325-4.

Приложение А
(обязательное)

Шаблоны профилей DeviceNet

А.1 Общие положения

Верхние уровни сети DeviceNet основаны на Общем Промышленном Протоколе (CIP). Этот протокол моделирует все коммуникационные и прикладные сущности как объекты. Передача специальных сообщений CIP требует выполнения сервисов на соответствующих экземплярах объекта (или на их атрибутах). Эта схема обеспечивает заданный доступ ко всем данным по конфигурации, состоянию и переменным рабочего цикла в узле. В то же самое время соединения ввода/вывода позволяют осуществлять прямой обмен с базой данных ввода/вывода без промежуточной обработки данных. В обоих случаях все адреса данных в устройстве устанавливаются с использованием маршрута CIP, т. е. октетного строкового потока, определяющего экземпляр прикладного объекта, атрибут и/или конечную точку соединения.

Для дистанционного конфигурирования устройств с коммуникационным интерфейсом CIP существуют разные опции, включающие в себя:

- информацию по приборам, сохраняемую в печатном или электронном формате;
- специализированные объекты Parameter Object, которые предоставляют заданный открытый интерфейс для отдельных значений данных по конфигурации/параметрам и также могут вводить дополнительную информацию по конфигурации, например описательный текст, тип данных, границы данных и данные по умолчанию;
- специализированную компоновку конфигурации, которая позволяет загружать и скачивать массивы данных по конфигурации путем группирования отдельных значений данных;
- комбинации методов, перечисленных выше.

Средства конфигурирования, доступные в настоящее время для устройств на основе CIP, используют специально отформатированный файл ASCII, называемый Электронная Информационная Таблица (EDS), который предоставляет:

- информацию, необходимую для идентификации подключенного устройства;
- описание данных по устройству, которые могут быть оценены через сеть (например, конфигурируемые параметры);
- описание коммуникационных возможностей, поддерживаемых устройством (например, соединения);
- дополнительную информацию от продавца.

EDS позволяет средствам конфигурирования автоматизировать процесс конфигурирования устройств. Требования EDS определяют открытый, согласованный и совместимый метод, позволяющий выполнить конфигурирование устройств в среде CIP.

Информация EDS во многом аналогична информации, необходимой как для профиля коммуникационной сети, так и для профиля устройств, поэтому следующие подклассы устанавливают формат для:

- шаблонов профилей коммуникационной сети и приборов, как определено в ИСО 15745-1;
- инкапсуляции унаследованных файлов EDS в шаблоны ИСО 15745 («упаковщики»);
- унаследованной Электронной Информационной Таблицы (EDS), включая общую семантическую информацию.

Примечание — EDS сети DeviceNET заданного прибора может быть получена из содержимого соответствующих XML файлов профилей приборов и коммуникационных сетей с использованием соответствующих таблиц стилей.

А.2 Описание шаблонов профилей устройств

А.2.1 Описание шаблонов профилей устройств на основе XML

А.2.1.1 Общие положения

Файлы XML профилей приборов должны соответствовать XML схеме профилей устройств, как установлено в А.2.1.3.3.

Содержимое XML схемы выводится из диаграмм классов профилей устройств, показанной в 6.1.1 и расширенной путем введения дополнительных элементов, обеспечивающих полное описание требований к приборам или возможностей устройств.

А.2.1.2 Семантика элементов XML схемы

А.2.1.2.1 ProfileBody (Тело профиля)

Данный основной элемент связан с набором атрибутов, предоставляющих дополнительную информацию о файле профилей.

Семантика этих атрибутов установлена в А.4.1.4.2.

А.2.1.2.2 DeviceIdentity (Идентичность устройств)

Данный элемент определяет поддерживаемые атрибуты экземпляра и операции объекта Идентичность (см. МЭК 62026-3), а также дополнительную информацию для полной идентификации устройств. В случае необходимости этот элемент также указывает истинные значения атрибутов экземпляра.

Семантика подэлементов DeviceIdentity_InstanceAttributes элемента DeviceIdentity установлена в таблице А.1.

Таблица А.1 — Элементы DeviceIdentity_InstanceAttributes

Элементы XML схемы	Атрибуты объекта	Семантика
SpecificationConformance	Нет	Строка, устанавливающая эталонную версию спецификаций DeviceNet
VendCode, ProdType, ProdCode, ProdRevision	Есть	А.4.1.4.3
VendName, ProdTypeStr, ProdName, Catalog, Icon, Exclude FromAdapterRackConnection	Нет	А.4.1.4.3
Status, SerialNumber, State, ConfigurationConsistencyValue, HeartbeatInterval	Есть	Не применяется
DeviceClassification	Нет	А.4.1.4.4 и А.4.2.2.1

А.2.1.2.3 DeviceManager (Управление устройствами)

Данный элемент определяет поддерживаемые атрибуты классов и операции объекта Identity (Идентичность) (см. МЭК 62026-3), а также дополнительную информацию для управления устройствами. В случае необходимости этот элемент также указывает истинные значения атрибутов экземпляра.

Семантика модульных подэлементов элемента DeviceManager установлена в А.4.1.5.2.

А.2.1.2.4 DeviceFunction (Функция устройств)

В настоящем стандарте содержимое этого элемента не детализировано.

А.2.1.2.5 ApplicationProcess (Прикладной процесс)

А.2.1.2.5.1 Assembly (Компоновка)

Данный элемент устанавливает поддерживаемые атрибуты классов и экземпляров объекта Assembly [см. МЭК 61158-5 и МЭК 61158-6 (тип 2)], а также описание отдельных экземпляров.

Семантика подэлементов Assem, ProxyAssem and ProxiedAssem элемента DeviceManager установлена в А.4.1.4.8 и А.4.1.5.3.2.

А.2.1.2.5.2 Parameter (Параметр)

Данный элемент устанавливает поддерживаемые атрибуты классов и экземпляров и операции объекта Parameter Object, а также описание отдельных экземпляров.

Семантика подэлемента Parameter_ClassAttributes элемента Parameter установлена в А.4.1.4.5.

Семантика подэлементов Param, ProxyParam and ProxiedParam элемента Parameter установлена в А.4.1.4.6 и А.4.1.5.3.1.

А.2.1.2.5.3 ParameterGroup (Группа Параметров)

Данный элемент устанавливает группы связанных параметров для конкретной цели.

Семантика подэлемента Group элемента ParameterGroup установлена в А.4.1.4.7.

А.2.1.3 XML схемы

А.2.1.3.1 Файл MasterTemplateTypes.xsd

Примечание — XML схема содержит все стили, определенные как часть основного шаблона в ИСО 15745-1.

```
<?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: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: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>
```

А.2.1.3.2 Файл CIPDataTypes.xsd

Примечание — XML схема определяет элементы, составляющие ее (например, типы данных, типы элементов, группы атрибутов), используемые в других XML схемах.

```
<?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="gost.e-ecolog.ru
<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 Файл CIP_Device_Profile.xsd

Примечание — XML схема включает в себя файлы «Master Template Types.xsd» (см. A.2.1.3.1) и «CIPDataTypes.xsd» (см. A.2.1.3.2).

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:xsd="gost.e-ecolog.ru                    " 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: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: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: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: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: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 Описание шаблонов профилей устройств — инкапсуляция XML файлов EDS

A.2.2.1 Общие положения

Файлы XML профилей устройств, используемые для инкапсуляции файлов EDS, должны соответствовать XML схеме профилей устройств, как установлено в А.2.2.2.

Семантика подэлементов элемента ExternalProfileHandle, используемого для обращения к имеющемуся файлу EDS, установлена в таблице А.2. В зависимости от значения атрибута WrapperReference к файлу EDS можно обратиться, используя элементы идентификации либо из самого файла EDS, либо из продукта, описанного этой EDS.

Примечание 1 — Выбор соответствующих элементов идентификации зависит от предполагаемого использования упаковочного (wrapper) файла.

Таблица А.2 — Элементы ExternalProfileHandle (Внешний Идентификатор Профиля)

Элементы XML схемы	WrapperReference = FILEINFO	WrapperReference = DEVICEINFO
ProfileIdentification	Текстовое описание файла EDS[a]	VendorID (идентификация вендора), Device Type (тип устройства), Product Code (код продукта) [b]
ProfileRevision	Модификация EDS[a]	Модификация продукта [b]
ProfileLocation	Собственный адрес (URL) EDS[a]	Имя файла, содержащего изображение иконки[a]

[a] Для получения более полной информации см. А.4.1.4.2.
[b] Для получения более полной информации см. А.4.1.4.3.

Элементы DeviceIdentity (Идентичность устройств), DeviceManager (Управление устройствами), DeviceFunction (Функция устройств) и ApplicationProcess (Прикладной процесс) должны быть совместимы с форматами, установленными в А.2.1.3.3.

Примечание 2 — Они могут быть использованы в течение перехода от унаследованного формата EDS к полному формату XML.

А.2.2.2 XML схема: файл EDS_Device_Profile_wrapper.xsd

Примечание — XML схема включает в себя файл «MasterTemplateTypes.xsd» (см. А.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>
```

А.3 Описание шаблонов профилей коммуникационных сетей

А.3.1 Описание шаблонов профилей коммуникационных сетей на основе XML

А.3.1.1 Общие положения

Файлы XML профилей коммуникационных сетей должны соответствовать XML схеме профилей коммуникационных сетей, как установлено в А.3.1.3.

Содержимое этой XML схемы выводится из диаграмм классов коммуникационных сетей, показанных в 6.1.2, и расширяется путем использования дополнительных элементов, что позволяет полностью описать требования к коммуникационным сетям или их возможности.

А.3.1.2 Семантика элементов XML схемы

А.3.1.2.1 ProfileBody (Тело профиля)

Данный основной элемент связан с набором атрибутов, которые предоставляют дополнительную информацию о файле профиля.

Семантика этих атрибутов установлена в А.4.1.4.2.

А.3.1.2.2 ApplicationLayers (Прикладные уровни)

А.3.1.2.2.1 ConnectionObject (Объект соединения)

Данный элемент устанавливает поддерживаемые атрибуты экземпляров и операции Объекта соединения (см. МЭК 62026-3), а также описание отдельных экземпляров соединений и характеристики ввода/вывода.

Элемент ConnectionDescriptions (Описание соединений) идентифицирует экземпляры Объекта Соединения, которые осуществляются в устройстве. Семантика подэлемента Connection (Соединение) установлена в таблице А.3.

Таблица А.3 — Элементы соединений

Элементы XML схемы	Семантика
InstanceType	Устанавливает тип соединения (определенное сообщение, ввод/вывод по запросу, побитовый стробированный ввод/вывод, динамический ввод/вывод)
ProductionTrigger	Устанавливает тип триггера (циклический, изменение состояния, запускаемый приложением)
TransportType	Устанавливает тип транспорта (сервер, клиент)
TransportClass	Устанавливает класс транспорта (0, 2 или 3)
ProducedConnectionSize	Устанавливает размер созданных данных в байтах
ConsumedConnectionSize	Устанавливает размер потребленных данных в байтах

Семантика элемента IO_Characteristics установлена в А.4.2.4.2.

А.3.1.2.2.2 MessageRouter (Маршрутизатор сообщений)

Данный элемент устанавливает поддерживаемые атрибуты экземпляров и операции Объекта Message Router (Маршрутизатор сообщений) (см. МЭК 62026-3).

А.3.1.2.3 TransportLayers (Транспортные уровни)

А.3.1.2.3.1 DNPhysicalLayer (Физический уровень DN)

Данный элемент идентифицирует характеристики физических уровней (например, коннекторы, скорости передачи данных в бодах, электрические характеристики).

Семантика его подэлементов установлена в таблице А.4.

Таблица А.4 — Элементы DNPhysicalLayer

Элементы XML схемы	Атрибут XML схемы	Семантика
Коннекторы	ConnectorType	Устанавливает тип коннекторов (открытая жесткая разводка, открытые съемные, герметизированные Мини, герметизированные Микро)
Светоизлучающие диоды (СИД)		Устанавливает поддерживаемые индикаторы LED
	Module	Если атрибут присутствует, то присутствует соответствующий LED
	Network	
	Combo_ModNet	
	IO	
Baud_Rate_Setting		Устанавливает поддерживаемые скорости передачи данных в бодах
	SwitchType	Устанавливает поддерживаемые методы установки выбранной скорости передачи данных в бодах
	SoftwareSettable	Определяет, может ли скорость передачи данных в бодах устанавливаться программными средствами
Electrica ICharacteristics		Подэлемент устанавливает максимальную потребляемую мощность сети в амперах при напряжении постоянного тока 11 В (наихудший случай)
	Isolated Physical Layer	Определяет, изолирован ли физический уровень

А.3.1.2.3.2 DNLinkLayer (Канальный уровень DN)

Данный элемент определяет некоторые характеристики, связанные с конфигурацией уровней каналов передачи данных.

Семантика его подэлементов установлена в таблице А.5.

Таблица А.5 — Элементы DNLinkLayer

Элементы XML схемы	Атрибут XML схемы	Семантика
Mac-IDSetting		Устанавливает MAC-ID (идентификатор контроля доступа к сети) по умолчанию
	SwitchType	Определяет поддерживаемые методы для установки MAC-ID
	SoftwareSettable	Определяет, можно ли установить MAC-ID с использованием программных средств
PredefinedMasterSlaveConnectionSet		Устанавливает поддерживаемый заранее определенный набор соединений (ни одно из общего числа, Клиент Группы 2 и/или Сервер Группы 2, Клиент только Группы 2, Сервер только Группы 2)
DynamicConnectionSupport		Устанавливает поддержку динамических соединений (ни одно из общего числа, Группа 1, Группа 2, Группа 3)
Fragmerited Explicit Messaging		Подэлементы устанавливают параметры для фрагментированной передачи сообщений
	Supported	Определяет, поддерживается ли фрагментированная передача сообщений

А.3.1.2.3.3 Элемент DeviceNetObject

Данный элемент устанавливает поддерживаемые атрибуты экземпляров и операции Объекта DeviceNet (см. МЭК 62026-3).

А.3.1.2.3.4 Порты (Ports)

Данный элемент идентифицирует порты устройств, которые способны маршрутизировать сообщения из одного канала связи в другой канал связи.

Семантика подэлемента Port элемента Ports установлена в А.4.1.4.10 и А.4.2.2.2.

А.3.1.2.4.1 NetworkManagement (Управление сетями)

Данный элемент устанавливает поддерживаемые атрибуты классов и операции Объекта DeviceNet (см. МЭК 62026-3).

А.3.1.2.4.2 Элемент NM-ConnectionObject (Объект NM-соединения)

Данный элемент устанавливает поддерживаемые атрибуты классов и операции объекта Connection (Соединение) (см. МЭК 62026-3).

А.3.1.2.4.3 Элемент NM-MessageRouter (Маршрутизатор NM-сообщений)

Данный элемент устанавливает поддерживаемые атрибуты классов и операции Объекта Message Router (Маршрутизатор сообщений) (см. МЭК 62026-3).

А.3.1.3 XML схема: файл DNet_CommNet_Profile.xsd

Примечание — XML схема включает в себя файлы «MasterTemplateTypes.xsd» (см. А.2.1.3.1) и «CIPDataTypes.xsd» (см. А.2.1.3.2).

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:xsd="gost.e-ecolog.ru                    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: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">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="Connection" minOccurs="0" maxOccurs="unbounded">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="InstanceType">
<xsd:simpleType>
<xsd:restriction base="xsd:NMTOKEN">
<xsd:enumeration value="ExplicitMessage"/>
<xsd:enumeration value="PolledIO"/>
<xsd:enumeration value="BitStrobedIO"/>
<xsd:enumeration value="DynamicIO"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:element>
<xsd:element name="ProductionTrigger">
<xsd:simpleType>
<xsd:restriction base="xsd:NMTOKEN">
<xsd:enumeration value="Cyclic"/>
<xsd:enumeration value="ChangeOfState"/>
<xsd:enumeration value="ApplicationTriggered"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:element>
<xsd:element name="TransportType">
<xsd:simpleType>
<xsd:restriction base="xsd:NMTOKEN">
<xsd:enumeration value="Server"/>
<xsd:enumeration value="Client"/>
</xsd:restriction>
```

```
</xsd:simpleType>
</xsd:element>
<xsd:element name="TransportClass">
<xsd:simpleType>
<xsd:restriction base="dt_SINT">
<xsd:enumeration value="0"/>
<xsd:enumeration value="2"/>
<xsd:enumeration value="3"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:element>
<xsd:element name="ProducedConnectionSize" type="dt_UINT"/>
<xsd:element name="ConsumedConnectionSize" type="dt_UINT"/>
<xsd:any namespace="##any" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="MessageRouter">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="MessageRouter_InstanceAttributes" minOccurs="0">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="Object_List">
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base="xsd:string">
<xsd:attribute name="Access_Rule" type="at_AccessType_OptionalGet"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="MaximumConnectionSupported">
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base="dt_UINT">
<xsd:attribute name="Access_Rule" type="at_AccessType_OptionalGet"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="NumberActiveConnections">
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base="dt_UINT">
<xsd:attribute name="Access_Rule" type="at_AccessType_OptionalGet"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="ActiveConnectionList">
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base="dt_UINT">
<xsd:attribute name="Access_Rule" type="at_AccessType_OptionalGet"/>
</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="MessageRouter_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"/>
</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="DNPhysicalLayer">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="Connectors" minOccurs="0">
<xsd:complexType>
<xsd:attribute name="ConnectorType">
<xsd:simpleType>
<xsd:restriction base="xsd:NMTOKEN">
<xsd:enumeration value="OpenHardWired"/>
<xsd:enumeration value="OpenPluggable"/>
<xsd:enumeration value="SealedMini"/>
<xsd:enumeration value="SealedMicro"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
<xsd:element name="LEDs" minOccurs="0">
<xsd:complexType>
<xsd:attribute name="Module" type="xsd:NMTOKEN"/>
<xsd:attribute name="Network" type="xsd:NMTOKEN"/>
<xsd:attribute name="Combo_ModNet" type="xsd:NMTOKEN"/>
<xsd:attribute name="IO" type="xsd:NMTOKEN"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="Baud_Rate_Setting" minOccurs="0">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="BR_125Kbits" minOccurs="0">
<xsd:complexType>
<xsd:attribute ref="Supported"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="BR_250Kbits" minOccurs="0">
<xsd:complexType>
<xsd:attribute ref="Supported"/>
</xsd:complexType>
```

```
</xsd:element>
<xsd:element name="BR_500Kbits" minOccurs="0">
<xsd:complexType>
<xsd:attribute ref="Supported"/>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
<xsd:attribute name="SwitchType" use="required">
<xsd:simpleType>
<xsd:restriction base="xsd:NMTOKEN">
<xsd:enumeration value="DipSwitch"/>
<xsd:enumeration value="Other"/>
<xsd:enumeration value="SoftwareOnly"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
<xsd:attribute name="SoftwareSettable" use="required">
<xsd:simpleType>
<xsd:restriction base="xsd:boolean">
<xsd:pattern value="true"/>
<xsd:pattern value="false"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
<xsd:element name="ElectricalCharacteristics" minOccurs="0">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="MaxNetworkPowerConsumption" type="dt_REAL"/>
</xsd:sequence>
<xsd:attribute name="IsolatedPhysicalLayer">
<xsd:simpleType>
<xsd:restriction base="xsd:boolean">
<xsd:pattern value="true"/>
<xsd:pattern value="false"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
<xsd:element name="Other" type="xsd:string" minOccurs="0"/>
<xsd:any namespace="##any" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="DNLinkLayer">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="MAC-IDSetting" minOccurs="0">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="DefaultMAC-ID" type="et_MAC-IDRange" minOccurs="0"/>
</xsd:sequence>
<xsd:attribute name="SwitchType" use="required">
<xsd:simpleType>
<xsd:restriction base="xsd:NMTOKEN">
<xsd:enumeration value="DipSwitch"/>
<xsd:enumeration value="Other"/>
<xsd:enumeration value="SoftwareOnly"/>
</xsd:restriction>
</xsd:simpleType>
```

```
</xsd:attribute>
<xsd:attribute name="SoftwareSettable" use="required">
<xsd:simpleType>
<xsd:restriction base="xsd:boolean">
<xsd:pattern value="true"/>
<xsd:pattern value="false"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
<xsd:element name="PredefinedMasterSlaveConnectionSet" minOccurs="0">
<xsd:simpleType>
<xsd:restriction base="xsd:NMTOKEN">
<xsd:enumeration value="NotSupported"/>
<xsd:enumeration value="Group2Client"/>
<xsd:enumeration value="Group2Server"/>
<xsd:enumeration value="Group2ClientServer"/>
<xsd:enumeration value="GroupOnly2Client"/>
<xsd:enumeration value="GroupOnly2Server"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:element>
<xsd:element name="DynamicConnectionSupport" minOccurs="0">
<xsd:simpleType>
<xsd:restriction base="xsd:NMTOKEN">
<xsd:enumeration value="NotSupported"/>
<xsd:enumeration value="Group1"/>
<xsd:enumeration value="Group2"/>
<xsd:enumeration value="Group3"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:element>
<xsd:element name="FragmentedExplicitMessaging" minOccurs="0">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="TransmissionTimeout" type="dt_UINT" minOccurs="0"/>
<xsd:element name="TypicalTargetAddress" minOccurs="0">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="Class" type="dt_USINT" minOccurs="0"/>
<xsd:element name="Instance" type="dt_USINT" minOccurs="0"/>
<xsd:element name="Attribute" type="dt_USINT" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
<xsd:attribute ref="Supported"/>
</xsd:complexType>
</xsd:element>
<xsd:any namespace="##any" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="DeviceNetObject">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="DeviceNetObject_InstanceAttributes" minOccurs="0">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="MAC_ID">
<xsd:complexType>
```

```
<xsd:attribute name="Access_Rule" type="at_AccessType_OptionalSet"
use="required"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="Baud_Rate">
<xsd:complexType>
<xsd:attribute name="Access_Rule" type="at_AccessType_OptionalSet"
use="required"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="BOI">
<xsd:complexType>
<xsd:attribute name="Access_Rule" type="at_AccessType_OptionalSet"
use="required"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="Bus_Off_Counter">
<xsd:complexType>
<xsd:attribute name="Access_Rule" type="at_AccessType_OptionalSet"
use="required"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="Allocation_Information">
<xsd:complexType>
<xsd:attribute name="Access_Rule" type="at_AccessType_OptionalGet"
use="required"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="MAC_ID_Switch_Changed">
<xsd:complexType>
<xsd:attribute name="Access_Rule" type="at_AccessType_OptionalGet"
use="required"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="Baud_Rate_Switch_Changed">
<xsd:complexType>
<xsd:attribute name="Access_Rule" type="at_AccessType_OptionalGet"
use="required"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="MAC_ID_Switch_Value">
<xsd:complexType>
<xsd:attribute name="Access_Rule" type="at_AccessType_OptionalGet"
use="required"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="Baud_Rate_Switch_Value">
<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="DeviceNetObject_InstanceOperations" minOccurs="0">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="Get_Attribute_Single">
<xsd:complexType>
```

```
<xsd:attribute ref="SupportedService"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="Set_Attribute_Single">
<xsd:complexType>
<xsd:attribute ref="SupportedService"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="Allocate_MS_Connection_Set">
<xsd:complexType>
<xsd:attribute ref="SupportedService"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="Release_Group2_Identifier_Set">
<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="Ports">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="Port" minOccurs="0" maxOccurs="unbounded">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="PortTypeName">
<xsd:simpleType>
<xsd:union>
<xsd:simpleType>
<xsd:restriction base="xsd:NMTOKEN">
<xsd:enumeration value="ControlNet"/>
<xsd:enumeration value="ControlNet_Redundant"/>
<xsd:enumeration value="TCP"/>
<xsd:enumeration value="DeviceNet"/>
</xsd:restriction>
</xsd:simpleType>
<xsd:simpleType>
<xsd:restriction base="et_VendorSpecificKeyword"/>
</xsd:simpleType>
</xsd:union>
</xsd:simpleType>
</xsd:element>
<xsd:element name="PortName" type="dt_EDS_Char_Array" minOccurs="0"/>
<xsd:element name="PortObject" type="dt_EPATH" minOccurs="0"/>
<xsd:element name="PortNumber" type="dt_UINT"/>
<xsd:element name="PortSpecific">
<xsd:complexType>
<xsd:sequence>
<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:attribute name="id" use="required">
<xsd:simpleType>
```

```
<xsd:restriction base="xsd:ID">
<xsd:pattern value="Port[1-9][0-9]{0,4}"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="NM-DeviceNetObject">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="DeviceNetObject_ClassAttributes" minOccurs="0">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="ObjectRevision">
<xsd:complexType>
<xsd:attribute name="Access_Rule" type="at_AccessType_Mandatory" fixed="Get"/>
</xsd:complexType>
</xsd:element>
<xsd:any namespace="##any" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="DeviceNetObject_ClassOperations" minOccurs="0">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="Get_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:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="NM-ConnectionObject">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="ConnectionObject_ClassAttributes" minOccurs="0">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="ObjectRevision">
<xsd:complexType>
<xsd:attribute name="Access_Rule" type="at_AccessType_OptionalGet"/>
</xsd:complexType>
</xsd:element>
<xsd:any namespace="##any" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="ConnectionObject_ClassOperations" minOccurs="0">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="Reset">
<xsd:complexType>
<xsd:attribute ref="SupportedService"/>
</xsd:complexType>
```

```
</xsd:element>
<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"/>
</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:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="NM-MessageRouter">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="MessageRouter_ClassAttributes" minOccurs="0">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="ObjectRevision">
<xsd:complexType>
<xsd:attribute name="Access_Rule" type="at_AccessType_OptionalGet"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="OptionalAttributeList">
<xsd:complexType>
<xsd:attribute name="Access_Rule" type="at_AccessType_OptionalGet"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="OptionalServiceList">
<xsd:complexType>
<xsd:attribute name="Access_Rule" type="at_AccessType_OptionalGet"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="MaxIDClassAttributes">
<xsd:complexType>
<xsd:attribute name="Access_Rule" type="at_AccessType_OptionalGet"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="MaxIDInstanceAttributes">
<xsd:complexType>
<xsd:attribute name="Access_Rule" type="at_AccessType_OptionalGet"/>
</xsd:complexType>
</xsd:element>
<xsd:any namespace="##any" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
```

```
</xsd:element>
<xsd:element name="MessageRouter_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="Get_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: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:attribute name="Supported" use="required">
<xsd:simpleType>
<xsd:restriction base="xsd:boolean">
<xsd:pattern value="true|false"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
<xsd:simpleType name="et_MAC-IDRange">
<xsd:restriction base="xsd:nonNegativeInteger">
<xsd:maxInclusive value="63"/>
</xsd:restriction>
</xsd:simpleType>
<xsd:complexType name="et_SpecificIOInfoType">
<xsd:sequence>
<xsd:element name="CompatibleIOTypeMask" type="dt_WORD"/>
<xsd:element name="DefaultProducingConnection" type="dt_UINT"/>
<xsd:element name="DefaultConsumingConnection" type="dt_UINT"/>
<xsd:any namespace="##any" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
<xsd:complexType name="et_GenericIOInfoType">
<xsd:sequence>
<xsd:element name="Size" type="dt_UINT"/>
<xsd:element name="NumberSignificantBits" type="dt_UINT"/>
<xsd:element name="CompatibleIOTypeMask" type="dt_WORD"/>
<xsd:element name="Name_String" type="dt_EDS_Char_Array"/>
<xsd:element name="ConnectionPathSize" type="dt_UINT"/>
<xsd:element name="Path" type="dt_EPATH"/>
<xsd:element name="Help_String" type="dt_EDS_Char_Array"/>
<xsd:any namespace="##any" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
<xsd:complexType name="et_IOInfoType">
```

```
<xsd:sequence>
<xsd:element name="Default">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="DefaultIOTypeMask" type="dt_WORD"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="PollInfo" type="et_SpecificIOInfoType" minOccurs="0"/>
<xsd:element name="StrobeInfo" type="et_SpecificIOInfoType" minOccurs="0"/>
<xsd:element name="MulticastPollInfo" type="et_SpecificIOInfoType" minOccurs="0"/>
<xsd:element name="COSInfo" type="et_SpecificIOInfoType" minOccurs="0"/>
<xsd:element name="CyclicInfo" type="et_SpecificIOInfoType" minOccurs="0"/>
<xsd:element name="Input" minOccurs="0" maxOccurs="unbounded">
<xsd:complexType>
<xsd:complexContent>
<xsd:extension base="et_GenericIOInfoType">
<xsd:attribute name="id" use="required">
<xsd:simpleType>
<xsd:restriction base="xsd:ID">
<xsd:pattern value="Input[1-9][0-9]{0,4}"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:extension>
</xsd:complexContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="Output" minOccurs="0" maxOccurs="unbounded">
<xsd:complexType>
<xsd:complexContent>
<xsd:extension base="et_GenericIOInfoType">
<xsd:attribute name="id" use="required">
<xsd:simpleType>
<xsd:restriction base="xsd:ID»>
<xsd:pattern value="Output[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:schema>
```

A.3.2 Описание шаблонов профилей коммуникационных сетей. XML инкапсуляция файлов EDS

A.3.2.1 Общие положения

Файлы XML профилей коммуникационных сетей, используемые для инкапсуляции файлов EDS, должны соответствовать XML схеме профилей коммуникационных сетей, как установлено в A.3.2.2.

Семантика подэлементов элемента ExternalProfileHandle, используемого для обращения к имеющемуся файлу EDS, установлена в таблице A.2. В зависимости от значения атрибута WrapperReference к файлу EDS можно обратиться, используя элементы идентификации либо из самого файла EDS, либо из продукта, описанного этой EDS.

Примечание — Выбор соответствующих элементов идентификации зависит от предполагаемого использования файла wrapper.

A.3.2.2 XML схема: файл EDS_CommNet_Profile_wrapper.xsd (см. A.2.1.3.1)

Примечание — XML схема включает в себя файл «MasterTemplateTypes.xsd» (см. 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="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 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>
```

А.4 Электронная информационная таблица EDS

А.4.1 Общие требования к EDS на основе CIP

А.4.1.1 Общие положения

Данный подраздел устанавливает требования к кодированию электронной информационной таблицы (EDS), которые являются общими для сетей на основе CIP. Указанные требования определяют стандартный формат кодирования файлов, используемых для продуктов CIP, без учета главной платформы средств конфигурирования или файловой системы.

Термин «файл», используемый в данном контексте, относится к любому общепринятому формату файла, связанного с файловой системой средств конфигурирования без учета средств для хранения файлов.

Файл EDS определяется как файл ASCII, который включает представление ASCII объектов в устройстве, которые могут быть оценены из сети [например, Parameter (Параметр) и Assembly (Компоновка)], и дополнительную информацию, необходимую для поддержки адресации объектов.

А.4.1.2 Содержимое EDS

А.4.1.2.1 Структура EDS

Один файл должен содержать всю EDS. EDS должна состоять из разделов. Таблица А.6 описывает структуру разделов, которые являются общими для нескольких сетей на основе CIP, соответствующие разрешенные разделители разделов и порядок следования этих разделов в EDS.

Таблица А.6 — Структура файла EDS CIP

Название раздела	Разрешенный разделитель	Расположение	Обязательно/ Дополнительно
Описание файла	[File]	1	Обязательно
Описание устройства	[Device]	2	Обязательно
Классификация устройства	[Device Classification]	a)	Дополнительно
Класс параметров	[ParamClass]	a)	Дополнительно
Параметры	[Params]	a)	Дополнительно
Группы параметров	[Groups]	a)	Дополнительно
Компоновка	[Assembly]	a)	Дополнительно
Характеристики соединений	[Connection Manager]	a)	Дополнительно
Порт	[Port]	a)	Дополнительно
Модульный	[Modular]	a)	Дополнительно
Со стороны продавца	[Vendor ID_vendorspecifickeyword]	Последний	Дополнительно
a) Эти дополнительные группы должны быть расположены после обязательных групп.			

Содержимое электронной информационной таблицы (EDS) должно быть организовано следующим образом:

- все файлы EDS должны содержать раздел Описание файла, который должен быть первым разделом файла EDS и должен использовать разрешенный разделитель [File];
- все файлы EDS должны содержать раздел Описание устройства, который должен сразу же следовать за разделом Описание файла и должен использовать разрешенный разделитель [Device];
- дополнительные разделы, описанные в данной спецификации, могут быть представлены в любом порядке при условии отсутствия в файле EDS ссылки вперед;

- дополнительный раздел (дополнительные разделы), относящийся (относящиеся) к продавцу, должен (должны) использовать разрешенный разделитель (разрешенные разделители) [VendorID_vendorspecifickeyword], установленный (установленные) в А.4.1.2.2.11, и должен (должны) быть расположен (расположены) после всех разделов, установленных в данной спецификации.

А.4.1.2.2 Правила форматирования EDS

А.4.1.2.2.1 Общие положения

Файл EDS должен состоять из разделов, входных данных, полей, комментариев и пробелов. Настоящий подраздел устанавливает правила, которые следует соблюдать при определении EDS.

А.4.1.2.2.2 Пробелы в EDS

В файле EDS могут быть использованы пробелы, но они должны игнорироваться всеми интерпретаторами EDS, если они находятся за пределами полей и массивов символов в двойных кавычках.

Интерпретатор EDS должен рассматривать нижеприведенные символы как символы пробелов. Эти символы, считываемые интерпретатором, но не кодированные как символы, читаемые человеком, указывают на наличие в файле пробела:

- знак пробела;
- новая строка;
- возврат каретки;
- перевод строки;
- табуляция, вертикальная и горизонтальная;
- перевод страницы;
- метка конца файла;
- комментарии.

А.4.1.2.2.3 Символы ключевых слов

Все ключевые слова в файле EDS должны состоять из символов ASCII из следующего списка:

- прописные буквы A—Z;
- строчные буквы a—z;
- цифры 0—9;
- специальный символ подчеркивания «_»;
- символ пробела.

Пробел следует использовать только в ключевом слове раздела. Пробел должен находиться внутри названия раздела, и использование множества последовательных пробелов запрещается.

А.4.1.2.2.4 Разделы

Файл EDS должен разделяться на обязательные и дополнительные разделы.

А.4.1.2.2.5 Разделители разделов

Каждый раздел в EDS должен быть надлежащим образом разделен ключевым словом раздела в квадратных скобках (Legal Delimiter). Допустимые разрешенные разделители разделов должны соответствовать разделителям, установленным в таблице А.6.

А.4.1.2.2.6 Ключевые слова разделов

Каждое ключевое слово раздела определяется текстом между началом разделителя ключевого слова раздела «[» и конечным разделителем «]». Символы, используемые в ключевых словах раздела, определены в А.4.1.2.2.3. Имеются два типа ключевых слов раздела, общедоступные ключевые слова и ключевые слова продавца.

А.4.1.2.2.7 Последовательность разделов

Каждый обязательный раздел должен располагаться в требуемом порядке, как установлено в А.4.1.2. Дополнительные разделы могут целиком пропускаться или включаться с пустыми символами — заменителями данных. За исключением раздела (разделов), относящегося (относящихся) к продавцу, дополнительные разделы могут располагаться в любом порядке. Раздел (разделы), относящийся (относящиеся) к продавцу, должен (должны) быть последним (последними) в файле EDS.

А.4.1.2.2.8 Ввод

Каждый раздел в EDS должен содержать один или несколько вводов, начинающихся с ключевого слова ввода, после которого следует знак равенства. Значение ключевого слова ввода должно быть глобальным в области действия, что позволяет использовать ключевые слова, определенные в одном разделе, также и в другом разделе. Каждый ввод должен оканчиваться точкой с запятой. Ввод может расширяться, включая множество строк, если запятые надлежащим образом разделяют поля.

А.4.1.2.2.9 Ключевые слова ввода

Ключевое слово ввода должно состоять из однозначно определяемой последовательности символов ключевого слова, как установлено в А.4.1.2.2.3. Имеются два типа ключевых слов ввода: общедоступные ключевые слова и ключевые слова продавца.

А.4.1.2.2.10 Общедоступное ключевое слово

Общедоступное ключевое слово должно всегда определяться в спецификации CIP ответственными ассоциациями продавцов. Общедоступное ключевое слово никогда не должно начинаться с цифры.

A.4.1.2.2.11 Ключевые слова продавца

Ключевые слова могут быть ключевыми словами, относящимися к продавцу. Эти ключевые слова должны начинаться с идентификатора компании продавца с добавлением после него символа подчеркивания (VendorID_ VendorSpecificKeyword). VendorID (идентификатор продавца) должен воспроизводиться в десятичной системе счисления и не должен содержать в начале нулей. Каждый продавец несет ответственность за сохранение и документирование своих ключевых слов.

A.4.1.2.2.12 Поля вводов

Каждый ввод должен содержать одно или несколько полей. Разделители в виде запятой должны разделять все поля. Значение поля (полей) должно зависеть от контекста раздела. Поля ввода являются либо обязательными, либо дополнительными, как установлено в этой спецификации. Пробелы и пустоты между запятыми должны быть использованы для непредставленных дополнительных полей. Точка с запятой может быть использована для указания отсутствия замыкающих дополнительных полей. Термин «номер поля» должен указывать положение поля ввода. Поля следует нумеровать слева направо (или сверху вниз), начиная с номера 1.

A.4.1.2.2.13 Ключевые слова полей

Ключевое слово полей должно состоять из однозначно определенной последовательности символов ключевого поля, как установлено в A.4.1.2.2.3. Имеются два типа ключевых слов полей: общедоступные ключевые слова и ключевые слова продавца.

A.4.1.2.2.14 Составные поля данных

Некоторые поля ввода следует устанавливать с использованием данных, которые не могут быть определены одним значением между разделителями в виде запятой. Возможность дополнительного разделения всего поля определяют путем использования одного или нескольких наборов символов в виде парных фигурных скобок «{" и "}». Содержимое между этими символами следует рассматривать как один элемент или ввод. Содержимое может быть сгруппировано в множестве фигурных скобок.

A.4.1.2.2.15 Комментарии

Комментарии должны разделяться символом в виде знака доллара ($) и символом новой строки. Интерпретатор EDS должен рассматривать все символы между разделителями комментариев как пробелы. Разделитель комментария $, обнаруживаемый внутри поля или набора символов в виде двойных кавычек, не должен рассматриваться как разделитель комментариев.

Пример – Ниже приводятся примеры некоторых комментариев:

```
$ This is a valid comment line                <NL>
1, 2, 3;                                      $ This is a valid comment <NL>
$ Comments cannot span <NL>
more than one line <NL>                       <= This is an error — no $
```

A.4.1.2.2.16 Пример структуры форматирования EDS

Примеры на рисунке А.1 показывают структуру электронной информационной таблицы.

```
[section name]
$ Comment — extends to end of line
Entry1=Field1, Field2, Field3;                $ Entire entry on one line
Entry2=Field1, Field2, Field3, Field4;        $ Entire entry on one line
Entry3=                                       $ Multiple line entry
        Field1,                               $ Field1
        Field2,                               $ Field2
        Field3;                               $ Field3
Entry4=                                       $ Combination
        Field1, Field2,                       $ Fields 1 and 2 on one line
        Field3,                               $ Field3
        Field4;                               $ Field4
Entry5= 1,                                    $ Field 1 specifies the value 1
        {1,2,3};                              $ Field 2 specifies an array or
                                              $ structure with three values
Entry6= { 44, {22,33,11} };                   $ Entry 6 specifies a single field.
                                              $ The field contains two sets of data.
                                              $ The first set is the single value 44
                                              $ The second set contains three values
65535_Entry=                                  $ Vendor Specific entry for
        Field1, Field2;                       $ Vendor_ID 65535 with two fields
```

Рисунок А.1 — Пример структуры форматирования EDS (информативный)

A.4.1.2.3 Требования к имени файла

Соглашения по присвоению имени файлам не принимаются для дисковых файлов EDS, за исключением файлов в среде DOS/Windows: эти файлы должны иметь суффикс «.EDS», приписываемый к имени файла.

А.4.1.3 Требования к кодированию данных EDS

А.4.1.3.1 Общие положения

В настоящем подразделе установлены требования к кодированию данных для файла EDS.

Информация, содержащаяся в файле EDS, может представлять конфигурируемые атрибуты экземпляров объектов в приборе. Все данные в файле EDS должны быть текстом, состоящим из набора символов ASCII, тогда как атрибуты класса объектов и экземпляров не нуждаются в использовании ASCII (доступные типы данных определяются в спецификации CIP). Поэтому может потребоваться преобразование (трансляция) между данными, содержащимися в файле EDS, и атрибутами объекта; эта трансляция устанавливается в последующих подразделах.

Элементарные типы данных, установленные в спецификации CIP, также используются для других элементов EDS, однако значение изменяется, как описано в последующих подразделах (см. А.4.1.3.3—А.4.1.3.10).

Некоторые типы данных используются только в файлах EDS (см. А.4.1.3.11—А.4.1.3.14).

А.4.1.3.2 Соглашение по файлу символов ASCII

Все данные в EDS должны кодироваться с использованием 8-битных символов ASCII, где все ссылки на символы ASCII означают 8-битный формат символов ASCII (как установлено в таблицах 1 и 2, строка 00 ИСО/МЭК 10646-1). Символы, которые не могут быть воспроизведены на терминале ANSI, не должны использоваться в именах идентификатора или в представлениях данных. Правильные значения символов ASCII должны включать в себя новую строку, табуляцию и значения от 32 до 126 в десятичной системе счисления.

А.4.1.3.3 Соглашение по строке символов — EDS_Char_Array

А.4.1.3.3.1 Общие положения

Все строковые данные в файле EDS должны быть строками символов фиксированной длины без символа конца строки и должны заключаться в двойные кавычки (EDS_Char_Array data type).

Существуют две формы преобразований строковых данных. Символы, заключенные между двойными кавычками, должны быть преобразованы в 8-битные символы ASCII. Символы, заключенные между двойными кавычками, которым предшествует заглавная буква L, должны быть преобразованы в 16-битные символы стандарта UNICODE (юникод).

Пример 1 — «Это результат в строке, состоящей из 8-битных символов».
Пример 2 — L «Строка символов UNICODE, включающая греческий символ Pi \u03C0».

Примечание — Текст \u03C0 определяет одиночный 16-битный символ со значением 03C0. В наборе символов UNICODE это соответствует таблице 9, строка 3, Basic Greek — символ нижнего регистра. Описания управляющей последовательности символов даны в А.4.1.3.3.5.

А.4.1.3.3.2 Действия в случае недостаточного количества символов в поле строки

Интерпретатор EDS должен использовать выравнивание символов в поле по правому знаку и заполнить неустановленные символы пробелами в начале строки текста (ASCII 0x20) для остающейся длины строки.

Пример — Если параметр имеет максимальную длину строки 8 и вмещает строку «123AB», строка интерпретируется как «~~~123AB», где символы тильда (~) представляют пробелы.

А.4.1.3.3.3 Действия в случае избыточного количества символов в поле строки

Если заданное поле строки содержит слишком много символов, интерпретатор EDS должен отсечь символы слева направо.

Пример — Если параметр имеет максимальную длину строки 8 и вмещает строку «I23ABCDEFG», строка отсекается и интерпретируется как «I23ABCDE».

А.4.1.3.3.4 Конкатенация строки

Множество строк без промежуточных запятых должно подвергаться конкатенации.

Пример 1 —
Строка: ***«ABC» «123» «XYZ»***
Интерпретируется как: ***«ABC123XYZ»***

Строки также могут находиться на отдельных строках текста.

Пример 2 —
Следующие строки:
«ABC ***$ Это комментарий***
«123»
«XYZ»
также интерпретируются как: «ABC123XYZ»

Для строки UNICODE (длинная строка) только первым двойным кавычкам должна предшествовать заглавная буква L.

Пример 3 — L «ABC» «123» «XYZ» – то же самое, что и L «ABC123XYZ».

А.4.1.3.3.5 Управляющие последовательности строк

Интерпретатор EDS должен опознавать все управляющие последовательности, перечисленные в таблице А.7. Интерпретация зависит от приложения.

Таблица А.7 — Управляющие последовательности строк

Управляющая последовательность	Трансляция
\\	\
\n	newline (новая строка)
\t	tab (табуляция)
\v	vertical tab (вертикальная табуляция)
\b	backspace (возврат на один символ)
\r	carriage return (возврат каретки)
\f	form feed (перевод страницы)
\a	BELL character (0x07) (символ Звонок)
\"	"
\'	'
\xnn	Одиночный байт, содержащий значение «nn» в шестнадцатеричной системе счисления
\unnnn	Пара байтов, содержащих значение «nnnn» в шестнадцатеричной системе счисления. Такая форма переключения строки является правильной только в случае, если результирующие данные строки имеют длину 16 бит, например, L" форма спецификации строк

Если встречаются с последовательностью, не указанной выше, интерпретирующее устройство должно отвергнуть целую строку и указать ошибку. Файлы EDS должны содержать только управляющие последовательности, установленные в таблице А.7.

А.4.1.3.4 Соглашение по строкам ASCII (STRING, SHORT_STRING, STRING2)

Все типы строковых данных (STRING, SHORT_STRING, STRING2), используемые в атрибутах объектов, должны преобразовываться в файл EDS EDS_Char_Array.

А.4.1.3.5 STRINGI

Тип данных Международной строки (STRINGI) CIP кодируется в файле EDS как представление сложного типа данных. Все содержимое ввода STRINGI должно заключаться в пару фигурных скобок. После числа языковых элементов, установленных как USINT, должны следовать определения языковых элементов, причем каждый такой элемент заключается в пару фигурных скобок и разделяется запятой. Каждый языковой элемент ввода STRINGI должен определяться в виде четырех полей. Первое поле (выбор языка) должно представлять собой точно три символьные строки фиксированной длины, заключенные в двойные кавычки, — код языка, установленный в ИСО 639-2/Т. Тип строковых данных должен быть представлен с использованием кода типа данных, как установлено в спецификации CIP для STRING, STRING2, STRINGN или SHORT_STRING. Выбор набора символов должен быть представлен как UINT, как установлено в Кодах принтера IANA MIB (RFC 1759). Часть содержимого строки языкового элемента должна быть представлена как строка или длинная строка.

Пример — Ниже представлен ввод STRINGI на трех языках:

```
Field1 =      { 3,
              {"eng",0xD0,4,"This is an ASCII English language string"},
              {"spa",0xD5,1000,L"Españoles palabras"},        $ "Spanish words"
                                                             $ using UNICODE
{"deu",0xD0,4,"Spanische Wörter auf Deutsch"}                $"Spanish words in German"
};
```

А.4.1.3.6 Маршрут CIP (EPATH)

Тип данных CIP (EPATH), используемый, в частности, для определения строк маршрута CIP, должен кодироваться в файлах EDS с использованием базового формата, установленного в настоящем стандарте для файла EDS_Char_Array. Кроме того, содержимое строки для маршрута CIP или других данных EPATH должно состоять из групп двух соседних шестнадцатеричных символов, разделенных пробелами. Могут быть использованы как верхний регистр, так и нижний регистр.

Пример 1 — «20 04 24 01»
Пример 2 — «20 05 24 02 30 04»

А.4.1.3.7 Соглашение по целым числам без знака ASCII (USINT, UINT, UDINT, ULINT)

Типы целочисленных данных без знака представляют положительные целочисленные значения. Целочисленные данные без знака следует вводить либо в десятичной системе счисления, либо в шестнадцатеричной системе счисления без пробелов или запятых между символами. Если для представления целочисленных символов без знака используют шестнадцатеричную систему счисления, последовательность двух символов 0x без пробела должна предшествовать целочисленным символам без знака.

диапазон разрешенных данных USINT:
- десятичная система счисления 0—255
- шестнадцатеричная система счисления 0x0—0xFF

диапазон разрешенных данных UINT:
- десятичная система счисления 0—65535
- шестнадцатеричная система счисления 0x—0xFFFF

диапазон разрешенных данных UDINT:
- десятичная система счисления 0—4294967295
- шестнадцатеричная система счисления 0x0—0xFFFFFFFF

диапазон разрешенных данных ULINT:
- десятичная система счисления 0—18446744073709551615
- шестнадцатеричная система счисления 0x0—0xFFFFFFFFFFFFFFFF

В десятичной системе счисления не следует использовать впередиидущие нули, но они могут быть использованы в шестнадцатеричной системе счисления. В шестнадцатеричной системе счисления могут быть использованы как верхний регистр, так и нижний регистр, а общее число символов для типа ULINT должно ограничиваться числом 10 (0x плюс еще 8) или 18 (0x плюс еще 16).

Пример — Десятичное значение UINT 254 может быть представлено как 254 (в десятичной системе счисления) или как 0xFE (в шестнадцатеричной системе счисления), а 0254 (десятичная система счисления) и 0x0000000FE (шестнадцатеричная система счисления) являются недопустимыми.

А.4.1.3.8 Соглашение по целым числам со знаком ASCII (SINT, INT, DINT, LINT)

Типы данных SINT, INT, DINT и LINT представляют значения целочисленных данных со знаком. Целочисленные данные со знаком следует вводить либо в десятичной системе счисления, либо в шестнадцатеричной системе счисления с пробелами или запятыми между символами. Если для представления целочисленных символов со знаком используют шестнадцатеричную систему счисления, последовательность двух символов 0x без пробела должна предшествовать целочисленной последовательности символов.

– диапазон разрешенных данных SINT:
- десятичная система счисления –128—127
- шестнадцатеричная система счисления –0x80—0x7

– диапазон разрешенных данных INT:
- десятичная система счисления –32768—32767
- шестнадцатеричная система счисления –0x8000—0x7FFF

– диапазон разрешенных данных DINT:
- десятичная система счисления –2147483648—2147483647
- шестнадцатеричная система счисления – 0x0—0xFFFFFFFFFFFFFFFF
- диапазон разрешенных данных LINT:

– десятичная система счисления
– 9223372036854775808—9223372036854775807
- шестнадцатеричная система счисления
0x8000000000000000—0x7FFFFFFFFFFFFFFF

В десятичной системе счисления не следует использовать впередиидущие нули, но они могут быть использованы в шестнадцатеричной системе счисления. В шестнадцатеричной системе счисления могут быть использованы как верхний регистр, так и нижний регистр, а общее число символов для типа LINT должно ограничиваться числом 10 (0x плюс еще 8) или 18 (0x плюс еще 16).

Пример — Десятичное значение INT 254 может быть представлено как 254 (в десятичной системе счисления) или как –0xFE (в шестнадцатеричной системе счисления) или как 0x000000FE (в шестнадцатеричной системе счисления), а 0254 (десятичная система счисления) и 0x0000000FE (шестнадцатеричная система счисления) являются недопустимыми.

А.4.1.3.9 Соглашение по словам ASCII (BYTE, WORD, DWORD, LWORD)

Типы данных BYTE, WORD, DWORD и LWORD представляют значения с побитовой адресацией. Эти значения считают дискретными значениями позиции двоичного разряда, и они не предназначены для представления целочисленных значений со знаком или без знака. Однако для удобства эти значения должны быть введены в десятичной, шестнадцатеричной или двоичной системе счисления без пробелов или запятых между символами.

Если для представления целочисленных символов используют шестнадцатеричную (соответственно двоичную) систему счисления, последовательность двух символов 0x (соответственно 0b) без пробела должна предшествовать символам значений.

диапазон разрешенных данных BYTE:
- десятичная система счисления 0—255
- шестнадцатеричная система счисления 0x0—0xFF
- двоичная система счисления 0b00000000—0b11111111

диапазон разрешенных данных WORD:
- десятичная система счисления 0—65535
- шестнадцатеричная система счисления 0x0—0xFFFF
- двоичная система счисления 0b0000000000000000—0b1111111111111111

диапазон разрешенных данных DWORD:
- десятичная система счисления 0—4294967295
- шестнадцатеричная система счисления 0x0—0xFFFFFFFF
- двоичная система счисления 0b00000000000000000000000000000000—0b11111111111111111111111111111111

диапазон разрешенных данных LWORD:
- десятичная система счисления 0—18446744073709551615
- шестнадцатеричная система счисления 0x0—0xFFFFFFFFFFFFFFFF
- двоичная система счисления 0b0000000000000000000000000000000000000000000000000000000000000000—0b1111111111111111111111111111111111111111111111111111111111111111

В десятичной системе счисления не следует использовать впередиидущие нули, но они могут быть использованы в шестнадцатеричной и двоичной системах счисления. В шестнадцатеричной системе счисления могут быть использованы как верхний регистр, так и нижний регистр, а общее число символов для типа LWORD должно ограничиваться числом 10 (0x плюс еще 8) или 18 (0x плюс еще 16).

Пример — Десятичное значение WORD 254 может быть представлено как 254 (в десятичной системе счисления) или как 0xFE (в шестнадцатеричной системе счисления), или как 0x000000FE (в шестнадцатеричной системе счисления), а 0254 (в десятичной системе счисления) и 0x0000000FE (в шестнадцатеричной системе счисления) являются недопустимыми.

A.4.1.3.10 Соглашение по плавающей точке ASCII (REAL, LREAL)

Типы данных REAL и LREAL представляют двоичные значения с плавающей точкой. Внутреннее представление форматов этих данных описано в IEEE 754. В этом же стандарте описаны как численные значения, так и последовательности битов, которые интерпретируются как символьные значения NaN («не число»), а также как плюс и минус бесконечности. Значения с плавающей точкой могут быть введены либо как целые значения, значения, основанные на десятичном представлении плавающей точки, либо как значения, вводимые в «экспоненциальном» представлении. Целые значения являются теми же, что и значения, приведенные для типов данных INT, DINT или LINT. Эти значения не могут быть использованы для представления дробных значений. Десятичные значения с плавающей точкой являются значениями, которые имеют как целую часть, так и дробную компоненту. Целое значение и дробные компоненты разделяются десятичной точкой «.» или символом периода. Форма значений в «экспоненциальном» представлении является такой же, как и представление дробных значений с добавлением экспоненциальной компоненты (показателя степени). Этот показатель степени всегда является целым числом со знаком до 10, в которое возводится значение основания системы счисления.

Примечание — Максимальная точность значений с плавающей точкой определяется возможностями внутреннего двоичного формата, т. е. числом двоичных разрядов, доступных для кодирования мантиссы. Поэтому использование большого числа десятичных разрядов в десятичной системе счисления (мантисса в экспоненциальном представлении) для значений с плавающей точкой является более удобным представлением. EDS определяет произвольные пределы для числа десятичных разрядов.

Диапазон разрешенных данных REAL (32-битный формат IEEE с ординарной точностью) устанавливают по формуле

$$\text{значение} = (-1)^{s} \cdot (2)^{e-127} \cdot (m),$$

где s — значение знакового бита;

e — 8-битный показатель степени. Допустимые значения этого показателя лежат в диапазоне от –126 до +127;

m — нормализованная 24-битная мантисса (23 бита внутренней памяти плюс один скрытый бит). Допустимые значения мантиссы лежат в диапазоне между 0 и 16777215.

Комбинирование e и m позволяет установить диапазон приближенных абсолютных значений 0—3,4028e^{38}.

EDS использует для данных REAL следующие представления значений с плавающей точкой:

- представление в виде целых чисел (фиксированных): от –16777215 до 1677725;
- десятичная система счисления (с плавающей точкой): 0.0 – ±9999999999999999,

где общее число разрядов не должно превышать 16 (помимо символа десятичной точки и символа знака). Оба этих символа могут быть опущены (знак + подразумевается, если символ знака опущен).

«Экспоненциальное» представление: 0.0 – ±nn.nnnnnnnnnE±xxxx,

где общее число разрядов в мантиссе не должно превышать 11 (помимо символа десятичной точки и символа знака), а число разрядов в показателе степени не должно превышать 4 (помимо символа «E» и символа знака). Десятичная точка может располагаться в мантиссе где угодно. Как символ десятичной точки, так и символ знака в мантиссе могут быть опущены (знак «+» подразумевается, если символ знака опущен).

Диапазон разрешенных данных LREAL (64-битный формат IEEE с двойной точностью) определяют по формуле

$$\text{значение} = (-1)^{s} \cdot (2)^{e-1023} \cdot (m),$$

где s — значение знакового бита;

e — 11-битный показатель степени. Допустимые значения этого показателя лежат в диапазоне от –1022 до +1023;

m — нормализованная 53-битная мантисса (52 бита внутренней памяти плюс один скрытый бит). Допустимые значения мантиссы лежат в диапазоне между 0 и 9007199254740991.

Комбинирование e и m позволяет установить диапазон приближенных абсолютных значений 0—1,7976e^{308}.

EDS использует для данных LREAL следующие представления значений с плавающей точкой:

- представление в виде целых чисел (фиксированных):
- от 9007199254740991 до + 9007199254740991;
- десятичная система счисления (с плавающей точкой):

0.0—±9999999999999999,

где общее число разрядов не должно превышать 16 (помимо символа десятичной точки и символа знака). Оба этих символа могут быть опущены (знак «+» подразумевается, если символ знака опущен).

«Экспоненциальное» представление:

0.0—±nnnn.nnnnnnnnnnnnnnE±xxxx,

где общее число разрядов в мантиссе не должно превышать 16 (помимо символа десятичной точки и символа знака), а число разрядов в показателе степени не должно превышать 4 (помимо символа «E» и символа знака). Десятичная точка может располагаться в мантиссе где угодно. Как символ десятичной точки, так и символ знака в мантиссе могут быть опущены (знак «+» подразумевается, если символ знака опущен).

В дополнение к вводимым значениям, указанным выше, представление плавающей точки учитывает два стиля вводов и два типа бесконечностей. Существуют два типа Nan («не число»), а именно сигнализация Nan и «тихое» NaN (Quiet NaN). Кроме того, формат устанавливает представления значений плюс и минус бесконечностей. В этих случаях резервируются следующие специальные слова, которые должны использоваться для представления ввода ассоциированного символа плавающей точки:

- «тихое» не число — QUIET-NAN;
- сигнализация не числа — SIGNAL-NAN;
- бесконечность плюс — INFINITY (или +INFINITY);
- бесконечность минус — -INFINITY.

A.4.1.3.11 EDS_Date

Тип данных EDS_Date должен иметь формат mm-dd-yyyy, где mm — месяц, dd — день, yyyy — год. Правильные значения месяца, дня и года mm-dd-yyyy должны быть следующими:

mm — 01—12;

dd — 01—31 (в зависимости от месяца и года);

yyyy — 1996—9999.

Представление года двумя символами может быть использовано в случае, если тип данных EDS_Date имеет формат mm-dd-yy, где mm — месяц, dd – день месяца, а yy — год. В этом случае две первые цифры года имеют предполагаемое значение 19, так что yy=96 должно означать год 1996. Правильные значения месяца, дня и года параметров mm-dd-yy должны быть следующими:

mm — 01—12;

dd — 01—31 (в зависимости от месяца и года);

yy — 98—99 (подразумеваются две первые цифры 19).

Примечание — Представление года двумя цифрами не рекомендуется.

A.4.1.3.12 EDS_Time_Of_Day

Тип данных EDS_Time_Of_Day должен иметь формат hh.mm.ss, где hh — часы, mm — минуты и ss — секунды. Правильные значения часов, минут и секунд должны быть следующими:

hh — 00—23;

mm — 00—59;

ss — 00—59.

A.4.1.3.13 EDS_Revision

Тип данных EDS_Revision должен иметь формат Major_Revision.Minor_Revision с правильными значениями:

Major_Revision 0—9;

Minor_Revision 0—9.

Значение EDS_Revision 0.0 следует рассматривать как ошибочное.

Пример — EDS_Revision 1.4 соответствует старшей версии 1 и младшей версии 4.

A.4.1.3.14 Универсальный указатель ресурсов EDS_URL

Все ссылки на EDS_URL в требованиях к EDS предназначены для получения формализованной информации, необходимой для определения местоположения ресурсов и доступа к ним через Интернет. EDS_URL должен кодироваться в файлах EDS с использованием базового формата, установленного в ИСО 15745-2 для EDS_Char_Array. Кроме того, содержимое строки для EDS_URL должно иметь формат, установленный Сетевой рабочей группой RFC 1738 «Универсальный указатель ресурсов (URL)» Интернета. Что касается спецификаций в файле EDS, EDS_URL должен ограничиваться любой из следующих форм:

http;

ftp;

file (файл).

A.4.1.4 Основные требования к файлу EDS

A.4.1.4.1 Обзор

В данном подразделе описаны основные разделы EDS, являющиеся общими для нескольких сетей на основе CIP, и установлены соответствующие требования к применению. В таблице А.8 указано местоположение подразделов определений этих разделов.

Таблица А.8 — Определение основных разделов

Разделы EDS	Определено в
Раздел описания файла	A.4.1.4.2
Раздел описания устройства	A.4.1.4.3
Раздел классификации устройства	A.4.1.4.4
Раздел классов параметров	A.4.1.4.5
Раздел параметров	A.4.1.4.6
Раздел групп параметров	A.4.1.4.7
Раздел компоновки	A.4.1.4.8
Раздел управления соединениями	A.4.1.4.9
Раздел портов	A.4.1.4.10
Раздел модулей	A.4.1.5.2

A.4.1.4.2 Раздел описания файла

Раздел описания файла должен содержать административную информацию о файле EDS. Средства конфигурирования должны считывать эту информацию, форматировать и воспроизводить ее пользователю. Пользователь также может иметь доступ к этому разделу с помощью программы просмотра текстового файла и воспроизводить неформатированную информацию. Раздел не требует внесения в него изменений, если только пользователь вручную не внес изменения в файл. Раздел описания файла должен содержать вводы, приведенные в таблице А.9.

Таблица А.9 — Формат описания файла

Имя вводимых данных (ввода)	Ключевое слово ввода	Номер поля	Тип данных	Обязательно/ Дополнительно
Текст описания файла	DescText	1	EDS_Char_Array	Обязательно
Дата создания файла	CreateDate	1	EDS_Date	Обязательно
Время создания файла	CreateTime	1	EDS_Time_Of_Day	Обязательно
Дата внесения последних изменений	ModDate	1	EDS_Date	Дополнительно
Время внесения последних изменений	ModTime	1	EDS_Time_Of_Day	Дополнительно
Пересмотр EDS	Revision	1	EDS_Revision	Обязательно
Базовый URL	HomeURL	1	EDS_URL	Дополнительно

Вводимые в раздел описания файла данные должны предоставлять информацию, как показано в таблице А.10.

Таблица А.10 — Вводимые (входные) данные описания файла

Вводимые данные	Описание
Текст описания файла	Отдельная строка текста, воспроизводимая средствами конфигурирования. Разработчик EDS должен определить для этого ввода значимую строку текста. Все наборы символов должны заключаться в двойные кавычки
Дата создания файла	Дата создания EDS, присваемая разработчиком EDS. Эта дата, предусмотренная только ради удобства, может быть использована для получения информации о версии файла. Средства конфигурирования не должны использовать эту информацию для любого типа управления версией, но они могут воспроизводить содержимое
Время создания файла	Время создания EDS, присваемое разработчиком EDS. Это время, предусмотренное только ради удобства, может быть использовано для получения информации о версии файла. Средства конфигурирования не должны использовать эту информацию для любого типа управления версией, но они могут воспроизводить содержимое
Дата внесения последних изменений	Дата внесения последних изменений в EDS. Средства конфигурирования, позволяющие вносить изменения в файл EDS, в случае необходимости должны обновлять это поле. Средства конфигурирования, предусмотренные только ради удобства, должны воспроизводить содержимое этих вводимых данных, если вносятся изменения. Если средства конфигурирования вносят изменения в EDS, то они должны обновлять это поле. Однако если изменения в EDS вносят вручную или с помощью текстового редактора, то это поле также должно обновляться. Такой ввод необходим, если - изменения в файл EDS вносят с использованием программных средств или если - представлены вводимые данные о времени внесения последних изменений
Время внесения последних изменений	Время внесения последних изменений в EDS. Средства конфигурирования, позволяющие вносить изменения в файл EDS, в случае необходимости должны обновлять эти вводимые данные. Средства конфигурирования, предусмотренные ради удобства, должны воспроизводить содержимое этого ввода, если вносятся изменения. Если средства конфигурирования вносят изменения в EDS, то они должны обновлять это поле. Однако если изменения в EDS вносят вручную или с помощью текстового редактора, то это поле также должно обновляться
Пересмотр EDS	Пересмотр EDS. Пересмотр EDS не взаимосвязан с каким-либо пересмотром продуктов; это просто пересмотр самого файла EDS

Окончание таблицы А.10

Вводимые данные	Описание
Базовый URL	Универсальный указатель ресурсов основного файла EDS, файл, содержащий изображение иконки, и другие файлы, связанные с данным EDS. HomeURL должен определять полный квалифицированный URL для ссылки на основную версию файла EDS. Кроме того, область ссылки (без спецификации имени файла) используется для определения области, в которой содержится (содержатся) другой связанный файл (другие связанные файлы), относящийся (относящиеся) к устройству, описываемому данным EDS

На рисунке А.2 приведен пример, представляющий типичный раздел [File].

```
[File]
        DescText = "Smart Widget EDS File";
        CreateDate = 04-03-94;                  $ created
        CreateTime = 17:51:44;
        ModDate = 04-06-94;                     $ last changed
        ModTime = 22:07:30;
        Revision = 2.1;                         $ Revision of EDS
        HomeURL = "http://www.odva.org/EDS/example.eds";
```

Рисунок А.2 — Пример раздела [File] (информативный)

А.4.1.4.3 Раздел описания устройств

Раздел описания устройств должен содержать информацию изготовителя по устройствам, включая те же значения, что и в объекте Идентичность устройств. Раздел описания устройств должен содержать входные данные, установленные в таблице А.11.

Таблица А.11 — Формат описания устройств

Входные данные	Ключевое слово	Номер поля	Тип данных	Обязательно/ Дополнительно
Идентификатор продавца [a], [b]	VendCode	1	UINT	Обязательно
Название продавца	VendorName	1	EDS_Char_Array	Обязательно
Тип устройства[a], [b]	ProdType	1	UINT	Обязательно
Строка типа устройства	ProdTypeStr	1	EDS_Char_Array	Обязательно
Код продукта[a], [b]	ProdCode	1	UINT	Обязательно
Старшая версия[a], [b]	MajRev	1	USINT	Обязательно
Младшая версия[a]	MinRev	1	USINT	Обязательно
Название продукта[c]	ProdName	1	EDS_Char_Array	Обязательно
Номер каталога	Catalog	1	EDS_Char_Array	Дополнительно
Исключить из соединения с использованием адаптерного блока	Исключить FromAdapterRackConnection	1	EDS_Char_Array	Дополнительно
Имя файла, содержащего изображение иконки	Icon	1	EDS_Char_Array	Дополнительно

[a] Эти вводимые данные представляют атрибут объекта Идентичность.
[b] Эти вводимые данные используют для согласования EDS с конкретным продуктом/версией.
[c] Эти вводимые данные представляют атрибут объекта Идентичность, хотя тип данных может слегка отличаться.

Название входных данных поля описания устройства устанавливает номер строки ввода.

Средства конфигурирования должны использовать требуемые входные данные в разделе описания устройств для согласования EDS с конфигурируемым устройством. Входные данные должны предоставлять информацию, установленную в таблице А.12.

Таблица А.12 — Входные данные для описания устройств

Вводимые данные (вводы)	Описание
Идентификатор продавца	Числовой идентификатор продавца, определяемый объектом Идентичность, Атрибут 1
Название продавца	Текстовое название продавца. При воспроизведении в зависимости от возможностей дисплея может возникнуть необходимость усечения текста
Тип устройства	Числовой идентификатор устройства, определяемый объектом Идентичность, Атрибут 2
Строка типа устройства	Текстовое описание типа устройства, как установлено в соответствующем профиле устройств на основе CIP. Отдельные продавцы могут выбрать строки для отдельных типов устройств продавца
Код продукта	Установленный числовой идентификатор кода продуктов продавца, определяемый объектом Идентичность, Атрибут 3. Каждый код продуктов должен иметь собственную EDS
Старшая версия	Номер старшей версии, установленной продавцом, как определяется объектом Идентичность, Атрибут 4. Номер старшей версии продукта обычно может быть повышен , если вносятся изменения в форму, подготовку или функцию устройства. Изменения, внесенные в старшие версии, должны использоваться средствами конфигурирования для согласования устройства с EDS
Младшая версия	Номер младшей версии, установленной продавцом, как определяется объектом Идентичность, Атрибут 4. Номер младшей версии должен использоваться для идентификации изменений, внесенных в продукт, которые не влияют на выбор конфигурации пользователя (например, изменения, вносимые в корректоры ошибок встроенных программ, дополнительные светоизлучающие диоды, внутренние аппаратные средства). Изменения, вносимые в младшие версии, не должны использоваться средствами конфигурирования для согласования прибора с EDS
Название продукта	Текстовое название продукта, как определено объектом Идентичность, Атрибут 7. При воспроизведении в зависимости от возможностей дисплея может возникнуть необходимость усечения текста
Номер каталога	Номер текстового каталога или модели. Один или несколько номеров каталога могут быть ассоциированы с индивидуальным кодом продукта. Примечание — В случае множества номеров каталога полезно предоставить столько номеров, сколько возможно. Например, номер 1438-BAC7xx, где 'xx' представляет варианты номеров каталога, поддерживаемые данным кодом продукта/EDS
Исключить из FromAdapterRackConnection	Это поле используется для описания, если устройство стойки необходимо исключить из адаптерного соединения стойки. Если значение поля соответствует строке «Yes», то этот модуль должен быть исключен из адаптерных соединений стойки путем сброса ассоциированных битов щелевой маски (вход, выход и конфигурация). Если значение поля соответствует строке «No» или это дополнительное поле пропущено, ассоциированные биты щелевой маски могут быть установлены

Окончание таблицы А.12

Вводимые данные (вводы)	Описание
Имя файла, содержащего изображение иконки	Имя файла, содержащего изображение иконки. Идентификация файла, содержащего графическое представление устройства. Файл должен иметь формат *.ICO MSWindows и должен содержать, как минимум, иконку 16×16. Файл также может содержать иконки 32×32, 48×48 и 64×64. Местоположение файла, содержащего изображение иконки, определяется с учетом местоположения, установленного ключевым словом HomeURL (без компоненты имени файла HomeURL), и имени файла, установленного этим ключевым словом. Это ключевое слово должно быть представлено только в случае, если ключевое слово HomeURL существует

На рисунке А.3 приводится пример, представляющий типичный раздел устройств [Device].

```
[Device]
    VendCode = 65535;
    VendName = "Widget-Works, Inc.";
    ProdType = 0;
    ProdTypeStr = "Generic";
    ProdCode = 42;
    MajRev = 1;                    $ Device Major Revision
    MinRev = 1;                    $ Device Minor Revision
    ProdName = "Smart-Widget";
    Catalog = "1499-DVG";
    Icon = "example.ico";
```

Рисунок А.3 — Пример раздела устройств (информативный)

А.4.1.4.4 Раздел классификации устройств

Раздел классификации устройств должен классифицировать устройство, описываемое EDS, как принадлежащее одному или нескольким категориям устройств. Ключевое слово ввода для всех классификаций должно состоять из набора символов «Class», объединенных с десятичным числом. Числа должны начинаться с 1 для первого класса и должны увеличиваться для каждого дополнительного класса.

Число полей для каждого ввода классификации должно быть переменным, что позволяет построить древовидную структуру классификации, аналогичную структуре директории систем. Подклассификация общедоступных классификаций должна резервироваться. Классификация со стороны продавца может быть подклассифицирована по усмотрению продавца. Первое поле должно представлять наивысший уровень в древовидной структуре и должно включать в себя одно из следующих ключевых слов:

- ControlNet;
- DeviceNet;
- EtherNetIP;
- ключевое слово поля со стороны продавца.

Ключевое слово со стороны продавца должно начинаться с идентификатора компании продавца, после чего следует символ подчеркивания (VendorID_VendorSpecificField). VendorID должен воспроизводиться в десятичной системе счисления и не должен содержать впередидущих нулей. Каждый продавец со своей стороны несет ответственность за поддержание и документирование ключевого слова поля.

А.4.1.4.5 Раздел классов параметров

Раздел классов параметров должен идентифицировать общие атрибуты параметров конфигурации, описываемые EDS, которые соответствуют поднабору атрибутов класса Parameter Object и установлены в Библиотеке Объектов CIP.

Раздел классов параметров должен содержать вводимые данные, установленные в таблице А.13.

Таблица А.13 — Формат классов параметров

Вводимые данные (вводы)	Ключевое слово вводимых данных	Число полей	Тип данных	Обязательно/ Дополнительно
Максимальное число экземпляров	MaxInst	1	UINT	Обязательно
Дискриптор классов параметров	Descriptor	1	WORD	Обязательно
Экземпляр компоновки конфигурации	CfgAssembly	1	UINT	Обязательно

Вводимые в раздел классов параметров данные должны предоставлять информацию, как показано в таблице А.14.

Таблица А.14 — Вводимые данные (вводы) классов параметров

Вводимые данные (вводы)	Описание
Максимальное число экземпляров	Идентифицирует общее число параметров конфигурации, содержащихся в устройстве, ассоциированных с EDS
Дескриптор классов параметров	Содержит битовые флаги, описывающие поведение объектов параметров устройства
Экземпляр компоновки конфигурации	Устанавливает число экземпляров объекта Компоновка, содержащих данные по конфигурации устройства

Ввод дескриптора классов параметров должен содержать биты, которые описывают характеристики параметров, как установлено в таблице А.15. Биты, не установленные в таблице А.15, не должны использоваться и им должны быть присвоены нулевые значения (0).

Таблица А.15 — Значения битов дескриптора классов параметров

Бит	Название	Значение и смысл битов
0	Поддерживает индивидуальный доступ к параметрам	0 = Ни к одному из параметров не имеется индивидуальный доступ. Используется только компоновка конфигурации. 1 = К параметрам имеется индивидуальный доступ
1	Поддерживает полные атрибуты	0 = Только текущее значение параметров доступно в устройстве. 1 = Все данные по конфигурации для параметра доступны в самом устройстве
2	Команда сохранения в энергонезависимом запоминающем устройстве	0 = Параметры, сохраняемые автоматически. 1 = Параметры, не сохраняемые автоматически. Должна быть выполнена команда сохранения в энергонезависимом запоминающем устройстве, если необходимые параметры сохраняются в энергонезависимом запоминающем устройстве
3	Параметры сохраняются в энергонезависимом запоминающем устройстве	0 = Параметры не сохраняются в энергонезависимом запоминающем устройстве. 1 = Все полные параметры сохраняются в энергонезависимом запоминающем устройстве

Рисунок А.4 является примером, показывающим типичный раздел класса параметров.

```
[ParamClass]
    MaxInst = 3;
    Descriptor = 0x0E;
    CfgAssembly = 3;
```

Рисунок А.4 — Пример раздела класса параметров [ParamClass] (информативный)

А.4.1.4.6 Раздел параметров

Раздел параметров должен идентифицировать параметры конфигурации в устройстве. Ключевое слово ввода должно быть одним из следующих наборов символов: «Param», «ProxyParam», «ProxiedParam», объединенным с числом (десятичным) экземпляров параметров для устройства, например «Param1». Фактический экземпляр объекта параметров может быть (но необязательно) реализован в устройстве. Наоборот, не требуется, чтобы все экземпляры объекта параметров имели соответствующий ввод «ParamN» в EDS. Однако если экземпляр объекта параметров существует в узле и если этот параметр также описывается в EDS, тогда значение «N» в «ParamN» должно соответствовать экземпляру объекта параметра.

Каждый ввод должен содержать форматированные поля, представленные в таблице А.16. Ключевые слова «ProxyParam» и «ProxiedParam» определяются далее в А.4.1.5.3.1 как часть требований EDS к модулям.

Таблица А.16 — Формат параметров

Имя поля	Номер поля	Тип данных	Обязательно/ Дополнительно
Зарезервированное	1	USINT	Обязательно
Размер маршрута связи	2	USINT	Дополнительно
Маршрут связи	3	EPATH	Дополнительно
Дескриптор	4	WORD	Обязательно
Тип данных	5	USINT/EPATH	Обязательно
Размер данных	6	USINT	Обязательно
Название параметра	7	EDS_Char_Array	Обязательно
Строка единиц измерения	8	EDS_Char_Array	Обязательно
Строка справки	9	EDS_Char_Array	Обязательно
Минимальное значение	10	data type	Дополнительно
Максимальное значение	11	data type	Дополнительно
Значение по умолчанию	12	data type	Обязательно
Масштабный множитель	13	UINT	Дополнительно
Масштабный делитель	14	UINT	Дополнительно
Масштабное основание	15	UINT	Дополнительно
Масштабное смещение	16	INT	Дополнительно
Связь множителя	17	UINT	Дополнительно
Связь делителя	18	UINT	Дополнительно
Связь основания	19	UINT	Дополнительно
Связь смещения	20	UINT	Дополнительно
Десятичная точность	21	USINT	Дополнительно
Международное название параметров	22	STRINGI	Дополнительно
Международные технические единицы	23	STRINGI	Дополнительно
Международная строка справки	24	STRINGI	Дополнительно
[a] Эти позиции будут определены в таблице А.20.			

Вводимые данные раздела параметров должны содержать информацию, установленную в таблицах А.17 и А.21.

Поля параметров, перечисленные в таблице А.17, являются общими для всех параметров.

Таблица А.17 — Общие поля параметров

Поля	Описание
Зарезервированное	Данное первое поле должно содержать нуль
Размер маршрута связи	Число байтов, используемых для представления маршрута. Если размер канала связи не согласуется с числом байтов в поле «Маршрут связи», тогда «Размер канала связи» должен быть проигнорирован. Если этот параметр не адресуется из канала связи, то данное поле должно быть пустым. Если это поле является пустым, а поле «Маршрут связи» непустым, «Размер канала связи» должен быть равен числу байтов в поле «Маршрут связи»

Окончание таблицы А.17

Поля	Описание
Маршрут связи	Маршрут CIP к атрибуту объекта, где было найдено значение параметра. Маршрут должен быть введен как набор символов, используя нотацию маршрута, описанную в МЭК 62026-3:2000, и формат, установленный в А.4.1.3.6. Если параметр, описываемый этим вводом ParamN, не непосредственно адресуется из сети, то данное поле должно быть пустым. Если это поле содержит пустую строку, "", параметр, описываемый этим вводом ParamN, должен быть адресован как атрибут данных (атрибут экземпляра 1) N-го экземпляра объекта параметров (т. е. используя маршрут «20 0F 24 N 30 01»)
Дескриптор	Дескриптор параметров. Содержит флаги битов, описывающих поведение отдельных параметров (см. таблицу А.18)
Тип данных	Идентификатор типов данных, как определено в МЭК 62026-3:2000 (Спецификация и Кодирование типов данных). Данный идентификатор должен кодироваться либо как USINT, либо как EPATH. Примечание — Старые версии файлов EDS могут использовать идентификаторы типов данных USINT, как установлено в таблице А.19, но эти версии в настоящее время устарели. Они представлены здесь по причинам совместимости
Размер (объем) данных	Значение размеров числовых данных. Для типов данных строки и EPATH это поле устанавливает число байтов в расчете на символ или ввод. Следовательно, для типов данных STRING и EPATH это значение должно быть равно 1. Для типа данных STRING2 оно должно равняться 2. Для типа данных STRINGN рассматриваемое значение должно быть установлено, как равное N
Название параметра	Текстовое название параметра. В случае необходимости для согласования с максимально допустимой длиной набора символов найденный текст должен быть усечен
Строка единиц измерения	Набор символов единиц измерения на текстовом дисплее. В случае необходимости для согласования с максимально допустимой длиной набора символов найденный текст должен быть усечен
Строка справки	Набор символов текстовой справки. В случае необходимости для согласования с максимально допустимой длиной набора символов найденный текст должен быть усечен
Минимальное значение	Что касается смысла и требований, основанных на типе данных параметров, см. таблицу А.20
Максимальное значение	Что касается смысла и требований, основанных на типе данных параметров, см. таблицу А.20
Значение по умолчанию	Числовое значение по умолчанию, присваемое значению данных параметров
Международное название параметров	Название параметра, представляемое в нотации STRINGI
Название международных технических единиц	Технические единицы измерения, представляемые в нотации STRINGI
Международная строка справки	Строка справки, представляемая в нотации STRINGI

Биты поля дискриптора должны быть такими, как установлено в таблице А.18

Таблица А.18 — Определения битов поля дескриптора

Бит	Определение	Значение и смысл битов
0	Поддерживает устанавливаемый маршрут	0 = Маршрут связи не может быть установлен. 1 = Маршрут связи может быть установлен
1	Поддерживает нумерованные строки	0 = Нумерованные строки не поддерживаются. 1 = Нумерованные строки поддерживаются и могут быть считаны

Окончание таблицы А.18

Бит	Определение	Значение и смысл битов
2	Поддерживает масштабирование	0 = Масштабирование не поддерживается. 1 = Масштабирование поддерживается. Атрибуты масштабирования реализуются, и значение представляется пользователю в технических единицах измерения
3	Поддерживает масштабируемые связи	0 = Масштабируемые связи не поддерживаются. 1 = Значения атрибутов масштабирования могут быть получены из других параметров
4	Только параметр считывания	0 = Значение параметра может быть записано (установлено) и считано (получено). 1 = Значение параметра может быть только считано (получено), а не установлено
5	Параметр мониторинга	0 = Значение параметра не обновляется устройством в реальном времени. 1 = Значение параметра обновляется устройством в реальном времени
6	Поддерживает расширенное точное масштабирование	0 = Расширенное точное масштабирование не поддерживается. 1 = Расширенное точное масштабирование должно быть осуществлено, и значение должно быть представлено пользователю в технических единицах измерения
7	Поддерживает непоследовательные нумерованные строки	0 = Непоследовательные нумерованные строки не поддерживаются. 1 = Непоследовательные нумерованные строки поддерживаются
8	Допускает как нумерованные, так и отдельные значения	0 = Как нумерованные, так и отдельные значения не поддерживаются. 1 = Как нумерованные, так и отдельные значения поддерживаются
9—15	Зарезервированные	Эти биты резервируются и должны устанавливаться на 0

Старые версии файлов EDS могут использовать идентификаторы типов данных, как установлено в таблице А.19.

Таблица А.19 — Идентификаторы типов данных (устаревшие)

Идентификатор типа данных	Определение	Описание типов данных
1	WORD	16-битное слово
2	UINT	16-битное целое число без знака
3	INT	16-битное целое число со знаком
4	BOOL	Булево
5	SINT	Короткое целое число
6	DINT	Двойное целое число
7	LINT	Длинное целое число
8	USINT	Короткое целое число без знака
9	UDINT	Двойное целое число без знака
10	ULINT	Длинное целое число без знака
11	REAL	Ординарный формат с плавающей точкой (IEEE 754)

Окончание таблицы А.19

Идентификатор типа данных	Определение	Описание типов данных
12	LREAL	Двойной формат с плавающей точкой (IEEE 754)
13	ITIME	Длительность (короткая)
14	TIME	Длительность
15	FTIME	Длительность (высокое разрешение)
16	LTIME	Длительность (продолжительная)
17	DATE	Дата
18	TIME_OF_DAY	Время суток
19	DATE_AND_TIME	Дата и время
20	STRING	8 бит в расчете на строку символов
21	STRING2	16 бит в расчете на строку символов
22	STRINGN	N байт в расчете на строку символов
23	SHORT_STRING	Короткая N-байтная строка символов
24	BYTE	8-битная строка
25	DWORD	32-битная строка
26	LWORD	64-битная строка

Таблица А.20 определяет смысл и конкретные требования, касающиеся вводов минимальных и максимальных значений, основанных на типе данных параметров.

Таблица А.20 — Семантика для вводов минимальных и максимальных значений

Тип данных	Описание и семантика	Семантика минимальных значений	Семантика максимальных значений	Обязательно/ Дополнительно/ Недопустимо
BYTE	Битовая строка — длина 8 бит	Минимальное и максимальное значения для этих типов данных не определены и не должны устанавливаться в файле EDS		Недопустимо
WORD	Битовая строка — длина 16 бит			
DWORD	Битовая строка — длина 32 бита			
LWORD	Битовая строка — длина 64 бита			
STRING [a)]	Строка (индикатор длиной 2 байта, 1 байт в расчете на символ)	Минимальная длина строки	Максимальная длина строки	Обязательно
STRING2 [a)]	Строка (индикатор длиной 2 байта, 2 байта в расчете на символ)	Минимальная длина строки	Максимальная длина строки	Обязательно
STRINGN [a)]	Строка (индикатор длиной 2 байта, N байтов в расчете на символ)	Минимальная длина строки	Максимальная длина строки	Обязательно

Окончание таблицы А.20

Тип данных	Описание и семантика	Семантика минимальных значений	Семантика максимальных значений	Обязательно/ Дополнительно/ Недопустимо
SHORT_STRING[a]	Строка символов (индикатор длиной 1 байт, 1-байтные символы)	Минимальная длина строки	Максимальная длина строки	Обязательно
EPATH[a]	Нумерованный маршрут	Минимальная длина строки	Максимальная длина строки	Дополнительно
Все другие типы данных		Минимальное числовое значение, которое может быть присвоено значению данных	Максимальное числовое значение, которое может быть присвоено значению данных	Дополнительно[b]

[a] Типы данных STRING, STRING2, STRINGN, SHORT_STRING и EPATH не имеют спецификации по минимальному или максимальному значению. Поля минимального и максимального значений используются для представления длин минимальной и максимальной строки или минимального и максимального маршрута. В этих случаях параметр Размер данных используется для представления числа байтов, необходимых в расчете на символ или кодируемую единицу.

[b] Если Минимальное значение и/или Максимальное значение не установлены, тогда минимальное и/или максимальное значения для параметра данных определяются в соответствии с МЭК 62026-3:2000, основываясь на типе данных параметра.

Поля параметров, перечисленные в таблице А.21, являются дополнительными и значимыми только в случае, если используются со следующими типами данных: SINT, INT, DINT, LINT, USINT, UINT, UDINT, ULINT, REAL и LREAL. Спецификация этих полей с любым другим типом данных запрещается.

Таблица А.21 — Поля параметров, зарезервированные для типов числовых данных

Поля	Описание
Масштабный множитель	Числовое значение множителя, применяемого к текущему значению данных параметра
Масштабный делитель	Числовое значение делителя, применяемого к текущему значению данных параметра
Масштабное основание	Числовое значение основания, применяемого к текущему значению данных параметра
Масштабное смещение	Числовое значение смещения, применяемого к текущему значению данных параметра
Связь множителя	Номер параметра, указывающий экземпляр объекта Параметр или другой атрибут объекта, который содержит числовое значение множителя, применяемого к текущему значению данных параметра
Связь делителя	Номер параметра, указывающий экземпляр объекта Параметр или другой атрибут объекта, который содержит числовое значение делителя, применяемого к текущему значению данных параметра
Связь основания	Номер параметра, указывающий экземпляр объекта Параметр или другой атрибут объекта, который содержит числовое значение основания, применяемого к текущему значению данных параметра
Связь смещения	Номер параметра, указывающий экземпляр объекта Параметр или другой атрибут объекта, который содержит числовое значение смещения, применяемое к текущему значению данных параметра
Десятичная точность	Числовое значение точности, применяемое к текущему значению данных параметра

Масштабирование не должно выполняться устройством, содержащим параметр, но должно выполняться средствами воспроизведения. Если масштабирование поддерживается, средство масштабирования должно использовать формулу, приведенную на рисунке А.5, позволяющую определить техническое значение параметра (т. е. воспроизводимое значение) из фактического значения параметра. Если масштабирование не поддерживается, тогда значение параметра должно воспроизводиться как оно есть.

$$EngValue = \frac{(ActualValue + Offset) \cdot Mult \cdot Base}{Div \cdot 10^{DecimalPrecision}}$$

Примечание — Если расширенное масштабирование не поддерживается, то эта формула должна использоваться с DecimalPrecision=0.

Рисунок А.5 – Формула масштабирования параметра

В разделе [Params] может также быть второе ключевое слово. Это ключевое слово должно использоваться для предоставления нумерованного списка выборов параметров, представляемых пользователю. Ключевое слово ввода для всех нумерованных параметров должно состоять из набора символов, «Enum», объединенных с десятичным числом из соответствующего ввода Param. Каждый ввод «Enum» должен состоять из пар целых чисел и строк.

На рисунке А.6 приведен пример типичного раздела Parameter.

```
[Params]
  Param1 = 0, 1,"20 02", 0x0E94, 1, 1,"Preset","V","User Manual p33", 0, 5, 1, 1, 1, 1, 0, 0, 0, 0, 0, 2;
  Param2 =                          $ parameter instance
  0,                                $ First field shall equal 0
  6, "20 04 24 01 30 03",           $ path size, path
  0x0A94,                           $ descriptor — in hex format
  1,                                $ data type
  1,                                $ data size
  "Trigger",                        $ name
  "Hz",                             $ units
  "User Manual p49",                $ help string
  0, 2, 0,                          $ min, max, default data values
  1, 1, 1, 0,                       $ mult, div, base, offset scaling
  , , , ,                           $ mult, div, base, offset links not used
  2;                                $ decimal places
  Param3 =                          $ not addressable from link
  0, , , 0x0082, 8, 1, "speed control", " ", " ", 3, 12, 3, ,,,,,,,;
  Enum3 = 3, "stop", 8, "slow", 12, "fast";
```

Рисунок А.6 — Пример раздела [Params]

А.4.1.4.7 Раздел групп параметров

Раздел групп параметров должен идентифицировать все группы параметров в устройстве. Каждая группа параметров должна содержать список параметров в группе. Ключевое слово ввода для каждой группы должно состоять из комбинации набора символов, «Group», и номера группы параметров (десятичного), например «Group1». Десятичные номера должны начинаться с единицы и увеличиваться на единицу.

Фактический экземпляр объекта группы параметров может быть (но необязательно) реализован в устройстве. Наоборот, не требуется, чтобы все экземпляры объекта группы параметров имели соответствующий ввод «GroupN» в EDS. Однако если экземпляр объекта группы параметров существует в узле и если эта группа параметров также описывается в EDS, тогда значение «N» в «Group N» должно соответствовать экземпляру объекта группы параметров.

Поля в каждом вводе должны содержать название группы, число элементов в группе и номера экземпляров параметров в группе. Раздел группы параметров должен содержать поля, показанные в таблице А.22.

Таблица А.22 — Формат группы параметров

Имя поля	Номер поля	Тип данных	Обязательно/Дополнительно
Строка названия группы	1	EDS_Char_Array	Обязательно
Число элементов	2	UINT	Обязательно
Параметр	3 до (числа членов + 2)	UINT	Обязательно

На рисунке А.7 представлен пример, показывающий раздел типичных групп параметров.

```
[Groups]
    Group1 = "Setup", 2, 1, 2; $ group 1
    Group2 = "Monitor", 2, 2, 3; $ group 2
    Group3 = "Maintenance", 2, 1, 3; $ group 3
```

Рисунок А.7 — Пример раздела [Groups]

А.4.1.4.8 Раздел компоновки

Раздел компоновки описывает структуру блока данных. Часто такой блок представляет собой атрибут данных объекта Компоновка; однако данный раздел EDS может быть использован для описания любой сложной структуры. Описание этого блока данных соответствует описанию механизма, используемого объектом Компоновка для описания списка своих членов.

Ключевое слово ввода «Revision» имеет поле 16-битных целых чисел, которое должно быть версией (атрибут класса 1) объекта Компоновка в приборе. Если этот дополнительный ввод опущен, версия объекта Компоновка должна быть версией 2.

Ключевое слово ввода для всех компоновок должно состоять из одного из следующих наборов символов: «Assem», «ProxyAssem», «ProxiedAssem», объединенных с номером (десятичным) экземпляра объекта Компоновка для прибора, например «Assem1». Если отдельный экземпляр объекта Компоновка адресуется из канала связи, должно быть взаимно однозначное соответствие между номером Assem в файле EDS и номером экземпляра компоновки в устройстве. Ключевые слова «ProxyAssem» and «ProxiedAssem» определены далее в А.4.1.5.3.2 как часть требований EDS к модулям.

Каждый ввод должен содержать форматированные поля, показанные в таблице А.23.

Таблица А.23 — Формат ключевых слов AssemN

Имя поля	Номер поля	Тип данных	Обязательно/Дополнительно
Название	1	EDS_Char_Array	Дополнительно
Маршрут	2	EDS_Char_Array	Дополнительно
Размер	3	UINT	Условно
Дескриптор	4	WORD	Дополнительно
Зарезервированное	7, 9, 11 ...	Пустое	
Размер элемента	8, 10, 12 ...	UINT	Условно
Элементная ссылка		AssemN, ProxyAssemN, ParamN, ProxyParamN, UDINT, or EPATH	Условно

Первое поле «Название» должно быть строкой, дающей имя блоку данных. Это дополнительное поле может быть использовано интерфейсом пользователя.

Второе поле «Маршрут» должно быть строкой, устанавливающей логическую цепь. Этот маршрут должен идентифицировать адрес блока данных в устройстве. Если блок, описываемый этим вводом AssemN, не непосредственно адресуется из канала связи, то это поле должно быть пустым. Если это поле является пустой строкой, "", блок данных должен адресоваться как атрибут данных (атрибут экземпляра 3) N-го экземпляра объекта Компоновка.

Третье поле «Размер» должно устанавливать размер блока данных в байтах. Если ни это поле, ни поля «Member Size»/«Member Reference» не представлены, размер блока данных должен быть равен 0. Однако оба этих поля могут быть представлены, поскольку они оба устанавливают размер блока, размеры, устанавливаемые обоими способами, должны быть согласованы.

Четвертое поле «Дескриптор» должно быть битовым полем, описывающим некоторые характеристики Компоновки. Биты этого поля должны интерпретироваться, как установлено в таблице А.24.

Таблица А.24 — Определение битов поля дескриптора Компоновки

Бит	Название	Значение
0	Разрешение редактирования значений	Если этот бит установлен (1), содержимое полей элементных ссылок Компоновки, определенное как значение, может быть отредактировано. При сбросе (0) содержимое этих полей элементных ссылок может не редактироваться. Если это поле является пустым, значение должно по умолчанию сбрасываться (0). Элементные ссылки, рассматриваемые как значения, являются ссылками, устанавливающими либо константу UDINT, либо маршрут, состоящий из Сегментов Данных (Data Segments)
1—15		Зарезервированные

Поля 5 и 6 должны быть зарезервированы и должны быть пустыми.

Остающиеся поля должны быть спаренными так, чтобы поле «Member Size (Размер элемента)» спаривалось с полем «Member Reference (Элементная Ссылка)», делая полное число полей четным. Число пар полей в таком вводе должно быть переменным. Пары должны соответствовать списку элементов объекта Компоновка.

Разрешенные значения поля «Member Reference» должны быть одним из следующих значений:

- ссылка ParamN или ProxyParamN из раздела [Params];
- ссылка AssemN или ProxyAssemN из раздела [Assembly];
- строка, представляющая маршрут (EPATH);
- константа UDINT;
- пустое поле;
- дополнительные значения, установленные для модульной EDS в А.4.1.5.3.2.

Если поле «Member Reference» является пустым, число битов, установленных полем «Member Size», должно использоваться как дополнение в объекте Компоновка. Поле «Member Reference», содержащее пустую строку, следует рассматривать так, как если бы поле было пустым. Поле «Member Reference» и его соответствующее поле «Member Size» не должны быть одновременно пустыми. Если поле «Member Reference» устанавливает EPATH, то этот маршрут должен состоять либо из Логических сегментов (маршрут к объекту в приборе), либо из Сегментов данных.

Поле «Member Size» должно измеряться в битах. Если поле «Member Size» является пустым, должен использоваться установленный размер поля «Member Reference». Установленный размер ввода Param должен быть таким, как определено в его 6-м поле (размер). Установленный размер ввода Assem должен быть таким, как определено в его 3-м поле (размер).

Элементы должны размещаться в первом младшем бите блока данных, так же как и в объекте Компоновка. Если размер поля «Member Size» меньше установленного размера соответствующего поля «Member Reference», должны использоваться младшие биты соответствующего поля «Member Reference». Если размер поля «Member Size» больше установленного размера соответствующего поля «Member Reference», после всего элемента должны следовать дополняющие нули, позволяющие увеличить размер элемента до «Member Size». Размер представленного блока данных должен быть целым числом байтов. Размер всех элементов должен соответствовать размеру поля «AssemN Size» (когда выражается в битах).

На рисунке А.8 представлен пример, показывающий типичный раздел Компоновка. В этом примере длина Assem5 составляет 1 байт, а его значение по умолчанию определяется как 0x21.

```
[Params]
    Param1 =
            0,                          $ first field shall equal 0
            6, "20 0F 24 01 30 01",     $ path size, path
            0x0000,                     $ descriptor
            2,                          $ data type : 16-bit WORD
            2,                          $ data size in bytes
            "Idle state",               $ name
            "",                         $ units
            "User Manual p48",          $ help string
            0, 2, 1,                    $ min, max, default data values
            0, 0, 0, 0,                 $ mult, dev, base, offset scaling not used
            0, 0, 0, 0,                 $ mult, dev, base, offset link not used
            0;                          $ decimal places not used
    Param2 =
            0, 6, "20 0F 24 02 30 01",  $ path size, path
            0x0000, 2, 2,
            "Fault state", "", "User Manual p49", 0, 2, 2, 0, 0, 0, 0, 0, 0, 0, 0, 0;
[Assembly]
    Revision = 2;
    Assem5 = "configuration", "20 04 24 05 30 03",1,,,
                4, Param1,
                3, Param2,
                1, ;
```

Рисунок А.8 — Пример раздела [Assembly]

Примечание — Ключевое слов Variant, объединенное с десятичным числом (например, «Variant1»), резервируется для определения в дальнейшем новых типов вводов в разделе Компоновка.

А.4.1.4.9 Раздел управления соединениями

Настоящий раздел CIP не используется в файлах EDS DeviceNet, поэтому он не рассмотрен в настоящем стандарте.

A.4.1.4.10 Раздел портов

В настоящем разделе должны описываться маршрутизируемые порты CIP, имеющиеся в приборе. Каждый маршрутизируемый порт CIP, рассматриваемый в настоящем разделе, должен иметь соответствующий ввод. Ключевое слово ввода для всех портов должно состоять из набора символов «Port», объединенных с десятичным числом, соответствующим экземпляру объекта Порт. Например, Port1 соответствует экземпляру 1 объекта Порт.

Примечание — Маршрутизируемый порт CIP представляет порт обмена сообщениями с другим портом CIP, подключенным к другому каналу связи CIP.

Каждый ввод должен содержать форматированные поля, установленные в таблице А.25.

Таблица А.25 — Формат ввода Порта

Имя поля	Номер поля	Тип данных	Обязательно/Дополнительно
Название типа порта	1	Ключевое слово поля продавца	Обязательно
Название порта	2	EDS_Char_Array	Дополнительно
Объект Порт	3	EDS_Char_Array	Дополнительно
Число портов	4	UINT	Обязательно
Зарезервированное	5, 6	Пустое	Не используется
Специфика порта	7, 8, ...	Специфика порта	Специфика порта

Первое поле «Название типа порта» (Port Type Name) должно быть одним из следующих ключевых слов:

- ControlNet;
- ControlNet_Redundant;
- TCP (указывает порт TCP EtherNet/IP);
- DeviceNet;
- Ключевое слово поля продавца, начинающееся с Идентификатора продавца и символа подчеркивания ('65535_').

Дополнительное поле «Название порта» (Port Name) должно быть строкой, устанавливающей название порта, и может быть использовано интерфейсом пользователя. Поле «Объект порта» (Port Object) должно иметь маршрут (EPATH), который идентифицирует объект связи, относящийся к сети, ассоциированный с портом.

Номер порта 1 должен соответствовать «порту» задней панели. Устройства с задней панелью, которые не могут маршрутизировать сообщения CIP, не должны иметь порт номер 1.

На рисунке А.9 представлен пример, показывающий типичный раздел Порт.

```
[Port]
        Port1 = DeviceNet,
                "Port A",                   $ name of port
                "20 03 24 01",              $ instance one of the DeviceNet object
                2;                          $ port number 2
        Port2 = 65535_Chassis,
                "Chassis",                  $ name of port
                "20 9A 24 01",              $ vendor specific backplane object
                1;                          $ port number 1
```

Рисунок А.9 — Пример раздела [Port]

А.4.1.5 Требования к файлу EDS модульных устройств

A.4.1.5.1 Общие положения

Настоящий подраздел описывает концепцию и содержимое EDS модульных устройств и устанавливает требования к их применению.

A.4.1.5.2 Раздел модульных устройств Modular section

A.4.1.5.2.1 Содержимое

Раздел Modular section должен описывать систему на основе стойки. Должны быть использованы два типа модульных устройств:

- стойку;
- модуль.

A.4.1.5.2.2 Устройство на основе стоек

Раздел Modular section, описывающий стойку, должен содержать установленное ключевое слово «DefineSlotsInRack». Единственное поле этого ввода должно содержать 16-битовое целое число без знака (UINT), устанавливающее число слотов в стойке. Даже если для стойки устанавливается электронный ключ, он необязательно должен адресовываться из канала связи. Ключевое слово SLOT (Слот), используемое в определениях маршрута в разделе Connection Manager, должно находиться в диапазоне от 0 до числа слотов минус 1.

Ключевое слово «SlotDisplayRule» является дополнительным. Единственное поле этого ввода должно быть параметром из раздела {Params} (только ParamN), определяющим трансляцию между внутренними и внешними слотами.

На рисунке А.10 представлен пример, показывающий EDS для устройств на основе стойки, включая раздел модулей.

```
[File]
        DescText = "Wonder Chassis EDS file";
        CreateDate = 09-01-1997;
        CreateTime = 17:23:00;
        Revision = 1.1;
[Device]
        VendCode = 65535;
        VendName = "Widget Works, Inc.";
        ProdType = 101;
        ProdTypeStr = "Widget Works Generic";
        ProdCode = 1;
        MajRev = 1;
        MinRev = 1;
        ProdName = "Widget Chassis";
        Catalog = "1234-chassis";
[Params]
        Param1 =
                0,                          $ first field shall equal 0
                ,,                          $ path size,path
                0x0004,                     $ descriptor
                8,                          $ data type: 32-bit Unsigned Long Integer
                1,                          $ data size in bytes
                "Slot Naming Convention",   $ name
                "",                         $ units
                "",                         $ help string
                0,4,0,                      $ min,max,default data values
                0,0,0,0,                    $ mult,dev,base,offset scaling
                0,0,0,0,                    $ mult,dev,base,offset link not used
                0;                          $ decimal places not used
        Enum1 = 0,"n/a",1,"0",2,"1",3,"2",4,"3";
[Modular]
        DefineSlotsInRack = 5;
        SlotDisplayRule = Param1;
```

Рисунок А.10 — Раздел [Modular], описывающий стойку

А.4.1.5.2.3 Модульное устройство (основные вводы)

Раздел [Modular], описывающий модуль, должен содержать вводы «Width (Ширина)» и «Rack (стойка)».

Требуемый ввод с ключевым словом «Width» должен иметь единственное поле, которое устанавливает, сколько слотов в стойке используется модулем. Поле должно содержать 16 битное целое число без знака (UINT).

Ключевое слово ввода для всех стоек, в которых может быть размещен модуль, должно состоять из набора символов «Rack», объединенного с десятичным числом. Числа должны начинаться с 1 для первой стойки и должны увеличиваться на 1 для каждой дополнительной стойки. Поля для вводов «Rack» должны быть такими, как показано в таблице А.26.

Таблица А.26 — Формат ввода Rack (стойка)

Имя поля	Номер поля	Тип данных	Обязательно/Дополнительно
Vendor ID (Идентификатор продавца)	1	UINT	Обязательно
Product Type (Тип продукта)	2	UINT	Обязательно
Product Code (Код продукта)	3	UINT	Обязательно
Major Revision (Старшая версия)	4	USINT	Обязательно
Minor Revision (Младшая версия)	5	USINT	Обязательно
Зарезервированное	6, 7, 8	Пустое	Не используется
Legal Slot (Разрешенный слот)	9, 10, 11 ...	UINT	Обязательно

Поля «Vendor ID», «Product Type», «Product Cod» «Major Revision» и «Minor Revision» должны идентифицировать электронный ключ стойки, в которую может быть вставлен модуль. Зарезервированное поле должно быть пустым. Поля «Legal Slot» должны идентифицировать слоты, в которые может быть вставлен модуль. EDS для модуля должен содержать один ввод «Rack» для каждой стойки, в которой может быть размещен модуль.

Рисунок А.11 представляет пример, показывающий типичный раздел Modular.

```
[Modular]
    Width = 1;
    Rack1 =                                  $ this module can plug into
    65535, 101, 1, 1, 1,,,,                  $ slots 1, 2, 3 and 4 of
    1, 2, 3, 4;                              $ this five slot chassis
```

Рисунок А.11 — Пример раздела [Modular]

А.4.1.5.2.4 Модульный прибор (дополнительные вводы)

Общее представление

В EDS устанавливаются дополнительные вводы, позволяющие идентифицировать устройство и кодировать его для модулей в системах на основе стойки, которые не поддерживают CIP.

С этой целью модульные устройства обычно разделяют на две категории:

- модули, имеющие подключения канала связи CIP, соответствующий объект Идентичность, адресуемый из канала связи, и размещаемые в слоте 0 (например, коммуникационные адаптеры);
- модули, не имеющие подключения канала связи CIP или адресуемого объекта Идентичность, не могут быть размещены в слоте 0 (например, модули ввода/вывода).

Примечание — CIP предоставляет другие механизмы идентификации устройств и их кодирования для модулей, поддерживающих адресуемый объект Идентичность канала связи CIP.

Вводы для модуля, не имеющего адресуемого объекта Идентичность канала связи

Раздел Modular, описывающий модуль, не имеющий адресуемого объекта Идентичность канала связи, может содержать ключевое слово ввода «ExternalID». Ключевое слово должно иметь единственное поле. Это поле должно быть байтовой строкой, идентифицирующей модуль. Байтовая строка должна кодироваться с использованием того же формата, как и для ERATH.

На рисунке А.12 представлен пример, показывающий типичный раздел [Modular], который описывает модуль без адресуемого объекта Идентичность канала связи.

```
[Modular]
    Width = 1;
    Rack1 =                                  $ this module can plug into
        65535, 101, 1, 1, 1,,,,              $ slots 1, 2, 3 and 4 of
        1, 2, 3, 4;                          $ this five slot chassis
    Rack2 =
        65535, 101, 2, 1, 1,,,,
        1, 2, 3, 4, 5, 6, 7;
    ExternalID = "12 34";
```

Рисунок А.12 — Пример раздела Modular (модуль без адресуемого объекта Идентичность канала связи)

Вводы для модуля, имеющего подключение канала связи и размещенного в слоте 0

Раздел Modular, описывающий модуль, имеющий подключение канала связи и размещенный в слоте 0, может содержать любое из следующих ключевых слов ввода или комбинацию этих слов.

Ключевое слово «GenericID» должно иметь единственное поле. Это поле должно быть байтовой строкой, которая должна включаться в сегмент данных для соединения модуля вместо ExternalID, если кодирование модуля не требуется. Эта байтовая строка должна кодироваться с использованием того же формата, который был установлен для EPATH.

Ключевое слово «ExternIDExactMatch» должно иметь единственное поле со значением Yes (Да) или No (Нет). Значение Yes должно указывать, что ExternalID устанавливает одно конкретное устройство. Значение No должно указывать, что ExternalID устанавливает один из наборов совместимых устройств. Если ключевое слово «ExternIDExactMatch» пропущено, по умолчанию ExternalID должно устанавливать одно конкретное устройство.

Ключевое слово «Query (Запрос)» должно иметь четыре поля. Первое слово должно быть маршрутом, идентифицирующим адресуемый атрибут канала связи, который содержит набор внешних идентификаторов, по одному для каждого слота в стойке, за исключением слота 0. Второе поле должно быть сервисом для использования с маршрутом запроса (т. е. 1 — получить все атрибуты или 14 — получить один атрибут). Третье поле должно быть целым числом, определяющим число байтов, используемых для идентификации каждого модуля, и должно находиться в диапазоне 1—16. Если двойной модуль слота находится в стойке, внешний идентификатор модуля должен появляться дважды в наборе, возвращенном из запроса. Запрос должен быть адресован только модулю в слоте 0. Четвертое поле ExternalID должно быть возвращаемым, когда имеется пустой слот, кодируемый с использованием того же формата, который был установлен для EPATH.

На рисунке А.13 представлен пример, показывающий типичный раздел [Modular], описывающий модуль с подключением канала связи, размещенный в слоте 0.

```
[Modular]
    Width = 1;
    Rack1 =                                  $ this module can only plug into
    65535, 101, 1, 1, 1,,,,                  $ slot 0 of this five slot chassis
    0;
    Rack2 = 65535, 101, 2, 1, 1,,,,                  0;
    Query = "20 04 24 07 30 03",1,2,"FF FF";
    GenericID = "00 00";
    ExternalIDExactMatch = No;
```

Рисунок А.13 — Пример раздела [Modular] (модуль с подключением канала связи в слоте 0)

А.4.1.5.3 Добавление модулей в основные разделы EDS

А.4.1.5.3.1 Добавления в раздел Параметр

Ключевые слова «ProxyParam» и «ProxiedParam» должны быть использованы для описания параметров, замещаемых адаптерным устройством DeviceNet в другом приборе, который не поддерживает протокол CIP. Примером такого устройства является адаптерный модуль DeviceNet (устройство, замещающее соединение) в многослотной стойке ввода/вывода с аналоговым модулем ввода/вывода (устройство, замещающее соединение).

Ключевое слово «ProxyParam» должно быть в EDS для устройства, выполняющего роль «заместителя».

Ключевое слово «ProxiedParam» должно быть в EDS для устройства, выполняющего роль «заместителя».

Информация в разделе Modular должна быть использована для ассоциации файлов EDS, содержащих ключевые слова «ProxyParam», с файлами EDS, содержащими ключевые слова «ProxiedParam». Эта ассоциация должна существовать, если оба файла EDS устанавливают согласованный ввод для стойки (Rack).

Десятичное число (объединяемое с «ProxyParam» и «ProxiedParam») должно быть использовано для согласования «ProxyParam» с «ProxiedParam». Значения поля согласованной пары «ProxyParam» и «ProxiedParam» должны быть объединены для составления такой же информации по значению поля, которая существует в единственном вводе «Param». Это объединение должно быть выполнено с использованием значения поля из «ProxyParam», если только значение этого поля не определяется ключевым словом «Module». Если значение поля, установленное в «ProxyParam», определяется ключевым словом «Module», должно быть использовано значение поля, установленное в «ProxiedParam». Разрешается устанавливать значения поля для вводов «ProxiedParam», соответствующее значение поля которых в «ProxyParam» не определяется ключевым полем «Module», однако эти значения поля не должны быть использованы, они должны существовать только для документации.

В разделе Param также может существовать другое ключевое слово. Это ключевое слово должно быть использовано для предоставления минимального, максимального значений и значения по умолчанию, добавляемых к минимальному, максимальному значениям и к значению по умолчанию «ProxyParam». Это ключевое слово ввода должно иметь вид «ProxyParamSizeAdder» и должно быть объединено с десятичным числом из соответствующего ввода «ProxyParam». Определение этих полей согласовывается с определениями «Param». Ключевое слово «ProxyParamSizeAdder» предоставляет средства для адаптера на соединении модуля (например, «ProxyConnect»), добавляя данные по адаптеру в данные по модулю и возвращая объединенные данные по соединению.

В разделе [Param] также может существовать другое ключевое слово, соответствующее «ProxyParam», «ProxyEnum». «ProxyEnum» определяется так же, как и «Enum», за исключением того, что оно ассоциируется с «ProxyParam», а не с «Param». В разделе [Param] также может существовать второе ключевое слово, соответствующее «ProxiedParam», «ProxiedEnum». «ProxiedEnum» определяется так же, как и «Enum», за исключением того, что оно ассоциируется с «ProxiedParam», а не с «Param».

А.4.1.5.3.2 Добавления в раздел Компоновка

Дополнительные ключевые слова ввода

Ключевые слова «ProxyAssem» и «ProxiedAssem» должны быть использованы для описания компоновок замещаемым адаптерным устройством CIP в другом устройстве, которое не поддерживает протокол CIP. Примером такого устройства является адаптерный модуль DeviceNet (устройство, замещающее соединение) в многослотной стойке ввода/вывода с аналоговым модулем ввода/вывода (прибор, замещающий соединение).

Ключевое слово «ProxyAssem» должно существовать в EDS для устройства, выполняющего роль «заместителя».

Информация в разделе Modular должна быть использована для ассоциации файлов EDS, содержащих ключевые слова «ProxyAssem», с файлами EDS, содержащими ключевые слова «ProxiedAssem». Эта ассоциация должна существовать, если оба файла EDS устанавливают согласованный ввод для стойки (Rack).

Десятичное число (объединяемое с «ProxyAssem» и «ProxiedAssem») должно быть использовано для согласования «ProxyAssem» с «ProxiedAssem». Значения поля согласованной пары «ProxyAssem» и «ProxiedAssem» должны быть объединены для составления такой же информации по значению поля, которая

существует в единственном вводе «Assem». Это объединение должно быть выполнено с использованием значения поля из «ProxyAssem», если только значение этого поля не определяется одним из ключевых слов «Module» или «ModuleMemberList». Если значение поля, установленное в «ProxyAssem», определяется ключевым словом «Module», должно быть использовано значение поля, установленное в «ProxiedAssem». Значение поля «Module» не должно быть использовано для полей «Member Size» или «Member Reference». «ModuleMemberList» должно быть использовано только вместо пары полей «Member Size» и «Member Reference». Если значение поля, установленное в «ProxyAssem», определяется как «ModuleMemberList», должны быть использованы все поля «Member Size» и «Member Reference», установленные в «ProxiedAssem». Разрешается устанавливать значения поля для вводов «ProxiedAssem», соответствующее значение поля которых в «ProxyAssem» не определяется ключевым полем «Module», однако эти значения поля не должны быть использованы, они должны существовать только для документации.

Дополнительные ключевые слова полей

Адаптерное соединение стойки является подключением к стойке адаптерного устройства, содержащего данные из модулей в стойке. Такое соединение также может быть использовано для посылки данных по конфигурации и кодированию для модулей в стойке (например, при установлении соединения).

Следующие ключевые слова являются дополнительными значениями, учитывающими поле «Member Reference» в разделе Компоновка, которое определяет специальное назначение данных, установленных элементом компоновки:

- ExternalID;
- InputSlotMask0 или InputSlotMask1;
- OutputSlotMask0 или OutputSlotMask1;
- ConfigSlotMask0 или ConfigSlotMask1.

Ключевое слово «ExternalID» устанавливает, что этот элемент компоновки должен содержать либо значение «ExternalID» модульного устройства, если кодирование необходимо, либо значение «GenericID», устанавливаемое в EDS адаптера, если кодирование модуля не требуется.

Ключевое слово «ExternalID», объединенное с десятичным числом (например, ExternalID2), должно быть использовано для кодирования отдельного устройства для адаптерных соединений стойки. Десятичное (положительное) число N в «ExternalIDN» определяет слот N в стойке. Ключевое слово «ExternalIDN» устанавливает, что данный элемент компоновки должен содержать либо значение «ExternalID» модульного устройства для слота N, если кодирование устройства для этого слота необходимо, либо значение «GenericID», устанавливаемое в EDS адаптера, если кодирование модуля для этого слота не требуется.

Примечание — Кодирование недоступно для слота 0.

Ключевое слово «InputSlotMask0» или «InputSlotMask1» должно указывать местоположение входной щелевой маски в компоновке. Входная щелевая маска является набором битов, которые представляют включение или исключение исполнителя модуля в (из) данные (данных) инициатора в адаптерном соединении стойки. Если используется ключевое слово «InputSlotMask0», то бит 0 в этом наборе представляет слот 0, бит 1 представляет слот 1 и т. д. Если используется ключевое слово «InputSlotMask1», то бит 0 в этом наборе представляет слот 1, бит 1 представляет слот 2 и т. д. Ключевые слова «InputSlotMask0» и «InputSlotMask1» не должны быть использованы одновременно в одной и той же компоновке. Предшествующее поле «Member size» должно быть обязательным.

Ключевое слово «OutputSlotMask0» или «OutputSlotMask1» должно указывать местоположение выходной щелевой маски в компоновке. Выходная щелевая маска является набором битов, которые представляют включение или исключение исполнителя модуля в данные (из данных) инициатора в адаптерном соединении стойки. Если используется ключевое слово «OutputSlotMask0», то бит 0 в этом наборе представляет слот 0, бит 1 представляет слот 1 и т. д. Если используется ключевое слово «OutputSlotMask1», то бит 0 в этом наборе представляет слот 1, бит 1 представляет слот 2 и т. д. Ключевые слова «OutputSlotMask0» и «OutputSlotMask1» не должны быть использованы одновременно в одной и той же компоновке. Предшествующее поле «Member size» должно быть обязательным.

Ключевое слово «ConfigSlotMask0» или «ConfigSlotMask1» должно указывать местоположение щелевой маски конфигурации в компоновке. Щелевая маска конфигурации является набором битов, которые представляют включение или исключение данных по конфигурации модуля в сервис установления адаптерного соединения стойки. Если используется ключевое слово «ConfigSlotMask0», то бит 0 в этом наборе представляет слот 0, бит 1 представляет слот 1 и т. д. Если используется ключевое слово «ConfigSlotMask1», то бит 0 в этом наборе представляет слот 1, бит 1 представляет слот 2 и т. д. Ключевые слова «ConfigSlotMask0» и «ConfigSlotMask1» не должны быть использованы одновременно в одной и той же компоновке. Предшествующее поле «Member size» должно быть обязательным.

А.4.1.5.3.3 Добавления в раздел Управление соединениями

Данный раздел CIP не используется в файлах EDS DeviceNet, поэтому в настоящем стандарте он не рассмотрен.

А.4.1.5.3.4 Примеры расширенного раздела EDS (информативные)

На рисунках А.14 и А.15 представлены примеры, показывающие использование модульных расширений EDS для разделов Параметр и Компоновка.

```
[Params]
        Param1 = 0,,,0x0010,2,2," Target Error Codes", "","",0,0xFFFF,0,0,0,0,0,0,0,0,0,0;
        ProxyParam1 = 0,,,0x0000,2,2,"input size", "","",Module,Module,Module,0,0,0,0,,,,,0;
        ProxyParamSizeAdder1 = 4,4,4;
[Assembly]
        Assem1 = "connection input format",,,,,,32,Param1, ,ProxyAssem1, ,ProxyAssem2;
        ProxyAssem1 = "real time input format","20 7D 24 SLOT 30 0A",,,,, ModuleMemberList;
ProxyAssem2 = "real time status format","20 7D 24 SLOT 30 0B",,,,, ModuleMemberList;
```

Рисунок А.14 — Пример вводов ProxyParam и ProxyAssem

```
[Params]
        ProxiedParam1 = ,,,,,,"input size","","",0,2,2,,,,,,,,,;

[Assembly]
        ProxiedAssem1 = "real time input format",,,,,;
        ProxiedAssem2 = "real time status format",,,,,,16,;
```

Рисунок А.15 — Пример согласованных вводов ProxiedParam и ProxiedAssem

А.4.2 Требования к EDS для DeviceNet

А.4.2.1 Содержание ES для DeviceNet

Настоящий подраздел устанавливает требования к кодированию файлов Электронной Информационной Таблицы (EDS) для сетей на основе DeviceNet.

Таблица А.27 описывает структуру разделов, которые могут быть представлены в DeviceNet EDS, соответствующие разрешенные разграничители разделов и порядок этих разделов в EDS. Некоторые из этих разделов являются общими для нескольких сетей на основе CIP, их конкретное использование для DeviceNet установлено в А.4.2.2, если это необходимо. Другие разделы являются специфическими для DeviceNet и установлены в А.4.2.4.

Таблица А.27 — Структура файлов DeviceNet EDS

Название раздела	Разрешенный разграничитель	Размещение	Обязательно/ Дополнительно
Описание файлов	[File]	1	Обязательно
Описание устройств	[Device]	2	Обязательно
Классификация устройств	[Device Classification]	a)	Условно b)
Класс параметров	[ParamClass]	a)	Дополнительно
Параметры	[Params]	a)	Дополнительно
Группы параметров	[Groups]	a)	Дополнительно
Компоновка	[Assembly]	a)	Дополнительно
Характеристики соединений	[Connection Manager]	Не применяется	Не применяется
Порт	[Port]	a)	Дополнительно
Модульный	[Modular]	a)	Дополнительно
Нумерации параметров	[EnumPar]	a)	Дополнительно
Характеристики ввода/вывода	[IO_Info]	a)	Условно c)
Спецификация продавца	[VendorID_vendorspecifickeyword]	Последний	Дополнительно

a) При размещении этих групп необходимо только следовать разделам Описание устройств и Классификация устройств.

b) Данный раздел необходим, если включен модульный раздел, в противном случае он может быть пропущен.

c) Данный раздел необходим, если реализуется соответствующая функциональность, в противном случае он может быть пропущен.

Содержимое DeviceNet EDS должно быть организовано следующим образом:

- все файлы EDS DeviceNet, включающие в себя модульный раздел, должны также содержать раздел Классификация приборов, который должен использовать разрешенный разграничитель [Device Classification], и могут быть размещены где угодно после раздела Описание файлов;

- дополнительные и условные разделы, описанные в данной спецификации, могут быть представлены в любом порядке при условии, что в файле EDS отсутствуют ссылки вперед.

А.4.2.2 Реализация общих требований CIP

А.4.2.2.1 Раздел классификации устройств

Для любого совместимого устройства DeviceNet раздел классификации устройств их EDS, если он представлен, должен содержать не менее одного ввода ключевого слова ClassN с его первым полем, установленным на DeviceNet. Дополнительная подклассификация классификации DeviceNet должна быть зарезервирована.

А.4.2.2.2 Раздел Порт

В разделе Порт EDS ввод PortN, соответствующий совместимому порту DeviceNet, должен устанавливаться следующим образом:

- поле «Port Type Name (Название типа порта)» должно иметь значение «DeviceNet»;

- дополнительное поле «Port Object (объект Порт»)» должно быть установлено на маршруте объекта DeviceNet для этого порта;

- в полях «Port Name (Название порта)» и «Port Number (Число портов)» не размещаются никакие дополнительные требования, кроме требований, установленных в общем подразделе CIP (см. А.4.1.4.10).

А.4.2.3 Дополнительные требования к кодированию данных

Для файлов DeviceNet EDS дополнительные требования к кодированию данных не устанавливают.

А.4.2.4 Дополнительные требования к файлам

А.4.2.4.1 Раздел нумерованных строк параметров

Настоящий раздел предоставляет нумерованный список параметров, выбираемых пользователем. Ключевое слово ввода для всех нумерованных строк параметров должно состоять из набора символов «Param», объединенного с номером (десятичным) экземпляра параметров для прибора. Каждый параметр, поддерживающий нумерованные строки и не имеющий соответствующего ввода Enum в разделе Параметры, должен иметь ввод в этом разделе. Вводы должны быть расположены в возрастающем порядке.

Поля в каждом вводе должны содержать нумерованные строки, разделенные запятыми. Положение в списке определяет нумерованное значение, присваемое строке. Первой строке в нумерованном списке присваивается минимальное значение параметра. Каждой последующей строке присваивается значение, увеличенное на единицу. Для каждого целого значения от минимального значения параметра до максимального значения параметра должна быть использована одна строка.

Каждый ввод в разделе Нумерованные строки параметров должен содержать форматированные поля, показанные в таблице А.28.

Таблица А.28 — Формат нумерованных строк параметров

Имя поля	Номер поля	Тип данных	Обязательно/Дополнительно
Нумерованная строка (нумерованные строки)	1 — число нумераций	EDS_Char_Array	Обязательно

На рисунке А.16 представлен пример, показывающий типичный раздел нумерованных строк параметров.

```
[EnumPar]
        Param2 =                    $ enums for param 2
          "1 ms input delay",
          "10 ms input delay",
          "25 ms input delay";

Param3 =                            $ enums for param 3
          "Light Operate",
          "Dark Operate";
```

Рисунок А.16 — Пример раздела [EnumPar]

А.4.2.4.2 Раздел характеристик ввода/вывода

А.4.2.4.2.1 Содержимое

Раздел характеристик ввода/вывода должен содержать информацию о возможностях ввода/вывода заранее определенного набора основных/подчиненных соединений прибора (см. МЭК 62026-3:2000). Этот раздел является обязательным, если прибор реализует соединение ввода/вывода, характеристики которого могут быть полностью представлены одним или несколькими вводами в этом разделе.

В данном разделе должны быть описаны следующие характеристики ввода/вывода прибора:
- тип триггера ввода/вывода (Опрос, Строб, Изменение состояния, Циклический);
- размер сообщения ввода/вывода (число байтов);
- заранее установленные маршруты соединений ввода/вывода.

Раздел характеристик ввода/вывода должен содержать вводы, установленные в таблице А.29.

Таблица А.29 — Формат характеристик ввода/вывода

Имя ввода	Ключевое слово ввода	Обязательно/Дополнительно
Информация ввода/вывода по умолчанию	Default	Обязательно
Информация по опросам	PollInfo	Условно
Информация по стробам	StrobeInfo	Условно
Информация по многоадресным запросам	MulticastPollInfo	Условно
Информация по изменению состояния	COSInfo	Условно
Информация по циклам	CyclicInfo	Условно
Соединение выхода устройства	Input	Условно
Соединение входа устройства	Output	Условно

Вводы в разделе характеристик ввода/вывода должны предоставлять информацию, как показано в таблице А.30.

Таблица А.30 — Вводы характеристик ввода/вывода

Вводы	Описание
Информация ввода/вывода по умолчанию	Определяет информацию ввода/вывода по умолчанию
Информация по опросам	Определяет типы ввода/вывода, которые могут быть использованы с соединением с опросами и с соединениями выхода и входа опросов по умолчанию. Если этот ввод не представлен, опрашиваемое соединение в приборе не поддерживается
Информация по стробам	Определяет типы ввода/вывода, которые могут быть использованы вместе со строб-соединением и строб-соединениями выхода и входа по умолчанию. Если этот ввод не представлен, строб-соединение в этом устройстве не поддерживается
Информация по изменению состояния	Определяет типы ввода/вывода, которые могут быть использованы вместе с соединением Изменение состояния и с соединениями выхода и входа Изменения состояния по умолчанию . Если этот ввод не представлен, соединение Изменение состояния в устройстве не поддерживается
Информация по циклам	Определяет типы ввода/вывода, которые могут быть использованы с циклическим соединением вместе с циклическими соединениями выхода и входа по умолчанию. Если этот ввод не представлен, циклическое соединение в приборе не поддерживается
Информация по многоадресным запросам	Определяет типы ввода/вывода, которые могут быть использованы с соединением многоадресных опросов и с соединениями выхода и входа многоадресных опросов по умолчанию . Если этот ввод не представлен, соединение многоадресных опросов в приборе не поддерживается
Соединение выхода прибора	Определяет размеры соединения в байтах и битах, которые могут быть использованы типами ввода/вывода, маршрут соединения, строку имени и строку справки
Соединение входа прибора	Определяет размеры соединения в байтах и битах, которые могут быть использованы типами ввода/вывода, маршрут соединения, строку имени и строку справки

А.4.2.4.2.2 Ввод информации ввода/вывода по умолчанию

Ввод информации ввода/вывода по умолчанию должен определять информацию ввода/вывода прибора по умолчанию.

Ввод информации ввода/вывода по умолчанию должен содержать форматированные поля, представленные в таблице А.31.

Таблица А.31 — Формат информации ввода/вывода по умолчанию

Имя поля	Номер поля	Тип данных	Обязательно/Дополнительно
Маска типа ввода/вывода по умолчанию	1	WORD	Обязательно

Поле маски типа ввод/вывода по умолчанию должно быть битовым отображаемым полем, позволяющим определять типы ввода/вывода по умолчанию для устройства. Присвоение битов установлено в таблице А.32.

Таблица А.32 — Присвоение битов маски типа ввода/вывода по умолчанию

Бит	Определение бита
0	Опрос
1	Строб
2	Изменение состояния
3	Циклический
4	Многоадресный опрос
5—15	Зарезервированный

А.4.2.4.2.3 Ввод информации по опросам

Ввод информации по опросам должен определять типы ввода/вывода, которые могут быть использованы вместе с соединением с опросами. Он также должен устанавливать соединения выхода и входа опросов по умолчанию.

Если данный ввод не представлен, тогда:

- соединение с опросами в приборе не поддерживается или
- соединение не может быть полностью описано логической структурой этого ключевого слова.

Ввод информации по опросам должен содержать форматированные поля, показанные в таблице А.33.

Таблица А.33 — Формат информации по опросам

Имя поля	Номер поля	Тип данных	Обязательно/Дополнительно
Совместимая маска типа ввода/вывода	1	WORD	Обязательно
Соединение выхода по умолчанию	2	UINT	Обязательно
Соединение входа по умолчанию	3	UINT	Обязательно

Ввод информации по опросам должен предоставлять информацию, как показано в таблице А.34.

Таблица А.34 — Поля информации по опросам

Поля	Описание
Совместимая маска типа ввода/вывода	Битовое отображаемое поле определяет типы ввода/вывода, с которыми может быть объединено опрашиваемое соединение. Присвоение битов должно быть таким же, как и для маски типа ввода/вывода по умолчанию (см. таблицу А.32). Должен быть установлен бит опроса

Окончание таблицы А.34

Поля	Описание
Соединение выхода по умолчанию	Устанавливает один из вводов соединения выхода по умолчанию для опрашиваемого соединения. Нуль должен указывать, что для соединения с опросами выходные данные отсутствуют
Соединение входа по умолчанию	Устанавливает один из вводов соединения входа по умолчанию для опрашиваемого соединения. Нуль должен указывать, что для соединения с опросами входные данные отсутствуют

А.4.2.4.2.4 Ввод информации по стробам

Ввод информации по стробам должен определять типы ввода/вывода, которые могут быть использованы вместе со строб-соединением. Он также должен устанавливать строб-соединения выхода и входа по умолчанию.

Если данный ввод не представлен, тогда:

- строб-соединение в приборе не поддерживается или
- соединение не может быть полностью описано логической структурой этого ключевого слова.

Ввод информации по стробам должен содержать форматированные поля, показанные в таблице А.35.

Таблица А.35 — Формат информации по стробам

Имя поля	Номер поля	Тип данных	Обязательно/Дополнительно
Совместимая маска типа ввода/вывода	1	WORD	Обязательно
Соединение выхода по умолчанию	2	UINT	Обязательно
Соединение входа по умолчанию	3	UINT	Обязательно

Ввод информации по стробам должен предоставлять информацию, как показано в таблице А.36.

Таблица А.36 — Поля информации по стробам

Поля	Описание
Совместимая маска типа ввода/вывода	Битовое отображаемое поле определяет типы ввода/вывода, с которыми может быть объединено строб-соединение. Присвоение битов должно быть таким же, как и для маски типа ввода/вывода по умолчанию (см. таблицу А.32). Должен быть установлен битовый строб
Соединение выхода по умолчанию	Устанавливает один из вводов соединения выхода по умолчанию для строб-соединения. Нуль должен указывать, что для строб-соединения выходные данные отсутствуют
Соединение входа по умолчанию	Устанавливает один из вводов соединения входа по умолчанию для строб-соединения. Нуль должен указывать, что для строб-соединения входные данные отсутствуют

А.4.2.4.2.5 Ввод информации по изменению состояния

Ввод информации по изменению состояния должен определять типы ввода/вывода, которые могут быть использованы вместе с соединением изменения состояния. Он также должен устанавливать соединения входа и выхода изменения состояния по умолчанию.

Если данный ввод не представлен, тогда:

- соединение Изменение Состояния в приборе не поддерживается;
- соединение не может быть полностью описано логической структурой этого ключевого слова.

Ввод информации по изменению состояния должен содержать форматированные поля, показанные в таблице А.37.

Таблица А.37 — Формат информации по изменению состояния

Имя поля	Номер поля	Тип данных	Обязательно/Дополнительно
Совместимая маска типа ввода/вывода	1	WORD	Обязательно
Соединение выхода по умолчанию	2	UINT	Обязательно
Соединение входа по умолчанию	3	UINT	Обязательно

Ввод информации по изменению состояния должен предоставлять информацию, как показано в таблице А.38.

Таблица А.38 — Поля информации по изменению состояния

Поля	Описание
Совместимая маска типа ввода/вывода	Битовое отображаемое поле определяет типы ввода/вывода, с которыми может быть объединено соединение Изменение Состояния. Присвоение битов должно быть таким же, как и для маски типа ввода/вывода по умолчанию (см. таблицу А.32). Должен быть установлен бит изменения состояния
Соединение выхода по умолчанию	Устанавливает один из вводов соединения выхода по умолчанию для соединения Изменение Состояния. Нуль должен указывать, что для соединения Изменение Состояния выходные данные отсутствуют
Соединение входа по умолчанию	Устанавливает один из вводов соединения ввода по умолчанию для соединения Изменение Состояния. Нуль должен указывать, что для соединения Изменение Состояния входные данные отсутствуют

А.4.2.4.2.6 Ввод информации по циклам

Ввод информации по циклам должен определять типы ввода/вывода, которые могут быть использованы вместе с циклическим соединением. Он также должен устанавливать циклические соединения выхода и входа по умолчанию.

Если данный ввод не представлен, тогда:

- циклическое соединение в устройстве не поддерживается;
- соединение не может быть полностью описано логической структурой этого ключевого слова.

Ввод информации по циклам должен содержать форматированные поля, показанные в таблице А.39.

Таблица А.39 — Формат информации по циклам

Имя поля	Номер поля	Тип данных	Обязательно/Дополнительно
Совместимая маска типа ввода/вывода	1	WORD	Обязательно
Соединение выхода по умолчанию	2	UINT	Обязательно
Соединение входа по умолчанию	3	UINT	Обязательно

Ввод информации по циклам должен предоставлять информацию, как показано в таблице А.40.

Таблица А.40 — Поля информации по циклам

Поля	Описание
Совместимая маска типа ввода/вывода	Битовое отображаемое поле определяет типы ввода/вывода, с которыми может быть объединено циклическое соединение. Присвоение битов должно быть таким же, как и для маски типа ввода/вывода по умолчанию (см. таблицу А.32). Должен быть установлен бит циклов

Окончание таблицы А.40

Поля	Описание
Соединение выхода по умолчанию	Устанавливает один из вводов соединения выхода по умолчанию для циклического соединения. Нуль должен указывать, что для циклического соединения выходные данные отсутствуют
Соединение входа по умолчанию	Устанавливает один из вводов соединения входа по умолчанию для циклического соединения. Нуль должен указывать, что для циклического соединения входные данные отсутствуют

А.4.2.4.2.7 Ввод информации по многоадресным опросам

Ввод информации по многоадресным опросам должен определять типы ввода/вывода, которые могут быть использованы вместе с соединением с многоадресными опросами. Он также должен устанавливать соединения вывода и ввода многоадресных опросов по умолчанию.

Если данный ввод не представлен, тогда:

- соединение с многоадресными опросами в приборе не поддерживается;
- соединение не может быть полностью описано логической структурой этого ключевого слова.

Ввод информации по многоадресным опросам должен содержать форматированные поля, показанные в таблице А.41.

Таблица А.41 — Формат информации по многоадресным опросам

Имя поля	Номер поля	Тип данных	Обязательно/Дополнительно
Совместимая маска типа ввода/вывода	1	WORD	Обязательно
Соединение выхода по умолчанию	2	UINT	Обязательно
Соединение входа по умолчанию	3	UINT	Обязательно

Ввод информации по многоадресным опросам должен представлять информацию, как показано в таблице А.42.

Таблица А.42 — Поля информации по многоадресным опросам

Поля	Описание
Совместимая маска типа ввода/вывода	Битовое отображаемое поле определяет типы ввода/вывода, с которыми может быть объединено соединение с многоадресными опросами. Присвоение битов должно быть таким же, как и для маски типа ввода/вывода по умолчанию (см. таблицу А.32). Должен быть установлен бит многоадресных опросов
Соединение выхода по умолчанию	Устанавливает один из вводов соединения выхода по умолчанию для соединения с многоадресными опросами. Нуль должен указывать, что для соединения с многоадресными опросами выходные данные отсутствуют
Соединение входа по умолчанию	Устанавливает один из вводов соединения входа по умолчанию для соединения с многоадресными опросами. Нуль должен указывать, что для соединения с многоадресными опросами входные данные отсутствуют

А.4.2.4.2.8 Вводимые данные соединений выходов устройства

Вводимые данные соединений выходов устройства должны быть использованы для определения логически кодированного маршрута к данным одного или нескольких объектов, которые могут быть созданы соединением ввода/вывода.

Отдельный ввод данных необходим для каждого случая, если:
- он может быть установлен в маршруте соединения выхода;
- он может быть описан в файле EDS.

Каждый ввод должен определять размер соединения в байтах и битах, если он использует типы ввода/вывода, маршрут соединения, строку имен и строку справки.

Ключевое слово для вводов соединений выхода должно состоять из комбинации набора символов «input», объединенных с десятичным числом, например «input1». Десятичное число должно начинаться с единицы и увеличиваться на единицу.

Каждый ввод соединения выхода должен содержать форматированные поля, как показано в таблице А.43.

Таблица А.43 — Формат соединения выхода

Имя поля	Номер поля	Тип данных	Обязательно/ Дополнительно
Размер	1	UINT	Обязательно
Число значимых битов	2	UINT	Обязательно
Совместимая маска типа ввода/вывода	3	WORD	Обязательно
Строка имен	4	EDS_Char_Array	Обязательно
Размер маршрута соединения	5	UINT	Обязательно
Маршрут соединения	6	EPATH	Обязательно
Строка справки	7	EDS_Char_Array	Обязательно

Ввод соединения выхода должен предоставлять информацию, показанную в таблице А.44.

Таблица А.44 — Поля соединений выхода

Поля	Описание
Размер	Размер этих данных в байтах
Число значимых битов	Число последовательных значимых битов этих данных (биты, которые фактически используются). Нуль должен указывать, что все биты данных являются значимыми
Совместимая маска типа ввода/вывода	Битовое отображаемое поле, определяющее типы ввода/вывода, с которыми эти данные могут быть использованы. Присвоение битов должно быть таким же, как и для маски типа ввода/вывода по умолчанию (см. таблицу А.32)
Строка имен	Текстовое имя данных (максимум 32 символа). При воспроизведении в зависимости от возможностей дисплея средств конфигурирования может возникнуть необходимость усечения текста
Размер маршрута соединения	Число байтов, используемых для представления маршрута
Маршрут соединения	Маршрут данных соединения. Маршрут должен вводиться как набор символов с использованием нотации маршрута, описанной в МЭК 62026-3:2000, и формата, установленного в А.4.1.3.6.
Строка справки	Текстовая строка справки. При воспроизведении в зависимости от возможностей дисплея средств конфигурирования может возникнуть необходимость усечения текста.

А.4.2.4.2.9 Вводимые данные соединений входов устройства

Вводимые данные соединений входов устройства должны быть использованы для определения логически кодированного маршрута из данных одного или нескольких объектов, которые могут приниматься соединением ввода/вывода.

Отдельный ввод необходим для каждого случая, если:
- он может быть установлен в маршруте соединения входа;
- он может быть описан в файле EDS.

Каждый ввод должен определять размер соединения в байтах и битах, если он использует типы ввода/вывода, маршрут соединения, строку имен и строку справки.

Ключевое слово для вводов соединений входов должно состоять из комбинации набора символов «Output», объединенных с десятичным числом, например «Output1». Десятичное число должно начинаться с единицы и увеличиваться на единицу.

Каждый ввод соединения входа должен содержать форматированные поля, как показано в таблице А.45.

Таблица А.45 — Формат соединения входа

Имя поля	Номер поля	Тип данных	Обязательно/ Дополнительно
Размер	1	UINT	Обязательно
Число значимых битов	2	UINT	Обязательно
Совместимая маска типа ввода/вывода	3	WORD	Обязательно
Строка имен	4	EDS_Char_Array	Обязательно
Размер маршрута соединения	5	UINT	Обязательно
Маршрут соединения	6	EPATH	Обязательно
Строка справки	7	EDS_Char_Array	Обязательно

Ввод соединения входа должен предоставлять информацию, представленную в таблице А.46.

Таблица А.46 — Поля соединений входа

Поля	Описание
Размер	Размер этих данных в байтах. Для строб-соединений единица должна устанавливать, что строб-бит используется, а нуль должен устанавливать, что он не используется
Число значимых битов	Число последовательных значимых битов этих данных (биты, которые фактически используются). Нуль должен указывать, что все биты данных являются значимыми
Совместимая маска типа ввода/вывода	Битовое отображаемое поле, определяющее типы ввода/вывода, с которыми эти данные могут быть использованы. Присвоение битов должно быть таким же, как и для маски типа ввода/вывода по умолчанию (см. таблицу А.32)
Строка имен	Текстовое имя данных (максимум 32 символа). При воспроизведении в зависимости от возможностей дисплея средств конфигурирования может возникнуть необходимость усечения текста
Размер маршрута соединения	Число байтов, используемых для представления маршрута
Маршрут соединения	Маршрут данных соединения. Маршрут должен вводиться как набор символов с использованием нотации маршрута, описанной в МЭК 62026-3:2000, и формата, установленного в А.4.1.3.6
Строка справки	Текстовая строка справки. При воспроизведении в зависимости от возможностей дисплея средств конфигурирования может возникнуть необходимость усечения текста

А.4.2.4.2.10 Примеры

Настоящий подраздел содержит примеры разделов файлов EDS, устанавливающие характеристики ввода/вывода устройств.

Рисунок А.17 представляет пример, показывающий типичный раздел характеристик ввода/вывода для прибора с возможностями опросов и стробов (Опрос по умолчанию).

```
[IO_Info]
    Default = 0x0001;                    $ Poll
                                         $ Bit mapped (0 = None)
                                         $ Bit 0 = Poll
                                         $ Bit 1 = Strobe
```

```
    PollInfo =
      0x0003,                          $ OK to Combine w/Poll or Strobe
      2,                               $ Default Input = Input2
      3;                               $ Default Output = Output3
    StrobeInfo =
      0x0003,                          $ OK to Combine w/Poll or Strobe
      1,                               $ Default Input = Input1
      4;                               $ Default Output = Output4
$ - - Input Connections - -
    Input1 =
      1,                               $ 1 byte
      1,                               $ 1 bit is significant
      0x0003,                          $ Strobe or Poll Connection
      "Data",                          $ Name String
      6,                               $ Path Size
      "20 04 24 20 30 03",             $ Assy Obj Inst 20 Attr 3
      "This is a help string.";        $ Help String
    Input2 = 2, 0, 0x0003, "Status", 6, "20 04 24 21 30 03", "This is a help string.";
    Input3 = 3, 0, 0x0003, "Data + Status", 6, "20 04 24 22 30 03", "This is a help string.";
$ - - Output Connections —
    Output1 =
      1,                               $ 1 byte
      0,                               $ All bits are significant
      0x0001,                          $ Poll Only Connection
      "Data",                          $ Name String
      6,                               $ Path Size
      "20 04 24 10 30 03",             $ Assy Obj Inst 10 Attr 3
      "This is a help string.";        $ Help String
    Output2 = 2, 0, 0x0001, "Reference", 6, "20 04 24 11 30 03", "This is a help string.";
    Output3 = 3, 0, 0x0001, "Data + Reference",6, "20 04 24 12 30 03", "This is a help string.";
    Output4 =
      1,                               $ Use Strobe Bit
      1,                               $ 1 significant bit
      0x0002,                          $ Strobe Only Connection
      "Data Bit",                      $ Name String
      4,                               $ Path Size
      "20 0E 24 01",                   $ Sensor Obj Inst 1
      "This is a help string.";        $ Help String
    Output5 =
      1,                               $ Use Strobe Bit
      1,                               $ 1 significant bit
      0x0002,                          $ Strobe Only Connection
      "Sync Bit",                      $ Name String
      4,                               $ Path Size
      "20 0E 24 02",                   $ Sensor Obj Inst 2
      "This is a help string.";        $ Help String
    Output6 =
      0,                               $ Strobe Bit Ignored, Don't Map
      0,                               $ All bits are significant (None)
      0x0002,                          $ Strobe Only Connection
      "Sync Pulse",                    $ Name String
      4,                               $ Path Size
      "20 0E 24 03",                   $ Sensor Obj Inst 3
      "This is a help string.";        $ Help String
```

Рисунок А.17 — Пример характеристик ввода/вывода.
Устройство с возможностями опросов и стробов. Опрос по умолчанию

Рисунок А.18 представляет пример, показывающий типичный раздел характеристик ввода/вывода для устройства Strobe Only (только один вход и один выход).

```
[IO_Info]
    Default = 0x0002;                    $ Strobe Only
    StrobeInfo =
    0x0002                               $ Strobe Only
    1;                                   $ Default Input = Input1
    1;                                   $ Default Output = Output1
$ - - Input Connections - -
  Input1 =
    1,                                   $ 1 byte
    2,                                   $ 2 bits are significant
    0x0002,                              $ Only Strobe Connection
    "Data & Status Bits",                $ Name String
    4,                                   $ Path Size
    "20 0E 24 01",                       $ Sensor Obj Inst 1
    "This is a help string.";            $ Help String
$ - - Output Connections - -
  Output1=
    0,                                   $ Strobe bit not used
    0,                                   $ All bits are significant
    0x0002,                              $ Only Strobe Connection
    "Strobe Out Bit Not Used",           $ Name String
    6,                                   $ Path Size
    "20 0E 24 01",                       $ Sensor Obj Inst 1
    "This is a help string.";            $ Help String
```

Рисунок А.18 — Пример характеристик ввода/вывода для устройства Strobe Only, только один вход и один выход

Рисунок А.19 представляет пример, показывающий типичный раздел характеристик ввода/вывода для устройства с возможностями опросов и изменения состояния (Опросы и изменение состояния по умолчанию).

```
[IO_Info]
  Default = 0x0005;                      $ Poll & Change of State
                                         $ Bit 0 = Poll
                                         $ Bit 2 =Change of State
  PollInfo =
    0x0005,                              $ OK to Combine w/Poll or COS
    2,                                   $ Default Input = Input2
    2;                                   $ Default Output = Output2
  COSInfo =
    0x0005,                              $ OK to Combine w/Poll or COS
    1,                                   $ Default Input = Input1
    1;                                   $ Default Output = Output1
$ - - Input Connections - -
   Input1 =
    1,                                   $ 1 byte
    1,                                   $ 1 bit is significant
    0x0005,                              $ Poll or COS Connection
    "Data",                              $ Name String
    6,                                   $ Path Size
    "20 04 24 20 30 03",                 $ Assy Obj Inst 20 Attr 3
    "This is a help string."; $ Help String
  Input2 = 2, 0, 0x0005, "Status", 6, "20 04 24 21 30 03", "This is a help string.";
  Input3 = 3, 0, 0x0005, "Data + Status", 6, "20 04 24 22 30 03", "This is a help string.";
$ - - Output Connections - -
  Output1 =
    1,                                   $ 1 byte
    0,                                   $ All bits are significant
    0x0005,                              $ Poll or COS Connection
    "Data",                              $ Name String
    6,                                   $ Path Size
    "20 04 24 10 30 03",                 $ Assy Obj Inst 10 Attr 3
    "This is a help string.";            $ Help String
  Output2 = 2, 0, 0x0005, "Reference", 6, "20 04 24 11 30 03", "This is a help string.";
  Output3 = 3, 0, 0x0005, "Data + Reference",6, "20 04 24 12 30 03", "This is a help string.";
```

Рисунок А.19 — Пример характеристик ввода/вывода. Устройство с возможностями опросов и изменения состояния. Опросы и изменение состояния по умолчанию

Приложение В
(обязательное)

Шаблоны профилей CANopen

В.1 Описание шаблонов профилей устройств

В.1.1 Общие положения

XML схемы шаблонов профилей устройств, определенные в В.1.5, содержат отображения диаграмм классов профилей устройств, представленных в 6.2.1. Помимо классов и атрибутов отображения они содержат дополнительные элементы с атрибутами XML или без них, облегчающие однозначное описание профилей устройств и устройств XML. Для повторного использования некоторых определений элементов и гибкого расширения описаний профиля устройств и устройств с использованием моделей, используемых здесь, классы были отражены в нескольких XML схемах. В таблице В.1 перечислены все XML схемы описания обмена шаблонами профилей устройств CANopen.

Примечание – Описание профилей устройств или описание устройств может не потребоваться для применения всех XML схем, установленных в настоящем приложении.

Таблица В.1 — Обзор XML схем

Имя	Содержимое	Пространство имен
COFDCML.xsd	Основные классы	http://www.can-cia.org/xml/canopen
FDCMLdt.xsd	Определение типов данных	gost.e-ecolog.ru
FDCMLИСО15745Device Function.xsd	Классы функций приборов	http://gost.e-ecolog.ru unction
FDCMLText Resource.xsd	XML схема для текстовых ресурсов	gost.e-ecolog.ru
xmldef.xsd	Определения в пространстве имен XML	gost.e-ecolog.ru
xlinkdef.xsd	Определения в пространстве имен Xlink	gost.e-ecolog.ru xlink

В.1.2 Основы

В.1.2.1 Элементы типов данных

Профиль устройств использует типы данных МЭК 61158. Для введения дополнительных атрибутов информация по типам данных моделируется с использованием объявлений типов элементов XML. Альтернативные имена типов данных МЭК 61131-3 и CANopen предоставляются как постоянные атрибуты. Эти элементы типов данных определяются с именем «FDCMLdt.xsd» в XML схеме, которая определяется в В.1.5.2.

В.1.2.2 Сборная модель

Профиль устройств использует модель сборных элементов. Эти элементы фигурируют в форме списка сборных элементов (пример: processDataDescriptionList/processDataDescription).

В.1.2.3 Описательный текст для элементов

В.1.2.3.1 Общие положения

Профиль устройств предлагает три разные возможности предоставления описательного текста для элементов. Должны быть использованы только В.1.2.3.2 и В.1.2.3.3. В.1.2.3.4 может быть использован вместе с В.1.2.3.2 или В.1.2.3.3.

В.1.2.3.2 Текст, вставленный в профиль устройств

Каждый элемент, требующий описательного текста, должен иметь дочерний элемент label с атрибутом xml:lang. Это позволяет использовать множество языков с одним профилем приборов. Дополнительно текст короткой справки может быть предоставлен с использованием элемента help. Он также имеет атрибут xml:lang. Этот атрибут состоит из двухбуквенного кода языка и дополнительного двухбуквенного кода страны, разделенных тире, и имеет следующий формат:

ISO 639 code for name of language ["-" ISO 3166-1-Alpha-2 code]

Пример — xml:lang='en-us' устанавливает текст на английском языке в американской редакции.

B.1.2.3.3 Текст, предоставляемый файлами текстовых ресурсов

Элементы labelRef и helpRef должны предоставлять указатель текстового ресурса, хранящегося в файле внешнего тестового ресурса. AIF должна заменять эти ресурсы посредством теста, предусмотренного файлом текстовых ресурсов. Файлы текстовых ресурсов, используемые профилем приборов, должны быть определены в элементе dictionary.

Формат файла тестовых ресурсов должен быть установлен в XML схеме «FDCMLTextResource.xsd», которая определена в B.1.5.4.

B.1.2.3.4 Указатель внешней документации

Дополнительно может быть предоставлен указатель внешней документации. Элемент hlpRefFile должен указывать положение в файле, определяемое элементом helpFile.

Пример — Примерами внешней документации являются файлы *.hlp, *.html, или *.pdf.

B.1.2.4 Допустимое значение элемента

Если элемент профиля устройств имеет значение, то должно использоваться одно из следующих описаний значения:

- const — постоянное значение элемента;
- edit — редактируемое значение строк;
- enumeration — нумерованное значение;
- range — диапазон значений, значение может иметь несколько диапазонов;
- yes, no — комбинация значений TRUE/FALSE;
- reference — ссылка на другой элемент;
- instanceValue — значение, если вместо профиля или типа описывается экземпляр.

B.1.2.5 Моделирование обусловленного поведения устройств

Для моделирования обусловленного поведения устройств должны быть использованы следующие элементы:

- disable — запрещает справочную характеристику устройства в зависимости от значения этой характеристики;
- enable — разрешает справочную характеристику устройства в зависимости от значения этой характеристики;
- change — изменяет справочную характеристику устройства в зависимости от значения этой характеристики.

Каждый возможный запрещаемый или разрешаемый целевой объект имеет в XML схеме разрешаемый атрибут.

B.1.2.6 Внутренние обращения элементов

Маршрут, установленный в атрибуте ref, должен быть допустимым маршрутом Xpath (см. [8]).

B.1.2.7 Однозначная идентификация элементов

Элементы, которые требуют идентификации, имеют атрибут uniqueID. Значение этого однозначного идентификатора должно состоять из:

token_index [[_subindex]...], индекса типа unsigned16 и субиндекса типа unsigned8.

B.1.2.8 Компоновки

Элементы компоновки позволяют сгруппировать соответствующие элементы объектов. Компоновка содержит список указателей Xpath своих объектов.

B.1.2.9 Категории со стороны продавца

Разработчик AIP может добавить элементы категорий в сборные элементы.

Пример — Разработчик AIP добавляет processDataCategory для входных сигналов и processDataCategory для выходных сигналов.

B.1.3 Объект DeviceManager

B.1.3.1 Объекты datatypeTemplateList, datatypeTemplate

Объект datatypeTemplate позволяет определить разработчика AIP или типы данных, относящиеся к профилям устройств. Эти типы данных вызываются с элементом datatypeInstance. Они могут быть описаны следующим образом:

- directlyDerivedType — тип непосредственно выводимых данных;
- enumeratedType — тип нумерованных данных или список констант (нумерация C-style);
- subrangeType — тип данных диапазона;
- arrayType — тип массива данных;
- structuredType — тип структурированных данных.

B.1.3.2 Атрибут объекта communicationEntity

Таблица B.2 описывает атрибуты объекта communicationEntity, определенные в 6.2.1.3.4.

Таблица В.2 — Атрибут объекта communicationEntity

Атрибут	Описание	Тип данных	Значение
Протокол	Коммуникационный протокол	xsd:string	'CANopen'
Коммуникатор	Устанавливает, играет ли этот объект активную роль в сетевых коммуникациях или нет	xsd:string	'YES' — принимает участие в сетевых коммуникациях. 'NO' — не принимает участия в сетевых коммуникациях
communicationEntityType	Тип коммуникационного объекта	xsd:string	'SLAVE' 'MASTER' 'CLIENT' 'SERVER' 'PEER' (объект, действующий как клиент и сервер) 'MASTER_SERVER' (объект, действующий как основной и подчиненный) 'DEVICEMODULE' (объект, нуждающийся в родителе для коммуникации через сеть) 'PASSIVE' (объект, не принимающий участия в сетевых коммуникациях)
communicationProfile	Идентификатор коммуникационных профилей	xsd:string	См. 6.2.2.2

B.1.3.3 Атрибуты объекта channel1

Таблица В.3 описывает атрибуты объекта channel1, определенного в 6.2.1.3.3.2.

Таблица В.3 — Атрибуты объекта channel1

Атрибут	Описание	Тип данных	Значение
channelType	Тип канала	xsd:string	
direction	Направление потока данных через этот канал	xsd:string	«I»: Вход «Q»: Выход «X»: Все равно

B.1.3.4 Атрибуты объектов MAU

Таблица В.4 описывает атрибуты объектов MAU, определенные в 6.2.1.3.3.3.

Таблица В.4 — Атрибуты объектов MAU

Атрибут	Описание	Тип данных	Значение
interfaceType	Предоставляет дополнительную информацию по типу MAU	xsd:string	«COREMOTE» «COLOCAL»
Direction	Определяет логическое направление потока данных через объект MAU	xsd:string	«INOUT»: передает и приемник «IN»: приемник и цепь обратной связи «OUT»: передает и цепь обратной связи «IN_UNI»: только принимает «OUT_UNI»: только передает
directlyConnected	Определяет, подключены ли устройства непосредственно к объекту MAU	xsd:string	«YES» «NO»

Окончание таблицы В.4

Атрибут	Описание	Тип данных	Значение
MAUType	Идентификатор типа объекта MAU со стороны продавца	xsd:string	Значение со стороны продавца
newLevel	Определяет, открывается ли объектом MAU новый структурный уровень (пример: локальное шинное соединение шинного терминала)	xsd:string	«YES» «NO»
Protocol	Определяет прогон протокола объектом MAU	xsd:string	«CANopen»
sequenceNumber	Идентификация объекта MAU, начиная с 1, отдельно нумеруемая для каждого направления	xsd: nonNegativeInteger	default = 1

В.1.3.5 Атрибуты объектов slot (слот)

Таблица В.5 описывает атрибуты объектов slot, определенные в 6.2.1.3.3.4.

Таблица В.5 — Атрибуты объектов slot

Атрибут	Описание	Тип данных	Значения
Номер	Указывает положение, в котором дочернее устройство может быть добавлено в последовательность дочерних устройств	xsd:string	Атрибут номер может содержать один номер ('3'), список номеров ('3,5,7'), диапазон номеров ('1—3') или комбинацию ('1—2,4,6,9–10')[1].
[1] Если дочерние устройства могут быть подключены к любому слоту, используют номер, определяющий максимальное число подключаемых дочерних устройств. Например, если 64 дочерних устройства могут быть подключены к родительскому устройству, используют slot number='1—64'. Для каждого положения слота требуется элемент MAUUsage, указывающий на подключенный объект MAU.			

В.1.3.6 Атрибуты объектов LED и LEDState

Таблица В.6 описывает атрибуты объекта LED, определенные в 6.2.1.3.3.5. Дочерний объект LEDState описывает разные состояния. Атрибуты LEDState описаны в таблице В.7.

Таблица В.6 — Атрибуты объекта LED

Атрибут	Описание	Тип данных	Значение
LEDType	Тип LED	xsd:string	«IOStatus» «IODiagnostic» «DeviceStatus» «DeviceDiagnostic» «CommStatus» «CommDiagnostic»

Таблица В.7 — Атрибуты объекта LEDState

Атрибут	Описание	Тип данных	Значение
LEDCondition	Положение LED в состоянии	xsd:string	«ON» «OFF»
LEDColor	Цвет LED в состоянии	xsd:string	«GREEN» «YELLOW» «RED» «ORANGE» «BLUE» «WHITE»
LEDFrequency	Частота миганий в состоянии в Гц	xsd:float	
LEDFlashCount	Число вспышек в состоянии	xsd:nonNegativeInteger	
Ref	XPath к объекту или состояние объекта, вызывающее состояние LED	xsd:string	

В.1.4 Описания дополнительных элементов

В.1.4.1 Объект accessPath

Объект accessPath должен описывать маршрут относительно объекта communicationEntity для получения доступа к объекту localDataDescription. Формат объекта:

- для объекта localDataDescription: не определен.

Примечание — Формат для объекта localDataDescription в настоящем стандарте не рассмотрен.

На рисунке В.1 показаны порядок и нумерация смещений в байтах и битах.

канал обрабатываемых данных																					
октет 0								октет 1								октет 2								...	октет n							
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	...	7	6	5	4	3	2	1	0

byteoffset

bitoffset

Рисунок В.1 — Смещения в байтах и битах в канале обрабатываемых данных

В.1.4.2 Объект datatype

Объект datatype должен определять тип данных родительского элемента.

В.1.4.3 Объект file

Объект file содержит ссылку XLink [7] на внешний файл.

В.1.4.4 Элементы gain, offset, maxVal, minVal, default, stepVal, on

Для уточнения элемента range должны быть использованы следующие элементы:

- gain — масштабный коэффициент;
- offset — масштабный коэффициент;
- maxVal — максимальное значение диапазона;
- minVal — минимальное значение диапазона;
- default — значение диапазона по умолчанию;
- stepVal — пошаговое значение;
- on — значение триггера для связи.

Все значения относятся к x_{actual}.

Масштабные коэффициенты, если они представлены, должны быть использованы, как установлено в следующей формуле: x_{scale} = offset + x_{actual} · gain.

В.1.4.5 Объекты picture, hotspotList, hotspot

Объект picture должен содержать ссылку на графическое представление элемента. Атрибуты элемента picture установлены в таблице В.8.

Таблица В.8 — Атрибуты элемента picture

Имя	Описание	Тип данных
picName	Имя файла изображения	xsd:string
picType	Идентификатор типа, расширение файла	xsd:string

Окончание таблицы В.8

Имя	Описание	Тип данных
xSize	Ширина в пикселях	xsd: positiveInteger
ySize	Высота в пикселях	xsd: positiveInteger
picClassification	Классификация пользователя содержимого изображения	xsd:string

Объект hotspotList является набором элементов hotspot. Элементы hotspot определяют положение объектов channel, LED или MAU в графическом представлении. Атрибуты объекта hotspot установлены в таблице В.9. Значения атрибутов xPos и yPos определяют относительное положение (см. рисунок В.2). Элемент hotElement является маршрутом Xpath к объектам channel, LED или MAU.

Таблица В.9 — Атрибуты элемента hotspot

Атрибуты	Описание	Тип данных	Значение
xPos	Положение горячей точки (hotspot) в процентах	xsd:nonNegativeInteger	0 <= значение <= 100
yPos	Положение горячей точки (hotspot) в процентах	xsd:nonNegativeInteger	0 <= значение <= 100
hotElement	Маршрут XPath к точке соединения	xsd:string	Допустимый XPath

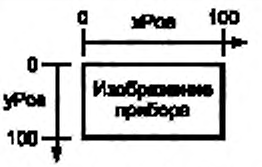


Рисунок В.2 — Положения hotspot точек соединения

В.1.4.6 Объект specificProperty

Элемент specificProperty резервируется для расширений, выполняемых разработчиком AIP.

Примечание — Использование элемента specificProperty в настоящем стандарте не рассмотрено.

В.1.4.7 Объект tool

Объект tool определяет специальную сервисную программу родительского элемента. Таблица В.10 описывает атрибуты элемента tool.

Таблица В.10 — Атрибуты элемента tool

Атрибут	Описание	Тип данных	Значение
toolClassification	Классифицирует задачу сервисной программы	xsd:string	
toolID	Однозначная идентификация операционной системы сервисной программы	xsd:string	
Примечание — PROGID или GUID являются примерами toolID.			

В.1.4.8 Объект uses

Элемент uses определяет ассоциацию «uses», как определено на рисунках 7 и 9.

В.1.5 XML схемы шаблонов профилей устройств

В.1.5.1 COFDCML.xsd

```
<?xml version = "1.0" encoding = "UTF-8"?>
<xsd:schema xmlns = "http://www.can-cia.org/xml/canopen"
targetNamespace = "http://
```

igost.e-ecolog.ru

```
xmlns:xlink = "gost.e-ecolog.ru                    "
xmlns:xsd = "gost.e-ecolog.ru
version = "1.0"
elementFormDefault = "qualified"
attributeFormDefault = "unqualified">
<xsd:import namespace = "gost.e-ecolog.ru                    " schemaLocation = "xlinkdef.xsd"/>
<xsd:include schemaLocation = "FDCMLdt.xsd"/>
<xsd:group name = "g_labels">
<xsd:choice>
<xsd:element ref = "label" maxOccurs = "unbounded"/>
<xsd:element ref = "labelRef"/>
</xsd:choice>
</xsd:group>
<xsd:group name = "g_help">
<xsd:choice>
<xsd:element ref = "help" maxOccurs = "unbounded"/>
<xsd:element ref = "helpRef"/>
</xsd:choice>
</xsd:group>
<xsd:group name = "g_naming">
<xsd:sequence>
<xsd:choice>
<xsd:sequence>
<xsd:element ref = "label" maxOccurs = "unbounded"/>
<xsd:element ref = "help" minOccurs = "0" maxOccurs = "unbounded"/>
</xsd:sequence>
<xsd:sequence>
<xsd:element ref = "labelRef"/>
<xsd:element ref = "helpRef" minOccurs = "0"/>
</xsd:sequence>
</xsd:choice>
<xsd:element ref = "helpFileRef" minOccurs = "0"/>
</xsd:sequence>
</xsd:group>
<xsd:group name = "g_stringValues">
<xsd:choice>
<xsd:element ref = "const" maxOccurs = "unbounded"/>
<xsd:element ref = "edit" maxOccurs = "unbounded"/>
<xsd:element ref = "labelRef"/>
<xsd:element ref = "instanceValue"/>
</xsd:choice>
</xsd:group>
<xsd:group name = "g_values">
<xsd:choice>
<xsd:element ref = "const" maxOccurs = "unbounded"/>
<xsd:element ref = "edit" maxOccurs = "unbounded"/>
<xsd:element ref = "enumeration" maxOccurs = "unbounded"/>
<xsd:element ref = "range" maxOccurs = "unbounded"/>
<xsd:sequence>
<xsd:element ref = "yes" minOccurs = "0"/>
<xsd:element ref = "no" minOccurs = "0"/>
</xsd:sequence>
<xsd:element ref = "reference" maxOccurs = "unbounded"/>
<xsd:element ref = "instanceValue"/>
</xsd:choice>
</xsd:group>
<xsd:attributeGroup name = "ag_FDCML">
<xsd:attribute name = "formatName" fixed = "FDCML" form = "unqualified" type = "xsd:string"/>
<xsd:attribute name = "formatVersion" fixed = "2.0" form = "unqualified" type = "xsd:string"/>
<xsd:attribute name = "fileName" use = "required" form = "unqualified" type = "xsd:string"/>
<xsd:attribute name = "fileCreator" use = "required" form = "unqualified" type = "xsd:string"/>
```

```
<xsd:attribute name = "fileCreationDate" use = "required" form = "unqualified" type = "xsd:date"/>
<xsd:attribute name = "fileModificationDate" use = "required" form = "unqualified" type =
«xsd:date»/>
<xsd:attribute name = «fileVersion» use = "required" form = "unqualified" type = "xsd:string"/>
</xsd:attributeGroup>
<xsd:attributeGroup name = "accessCategory">
<xsd:attribute name = "accessCategory" use = "required">
<xsd:simpleType>
<xsd:restriction base = "xsd:NMTOKEN">
<xsd:enumeration value = "read"/>
<xsd:enumeration value = "write"/>
<xsd:enumeration value = "readWrite"/>
<xsd:enumeration value = "noAccess"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:attributeGroup>
<xsd:attributeGroup name = "dataType">
<xsd:attribute name = "dataType" use = "required">
<xsd:simpleType>
<xsd:restriction base = "xsd:NMTOKEN">
<xsd:enumeration value = "bool"/>
<xsd:enumeration value = "byte"/>
<xsd:enumeration value = "char"/>
<xsd:enumeration value = "word"/>
<xsd:enumeration value = "dWord"/>
<xsd:enumeration value = "lWord"/>
<xsd:enumeration value = "sInt"/>
<xsd:enumeration value = "usInt"/>
<xsd:enumeration value = "int"/>
<xsd:enumeration value = "uInt"/>
<xsd:enumeration value = "dInt"/>
<xsd:enumeration value = "udInt"/>
<xsd:enumeration value = "lInt"/>
<xsd:enumeration value = "ulInt"/>
<xsd:enumeration value = "real"/>
<xsd:enumeration value = "lReal"/>
<xsd:enumeration value = "string"/>
<xsd:enumeration value = "unicode"/>
<xsd:enumeration value = "struct"/>
<xsd:enumeration value = "physical"/>
<xsd:enumeration value = "array"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:attributeGroup>
<xsd:attributeGroup name = "persistent">
<xsd:attribute name = "persistent" use = "required">
<xsd:simpleType>
<xsd:restriction base = "xsd:NMTOKEN">
<xsd:enumeration value = "false"/>
<xsd:enumeration value = "true"/>
<xsd:enumeration value = "n.a"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:attributeGroup>
<xsd:attributeGroup name = "type">
<xsd:attribute name = "type" use = "required">
<xsd:simpleType>
<xsd:restriction base = "xsd:NMTOKEN">
<xsd:enumeration value = "fixedValue"/>
```

```
<xsd:enumeration value = "vendorSpecific"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:attributeGroup>
<xsd:attributeGroup name = "use">
<xsd:attribute name = "use" use = "required">
<xsd:simpleType>
<xsd:restriction base = "xsd:NMTOKEN">
<xsd:enumeration value = "required"/>
<xsd:enumeration value = "optional"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:attributeGroup>
<!- -xmlns:xml="gost.e-ecolog.ru                                    ->
<xsd:element name = "ISO15745Profile">
<xsd:annotation>
<xsd:documentation>Document Element</xsd:documentation>
</xsd:annotation>
<xsd:complexType>
<xsd:sequence>
<xsd:element ref = "ProfileHeader"/>
<xsd:choice>
<xsd:element ref = "ProfileBody"/>
<xsd:element ref = "ProfilesBody"/>
</xsd:choice>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name = "accessPath" type = "xsd:string"/>
<xsd:element name = "additionalItem">
<xsd:complexType>
<xsd:sequence>
<xsd:group ref = "g_naming"/>
<xsd:element ref = "pictureList" minOccurs = "0"/>
<xsd:choice minOccurs = "0">
<xsd:group ref = "g_values"/>
</xsd:choice>
<xsd:element ref = "specificProperty" minOccurs = "0" maxOccurs = "unbounded"/>
<xsd:element ref = "additionalItem" minOccurs = "0" maxOccurs = "unbounded"/>
<xsd:element ref = "instances" minOccurs = "0"/>
</xsd:sequence>
<xsd:attribute name = "uniqueID" use = "required" type = "xsd:ID"/>
<xsd:attribute name = "additionalItemType" use = "required" type = "xsd:string"/>
<xsd:attribute name = "enabled" default = "YES">
<xsd:simpleType>
<xsd:restriction base = "xsd:string">
<xsd:enumeration value = "YES"/>
<xsd:enumeration value = "NO"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
<xsd:element name = "additionalItemCategory">
<xsd:complexType>
<xsd:sequence>
<xsd:group ref = "g_naming"/>
<xsd:element ref = "additionalItem" maxOccurs = "unbounded"/>
</xsd:sequence>
```

```
<xsd:attribute name = "uniqueID" type = "xsd:ID"/>
<xsd:attribute name = "enabled" default = "YES">
<xsd:simpleType>
<xsd:restriction base = "xsd:string">
<xsd:enumeration value = "YES"/>
<xsd:enumeration value = "NO"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
<xsd:element name = "additionalItemList">
<xsd:complexType>
<xsd:sequence>
<xsd:group ref = "g_naming"/>
<xsd:choice maxOccurs = "unbounded">
<xsd:element ref = "additionalItemCategory"/>
<xsd:element ref = "additionalItem"/>
</xsd:choice>
</xsd:sequence>
<xsd:attribute name = "uniqueID" type = "xsd:ID"/>
<xsd:attribute name = "additionalItemsType" use = "required" type = "xsd:string"/>
<xsd:attribute name = "enabled" default = "YES">
<xsd:simpleType>
<xsd:restriction base = "xsd:string">
<xsd:enumeration value = "YES"/>
<xsd:enumeration value = "NO"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
<xsd:element name = "alignment">
<xsd:complexType>
<xsd:attribute name = "type" use = "required">
<xsd:simpleType>
<xsd:restriction base = "xsd:NMTOKEN">
<xsd:enumeration value = "byte"/>
<xsd:enumeration value = "word"/>
<xsd:enumeration value = "dword"/>
<xsd:enumeration value = "lword"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
<xsd:element name = "ApplicationProcess">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref = "externalSchema"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name = "arrayType">
<xsd:complexType>
<xsd:sequence>
<xsd:choice minOccurs = "0">
<xsd:group ref = "g_help"/>
</xsd:choice>
<xsd:group ref = "g_datatypes"/>
<xsd:element ref = "subrange" maxOccurs = "unbounded"/>
</xsd:sequence>
```

```
<xsd:attribute name = "initialValues" type = "xsd:string"/>
</xsd:complexType>
</xsd:element>
<xsd:element name = "buildDate">
<xsd:complexType>
<xsd:choice>
<xsd:group ref = "g_labels"/>
</xsd:choice>
<xsd:attribute name = "readOnly" default = "YES">
<xsd:simpleType>
<xsd:restriction base = "xsd:string">
<xsd:enumeration value = "YES"/>
<xsd:enumeration value = "NO"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
<xsd:element name = "capabilities">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref = "externalSchema"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name = "cfgItemList">
<xsd:complexType>
<xsd:sequence>
<xsd:sequence minOccurs = "0">
<xsd:group ref = "g_naming"/>
</xsd:sequence>
<xsd:element ref = "CANopenDedicatedCfgCategory"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name = "change">
<xsd:complexType>
<xsd:choice>
<xsd:group ref = "g_datatypes"/>
<xsd:element ref = "datatypeInstance"/>
</xsd:choice>
<xsd:attribute name = "ref" use = "required" type = "xsd:string"/>
</xsd:complexType>
</xsd:element>
<xsd:element name = "channel">
<xsd:complexType>
<xsd:sequence>
<xsd:group ref = "g_naming"/>
<xsd:element ref = "pictureList" minOccurs = "0"/>
<xsd:element ref = "accessPath" minOccurs = "0"/>
<xsd:element ref = "specificProperty" minOccurs = "0" maxOccurs = "unbounded"/>
<xsd:element ref = "uses" minOccurs = "0"/>
<xsd:element ref = "provides" minOccurs = "0" maxOccurs = "unbounded"/>
<xsd:element ref = "channel" minOccurs = "0" maxOccurs = "unbounded"/>
<xsd:element ref = "instances" minOccurs = "0" maxOccurs = "unbounded"/>
</xsd:sequence>
<xsd:attribute name = "uniqueID" use = "required" type = "xsd:ID"/>
<xsd:attribute name = "channelType" type = "xsd:string"/>
<xsd:attribute name = "direction" default = "X">
<xsd:simpleType>
<xsd:restriction base = "xsd:string">
```

```
<xsd:enumeration value = "I"/>
<xsd:enumeration value = "Q"/>
<xsd:enumeration value = "X"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
<xsd:attribute name = "enabled" default = "YES">
<xsd:simpleType>
<xsd:restriction base = "xsd:string">
<xsd:enumeration value = "YES"/>
<xsd:enumeration value = "NO"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
<xsd:element name = "channelList">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref = "channel" maxOccurs = "unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name = "communicationEntity">
<xsd:complexType>
<xsd:sequence>
<xsd:sequence minOccurs = "0">
<xsd:group ref = "g_naming"/>
</xsd:sequence>
<xsd:element ref = "cfgItemList" minOccurs = "0"/>
<xsd:element ref = "CANopenIdentity"/>
<xsd:element name = "CANopenCommunicationFunctionList" minOccurs = "0">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref = "function" maxOccurs = "unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name = "CANopenCommunicationParameterList">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref = "parameter" maxOccurs = "unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element ref = "CANopenObjectAccessList"/>
</xsd:sequence>
<xsd:attribute name = "uniqueID" type = "xsd:ID"/>
<xsd:attribute name = "protocol" use = "required" type = "xsd:string"/>
<xsd:attribute name = "communicator" default = "YES">
<xsd:simpleType>
<xsd:restriction base = "xsd:string">
<xsd:enumeration value = "YES"/>
<xsd:enumeration value = "NO"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
<xsd:attribute name = "communicationEntityType" default = "SLAVE">
<xsd:simpleType>
<xsd:restriction base = "xsd:string">
<xsd:enumeration value = "SLAVE"/>
<xsd:enumeration value = "MASTER"/>
```

```
<xsd:enumeration value = "CLIENT"/>
<xsd:enumeration value = "SERVER"/>
<xsd:enumeration value = "INTERCONNECTION"/>
<xsd:enumeration value = "PEER"/>
<xsd:enumeration value = "MASTER_SLAVE"/>
<xsd:enumeration value = "DEVICEMODULE"/>
<xsd:enumeration value = "PASSIVE"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
<xsd:attribute name = "communicationProfile" type = "xsd:string"/>
<xsd:attribute name = "enabled" default = "YES">
<xsd:simpleType>
<xsd:restriction base = "xsd:string">
<xsd:enumeration value = "YES"/>
<xsd:enumeration value = "NO"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
<xsd:element name = "connection">
<xsd:complexType>
<xsd:sequence minOccurs = "0">
<xsd:element ref = "specificProperty" maxOccurs = "unbounded"/>
</xsd:sequence>
<xsd:attribute name = "destination" use = "required" type = "xsd:string"/>
<xsd:attribute name = "source" use = "required" type = "xsd:string"/>
</xsd:complexType>
</xsd:element>
<xsd:element name = "connectionList">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref = "connection" minOccurs = "0" maxOccurs = "unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name = "const">
<xsd:complexType>
<xsd:choice>
<xsd:group ref = "g_datatypes"/>
<xsd:element ref = "datatypeInstance"/>
</xsd:choice>
<xsd:attribute ref = "xml:lang"/>
<xsd:attribute ref = "xlink:type"/>
<xsd:attribute ref = "xlink:href"/>
<xsd:attribute name = "format" type = "xsd:string"/>
<xsd:attribute name = "unit" type = "xsd:string"/>
</xsd:complexType>
</xsd:element>
<xsd:element name = "datatype">
<xsd:complexType>
<xsd:choice>
<xsd:group ref = "g_datatypes"/>
<xsd:element ref = "datatypeInstance"/>
</xsd:choice>
</xsd:complexType>
</xsd:element>
<xsd:element name = "datatypeInstance" nillable = "true">
<xsd:complexType>
<xsd:attribute name = "ref" use = "required" type = "xsd:string"/>
</xsd:complexType>
```

```
</xsd:element>
<xsd:element name = "datatypeTemplate">
<xsd:complexType>
<xsd:sequence>
<xsd:group ref = "g_naming"/>
<xsd:choice>
<xsd:element ref = "directlyDerivedType"/>
<xsd:element ref = "enumeratedType"/>
<xsd:element ref = "subrangeType"/>
<xsd:element ref = "arrayType"/>
<xsd:element ref = "structuredType"/>
</xsd:choice>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name = "datatypeTemplateList">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref = "alignment" minOccurs = "0"/>
<xsd:element ref = "endianess" minOccurs = "0"/>
<xsd:element ref = "datatypeTemplate" maxOccurs = "unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name = "default">
<xsd:complexType>
<xsd:choice>
<xsd:group ref = "g_numericDatatypes"/>
<xsd:group ref = "g_booleanDatatypes"/>
<xsd:group ref = "g_userDatatypes"/>
<xsd:group ref = "g_timeDatatypes"/>
</xsd:choice>
</xsd:complexType>
</xsd:element>
<xsd:element name = "deviceFamily">
<xsd:complexType>
<xsd:choice>
<xsd:group ref = "g_labels"/>
</xsd:choice>
<xsd:attribute name = "readOnly" default = "YES">
<xsd:simpleType>
<xsd:restriction base = "xsd:string">
<xsd:enumeration value = "YES"/>
<xsd:enumeration value = "NO"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
<xsd:element name = "defaultfile">
<xsd:complexType>
<xsd:attribute ref = "xlink:type"/>
<xsd:attribute ref = "xlink:href" use = "required"/>
</xsd:complexType>
</xsd:element>
<xsd:element name = "deleteEntity">
<xsd:complexType>
<xsd:attribute name = "node" use = "required" type = "xsd:string"/>
</xsd:complexType>
</xsd:element>
<xsd:element name = "DeviceFunction">
<xsd:complexType>
```

```
<xsd:choice maxOccurs = "unbounded">
<xsd:element ref = "externalSchema" minOccurs = "0" maxOccurs = "unbounded"/>
</xsd:choice>
</xsd:complexType>
</xsd:element>
<xsd:element name = "DeviceIdentity">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref = "vendorName"/>
<xsd:element ref = "vendorID" minOccurs = "0"/>
<xsd:element ref = "vendorText" minOccurs = "0"/>
<xsd:element ref = "deviceFamily"/>
<xsd:element ref = "capabilities" minOccurs = "0"/>
<xsd:element ref = "productFamily" minOccurs = "0"/>
<xsd:element ref = "productName"/>
<xsd:element ref = "productID" minOccurs = "0"/>
<xsd:element ref = "productText" minOccurs = "0"/>
<xsd:element ref = "orderNumber" minOccurs = "0"/>
<xsd:element ref = "version" minOccurs = "0" maxOccurs = "unbounded"/>
<xsd:element ref = "buildDate" minOccurs = "0"/>
<xsd:element ref = "specificationRevision" minOccurs = "0"/>
<xsd:element ref = "instanceName" minOccurs = "0"/>
<xsd:element ref = "serialNumber" minOccurs = "0"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name = "DeviceManager">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref = "importList" minOccurs = "0"/>
<xsd:element ref = "datatypeTemplateList" minOccurs = "0" maxOccurs = "unbounded"/>
<xsd:element ref = "dictionaryList" minOccurs = "0"/>
<xsd:element ref = "helpFileList" minOccurs = "0"/>
<xsd:element ref = "toolList" minOccurs = "0"/>
<xsd:element ref = "pictureList" minOccurs = "0"/>
<xsd:element ref = "deviceStructure" minOccurs = "0"/>
<xsd:element ref = "localDataDescriptionList" minOccurs = "0"/>
<xsd:element ref = "additionalItemList" minOccurs = "0" maxOccurs = "unbounded"/>
<xsd:element ref = "communicationEntity" maxOccurs = "unbounded"/>
<xsd:element ref = "processingEntity" maxOccurs = "unbounded"/>
<xsd:element ref = "externalSchema" minOccurs = "0" maxOccurs = "unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name = "deviceStructure">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref = "channelList" minOccurs = "0"/>
<xsd:element ref = "MAUList" minOccurs = "0"/>
<xsd:element ref = "slotList" minOccurs = "0"/>
<xsd:element ref = "indicatorList" minOccurs = "0"/>
<xsd:element ref = "externalSchema" minOccurs = "0" maxOccurs = "unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name = "dictionary">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref = "file"/>
</xsd:sequence>
<xsd:attribute ref = "xml:lang" use = "required"/>
```

```
<xsd:attribute name = "dictID" type = "xsd:string"/>
</xsd:complexType>
</xsd:element>
<xsd:element name = "dictionaryList">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref = "dictionary" maxOccurs = "unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name = "directlyDerivedType">
<xsd:complexType>
<xsd:sequence>
<xsd:choice minOccurs = "0">
<xsd:group ref = "g_help"/>
</xsd:choice>
<xsd:group ref = "g_datatypes"/>
</xsd:sequence>
<xsd:attribute name = "initialValue" type = "xsd:string"/>
</xsd:complexType>
</xsd:element>
<xsd:element name = "disable">
<xsd:complexType>
<xsd:attribute name = "ref" use = "required" type = "xsd:string"/>
</xsd:complexType>
</xsd:element>
<xsd:element name = "edit">
<xsd:complexType>
<xsd:choice>
<xsd:group ref = "g_stringDatatypes"/>
<xsd:element ref = "datatypeInstance"/>
</xsd:choice>
<xsd:attribute ref = "xml:lang"/>
<xsd:attribute ref = "xlink:type"/>
<xsd:attribute ref = "xlink:href"/>
<xsd:attribute name = "format" type = "xsd:string"/>
<xsd:attribute name = "unit" type = "xsd:string"/>
</xsd:complexType>
</xsd:element>
<xsd:element name = "enable">
<xsd:complexType>
<xsd:attribute name = "ref" use = "required" type = "xsd:string"/>
</xsd:complexType>
</xsd:element>
<xsd:element name = "endianess">
<xsd:complexType>
<xsd:attribute name = "type" use = "required">
<xsd:simpleType>
<xsd:restriction base = "xsd:NMTOKEN">
<xsd:enumeration value = "little"/>
<xsd:enumeration value = "big"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
<xsd:element name = "enumeratedType">
<xsd:complexType>
<xsd:sequence>
<xsd:choice minOccurs = "0">
<xsd:group ref = "g_help"/>
```

```
</xsd:choice>
<xsd:element ref = "enumeratedValue" maxOccurs = "unbounded"/>
</xsd:sequence>
<xsd:attribute name = "initialValue" type = "xsd:string"/>
</xsd:complexType>
</xsd:element>
<xsd:element name = "enumeratedValue">
<xsd:complexType>
<xsd:sequence>
<xsd:group ref = "g_naming"/>
<xsd:choice minOccurs = "0">
<xsd:group ref = "g_datatypes"/>
</xsd:choice>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name = "enumeration">
<xsd:complexType>
<xsd:sequence>
<xsd:group ref = "g_labels"/>
<xsd:choice>
<xsd:group ref = "g_numericDatatypes"/>
<xsd:group ref = "g_booleanDatatypes"/>
<xsd:group ref = "g_stringDatatypes"/>
<xsd:group ref = "g_userDatatypes"/>
</xsd:choice>
<xsd:element ref = "relations" minOccurs = "0"/>
</xsd:sequence>
<xsd:attribute name = "default" default = "NO">
<xsd:simpleType>
<xsd:restriction base = "xsd:string">
<xsd:enumeration value = "NO"/>
<xsd:enumeration value = "YES"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
<xsd:attribute name = "multipleSelection">
<xsd:simpleType>
<xsd:restriction base = "xsd:string">
<xsd:enumeration value = "YES"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
<xsd:element name = "externalSchema">
<xsd:complexType>
<xsd:sequence>
<xsd:any namespace = "##any" processContents = "strict"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name = "file">
<xsd:complexType>
<xsd:attribute ref = "xlink:type"/>
<xsd:attribute ref = "xlink:href" use = "required"/>
</xsd:complexType>
</xsd:element>
<xsd:element name = "gain">
<xsd:complexType>
<xsd:choice>
```

```
<xsd:group ref = "g_numericDatatypes"/>
<xsd:group ref = "g_userDatatypes"/>
</xsd:choice>
</xsd:complexType>
</xsd:element>
<xsd:element name = "help">
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base = "xsd:string">
<xsd:attribute ref = "xml:lang" use = "required"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:element name = "helpFile">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref = "file"/>
</xsd:sequence>
<xsd:attribute ref = "xml:lang" use = "required"/>
<xsd:attribute name = "helpFileID" type = "xsd:string"/>
</xsd:complexType>
</xsd:element>
<xsd:element name = "helpFileList">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref = "helpFile" maxOccurs = "unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name = "helpFileRef">
<xsd:complexType>
<xsd:attribute name = "helpFileID" type = "xsd:string"/>
<xsd:attribute name = "helpID" type = "xsd:string"/>
</xsd:complexType>
</xsd:element>
<xsd:element name = "helpRef">
<xsd:complexType>
<xsd:attribute name = "dictID" type = "xsd:string"/>
<xsd:attribute name = "textID" use = "required" type = "xsd:string"/>
</xsd:complexType>
</xsd:element>
<xsd:element name = "hotspot">
<xsd:complexType>
<xsd:attribute name = "xPos" use = "required">
<xsd:simpleType>
<xsd:restriction base = "xsd:nonNegativeInteger">
<xsd:maxInclusive value = "100"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
<xsd:attribute name = "yPos" use = "required">
<xsd:simpleType>
<xsd:restriction base = "xsd:nonNegativeInteger">
<xsd:maxInclusive value = "100"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
<xsd:attribute name = "hotElement" use = "required" type = "xsd:string"/>
</xsd:complexType>
</xsd:element>
```

```
<xsd:element name = "hotspotList">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref = "hotspot" maxOccurs = "unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:simpleType name = "IASInterface_DataType">
<xsd:union memberTypes = "">
<xsd:simpleType>
<xsd:restriction base = "xsd:string">
<xsd:enumeration value = "CSI"/>
<xsd:enumeration value = "HCI"/>
<xsd:enumeration value = "ISI"/>
<xsd:enumeration value = "API"/>
<xsd:enumeration value = "CMI"/>
<xsd:enumeration value = "ESI"/>
<xsd:enumeration value = "FSI"/>
<xsd:enumeration value = "MTI"/>
<xsd:enumeration value = "SEI"/>
<xsd:enumeration value = "USI"/>
</xsd:restriction>
</xsd:simpleType>
<xsd:simpleType>
<xsd:restriction base = "xsd:string">
<xsd:length value = "4"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:union>
</xsd:simpleType>
<xsd: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:element name = "identity">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref = "vendorName"/>
<xsd:element ref = "typeName"/>
<xsd:element ref = "version" minOccurs = "0" maxOccurs = "unbounded"/>
<xsd:element ref = "buildDate"/>
<xsd:element ref = "specificationRevision" minOccurs = "0"/>
<xsd:element ref = "instanceName" minOccurs = "0"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name = "importList">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref = "file" maxOccurs = "unbounded"/>
</xsd:sequence>
</xsd:complexType>
```

```
</xsd:element>
<xsd:element name = "indicatorList">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref = "LEDList" minOccurs = "0"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name = "instanceName">
<xsd:complexType>
<xsd:choice>
<xsd:group ref = "g_labels"/>
</xsd:choice>
<xsd:attribute name = "readOnly" default = "YES">
<xsd:simpleType>
<xsd:restriction base = "xsd:string">
<xsd:enumeration value = "YES"/>
<xsd:enumeration value = "NO"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
<xsd:element name = "instances">
<xsd:complexType>
<xsd:choice>
<xsd:element ref = "externalSchema" maxOccurs = "unbounded"/>
<xsd:choice maxOccurs = "unbounded">
<xsd:element ref = "modifyValue"/>
<xsd:element ref = "modifyEntity"/>
<xsd:element ref = "recalcValue"/>
<xsd:element ref = "recalcEntity"/>
<xsd:element ref = "deleteEntity"/>
</xsd:choice>
</xsd:choice>
<xsd:attribute name = "maxInstances" type = "xsd:nonNegativeInteger"/>
</xsd:complexType>
</xsd:element>
<xsd:element name = "instanceValue" type = "xsd:string"/>
<xsd:element name = "internalConnectionPoint">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref = "uses" minOccurs = "0"/>
<xsd:element ref = "specificProperty" minOccurs = "0" maxOccurs = "unbounded"/>
<xsd:element ref = "instances" minOccurs = "0"/>
</xsd:sequence>
<xsd:attribute name = "uniqueID" use = "required" type = "xsd:ID"/>
<xsd:attribute name = "ref" use = "required" type = "xsd:string"/>
<xsd:attribute name = "enabled" default = "YES">
<xsd:simpleType>
<xsd:restriction base = "xsd:string">
<xsd:enumeration value = "YES"/>
<xsd:enumeration value = "NO"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
<xsd:element name = "internalConnectionPointList">
<xsd:complexType>
<xsd:sequence>
```

```
<xsd:element ref = "internalConnectionPoint" maxOccurs = "unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name = "ISO15745Edition" type = "xsd:positiveInteger"/>
<xsd:element name = "ISO15745Part" type = "xsd:positiveInteger"/>
<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:element name = "label">
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base = "xsd:string">
<xsd:attribute ref = "xml:lang" use = "required"/>
<xsd:attribute ref = "xlink:type"/>
<xsd:attribute ref = "xlink:href"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:element name = "labelRef">
<xsd:complexType>
<xsd:attribute name = "dictID" type = "xsd:string"/>
<xsd:attribute name = "textID" use = "required" type = "xsd:string"/>
<xsd:attribute ref = "xlink:type"/>
<xsd:attribute ref = "xlink:href"/>
</xsd:complexType>
</xsd:element>
<xsd:element name = "LED">
<xsd:complexType>
<xsd:sequence>
<xsd:group ref = "g_naming"/>
<xsd:element ref = "LEDState" maxOccurs = "unbounded"/>
<xsd:element ref = "specificProperty" minOccurs = "0" maxOccurs = "unbounded"/>
</xsd:sequence>
<xsd:attribute name = "uniqueID" use = "required" type = "xsd:ID"/>
<xsd:attribute name = "LEDType" use = "required">
<xsd:simpleType>
<xsd:restriction base = "xsd:string">
<xsd:enumeration value = "IOStatus"/>
<xsd:enumeration value = "IODiagnostic"/>
<xsd:enumeration value = "DeviceStatus"/>
<xsd:enumeration value = "DeviceDiagnostic"/>
<xsd:enumeration value = "CommStatus"/>
<xsd:enumeration value = "CommDiagnostic"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
<xsd:attribute name = "enabled" default = "YES">
<xsd:simpleType>
<xsd:restriction base = "xsd:string">
<xsd:enumeration value = "YES"/>
<xsd:enumeration value = "NO"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
```

```
<xsd:element name = "LEDList">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref = "LED" maxOccurs = "unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name = "LEDState">
<xsd:complexType>
<xsd:sequence>
<xsd:sequence minOccurs = "0">
<xsd:group ref = "g_naming"/>
</xsd:sequence>
<xsd:element ref = "specificProperty" minOccurs = "0" maxOccurs = "unbounded"/>
</xsd:sequence>
<xsd:attribute name = "LEDCondition" use = "required">
<xsd:simpleType>
<xsd:restriction base = "xsd:string">
<xsd:enumeration value = "ON"/>
<xsd:enumeration value = "OFF"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
<xsd:attribute name = "LEDColor">
<xsd:simpleType>
<xsd:restriction base = "xsd:string">
<xsd:enumeration value = "GREEN"/>
<xsd:enumeration value = "YELLOW"/>
<xsd:enumeration value = "RED"/>
<xsd:enumeration value = "ORANGE"/>
<xsd:enumeration value = "BLUE"/>
<xsd:enumeration value = "WHITE"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
<xsd:attribute name = "LEDFrequency" type = "xsd:float"/>
<xsd:attribute name = "LEDFlashCount" type = "xsd:nonNegativeInteger"/>
<xsd:attribute name = "ref" type = "xsd:string"/>
</xsd:complexType>
</xsd:element>
<xsd:element name = "localDataCategory">
<xsd:complexType>
<xsd:sequence>
<xsd:group ref = "g_naming"/>
<xsd:element ref = "localDataDescription" maxOccurs = "unbounded"/>
</xsd:sequence>
<xsd:attribute name = "uniqueID" type = "xsd:ID"/>
<xsd:attribute name = "enabled" default = "YES">
<xsd:simpleType>
<xsd:restriction base = "xsd:string">
<xsd:enumeration value = "YES"/>
<xsd:enumeration value = "NO"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
<xsd:element name = "localDataDescription">
<xsd:complexType>
<xsd:sequence>
<xsd:group ref = "g_naming"/>
<xsd:element ref = "pictureList" minOccurs = "0"/>
```

```
<xsd:element ref = "accessPath"/>
<xsd:element ref = "datatype"/>
<xsd:element ref = "specificProperty" minOccurs = "0" maxOccurs = "unbounded"/>
<xsd:element ref = "uses" minOccurs = "0"/>
<xsd:element ref = "provides" minOccurs = "0" maxOccurs = "unbounded"/>
<xsd:element ref = "localDataDescription" minOccurs = "0" maxOccurs = "unbounded"/>
<xsd:element ref = "instances" minOccurs = "0"/>
</xsd:sequence>
<xsd:attribute name = "uniqueID" use = "required" type = "xsd:ID"/>
<xsd:attribute name = "direction" use = "required">
<xsd:simpleType>
<xsd:restriction base = "xsd:string">
<xsd:enumeration value = "I"/>
<xsd:enumeration value = "Q"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
<xsd:attribute name = "localDataDescriptionType" use = "required" type = "xsd:string"/>
<xsd:attribute name = "enabled" default = "YES">
<xsd:simpleType>
<xsd:restriction base = "xsd:string">
<xsd:enumeration value = "YES"/>
<xsd:enumeration value = "NO"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
<xsd:element name = "localDataDescriptionList">
<xsd:complexType>
<xsd:sequence>
<xsd:sequence minOccurs = "0">
<xsd:group ref = "g_naming"/>
</xsd:sequence>
<xsd:choice maxOccurs = "unbounded">
<xsd:element ref = "localDataCategory"/>
<xsd:element ref = "localDataDescription"/>
</xsd:choice>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name = "logicalConnectionPoint">
<xsd:complexType>
<xsd:sequence>
<xsd:group ref = "g_naming"/>
<xsd:element ref = "pictureList" minOccurs = "0"/>
<xsd:element ref = "specificProperty" minOccurs = "0" maxOccurs = "unbounded"/>
<xsd:element ref = "uses" minOccurs = "0"/>
<xsd:element ref = "provides" maxOccurs = "unbounded"/>
<xsd:element ref = "instances" minOccurs = "0"/>
</xsd:sequence>
<xsd:attribute name = "uniqueID" use = "required" type = "xsd:ID"/>
<xsd:attribute name = "logicalConnectionPointType" use = "required" type = "xsd:string"/>
<xsd:attribute name = "role" use = "required">
<xsd:simpleType>
<xsd:restriction base = "xsd:string">
<xsd:enumeration value = "CLIENT"/>
<xsd:enumeration value = "SERVER"/>
<xsd:enumeration value = "PEER"/>
<xsd:enumeration value = "PUBLISHER"/>
<xsd:enumeration value = "SUBSCRIBER"/>
<xsd:enumeration value = "PUBLISHERSUBSCRIBER"/>
```

```
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
<xsd:attribute name = "maxRelationships" default = "1" type = "xsd:nonNegativeInteger"/>
<xsd:attribute name = "newLevel" default = "NO">
<xsd:simpleType>
<xsd:restriction base = "xsd:string">
<xsd:enumeration value = "YES"/>
<xsd:enumeration value = "NO"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
<xsd:attribute name = "enabled" default = "YES">
<xsd:simpleType>
<xsd:restriction base = "xsd:string">
<xsd:enumeration value = "YES"/>
<xsd:enumeration value = "NO"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
<xsd:element name = "logicalConnectionPointAssembly">
<xsd:complexType>
<xsd:sequence>
<xsd:group ref = "g_naming"/>
<xsd:element ref = "specificProperty" minOccurs = "0" maxOccurs = "unbounded"/>
<xsd:element ref = "uses" minOccurs = "0"/>
<xsd:element ref = "provides" maxOccurs = "unbounded"/>
<xsd:element ref = "instances" minOccurs = "0"/>
</xsd:sequence>
<xsd:attribute name = "uniqueID" use = "required" type = "xsd:ID"/>
<xsd:attribute name = "logicalConnectionPointAssemblyType" use = "required" type =
"xsd:string"/>
<xsd:attribute name = "enabled" default = "YES">
<xsd:simpleType>
<xsd:restriction base = "xsd:string">
<xsd:enumeration value = "YES"/>
<xsd:enumeration value = "NO"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
<xsd:element name = "logicalConnectionPointAssemblyList">
<xsd:complexType>
<xsd:sequence>
<xsd:sequence minOccurs = "0">
<xsd:group ref = "g_naming"/>
</xsd:sequence>
<xsd:element ref = "logicalConnectionPointAssembly" maxOccurs = "unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name = "logicalConnectionPointList">
<xsd:complexType>
<xsd:sequence>
<xsd:sequence minOccurs = "0">
<xsd:group ref = "g_naming"/>
</xsd:sequence>
<xsd:element ref = "logicalConnectionPoint" maxOccurs = "unbounded"/>
```

```
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name = "MAU">
<xsd:complexType>
<xsd:sequence>
<xsd:group ref = "g_naming"/>
<xsd:element ref = "pictureList" minOccurs = "0"/>
<xsd:element ref = "specificProperty" minOccurs = "0" maxOccurs = "unbounded"/>
<xsd:element ref = "instances" minOccurs = "0"/>
</xsd:sequence>
<xsd:attribute name = "uniqueID" use = "required" type = "xsd:ID"/>
<xsd:attribute name = "protocol" use = "required" type = "xsd:string"/>
<xsd:attribute name = "interfaceType" use = "required" type = "xsd:string"/>
<xsd:attribute name = "MAUType" type = "xsd:string"/>
<xsd:attribute name = "newLevel" default = "NO">
<xsd:simpleType>
<xsd:restriction base = "xsd:string">
<xsd:enumeration value = "NO"/>
<xsd:enumeration value = "YES"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
<xsd:attribute name = "directlyConnected" use = "required">
<xsd:simpleType>
<xsd:restriction base = "xsd:string">
<xsd:enumeration value = "NO"/>
<xsd:enumeration value = "YES"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
<xsd:attribute name = "direction" use = "required">
<xsd:simpleType>
<xsd:restriction base = "xsd:string">
<xsd:enumeration value = "IN"/>
<xsd:enumeration value = "IN_UNI"/>
<xsd:enumeration value = "OUT"/>
<xsd:enumeration value = "OUT_UNI"/>
<xsd:enumeration value = "INOUT"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
<xsd:attribute name = "sequenceNumber" default = "1" type = "xsd:nonNegativeInteger"/>
<xsd:attribute name = "enabled" default = "YES">
<xsd:simpleType>
<xsd:restriction base = "xsd:string">
<xsd:enumeration value = "YES"/>
<xsd:enumeration value = "NO"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
<xsd:element name = "MAUList">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref = "MAU" maxOccurs = "unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name = "MAUUsage">
<xsd:complexType>
```

```
<xsd:attribute name = "ref" use = "required" type = "xsd:string"/>
</xsd:complexType>
</xsd:element>
<xsd:element name = "MAUUsageList">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref = "MAUUsage" maxOccurs = "unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name = "maxVal">
<xsd:complexType>
<xsd:choice>
<xsd:group ref = "g_numericDatatypes"/>
<xsd:group ref = "g_booleanDatatypes"/>
<xsd:group ref = "g_userDatatypes"/>
<xsd:group ref = "g_timeDatatypes"/>
</xsd:choice>
</xsd:complexType>
</xsd:element>
<xsd:element name = "minVal">
<xsd:complexType>
<xsd:choice>
<xsd:group ref = "g_numericDatatypes"/>
<xsd:group ref = "g_booleanDatatypes"/>
<xsd:group ref = «g_userDatatypes"/>
<xsd:group ref = "g_timeDatatypes"/>
</xsd:choice>
</xsd:complexType>
</xsd:element>
<xsd:element name = "modifyEntity">
<xsd:complexType>
<xsd:choice maxOccurs = "unbounded">
<xsd:any namespace = "gost.e-ecolog.ru                    processContents = "strict"/>
</xsd:choice>
<xsd:attribute name = "node" use = "required" type = "xsd:string"/>
</xsd:complexType>
</xsd:element>
<xsd:element name = "modifyValue">
<xsd:complexType>
<xsd:choice>
<xsd:group ref = "g_values"/>
</xsd:choice>
<xsd:attribute name = "node" use = "required" type = "xsd:string"/>
</xsd:complexType>
</xsd:element>
<xsd:element name = "no">
<xsd:complexType>
<xsd:sequence>
<xsd:group ref = "g_naming"/>
<xsd:element ref = "relations" minOccurs = "0"/>
</xsd:sequence>
<xsd:attribute name = "default" default = "NO">
<xsd:simpleType>
<xsd:restriction base = "xsd:string">
<xsd:enumeration value = "NO"/>
<xsd:enumeration value = "YES"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
```

```
<xsd:element name = "nonStandardizedExtension">
<xsd:complexType>
<xsd:sequence>
<xsd:any namespace = "##any" processContents = "strict"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name = "offset">
<xsd:complexType>
<xsd:choice>
<xsd:group ref = "g_numericDatatypes"/>
<xsd:group ref = "g_userDatatypes"/>
</xsd:choice>
</xsd:complexType>
</xsd:element>
<xsd:element name = "on">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref = "relations"/>
</xsd:sequence>
<xsd:attribute name = "value" use = "required" type = "xsd:string"/>
</xsd:complexType>
</xsd:element>
<xsd:element name = "orderNumber">
<xsd:complexType>
<xsd:choice>
<xsd:group ref = "g_labels"/>
</xsd:choice>
<xsd:attribute name = "readOnly" default = "YES">
<xsd:simpleType>
<xsd:restriction base = "xsd:string">
<xsd:enumeration value = "YES"/>
<xsd:enumeration value = "NO"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
<xsd:element name = "picture">
<xsd:complexType>
<xsd:sequence>
<xsd:sequence minOccurs = "0">
<xsd:group ref = "g_naming"/>
</xsd:sequence>
<xsd:element ref = "hotspotList" minOccurs = "0"/>
</xsd:sequence>
<xsd:attribute name = "picClassification" type = "xsd:string"/>
<xsd:attribute name = "picType" use = "required" type = "xsd:string"/>
<xsd:attribute name = "picName" use = "required" type = "xsd:string"/>
<xsd:attribute name = "xSize" type = "xsd:positiveInteger"/>
<xsd:attribute name = "ySize" type = "xsd:positiveInteger"/>
</xsd:complexType>
</xsd:element>
<xsd:element name = "pictureList">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref = "picture" maxOccurs = "unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name = "processingEntity">
<xsd:complexType>
```

```
<xsd:sequence>
<xsd:sequence minOccurs = "0">
<xsd:group ref = "g_naming"/>
</xsd:sequence>
<xsd:element ref = "identity" minOccurs = "0"/>
<xsd:element ref = "helpFileList" minOccurs = "0"/>
<xsd:element ref = "toolList" minOccurs = "0"/>
<xsd:element ref = "pictureList" minOccurs = "0"/>
<xsd:element ref = "cfgItemList" minOccurs = "0"/>
<xsd:element ref = "additionalItemList" minOccurs = "0" maxOccurs = "unbounded"/>
<xsd:element ref = "logicalConnectionPointList" minOccurs = "0"/>
<xsd:element ref = "logicalConnectionPointAssemblyList" minOccurs = "0"/>
<xsd:element ref = "internalConnectionPointList" minOccurs = "0"/>
<xsd:element ref = "externalSchema" minOccurs = "0" maxOccurs = "unbounded"/>
</xsd:sequence>
<xsd:attribute name = "uniqueID" type = "xsd:ID"/>
<xsd:attribute name = "processingEntityType" use = "required" type = "xsd:string"/>
<xsd:attribute name = "enabled" default = "YES">
<xsd:simpleType>
<xsd:restriction base = "xsd:string">
<xsd:enumeration value = "YES"/>
<xsd:enumeration value = "NO"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
<xsd:element name = "productFamily">
<xsd:complexType>
<xsd:choice>
<xsd:group ref = "g_labels"/>
</xsd:choice>
<xsd:attribute name = "readOnly" default = "YES">
<xsd:simpleType>
<xsd:restriction base = "xsd:string">
<xsd:enumeration value = "YES"/>
<xsd:enumeration value = "NO"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
<xsd:element name = "productID">
<xsd:complexType>
<xsd:choice>
<xsd:group ref = "g_labels"/>
</xsd:choice>
<xsd:attribute name = "readOnly" default = "YES">
<xsd:simpleType>
<xsd:restriction base = "xsd:string">
<xsd:enumeration value = "YES"/>
<xsd:enumeration value = "NO"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
<xsd:element name = "productName">
<xsd:complexType>
<xsd:choice>
<xsd:group ref = "g_labels"/>
</xsd:choice>
```

```
<xsd:attribute name = "readOnly" default = "YES">
<xsd:simpleType>
<xsd:restriction base = "xsd:string">
<xsd:enumeration value = "YES"/>
<xsd:enumeration value = "NO"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
<xsd:element name = "productText">
<xsd:complexType>
<xsd:choice>
<xsd:group ref = "g_labels"/>
</xsd:choice>
<xsd:attribute name = "readOnly" default = "YES">
<xsd:simpleType>
<xsd:restriction base = "xsd:string">
<xsd:enumeration value = "YES"/>
<xsd:enumeration value = "NO"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
<xsd:element name = "ProfileBody">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref = "DeviceIdentity"/>
<xsd:element ref = "DeviceManager"/>
<xsd:element ref = "DeviceFunction" maxOccurs = "unbounded"/>
<xsd:element ref = "ApplicationProcess" minOccurs = "0" maxOccurs = "unbounded"/>
<xsd:element ref = "nonStandardizedExtension" minOccurs = "0" maxOccurs = "unbounded"/>
</xsd:sequence>
<xsd:attributeGroup ref = "ag_FDCML"/>
<xsd:attribute name = "supportedLanguages" type = "xsd:NMTOKENS"/>
<xsd:attribute name = "uniqueID" type = "xsd:ID"/>
</xsd:complexType>
</xsd:element>
<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"/>
<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:element name = "ProfilesBody">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref = "DeviceIdentity" minOccurs = "0"/>
<xsd:element ref = "ProfileBody" maxOccurs = "unbounded"/>
<xsd:element ref = "connectionList"/>
</xsd:sequence>
```

```
<xsd:attributeGroup ref = "ag_FDCML"/>
<xsd:attribute name = "supportedLanguages" type = "xsd:NMTOKENS"/>
</xsd:complexType>
</xsd:element>
<xsd:element name = "provides">
<xsd:complexType>
<xsd:attribute name = "ref" use = "required" type = "xsd:string"/>
</xsd:complexType>
</xsd:element>
<xsd:element name = "range">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref = "minVal" minOccurs = "0"/>
<xsd:element ref = "maxVal" minOccurs = "0"/>
<xsd:element ref = "stepVal" minOccurs = "0"/>
<xsd:element ref = "offset" minOccurs = "0"/>
<xsd:element ref = "gain" minOccurs = "0"/>
<xsd:element ref = "default" minOccurs = "0"/>
<xsd:element ref = "on" minOccurs = "0" maxOccurs = "unbounded"/>
</xsd:sequence>
<xsd:attribute name = "format" type = "xsd:string"/>
<xsd:attribute name = "unit" type = "xsd:string"/>
</xsd:complexType>
</xsd:element>
<xsd:element name = "recalcEntity">
<xsd:complexType>
<xsd:attribute name = "node" use = "required" type = "xsd:string"/>
</xsd:complexType>
</xsd:element>
<xsd:element name = "recalcValue">
<xsd:complexType>
<xsd:attribute name = "node" use = "required" type = "xsd:string"/>
</xsd:complexType>
</xsd:element>
<xsd:element name = "reference">
<xsd:complexType>
<xsd:attribute name = "ref" use = "required" type = "xsd:string"/>
</xsd:complexType>
</xsd:element>
<xsd:element name = "relations">
<xsd:complexType>
<xsd:choice maxOccurs = "unbounded">
<xsd:element ref = "enable"/>
<xsd:element ref = "disable"/>
<xsd:element ref = "change"/>
</xsd:choice>
</xsd:complexType>
</xsd:element>
<xsd:element name = "serialNumber">
<xsd:complexType>
<xsd:choice>
<xsd:group ref = "g_labels"/>
</xsd:choice>
<xsd:attribute name = "readOnly" default = "YES">
<xsd:simpleType>
<xsd:restriction base = "xsd:string">
<xsd:enumeration value = "YES"/>
<xsd:enumeration value = "NO"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:complexType>
```

```
</xsd:element>
<xsd:element name = "slot">
<xsd:complexType>
<xsd:sequence>
<xsd:group ref = "g_naming"/>
<xsd:element ref = "MAUUsageList"/>
<xsd:element ref = "defaultfile" minOccurs = "0"/>
<xsd:element ref = "file" maxOccurs = "unbounded"/>
<xsd:element ref = "specificProperty" minOccurs = "0" maxOccurs = "unbounded"/>
<xsd:element ref = "instances" minOccurs = "0"/>
</xsd:sequence>
<xsd:attribute name = "uniqueID" use = "required" type = "xsd:ID"/>
<xsd:attribute name = "number" use = "required" type = "xsd:string"/>
<xsd:attribute name = "enabled" default = "YES">
<xsd:simpleType>
<xsd:restriction base = "xsd:string">
<xsd:enumeration value = "YES"/>
<xsd:enumeration value = "NO"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
<xsd:element name = "slotList">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref = "slot" maxOccurs = "unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name = "specificationRevision">
<xsd:complexType>
<xsd:choice>
<xsd:group ref = "g_labels"/>
</xsd:choice>
<xsd:attribute name = "readOnly" default = "YES">
<xsd:simpleType>
<xsd:restriction base = "xsd:string">
<xsd:enumeration value = "YES"/>
<xsd:enumeration value = "NO"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
<xsd:element name = "specificProperty">
<xsd:complexType>
<xsd:sequence>
<xsd:sequence minOccurs = "0">
<xsd:group ref = "g_naming"/>
</xsd:sequence>
<xsd:group ref = "g_values"/>
</xsd:sequence>
<xsd:attribute name = "propertyType" use = "required" type = "xsd:string"/>
<xsd:attribute name = "enabled" default ="YES">
<xsd:simpleType>
<xsd:restriction base = "xsd:string">
<xsd:enumeration value = "NO"/>
<xsd:enumeration value = "YES"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
```

```
</xsd:complexType>
</xsd:element>
<xsd:element name = "stepVal">
<xsd:complexType>
<xsd:choice>
<xsd:group ref = "g_numericDatatypes"/>
<xsd:group ref = "g_booleanDatatypes"/>
<xsd:group ref = "g_userDatatypes"/>
<xsd:group ref = "g_timeDatatypes"/>
</xsd:choice>
</xsd:complexType>
</xsd:element>
<xsd:element name = "structuredType">
<xsd:complexType>
<xsd:sequence>
<xsd:choice minOccurs = "0">
<xsd:group ref = "g_help"/>
</xsd:choice>
<xsd:choice maxOccurs = "unbounded">
<xsd:element ref = "varDeclaration"/>
<xsd:element ref = "subrangeVarDeclaration"/>
</xsd:choice>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name = "subrange">
<xsd:complexType>
<xsd:attribute name = "lowerLimit" use = "required" type = "xsd:string"/>
<xsd:attribute name = "upperLimit" use = "required" type = "xsd:string"/>
</xsd:complexType>
</xsd:element>
<xsd:element name = "subrangeType">
<xsd:complexType>
<xsd:sequence>
<xsd:choice minOccurs = "0">
<xsd:group ref = "g_help"/>
</xsd:choice>
<xsd:group ref = "g_integerDatatypes"/>
<xsd:element ref = "subrange"/>
</xsd:sequence>
<xsd:attribute name = "initialValue" type = "xsd:string"/>
</xsd:complexType>
</xsd:element>
<xsd:element name = "subrangeVarDeclaration">
<xsd:complexType>
<xsd:sequence>
<xsd:group ref = "g_naming"/>
<xsd:group ref = "g_integerDatatypes"/>
<xsd:element ref = "subrange" minOccurs = "0"/>
</xsd:sequence>
<xsd:attribute name = "initialValue" type = "xsd:string"/>
</xsd:complexType>
</xsd:element>
<xsd:element name = "tool">
<xsd:complexType>
<xsd:attribute name = "toolClassification" use = "required" type = "xsd:string"/>
<xsd:attribute name = "toolID" use = "required" type = "xsd:string"/>
</xsd:complexType>
</xsd:element>
<xsd:element name = "toolList">
<xsd:complexType>
<xsd:sequence>
```

```
<xsd:element ref = "tool" maxOccurs = "unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name = "typeName">
<xsd:complexType>
<xsd:choice>
<xsd:group ref = "g_labels"/>
</xsd:choice>
<xsd:attribute name = "readOnly" default = "YES">
<xsd:simpleType>
<xsd:restriction base = "xsd:string">
<xsd:enumeration value = "YES"/>
<xsd:enumeration value = "NO"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
<xsd:element name = "uses">
<xsd:complexType>
<xsd:attribute name = "ref" use = "required" type = "xsd:string"/>
</xsd:complexType>
</xsd:element>
<xsd:element name = "varDeclaration">
<xsd:complexType>
<xsd:sequence>
<xsd:group ref = "g_naming"/>
<xsd:choice>
<xsd:group ref = "g_datatypes"/>
<xsd:element ref = "uses"/>
</xsd:choice>
<xsd:element ref = "subrange" minOccurs = "0" maxOccurs = "unbounded"/>
</xsd:sequence>
<xsd:attribute name = "initialValue" type = "xsd:string"/>
</xsd:complexType>
</xsd:element>
<xsd:element name = "vendorID">
<xsd:complexType>
<xsd:choice>
<xsd:group ref = "g_labels"/>
</xsd:choice>
<xsd:attribute name = "readOnly" default = "YES">
<xsd:simpleType>
<xsd:restriction base = "xsd:string">
<xsd:enumeration value = "YES"/>
<xsd:enumeration value = "NO"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
<xsd:element name = "vendorName">
<xsd:complexType>
<xsd:choice>
<xsd:group ref = "g_labels"/>
</xsd:choice>
<xsd:attribute name = "readOnly" default = "YES">
<xsd:simpleType>
<xsd:restriction base = "xsd:string">
<xsd:enumeration value = "YES"/>
<xsd:enumeration value = "NO"/>
```

```
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
<xsd:element name = "vendorText">
<xsd:complexType>
<xsd:choice>
<xsd:group ref = "g_labels"/>
</xsd:choice>
<xsd:attribute name = "readOnly" default = "YES">
<xsd:simpleType>
<xsd:restriction base = "xsd:string">
<xsd:enumeration value = "YES"/>
<xsd:enumeration value = "NO"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
<xsd:element name = "version">
<xsd:complexType>
<xsd:choice>
<xsd:group ref = "g_labels"/>
</xsd:choice>
<xsd:attribute name = "versionType" use = "required" type = "xsd:string"/>
<xsd:attribute name = "readOnly" default = "YES">
<xsd:simpleType>
<xsd:restriction base = "xsd:string">
<xsd:enumeration value = "YES"/>
<xsd:enumeration value = "NO"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
<xsd:element name = "yes">
<xsd:complexType>
<xsd:sequence>
<xsd:group ref = "g_naming"/>
<xsd:element ref = "relations" minOccurs = "0"/>
</xsd:sequence>
<xsd:attribute name = "default" default = "NO">
<xsd:simpleType>
<xsd:restriction base = "xsd:string">
<xsd:enumeration value = "YES"/>
<xsd:enumeration value = "NO"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
<xsd:element name = "associatedParameter">
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base = "xsd:string">
<xsd:attribute name = "associatedParameterValue" use = "required" type = "xsd:string"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
```

```
<xsd:element name = "configParametersList">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref = "parameterRef" maxOccurs = "unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name = "constraint">
<xsd:annotation>
<xsd:documentation>The elements under Constraints aren't checked by the parser. The languages,
to use to describe the constraints, don't refer to a schema.
</xsd:documentation>
</xsd:annotation>
<xsd:complexType>
<xsd:sequence>
<xsd:any namespace = "##any" processContents = "skip" minOccurs = "0" maxOccurs =
"unbounded"/>
</xsd:sequence>
<xsd:attribute name = "constraintID" use = "required">
<xsd:simpleType>
<xsd:restriction base = "xsd:ID">
<xsd:pattern value = "C\d{1,6}"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
<xsd:attribute name = "constraintType" use = "required">
<xsd:simpleType>
<xsd:restriction base = "xsd:NMTOKEN">
<xsd:enumeration value = "among_parameters"/>
<xsd:enumeration value = "display"/>
<xsd:enumeration value = "network"/>
<xsd:enumeration value = "device"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
<xsd:attribute name = "constraintLanguage" use = "required" type = "xsd:string"/>
</xsd:complexType>
</xsd:element>
<xsd:element name = "constraintsList">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref = "constraint" maxOccurs = "unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name = "commandParameter">
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base = "xsd:string">
<xsd:attribute name = "commandParameterValue" use = "required" type = "xsd:string"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:element name = "defaultValue">
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base = "xsd:string">
<xsd:attributeGroup ref = "type"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
```

```
</xsd:element>
<xsd:element name = "function">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref = "inputsList" minOccurs = "0"/>
<xsd:element ref = "outputsList" minOccurs = "0"/>
<xsd:element ref = "configParametersList" minOccurs = "0"/>
<xsd:element ref = "constraintsList" minOccurs = "0"/>
<xsd:element name = "functionStateTransitionDiagram" minOccurs = "0">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref = "stateTransitionDiagram"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
<xsd:attribute name = "name" use = "required" type = "xsd:string"/>
<xsd:attribute name = "description" type = "xsd:string"/>
<xsd:attribute name = "functionID" use = "required" type = "xsd:ID"/>
<xsd:attribute name = "version" use = "required" type = "xsd:string"/>
</xsd:complexType>
</xsd:element>
<xsd:element name = "functionRef">
<xsd:annotation>
<xsd:documentation>By this way the hierarchical view of the device can be describe. There is no
limitation in the level number.</xsd:documentation>
</xsd:annotation>
<xsd:complexType>
<xsd:sequence>
<xsd:element ref = "functionRef" minOccurs = "0" maxOccurs = "unbounded"/>
</xsd:sequence>
<xsd:attribute name = "functionName" use = "required" type = "xsd:string"/>
<xsd:attribute name = "functionIDREF" use = "required" type = "xsd:string"/>
<xsd:attribute name = "functionURL" type = "xsd:string"/>
<xsd:attribute name = "functionDescription" type = "xsd:string"/>
</xsd:complexType>
</xsd:element>
<xsd:element name = "highLimit">
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base = "xsd:string">
<xsd:attributeGroup ref = "type"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:element name = "inputsList">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref = "parameterRef" maxOccurs = "unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name = "listOfValues">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref = "value" maxOccurs = "unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name = "lowLimit">
<xsd:complexType>
```

```
<xsd:simpleContent>
<xsd:extension base = "xsd:string">
<xsd:attributeGroup ref = "type"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:element name = "member">
<xsd:annotation>
<xsd:documentation>For dataType = array, only first member is needed</xsd:documentation>
</xsd:annotation>
<xsd:complexType>
<xsd:sequence>
<xsd:element ref = "defaultValue" minOccurs = "0"/>
<xsd:element ref = "lowLimit" minOccurs = "0"/>
<xsd:element ref = "highLimit" minOccurs = "0"/>
<xsd:element ref = "unit" minOccurs = "0"/>
<xsd:element ref = "listOfValues" minOccurs = "0"/>
<xsd:element ref = "membersList" minOccurs = "0"/>
<xsd:sequence minOccurs = "0">
<xsd:group ref = "g_naming"/>
</xsd:sequence>
<xsd:element ref = "actualValue" minOccurs = "0"/>
</xsd:sequence>
<xsd:attribute name = "name" use = "required" type = "xsd:string"/>
<xsd:attribute name = "description" type = "xsd:string"/>
<xsd:attribute name = "memberID" use = "required" type = "xsd:string"/>
<xsd:attributeGroup ref = "dataType"/>
</xsd:complexType>
</xsd:element>
<xsd:element name = "membersList">
<xsd:annotation>
<xsd:documentation>membersList is mandatory if dataType = struct or array.</xsd:documentation>
</xsd:annotation>
<xsd:complexType>
<xsd:sequence>
<xsd:element ref = "member" maxOccurs = "unbounded"/>
</xsd:sequence>
<xsd:attribute name = "numberOfMembers" use = "required" type = "xsd:integer"/>
</xsd:complexType>
</xsd:element>
<xsd:element name = "outputsList">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref = "parameterRef" maxOccurs = "unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name = "parameter" type = "parameterType"/>
<xsd:element name = "state" type = "stateType"/>
<xsd:element name = "statesTab">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref = "state" maxOccurs = "unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name = "stateTransitionDiagram" type = "stateTransitionDiagramType"/>
<xsd:element name = "transition" type = "transitionType"/>
<xsd:element name = "transtionsTab">
<xsd:complexType>
```

```
<xsd:sequence>
<xsd:element ref = "transition" maxOccurs = "unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name = "unit">
<xsd:complexType>
<xsd:attribute name = "multiplier" use = "required" type = "xsd:string"/>
<xsd:attribute name = "unitURI" type = "xsd:anyURI"/>
</xsd:complexType>
</xsd:element>
<xsd:element name = "value">
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base = "xsd:string">
<xsd:attribute name = "meaning" type = "xsd:string"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:complexType name = "parameterType">
<xsd:sequence>
<xsd:element ref = "defaultValue" minOccurs = "0"/>
<xsd:element ref = "lowLimit" minOccurs = "0"/>
<xsd:element ref = "highLimit" minOccurs = "0"/>
<xsd:element ref = "unit" minOccurs = "0"/>
<xsd:element ref = "listOfValues" minOccurs = "0"/>
<xsd:element ref = "membersList" minOccurs = "0"/>
<xsd:element ref = "actualValue" minOccurs = "0"/>
<xsd:sequence minOccurs = "0">
<xsd:group ref = "g_naming"/>
</xsd:sequence>
</xsd:sequence>
<xsd:attribute name = "name" use = "required" type = "xsd:string"/>
<xsd:attribute name = "denotation" type = "xsd:string"/>
<xsd:attribute name = "description" type = "xsd:string"/>
<xsd:attribute name = "paramID" use = "required" type = "xsd:ID"/>
<xsd:attributeGroup ref = "accessCategory"/>
<xsd:attribute name = "constraintIDREF" type = "xsd:IDREF"/>
<xsd:attributeGroup ref = "dataType"/>
<xsd:attributeGroup ref = "persistent"/>
<xsd:attributeGroup ref = "use"/>
</xsd:complexType>
<xsd:complexType name = "stateType">
<xsd:sequence>
<xsd:element ref = "associatedParameter"/>
</xsd:sequence>
<xsd:attribute name = "description" type = "xsd:string"/>
<xsd:attribute name = "initialState" use = "required">
<xsd:simpleType>
<xsd:restriction base = "xsd:NMTOKEN">
<xsd:enumeration value = "false"/>
<xsd:enumeration value = "true"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
<xsd:attribute name = "stateName" use = "required" type = "xsd:ID"/>
<xsd:attribute name = "stateParent" type = "xsd:string"/>
</xsd:complexType>
<xsd:complexType name = "transitionType">
<xsd:sequence>
```

```
<xsd:element ref = "commandParameter" maxOccurs = "unbounded"/>
</xsd:sequence>
<xsd:attribute name = "description" type = "xsd:string"/>
<xsd:attribute name = "transitionNumber" use = "required">
<xsd:annotation>
<xsd:documentation>Number format = Tx to Txxxxxx</xsd:documentation>
</xsd:annotation>
<xsd:simpleType>
<xsd:restriction base = "xsd:ID">
<xsd:pattern value = "T\d{1,6}"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
<xsd:attribute name = "fromState" use = "required" type = "xsd:IDREF"/>
<xsd:attribute name = "toState" use = "required" type = "xsd:IDREF"/>
</xsd:complexType>
<xsd:complexType name = "stateTransitionDiagramType">
<xsd:sequence>
<xsd:element ref = "statesTab"/>
<xsd:element ref = "transtionsTab"/>
</xsd:sequence>
</xsd:complexType>
<xsd:element name = "actualValue" type = "xsd:string"/>
<xsd:element name = "CANopenIdentity">
<xsd:complexType>
<xsd:sequence>
<xsd:element name = "CANopenVendorId">
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base = "xsd:unsignedLong">
<xsd:attribute name = "paramId" use = "required" type = "xsd:ID"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:element name = "CANopenProductCode" minOccurs = "0">
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base = "xsd:unsignedLong">
<xsd:attribute name = "paramId" use = "required" type = "xsd:ID"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:element name = "CANopenRevisionNumber" minOccurs = "0">
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base = "xsd:unsignedLong">
<xsd:attribute name = "paramId" use = "required" type = "xsd:ID"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:element name = "CANopenSerialNumber" minOccurs = "0">
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base = "xsd:unsignedLong">
<xsd:attribute name = "paramId" use = "required" type = "xsd:ID"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
```

```
</xsd:element>
<xsd:element name = "CANopenManufacturerDeviceName" minOccurs = "0">
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base = "xsd:string">
<xsd:attribute name = "paramId" use = "required" type = "xsd:ID"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:element name = "CANopenManufacturerHardwareVersion" minOccurs = "0">
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base = "xsd:string">
<xsd:attribute name = "paramId" use = "required" type = "xsd:ID"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:element name = "CANopenManufacturerSoftwareVersion" minOccurs = "0">
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base = "xsd:string">
<xsd:attribute name = "paramId" use = "required" type = "xsd:ID"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name = "CANopenDedicatedCfgCategory">
<xsd:complexType>
<xsd:sequence>
<xsd:element name = "CANopenGeneralCapabilities">
<xsd:complexType/>
</xsd:element>
<xsd:element name = "CANopenDeviceCommissioning" minOccurs = "0">
<xsd:complexType/>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name = "parameterRef">
<xsd:complexType>
<xsd:attribute name = "paramID" use = "required" type = "xsd:IDREF"/>
</xsd:complexType>
</xsd:element>
<xsd:element name = "CANopenObjectAccessList">
<xsd:complexType>
<xsd:sequence>
<xsd:element name = "CANopenObject" maxOccurs = "65536">
<xsd:complexType>
<xsd:sequence>
<xsd:element name = "CANopenSubObject" maxOccurs = "255">
<xsd:complexType>
<xsd:attribute name = "subIndex" use = "required" type = "xsd:unsignedByte"/>
<xsd:attribute name = "pdoMapping" use = "required">
<xsd:simpleType>
```

```
<xsd:restriction base = "xsd:NMTOKENS">
<xsd:enumeration value = "NotAllowed"/>
<xsd:enumeration value = "ReceiveOnly"/>
<xsd:enumeration value = "TransmitOnly"/>
<xsd:enumeration value = "ReceiveTransmit"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
<xsd:attribute name = "paramID" type = "xsd:IDREF"/>
<xsd:attribute name = "maxSubNumber" type = "xsd:unsignedByte"/>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
<xsd:attribute name = "index" use = "required" type = "xsd:unsignedShort"/>
<xsd:attribute name = "objectCode" use = "required" type = "xsd:unsignedByte"/>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
</xsd:schema>
```

B.1.5.2 FDCMLdt.xsd

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema targetNamespace="gost.e-ecolog.ru
xmlns="gost.e-ecolog.ru
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xlink="http://gost.e-ecolog.ru
elementFormDefault="qualified"
attributeFormDefault="unqualified"
version="1.0">
<xsd:include schemaLocation="FDCMLdt.xsd"/>
<xsd:import namespace="gost.e-ecolog.ru                    schemaLocation="xlinkdef.xsd"/>
<xsd:element name="binaryDate" nillable="true">
<xsd:annotation>
<xsd:documentation>Binary Date w/ 7 octets</xsd:documentation>
</xsd:annotation>
<xsd:simpleType>
<xsd:restriction base="xsd:string"/>
</xsd:simpleType>
</xsd:element>
<xsd:element name="binaryDate2000" nillable="true">
<xsd:annotation>
<xsd:documentation>Binary Date w/ 8 octets</xsd:documentation>
</xsd:annotation>
<xsd:simpleType>
<xsd:restriction base="xsd:string"/>
</xsd:simpleType>
</xsd:element>
<xsd:element name="binaryTime0" nillable="true">
<xsd:annotation>
<xsd:documentation>binary number w/ 2 octets, unit is 10 us</xsd:documentation>
</xsd:annotation>
<xsd:simpleType>
<xsd:restriction base="xsd:unsignedShort"/>
</xsd:simpleType>
</xsd:element>
<xsd:element name="binaryTime1" nillable="true">
<xsd:annotation>
<xsd:documentation>binary number w/ 2 octets, unit is 100 us</xsd:documentation>
</xsd:annotation>
<xsd:simpleType>
```

```
<xsd:restriction base="xsd:unsignedShort"/>
</xsd:simpleType>
</xsd:element>
<xsd:element name="binaryTime2" nillable="true">
<xsd:annotation>
<xsd:documentation>binary number w/ 2 octets, unit is 1 ms</xsd:documentation>
</xsd:annotation>
<xsd:simpleType>
<xsd:restriction base="xsd:unsignedShort"/>
</xsd:simpleType>
</xsd:element>
<xsd:element name="binaryTime3" nillable="true">
<xsd:annotation>
<xsd:documentation>binary number w/ 2 octets, unit is 10 ms</xsd:documentation>
</xsd:annotation>
<xsd:simpleType>
<xsd:restriction base="xsd:unsignedShort"/>
</xsd:simpleType>
</xsd:element>
<xsd:element name="binaryTime4" nillable="true">
<xsd:annotation>
<xsd:documentation>binary number w/ 4 octets, unit is 10 us</xsd:documentation>
</xsd:annotation>
<xsd:simpleType>
<xsd:restriction base="xsd:unsignedInt"/>
</xsd:simpleType>
</xsd:element>
<xsd:element name="binaryTime5" nillable="true">
<xsd:annotation>
<xsd:documentation>binary number w/ 4 octets, unit is 100 us</xsd:documentation>
</xsd:annotation>
<xsd:simpleType>
<xsd:restriction base="xsd:unsignedInt"/>
</xsd:simpleType>
</xsd:element>
<xsd:element name="binaryTime6" nillable="true">
<xsd:annotation>
<xsd:documentation>binary number w/ 4 octets, unit is 1 ms</xsd:documentation>
</xsd:annotation>
<xsd:simpleType>
<xsd:restriction base="xsd:unsignedInt"/>
</xsd:simpleType>
</xsd:element>
<xsd:element name="binaryTime7" nillable="true">
<xsd:annotation>
<xsd:documentation>binary number w/ 6 octets, unit is 1 ms</xsd:documentation>
</xsd:annotation>
<xsd:simpleType>
<xsd:restriction base="xsd:unsignedLong">
<xsd:maxInclusive value="281474976710655"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:element>
<xsd:element name="binaryTime8" nillable="true">
<xsd:annotation>
<xsd:documentation>binary number w/ 6 octets, unit is 10 us</xsd:documentation>
</xsd:annotation>
<xsd:simpleType>
<xsd:restriction base="xsd:unsignedLong">
<xsd:maxInclusive value="281474976710655"/>
</xsd:restriction>
```

```
</xsd:simpleType>
</xsd:element>
<xsd:element name="binaryTime9" nillable="true">
<xsd:annotation>
<xsd:documentation>binary number w/ 6 octets, unit is 100 us</xsd:documentation>
</xsd:annotation>
<xsd:simpleType>
<xsd:restriction base="xsd:unsignedLong">
<xsd:maxInclusive value="281474976710655"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:element>
<xsd:simpleType name="t_bitstring">
<xsd:restriction base="xsd:string">
<xsd:pattern value="(([0-1]{8})_?)*([0-1]{8})"/>
</xsd:restriction>
</xsd:simpleType>
<xsd:element name="bitstring" nillable="true">
<xsd:annotation>
<xsd:documentation>sting of bits (length in octets)</xsd:documentation>
</xsd:annotation>
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base="t_bitstring">
<xsd:attribute name="length" type="xsd:nonNegativeInteger" use="required"/>
<xsd:attribute name="IEC61131" type="xsd:string" use="optional"/>
<xsd:attribute name="WinVT" type="xsd:string" use="optional"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:simpleType name="t_bitstring16">
<xsd:restriction base="xsd:string">
<xsd:pattern value="0x([0-9][A-F]){4}"/>
</xsd:restriction>
</xsd:simpleType>
<xsd:simpleType name="t_bitstring32">
<xsd:restriction base="xsd:string">
<xsd:pattern value="0x([0-9][A-F]){8}"/>
</xsd:restriction>
</xsd:simpleType>
<xsd:simpleType name="t_bitstring64">
<xsd:restriction base="xsd:string">
<xsd:pattern value="0x([0-9][A-F]){16}"/>
</xsd:restriction>
</xsd:simpleType>
<xsd:simpleType name="t_bitstring8">
<xsd:restriction base="xsd:string">
<xsd:pattern value="0x([0-9][A-F]){2}"/>
</xsd:restriction>
</xsd:simpleType>
<xsd:element name="boolean" nillable="true">
<xsd:annotation>
<xsd:documentation>one bit representing the values true and false</xsd:documentation>
</xsd:annotation>
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base="xsd:boolean">
<xsd:attribute name="IEC61131" type="xsd:string" fixed="BOOL"/>
<xsd:attribute name="WinVT" type="xsd:string" fixed="VT_BOOL"/>
</xsd:extension>
</xsd:simpleContent>
```

```
</xsd:complexType>
</xsd:element>
<xsd:element name="bool" nillable="true">
<xsd:annotation>
<xsd:documentation>octet representing the values true and false</xsd:documentation>
</xsd:annotation>
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base="xsd:boolean">
<xsd:attribute name="IEC61131" type="xsd:string" fixed="BOOL"/>
<xsd:attribute name="WinVT" type="xsd:string" fixed="VT_BOOL"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="byte" nillable="true">
<xsd:annotation>
<xsd:documentation>string of 8 bit (byte)</xsd:documentation>
</xsd:annotation>
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base="t_bitstring8">
<xsd:attribute name="IEC61131" type="xsd:string" fixed="BYTE"/>
<xsd:attribute name="WinVT" type="xsd:string" fixed="VT_UI1"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:simpleType name="t_compactBooleanArray">
<xsd:restriction base="xsd:string">
<xsd:pattern value="0x([0-9]|[A-F]){2}*/>
</xsd:restriction>
</xsd:simpleType>
<xsd:element name="compactBooleanArray" nillable="true">
<xsd:annotation>
<xsd:documentation>array of 8 bits, length one octet</xsd:documentation>
</xsd:annotation>
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base="t_compactBooleanArray">
<xsd:attribute name="IEC61131" type="xsd:string" fixed="BYTE"/>
<xsd:attribute name="WinVT" type="xsd:string" fixed="VT_UI1"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="date_and_time" nillable="true">
<xsd:annotation>
<xsd:documentation>structured datatype consiting of timeOfDay4 and date2</xsd:documentation>
</xsd:annotation>
<xsd:complexType>
<xsd:annotation>
<xsd:documentation>represented by ISO 8601 dateTime</xsd:documentation>
</xsd:annotation>
<xsd:simpleContent>
<xsd:extension base="xsd:dateTime">
<xsd:attribute name="IEC61131" type="xsd:string" fixed="DATE_AND_TIME"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
```

```
<xsd:element name="date" nillable="true">
<xsd:annotation>
<xsd:documentation>Date without time indication</xsd:documentation>
</xsd:annotation>
<xsd:complexType>
<xsd:annotation>
<xsd:documentation>represented by ISO 8601 date</xsd:documentation>
</xsd:annotation>
<xsd:simpleContent>
<xsd:extension base="xsd:date">
<xsd:attribute name="IEC61131" type="xsd:string" fixed="DATE"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="date7" nillable="true">
<xsd:annotation>
<xsd:documentation>Date with time indication</xsd:documentation>
</xsd:annotation>
<xsd:complexType>
<xsd:annotation>
<xsd:documentation>restricted later ?</xsd:documentation>
</xsd:annotation>
<xsd:simpleContent>
<xsd:extension base="xsd:string"/>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="dint" nillable="true">
<xsd:annotation>
<xsd:documentation>binary number w/ 4 octets</xsd:documentation>
</xsd:annotation>
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base="xsd:int">
<xsd:attribute name="IEC61131" type="xsd:string" fixed="DINT"/>
<xsd:attribute name="WinVT" type="xsd:string" fixed="VT_I4"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="dword" nillable="true">
<xsd:annotation>
<xsd:documentation>string of 32 bit (dword)</xsd:documentation>
</xsd:annotation>
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base="t_bitstring32">
<xsd:attribute name="IEC61131" type="xsd:string" fixed="DWORD"/>
<xsd:attribute name="WinVT" type="xsd:string" fixed="VT_UI4"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="fieldbusTime" nillable="true">
<xsd:annotation>
<xsd:documentation>IEC 61158-4 DL-Time</xsd:documentation>
</xsd:annotation>
<xsd:complexType>
<xsd:annotation>
<xsd:documentation>restricted later ?</xsd:documentation>
</xsd:annotation>
```

```
<xsd:simpleContent>
<xsd:extension base="xsd:string"/>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="fTime" nillable="true">
<xsd:annotation>
<xsd:documentation>binary number w/ 4 octets, unit is 1 us</xsd:documentation>
</xsd:annotation>
<xsd:complexType>
<xsd:annotation>
<xsd:documentation>4 octets, unit is 1 us</xsd:documentation>
</xsd:annotation>
<xsd:simpleContent>
<xsd:extension base="xsd:int">
<xsd:attribute name="IEC61131" type="xsd:string" fixed="DINT"/>
<xsd:attribute name="WinVT" type="xsd:string" fixed="VT_I4"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="int" nillable="true">
<xsd:annotation>
<xsd:documentation>binary number w/ 2 octets</xsd:documentation>
</xsd:annotation>
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base="xsd:short">
<xsd:attribute name="IEC61131" type="xsd:string" fixed="INT"/>
<xsd:attribute name="WinVT" type="xsd:string" fixed="VT_I2"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="IPV4Address" nillable="true">
<xsd:annotation>
<xsd:documentation>IP V4 address in dotted notation</xsd:documentation>
</xsd:annotation>
<xsd:simpleType>
<xsd:restriction base="xsd:string">
<xsd:pattern value="((([0-9])|([1-9][0-9])|(1[0-9][0-9])|(2[0-4][0-9])|(25[0-5]))\.){3}(([0-
9])|([1-9][0-9])|(1[0-9][0-9])|(2[0-4][0-9])|(25[0-5]))"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:element>
<xsd:element name="IPV6Address" nillable="true">
<xsd:annotation>
<xsd:documentation>IP V6 address in dotted notation</xsd:documentation>
</xsd:annotation>
<xsd:simpleType>
<xsd:restriction base="xsd:string">
<xsd:pattern value="((([0-9])|([1-9][0-9])|(1[0-9][0-9])|(2[0-4][0-9])|(25[0-5]))\.){5}(([0-
9])|([1-9][0-9])|(1[0-9][0-9])|(2[0-4][0-9])|(25[0-5]))"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:element>
<xsd:element name="iTime" nillable="true">
<xsd:annotation>
<xsd:documentation>binary number w/ 2 octets, unit is 1 ms</xsd:documentation>
</xsd:annotation>
<xsd:complexType>
<xsd:annotation>
```

```
<xsd:documentation>2 octets, unit is 1 ms</xsd:documentation>
</xsd:annotation>
<xsd:simpleContent>
<xsd:extension base="xsd:short">
<xsd:attribute name="IEC61131" type="xsd:string" fixed="INT"/>
<xsd:attribute name="WinVT" type="xsd:string" fixed="VT_I2"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="lint" nillable="true">
<xsd:annotation>
<xsd:documentation>binary number w/ 8 octets</xsd:documentation>
</xsd:annotation>
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base="xsd:long">
<xsd:attribute name="IEC61131" type="xsd:string" fixed="LINT"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="lreal" nillable="true">
<xsd:annotation>
<xsd:documentation>ANSI/IEEE 754 double precision</xsd:documentation>
</xsd:annotation>
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base="xsd:double">
<xsd:attribute name="IEC61131" type="xsd:string" fixed="LREAL"/>
<xsd:attribute name="WinVT" type="xsd:string" fixed="VT_R8"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="ltime" nillable="true">
<xsd:annotation>
<xsd:documentation>binary number w/ 8 octets, unit is 1 us</xsd:documentation>
</xsd:annotation>
<xsd:complexType>
<xsd:annotation>
<xsd:documentation>8 octets, unit is 1 us</xsd:documentation>
</xsd:annotation>
<xsd:simpleContent>
<xsd:extension base="xsd:long">
<xsd:attribute name="IEC61131" type="xsd:string" fixed="LINT"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="lword" nillable="true">
<xsd:annotation>
<xsd:documentation>string of 64 bit (lword)</xsd:documentation>
</xsd:annotation>
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base="t_bitstring64">
<xsd:attribute name="IEC61131" type="xsd:string" fixed="LWORD"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
```

```
<xsd:simpleType name="t_octetString">
<xsd:restriction base="xsd:string">
<xsd:pattern value="(0x([0-9][A-F]){2},)*0x([0-9][A-F]){2}"/>
</xsd:restriction>
</xsd:simpleType>
<xsd:element name="octetString" nillable="true">
<xsd:annotation>
<xsd:documentation>ordered sequence of octets, length in octets</xsd:documentation>
</xsd:annotation>
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base="t_octetString">
<xsd:attribute name="length" type="xsd:nonNegativeInteger" use="required"/>
<xsd:attribute name="WinVT" type="xsd:string" fixed="VT_BSTR"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:simpleType name="t_octetString1">
<xsd:restriction base="xsd:string">
<xsd:pattern value="0x([0-9][A-F]){2}"/>
</xsd:restriction>
</xsd:simpleType>
<xsd:element name="octetString1" nillable="true">
<xsd:annotation>
<xsd:documentation>1 octet</xsd:documentation>
</xsd:annotation>
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base="t_octetString1">
<xsd:attribute name="IEC61131" type="xsd:string" fixed="BYTE"/>
<xsd:attribute name="WinVT" type="xsd:string" fixed="VT_UI1"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:simpleType name="t_octetString16">
<xsd:restriction base="xsd:string">
<xsd:pattern value="0x([0-9][A-F]){32}"/>
</xsd:restriction>
</xsd:simpleType>
<xsd:element name="octetString16" nillable="true">
<xsd:annotation>
<xsd:documentation>ordered sequence of 16 octets</xsd:documentation>
</xsd:annotation>
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base="t_octetString16">
<xsd:attribute name="WinVT" type="xsd:string" fixed="VT_BSTR"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:simpleType name="t_octetString2">
<xsd:restriction base="xsd:string">
<xsd:pattern value="0x([0-9][A-F]){4}"/>
</xsd:restriction>
</xsd:simpleType>
<xsd:element name="octetString2" nillable="true">
<xsd:annotation>
<xsd:documentation>ordered sequence of 2 octets</xsd:documentation>
</xsd:annotation>
```

```
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base="t_octetString2">
<xsd:attribute name="IEC61131" type="xsd:string" fixed="WORD"/>
<xsd:attribute name="WinVT" type="xsd:string" fixed="VT_UI2"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:simpleType name="t_octetString4">
<xsd:restriction base="xsd:string">
<xsd:pattern value="0x([0-9][A-F]){8}"/>
</xsd:restriction>
</xsd:simpleType>
<xsd:element name="octetString4" nillable="true">
<xsd:annotation>
<xsd:documentation>ordered sequence of 4 octets</xsd:documentation>
</xsd:annotation>
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base="t_octetString4">
<xsd:attribute name="IEC61131" type="xsd:string" fixed="DWORD"/>
<xsd:attribute name="WinVT" type="xsd:string" fixed="VT_UI4"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:simpleType name="t_octetString8">
<xsd:restriction base="xsd:string">
<xsd:pattern value="0x([0-9][A-F]){16}"/>
</xsd:restriction>
</xsd:simpleType>
<xsd:element name="octetString8" nillable="true">
<xsd:annotation>
<xsd:documentation>ordered sequence of 8 octets</xsd:documentation>
</xsd:annotation>
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base="t_octetString8">
<xsd:attribute name="IEC61131" type="xsd:string" fixed="LWORD"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="real" nillable="true">
<xsd:annotation>
<xsd:documentation>ANSI/IEEE 754 single precision</xsd:documentation>
</xsd:annotation>
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base="xsd:float">
<xsd:attribute name="IEC61131" type="xsd:string" fixed="REAL"/>
<xsd:attribute name="WinVT" type="xsd:string" fixed="VT_R4"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="sint" nillable="true">
<xsd:annotation>
<xsd:documentation>binary number w/ 1 octet</xsd:documentation>
</xsd:annotation>
<xsd:complexType>
```

```
<xsd:simpleContent>
<xsd:extension base="xsd:byte">
<xsd:attribute name="IEC61131" type="xsd:string" fixed="SINT"/>
<xsd:attribute name="WinVT" type="xsd:string" fixed="VT_I1"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="time" nillable="true">
<xsd:annotation>
<xsd:documentation>binary number w/ 4 octets, unit is 1 ms</xsd:documentation>
</xsd:annotation>
<xsd:complexType>
<xsd:annotation>
<xsd:documentation>4 octets, unit is 1 ms</xsd:documentation>
</xsd:annotation>
<xsd:simpleContent>
<xsd:extension base="xsd:int">
<xsd:attribute name="IEC61131" type="xsd:string" fixed="TIME"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="timeDifference4" nillable="true">
<xsd:annotation>
<xsd:documentation>binary number w/ 4 octets, unit is 1 ms</xsd:documentation>
</xsd:annotation>
<xsd:complexType>
<xsd:annotation>
<xsd:documentation>TimeDifference without date indication</xsd:documentation>
</xsd:annotation>
<xsd:simpleContent>
<xsd:extension base="xsd:unsignedInt"/>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="timeDifference6" nillable="true">
<xsd:annotation>
<xsd:documentation>time difference with additional difference in days</xsd:documentation>
</xsd:annotation>
<xsd:complexType>
<xsd:annotation>
<xsd:documentation>restricted later ?</xsd:documentation>
</xsd:annotation>
<xsd:simpleContent>
<xsd:extension base="xsd:string"/>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="timeOfDay6" nillable="true">
<xsd:annotation>
<xsd:documentation>time of day with date indication</xsd:documentation>
</xsd:annotation>
<xsd:complexType>
<xsd:annotation>
<xsd:documentation>restricted later ?</xsd:documentation>
</xsd:annotation>
<xsd:simpleContent>
<xsd:extension base="xsd:string"/>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
```

```
<xsd:element name="time_of_day" nillable="true">
<xsd:annotation>
<xsd:documentation>time of day without date indication</xsd:documentation>
</xsd:annotation>
<xsd:complexType>
<xsd:annotation>
<xsd:documentation>TimeOfDay without date indication</xsd:documentation>
</xsd:annotation>
<xsd:simpleContent>
<xsd:extension base="xsd:unsignedInt">
<xsd:attribute name="IEC61131" type="xsd:string" fixed="TIME_OF_DAY"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="timeValue" nillable="true">
<xsd:annotation>
<xsd:documentation>binary number w/ 8 octets, unit is 1/32 ms</xsd:documentation>
</xsd:annotation>
<xsd:complexType>
<xsd:annotation>
<xsd:documentation>unit of time is 1/32 millisecond</xsd:documentation>
</xsd:annotation>
<xsd:simpleContent>
<xsd:extension base="xsd:unsignedLong"/>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="universalTime" nillable="true">
<xsd:annotation>
<xsd:documentation>12 octet string YYMMDDHHMMSS</xsd:documentation>
</xsd:annotation>
<xsd:simpleType>
<xsd:restriction base="xsd:string">
<xsd:pattern value="([0-9]{2})((0[1-9])|(1[0-2]))((0[1-9])|([1-2][0-9])|(3[0-1]))((
[0-1][0-
9])|(2[0-3]))((
[0-5][0-9]){2})"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:element>
<xsd:element name="udint" nillable="true">
<xsd:annotation>
<xsd:documentation>binary number w/ 4 octets and sign bit</xsd:documentation>
</xsd:annotation>
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base="xsd:unsignedInt">
<xsd:attribute name="IEC61131" type="xsd:string" fixed="UDINT"/>
<xsd:attribute name="WinVT" type="xsd:string" fixed="VT_UI4"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="uint" nillable="true">
<xsd:annotation>
<xsd:documentation>binary number w/ 2 octets and sign bit</xsd:documentation>
</xsd:annotation>
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base="xsd:unsignedShort">
<xsd:attribute name="IEC61131" type="xsd:string" fixed="UINT"/>
<xsd:attribute name="WinVT" type="xsd:string" fixed="VT_UI2"/>
</xsd:extension>
```

```
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="ulint" nillable="true">
<xsd:annotation>
<xsd:documentation>binary number w/ 8 octets and sign bit</xsd:documentation>
</xsd:annotation>
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base="xsd:unsignedLong">
<xsd:attribute name="IEC61131" type="xsd:string" fixed="ULINT"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="usint" nillable="true">
<xsd:annotation>
<xsd:documentation>binary number w/ 1 octet and sign bit</xsd:documentation>
</xsd:annotation>
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base="xsd:unsignedByte">
<xsd:attribute name="IEC61131" type="xsd:string" fixed="USINT"/>
<xsd:attribute name="WinVT" type="xsd:string" fixed="VT_UI1"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="visibleString" nillable="true">
<xsd:annotation>
<xsd:documentation>VisibleString1, VisibleString2, VisibleString4, VisibleString8,
VisibleString16, UNICODE Char, VisibleString, UNICODEString</xsd:documentation>
</xsd:annotation>
<xsd:complexType>
<xsd:annotation>
<xsd:documentation>this type resembles the 61158 datatypes visibleString[n], UNICODE char,
visibleString, UNICODEString, string, string1, string2, stringn, shortString</xsd:documentation>
</xsd:annotation>
<xsd:simpleContent>
<xsd:extension base="xsd:string">
<xsd:attribute ref="xml:lang" use="optional"/>
<xsd:attribute name="maxLen" type="xsd:nonNegativeInteger" use="optional"/>
<xsd:attribute name="charSet" type="xsd:string" use="optional"/>
<xsd:attribute name="IEC61131" type="xsd:string" fixed="STRING"/>
<xsd:attribute name="WinVT" type="xsd:string" fixed="BSTR"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="word" nillable="true">
<xsd:annotation>
<xsd:documentation>string of 16 bit (word)</xsd:documentation>
</xsd:annotation>
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base="t_bitstring16">
<xsd:attribute name="IEC61131" type="xsd:string" fixed="WORD"/>
<xsd:attribute name="WinVT" type="xsd:string" fixed="VT_UI2"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
```

```
<xsd:element name="array" nillable="true">
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base="xsd:string">
<xsd:attribute name="dim" type="xsd:nonNegativeInteger" use="required"/>
<xsd:attribute name="maxDim" type="xsd:nonNegativeInteger" use="optional"/>
<xsd:attribute name="embeddedDatatype" use="required">
<xsd:simpleType>
<xsd:restriction base="xsd:string">
<xsd:enumeration value="boolean"/>
<xsd:enumeration value="bool"/>
<xsd:enumeration value="sint"/>
<xsd:enumeration value="int"/>
<xsd:enumeration value="dint"/>
<xsd:enumeration value="lint"/>
<xsd:enumeration value="usint"/>
<xsd:enumeration value="uint"/>
<xsd:enumeration value="udint"/>
<xsd:enumeration value="ulint"/>
<xsd:enumeration value="real"/>
<xsd:enumeration value="lreal"/>
<xsd:enumeration value="byte"/>
<xsd:enumeration value="word"/>
<xsd:enumeration value="dword"/>
<xsd:enumeration value="lword"/>
<xsd:enumeration value="binaryDate"/>
<xsd:enumeration value="binaryDate2000"/>
<xsd:enumeration value="date7"/>
<xsd:enumeration value="date"/>
<xsd:enumeration value="timeOfDay6"/>
<xsd:enumeration value="time_of_day"/>
<xsd:enumeration value="timeDifference6"/>
<xsd:enumeration value="timeDifference4"/>
<xsd:enumeration value="timeValue"/>
<xsd:enumeration value="time"/>
<xsd:enumeration value="iTime"/>
<xsd:enumeration value="fTime"/>
<xsd:enumeration value="lTime"/>
<xsd:enumeration value="universalTime"/>
<xsd:enumeration value="fieldbusTime"/>
<xsd:enumeration value="visibleString"/>
<xsd:enumeration value="octetString"/>
<xsd:enumeration value="octetString1"/>
<xsd:enumeration value="octetString2"/>
<xsd:enumeration value="octetString4"/>
<xsd:enumeration value="octetString8"/>
<xsd:enumeration value="octetString16"/>
<xsd:enumeration value="binaryTime0"/>
<xsd:enumeration value="binaryTime1"/>
<xsd:enumeration value="binaryTime2"/>
<xsd:enumeration value="binaryTime3"/>
<xsd:enumeration value="binaryTime4"/>
<xsd:enumeration value="binaryTime5"/>
<xsd:enumeration value="binaryTime6"/>
<xsd:enumeration value="binaryTime7"/>
<xsd:enumeration value="binaryTime8"/>
<xsd:enumeration value="binaryTime9"/>
<xsd:enumeration value="compactBooleanArray"/>
<xsd:enumeration value="date_and_time"/>
<xsd:enumeration value="record"/>
<xsd:enumeration value="array"/>
```

```
<xsd:enumeration value="IPV4Address"/>
<xsd:enumeration value="IPV6Address"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
<xsd:attribute name="embeddedStringLength" type="xsd:nonNegativeInteger" use="optional"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="record" nillable="true">
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base="xsd:string">
<xsd:attribute name="elements" type="xsd:nonNegativeInteger" use="required"/>
<xsd:attribute name="maxElements" type="xsd:nonNegativeInteger" use="optional"/>
<xsd:attribute name="octetLength" type="xsd:nonNegativeInteger" use="optional"/>
<xsd:attribute name="format" type="xsd:string" use="optional"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="udt" nillable="true">
<xsd:annotation>
<xsd:documentation>user defined simple data type</xsd:documentation>
</xsd:annotation>
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base="xsd:string">
<xsd:attribute name="name" type="xsd:string" use="required"/>
<xsd:attribute name="bitLength" type="xsd:integer" use="required"/>
<xsd:attribute name="interpretation" use="required">
<xsd:simpleType>
<xsd:restriction base="xsd:string">
<xsd:enumeration value="BIT"/>
<xsd:enumeration value="UNSIGNED"/>
<xsd:enumeration value="INTEGER"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
<xsd:attribute name="IEC61131" type="xsd:string" use="optional"/>
<xsd:attribute name="WinVT" type="xsd:string" use="optional"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:group name="g_booleanDatatypes">
<xsd:annotation>
<xsd:documentation>group of boolean data types</xsd:documentation>
</xsd:annotation>
<xsd:choice>
<xsd:element ref="boolean"/>
<xsd:element ref="bool"/>
<xsd:element ref="byte"/>
<xsd:element ref="word"/>
<xsd:element ref="dword"/>
<xsd:element ref="lword"/>
<xsd:element ref="octetString1"/>
<xsd:element ref="octetString2"/>
<xsd:element ref="octetString4"/>
<xsd:element ref="octetString8"/>
```

```
<xsd:element ref="octetString16"/>
<xsd:element ref="compactBooleanArray"/>
</xsd:choice>
</xsd:group>
<xsd:group name="g_signedIntegerDatatypes">
<xsd:annotation>
<xsd:documentation>group of signed integer data types</xsd:documentation>
</xsd:annotation>
<xsd:choice>
<xsd:element ref="sint"/>
<xsd:element ref="int"/>
<xsd:element ref="dint"/>
<xsd:element ref="lint"/>
</xsd:choice>
</xsd:group>
<xsd:group name="g_unsignedIntegerDatatypes">
<xsd:annotation>
<xsd:documentation>group of unsigned integer data types</xsd:documentation>
</xsd:annotation>
<xsd:choice>
<xsd:element ref="usint"/>
<xsd:element ref="uint"/>
<xsd:element ref="udint"/>
<xsd:element ref="ulint"/>
</xsd:choice>
</xsd:group>
<xsd:group name="g_floatDatatypes">
<xsd:annotation>
<xsd:documentation>group of float data types</xsd:documentation>
</xsd:annotation>
<xsd:choice>
<xsd:element ref="real"/>
<xsd:element ref="lreal"/>
</xsd:choice>
</xsd:group>
<xsd:group name="g_integerDatatypes">
<xsd:annotation>
<xsd:documentation>group of integer data types</xsd:documentation>
</xsd:annotation>
<xsd:choice>
<xsd:group ref="g_signedIntegerDatatypes"/>
<xsd:group ref="g_unsignedIntegerDatatypes"/>
</xsd:choice>
</xsd:group>
<xsd:group name="g_numericDatatypes">
<xsd:annotation>
<xsd:documentation>group of numeric data types</xsd:documentation>
</xsd:annotation>
<xsd:choice>
<xsd:group ref="g_integerDatatypes"/>
<xsd:group ref="g_floatDatatypes"/>
</xsd:choice>
</xsd:group>
<xsd:group name="g_userDatatypes">
<xsd:annotation>
<xsd:documentation>group of user defined simple data types</xsd:documentation>
</xsd:annotation>
<xsd:choice>
<xsd:element ref="udt"/>
</xsd:choice>
</xsd:group>
<xsd:group name="g_dateDatatypes">
```

```
<xsd:annotation>
<xsd:documentation>group of date data types</xsd:documentation>
</xsd:annotation>
<xsd:choice>
<xsd:element ref="binaryDate"/>
<xsd:element ref="binaryDate2000"/>
<xsd:element ref="date7"/>
<xsd:element ref="date"/>
</xsd:choice>
</xsd:group>
<xsd:group name="g_dtDatatypes">
<xsd:annotation>
<xsd:documentation>group of date and time data types</xsd:documentation>
</xsd:annotation>
<xsd:choice>
<xsd:element ref="timeOfDay6"/>
<xsd:element ref="timeDifference6"/>
<xsd:element ref="universalTime"/>
<xsd:element ref="date_and_time"/>
<xsd:element ref="fieldbusTime"/>
</xsd:choice>
</xsd:group>
<xsd:group name="g_timeDatatypes">
<xsd:annotation>
<xsd:documentation>group of time data types</xsd:documentation>
</xsd:annotation>
<xsd:choice>
<xsd:element ref="time_of_day"/>
<xsd:element ref="timeDifference4"/>
<xsd:element ref="time"/>
<xsd:element ref="fTime"/>
<xsd:element ref="lTime"/>
<xsd:element ref="binaryTime0"/>
<xsd:element ref="binaryTime1"/>
<xsd:element ref="binaryTime2"/>
<xsd:element ref="binaryTime3"/>
<xsd:element ref="binaryTime4"/>
<xsd:element ref="binaryTime5"/>
<xsd:element ref="binaryTime6"/>
<xsd:element ref="binaryTime7"/>
<xsd:element ref="binaryTime8"/>
<xsd:element ref="binaryTime9"/>
<xsd:element ref="iTime"/>
</xsd:choice>
</xsd:group>
<xsd:group name="g_stringDatatypes">
<xsd:annotation>
<xsd:documentation>group of string data types</xsd:documentation>
</xsd:annotation>
<xsd:choice>
<xsd:element ref="visibleString"/>
<xsd:element ref="octetString"/>
<xsd:element ref="bitstring"/>
<xsd:element ref="IPV4Address"/>
<xsd:element ref="IPV6Address"/>
</xsd:choice>
</xsd:group>
<xsd:group name="g_complexDatatypes">
<xsd:annotation>
<xsd:documentation>group of complex data types</xsd:documentation>
</xsd:annotation>
<xsd:choice>
```

```
<xsd:element ref="array"/>
<xsd:element ref="record"/>
</xsd:choice>
</xsd:group>
<xsd:group name="g_datatypes">
<xsd:annotation>
<xsd:documentation>group of all FDCML data types</xsd:documentation>
</xsd:annotation>
<xsd:choice>
<xsd:group ref="g_numericDatatypes"/>
<xsd:group ref="g_userDatatypes"/>
<xsd:group ref="g_dateDatatypes"/>
<xsd:group ref="g_stringDatatypes"/>
<xsd:group ref="g_complexDatatypes"/>
<xsd:group ref="g_booleanDatatypes"/>
<xsd:group ref="g_dtDatatypes"/>
<xsd:group ref="g_timeDatatypes"/>
</xsd:choice>
</xsd:group>
</xsd:schema>
```

B.1.5.3 FDCMLISO15745DeviceFunction.xsd

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema targetNamespace="http://gost.e-ecolog.ru                    "
xmlns:xlink="http://www.w3.org/1999/xlink" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns="http://gost.e-ecolog.ru                    elementFormDefault="qualified"
attributeFormDefault="unqualified">
<xsd:import namespace="http://www.w3.org/XML/1998/namespace" schemaLocation="xmldef.xsd"/>
<xsd:import namespace="http://gost.e-ecolog.ru                    schemaLocation="xlinkdef.xsd"/>
<xsd:complexType name="parameterType">
<xsd:sequence>
<xsd:element ref="defaultValue" minOccurs="0"/>
<xsd:element ref="lowLimit" minOccurs="0"/>
<xsd:element ref="highLimit" minOccurs="0"/>
<xsd:element ref="unit" minOccurs="0"/>
<xsd:element ref="listOfValues" minOccurs="0"/>
<xsd:element ref="membersList" minOccurs="0"/>
<xsd:element ref="actualValue" minOccurs="0"/>
<xsd:sequence minOccurs="0">
<xsd:group ref="g_naming"/>
</xsd:sequence>
</xsd:sequence>
<xsd:attribute name="name" type="xsd:string" use="required"/>
<xsd:attribute name="denotation" type="xsd:string"/>
<xsd:attribute name="description" type="xsd:string"/>
<xsd:attribute name="paramID" type="xsd:ID" use="required"/>
<xsd:attributeGroup ref="accessCategory"/>
<xsd:attribute name="constraintIDREF" type="xsd:IDREF"/>
<xsd:attributeGroup ref="dataType"/>
<xsd:attributeGroup ref="persistent"/>
<xsd:attributeGroup ref="use"/>
</xsd:complexType>
<xsd:complexType name="stateTransitionDiagramType">
<xsd:sequence>
<xsd:element ref="statesTab"/>
<xsd:element ref="transitionsTab"/>
</xsd:sequence>
</xsd:complexType>
<xsd:complexType name="stateType">
<xsd:sequence>
<xsd:element ref="associatedParameter"/>
</xsd:sequence>
```

```
<xsd:attribute name="description" type="xsd:string"/>
<xsd:attribute name="initialState" use="required">
<xsd:simpleType>
<xsd:restriction base="xsd:NMTOKEN">
<xsd:enumeration value="false"/>
<xsd:enumeration value="true"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
<xsd:attribute name="stateName" type="xsd:ID" use="required"/>
<xsd:attribute name="stateParent" type="xsd:string"/>
</xsd:complexType>
<xsd:complexType name="transitionType">
<xsd:sequence>
<xsd:element ref="commandParameter" maxOccurs="unbounded"/>
</xsd:sequence>
<xsd:attribute name="description" type="xsd:string"/>
<xsd:attribute name="transitionNumber" use="required">
<xsd:annotation>
<xsd:documentation>Number format = Tx to Txxxxxx</xsd:documentation>
</xsd:annotation>
<xsd:simpleType>
<xsd:restriction base="xsd:ID">
<xsd:pattern value="T\d{1,6}"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
<xsd:attribute name="fromState" type="xsd:IDREF" use="required"/>
<xsd:attribute name="toState" type="xsd:IDREF" use="required"/>
</xsd:complexType>
<xsd:element name="actualValue" type="xsd:string"/>
<xsd:element name="associatedParameter">
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base="xsd:string">
<xsd:attribute name="associatedParameterValue" type="xsd:string" use="required"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="commandParameter">
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base="xsd:string">
<xsd:attribute name="commandParameterValue" type="xsd:string" use="required"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="configParametersList">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="parameterRef" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="constraint">
<xsd:annotation>
<xsd:documentation>The elements under Constraints aren't checked by the parser. The languages,
to use to describe the constraints, don't refer to a schema.
</xsd:documentation>
</xsd:annotation>
```

```
<xsd:complexType>
<xsd:sequence>
<xsd:any namespace="##any" processContents="skip" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
<xsd:attribute name="constraintID" use="required">
<xsd:simpleType>
<xsd:restriction base="xsd:ID">
<xsd:pattern value="C\d{1,6}"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
<xsd:attribute name="constraintType" use="required">
<xsd:simpleType>
<xsd:restriction base="xsd:NMTOKEN">
<xsd:enumeration value="among_parameters"/>
<xsd:enumeration value="display"/>
<xsd:enumeration value="network"/>
<xsd:enumeration value="device"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
<xsd:attribute name="constraintLanguage" type="xsd:string" use="required"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="constraintsList">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="constraint" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="defaultValue">
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base="xsd:string">
<xsd:attributeGroup ref="type"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="function">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="inputsList" minOccurs="0"/>
<xsd:element ref="outputsList" minOccurs="0"/>
<xsd:element ref="configParametersList" minOccurs="0"/>
<xsd:element ref="constraintsList" minOccurs="0"/>
<xsd:element name="functionStateTransitionDiagram" minOccurs="0">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="stateTransitionDiagram"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
<xsd:attribute name="name" type="xsd:string" use="required"/>
<xsd:attribute name="description" type="xsd:string"/>
<xsd:attribute name="functionID" type="xsd:ID" use="required"/>
<xsd:attribute name="version" type="xsd:string" use="required"/>
</xsd:complexType>
</xsd:element>
```

```
<xsd:element name="functionRef">
<xsd:annotation>
<xsd:documentation>By this way the hierarchical view of the device can be described. There is no
limitation in the level number.</xsd:documentation>
</xsd:annotation>
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="functionRef" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
<xsd:attribute name="functionName" type="xsd:string" use="required"/>
<xsd:attribute name="functionIDREF" type="xsd:string" use="required"/>
<xsd:attribute name="functionURL" type="xsd:string"/>
<xsd:attribute name="functionDescription" type="xsd:string"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="functionView">
<xsd:annotation>
<xsd:documentation>Network independent description of the intrinsic function of a device, from
a configuration and service (i.e.: tuning, maintenance, diagnostic,Â¶) perspective</xsd:documentation>
</xsd:annotation>
<xsd:complexType>
<xsd:sequence>
<xsd:element name="functionList" minOccurs="0">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="function" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="parameterList">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="parameter" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="functionAnalysis" minOccurs="0">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="functionRef" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="help">
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base="xsd:string">
<xsd:attribute ref="xml:lang" use="required"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="helpFileRef">
<xsd:complexType>
<xsd:attribute name="helpFileID" type="xsd:string"/>
<xsd:attribute name="helpID" type="xsd:string"/>
</xsd:complexType>
</xsd:element>
```

```
<xsd:element name="helpRef">
<xsd:complexType>
<xsd:attribute name="dictID" type="xsd:string"/>
<xsd:attribute name="textID" type="xsd:string" use="required"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="highLimit">
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base="xsd:string">
<xsd:attributeGroup ref="type"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="inputsList">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="parameterRef" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="label">
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base="xsd:string">
<xsd:attribute ref="xml:lang" use="required"/>
<xsd:attribute ref="xlink:type"/>
<xsd:attribute ref="xlink:href"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="labelRef">
<xsd:complexType>
<xsd:attribute name="dictID" type="xsd:string"/>
<xsd:attribute name="textID" type="xsd:string" use="required"/>
<xsd:attribute ref="xlink:type"/>
<xsd:attribute ref="xlink:href"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="listOfValues">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="value" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="lowLimit">
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base="xsd:string">
<xsd:attributeGroup ref="type"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="member">
<xsd:annotation>
<xsd:documentation>For dataType = array, only first member is needed</xsd:documentation>
</xsd:annotation>
<xsd:complexType>
```

```
<xsd:sequence>
<xsd:element ref="defaultValue" minOccurs="0"/>
<xsd:element ref="lowLimit" minOccurs="0"/>
<xsd:element ref="highLimit" minOccurs="0"/>
<xsd:element ref="unit" minOccurs="0"/>
<xsd:element ref="listOfValues" minOccurs="0"/>
<xsd:element ref="membersList" minOccurs="0"/>
<xsd:sequence minOccurs="0">
<xsd:group ref="g_naming"/>
</xsd:sequence>
<xsd:element ref="actualValue" minOccurs="0"/>
</xsd:sequence>
<xsd:attribute name="name" type="xsd:string" use="required"/>
<xsd:attribute name="description" type="xsd:string"/>
<xsd:attribute name="memberID" type="xsd:string" use="required"/>
<xsd:attributeGroup ref="dataType"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="membersList">
<xsd:annotation>
<xsd:documentation>membersList is mandatory if dataType = struct or array.</xsd:documentation>
</xsd:annotation>
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="member" maxOccurs="unbounded"/>
</xsd:sequence>
<xsd:attribute name="numberOfMembers" type="xsd:integer" use="required"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="outputsList">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="parameterRef" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="parameterRef">
<xsd:complexType>
<xsd:attribute name="paramID" type="xsd:IDREF" use="required"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="parameter" type="parameterType"/>
<xsd:element name="state" type="stateType"/>
<xsd:element name="statesTab">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="state" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="stateTransitionDiagram" type="stateTransitionDiagramType"/>
<xsd:element name="transition" type="transitionType"/>
<xsd:element name="transitionsTab">
<xsd:complexType>
<xsd:sequence>
<xsd:element ref="transition" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="unit">
<xsd:complexType>
```

```
<xsd:attribute name="multiplier" type="xsd:string" use="required"/>
<xsd:attribute name="unitURI" type="xsd:anyURI"/>
</xsd:complexType>
</xsd:element>
<xsd:element name="value">
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base="xsd:string">
<xsd:attribute name="meaning" type="xsd:string"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:attributeGroup name="accessCategory">
<xsd:attribute name="accessCategory" use="required">
<xsd:simpleType>
<xsd:restriction base="xsd:NMTOKEN">
<xsd:enumeration value="read"/>
<xsd:enumeration value="write"/>
<xsd:enumeration value="readWrite"/>
<xsd:enumeration value="noAccess"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:attributeGroup>
<xsd:attributeGroup name="dataType">
<xsd:attribute name="dataType" use="required">
<xsd:simpleType>
<xsd:restriction base="xsd:NMTOKEN">
<xsd:enumeration value="bool"/>
<xsd:enumeration value="byte"/>
<xsd:enumeration value="char"/>
<xsd:enumeration value="word"/>
<xsd:enumeration value="dWord"/>
<xsd:enumeration value="lWord"/>
<xsd:enumeration value="sInt"/>
<xsd:enumeration value="usInt"/>
<xsd:enumeration value="int"/>
<xsd:enumeration value="uInt"/>
<xsd:enumeration value="dInt"/>
<xsd:enumeration value="udInt"/>
<xsd:enumeration value="lInt"/>
<xsd:enumeration value="ulInt"/>
<xsd:enumeration value="real"/>
<xsd:enumeration value="lReal"/>
<xsd:enumeration value="string"/>
<xsd:enumeration value="unicode"/>
<xsd:enumeration value="struct"/>
<xsd:enumeration value="physical"/>
<xsd:enumeration value="array"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:attributeGroup>
<xsd:attributeGroup name="persistent">
<xsd:attribute name="persistent" use="required">
<xsd:simpleType>
<xsd:restriction base="xsd:NMTOKEN">
<xsd:enumeration value="false"/>
<xsd:enumeration value="true"/>
<xsd:enumeration value="n.a"/>
</xsd:restriction>
```

```
</xsd:simpleType>
</xsd:attribute>
</xsd:attributeGroup>
<xsd:attributeGroup name="type">
<xsd:attribute name="type" use="required">
<xsd:simpleType>
<xsd:restriction base="xsd:NMTOKEN">
<xsd:enumeration value="fixedValue"/>
<xsd:enumeration value="vendorSpecific"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:attributeGroup>
<xsd:attributeGroup name="use">
<xsd:attribute name="use" use="required">
<xsd:simpleType>
<xsd:restriction base="xsd:NMTOKEN">
<xsd:enumeration value="required"/>
<xsd:enumeration value="optional"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:attributeGroup>
<xsd:group name="g_naming">
<xsd:sequence>
<xsd:choice>
<xsd:sequence>
<xsd:element ref="label" maxOccurs="unbounded"/>
<xsd:element ref="help" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
<xsd:sequence>
<xsd:element ref="labelRef"/>
<xsd:element ref="helpRef" minOccurs="0"/>
</xsd:sequence>
</xsd:choice>
<xsd:element ref="helpFileRef" minOccurs="0"/>
</xsd:sequence>
</xsd:group>
</xsd:schema>
```

B.1.5.4 FDCMLTextResource.xsd

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema targetNamespace="gost.e-ecolog.ru                    " xmlns="gost.e-ecolog.ru                    "
xmlns:xsd="gost.e-ecolog.ru                    elementFormDefault="qualified"
attributeFormDefault="unqualified" version="1.0">
<xsd:import namespace="http://www.w3.org/XML/1998/namespace" schemaLocation="xmldef.xsd"/>
<xsd:element name="FDCMLTextResource">
<xsd:annotation>
<xsd:documentation>text resource for FDCML</xsd:documentation>
</xsd:annotation>
<xsd:complexType>
<xsd:sequence>
<xsd:element name="textEntry" type="t_textEntry" maxOccurs="unbounded"/>
</xsd:sequence>
<xsd:attribute ref="xml:lang" use="required"/>
</xsd:complexType>
</xsd:element>
<xsd:complexType name="t_textEntry">
<xsd:annotation>
<xsd:documentation>single text</xsd:documentation>
</xsd:annotation>
<xsd:simpleContent>
```

```
<xsd:extension base="xsd:string">
<xsd:attribute name="textID" type="xsd:ID" use="required"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:schema>
```

B.1.5.5 xmldef.xsd

```
<?xml version="1.0"?>
<xsd:schema targetNamespace="gost.e-ecolog.ru
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
elementFormDefault="unqualified"
attributeFormDefault="unqualified"
version="1.0">
<xsd:attribute name="lang" type="xsd:language"/>
<xsd:attribute name="space" default="preserve">
<xsd:simpleType>
<xsd:restriction base="xsd:NCName">
<xsd:enumeration value="default"/>
<xsd:enumeration value="preserve"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:schema>
```

B.1.5.6 xlinkdef.xsd

```
<?xml version="1.0"?>
<xsd:schema targetNamespace="http://www.w3.org/1999/xlink"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xlink="gost.e-ecolog.ru                    "
elementFormDefault="unqualified"
attributeFormDefault="unqualified"
version="1.0">
<xsd:attribute name="type">
<xsd:simpleType>
<xsd:restriction base="xsd:string">
<xsd:enumeration value="simple"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
<xsd:attribute name="href" type="xsd:anyURI"/>
</xsd:schema>
```

B.2 Описание шаблонов профилей коммуникационных сетей

XML файлы коммуникационных профилей должны соответствовать XML схеме профилей коммуникационных сетей, как установлено ниже.

```
<?xml version = "1.0" encoding = "UTF-16"?>
<xsd:schema xmlns = "http://www.can-cia.org/xml/canopen/commnetworkprofile/"
targetNamespace = "gost.e-ecolog.ru                                                    /"
xmlns:xlink = "gost.e-ecolog.ru                    "
xmlns:xsd = "gost.e-ecolog.ru
version = "1.0"
elementFormDefault = "qualified"
attributeFormDefault = "unqualified">
<xsd:import namespace = "http://gost.e-ecolog.ru                    schemaLocation = "xlinkdef.xsd"/>
<xsd:attributeGroup name = "use">
<xsd:attribute name = "use" use = "required">
<xsd:simpleType>
<xsd:restriction base = "xsd:NMTOKEN">
<xsd:enumeration value = "required"/>
<xsd:enumeration value = "optional"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
```

```
</xsd:attributeGroup>
<xsd:attributeGroup name = "type">
<xsd:attribute name = "type" use = "required">
<xsd:simpleType>
<xsd:restriction base = "xsd:NMTOKEN">
<xsd:enumeration value = "fixedValue"/>
<xsd:enumeration value = "vendorSpecific"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:attributeGroup>
<xsd:attributeGroup name = "persistent">
<xsd:attribute name = "persistent" use = "required">
<xsd:simpleType>
<xsd:restriction base = "xsd:NMTOKEN">
<xsd:enumeration value = "false"/>
<xsd:enumeration value = "true"/>
<xsd:enumeration value = "n.a"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:attributeGroup>
<xsd:attributeGroup name = "dataType">
<xsd:attribute name = "dataType" use = "required">
<xsd:simpleType>
<xsd:restriction base = "xsd:NMTOKEN">
<xsd:enumeration value = "bool"/>
<xsd:enumeration value = "byte"/>
<xsd:enumeration value = "char"/>
<xsd:enumeration value = "word"/>
<xsd:enumeration value = "dWord"/>
<xsd:enumeration value = "lWord"/>
<xsd:enumeration value = "sInt"/>
<xsd:enumeration value = "usInt"/>
<xsd:enumeration value = "int"/>
<xsd:enumeration value = "uInt"/>
<xsd:enumeration value = "dInt"/>
<xsd:enumeration value = "udInt"/>
<xsd:enumeration value = "lInt"/>
<xsd:enumeration value = "ulInt"/>
<xsd:enumeration value = "real"/>
<xsd:enumeration value = "lReal"/>
<xsd:enumeration value = "string"/>
<xsd:enumeration value = "unicode"/>
<xsd:enumeration value = "struct"/>
<xsd:enumeration value = "physical"/>
<xsd:enumeration value = "array"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:attributeGroup>
<xsd:attributeGroup name = "accessCategory">
<xsd:attribute name = "accessCategory" use = "required">
<xsd:simpleType>
<xsd:restriction base = "xsd:NMTOKEN">
<xsd:enumeration value = "read"/>
<xsd:enumeration value = "write"/>
<xsd:enumeration value = "readWrite"/>
<xsd:enumeration value = "noAccess"/>
</xsd:restriction>
</xsd:simpleType>
```

```
</xsd:attribute>
</xsd:attributeGroup>
<xsd:attributeGroup name = "ag_FDCML">
<xsd:attribute name = "formatName" fixed = "FDCML" form = "unqualified" type = "xsd:string"/>
<xsd:attribute name = "formatVersion" fixed = "2.0" form = "unqualified" type = "xsd:string"/>
<xsd:attribute name = "fileName" use = "required" form = "unqualified" type = "xsd:string"/>
<xsd:attribute name = "fileCreator" use = "required" form ="unqualified" type = "xsd:string"/>
<xsd:attribute name = "fileCreationDate" use = "required" form = "unqualified" type = "xsd:date"/>
<xsd:attribute name = "fileModificationDate" use = "required" form = "unqualified" type =
"xsd:date"/>
<xsd:attribute name = "fileVersion" use = "required" form = "unqualified" type = "xsd:string"/>
</xsd:attributeGroup>
<xsd:group name = "g_naming">
<xsd:sequence>
<xsd:choice>
<xsd:sequence>
<xsd:element ref = "label" maxOccurs = "unbounded"/>
<xsd:element ref = "help" minOccurs = "0" maxOccurs = "unbounded"/>
</xsd:sequence>
<xsd:sequence>
<xsd:element ref = "labelRef"/>
<xsd:element ref = "helpRef" minOccurs = "0"/>
</xsd:sequence>
</xsd:choice>
<xsd:element ref = "helpFileRef" minOccurs = "0"/>
</xsd:sequence>
</xsd:group>
<!- -xmlns:xml="gost.e-ecolog.ru                                                    ->
<!- -
################ global elements
- ->
<xsd:element name = "ISO15745Profile">
<xsd:annotation>
<xsd:documentation>Document Element</xsd:documentation>
</xsd:annotation>
<xsd:complexType>
<xsd:sequence>
<xsd:element ref = "ProfileHeader"/>
<xsd:element ref = "ProfileBody"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<!- -
################ types
- ->
<xsd:element name = "help" type = "xsd:string"/>
<xsd:element name = "helpFileRef">
<xsd:complexType>
<xsd:attribute name = "helpFileID" type = "xsd:string"/>
<xsd:attribute name = "helpID" type = "xsd:string"/>
</xsd:complexType>
</xsd:element>
<xsd:element name = "helpRef">
<xsd:complexType>
<xsd:attribute name = "dictID" type = "xsd:string"/>
<xsd:attribute name = "textID" use = "required" type = "xsd:string"/>
</xsd:complexType>
</xsd:element>
<xsd:simpleType name = "IASInterface_DataType">
<xsd:union memberTypes = " ">
<xsd:simpleType>
```

```
<xsd:restriction base = "xsd:string">
<xsd:enumeration value = "CSI"/>
<xsd:enumeration value = "HCI"/>
<xsd:enumeration value = "ISI"/>
<xsd:enumeration value = "API"/>
<xsd:enumeration value = "CMI"/>
<xsd:enumeration value = "ESI"/>
<xsd:enumeration value = "FSI"/>
<xsd:enumeration value = "MTI"/>
<xsd:enumeration value = "SEI"/>
<xsd:enumeration value = "USI"/>
</xsd:restriction>
</xsd:simpleType>
<xsd:simpleType>
<xsd:restriction base = "xsd:string">
<xsd:length value = "4"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:union>
</xsd:simpleType>
<xsd: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:element name = "ISO15745Edition" type = "xsd:positiveInteger"/>
<xsd:element name = "ISO15745Part" type = "xsd:positiveInteger"/>
<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:element name = "label">
<xsd:complexType>
<xsd:simpleContent>
<xsd:extension base = "xsd:string">
<xsd:attribute ref = "xlink:type"/>
<xsd:attribute ref = "xlink:href"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:element name = "labelRef">
<xsd:complexType>
<xsd:attribute name = "dictID" type = "xsd:string"/>
<xsd:attribute name = "textID" use = "required" type = "xsd:string"/>
<xsd:attribute ref = "xlink:type"/>
<xsd:attribute ref = "xlink:href"/>
</xsd:complexType>
</xsd:element>
<xsd:element name = "ProfileBody">
```

```
<xsd:complexType>
<xsd:sequence>
<xsd:element ref = "TransportLayers"/>
<xsd:element ref = "ApplicationLayers"/>
<xsd:element ref = "NetworkManagement" minOccurs = "0"/>
</xsd:sequence>
<xsd:attributeGroup ref = "ag_FDCML"/>
<xsd:attribute name = "supportedLanguages" type = "xsd:NMTOKENS"/>
<xsd:attribute name = "uniqueID" type = "xsd:ID"/>
</xsd:complexType>
</xsd:element>
<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"/>
<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>
<!--
################ groups
-->
<!--
################ attribute groups
-->
<xsd:element name = "communicationProfile" type = "xsd:string"/>
<xsd:element name = "TransportLayers">
<xsd:complexType>
<xsd:sequence>
<xsd:element name = "cfgItemList">
<xsd:complexType>
<xsd:sequence>
<xsd:sequence minOccurs = "0">
<xsd:group ref = "g_naming"/>
</xsd:sequence>
<xsd:element name = "CANopenDedicatedCfgCategory">
<xsd:complexType>
<xsd:sequence>
<xsd:element name = "CANopenBaudRate">
<xsd:complexType>
<xsd:sequence>
<xsd:element name = "BaudRate10">
<xsd:complexType>
<xsd:attribute name = "Supported" use = "required">
<xsd:simpleType>
<xsd:restriction base = "xsd:NMTOKENS">
<xsd:enumeration value = "No"/>
<xsd:enumeration value = "Yes"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
```

```
<xsd:element name = "BaudRate20">
<xsd:complexType>
<xsd:attribute name = "Supported" use = "required">
<xsd:simpleType>
<xsd:restriction base = "xsd:NMTOKENS">
<xsd:enumeration value = "No"/>
<xsd:enumeration value = "Yes"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
<xsd:element name = "BaudRate50">
<xsd:complexType>
<xsd:attribute name = "Supported" use = "required">
<xsd:simpleType>
<xsd:restriction base = "xsd:NMTOKENS">
<xsd:enumeration value = "No"/>
<xsd:enumeration value = "Yes"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
<xsd:element name = "BaudRate125">
<xsd:complexType>
<xsd:attribute name = "Supported" use = "required">
<xsd:simpleType>
<xsd:restriction base = "xsd:NMTOKENS">
<xsd:enumeration value = "No"/>
<xsd:enumeration value = "Yes"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
<xsd:element name = "BaudRate250">
<xsd:complexType>
<xsd:attribute name = "Supported" use = "required">
<xsd:simpleType>
<xsd:restriction base = "xsd:NMTOKENS">
<xsd:enumeration value = "No"/>
<xsd:enumeration value = "Yes"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
<xsd:element name = "BaudRate500">
<xsd:complexType>
<xsd:attribute name = "Supported" use = "required">
<xsd:simpleType>
<xsd:restriction base = "xsd:NMTOKENS">
<xsd:enumeration value = "No"/>
<xsd:enumeration value = "Yes"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
```

```
<xsd:element name = "BaudRate800">
<xsd:complexType>
<xsd:attribute name = "supported" use = "required">
<xsd:simpleType>
<xsd:restriction base = "xsd:NMTOKENS">
<xsd:enumeration value = "No"/>
<xsd:enumeration value = "Yes"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
<xsd:element name = "BaudRate1000">
<xsd:complexType>
<xsd:attribute name = "Supported" use = "required">
<xsd:simpleType>
<xsd:restriction base = "xsd:NMTOKENS">
<xsd:enumeration value = "No"/>
<xsd:enumeration value = "Yes"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name = "CANopenGeneralCapabilities">
<xsd:complexType>
<xsd:sequence>
<xsd:element name = "GroupMessaging">
<xsd:complexType>
<xsd:attribute name = "supported" use = "required">
<xsd:simpleType>
<xsd:restriction base = "xsd:NMTOKENS">
<xsd:enumeration value = "No"/>
<xsd:enumeration value = "Yes"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
<xsd:element name = "DynamicChannels">
<xsd:complexType>
<xsd:attribute name = "supported" use = "required">
<xsd:simpleType>
<xsd:restriction base = "xsd:NMTOKENS">
<xsd:enumeration value = "No"/>
<xsd:enumeration value = "Yes"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
<xsd:element name = "SDORequestingDevice">
<xsd:complexType>
<xsd:attribute name = "supported" use = "required">
<xsd:simpleType>
<xsd:restriction base = "xsd:NMTOKENS">
<xsd:enumeration value = "No"/>
<xsd:enumeration value = "Yes"/>
```

```
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name = "CANopenDeviceCommissioning" minOccurs = "0">
<xsd:complexType>
<xsd:sequence>
<xsd:element name = "BaudRate">
<xsd:simpleType>
<xsd:restriction base = "xsd:NMTOKENS">
<xsd:enumeration value = "BaudRate10"/>
<xsd:enumeration value = "BaudRate20"/>
<xsd:enumeration value = "BaudRate50"/>
<xsd:enumeration value = "BaudRate125"/>
<xsd:enumeration value = "BaudRate250"/>
<xsd:enumeration value = "BaudRate500"/>
<xsd:enumeration value = "BaudRate800"/>
<xsd:enumeration value = "BaudRate1000"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name = "ApplicationLayers">
<xsd:complexType>
<xsd:sequence>
<xsd:element name = "cfgItemList">
<xsd:complexType>
<xsd:sequence>
<xsd:sequence minOccurs = "0">
<xsd:group ref = "g_naming"/>
</xsd:sequence>
<xsd:element name = "CANopenDedicatedCfgCategory">
<xsd:complexType>
<xsd:sequence>
<xsd:element name = "CANopenDummyMappingDataTypes">
<xsd:complexType>
<xsd:sequence>
<xsd:element name = "dmBoolean">
<xsd:complexType>
<xsd:attribute name = "supported" use = "required">
<xsd:simpleType>
<xsd:restriction base = "xsd:NMTOKENS">
<xsd:enumeration value = "No"/>
<xsd:enumeration value = "Yes"/>
</xsd:restriction>
</xsd:simpleType>
```

```
</xsd:attribute>
</xsd:complexType>
</xsd:element>
<xsd:element name = "dmInteger8">
<xsd:complexType>
<xsd:attribute name = "supported" use = "required">
<xsd:simpleType>
<xsd:restriction base = "xsd:NMTOKENS">
<xsd:enumeration value = "No"/>
<xsd:enumeration value = "Yes"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
<xsd:element name = "dmInteger16">
<xsd:complexType>
<xsd:attribute name = "supported" use = "required">
<xsd:simpleType>
<xsd:restriction base = "xsd:NMTOKENS">
<xsd:enumeration value = "No"/>
<xsd:enumeration value = "Yes"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
<xsd:element name = "dmInteger32">
<xsd:complexType>
<xsd:attribute name = "supported" use = "required">
<xsd:simpleType>
<xsd:restriction base = "xsd:NMTOKENS">
<xsd:enumeration value = "No"/>
<xsd:enumeration value = "Yes"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
<xsd:element name = "dmUnsigned8">
<xsd:complexType>
<xsd:attribute name = "supported" use = "required">
<xsd:simpleType>
<xsd:restriction base = "xsd:NMTOKENS">
<xsd:enumeration value = "No"/>
<xsd:enumeration value = "Yes"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
<xsd:element name = "dmUnsigned16">
<xsd:complexType>
<xsd:attribute name = "supported" use = "required">
<xsd:simpleType>
<xsd:restriction base = "xsd:NMTOKENS">
<xsd:enumeration value = "No"/>
<xsd:enumeration value = "Yes"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
```

```
</xsd:complexType>
</xsd:element>
<xsd:element name = "dmUnsigned32">
<xsd:complexType>
<xsd:attribute name = "supported" use = "required">
<xsd:simpleType>
<xsd:restriction base = "xsd:NMTOKENS">
<xsd:enumeration value = "No"/>
<xsd:enumeration value = "Yes"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name = "CANopenGeneralCapabilities">
<xsd:complexType>
<xsd:sequence>
<xsd:element name = "Granularity">
<xsd:complexType>
<xsd:attribute name = "value" type = "xsd:unsignedShort"/>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name = "CANopenDeviceCommissioning" minOccurs = "0">
<xsd:complexType>
<xsd:sequence/>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name = "NetworkManagement">
<xsd:complexType>
<xsd:sequence>
<xsd:element name = "cfgItemList">
<xsd:complexType>
<xsd:sequence>
<xsd:sequence minOccurs = "0">
<xsd:group ref = "g_naming"/>
</xsd:sequence>
<xsd:element name = "CANopenDedicatedCfgCategory">
<xsd:complexType>
<xsd:sequence>
<xsd:element name = "CANopenGeneralCapabilities">
<xsd:complexType>
<xsd:sequence>
<xsd:element name = "LayerSettingServiceSlave">
<xsd:complexType>
<xsd:attribute name = "supported" use = "required">
<xsd:simpleType>
```

```
<xsd:restriction base = "xsd:NMTOKENS">
<xsd:enumeration value = "No"/>
<xsd:enumeration value = "Yes"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
<xsd:element name = "SelfStartingDevice">
<xsd:complexType>
<xsd:attribute name = "supported" use = "required">
<xsd:simpleType>
<xsd:restriction base = "xsd:NMTOKENS">
<xsd:enumeration value = "No"/>
<xsd:enumeration value = "Yes"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name = "CANopenManagerCapabilities" minOccurs = "0">
<xsd:complexType>
<xsd:sequence>
<xsd:element name = "MasterSwitchable">
<xsd:complexType>
<xsd:attribute name = "supported" use = "required">
<xsd:simpleType>
<xsd:restriction base = "xsd:NMTOKENS">
<xsd:enumeration value = "No"/>
<xsd:enumeration value = "Yes"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
<xsd:element name = "FlyingMaster">
<xsd:complexType>
<xsd:attribute name = "supported" use = "required">
<xsd:simpleType>
<xsd:restriction base = "xsd:NMTOKENS">
<xsd:enumeration value = "No"/>
<xsd:enumeration value = "Yes"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
<xsd:element name = "SDOManager">
<xsd:complexType>
<xsd:attribute name = "supported" use = "required">
<xsd:simpleType>
<xsd:restriction base = "xsd:NMTOKENS">
<xsd:enumeration value = "No"/>
<xsd:enumeration value = "Yes"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:complexType>
```

```
</xsd:element>
<xsd:element name = "ConfigurationManager">
<xsd:complexType>
<xsd:attribute name = "supported" use = "required">
<xsd:simpleType>
<xsd:restriction base = "xsd:NMTOKENS">
<xsd:enumeration value = "No"/>
<xsd:enumeration value = "Yes"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
<xsd:element name = "LayerSettingServiceMaster">
<xsd:complexType>
<xsd:attribute name = "suppoerted" use = "required">
<xsd:simpleType>
<xsd:restriction base = "xsd:NMTOKENS">
<xsd:enumeration value = "No"/>
<xsd:enumeration value = "Yes"/>
</xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name = "CANopenDeviceCommissioning" minOccurs = "0">
<xsd:complexType>
<xsd:sequence>
<xsd:element name = "NodeId" type = "xsd:unsignedShort"/>
<xsd:element name = "NodeName" type = "sd:string" minOccurs =
"0"/>
<xsd:element name = "NetNumber" type = "xsd:unsignedLong"
minOccurs = "0"/>
<xsd:element name = "NetworkName" type = "xsd:string" minOccurs =
"0"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element ref = "communicationProfile" minOccurs = "0" maxOccurs = "unbounded"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
</xsd:schema>
```

Приложение ДА
(справочное)

Сведения о соответствии ссылочных международных и европейского регионального стандартов ссылочным национальным стандартам Российской Федерации

Таблица ДА.1

Обозначение ссылочного международного, европейского регионального стандарта	Степень соответствия	Обозначение и наименование соответствующего национального стандарта
ИСО 639-1:2002		*
ИСО 639-2:1998		*
ИСО 3166-1:1997		*
ИСО/МЭК 10646-1:2000		*
ИСО 11898:1993		*
ИСО 15745-1:2003		*
МЭК 61158 (все части)		*
МЭК 61784-1:2003		*
МЭК 62026-3:2000		*
ЕН 50325-4:2002		*

* Соответствующий национальный стандарт отсутствует. До его утверждения рекомендуется использовать перевод на русский язык данного международного европейского регионального стандарта. Перевод данного международного европейского регионального стандарта находится в Федеральном информационном фонде технических регламентов и стандартов.

Библиография

[1] МЭК 61131-3:2003 Контроллеры программируемые. Часть 3. Языки программирования

[2] МЭК/PAS 61499 (все части) Блоки функциональные для систем измерения и управления в производственных процессах

[3] МЭК ТС 61915:2003 Комплектные распределительные устройства низковольтные. Принципы разработки приборных профилей для сетевых промышленных устройств

[4] ЕН 50325-1:2000 Подсистема промышленной коммуникации на основе ИСО 11898 (CAN) для интерфейсов с управляемым устройством. Часть 1. Общие требования

[5] ЕН 50325-2:2000 Подсистема промышленной коммуникации на основе ИСО 11898 (CAN) для интерфейсов с управляемым устройством. Часть 2. DeviceNet

[6] IEEE OUI and Company_id Assignments [см. http://standards.ieee.org/regauth/oui/index.shtml]

[7] XML Linking Language (XLink) Version 1.0, W3C Recommendation, 27 June 2001

[8] XML Path Language (XPath) Version 1.0, W3C Recommendation, 16 November 1999

УДК 658.52.011.56 ОКС 25.040.40 Т58

Ключевые слова: автоматизированные промышленные системы, интеграция, жизненный цикл систем, управление производством

Редактор *А. Д. Чайка*
Технический редактор *В. Н. Прусакова*
Корректор *Л. Я. Митрофанова*
Компьютерная верстка *В. Н. Романовой*

Сдано в набор 18.09.2013. Подписано в печать 21.11.2013. Формат 60×84 1/8. Бумага офсетная. Гарнитура Ариал. Печать офсетная. Усл. печ. л. 20,00. Уч.-изд. л. 18.85. Тираж 58 экз. Зак. 1358

ФГУП «СТАНДАРТИНФОРМ», 123995 Москва, Гранатный пер., 4.
o@gostinfo.ru
Набрано и отпечатано в Калужской типографии стандартов, 248021 Калуга, ул. Московская, 256.

gost.e-ecolog.ru