

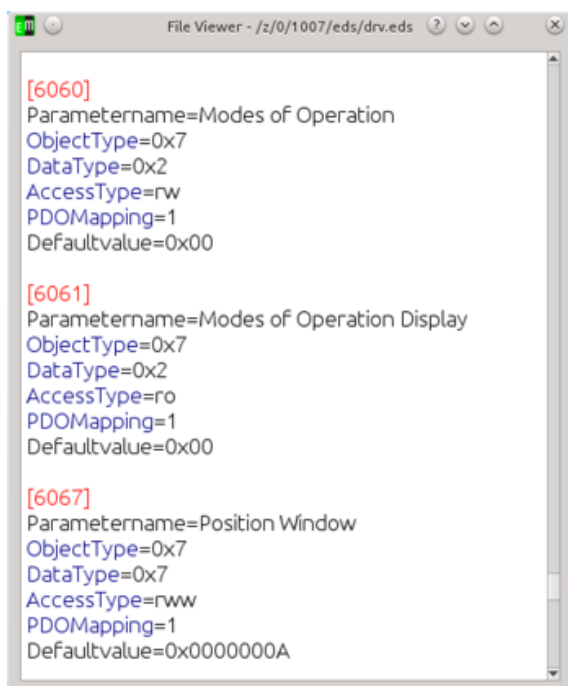
Use cases and advantages of the XML device description format for CANopen FD devices

Torsten Gedenk, emtas GmbH

The new XML device description (CiA 311) is mandatory for CANopen FD devices. It substitutes the existing electronic data sheets as defined in CiA 306. The new format provides much more possibilities to describe CANopen FD devices in detail – both the application and the communication parts.

This paper gives an overview of the XML device descriptions (XDD) and compares it with the previous EDS format. It focuses on use cases and advantages for CANopen FD device manufactures, system integrators and tool manufacturers. Additionally, migration paths from existing EDS files to modern XML device description files are discussed.

A standardized, machine-readable device description format is required for each industrial communication protocol – this has been realized many years ago and so the “Electronic Data Sheets” (EDS files) have been specified as a device description for CANopen devices two decades ago. The EDS files are currently used for the exchange of data between different CANopen tools for commissioning, network planing and device development and PLCs. Nevertheless the existing EDS format as shown below has some distinct disadvantages.



```

[6060]
Parametername=Modes of Operation
ObjectType=0x7
DataType=0x2
AccessType=rw
PDOMapping=1
Defaultvalue=0x00

[6061]
Parametername=Modes of Operation Display
ObjectType=0x7
DataType=0x2
AccessType=ro
PDOMapping=1
Defaultvalue=0x00

[6067]
Parametername=Position Window
ObjectType=0x7
DataType=0x7
AccessType=rww
PDOMapping=1
Defaultvalue=0x0000000A

```

Figure 1: objects in existing EDS files

The Windows INI format is used for the EDS files and it only allows a structure with sections and key-value-pairs but not any additional structuring or grouping of the parameters. As an example for all parameters of a device, the parameter „Modes of operation“ of a drive according to CiA-402 shall be examined in detail. According to the drive profile (CiA-402) this parameter is addressed via object 0x6060 and its description using the old EDS specification (CiA-306 V1.3) is illustrated in figure 1. Although all important attributes can be found in the EDS file, one particular property of this parameter is missing: a parameter containing 255 different values. It is not possible to describe the meaning of all 255 different values of this parameter. Additionally it is not possible to add a textual description to this parameter or each possible value. Concerning other parameters like measured values, it is a problem, that one cannot indicate offsets, scaling factors or physical units for these values. Another group of parameters is transmitted via CANopen FD as a simple data type like UNSIGNED32, but they represent a bit field with different meanings for specific groups of bits, which cannot be represented in EDS files, too. Finally there is one common disadvantage of the old EDS files: descriptions in multiple languages are not supported. To overcome these disadvantages the members of the task force XML (TF XML) within the CiA have specified a new device description

based on XML. Using XML instead of INI files provides advantages for the validation, transformation, query and presentation of the data using XML mechanisms such as XML schemes, XSLT or XQuery, which shall not be discussed in this paper. The first version of the new XML device description for CANopen has already been published in 2007. The new format has not been used by many device manufacturers or tool vendors in the past. But with the release of CANopen FD the XML device description files are mandatory and a wide-spread usage of the XML based device description is expected in the future.

Structure of XML device description files

The device description format developed by the TF XML was based on existing work of the ISO standard 15745. To ease the interoperability of devices and the integration of plants the ISO standardized an „Application Integration Framework“. This framework describes models and profiles with basic functions and a basic set of device properties. The standard ISO-15745-1 describes general rules and in the standard ISO-15745-2 the description of CAN devices and their communication is specified. These descriptions can be found in the Device Profile and in the Communication Network Profile. Both profiles are derived from the type ISO15745 profile and consist of a ProfileHeader and a ProfileBody. The body contains the actual data and the header provides administrative information about the used ISO15745 profile. A profile container according to ISO-15745-1:2005/Amd1 combines both profiles in an XML Device Description file (XDD), which will be explained in the following sections.

Field bus independent descriptions

In the first part of the device description, the device profile, the device is described regardless of the used communication protocol without any references to CANopen FD or other protocols. As shown in figure 2 this profile consists of the following sections:

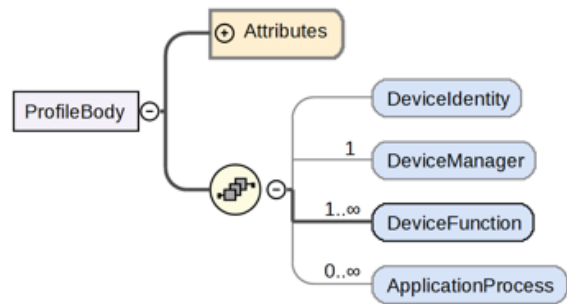


Figure 2: Device profile structure

Element	Description
Device Manager	Attributes to monitor the device (for example LED indicators)
Device Identity	Device name, type, vendor, part number, ...
Device Function	Description of the function using the terminology of the device type: examples: - number of axles: 6 - nominal voltage: 24 V - peak current: 31.41 A
Application Process	Description of data types, templates function parameters, parameter groups and assignment of parameters to functions

The Application Process containing the description of the parameters is the largest part of this profile. The parameter „Modes of operation“ is shown in figure 3. The following examples show the extended features of the XDD format:

- support for multiple languages
- additional descriptive texts
- description of the meaning of particular values
- description of value ranges and specific increments

```

101 <parameter uniqueID="UID_44" access="readWrite">
102 <label lang="en">Modes of Operation</label>
103 <label lang="de">Betriebsart</label>
104 <description lang="en">
105 This object shall indicate the requested operation mode.
106 </description>
107 <description lang="de">
108 Dieser Parameter definiert die angeforderte Betriebsart
109 des Antriebs. Die aktuelle tatsächliche Betriebsart wird
110 im Parameter Modes of Operation Display dargestellt.
111 </description>
112 <SINT/>
113 <defaultValue value="0"/>
114 <allowedValues>
115 <value value="0">
116 <label lang="en">No mode assigned</label>
117 <label lang="de">Keine Betriebsart</label>
118 </value>
119 <value value="1">
120 <label lang="en">Profile Position Mode</label>
121 <label lang="de">Profile Position Mode</label>
122 </value>
123 <range>
124 <step value="1"/></step>
125 <minvalue value="-128">
126 <label lang="en">Manufacturer specific</label>
127 </minvalue>
128 <maxvalue value="-1">
129 <label lang="en">Manufacturer specific</label>
130 </maxvalue>

```

Figure 3: Parameter description (example)

Regarding other parameters like measured values engineering units as well as scaling factors or offsets can be specified. Using these information generic CANopen tools are able to display these values in a more user-friendly way, and system integrators benefit from this specified machine-readable format as well. Figure 4 shows such an example below.

```

132 <parameter uniqueID="UID_47" offset="-100" multiplier="0.05">
133 <label lang="de">Strömungsgeschwindigkeit</label>
134 <label lang="sv">strömshastighet</label>
135 <INT/>
136 <unit>
137 <label m/s</label>
138 </unit>
139 </parameter>

```

Figure 4: Value description (example)

This specification allows the use of templates for similar parameters to reduce the file size and to ensure a consistent description of similar parameters. Additionally, parameters can be combined in hierarchical groups. To provide different views on the parameter or to visualize complex menu structures in engineering tools parameters can even belong to different groups. The groups themselves can have names and descriptions in multiple languages. A further feature of the XDD format is the possibility to describe functions with their input, output and configuration parameters. Function types can be created and instanced. These function instances can be linked with parameters of the application by references. This approach, to be independent of the

communication interface, offers several advantages for device manufacturers:

- focus on the application of the device
- consistent, standardized description
- re-usability for other field buses or real-time Ethernet solutions.

If a description of the application is available, the parameters can be linked to the communication mechanisms of the used communication protocol. This might be Modbus registers, POWERLINK objects or like in our case CANopen FD objects, which will be discussed in the next section in detail. A further advantage of the format is the support of multiple languages. Both embedded elements and references to external dictionaries are supported by the XDD format. Languages can be added by external dictionaries without changing the description file. Furthermore a manufacturer may maintain all dictionaries for various languages and products at a central place to ensure consistent naming among its different product lines.

CANopen FD communication description

The profile „Communication Network_CANopen“ is used to describe the CANopen properties of a device. Beside information about the CAN FD bit rates for the arbitration phase and the data phase and data about the device manufacturer it contains a list of all CANopen FD objects. Each object can be linked with a parameter of the application using references like the attribute „uniqueIDRef“. This is shown in figure 5. A link between an object and a parameter means that the CANopen object owns all attributes of the referenced parameter. CANopen objects without a link to an application parameter like PDO mapping objects, can be still described in the object list. (See object 0x1000 in figure 5). A further property of CANopen are network management parameters, which are described in a separate network management section. These parameters are basically the data from the [DeviceInfo] section of the EDS file as well as some additional master attributes.

```

280 <ProfileBody
281   xsi:type="ProfileBody_CommunicationNetwork_CANopen">
282   <ApplicationLayers>
283     <identity>
284       <vendorId>0x34</vendorId>
285     </identity>
286     <CANopenObjectList>
287       <CANopenObject index="1000" name="Device Type"
288         objectType="7" PDOmapping="no" dataType="7"
289         accessType="const" defaultValue="0x000F0191"
290         highLimit="0xFFFFFFFF" />
291       <CANopenObject index="6060" name="Mode of Operation"
292         PDOmapping="yes" uniqueIDRef="UID_44" />
293       <CANopenObject index="6061"
294         name="Mode of Operation Display"
295         PDOmapping="yes" uniqueIDRef="UID_45" />
296     </CANopenObjectList>
297     <dummyUsage>
298       <dummy entry="Dummy0005=1"/>
299     </dummyUsage>

```

Figure 5: CANopen object list

The main item of the communication profile – the CANopen object list – is likewise used by Ethernet POWERLINK in a nearly identical format providing the advantage of re-usability.

Advantages for CANopen tools

There are several use cases for generic CANopen FD tools that benefit from the XML device descriptions. Figure 6 shows how a generic tool can offer all particular values with the meaning to the user (using the already discussed parameter „Modes of operation“ as an example). This is an advantage especially for manufacturer specific parameters that are not specified by CiA profiles. Compared to the numerical input of a number in the value range of the data type INTEGER8 the usability can be increased by simultaneous minimization of faulty inputs. Additionally, measured values with offset and scaling factors can now be presented with the raw data and the calculated result. Another possibility to present the parameters is the use of parameter groups. Beside the widely used tree view of objects the parameters of a device can be presented in groups. The device manufacturer can combine the parameters in groups depending on their function, and the tool presents the groups accordingly. As a further step, tools can visualize the functions with inputs, outputs and configuration parameters and thus provide an application view on the device, which may furthermore contain descriptions in multiple languages. Concerning these considerations, thinking ahead leads us to the possibility of network planning tools visualizing complete function diagrams of

the network based on the PDO linking of the devices and their functional descriptions.

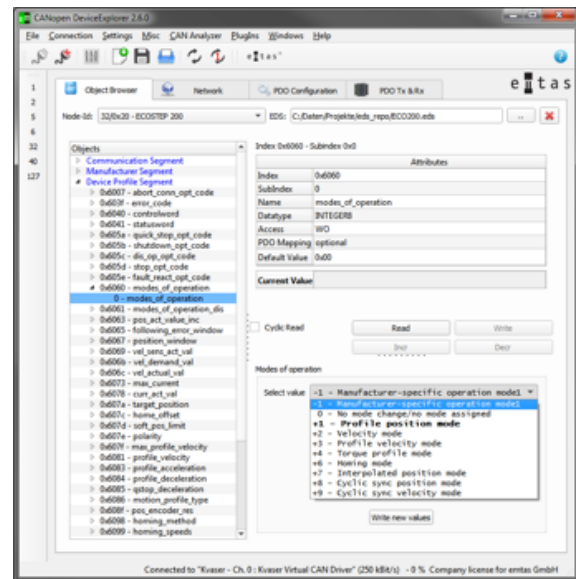


Figure 6: improved configuration options in generic CANopen tools

In these function diagrams the output parameters of one function of one device can be connected with the input parameters of functions of other devices. Further CANopen FD simulation tools, which simulate a CANopen FD slave by reading its device description file, benefit from the additional information in the XDD files. For example the writing of particular values to parameters, which can be rejected separately depending on the value using the additional information about accepted values in the new CANopen FD XDD format.

Device design

At the beginning of a device development both, the design as well as the test at the end of the development benefit from the new XDD format. Because of the present format of the XML device description files the device functions and the communication interface can be already described in a machine-readable format. Thus the XML device description files can be used as a standardized and machine-readable way of specification of such devices. At least for large parts of the device a uniform description language is available for the specification, for the test and for the integration by the end user. By the detailed description of the

device functionality in a machine-readable format tests can be automated in a more comprehensive way. Thus the possibilities of the new XML device descriptions exceed the potential of the hitherto used EDS files.

Creation of XDD files

Different tools with different specializations can be used to create XDD files. If one considers them as pure text files – as a remarkable number of authors of EDS files still do – a simple text editor like for example Notepad might seem usable. XML editor tools instead offer more comfort but most comfort is provided by specialized CANopen FD XDD tools, which have been developed for this purpose. The following section compares these three categories with their advantages and disadvantages.

Text editors Pros

- available free of charge
- for every operating system

Cons

- expert knowledge of CANopen XDD files required
- no checks for wrong syntax
- no semantic checks of the input
- very error-prone
- very time consuming

XML editors Pros

- some tools available as open source
- validation using an XML schema
- input assistance based on the XML schema

Cons

- expert knowledge of CANopen FD XDD files required
- no semantic checks of the input
- time consuming

XDD tools Pros

- validation using an XML schema
- semantic check of the content
- use of CANopen FD profiles possible
- no detailed knowledge of device descriptions required
- time saving compared to XML editors
- possibilities to import existing EDS files

Cons

- only commercial products available
- not available for all operating systems

This compilation shows that only special XDD tools hiding the internals of device description files are suitable for professional use. Several CANopen FD software providers offer such commercial tools.

Migration from EDS files

The fastest way to migrate from EDS to XDD files is to use a commercial XDD tool that supports the import of existing EDS files and generates an XDD file based on the information in the former file. This way will fulfill the formal requirement to provide an XDD file along with the CANopen FD device. Due to the limited set of information in the EDS file the generated XDD will not contain more information but only the ProfileBody CommunicationNetwork_CANopen with the CANopenObjectList. In order to use the new features of the XDD files additional effort has to be made to add the ApplicationProcess including a detailed parameter and function description. There is no possibility to do it automatically, but manually.

Summary and future developments

The discussed use cases and advantages of the CANopen FD XML device descriptions show that the XDD format offers a broad range of new possibilities to describe CANopen FD devices and their applications using a standardized format. It covers more details and exceeds the range of functions of the previous EDS file significantly. In order to address the problem of the little usage of the XDD format and to ease the use of the XDD files the TF XML is currently reviewing the specification. It is furthermore planned to provide an updated, simplified version or clarifications and application notes that explain the use of the XDD format.

References

- [1] CiA DS 301, CANopen application layer and communication profile
- [2] CiA DSP 311, CANopen device description – XML schema definition
- [3] ISO, Industrial automation systems and integration – Open systems application integration framework 15745 Part 1: Generic reference description
- [4] ISO, Industrial automation systems and integration – Open systems application integration framework 15745 Part 2: Reference description for ISO 11898-based control systems
- [5] Schumann, Thilo; Gedenk, Torsten: Gerätebeschreibungen im XML-Format, etz 08/2005

Torsten Gedenk
emtas GmbH
Fritz-Haber-Str. 9
DE-06217 Merseburg
Tel.: +49 3461-79416-0
ged@emtas.de
www.emtas.de