PUBLICLY
AVAILABLE
SPECIFICATION

IEC PAS 62453-4

Pre-Standard

First edition
2006-05

Field Device Tool (FDT) interface specification –

Part 4:
HART communication

Reference number
IEC/PAS 62453-4:2006(E)

Publication numbering

As from 1 January 1997 all IEC publications are issued with a designation in the 60000 series. For example, IEC 34-1 is now referred to as IEC 60034-1.

Consolidated editions

The IEC is now publishing consolidated versions of its publications. For example, edition numbers 1.0, 1.1 and 1.2 refer, respectively, to the base publication, the base publication incorporating amendment 1 and the base publication incorporating amendments 1 and 2.

Further information on IEC publications

The technical content of IEC publications is kept under constant review by the IEC, thus ensuring that the content reflects current technology. Information relating to this publication, including its validity, is available in the IEC Catalogue of publications (see below) in addition to new editions, amendments and corrigenda. Information on the subjects under consideration and work in progress undertaken by the technical committee which has prepared this publication, as well as the list of publications issued, is also available from the following:

- **IEC Web Site (www.iec.ch)**

- **Catalogue of IEC publications**

  The on-line catalogue on the IEC web site (www.iec.ch/searchpub) enables you to search by a variety of criteria including text searches, technical committees and date of publication. On-line information is also available on recently issued publications, withdrawn and replaced publications, as well as corrigenda.

- **IEC Just Published**

  This summary of recently issued publications (www.iec.ch/online_news/ justpub) is also available by email. Please contact the Customer Service Centre (see below) for further information.

- **Customer Service Centre**

  If you have any questions regarding this publication or need further assistance, please contact the Customer Service Centre:

  Email: custserv@iec.ch
  Tel: +41 22 919 02 11
  Fax: +41 22 919 03 00

PUBLICLY
AVAILABLE
SPECIFICATION

IEC PAS 62453-4

Pre-Standard

First edition
2006-05

Field Device Tool (FDT) interface specification –

Part 4:
HART communication

International Electrotechnical Commission, 3, rue de Varembé, PO Box 131, CH-1211 Geneva 20, Switzerland
Telephone: +41 22 919 02 11 Telefax: +41 22 919 03 00 E-mail: inmail@iec.ch Web: www.iec.ch

Commission Electrotechnique Internationale
International Electrotechnical Commission
Международная Электротехническая Комиссия

PRICE CODE U

For price, see current catalogue

CONTENTS

FOREWORD....3
INTRODUCTION....5
1 Scope....6
2 Normative references....6
3 Provided data....6
3.1 Interface IDtmParameter....6
3.2 SingleDataAccess interfaces....6
4 Protocol specific usage of XML attributes....6
5 Bus category....7
6 FDT sequence charts....7
6.1 HART burst mode subscription....7
7 HARTCommunicationSchema....9
8 ChannelParameterSchema....14
9 Topology scan schema....17
10 HART device identification....18
10.1 FDTHARTIdentSchema....18
10.2 FDTHARTScanIdentSchema....22
10.3 FDTHARTDeviceTypeIdentSchema....24
BIBLIOGRAPHY....26

Figure 1 – HART burst mode subscription....8

Table 1 – Protocol specific usage of XML attributes....6
Table 2 – HARTCommunicationschema – attributes and elements....9
Table 3 – ChannelParameterSchema....14
Table 4 – FDTHARTIdentSchema – attributes with protocol specific mapping....18
Table 5 – FDTHARTIdentSchema – attributes without protocol independent semantics....20
Table 6 – FDTHARTIdentSchema – attributes and elements with protocol independent semantics....21
Table 7 – FDTHARTScanIdentSchema – attributes and elements....22
Table 8 – FDTHARTDeviceTypeIdentSchema attributes and elements....24

INTERNATIONAL ELECTROTECHNICAL COMMISSION

_____

Field Device Tool (FDT) interface specification –

Part 4: HART communication

FOREWORD

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

A PAS is a technical specification not fulfilling the requirements for a standard but made available to the public.

IEC-PAS 62453-4 has been processed by subcommittee 65C: Digital communications, of IEC technical committee 65: Industrial-process measurement and control.

The text of this PAS is based on the following document:	This PAS was approved for publication by the P-members of the committee concerned as indicated in the following document
Draft PAS	**Report on voting**
65C/398A/NP	65C/411/RVN

Following publication of this PAS, which is a pre-standard, the technical committee or subcommittee concerned will transform it into an International Standard.

This PAS shall remain valid for an initial maximum period of three years starting from 2006-05. The validity may be extended for a single three-year period, following which it shall be revised to become another type of normative document or shall be withdrawn.

IEC 62453 consists of the following parts under the general title *Field Device Tool (FDT) interface specification:*

Part 1: Concepts and detailed description

Part 2: INTERBUS communication

Part 3: PROFIBUS communication

Part 4: HART communication

Part 5: FOUNDATION FIELDBUS communication

INTRODUCTION

This PAS is an interface specification for developers of FDT components for Function Control and Data Access within a Client Server architecture. The specification is a result of an analysis and design process to develop standard interfaces to facilitate the development of servers and clients by multiple vendors that shall inter-operate seamlessly.

With the integration of fieldbusses into control systems, there are a few other tasks which must be performed. This applies to fieldbusses in general. Although there are fieldbus- and device-specific tools, there is no unified way to integrate those tools into higher level system-wide planning or engineering tools. In particular, for use in extensive and heterogeneous control systems, typically in the area of the process industry, the unambiguous definition of engineering interfaces that are easy to use for all those involved, is of great importance.

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

Field Device Tool (FDT) interface specification –

Part 4: HART communication

1 Scope

This part of IEC 62435 provides information for integrating the HART[1] protocol into the FDT interface specification (IEC 62453-1). This PAS neither contains the FDT specification nor modifies it.

2 Normative references

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

IEC 62453-1: *Field Device Tool (FDT) interface specification – Part 1: Concepts and detailed description*

3 Provided data

3.1 Interface IDtmParameter

The minimum set of provided data should be:

- The first four provided process related values (PV, SV, ...) - if available - are modeled as channel references. The referenced channel must include ranges and scaling.

3.2 SingleDataAccess interfaces

At least all parameters of the Universal and Common Practice (as far as the device supports the function) commands must be exposed via the interfaces IDtmSingleDeviceDataAccess and IDtmSingleInstanceDataAccess.

Furthermore the Response Byte 0 and the Response Byte 1 for each command must be exposed.

4 Protocol specific usage of XML attributes

The following Table 1 shows how general attributes are used with HART devices.

Table 1 – Protocol specific usage of XML attributes

Attribute	Description for use in HART
address	The address attribute (defined in FDTDataTypesSchema.xml) is not mandatory for the exposed parameters in the DTMs. But if the address attribute is used the string must be constructed according to the rules of the semanticId. That means the attribute ‘semanticId’ is always the same as the attribute ‘address’
busCategory	See Clause 5

[1] HART ® is the trade name of the a product supplied byHART Communication Foundation. This information is given for convenience of users of this document and does not constitute an endorsment by IEC of the product named. Equivalent products may be used if they can be shown to lead to the same results.

Attribute	Description for use in HART
deviceTypeId	The attribute "fdt:DtmDeviceType/@deviceTypeId" must contain the DeviceTypeID of the supported physical device according to the HCF online product catalog
manufacturerId	Enter manufacturer according to HCF list
semanticId applicationDomain	The applicationDomain attribute is: FDT_HART The sematicId for HART protocol-related parameter is directly related to the protocol specification. The definition of the HART commands is the base for the semanticId. The semanticId for a parameter follows the following definition: CMDxxBy and CMD31EXTENDEDxxBy for extended HART 6 device family commands. The semanticIds for the Response Byte 0 and 1 defined in the HART specification are: CMDxxRESPONSE_BYTE_0 CMDxxRESPONSE_BYTE_1 xx: represents the command number, getting the parameter via HART protocol or the device family command number y: start byte within the command definition xx, yy are based on decimal format without leading '0'
subDeviceType	Enter manufacturer-specific value

5 Bus category

HART protocol is identified by the following unique identifier in busCategory attributes within XML BusCategory elements.

BusCategory Element	Description
036D1498-387B-11D4-86E1-00E0987270B9	Support of HART protocol

6 FDT sequence charts

6.1 HART burst mode subscription

A subscription of device initiated data transfer can be requested by sending a TransactionRequest with SubscribeRequest content (see Figure 1). The Communication Channel may detect if the device is already in Burst Mode. In HART 5 this can be detected only when HART Burst frames are received from the device. In HART 6 the Burst mode can be detected using HART command 105. The Communication Channel answers on a SubscribeRequest by calling OnTransactionResponse with a SubscribeResponse content. If HART Burst frames are received, the device is in Burst mode and burstModeDetected value is set to TRUE. This means that device DTM will start to receive Burst messages via OnTransactionResponse mechanism. In case no burst messages were received, burstModeDetected value is set to FALSE. It is up to device DTM to set device into Burst mode. Then device DTM may call TransactionRequest with SubscribeRequest content again in order to receive Burst messages.

In order to unsubscribe, the device DTM sends a TransactionRequest with a UnsubcribeRequest. The Communication Channel answers by calling OnTransactionResponse with a SubscribeResponse where burstModeDetected value is set to FALSE. Device DTM will not receive any more Burst information via OnTransactionResponse mechanism. The Communication Channel does not switch off the Burst Mode in the device. The device DTM may switch Burst Mode on or off by using normal TransactionRequests (HART command 109). This is independent of the subscription.

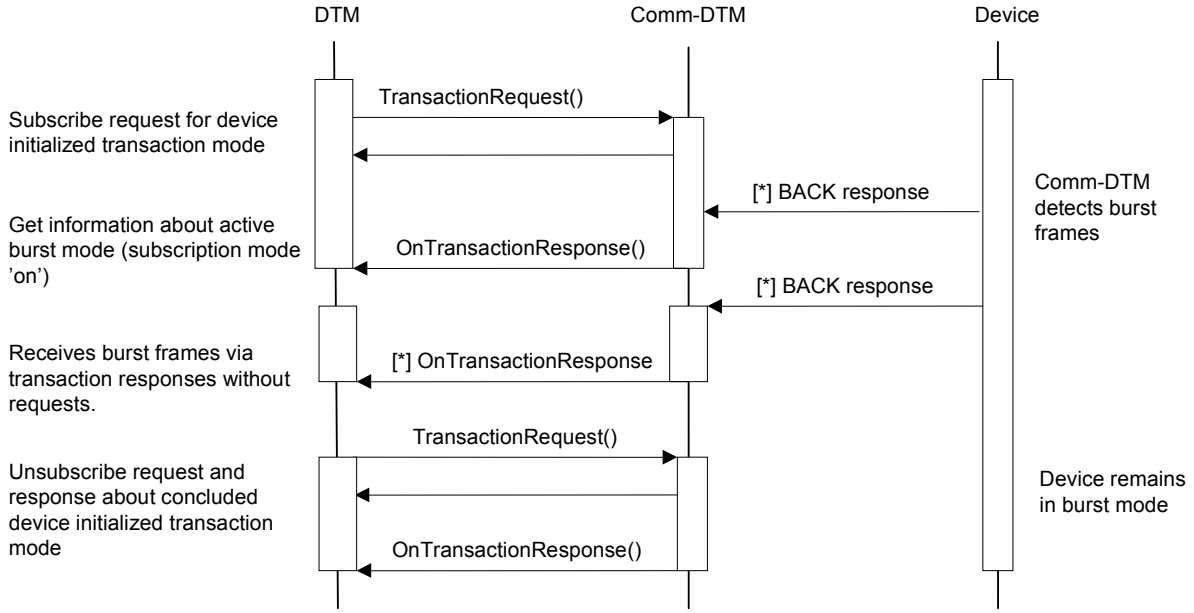

Figure 1 – HART burst mode subscription

Used methods:

IFdtCommunication::TransactionRequest()

IFdtCommunication::OnTransactionResponse()

7 HARTCommunicationSchema

Used at: IFdtCommunication::ConnectRequest()

IFdtCommunicationEvents2::OnConnectResponse2()

IFdtCommunication::DisconnectRequest()

IFdtCommunicationEvents::OnDisconnectResponse ()

IFdtCommunication::TransactionRequest()

IFdtCommunicationEvents::OnTransactionResponse()

The XML document contains the address information and the communication data which are explained in Table 2.

Table 2 – HARTCommunicationschema – attributes and elements

Attribute	Description
address1	Address information according to the HART specification
address2	Address information according to the HART specification
address3	Address information according to the HART specification
burstFrame	Information whether the HART response is a Burst frame (message or not)
burstModeDetected	Indicates whether the Communication Channel has detected that the device is already in burst mode. This is detected during a subscription request
commandNumber	Address information according to the HART specification
communicationReference	Mandatory identifier for a communication link to a device. This identifier is allocated by the communication component during the connect. The address information has to be used for all following communication calls
delayTime	Minimum delay time in [ms] between two communication calls
deviceStatus	Changed description Status information. This is the second status byte returned in HART command responses according to the HART specification
deviceTypeId	Address information according to the HART specification
longFrameRequired	Address information according to the HART specification
manufacturerId	Address information according to the HART specification (see table relating to MANUFACTURER IDENTIFICATION CODES)
preambleCount	At the connect request the attribute is optional and contains a hint for the communication component about the number of preambles, required by the device type. At the connect response the attribute is mandatory and contains the information about the currently used preambleCount
primaryMaster	At the connect request the attribute is optional and contains a hint for a communication component that a DTM requires communication as primary or secondary master. At the connect response the attribute is mandatory contains the information about the current state of the master
schemaVersion	Defines the version of the schema
sequenceTime	Period of time in [ms] for the whole sequence
shortAddress	Address information according to the HART specification. This value is accessible via the attribute slaveAddress defined within the DTMParameterSchema. SlaveAddress is part of the BusInformation structure. These values must be set by the responsible component as described in clause Nested Communication of IEC/PAS 62453-1
value	Variable for status information
systemTag	System Tag of a DTM. It is strongly recommended to provide the attribute in the Request document

Tag	Description
Abort	Describes the abort
CommandResponse	– Changed description Status information. This is computed from the first status byte returned in HART command responses according to the HART specification. If bit 7 of the first status byte is clear this value contains the value in the first status byte. If bit 7 is set this element is not returned in the status structure
CommunicationStatus	– Changed description Status information. This is computed from the first status byte returned in HART command responses according to the HART specification. If bit 7 of the first status byte is set this value contains the value in the first status byte (this is where we need to state whether it is the first status byte or bits 0-6 of the first status byte). If bit 7 is clear this element is not returned in the status structure
ConnectRequest	Describes the communication request
ConnectResponse	Describes the communication response
DataExchangeRequest	Describes the communication request
DataExchangeResponse	Describes the communication response
DisconnectRequest	Describes the communication request
DisconnectResponse	Describes the communication response
SubscribeRequest	Describes the subscription request for device initiated data transfer (HART Burst mode)
SubscribeResponse	Describes the subscription response request for device initiated data transfer (HART Burst mode)
UnsubscribeRequest	Describes the request to release the subscription for device initiated data transfer (HART Burst mode)
UnsubscribeResponse	Describes the response request to release the subscription for device initiated data transfer (HART Burst mode)
FDT	Root tag
LongAddress	Address information according to the HART specification (only supported by HART devices based on HART revision > 5, see related documentation) In the HART protocol Manufacturer ID and Device type ID are contained in the longaddress. If the channel delivers different values in fdthart:manufacturerId / fdthart:deviceTypeId and in the corresponding bytes in fdthart:LongAddress, the following rule applies: * The fdthart:LongAddress has to be used for communication and * The fdthart:manufacturerId and fdthart:deviceTypeId may be used only as information about the manufacturer and the type of device
SequenceBegin	Describes the sequence begin
SequenceEnd	Describes the sequence end
SequenceStart	Describes the sequence start
ShortAddress	Address information according to the HART specification
Status	Status information according to the HART specification

NOTE The attribute ‘fdt:tag’, used within this schema, is accessible via the attribute ‘fdt:tag’ defined within the DTMParameterSchema. ‘fdt:tag’ is part of the DtmDevice structure and contains the HART-specific value called TAG, which is used e.g. within command #11, ‘READ UNIQUE IDENTIFIER ASSOCIATED WITH TAG’. These values must be set by the responsible component as described in clause Nested Communication of IEC/PAS 62453-1.

```
<Schema name="FDTHARTCommunicationSchema" xmlns="urn:schemas-microsoft-com:xml-data" xmlns:dt="urn:schemas-microsoft-com:datatypes" xmlns:fdt="x-schema:FDTDataTypesSchema.xml">
    <!--Definition of Attributes-->
    <AttributeType name="schemaVersion" dt:type="number" default="1.21"/>
```

```
<AttributeType name="address1" dt:type="ui1"/>
<AttributeType name="address2" dt:type="ui1"/>
<AttributeType name="address3" dt:type="ui1"/>
<AttributeType name="commandNumber" dt:type="ui1"/>
<AttributeType name="communicationReference" dt:type="uuid"/>
<AttributeType name="deviceStatus" dt:type="ui1"/>
<AttributeType name="deviceTypeId" dt:type="ui1"/>
<AttributeType name="longFrameRequired" dt:type="boolean"/>
<AttributeType name="manufacturerId" dt:type="ui1"/>
<AttributeType name="preambleCount" dt:type="ui1"/>
<AttributeType name="primaryMaster" dt:type="boolean"/>
<AttributeType name="shortAddress" dt:type="ui1"/>
<AttributeType name="value" dt:type="ui1"/>
<AttributeType name="sequenceTime" dt:type="ui4"/>
<AttributeType name="delayTime" dt:type="ui4"/>
<AttributeType name="burstFrame" dt:type="boolean"/>
<AttributeType name="burstModeDetected" dt:type="boolean"/>

<!--Definition of Elements-->
<ElementType name="CommunicationStatus" content="empty" model="closed">
    <attribute type="fdt:nodeId" required="no"/>
    <attribute type="value" required="yes"/>
</ElementType>
<ElementType name="CommandResponse" content="empty" model="closed">
    <attribute type="fdt:nodeId" required="no"/>
    <attribute type="value" required="yes"/>
</ElementType>
<ElementType name="Status" content="eltOnly" model="closed">
    <attribute type="fdt:nodeId" required="no"/>
    <attribute type="deviceStatus" required="yes"/>
    <group order="one" minOccurs="1" maxOccurs="1">
        <element type="CommunicationStatus"/>
        <element type="CommandResponse"/>
    </group>
</ElementType>
<ElementType name="LongAddress" content="empty" model="closed">
    <attribute type="fdt:nodeId" required="no"/>
    <attribute type="manufacturerId" required="yes"/>
    <attribute type="deviceTypeId" required="yes"/>
    <attribute type="address1" required="yes"/>
    <attribute type="address2" required="yes"/>
    <attribute type="address3" required="yes"/>
</ElementType>
<ElementType name="ShortAddress" content="empty" model="closed">
    <attribute type="fdt:nodeId" required="no"/>
    <attribute type="shortAddress" required="yes"/>
</ElementType>
<ElementType name="ConnectRequest" content="eltOnly" model="closed">
    <attribute type="fdt:nodeId" required="no"/>
    <attribute type="fdt:tag" required="yes"/>
    <attribute type="preambleCount" required="no"/>
    <attribute type="primaryMaster" required="no"/>
    <attribute type="longFrameRequired" required="no"/>
    <attribute type="fdt:systemTag" required="no"/>
    <element type="LongAddress" minOccurs="0" maxOccurs="1"/>
    <element type="ShortAddress" minOccurs="1" maxOccurs="1"/>
</ElementType>
<ElementType name="ConnectResponse" content="eltOnly" model="closed">
    <attribute type="fdt:nodeId" required="no"/>
    <attribute type="fdt:tag" required="yes"/>
    <attribute type="preambleCount" required="yes"/>
    <attribute type="primaryMaster" required="yes"/>
    <attribute type="communicationReference" required="yes"/>
    <element type="LongAddress" minOccurs="0" maxOccurs="1"/>
    <element type="ShortAddress" minOccurs="1" maxOccurs="1"/>
</ElementType>
<ElementType name="DisconnectRequest" content="empty" model="closed">
    <attribute type="fdt:nodeId" required="no"/>
    <attribute type="communicationReference" required="yes"/>
</ElementType>
<ElementType name="DisconnectResponse" content="empty" model="closed">
    <attribute type="fdt:nodeId" required="no"/>
    <attribute type="communicationReference" required="yes"/>
</ElementType>
<ElementType name="DataExchangeRequest" content="eltOnly" model="closed">
    <attribute type="fdt:nodeId" required="no"/>
    <attribute type="commandNumber" required="yes"/>
    <attribute type="communicationReference" required="yes"/>
```

```
        <element type="fdt:CommunicationData" minOccurs="0" maxOccurs="1"/>
    </ElementType>
    <ElementType name="DataExchangeResponse" content="eltOnly" model="closed">
        <attribute type="fdt:nodeId" required="no"/>
        <attribute type="commandNumber" required="yes"/>
        <attribute type="communicationReference" required="yes"/>
        <attribute type="burstFrame" required="no"/>
        <element type="fdt:CommunicationData" minOccurs="0" maxOccurs="1"/>
        <element type="Status" minOccurs="1" maxOccurs="1"/>
    </ElementType>
    <ElementType name="SequenceBegin" content="empty" model="closed">
        <attribute type="sequenceTime" required="no"/>
        <attribute type="delayTime" required="no"/>
        <attribute type="communicationReference" required="yes"/>
    </ElementType>
    <ElementType name="SequenceEnd" content="empty" model="closed">
        <attribute type="communicationReference" required="yes"/>
    </ElementType>
    <ElementType name="SequenceStart" content="empty" model="closed">
        <attribute type="communicationReference" required="yes"/>
    </ElementType>
    <ElementType name="Abort" content="empty" model="closed">
         <attribute type="communicationReference" required="no"/>
    </ElementType>
   <ElementType name="SubscribeRequest" content="empty" model="closed">
        <attribute type="fdt:nodeId" required="no"/>
        <attribute type="communicationReference" required="yes"/>
    </ElementType>
    <ElementType name="SubscribeResponse" content="empty" model="closed">
        <attribute type="fdt:nodeId" required="no"/>
        <attribute type="communicationReference" required="yes"/>
        <attribute type="burstModeDetected" required="yes"/>
        <attribute type="fdt:communicationError" required="no"/>
     </ElementType>
     <ElementType name="UnsubscribeRequest" content="empty" model="closed">
        <attribute type="fdt:nodeId" required="no"/>
        <attribute type="communicationReference" required="yes"/>
     </ElementType>
     <ElementType name="UnsubscribeResponse" content="empty" model="closed">
        <attribute type="fdt:nodeId" required="no"/>
        <attribute type="communicationReference" required="yes"/>
        <attribute type="fdt:communicationError" required="no"/>
      </ElementType>
    <ElementType name="FDT" content="eltOnly" model="closed">
        <attribute type="schemaVersion" required="no"/>
        <attribute type="fdt:nodeId" required="no"/>
        <group order="one" minOccurs="1" maxOccurs="1">
            <element type="ConnectRequest"/>
            <element type="ConnectResponse"/>
            <element type="DisconnectRequest"/>
            <element type="DisconnectResponse"/>
            <element type="DataExchangeRequest"/>
            <element type="DataExchangeResponse"/>
            <element type="SequenceBegin"/>
            <element type="SequenceEnd"/>
            <element type="SequenceStart"/>
            <element type="Abort"/>
            <element type="SubscribeRequest"/>
            <element type="SubscribeResponse"/>
            <element type="UnsubscribeRequest"/>
            <element type="UnsubscribeResponse"/>
            <element type="fdt:CommunicationError"/>
        </group>
    </ElementType>
</Schema>
```

Example:

```
<?xml version="1.0"?>
<FDT xmlns="x-schema:FDTHARTCommunicationSchema.xml" xmlns:fdt="x-schema:FDTDataTypesSchema.xml"
schemaVersion="1.21">
    <DataExchangeRequest fdt:nodeId="myId" commandNumber="42" communicationReference="6B29FC40-CA47-1067-
B31D-00DD010662DA"/>
</FDT>
```

```
<?xml version="1.0"?>
<FDT xmlns="x-schema:FDTHARTCommunicationSchema.xml" xmlns:fdt="x-schema:FDTDataTypesSchema.xml"
schemaVersion="1.21">
    <DataExchangeResponse commandNumber="1" communicationReference="6B29FC40-CA47-1067-B31D-
00DD010662DA">
        <fdt:CommunicationData byteArray="ff02"/>
        <Status deviceStatus="0">
            <CommandResponse value="1"/>
        </Status>
    </DataExchangeResponse>
</FDT>
```

Examples for HART burst mode:

The request from the DeviceDTM:

```
<?xml version="1.0"?>
<FDT xmlns="x-schema:FDTHARTCommunicationSchema.xml" xmlns:fdt="x-schema:FDTDataTypesSchema.xml">
    <SubscribeRequest fdt:nodeId="myId" communicationReference="6B29FC40-CA47-1067-B31D-00DD010662DA"/>
</FDT>
```

Answer if the device is already in burst mode:

```
<?xml version="1.0"?>
<FDT xmlns="x-schema:FDTHARTCommunicationSchema.xml" xmlns:fdt="x-schema:FDTDataTypesSchema.xml">
    <SubscribeResponse communicationReference="6B29FC40-CA47-1067-B31D-00DD010662DA"
            burstModeDetected="1" />
</FDT>
```

Receiving a burst frame:

```
<FDT xmlns="x-schema:FDTHARTCommunicationSchema.xml" xmlns:fdt="x-schema:FDTDataTypesSchema.xml">
    <DataExchangeResponse commandNumber="1" communicationReference="6B29FC40-CA47-1067-B31D-
00DD010662DA" burstFrame=”1”>
        <fdt:CommunicationData byteArray="ff02"/>
        <Status deviceStatus="0">
            <CommandResponse value="1"/>
        </Status>
    </DataExchangeResponse>
</FDT>
```

Unsubscribing:

```
<FDT xmlns="x-schema:FDTHARTCommunicationSchema.xml" xmlns:fdt="x-schema:FDTDataTypesSchema.xml">
    <UnsubscribeRequest fdt:nodeId="myId" communicationReference="6B29FC40-CA47-1067-B31D-00DD010662DA"/>
</FDT>
```

Answer to unsubscribe:

```
<?xml version="1.0"?>
<FDT xmlns="x-schema:FDTHARTCommunicationSchema.xml" xmlns:fdt="x-schema:FDTDataTypesSchema.xml">
    <UnsubscribeResponse communicationReference="6B29FC40-CA47-1067-B31D-00DD010662DA" />
</FDT>
```

8 ChannelParameterSchema

It is up to a DTM whether it provides any channels. If a DTM allows a Frame Application, other DTMs, or a controller the direct access to its process values via HART protocol it should provide channel objects as described in this clause. Only the complete description of all channels belonging to a HART command allow a proper access for external applications.

The description of channels, especially of the process values, allows the Frame Application to support the device in a more sufficient way.

Used at: IFdtChannel::GetChannelParameters()

IFdtChannel::SetChannelParameters()

The XML document describes a how to access a channel via a HART command, see Table 3.

Table 3 – ChannelParameterSchema

Attribute	Description
byteLength	Number of static bytes in a Request or in a Reply
commandNumber	Number of the HART command containing the channel value
frameApplicationTag	Frame Application specific tag used for identification and navigation. The DTM should display this tag at channel-specific user interfaces
gatewayBusCategory	Unique identifier for a supported bus type like Profibus or HART according to the FDT specific CATID
protectedByChannelAssignment	TRUE if the channels is set to read only by the Frame Application. Usually set to TRUE if a channel assignment exists
schemaVersion	Defines the version of the schema
value	Current value of a channel for read or write
Tag	**Description**
FDT	Root tag
CommandParameters	Static command parameter bytes in a Request or in a Reply.
FDTChannel	Description of the channel
FDTChannelType	Description of the channel component in case of channels with gateway functionality
ReadCommand	Description of the command to read the channel from a device
Reply	Description of the reply structure of a HART command according to the HART specification
Request	Description of the request structure of a HART command according to the HART specification
ResponseCodes	Collection of HART specific response codes according to the HART specification (known as COMMAND-SPECIFIC RESPONSE CODES)
WriteCommand	Description of the command to write the channel to a device

```
<Schema name="FDTHARTChannelParameterSchema" xmlns="urn:schemas-microsoft-com:xml-data" xmlns:dt="urn:schemas-microsoft-com:datatypes" xmlns:fdt="x-schema:FDTDataTypesSchema.xml" xmlns:appId="x-schema:FDTApplicationIdSchema.xml">
    <!--Definition of Attributes-->
    <AttributeType name="schemaVersion" dt:type="number" default="1.21"/>
    <AttributeType name="byteLength" dt:type="ui1"/>
    <AttributeType name="commandNumber" dt:type="ui4"/>
    <AttributeType name="frameApplicationTag" dt:type="string"/>
    <AttributeType name="gatewayBusCategory" dt:type="uuid"/>
```

```
    <AttributeType name="protectedByChannelAssignment" dt:type="boolean"/>
    <AttributeType name="value" dt:type="string"/>
    <!--Definition of Elements-->
  <ElementType name="CommandParameters" content="empty" model="closed">
    <attribute type="fdt:nodeId" required="no"/>
    <attribute type="fdt:binData" required="no"/>
    <attribute type="byteLength" required="yes"/>
  </ElementType>
    <ElementType name="Request" content="eltOnly" model="closed" order="many">
        <attribute type="fdt:nodeId" required="no"/>
          <group order="many">
      <element type="fdt:ChannelReference" minOccurs="0" maxOccurs="*"/>
      <element type="CommandParameters" minOccurs="0" maxOccurs="*"/>
    </group>
    </ElementType>
    <ElementType name="ResponseCodes" content="eltOnly" model="closed">
        <attribute type="fdt:nodeId" required="no"/>
        <element type="fdt:EnumeratorEntry" minOccurs="1" maxOccurs="*"/>
    </ElementType>
    <ElementType name="Reply" content="eltOnly" model="closed">
        <attribute type="fdt:nodeId" required="no"/>
          <group order="many">
      <element type="fdt:ChannelReference" minOccurs="0" maxOccurs="*"/>
      <element type="CommandParameters" minOccurs="0" maxOccurs="*"/>
    </group>
        <element type="ResponseCodes" minOccurs="0" maxOccurs="1"/>
    </ElementType>
    <ElementType name="ReadCommand" content="eltOnly" model="closed">
        <attribute type="fdt:nodeId" required="no"/>
        <attribute type="commandNumber" required="yes"/>
        <element type="Request" minOccurs="0" maxOccurs="1"/>
        <element type="Reply" minOccurs="0" maxOccurs="1"/>
        <element type="ResponseCodes" minOccurs="0" maxOccurs="1"/>
    </ElementType>
    <ElementType name="WriteCommand" content="eltOnly" model="closed">
        <attribute type="fdt:nodeId" required="no"/>
        <attribute type="commandNumber" required="yes"/>
        <element type="Request" minOccurs="0" maxOccurs="1"/>
        <element type="Reply" minOccurs="0" maxOccurs="1"/>
        <element type="ResponseCodes" minOccurs="0" maxOccurs="1"/>
    </ElementType>
    <ElementType name="FDTChannel" content="eltOnly" model="closed" order="seq">
        <attribute type="schemaVersion" required="no"/>
        <attribute type="fdt:nodeId" required="no"/>
        <attribute type="fdt:tag" required="yes"/>
        <attribute type="fdt:id" required="yes"/>
        <attribute type="fdt:descriptor" required="no"/>
        <attribute type="protectedByChannelAssignment" required="yes"/>
        <attribute type="fdt:dataType" required="yes"/>
        <attribute type="byteLength" required="yes"/>
        <attribute type="fdt:signalType" required="yes"/>
        <attribute type="frameApplicationTag" required="no"/>
        <attribute type="appId:applicationId" required="no"/>
        <element type="fdt:SemanticInformation" minOccurs="0" maxOccurs="*"/>
        <element type="fdt:BitEnumeratorEntries" minOccurs="0" maxOccurs="1"/>
        <element type="fdt:EnumeratorEntries" minOccurs="0" maxOccurs="1"/>
        <element type="fdt:Unit" minOccurs="0" maxOccurs="1"/>
        <element type="ReadCommand" minOccurs="0" maxOccurs="1"/>
        <element type="WriteCommand" minOccurs="0" maxOccurs="1"/>
        <element type="fdt:Alarms" minOccurs="0" maxOccurs="1"/>
        <element type="fdt:Ranges" minOccurs="0" maxOccurs="1"/>
        <element type="fdt:Deadband" minOccurs="0" maxOccurs="1"/>
        <element type="fdt:SubstituteValue" minOccurs="0" maxOccurs="1"/>
    </ElementType>
    <ElementType name="FDTChannelType" content="eltOnly" model="closed">
        <attribute type="fdt:nodeId" required="no"/>
        <element type="fdt:VersionInformation" minOccurs="1" maxOccurs="1"/>
        <attribute type="gatewayBusCategory" required="no"/>
    </ElementType>
    <ElementType name="FDT" content="eltOnly" model="closed">
        <attribute type="fdt:nodeId" required="no"/>
        <attribute type="schemaVersion" required="no"/>
        <element type="FDTChannelType" minOccurs="1" maxOccurs="1"/>
        <element type="FDTChannel" minOccurs="1" maxOccurs="1"/>
    </ElementType>
</Schema>
```

Example:

```
<?xml version="1.0"?>
<FDT xmlns="x-schema:FDTHARTChannelParameterSchema.xml" xmlns:fdt="x-schema:FDTDataTypesSchema.xml">
    <FDTChannelType>
        <fdt:VersionInformation name="myname" vendor="myVendor" version="1.0" date="2000-08-05"/>
    </FDTChannelType>
    <FDTChannel fdt:tag="myTag" fdt:id="PV" protectedByChannelAssignment="0" fdt:dataType="float" byteLength="4"
fdt:signalType="output">
        <ReadCommand commandNumber="1">
            <Reply>
                <fdt:ChannelReference idref="PV_UNIT"/>
                <fdt:ChannelReference idref="PV"/>
            </Reply>
            <ResponseCodes>
                <fdt:EnumeratorEntry index="8" name="Warning: Update Failure"/>
            </ResponseCodes>
        </ReadCommand>
        <fdt:Alarms>
            <fdt:Alarm alarmType="lowAlarm">
                <fdt:StaticValue staticValue="25"/>
            </fdt:Alarm>
            <fdt:Alarm alarmType="highAlarm">
                <fdt:StaticValue staticValue="100"/>
            </fdt:Alarm>
        </fdt:Alarms>
        <fdt:Ranges>
            <fdt:Range>
                <fdt:LowerRange>
                    <fdt:ChannelReference idref="PV_LOWER_RANGE_VALUE"/>
                </fdt:LowerRange>
                <fdt:UpperRange>
                    <fdt:ChannelReference idref="PV_UPPER_RANGE_VALUE"/>
                </fdt:UpperRange>
                <fdt:Unit>
                    <fdt:ChannelReference idref="PV_RANGE_VALUES_UNITS_CODE"/>
                </fdt:Unit>
            </fdt:Range>
        </fdt:Ranges>
    </FDTChannel>
</FDT>


<?xml version="1.0"?>
<FDT xmlns="x-schema:FDTHARTChannelParameterSchema.xml" xmlns:fdt="x-schema:FDTDataTypesSchema.xml">
    <FDTChannelType>
        <fdt:VersionInformation name="myname" vendor="myVendor" version="1.0" date="2000-08-05"/>
    </FDTChannelType>
    <FDTChannel fdt:tag="myTag" fdt:id="PV_UNIT" protectedByChannelAssignment="0" fdt:dataType="byte" byteLength="1"
fdt:signalType="input">
        <fdt:EnumeratorEntries>
            <fdt:EnumeratorEntry index="7" name="bar"/>
            <fdt:EnumeratorEntry index="8" name="mbar"/>
        </fdt:EnumeratorEntries>
        <ReadCommand commandNumber="1">
            <Reply>
                <fdt:ChannelReference idref="PV_UNIT"/>
                <fdt:ChannelReference idref="PV"/>
            </Reply>
            <ResponseCodes>
                <fdt:EnumeratorEntry index="8" name="Warning: Update Failure"/>
            </ResponseCodes>
        </ReadCommand>
        <WriteCommand commandNumber="44">
            <Request>
                <fdt:ChannelReference idref="PV_UNIT"/>
            </Request>
            <Reply>
                <fdt:ChannelReference idref="PV_UNIT"/>
            </Reply>
        </WriteCommand>
    </FDTChannel>
</FDT>
```

9 Topology scan schema

used at: IDtmEvents::OnScanResponse()

The XML document describes one entry in the list of scanned HART-Devices.

Tag	Description
HARTDevice	Definiton of a HART device concerning the scan response

```
<?xml version="1.0"?>
<Schema name="DTMHARTDeviceSchema" xmlns="urn:schemas-microsoft-com:xml-data" xmlns:fdt="x-
schema:FDTDataTypesSchema.xml" xmlns:fdthart="x-schema:FDTHARTCommunicationSchema.xml" xmlns:dtminfo="x-
schema:DTMInformationSchema.xml" xmlns:dt="urn:schemas-microsoft-com:datatypes">
    <!--Definition of Attributes-->
    <AttributeType name="schemaVersion" dt:type="number" default="1.21"/>
    <!--Definition of Elements-->
    <ElementType name="HARTDevice" content="eltOnly" model="closed">
        <attribute type="fdt:nodeId" required="no"/>
        <attribute type="schemaVersion" required="no"/>
        <element type="fdthart:LongAddress" minOccurs="0" maxOccurs="1"/>
        <attribute type="fdthart:manufacturerId" required="no"/>
        <attribute type="fdthart:deviceTypeId" required="no"/>
        <attribute type="fdt:subDeviceType" required="no"/>
        <attribute type="fdt:tag" required="yes"/>
        <attribute type="fdthart:shortAddress" required="no"/>
    </ElementType>
</Schema>
```

10 HART device identification

HART char array rules:

- In all strings with char ranges, the leading spaces are left trimmed. The char array is to be filled with 0x20h (blank).
- In VisibleStrings, invisible characters provided by a device have to be replaced by '?'

10.1 FDTHARTIdentSchema

The HARTIdentSchema provides elements and attributes with a protocol specific semantic (see Table 4 and Table 5) as well as document nodes without such a mapping (see Table 6).

Table 4 – FDTHARTIdentSchema – attributes with protocol specific mapping

HART Attribute	Semantic element name	Data request in physical device	Protocol specific name	HART data format	XML-FDT format (display format)	Specification Reference
shortAddress	IdAddress	Poll possible address range (HART5: [0-15], HART 6: [0-63]) by calling Cmd 0. If Cmd 0 response is available, a physical device is connected to this address. Cmd #0 response do not contain short address value whether short or long format is used. If master using short address for polling receives a response, it can assume that short address of device is the same as used in polling request. In addition to this, polling address can be read from HART 6 device with cmd #7	Polling Address	Unsigned 8	ui1	[1]† Chapter 6.8 Command 7 Read Loop Configuration
busProtocol	IdBusProtocol	CommChannel has to pass HART in this attribute	HART	Enumeration : “HART” for HART5 and HART6	enum	
universalCommandRevisionLevel	IdBusProtocolVersion	Command 0 Byte 4	HART Revision	8 bit unsigned integer	ui1 (dec)	[3] Chapter 6.1 Command 0 Read Unique Identifier

† Figures in brackets refer to the Bibliograpy.

HART Attribute	Semantic element name	Data request in physical device	Protocol specific name	HART data format	XML-FDT format (display format)	Specification Reference
manufacturerIdentificationCode	IdManufacturer	Command 0 Byte 1 – HART6: Manufacturer Identification Code HART 5: Manufacturer Device Type Code	Manufacturer Identification Code	8 bit unsigned integer Example : Endress+Hauser: 17 (0x11)	ui1 (dec)	[2] Manufacturer Identification Codes
deviceTypeID	IdTypeID	Command 0 Byte 2 – Manufacturers Device Type code	Device Type Code	8 bit unsigned integer	ui1 (dec)	[3] Chapter 6.1 Command 0 Read Unique Identifier
softwareRevision	IdSoftwareRevision	Command 0 Byte 6	Software Revision	8 bit unsigned integer	ui1	[3] Chapter 6.1 Command 0 Read Unique Identifier
hardwareRevision	IdHardwareRevision	Command 0 Byte 7	Hardware Revision	8 bit unsigned integer (mapped to float : xxxxx.yyy) First 5 bits (x) refers to HW revision level. Last 3 bits (y) to Physical Signaling Code	float	[3] Chapter 6.1 Command 0 Read Unique Identifier
tag	IdTag	Command 13 Bytes 0 - 5	Tag	6 Bytes or Packed ASCII characters	string	[3] Chapter 6.13 Command 13 Read Tag, Descriptor, Date
deviceID	IdSerialNumber	Command 0 Bytes 9 - 11	Device Identification Number	Unsigned 24	ui4	[3] Chapter 6.1 Command 0 Read Unique Identifier
N/A	IdDTMSupportLevel	Not applicable for scan / physical device. Attribute to be used only in DTMDeviceType identification. Enumeration: GenericDTM, ProfileDTM, BlockSpecificProfileDTM	DTM Support Level	–	enum	–

Table 5 – FDTHARTIdentSchema – attributes without protocol independent semantics

HART Attribute	Semantic element name	Data request in physical device	Protocol specific name	HART data format	XML-FDT format (display format)	Specification Reference
deviceCommandRevisionLevel	-	Command 0 Byte 5	Device Revision Level	8 bit unsigned integer	ui1 (dec)	[3] Chapter 6.1 Command 0 Read Unique Identifier
deviceFlag	-	Command 0 Byte 8	Flags	Bit value according Flag Assignment table. 8 bit – unsigned int	ui1 (hex)	[2] Chapter 5.11 Table Flag Assignments
manufacturerSpecificExtension		Can be used by DTM for a vendor specific device identification information, e.g. by combining a number of device parameter values into one string value. This can be used to identify a specific device variant			string	

Table 6 – FDTHARTIdentSchema – attributes and elements with protocol independent semantics

Attribute	Description
schemaVersion	Identifies the schema version
idDTMSupportLevel	enumeration genericSupport profileSupport blockspecificProfileSupport specificSupport
match	Used by device DTM to define a regular expression which must match to scanned physical define identification information
nomatch	Used by device DTM to define a regular expression which must not match to scanned physical define identification information. Used by device DTM to indicate if identification information may not match
Elements	**Description**
RegExpr	Includes regular expression string – either for match or for nomatch

```
<Schema name="FDTHARTIdentSchema" xmlns="urn:schemas-microsoft-com:xml-data" xmlns:dt="urn:schemas-microsoft-com:datatypes">
    <!--Definition of Attributes-->
    <AttributeType name="schemaVersion" dt:type="number" default="1.21"/>
    <AttributeType name="busProtocol" dt:type="enumeration" dt:values="HART"/>
    <AttributeType name="universalCommandRevisionLevel" dt:type="ui1"/>
    <!-- Command 0 Byte 4 -->
    <AttributeType name="shortAddress" dt:type="ui1"/>
    <!-- Poll to check Command 0 response -->
    <AttributeType name="manufacturerIdentificationCode" dt:type="ui1"/>
    <!-- Command 0 Byte 1 -->
    <AttributeType name="deviceTypeID" dt:type="ui1"/>
    <!-- Command 0 Byte 2 -->
    <AttributeType name="softwareRevision" dt:type="ui1"/>
    <!-- Command 0 Byte 6 -->
    <AttributeType name="hardwareRevision" dt:type="float"/>
    <!-- Command 0 Byte 7 - conversion: xxxxx.yyy-->
    <AttributeType name="tag" dt:type="string"/>
    <!-- Command 13 Bytes 0 - 5  -->
    <AttributeType name="deviceID" dt:type="ui4"/>
    <!-- Command 0 Bytes 9 - 11 -->
    <!--                    end of semantic information for HART                    -->
    <AttributeType name="deviceCommandRevisionLevel" dt:type="ui1"/>
    <!-- Command 0 Byte 5 -->
    <AttributeType name="deviceFlag" dt:type="ui1"/>
    <!-- Command 0 Byte 8 -->
    <AttributeType name="manufacturerSpecificExtension" dt:type="string"/>
    <AttributeType name="idDTMSupportLevel" dt:type="enumeration" dt:values="genericSupport profileSupport blockspecificProfileSupport specificSupport identSupport"/>
    <AttributeType name="match" dt:type="string"/>
    <AttributeType name="nomatch" dt:type="string"/>
    <ElementType name="RegExpr" content="empty" model="closed">
        <attribute type="match" required="no"/>
        <attribute type="nomatch" required="no"/>
    </ElementType>
</Schema>
```

10.2 FDTHARTScanIdentSchema

The schema defines attributes and elements, that are used to provide protocol specific scanning, see Table 7.

Table 7 – FDTHARTScanIdentSchema – attributes and elements

Attribute	Description
schemaVersion	Identifies the schema version
resultState	Identifies if the result is one of the provisional results or the final result of the split scan results
configuredState	A communication master must indicate in this attribute, if the scan response is related to a detected physical device which is configured or unconfigured

Tag	Description
IdAddress	All elements contain exactly one attribute each including the value of the scanned physical device. All elements with semantic meaning have a prefix “Id” for better identification
IdBusProtocol	
IdBusProtocolVersion	
IdManufacturer	
IdTypeID	
IdSoftwareRevision	
IdHardwareRevision	
IdTag	
IdSerialNumber	
DeviceCommandRevisionLevel	All elements without semantic prefix “Id” are transformed by XSL to name value pairs. These elements contain exactly one attribute defined in FDTHARTIdentSchema, each including one value of the physical device
DeviceFlag	
ManufacturerSpecificExtension	
ScanIdentification	Element contains all HART scan identification elements for one scanned physical device
ScanIdentifications	Collection of ScanIdentification elements

```
<Schema name="FDTHARTScanIdentSchema" xmlns="urn:schemas-microsoft-com:xml-data" xmlns:dt="urn:schemas-
microsoft-com:datatypes" xmlns:hartident="x-schema:FDTHARTIdentSchema.xml" xmlns:fdt="x-
schema:FDTDataTypesSchema.xml">
    <!--Definition of Attributes-->
    <AttributeType name="schemaVersion" dt:type="number" default="1.21"/>
    <AttributeType name="resultState" dt:type="enumeration" dt:values="provisional final error"/>
    <AttributeType name="configuredState" dt:type="enumeration" dt:values="configuredAndPhysicallyAvailable
configuredAndNotPhysicallyAvailable availableButNotConfigured notApplicable"/>
    <!--Definition of elements-->
    <ElementType name="IdAddress" content="empty" model="closed">
        <attribute type="hartident:shortAddress" required="yes"/>
    </ElementType>
    <ElementType name="IdBusProtocol" content="empty" model="closed">
        <attribute type="hartident:busProtocol" required="yes"/>
    </ElementType>
    <ElementType name="IdBusProtocolVersion" content="empty" model="closed">
        <attribute type="hartident:universalCommandRevisionLevel" required="yes"/>
    </ElementType>
    <ElementType name="IdManufacturer" content="empty" model="closed">
        <attribute type="hartident:manufacturerIdentificationCode" required="yes"/>
    </ElementType>
    <ElementType name="IdTypeID" content="empty" model="closed">
        <attribute type="hartident:deviceTypeID" required="yes"/>
    </ElementType>
    <ElementType name="IdSoftwareRevision" content="empty" model="closed">
```

```
        <attribute type="hartident:softwareRevision" required="yes"/>
    </ElementType>
    <ElementType name="IdHardwareRevision" content="empty" model="closed">
        <attribute type="hartident:hardwareRevision" required="yes"/>
    </ElementType>
    <ElementType name="IdTag" content="empty" model="closed">
        <attribute type="hartident:tag" required="yes"/>
    </ElementType>
    <ElementType name="IdSerialNumber" content="empty" model="closed">
        <attribute type="hartident:deviceID" required="yes"/>
    </ElementType>
    <ElementType name="DeviceCommandRevisionLevel" content="empty" model="closed">
        <attribute type="hartident:deviceCommandRevisionLevel" required="yes"/>
    </ElementType>
    <ElementType name="DeviceFlag" content="empty" model="closed">
        <attribute type="hartident:deviceFlag" required="yes"/>
    </ElementType>
    <ElementType name="ManufacturerSpecificExtension" content="empty" model="closed">
        <attribute type="hartident:manufacturerSpecificExtension" required="yes"/>
    </ElementType>
    <ElementType name="ScanIdentification" content="eltOnly" model="closed">
        <attribute type="configuredState" required="no"/>
        <!--    attributes with semantic meaning: -->
        <element type="fdt:CommunicationError" minOccurs="0" maxOccurs="1"/>
        <element type="IdAddress" minOccurs="1" maxOccurs="1"/>
        <element type="IdBusProtocol" minOccurs="1" maxOccurs="1"/>
        <element type="IdBusProtocolVersion" minOccurs="1" maxOccurs="1"/>
        <element type="IdManufacturer" minOccurs="1" maxOccurs="1"/>
        <element type="IdTypeID" minOccurs="1" maxOccurs="1"/>
        <element type="IdSoftwareRevision" minOccurs="1" maxOccurs="1"/>
        <element type="IdHardwareRevision" minOccurs="1" maxOccurs="1"/>
        <element type="IdTag" minOccurs="1" maxOccurs="1"/>
        <element type="IdSerialNumber" minOccurs="1" maxOccurs="1"/>
        <!--    non semantic attributes: -->
        <element type="DeviceCommandRevisionLevel" minOccurs="1" maxOccurs="1"/>
        <element type="DeviceFlag" minOccurs="1" maxOccurs="1"/>
        <!--    manufacturer specific extension, added in manufacturer specific identification -->
        <element type="ManufacturerSpecificExtension" minOccurs="0" maxOccurs="1"/>
    </ElementType>
    <ElementType name="ScanIdentifications" content="eltOnly" model="closed">
        <attribute type="fdt:busCategory" required="yes"/>
        <attribute type="resultState" required="yes"/>
        <element type="ScanIdentification" minOccurs="0" maxOccurs="*"/>
    </ElementType>
    <ElementType name="FDT" content="eltOnly" model="closed">
        <element type="ScanIdentifications" minOccurs="1" maxOccurs="1"/>
    </ElementType>
</Schema>
```

10.3 FDTHARTDeviceTypeIdentSchema

The schema provides attributes and elements for providing protocol specific information for device types, see Table 8.

Table 8 – FDTHARTDeviceTypeIdentSchema attributes and elements

Attribute	Description
schemaVersion	Identifies the schema version

Tag	Description
IdBusProtocol	All elements contain exactly one attribute, each including the value of the scanned physical device. All elements with semantic meaning have a prefix “Id” for better identification
IdBusProtocolVersion	
IdManufacturer	
IdTypeID	
IdSoftwareRevision	
IdHardwareRevision	
DeviceCommandRevisionLevel	All elements without semantic prefix “Id” are transformed by XSL to name value pairs. These elements contain exactly one attribute defined in FDTHARTIdentSchema, each including one value of the physical device
DeviceFlag	
ManufacturerSpecificExtension	
DeviceIdentification	Contains all HART specific identification elements
DeviceIdentifications	Contains a list of device identifications

```
<Schema name="FDTHARTDeviceIdentSchema" xmlns="urn:schemas-microsoft-com:xml-data" xmlns:dt="urn:schemas-microsoft-com:datatypes" xmlns:hartident="x-schema:FDTHARTIdentSchema.xml" xmlns:ident="x-schema:DTMIdentSchema.xml" xmlns:fdt="x-schema:FDTDataTypesSchema.xml">
    <!--Definition of Attributes-->
    <AttributeType name="schemaVersion" dt:type="number" default="1.21"/>
    <ElementType name="IdBusProtocol" content="eltOnly" model="closed">
        <attribute type="hartident:busProtocol" required="no"/>
        <element type="hartident:RegExpr" minOccurs="0" maxOccurs="*"/>
    </ElementType>
    <ElementType name="IdBusProtocolVersion" content="eltOnly" model="closed">
        <attribute type="hartident:universalCommandRevisionLevel" required="no"/>
        <element type="hartident:RegExpr" minOccurs="0" maxOccurs="*"/>
    </ElementType>
    <ElementType name="IdManufacturer" content="eltOnly" model="closed">
        <attribute type="hartident:manufacturerIdentificationCode" required="no"/>
        <element type="hartident:RegExpr" minOccurs="0" maxOccurs="*"/>
    </ElementType>
    <ElementType name="IdTypeID" content="eltOnly" model="closed">
        <attribute type="hartident:deviceTypeID" required="no"/>
        <element type="hartident:RegExpr" minOccurs="0" maxOccurs="*"/>
    </ElementType>
    <ElementType name="IdSoftwareRevision" content="eltOnly" model="closed">
        <attribute type="hartident:softwareRevision" required="no"/>
        <element type="hartident:RegExpr" minOccurs="0" maxOccurs="*"/>
    </ElementType>
    <ElementType name="IdHardwareRevision" content="eltOnly" model="closed">
        <attribute type="hartident:hardwareRevision" required="no"/>
        <element type="hartident:RegExpr" minOccurs="0" maxOccurs="*"/>
    </ElementType>
    <ElementType name="DeviceCommandRevisionLevel" content="eltOnly" model="closed">
        <attribute type="hartident:deviceCommandRevisionLevel" required="no"/>
        <element type="hartident:RegExpr" minOccurs="0" maxOccurs="*"/>
    </ElementType>
    <ElementType name="DeviceFlag" content="eltOnly" model="closed">
        <attribute type="hartident:deviceFlag" required="no"/>
        <element type="hartident:RegExpr" minOccurs="0" maxOccurs="*"/>
    </ElementType>
```

```
    <ElementType name="ManufacturerSpecificExtension" content="empty" model="closed">
        <attribute type="hartident:manufacturerSpecificExtension" required="yes"/>
    </ElementType>
    <ElementType name="DeviceIdentification" content="eltOnly" model="closed">
        <attribute type="hartident:idDTMSupportLevel" required="yes"/>
        <element type="IdBusProtocol" minOccurs="1" maxOccurs="1"/>
        <element type="IdBusProtocolVersion" minOccurs="1" maxOccurs="1"/>
        <element type="IdManufacturer" minOccurs="1" maxOccurs="1"/>
        <element type="IdTypeID" minOccurs="1" maxOccurs="1"/>
        <element type="IdSoftwareRevision" minOccurs="1" maxOccurs="1"/>
        <element type="IdHardwareRevision" minOccurs="1" maxOccurs="1"/>
        <element type="DeviceCommandRevisionLevel" minOccurs="1" maxOccurs="1"/>
        <element type="DeviceFlag" minOccurs="1" maxOccurs="1"/>
        <!--    manufacturer specific extension, added in manufacturer specific identification -->
        <element type="ManufacturerSpecificExtension" minOccurs="0" maxOccurs="*"/>
    </ElementType>
    <ElementType name="DeviceIdentifications" content="eltOnly" model="closed">
        <attribute type="fdt:busCategory" required="yes"/>
        <element type="DeviceIdentification" minOccurs="1" maxOccurs="*"/>
    </ElementType>
    <ElementType name="FDT" content="eltOnly" model="closed">
        <element type="DeviceIdentifications" minOccurs="1" maxOccurs="1"/>
    </ElementType>
</Schema>
```

BIBLIOGRAPHY

[1] HART 6 Spec127 Universal Commands

[2] HCF SPEC-183 Common Tables Specification

[3] HCF SPEC-127 Universal Command Specification

Standards Survey

The IEC would like to offer you the best quality standards possible. To make sure that we continue to meet your needs, your feedback is essential. Would you please take a minute to answer the questions overleaf and fax them to us at +41 22 919 03 00 or mail them to the address below. Thank you!

Customer Service Centre (CSC)

International Electrotechnical Commission
3, rue de Varembé
1211 Genève 20
Switzerland

or

Fax to: **IEC**/CSC at +41 22 919 03 00

Thank you for your contribution to the standards-making process.

Nicht frankieren
Ne pas affranchir

Non affrancare
No stamp required

A Prioritaire

RÉPONSE PAYÉE

SUISSE

Customer Service Centre (CSC)
International Electrotechnical Commission
3, rue de Varembé
1211 GENEVA 20
Switzerland

Q1 Please report on **ONE STANDARD** and **ONE STANDARD ONLY**. Enter the exact number of the standard: *(e.g. 60601-1-1)*

.....

Q2 Please tell us in what capacity(ies) you bought the standard *(tick all that apply).* I am the/a:

purchasing agent ❑
librarian ❑
researcher ❑
design engineer ❑
safety engineer ❑
testing engineer ❑
marketing specialist ❑
other.....

Q3 I work for/in/as a: *(tick all that apply)*

manufacturing ❑
consultant ❑
government ❑
test/certification facility ❑
public utility ❑
education ❑
military ❑
other.....

Q4 This standard will be used for: *(tick all that apply)*

general reference ❑
product research ❑
product design/development ❑
specifications ❑
tenders ❑
quality assessment ❑
certification ❑
technical documentation ❑
thesis ❑
manufacturing ❑
other.....

Q5 This standard meets my needs: *(tick one)*

not at all ❑
nearly ❑
fairly well ❑
exactly ❑

Q6 If you ticked NOT AT ALL in Question 5 the reason is: *(tick all that apply)*

standard is out of date ❑
standard is incomplete ❑
standard is too academic ❑
standard is too superficial ❑
title is misleading ❑
I made the wrong choice ❑
other .....

Q7 Please assess the standard in the following categories, using the numbers:
(1) unacceptable,
(2) below average,
(3) average,
(4) above average,
(5) exceptional,
(6) not applicable

timeliness.....
quality of writing.....
technical contents.....
logic of arrangement of contents .....
tables, charts, graphs, figures.....
other .....

Q8 I read/use the: *(tick one)*

French text only ❑
English text only ❑
both English and French texts ❑

Q9 Please share any comment on any aspect of the IEC that you would like us to know:

.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....

ISBN 2-8318-8644-9

ICS 25.040.40; 35.240.50; 35.100.05

Typeset and printed by the IEC Central Office
GENEVA, SWITZERLAND