EDDL Overview

Terry Blevins, Chairman SP104
for NeSSI Meeting
April 16th, 2007 Houston
Agenda

• Background on EDDL and IEC61804
• ECT Cooperation Project
• EDDL Components – Technical Overview
• Example Device Applications
• EDDL Development Tools for Device Manufacturers
What is EDDL? International Standard for Interoperability

- EDDL is an Electronic Device Description Language
- EDDL is an international standard
  - Standardized by IEC (IEC 61804-3)
- EDDL is endorsed by four major foundations
  - Fieldbus Foundation
  - HART Communication Foundation
  - Profibus Nutzerorganisation e.V (PNO)
  - The OPC Foundation
Position of the IEC 61804 series related to other standards

Abstract models
IEC 61499-1
ISO 15745-1

Conceptual FB specification for the process sector
Solutions (technology) profiles

IEC 61804-2
FB concept

IEC 61804-3
EDDL
IEC/TR 61804-4
Interoperability Guideline

Implementation
Products

FF FB application
PROFIBUS PA profile
Controlnet
FIP companion standards
EDDL Consistent with NAMUR NE105 Requirements

- √ 3.1 Investment Safety
- √ 3.2 Version Conflicts
- √ 4.1 Device Integration with Tools
- √ 4.2 User Guidance
- √ 4.3 Display of Devices
- √ 4.4 Standard Profiles
- √ 5.1 Device Descriptions
- √ 5.2 Licensing of Device Descriptions
- √ 5.3 Cross-Platform Compatibility
- √ 5.4 Full Support of Device Functionality
- √ 5.5 Standardized Data filing
- √ 6.0 Certification

FOUNDATION fieldbus DD technology meets NE105 requirements
IEC 61804-3 Standard

• This standard specifies EDDL as a generic language for describing the properties of automation system components. EDDL is capable of describing
  – device parameters and their dependencies;
  – device functions, for example, simulation mode, calibration;
  – graphical representations, for example, menus;
  – interactions with control devices
  – graphical representations
  – persistent data store.

• EDDL is to be used to create Electronic Device Description (EDD). This EDD is used with appropriate tools to generate interpretative code to support parameter handling, operation, and monitoring of automation system components such as remote I/Os, controllers, sensors, and programmable controllers.
• ISA standards committee that is working to adopt the generic device description language specified by IEC 61804 for device integration.

• The ISA SP104 committee has worldwide participation and is committed to harmonizing its ongoing work with the IEC SC65E WG7 that has created and is maintaining this standard.

• The committee has voted in October, 2006 to adopt the IEC 61804 standard as an ANSI/ISA standard and is committed to provide information that will help users and integrators fully utilize the EDDL interface to support a wide gamut of intelligent devices.
EDDL- The Standard for Device Integration

- 2007 Completing phase 2 and publishing through the organizations and IEC (including OPC-UA)
- 2006 IEC 61804-3 and 61804-4 approved
- 2004 EDDL Enhancement Cooperation Project
  Phase 2 and OPCF join the cooperation
- 2004 IEC 61804-2 approved
- 2003 EDDL Enhancement Cooperation Project
  Phase 1
- 2003 Standardization in CENELEC
- 2000 EDDL gets PNO standard
- 1997 first PROFIBUS devices are described with EDDL
- 1996 EDDL Standard in the Fieldbus Foundation
- 1992 EDDL Standard in the HART Communication Foundation
- 1990 EDDL definition in the International Fieldbus Group
- 1988 first intelligent HART devices
EDDL Acceptance in the Process Industry

- About 1800 devices from more than 100 manufacturers are described with EDDL.
- In plants, more than 16,000,000 devices are in use.
- Because of operating system independence, EDD’s from 1992 are still used without changes.
- EDDL is operating and automation system independent.
- EDDL source is stored in ASCII files that may contain UTF8 string constants.
- EDDL is mainly a descriptive language, but also allows conditions and c-like methods.
Host Applications Supporting EDDL

ABB - Industrial IT Freelance 800F
ABB - Industrial IT System 800xA
Emerson Process Management - 375 Field Communicator
Emerson Process Management - DeltaV
Emerson Process Management - Ovation
Endress+Hauser - ControlCare
Foxboro - I/A Series FoxCAE
Honeywell - PlantScape
Honeywell - Experion-PKS
National Instruments - NI-FBUS Configurator
Rockwell - Process Logix
Rockwell Automation – RSFieldBus
Siemens - PDM
Smar - System 302
Yamatake - Industrial-DEO
Yokogawa - CENTUM
Yokogawa - STARDOM

Updated April 2006
EDDL Approach

• EDDL technology was designed to avoid the need for special, proprietary, and operating system-specific host application files
• It allows a host system to both configure as well as monitor devices on-line
What is an EDD?

- An EDD is the computer readable file written in Electronic Device Description Language (EDDL) that describes the data in a field device.
- It is the file that the Host application reads in order to learn how to retrieve information from the field device.
EDDL: Operating System Independent

- Application on PC or Handheld uses the same EDD
- Fully backward compatible
EDD’s and Interoperability

EDD’s enable:
- Devices from different suppliers to interoperate with a single Host
- The same device to interoperate with different Hosts.

Describes
How the device functions per IEC 61804
Small ASCII files (< 200k)
EDDL Benefit #1

• EDDL / EDDs are Independent from:
  – Operating systems and versions
  – DCS Platforms
  – Communication and interface paths
EDDL Benefit #2 - EDDL is Easy to Use

• One tool for all devices
  – Common transparent data base
  – A new device just a new EDD

• Build in state of the art graphics
  – Trends, Bar graphs
EDDL Benefit #3 - No influence on the runtime stability

- There is no executable code with EDDs which may have an effect to the stability of the operating system
- EDDs are interpreted and therefore encapsulated
  - No impact of one EDD to others
  - Easy update and device additions during operation
EDDL Benefit #4  Easy to Learn

- EDDL is easy to learn
  - Like Visual Basic
  - Standard text editor
  - Context sensitive editors available
  - Existing EDDs can be used as a basis for a new device
  - Learning by doing
  - One EDDL – many languages for localisation

VARIABLE tag
{
  LABEL  "TAG";
  HELP   "[en]Text that can be used in any way."
  CLASS  DEVICE;
  TYPE   INTEGER(1)
}

Definition of a variable of a device

- Adapting an EDD to a new device version
  - normally means copy/paste and small adjustments
EDDL – Fits Best in All Environments

• EDDL is used
  – from Handheld
  – from simple devices

• to MES\(^1\)

• to very complex devices

\(^1\) Manufacturing Execution System
Central configuration of field devices
Benefits for End users and Vendors -Summary

<table>
<thead>
<tr>
<th>Independent from</th>
<th>Low Cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operating Systems</td>
<td>Education and training</td>
</tr>
<tr>
<td>DCS Platforms and Versions</td>
<td>Development</td>
</tr>
<tr>
<td>Communication Technology</td>
<td>Maintenance</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Easy to use</th>
<th>No Investment Risk</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unified user interface</td>
<td>Controlled life cycle management</td>
</tr>
<tr>
<td>One tool for all devices</td>
<td>Backward compatibility</td>
</tr>
<tr>
<td>Build in state of the art graphics</td>
<td>Stable and consistent standard</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Safe Operation</th>
<th>Scaleable</th>
</tr>
</thead>
<tbody>
<tr>
<td>no influence on the runtime stability</td>
<td>from Handheld to MES</td>
</tr>
<tr>
<td>Easy update and device additions during operation</td>
<td>from simple to complex devices</td>
</tr>
</tbody>
</table>
Agenda

• Background on EDDL and IEC61804
• ECT Cooperation Project
• EDDL Components – Technical Overview
• Example Device Applications
• EDDL Development Tools for Device Manufacturers
EDDL Cooperation Project

- Joint Fieldbus Foundation, PROFIBUS and HART Communication Foundation project to specify visualization and data storage management extensions
EDDL Cooperation Team (ECT) Principles

1. Existing EDDs in FF/HCF/OPCF/PNO shall continue to operate without requiring any modifications. All EDDL enhancements shall be backward compatible with existing EDDs.

2. The technical basis of EDDL enhancements shall be IEC 61804-3 and 4, plus the current work of the ECT.

3. All EDDL enhancements shall be operating system and platform independent.

4. All EDD shall be operating and platform independent. It must be possible to develop and test the EDD once for a given device and device revision.

5. All enhancements shall be traceable to requirements derived from the use cases.
Phase 1 Scope

- **Enhanced User Interface**
  - Parameter Organization
  - Images

- **Graphing System**
  - Support for Charts and Graphs to visualize complex data

- **Persistent Data Store**
  - Archive and retrieve data
  - Aids diagnostics executed by devices
Phase 2 – Joint Work With OPC Foundation

EDDL Cooperation

Fieldbus Foundation
HCF
OPCF
PNO

OPC Foundation Joins Industry Cooperation Project

EDD Selected as Data Structure for OPC Unified Architecture Specifications
January 2005
Phase 2 Scope

- Enhanced support for devices connected to process
  - Automation systems
  - Procedures - e.g. device setup and maintenance
- Enhanced access to data references in large
  - Databases and look up tables
- Extended access to product information
  - (e.g. contact, device classification, etc.)
- Information model for OPC UA
  - Includes device and EDD information
- Support of modular devices
Enterprise Application

Client applications using OPC UA server to have access to device information e.g. process data, product data and diagnostic information
Agenda

- Background on EDDL and IEC61804
- ECT Cooperation Project
- EDDL Components – Technical Overview
- Example Device Applications
- EDDL Development Tools for Device Manufacturers
A binary form of the source is stored in the Host or hand-help device.
What is a Device Description?

• A clear and unambiguous structured text description that precisely describes field device data to host systems.

• An EDD contains the following information about the parameters of a device:
  – Attributes like coding, name, engineering unit, write protection, how to display etc.
  – The arrangement of the parameters in a menu structure, names of menus and submenus.
  – Information about the relation of parameters to others.
  – Information about help texts and help procedures.
  – Information about necessary operating interactions (e.g. calibration), also called methods.
  – Information about visualization tools (i.e.: charts and graphs)
Language Structure and Key Words

Identification and Version Information
- MANUFACTURER, DEVICE_TYPE
- DEVICE_REVISION and EDD_REVISION

Data Description
- VARIABLES
  - LABEL
  - HELP
  - TYPE
  - CLASS
  - DEFAULT_VALUE
  - MIN/MAX_VALUE
  - UNIT
  - ACTIONS (METHODs)
- ARRAY, ITEM_ARRAY
- BLOCK, RECORD
- COLLECTION
- LIST
- FILE
- ...

User Interface Description
- MENU
- WINDOW, DIALOG
- PAGE, GROUP
- TABLE, GRID
- IMAGE
- CHART
  - hor. and ver. BAR
  - GAUGE
  - SCOPE, STRIP, SWEEP
- GRAPH YT, XY
- ACTIONS (METHODs)
- ...

Communication Description
- COMMAND
  - Data ordering
  - Bit-masks and -positioning
  - Upload and Download of offline and online configurations
  - Ordering of COMMANDs
  - Control of time conditions
  - Error handling and Error messages
  - Relative and absolute addressing...
EDDL is Declarative

```plaintext
RECORD __pv
{
    LABEL "|en|PV" ;
    MEMBERS
    {
        STATUS, __statuscontained_r ;
        VALUE, __floatcontained_r ;
    }
}
```

```plaintext
VARIABLE __statuscontained_r
{
    LABEL "|en|Status" ;
    HELP [statuscontained_help] ;
    CLASS CONTAINED & DYNAMIC ;
    TYPE ENUMERATED (1)
    {
        __FF_STATUS_VALUES
    }
    CONSTANT_UNIT [blank] ;
    HANDLING READ ;
    /* RESPONSE_CODES xxx ; */
}
```

```plaintext
VARIABLE __floatcontained_r
{
    LABEL "|en|Value" ;
    HELP [floatcontained_help] ;
    CLASS CONTAINED & DYNAMIC ;
    TYPE FLOAT
    CONSTANT_UNIT [blank] ;
    HANDLING READ ;
    /* RESPONSE_CODES xxx ; */
}
```

Value: 25.5 deg C
Status: Good
Example - Data Description

Parameter can be described with there label, help text, data type, min and max values, read/write handling, etc. The data definitions can be used in structures like BLOCK, RECORD, COLLECTION, ARRAY, LIST, FILE, etc. Any device model and data archives can be described with EDDL.

```plaintext
#define LINEAR 0

VARIABLE trans1_temperature_unit
{
  LABEL [digital_units];
  HELP [temperature_unit_help];
  CLASS CONTAINED;
  HANDLING READ & WRITE;
  TYPE ENUMERATED (2)
  {
    DEFAULT.VALUE 32;
    32. [degC], [degC_help]
    33. [degF], [degF_help]
    34. [degR], [degR_help]
    35. [Kelvin], [Kelvin_help]
    IF (trans1_sensor_type == LINEAR)
    {
      36. [mV], [mV_help]
      37. [Ohm], [Ohm_help]
      39. [mA], [mA_help]
    }
  }
}
```

References to text dictionaries allows to use common wording and translations.

Conditional expression allows to define e.g. value ranges, read/write handling dependent of any other parameters.
EDDL Methods

- Permits Device/User Procedures
- Interpreted ANSI “C” (limited) executed by host application – Not compiled Code.
- Secure access to User Interface and Device handled through pre-defined functions call “built-ins”
It's possible to describe

- Communication orders (using COMMANDs)
- Sorting of parameters
- Bit positioning and bit length of parameters
- Sorting of communication orders
- Read and write timeouts
- Error handling including user messages

Example of a communication order

```plaintext
COMMAND read_fb_ai_
{
  BLOCK function_block;
  INDEX 11;
  OPERATION READ;
  TRANSACTION
  {
    REQUEST
    {
    }
    REPLY
    {
      fb_ai_pv_upper_range_value,  
      fb_ai_pv_lower_range_value,  
      pres_primary_val_unit,       
      pres_decimal_point
    }
  }
}
```
EDDL Visualization Extensions

• Improved User Interface (UI) – DD Developer can describe screen layout
  – Enhanced MENU construct with screen layout attributes (e.g. dialog boxes)
  – New IMAGE construct to support images
  – New GRID construct for easy table entry

• Visualization Built-ins
  – MenuDisplay for enabling “Wizard-like” interface using enhanced Menus

![Diagram of an EDDL visualization interface]
EDD User Interface Description

Very simply hierarchies of menus, dialogs, windows, table views with parameter groups, images, graphs, charts, etc. can be created.

Online Configuration
Online Dialogs are reading data out of the device. E.g. process data will be continuously refreshed. The user can transfer his inputs on changeable parameter to the device.

Graphical Data Views
It very easy to define graphs or charts with different styles. Therefore in the EDD the ranges, unit, the data or data list and optional some additional information have to be defined.
Graphical Examples

Through the graphical possibilities user friendly user interfaces for complex devices.

Today a large set of different device are available with EDD
• Frequency Controller
• Switchgears
• Electrical, pneumatic and hydraulic Drivers
• Valve Positioners
• Close loop Controller
• Fluid and Gas Analyzer
• Sensors for
  • Temperature
  • Pressure
  • Density
  • Level
  • Flow
  • etc.
• Remote I/Os
• etc.

Example: Control Panel of an frequency controller

Direct device control

Dynamic visualization of control flow
EDDL Visualization Extensions

• Charts and Graphs – Enables graphical display of static and real-time (continuous) data
  – New CHART construct to define display characteristics
  – New SOURCE construct enables multiples curves on a CHART
  – New GRAPH construct to define display characteristics
  – New WAVEFORM construct enables multiple curves on a GRAPH.
  – New AXIS construct
Persistent Data Storage

• Improved Data Storage- Enables DD Developer to securely store data on the host
  – New FILE construct describes parameters that will be stored
  – New LIST construct is used with FILE to access specific parameters

• List Bultins (persistent storage)
  – ListInsert inserts an element into a list
  – ListDeleteElementAt deletes an element from a list
EDDL supports the definition of archives through persistent stored dynamic lists. Complex diagnostic analytics can be define that compares archived data with current data to get diagnostic information with a problem description, prognoses and hints to solve the problem.
Agenda

• Background on EDDL and IEC61804
• ECT Cooperation Project
• EDDL Components – Technical Overview
• Example Device Applications
• EDDL Development Tools for Device Manufacturers
EDDL Example Applications

- Charting – Enables graphical display of real-time (continuous) data from device
  - New CHART construct to define display characteristics
  - New SOURCE construct enables multiples curves on a CHART
  - New AXIS construct

- Graphing – Enables graphical display of static Y-t and XY data
  - New GRAPH construct to define display characteristics
  - New WAVEFORM construct enables multiple curves on a GRAPH
  - New AXIS construct

- Improved Data Storage- Enables DD Developer to securely store data on the host
  - New FILE construct describes parameters that will be stored
  - New LIST construct is used with FILE to access specific parameters

- Improved User Interface (UI) – DD Developer can describe screen layout
  - Enhanced MENU construct with screen layout attributes (e.g. dialog boxes)
EDDL Capabilities – Temperature Example

Device DD File

Dialogs

Group Boxes

Parameter Organization

Window
Enhanced MENUS and METHOD are used to build dialog boxes displaying motor starts, operating hours, number of overload trips, etc.

Image

- Static bit map
- Objective is to provide a visual representation of the parameters
A GRAPH is used to present the echo WAVEFORM to enable configuration of thresholds and false echoes areas in the device.

- Trigger device to build WAVEFORM data
- Retrieve WAVEFORM data
- Update the GRAPH
A GRAPH is used to present the Valve Signature (Hysteresis) WAVEFORM as a measure of the air pressure to stroke the valve open and close.

- Trigger device to build WAVFORM data
- Retrieve WAVFORM data
- Update the GRAPH

GRAPH

WAVFORM (Data from Device)

AXIS

MENUS & METHODS (Enhanced UI)

FILE/LIST (Persistent Data)

ARRAY(s) (Device Data)

EDDL Capabilities - Valve Signature Example
A CHART is used to present the Real-time (continuous) Step Response SOURCE of a valve.

- Trigger device to build
- SOURCE data
- Retrieve SOURCE data
- Update the CHART

**MENUS & METHODS** (Enhanced UI)

**FILE** (Persistent Data)

**ARRAY(s)** (Device Data)

**SOURCEs** (Stored Data and Data from Device)

**AXIS**

**Valve Step Response Diagnostics**

- Travel (From device)
- Setpoint (Stored)
Example - SIMATIC PDM Parameter Online - View

- EDD-based Views
- Central common View (same look and feel for all Devices)
- Contents are defined in the Device description (EDD).
Example - Emerson Process Management AMS and 375
Presentation of Device Information

• The *look and feel* of the User Interface is determined by the Host System
  – All devices on a given Host system will have the same look and feel.
  – Necessary for efficient utilization by operator and maintenance personnel.
  – The same field device will have a different look and feel on each Host system.

• The *detailed information* of the Device is still determined by the Device Manufacturer in the EDD
EDDL - Consistent Look & Feel for a given Host
### Example: Rosemount 5400 Radar Level EDD on 4 different Hosts

The *look and feel* of the *User Interface* is determined by the Host System.

<table>
<thead>
<tr>
<th>Host</th>
<th>Siemens PDM</th>
<th>HCF SDC625</th>
<th>Emerson AMS</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Process</strong></td>
<td><img src="image1" alt="Screenshot" /></td>
<td><img src="image2" alt="Screenshot" /></td>
<td><img src="image3" alt="Screenshot" /></td>
</tr>
<tr>
<td><strong>Variables</strong></td>
<td><img src="image4" alt="Screenshot" /></td>
<td><img src="image5" alt="Screenshot" /></td>
<td><img src="image6" alt="Screenshot" /></td>
</tr>
<tr>
<td><strong>Trending</strong></td>
<td><img src="image7" alt="Screenshot" /></td>
<td><img src="image8" alt="Screenshot" /></td>
<td><img src="image9" alt="Screenshot" /></td>
</tr>
<tr>
<td><strong>Geometry</strong></td>
<td><img src="image10" alt="Screenshot" /></td>
<td><img src="image11" alt="Screenshot" /></td>
<td><img src="image12" alt="Screenshot" /></td>
</tr>
<tr>
<td><strong>Echo</strong></td>
<td><img src="image13" alt="Screenshot" /></td>
<td><img src="image14" alt="Screenshot" /></td>
<td><img src="image15" alt="Screenshot" /></td>
</tr>
<tr>
<td><strong>Tuning</strong></td>
<td><img src="image16" alt="Screenshot" /></td>
<td><img src="image17" alt="Screenshot" /></td>
<td><img src="image18" alt="Screenshot" /></td>
</tr>
</tbody>
</table>
Agenda

- Background on EDDL and IEC61804
- ECT Cooperation Project
- EDDL Components – Technical Overview
- Example Device Applications
- EDDL Development Tools for Device Manufacturers
HART EDDL Technology – Development Tools

DD-IDE Integrated Environment for DD Development

Description and Characteristics
- A new and comprehensive DD development environment
- Developed, administered, and fully endorsed by the HCF
- Completely open and available to all HCF member companies
- Fully supports all HART devices - HART 6.0 and 5.x

Benefits
- DD development is more efficient and cost effective
- DD’s are more robust, compliant, consistent, and well behaved
- End users and system suppliers can trust that the DD is fully compliant
Efficient DD Development

Device vendors use IDE to develop their DD’s quickly and cost effectively

- **Step-1:** Vendor develops the device per protocol specifications
- **Step-2:** Using IDE the vendor develops a compliant DD for the device
- **Step-3:** SDC-625 is used to validate the device and the DD

**DD-IDE**
- DD Development Software
- DD Editor
- Device simulator
- DD Library & sample DD’s
- Tokenizer
- SDC-625 REFERENCE HOST
SDC-625 for DD Development & Test Cycle

Device developer tests his DD using SDC-625

- Validates look and feel
- Validates display navigation
- Validates information content
- Tests all configuration permutations
- Tests all methods
- Validates full device operation
- Validates status reporting

Device quality:
- Complies with all specifications
- Device and DD registered at HCF

DD quality:
- DD-IDE compliant
- SDC-625 validated
HART SDC625 – Example Interface Display
The DD-IDE and SDC-625 Technology

Major Benefits & Success Factors

- Developed and endorsed by HCF!
- Available for use by all HCF members!
- Open and consistent standard!
- Easy and economical for all users!

Result... Robust and Reliable Universal configuration tools and Asset Management solutions for all HART devices that use only the DD’s Registered with HCF to fully support:
- All Device Functionality
- Universal commands
- Common Practice commands
- Device Specific commands
Fieldbus Foundation - EDDL Products

- DD Services v5 for Host Applications
- DD Tokenizer v5 for Device Developers
- DD “Super” Viewer with full extension support (Development Support)
- Testing and Registration supported with Interoperability Test Kit (ITK) 5
FF Device Testing and Registration Field Devices

- **DD Tokenizer (Offline Test)**
  - Validates the DDL Syntax
  - Enforces protocol specific rules
  - Enforces profiles and restrictions from FF-901

- **Interoperability Test Kit (Online Test)**
  - Validates consistency between DD, CFF and Device
Example – FF Testing and Registration

H1 Device

H1 Device

AT-420
H1 Interoperability Testing Kit (H1 ITK)

AT-410
H1 Conformance Testing Kit (H1 CTK)

AT-400
DD Tokenizer

DD Source

Function Blocks Test

CFF

H1 Device

H1 CTK Tests Run at Fraunhofer

H1 ITK Tests Run at Fieldbus Foundation

Vendor executes Physical Layer Test

Report issued to Device Manufacturer

Registered H1 Device

FF-830
H1 Physical Layer Compliance

Example – FF Testing and Registration
Fieldbus Foundation - EDD IDE and Viewer
Today, the PNO EDDs are tested by test laboratories. The EDDL syntax of the EDDs are tested with a EDD test tool and the tool EDD compiler.

PNO has created a TC 1 WG 10 for EDD certification to specify the test requirements and test procedures.

EDD test requirement specification is ready to start the PNO review. Within September the review will be completed. PNO EDD certification will start end of this year.

The host certification will be following.
Today the EDD applications including the EDD library. The library includes standard include files, dictionaries and profiles.

Since June 2005 enhanced EDDs are developed by the different manufacturer for their PROFIBUS devices.

The EDD library will be available on the PNO internet server.
Where to Get More Information about EDDL

- IEC61804 Web site

- SP104 Web site
  http://www.isa.org/MSTemplate.cfm?MicrositeID=1170&CommitteeID=6927

- Fieldbus Foundation
  http://www.fieldbus.org/index.html

- HART Communications Foundation
  http://www.hartcomm2.org/

- Profibus Nutzerorganisation e.V (PNO)
  http://www.profibus.com/