

Device Revision and Lifecycle Management Guide

How to keep systems up to date and compatible with new devices using EDDL

Background

The challenge of device revision management has existed since the very first smart devices.

Plant Challenge: Device Revisions

Devices get better and better. The challenge for a new project and over the life cycle of an operational plant is managing different versions¹ for devices using digital communication. More than one version of the device may arrive on site, from the manufacturer's different locations, reseller's stock, as part of package units, or as ongoing replacements. Managing all these device types and versions is a challenge. New types of HART, FOUNDATION fieldbus, PROFIBUS, and WirelessHART devices and new versions of devices are made available from hundreds of manufacturers every week.

Solution: EDDL

EDDL (IEC 61804-3) is a device integration technology created as a solution to the device revision problem. EDDL makes managing devices of different types and versions easier. EDDL² is a file that is loaded onto the computer or handheld field communicator. There is no EDDL inside the device³ itself. Thus there is no such thing as an “EDDL device” or a “non-EDDL device”. Devices are HART, FOUNDATION fieldbus, PROFIBUS, and WirelessHART devices, and these protocols support EDDL.



EDDL works the same way for HART, WirelessHART, FOUNDATION fieldbus, and PROFIBUS

The Electronic⁴ Device Description Language (EDDL), formerly known as just Device Description (DD), has been supporting device revision management since circa 1992. EDDL is a *device integration technology*. EDDL is an integral part of HART, FOUNDATION fieldbus, PROFIBUS, and WirelessHART protocols. EDDL is mainly associated with FOUNDATION fieldbus because it is a necessity to make the system work. However, EDDL is also used for device integration with the other bus technologies whenever detail setup or diagnostics is done. 4-20 mA works without EDDL, the system gets the process variable, but HART requires EDDL for detailed diagnostics, sensor calibration trim, and complete setup.

¹ HART, FOUNDATION fieldbus, PROFIBUS, and WirelessHART literature use the term “revision” when speaking of different versions. In this guideline the terms “revision” and “version” are used interchangeably.

² Technically EDDL is a language used to write an EDD (file) describing a device just like HTML is a language used to write an HTM file describing a web page but common industry practice has become to say EDDL-file just like we say HTML-file

³ FOUNDATION fieldbus does provide the ability to store the EDDL file in the device so that it can be uploaded in the system, but this feature is currently not used

⁴ Some literature erroneously state ‘E’ in EDDL means enhanced, but this is incorrect. Similarly, the abbreviation “eEDDL” for enhanced EDDL should also not be used.

If the system is not updated in lock-step with new devices and revisions, it will not be able to communicate with these new devices once they are installed in the plant. However, the system manufacturer cannot possibly update and test system software as fast as device manufacturers release new devices. The solution provided through EDDL is that the device manufacturer supplies the EDDL file for their device. Every version of every device type from every manufacturer has its own unique file. In the plant, the technician then loads the EDDL file onto the system or handheld, the system and handheld are now able to communicate with the new device without making changes to the software.

The purpose of EDDL is to ensure there is no need match software version to device version

That is, thanks to EDDL the system is kept evergreen and up to date with new devices. The problem of managing device revisions is solved.

When purchasing a system, software, or handheld field communicator it comes with most EDDL files pre-loaded. However, new device revisions become available every month. If a new device is purchased months later, the EDDL file for it must be loaded by plant personnel.

In order for a DCS, intelligent device management software as part of an asset management solution, or a handheld field communicator to communicate with a device it needs to know what to transmit, what it receives, and what to display to the technician. The EDDL file for the device provides this function. The EDDL file tells the system which commands to send to the device and how to interpret the response from the device. Thanks to the EDDL file, any system or handheld can communicate with any type of device from any manufacturer and display the information.

That is, keep systems up to date and compatible with new devices by loading the EDDL file for the device onto system

EDDL is the solution to device revision management in that all versions of all device types from all manufacturers can thanks to EDDL be managed from the same software

There are other technologies to display setup and diagnostics, but EDDL is the simplest and most robust solution. Other device integration technologies have drawbacks which EDDL avoids.

Basics

Device revision management is all about keeping the system and handheld field communicator up to date with the latest devices.

System and Handheld Update

The EDDL file can be obtained from one of many sources: device manufacturer website, HART Communication Foundation website, Fieldbus Foundation website, PROFIBUS International website, or system manufacturer website. Typically it is best⁵ to obtain the EDDL file through the manufacturer of the system used (the EDDL file is created by the devices manufacturer, but system manufacturers typically distribute them to their users). Normally the EDDL files are downloaded from a website, but are so small they can also be emailed as attachments. Because they are small they can also be downloaded across a poor Internet connection without suffering time-outs. In some case whole “libraries” of multiple EDDL files are provided on CD/DVD or through Internet on-line update. EDDL files can be copied and pasted from one system to another. This makes managing device revisions to keep the system up to date easy.

⁵ Systems typically also use system-specific files created uniquely for each device type by the systems manufacturer for their system for alarm management etc. which can be obtained together with the standard EDDL files for the device.

The device revision⁶ of the EDDL file to be loaded on the system shall match the device revision of the device. When the system or handheld encounters a new device for which it does not have the corresponding EDDL file with the correct device revision, the system will prompt the technician which EDDL file it needs the technician to obtain and load. This makes getting the right file easy.

In A Nutshell

If the plant receives a new revision of a device, obtain the EDDL file for that device revision and load on the system.

For instance, if the plant receives a revision 7 device, get an EDDL file for device revision 7.

The EDDL files are copied into folders and sub-folders for each manufacturer and device type. Most systems have wizards that automatically transfer new EDDL files into its correct folder. This eliminates mistakes and makes keeping the system up to date easy.

Managing device revisions is very simple, but many get confused by "device revision" and "DD revision"

Upgrading an old system from traditional DD technology to EDDL technology does not require change of skills. Just copy and paste EDDL files like HART and Fieldbus traditional DD files are copied today, no software installation has to be learnt.

Glossary

Just like any technology, EDDL has its own set of terminology and acronyms

Terminology

Device Revision	Version of the device
DD Revision	Version of the EDDL file itself, the user interface
Capability File Revision	Version of the capability file
Capability File	A file used for FOUNDATION fieldbus devices declaring resources and other characteristics in the device required for off-line configuration.
Forward compatibility mode	When system does not have the EDDL file matching the connected physical HART device's revision, it uses an earlier lower version EDDL file in its place.
Device Management	Configuration/setup, sensor calibration trim, and diagnostics etc. of field devices.
Plant Asset Management	Machinery health management, processing equipment management, and intelligent device management

In this guideline the terms “revision” and “version” are used interchangeably.

Acronyms

DD	Device Description (old abbreviation and term, now replaced by EDDL-file)
EDDL	Electronic Device Description
CFF	Capability File Format
GSD	Geräte Stamm Datei
HART	Highway Addressable Remote Transducer

⁶ There is also something which is called “DD revision” for the EDDL file, but this is largely uninteresting to the system and therefore not stated. In most instances the term “DD revision” is used erroneously (refer to glossary).

DDL	Device Description Language (old abbreviation and term, now replaced by EDDL)
EDD	Electronic Device Description (same as EDDL-file)
HTML	HyperText Markup Language
DCS	Distributed Control System

References

EDDL Brochure and Technical Description on www.eddl.org site

Jonas Berge, "Fieldbuses for Process Control: Engineering, Operation, and Maintenance", ISA, 2002, ISBN 1-55617-760-7

NAMUR NE 105, "Specifications for Integrating Fieldbus Devices in Engineering Tools for Field Devices", Version: 24.08.2004

Appendix 1 How Revision Management Works

A new revision device has new features such as new function blocks, new parameters (setup, diagnostics, monitoring), new options (units, sensor types, etc.). The system must know the features of every revision of every device type in order to make all of the features available to the technician.

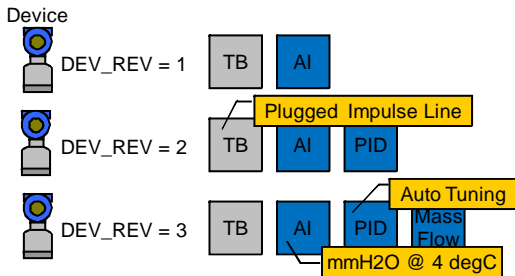


Figure 1 Devices keep getting better and better with new features and options

Every revision of every type of device has a unique EDDL file. This is because by definition each revision of a device has some new feature. It must have a new EDDL file so the device manufacturer can describe how this feature is communicated and how it shall be displayed in the system.

Version Management (Device)

The EDDL file lets the system know the features of each device. Therefore the system requires an EDDL file to be loaded for each device to describe its features.

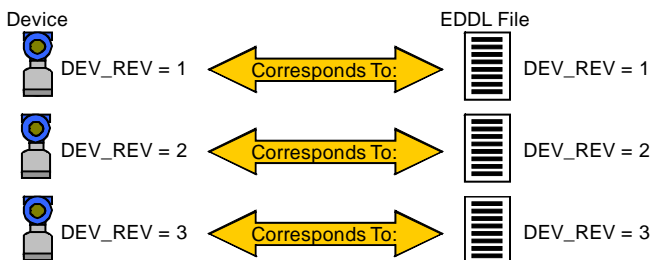


Figure 2 Each device version has a corresponding EDDL file

An important concept in device revision management using EDDL files is that there is no tampering with what is already working. Existing EDDL files are not removed or overwritten, just add new files. For instance, when a new revision 2 device comes to the plant, the existing EDDL file for device revision 1 is still left on the system. A new EDDL file for device revision 2 is simply added. That is, EDDL files for all device revisions remain on the system. There is no risk of an existing working file being overwritten by a new file which is not working. This ensures a new device does not break the operation of an old device. Old and new devices can operate together in the same system. There is no risk of breaking what is working. EDDL makes device revision management easy and maintains system robustness.

When EDDL file for a new revision device is added, the existing files are not touched, they continue to work as is. What is running is not upset. Old and new versions can coexist.

As far as device revision management and EDDL files is concerned, device "firmware version" is irrelevant. Usually when there is a new firmware version for the device there is also a new device revision, but it may not always be the case since the firmware change may not be related to communication. That is, do not confuse documentation by including firmware version, stick to device revision. Moreover, some devices have a separate communication board sandwiched

together with the main processor board in the device. Device revision reflects the firmware version of communication board, not the main board. Again, stick to device revision.

Version Management (EDDL file itself)

The device manufacturer may also choose to create a new EDDL file even though there is no change in the device, for instance to improve the device user interface with easier to use wizards, more intuitive menus, better graphics, and conditionals etc. When there is a new EDDL file for the same device revision, this is called a new “DD revision”

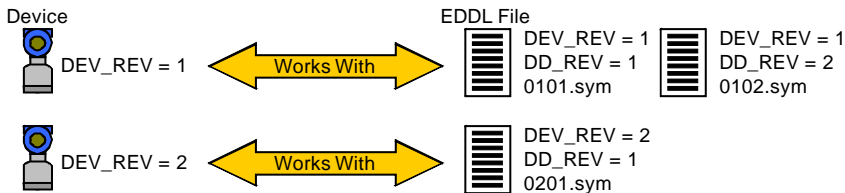


Figure 3 The device manufacturer may improve the user interface without changing the device

Common Misunderstanding

Many assume that DD revision of the file must equal device revision of the device, but that’s NOT how it works. “What is the DD revision of the device?” is generally the wrong question.

Note that when device revision is incremented (new version of the device), the DD (file) revision starts over from 1.

Table 1 DD revision restarts from 1 for new device revisions

Device Revision	DD Revision	Remark
01	01	Original
02	01	New device revision
02	02	New EDDL file revision
03	01	New device revision

System software is not concerned with DD revision (it’s the device revision which really matters). The system will simply use the highest DD revision loaded on the system.

An increment in the DD revision means the device itself is still the same; there is no change in the device hardware or firmware. It is only the EDDL file itself which has changed (better user interface).

FOUNDATION fieldbus devices have a DD_REV parameter (in the resource block) stating the minimum compatible DD file version. A field device that has a DD_REV parameter of 01 will work with any EDDL file for that device. A field device that has a DD_REV parameter of 03 will only work with EDDL files that are DD revision 3 or greater. Usually this parameter is 01 so this does not present a problem in the field.

In summary, a device does not have a DD revision; the device has a device revision. DD revision is the version of the file. Therefore DD revision need not be stated in instrument specification sheets when purchasing.

There has been some confusion on DD revision and device revision in the past. Therefore some documents erroneously state "DD revision" although it is evident from the context they really mean device revision.

Table 2 User interface version is independent of device version

Device Revision	DD (file) revision
Version of the device	Version of the user interface
Incremented when device firmware is improved with new features: <ul style="list-style-type: none"> • New blocks • New parameters • New options • New diagnostics 	Incremented when the user interface is improved with new features: <ul style="list-style-type: none"> • Better graphics • Smarter wizards • More helpful conditionals • More intuitive menus

If device firmware will be upgraded, make sure to have the new EDDL file on hand first.

The Device Revision for the EDDL file must match the device revision of the actual device, otherwise the system cannot work with the new device. The system does not care about the DD (file) revision; it picks the latest available EDDL file for the device.

Understanding EDDL File Names

The EDDL file names consist of:

- Device revision [2 digits]
- DD (file) revision [2 digits]

Note that these numbers are in hexadecimal

The advantage of this naming scheme is that each version of the device (and the device description) has an individual file with a unique name. When the file for a new version device is added, the files for older versions remain untouched. Files are not replaced or removed. Old and new devices can coexist in the system. Working files are not removed, established relationships are not broken. The file naming convention (and folder hierarchy) enables the system to automatically pick the correct EDDL file for a device, without having to manually associate a new device with its file.

Table 3 DD file nomenclature

	Example 1	Example 2
File Name	0701.sym	1703.sym
Interpretation	07 = Device Revision 7 01 = DD File Revision 1	17 = Device Revision 23 (17 in hexadecimal is 23 in decimal) 03 = DD File Revision 3

Each protocol use a different file extension for the system software to identify them but as far as humans are concerned the files are all managed the same way. This makes managing EDDL files consistent and easy. In fact, the file extension is a good way to tell if the file is for the HART protocol or FF protocol version of the same device.

HART

HART device files are identified with the following extensions:

- *.fms (Traditional)
- *.sym (Traditional)
- *.fm6 (Enhanced)

With *.fm6 and *.fm8, a separate *.sym file is no longer required

FOUNDATION fieldbus (FF)

FF device files are identified with the following extensions:

- *.ffo (Traditional)
- *.sym (Traditional)
- *.cff
- *.ff5 (Enhanced)
- *.sy5 (Enhanced)

‘Incremental DD’ is a scheme of using multiple partial DD files. However, multiple DD files are more difficult to manage for both device manufacturers and system administrators so this scheme is not recommended.

Understanding Capability File Names

The FF device capability file names consist of:

- Device revision [2 digits]
- DD file revision [2 digits]
- CF revision [2 digits]

Note that these numbers are in hexadecimal

Table 4 Capability file nomenclature

	Example
File Name	070101.cff
Interpretation	07 = Device Revision 7 01 = DD File Revision 1 01 = Capability File Revision 1

Note that when device revision is incremented (new version of the device), the capability file revision starts over from 1. However, when DD (file) revision is incremented (new version of the EDDL file), the capability file revision does not start over. That is, the earlier capability file continues to be used with the new EDDL file.

Table 5 Previous capabilities files can be used newer DD files

Device Revision	EDDL file name	CF file name	Remark
01	0101	010101	Original
01	0101	010102	New capability file
01	0102	010202	New EDDL file

Understanding EDDL Folder Names

Software and handheld field communicators store EDDL files in folders and sub-folders for each protocol, each manufacturer, and each device type. Systems have wizards that automatically copy the EDDL files into the correct folder.

If need be, it is relatively easy for a human to find the location of an EDDL file associated with a particular device on the computer. In the computer file system, each manufacturer folder is identified by a hexadecimal number (6 digit manufacturer ID, see appendix 6). The manufacturer folder in turn has sub-folders for each device also identified by hexadecimal number (4 digit device type), inside which the EDDL files are stored. The system software has a device library that displays all the available manufacturers and device types as plain text, i.e. names and model numbers recognizable by humans. The manufacturer IDs are not the same for the different protocols, but for the system administration purposes this does not matter since system software's device library utility shows manufacture names in plain text, not as ID.

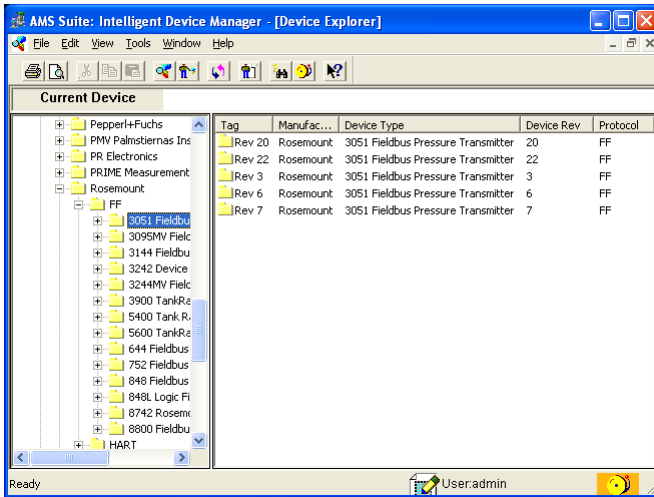


Figure 4 In intelligent device management software devices appear in plain text rather than codes

The folder hierarchy (and file naming convention) enables the system to automatically pick the correct EDDL file for a device when the new device is connected, without having to manually associate a new device with its file.

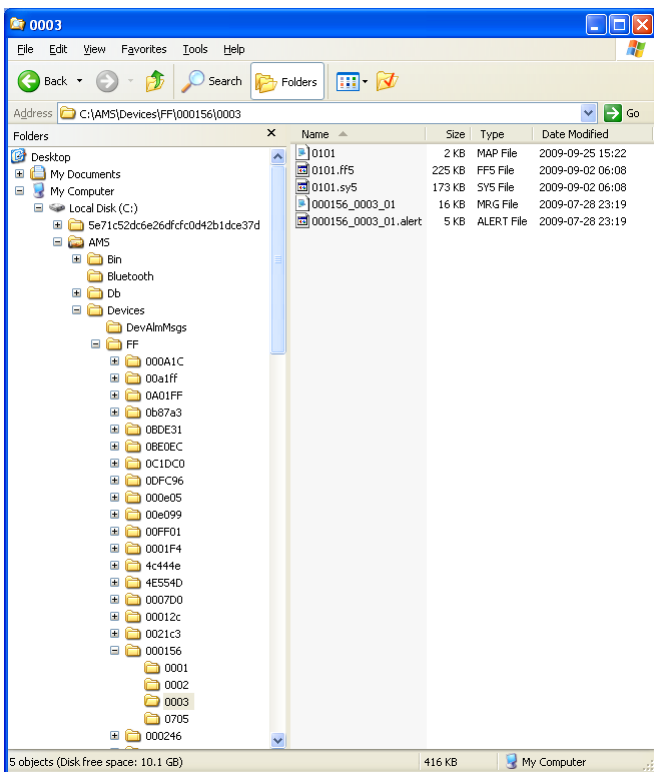


Figure 5 In Windows Explorer files and folders appear with their coded names

Note that since EDDL files are compressed text, not software, they are copied like documents, not installed like software programs. This has many benefits. First, 'administrator' level password need not be given to every technician that need to be able to commission a new type or version device. This has a cyber security advantage since lower level passwords can instead be given, for which software cannot be installed on the system. The second benefit is that conflicts with drivers and programs are avoided as DLL files and other executables are not installed and no Windows registry changes are made. EDDL files can easily be removed completely if need be. That is, since an EDDL file is text, not software, this makes it very much easier for the system administrator to support a system based on EDDL.

Plug-'n'-Play

EDDL has auto-discovery to automatically associate all the hundreds or thousands of devices in the plant with their respective EDDL file. This eliminates a lot of configuration work to associate a new www.eddl.org

device with its EDDL file. The system auto-detect devices and read manufacturer ID, device type, and revision, then automatically associate the device with a particular EDDL file, which is automatic since the EDDL file is named according to the version and is located in a folder and sub-folder named corresponding to manufacturer ID and device type. That is, not only automatically detect a device but to also automatically associate the device to its EDDL file; plug & play.

Grappling with a second device integration technology in addition to EDDL comes at a high price of both time and money. Using only EDDL avoids these issues reducing cost and saving time.

Appendix 2 Troubleshooting

EDDL files themselves do not fail, they are compressed text files, not executed software programs. Also, since they are copied onto the system, not installed like driver software, they do not cause DLL and other executable software conflicts or Windows registry issues. However, the software that use EDDL files may show prompts related to device revision management (EDDL files).

DD Missing

If a device is connected to the system for which the EDDL file has not been loaded, the system will tell the technician the device revision so the technician can go get the right EDDL file to load. That is, if a device with device revision 02 is connected and the EDDL file for device revision 02 has not been loaded, an error will be displayed.

EDDL is only required to access the configuration of the blocks and parameters in the device like diagnostics and setup. The system is able to detect the presence of a device without requiring EDDL.

A FOUNDATION fieldbus device can be detected but cannot be commissioned without its matching EDDL file. Make sure to have the EDDL file available before upgrading or replacing a device. The device requires the file to be loaded onto the system.

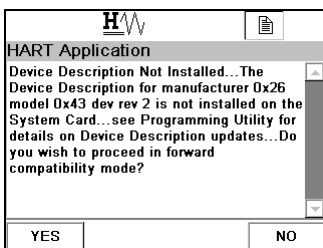


Figure 6 When EDDL file is missing system notifies which file is required

The 4-20 mA signal of a HART device is available even if there is no EDDL file, and it may be possible to change range, unit, and damping, as well as check on transfer function, write protection status, and sensor limits plus running self test and loop test (fixed current forced output simulation) wizards for the device by operating it in generic mode or forward compatibility mode, but detail setup, sensor calibration trim, and diagnostics is not possible. Basic process variable monitoring is also possible.

DD Mismatch Error

If the system database tag (place holder) is created for a device revision different from the device revision of the instrument connected, the system will not permit the device to be commissioned to that tag. The database configuration has to be modified. For instance, if the system database tag was created for device revision 02 but a device with device revision 03 is connected, it cannot be commissioned.

For a FOUNDATION fieldbus device upgrade, the DCS configuration has to be modified for the new device revision. Depending on the system this can be a simple operation. The intelligent device

management software in the system will generally not require a configuration change since it just reads data online.

For HART device upgrade, the DCS configuration generally need not be changed as the DCS does not use device-specific HART features. The exception may be smart logic solvers which monitor device type changes at replacement to maintain the integrity of the safety instrumented function. The intelligent device management software in the system will generally not require a configuration change since it just reads data online.

Device is not detected

EDDL files are not required to detect the presence of a device. If a device is not detected, it is not a problem with EDDL.

Generic HART Mode

If no EDDL file for a HART device is available, the technician may be given the option to operate the device in generic HART mode. This means that some basic functions of the device can be accessed. These are functions associated with universal commands and some common practice commands. The information is presented with generic graphics.

In generic HART mode it is possible to monitor the process variable in engineering unit and percentage of range, as well as the output current.

The generic HART mode provides identification information including manufacturer name, hardware and software revision, descriptor, message, date, final assembly and sensor serial numbers, tag, Device ID, universal and device specific command revision.

Polling address and burst mode can be configured in generic HART mode. Write protection status, number of requested preambles, and physical signal type can be checked.

It is possible to configure range, unit, and damping. Transfer function and analog output device alarm can be checked along with sensor limits including minimum span.

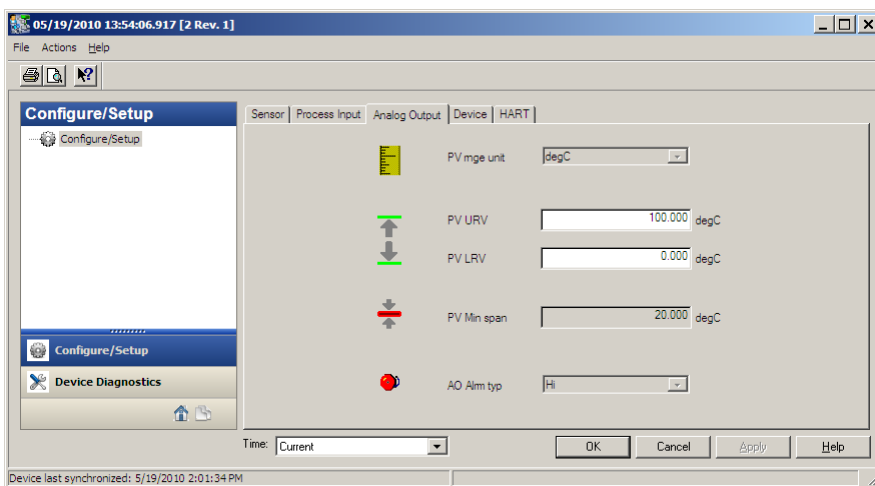


Figure 7 Very basic setup, diagnostics, and monitoring is possible without EDDL file

Wizards available in generic HART mode includes output current trim, self diagnostics test, and loop test (fixed current forced output simulation).

Diagnostics in generic HART mode is limited to PV out of limits, non-PV out of limits (for instance sensor temperature), analog output saturated (below 4 mA or above 20 mA), analog output fixed (simulation loop test mode), cold start (power lost and reapplied), configuration changed, and general device malfunction.

Full sensor calibration trim, setup, and advanced diagnostics requires EDDL file to be loaded.

Forward Compatibility Mode

If the correct device revision of the EDDL file for a device is not available, the technician may be given the option to operate the device in forward compatibility mode. This means the system will use an EDDL file for an earlier device revision to communicate with the device. Not all features of the new device will be accessible.

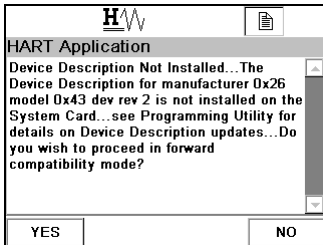


Figure 8 When EDDL file is missing system may provide an option to operate in forward compatibility mode

Appendix 3: System Files

EDDL files are device specific, written by the device manufacturer according to the IEC 61804-3 standard. Some systems may use additional files which are system specific (non-standard). In this case, system suppliers additionally provide system files for each device, supporting for instance to the system's management of device alerts - mechanism which is different in every system. This could be for an alarm filtering software running in the background of the system monitoring all devices, which is independent of EDDL (or any other device integration technology) display for device management tasks such as troubleshooting. Therefore, for most systems it may be necessary to download these system files for a device from the system manufacturer's web site. Note that these additional files are not EDDL files. A "package" of files typically also includes the original EDDL file from the device manufacturer.

Traditional DD did not support graphics. Therefore some systems added their own graphics through proprietary graphics files. Some systems still use such files to display device alerts integrated in the DCS operator console. These proprietary graphics files must be developed and tested for new device types. Therefore system testing of devices is necessary for such systems. EDDL with graphical enhancements eliminates the need for proprietary graphics.

Table 6 Differences between standard device files and system specific files

Device Files	System Files
Standard (IEC 61804-3)	Non-standard
Used by all systems	Unique to each system
Created by device manufacturer	Created by system manufacturer
This is the EDDL files	Other files
Supports device functionality	Support system functionality <ul style="list-style-type: none"> • System graphics (before EDDL) • Alarm management • Default settings • etc.
Download from bus organization site or device manufacturer site	Download from system manufacturer site. May also include EDDL file as a single package

System Files: Device Alarm Management

Some systems do device alert prioritizations in the system (not in the device)

The prioritization/classification of device diagnostics alerts are stored in additional proprietary files

These proprietary alarm management files must be developed and tested for new device types
Therefore system testing of devices may be necessary

Appendix 4: Poor Hosts

Original DD technology from 1992 made it possible to access different types of devices from different manufacturers using the same handheld field communicator or laptop software. Before traditional DD only proprietary solutions existed.

The original DD technology already included "wizards" (aka "methods") which is a kind of script created by the device manufacturer to guide the technician through more sophisticated procedures like setup, sensor calibration trim, or diagnostics (see separate white papers on calibration, device diagnostics, and setup). Wizards thus make using devices easy. Help and conditionals were also part of original DD technology. However, not all intelligent device management software supported wizards. That is, on many systems, advanced diagnostics and setup in the past was not so easy, particularly for FOUNDATION fieldbus devices which back then had to set the correct mode, in the correct block, write value to correct parameter, and remember to return block mode. At the same time the technician also had to remember to inform operations. All of this in addition to the actual task itself such as verifying valve moment or applying an input etc. Lack of wizards was not a problem with the DD technology itself; it was poor implementation in many early products.

Since not all systems supported the full capability of DD, device manufacturers had to provide a simplified DD file with limited functionality for these simpler systems, for instance without "conditionals" so that these limited systems could work.

The EDDL enhancements are now mandatory for all Foundation fieldbus and HART compliant systems. This includes graphics, menu system, wizards, and conditionals.

Appendix 5: Project Execution Considerations

EDDL simplifies the project execution phase of the system life cycle including for instance system integration and device commissioning.

Design: System Specification

In the design phase, make sure the system specification explicitly states that the system software shall render device displays based on Electronic Device Description Language (EDDL) files with enhancements according to IEC 61804-3. Detail device diagnostics display shall open also in operator console when device alert is selected.

Grappling with a second device integration technology in addition to EDDL comes at a high price of both time to obtain and manage additional files and money for license fees etc. Using only EDDL avoids these issues reducing cost and saving time.

Design: Instrument Specification

In the design phase, make sure the device specifications explicitly state that device shall be supplied with Electronic Device Description Language (EDDL) file with graphical enhancements according to IEC 61804-3. The EDDL file for the device shall include user guidance in the form of wizards based on EDDL methods for complex procedures, help text, as well as explaining illustrations where applicable.

System Implementation

Before system configuration work starts, the system supplier can request the device manufacturers to confirm which device revision they will be shipping. The system software does not require input of DD (file) revision or CFF revision, thus there is no need to obtain such information. Then build

the system database according to these device revisions. It may be a good idea to consolidate all device types in a checklist to easily verify that EDDL files for all devices have been received and loaded on the system. Typically the project documentation already includes a list of different device types and their manufacturers. Just add a device revision column to it.

Table 7 Device revision schedule

Protocol`	Manufacturer	Model	Description	Device Revision	Remarks
HART	Yokogawa	YTA320	Temperature	3	SIS
HART	Metso	VG800	Valve Guard	6	SIS
FF	CSI	9210	Machinery Health Transmitter	3	
FF	Rotork	FF01 Mk2	Electric Actuator	2	
WirelessHART	Rosemount	3051S	Pressure	1	
WirelessHART	Siemens	TF280	Temperature	1	
PROFIBUS-DP	Siemens	MM440	Drive	2	MCC

Make sure to load the EDDL files for all the device manufacturers the system supports, not just the device types and device revisions believed to be part of the project. The reason being that at the time of plant pre-commissioning it is almost always discovered that there are other undocumented devices at site, for instance on package units. If the system is already loaded with the EDDL files, these devices can easily be integrated and commissioned. EDDL files are small and the entire device library only takes up a tiny fraction of the hard disk capacity.

The device revision of the EDDL file shall match the device revision⁷ of the device. This is most important. DD (file) revision is not documented because it is not required by the system

Integration and FAT

At Factory Acceptance Test (FAT) it is common to connect one device of each type, of each protocol, from each manufacturer. If the EDDL files for a particular device have not been loaded on one of the workstations, it will prompt. This is a good way to physically check all EDDL files are loaded to minimize surprises at site.

Installation

There are no EDDL related activities at device installation.

Commissioning

If a tag in the system database is created for a FF device of a particular revision, but another device revision is shipped to site, that FF device cannot be commissioned. The configuration for the device tag must be changed in the system database before it can be downloaded. For instance, the device manufacturer may have shipped a device of a different device revision than originally stated. Or a device received on one of the package units may be of a different device revision. Depending on the system, this can be relatively easy.

SAT

During Site Acceptance Test (SAT) optionally test access to different device types from every workstation to make sure the EDDL files have been loaded.

⁷ for FF the DD (file) revision of the file must be higher than or equal to what is specified in the DD_REV parameter of the device - which is the minimum DD (file) revision required but this has not been a problem so far

Appendix 6: Manufacturer ID

All IDs in hexadecimal

Table 8 Manufacturer ID of some device manufacturers

Manufacturer	HART	FF
ABB	0005	000320
ABB	0012	
ABB	0016	
ABB	001a	
Berthold Technologies	00a1	00a1ff
Biffi	00b7	424946
Brooks Instrument	000a	000246
CSI	601a	435349
Dresser Valve Division	0065	445644
Dynisco Instruments	0072	44594E
Endress+Hauser	0011	452b48
Fisher Controls	0013	005100
Fisher-Rosemount Performance Technologies	007d	0004D2
Flowserve	0030	464c53
Fuji Electric	0015	000309
Honeywell	0017	48574c
HuaKong Technology	0085	0022B8
Invensys	00a9	385884
Knick	0061	000102
KOSO	00cf	00094b
KROHNE Messtechnik	0045	00012c
K-TEK	0050	000101
Magnetrol	0056	000156
Measurement Technology	0040	0BE0EC
Metso Automation	002f	000e05
Metso Automation	0057	000e05
Mettler Toledo	008e	465255
Micro Motion	001f	000310
Mobrey	003b	000103
M-System	001d	010253
Ohmart	0067	000457
OVAL Corporation	0064	00043E
Pepperl+Fuchs	005d	502b46
PR Electronics	006d	0007d0
R. Stahl	009e	0001F4
Rosemount	0026	001151
Rosemount Analytical	002e	524149
Samson	0042	00e099
Siemens	002a	534147
SMAR	003e	000302
TopWorx	6014	545758
VEGA	0062	56474b
Westlock Controls	004d	574343
Yamatake	0036	0dfc96
Yokogawa	0037	594543

Appendix 7: Frequently Asked Question (FAQ)



Why is there 0x in front of the revision number and manufacturer ID?

0x indicates the number is in hexadecimal format. For instance, 0x17 equals revision 23 in decimal.

What is the difference between DD and EDDL?

DD is the old name used by the HART Communication Foundation and the Fieldbus Foundation. PROFIBUS International has called it EDDL all along. In 2004 when the EDDL Cooperation Team (ECT) was formed, the common name agreed up on is EDDL. However, DD abbreviation is still often used. Note that EDDL is a collective name covering both the original technology from 1992 as well as the enhanced technology with graphics, advanced menu system, and persistent data storage.

Table 9 Comparison of original EDDL vs EDDL with enhancements

	Original EDDL	EDDL with Enhancements
Alias	DD, DDL, EDD, DD v4	eEDDL, Enhanced EDDL, DD v5
Year	1992	2006
Logo		
Graphics support	No	Yes
Menu support	Limited	Yes
Persistent data support	No	Yes
Standard	IEC 61804-2 Ed 1	IEC 61804-3

Why can't the EDDL file for an older version device be used with the new version device?

The EDDL file for one revision of a device is not used for other revisions of a device because the new device has new features the old device did not have. Therefore every revision of a device requires a unique EDDL file.

It is somewhat like the old printer's driver is not compatible with the new printer. The driver is the way to make the computer compatible with the new printer. The new printer is different from the old printer, that's why a new printer driver is needed.

Devices using the HART, FF, PROFIBUS, and WirelessHART digital communication technologies are forwards and backwards compatible with system software and handheld field communicators thanks to EDDL technology. That is, an EDDL file is dedicated to one specific version of a device, the EDDL file enables any system to work with that version of the device. Therefore load the EDDL files for all versions on the system. This way any system can work with any version device, older or newer than the system itself. That is, EDDL ensures backwards and forwards compatibility. When a manufacturer creates a new version device, they use EDDL to declare to the system the features of the new device so that old systems can use this new device. Devices keep changing, but software does not change. The EDDL file fills the gap.

Does "DD revision" have to match "device revision"?

No. This is wrong. Many assume that DD revision of the file must equal device revision of the device, but that's NOT how it works. The device revision of the file shall match the device revision of the device.

Why do some devices have different EDDL files for different systems?

Some (older) systems do not support conditionals in the EDDL file and most (new) systems do, so device manufacturers need to have two EDDL files. This is not a problem with EDDL technology, it is a limitation in some early systems which don't implement all EDDL features. Testing and registration ensures better support for EDDL in systems.

To support older limited systems, device manufacturers have a simplified/non-conditional EDDL-file. This EDDL-file will work on all systems. The device manufacturer may identify this simplified EDDL file by an odd DD (file) revision number (e.g. 0x01). Since many systems do support conditionals, the device manufacturer provides full-featured EDDL-files for these customers. The device manufacturer may identify this full-featured EDDL file by an even DD (file) revision number (e.g. 0x02).

That is, since systems don't pay attention to DD (file) revision, device manufacturers sometimes use the DD (file) revision to distinguish between the full-featured and simplified version of the EDDL file.

Appendix 8: Plant Operations & Maintenance

EDDL plays an important role in simplifying device revision management and system administration tasks during the operations & maintenance phase of the system life cycle.

Device Replacement

When replacing an old device with a new device of a higher revision, make sure to first obtain the EDDL file for that device and load it on the system. The device cannot be commission or fully configured or calibrated without the EDDL file of matching device revision. The new device requires the new EDDL file to be loaded on the system.

Device Upgrade

When device firmware is upgraded, make sure to first have the new EDDL file on hand. That is, find out which device revision the device will become after upgrade, obtain the EDDL file for that device revision, and load it on the system. The device cannot be commission or fully configured without the EDDL file of matching device revision.

Updated System Files

The system may require additional system files to support the device (alarm management, and possibly graphics for systems dated before EDDL enhancements). The files must be obtained from the system manufacturer before a device is upgraded or replaced. Make sure the system supplier has made these files available.

Operating System Upgrade

EDDL files are compressed text, not software, independent of Windows operating system. Therefore there is no need to obtain new EDDL files if the Windows or NET framework will be upgraded, or if service packs or hot fixes will be installed.