

CONTROL

FIELDBUS WARS CONTINUE: EDDL VS. FDT/DTM

Now That EDDs Have Been Defined as a Standard by the IEC, is There any Need for FDT?

By Dick Caro, CMC Associates, Acton, MA

For many years, the User Layer working group of ISA standards committee SP50—the creator of the function block concept now used in Foundation Fieldbus—characterized the function block as a software object resident on any network node. As in any software object, the internal algorithm of the function block is not visible or programmable by the user, though the supplier typically documents it. All the function block's communication with the outside world is through its parameters, called "attributes" in software object language. The attributes are defined by the programming of the function block as specified in Foundation fieldbus specifications or the supplier's proprietary specifications, but their values are supplied from one of four sources:

1. Dynamic output attributes, such as the desired output valve position of a PID, from the function block calculation;
2. Dynamic upstream cascade linkage, such as setpoints in a multi-level cascade;
3. Data entry by the configuration engineer during function block configuration, such as for initialization options; and
4. Operator or control engineer data entry during operation, such as for tuning parameters or manual setpoints, from the operator HMI.

Following this design, the name, data type, and any initial default values for function block attributes in Foundation Fieldbus were defined in the Device Descriptions or DDs that were exactly modeled on the equivalent definitions of HART. Profibus-PA also had similar definitions called EDDs, but were less well-defined, since there were no specified computations specified in the Profibus-PA standard for field devices. All of this changed when the Fieldbus Foundation, HART Communications Foundation, and Pro-

fibus International recently agreed to use the common EDDL being defined in a draft international standard: IEC 61804-2 (2004-05), "Function Blocks (FB) For Process Control - Part 2: Specification of FB Concept and Electronic Device Description Language (EDDL)." All three consortiums are adapting the attributes specified in the draft standard to the same functions previously expressed by their proprietary data definitions. Therefore,

FIGURE 1.
FIELDBUS AT WORK



More than 20 differential pressure transducers connected via Foundation Fieldbus, saving wiring cost and making operation easier.

all suppliers conforming to any of these three fieldbus specifications will accept the same terminology for any one attribute.

All three fieldbus organizations have contributed to forming the EDDL standard and continue to enhance it, especially in the area of display attributes. In the base DDs of HART and Foundation Fieldbus, display of data from a function block was limited to data visible on a text-

oriented handheld display. Realizing the limitations of this definition, EDDL is being extended to provide more extensive formatting information for presenting attribute data on graphic operator stations used for HMI.

The following is a description of EDDL from the announcement of joint cooperation by the Fieldbus Foundation, HART Communications Foundation, Profibus Nutzerorganisation e.V., and OPC Foundation:

“With EDDL, device developers do not need to deal with the burden of designing and programming a graphic display system to run under a variety of platforms and environments, from large HMIs to the small handheld. Instead, they can utilize common graphic display capabilities provided by commands in the EDDL. Since many host systems today already implement EDDL-based graphic display systems, devices using the extended EDDL have a common look and feel with existing devices. This permits uniform integration, configuration/setup, operation and diagnostics/maintenance, all very important in an interoperable, multi-vendor environment.”

EDDL seems to handle the problems of data exchange for both real-time access and for engineering use as long as the host system is a process control-oriented DCS or similar computer-based system. However, for general data access to user written or utility data historian software, another data access mechanism is necessary. This application need has been filled by OPC. Again, OPC assumes the presence of software objects in the real-time system, and access to the named attributes of those objects, and is the reason OPC supports the EDDL agreement.

FDT (Field Device Tools) was created, several years before the EDDL agreement, so the user wouldn't have to maintain the different attribute definitions for each fieldbus in use. FDT allows the field device supplier to offer a single DTM (Device Type Manager) independent of the fieldbus to be used for a project, whereas the host device uses an FDT Framework server that communicates with the DTMs of the field devices. Only the field device manufacturer knows the relationship between the DTM parameters and the equivalent EDDs. Most process control suppliers seem to be supporting FDT, especially those that also support Profibus-PA. FDT does not seem to have this same level of support in the factory automation world, since few PLC suppliers have created FDT Framework servers or DTMs for binary field devices.

The following is extracted from paragraph 1.4 of the FDT Interface Specification, Ver 1.2, 2004: “Within FDT only interfaces in ActiveX technology are specified for the engineering components of field devices. If the engineering system implements the corresponding interfaces, the ActiveX technology provides the automatic integration of the components and takes care of the interaction between the engineering system and the software components of the devices. Furthermore the FDT interface specification

allows the integration of device components with integrated user interfaces as well as the embedding of ActiveX controls provided by the device component for special engineering tasks.”

FDT advocates are quick to point out that FDT was designed to support the field device configuration tools, but not the real-time operational software for which EDDs are designed. The question is: Now that EDDs have been defined as a standard by the IEC, making no distinction between engineering use and real-time use, and have been accepted by all three process control fieldbus consortiums, is there any need for FDT?

To complicate the matter even further, OPC (Online Linking and Embedding (OLE) for Process Control) also has applications in this same space: the interface between the control system and the field device. OPC states that it is a high-level protocol for standardizing “host-to-controller communications.” However, with controllers potentially resident in field devices, OPC impacts this same area of communications. FDT claims to be a high-level protocol for standardizing field device to engineering configuration tools. Both are built on Microsoft COM/DCOM and use XML-encoded data frames (in OPC/DX and OPC/XML). OPC states that it is intended for Ethernet-based automation networks, whereas FDT is designed for fieldbus networks. However, OPC has been used on a variety of networks other than Ethernet, as well as with operating systems other than Windows. Furthermore, OPC is now used with Foundation Fieldbus networks to access measurement and control attributes of Foundation Fieldbus function blocks in both controller and field devices. Finally, as stated in the announcement of joint cooperation, the OPC Foundation supports EDDL. Clearly, as Ethernet-based control networks converge into the domain of fieldbus, there will eventually be a conflict between OPC/DX, FDT/DTM, and EDDL.

Practical Analysis (End User Viewpoint)

Process control end users do not write programs, and do not see these interfaces directly. They see only software tools provided by the device or system manufacturer that usually express the attributes by an expressive phrase such as “High-High Limit.” This permits the software supplier to build data input scripts in different languages according to the native language of the end user. Entire closed loop regulatory control systems are configured without knowing EDDL, FDT; or OPC using only the fill-in-the-blanks on-screen forms provided by the control system supplier's engineering software. Likewise, HMI screens and SCADA systems can be configured using only the tools provided by the HMI or SCADA software supplier without knowing EDDL, FDT; or OPC.

Finally, the main issue is: support by manufacturers. Emerson and Siemens have declared that they will not support

FDT/DTM on their products. This means that Rosemount field instruments, DeltaV DCSs, MicroMotion flowmeters, Brooks instruments, Fisher control valves, Siemens field instruments (formerly Moore Products), and Simatic 7 DCSs will not provide this support. No change in supplier preferences are predicted from this lack of FDT/DTM support, since the end user will not see any material difference in using these products, all of which will support EDDL, from using competitive products from Foxboro, ABB, Endress+Hauser or Yokogawa. Honeywell has not yet committed to FDT/DTM. All have committed to support of EDDL, however.

DDL is presently used by most DCS suppliers to design robust operator HMI and engineering tools for system configuration. The EDDL modifications simplify this software by eliminating many proprietary extensions and differences between the fieldbuses. Legacy DDL-based field devices from all suppliers will interoperate with DCSs written to this new standard. DCS suppliers will be free to make any changes without affecting field device performance.

FDT/DTM was developed with goals similar to DDL, but directed toward simplification of engineering software used to configure systems with intelligent field devices operating on several different networks. The goals for the development of FDT/DTM are admirable, but they were developed well before the evolution of EDDL. Now, it appears that EDDL exceeds the goals of FDT/DTM in the following ways:

1. More field instruments will continue to be available with EDDL, and not FDT/DTM;
2. HMI displays of field device data will be more clear and uniform since they will be able to use EDDL;
3. Use of EDDL for both configuration and operation provides much better integration with HMI and control systems; and
4. Control system and field device suppliers need only maintain a single interface data set.

It should also be noted that FDT/DTM does not provide for servicing the millions of HART, Profibus DP and Foundation Fieldbus devices already installed by users without DTM support, but which support legacy DDL. FDT/DTM is also not currently supported on thousands of existing modern control systems installed over the past decade that support legacy DDL, and can use software updates to support EDDL.

Technical Analysis

FDT is designed to be a “software receptacle” into which a DTM from each field device is “inserted.” FDT provides interfaces to the control system, and becomes an integral part of it. FDT provides access to underlying services, such as display, keyboard, database, digital communication interfaces, and other control system features.

The mechanism is much the same as used within Microsoft Office to “insert” an Excel spreadsheet into a Word document.

The FDT specification requires the control system to be implemented using a Microsoft Windows Operating System (OS) since it is built on Microsoft’s Component Object Model (COM) and Distributed Component Object Model (DCOM) technology. Since COM/DCOM was not

FIGURE 2.
VERY DISTRIBUTED CONTROL



Fieldbus and smart valves permit locating the controller in the control valve, distributing control to the exact point it is needed.

intended to communicate with process automation field devices, FDT extends these technologies to accommodate this requirement.

FDT provides the interface to the HART, Profibus, and/or Foundation Fieldbus communication drivers, if they are present in the control system.

DTM software is supplied by the field device supplier as

an Active X component on a disk or via the Internet. FDT is installed on the computer to be used for configuration as a component of its software architecture. The DTMs from each type of field device are then “plugged into” the FDT, using Microsoft Active X. Using available FDT services, the DTM provides all access to the field device during configuration.

All the information the field device manufacturer wishes to make available is programmed into the DTM. This includes real-time data, alarms, events, configuration information, screen displays, multilingual help screens, device specific

all of the versions of Windows used by the DCSs owned by their customers.

Finally, it is not reasonable to expect that all future HMI software will operate on a Microsoft Windows operating system, and will support COM/DCOM and ActiveX objects. Such operating system dependencies are at odds with the needs of industrial automation, even though WinTel-based systems have been in use for several years. The history of this industry also involves products constructed with various versions of UNIX. Use of FDT/DTM would preclude the use of Linux in future HMIs simply because it does not sup-

Support of FDT/DTM is no longer important and only serves to mask the important issues.

documentation, parameter validity check, generation of dependent variables, diagnostic functions, and the device calibration sequence. A DTM can support more than one type of field device, but this increases its programming complexity.

In the FDT/DTM architecture, software for implementing HMI visualization of field device data is the responsibility of the field device suppliers. The HMI screen images appearing with data from the field devices have a different look and feel depending on the definitions of each field device supplier, and are not designed by the HMI supplier or the control system supplier. In contrast, EDDL ensures a consistent look and feel on the HMI of the control system, because the control system supplier designs a single interpretation for all device description files. The field device supplier only defines the specific DDs used for that field device and any of its function blocks, after they have been configured. FDT/DTM is not presently capable of providing a DTM that can display the DDs of a configured (instantiated) function block in a field device.

EDDL is designed for interoperability with all operating systems, since it does not depend upon any operating system support. FDT/DTM requires drivers (DTMs) that tightly integrate the field device software with the HMI or DCS operating system. This executable software must be maintained by the field device manufacturer and integrated by the control system supplier each time the operating system is updated.

FDT/DTM technology’s operating system dependence, and its requirement for tight integration of software, lead to more challenging upgrades. For example, there are significant differences between Microsoft Windows 95, 98, Me, NT, 2000, and XP operating systems. More significant changes are coming in future Microsoft Windows releases that may require significant graphical user interface (GUI) changes. Each new operating system will likely require changes in ActiveX components that are part of the DTM. It is not reasonable to expect field device suppliers to maintain different software versions of their DTMs compatible with

port the ActiveX architecture.

EDDL is operating system independent, and can be implemented without any dependencies on a specific host architecture. EDDL will certainly be supported on a Windows environment, but is easily implemented in a Linux or any other operating system environment. Field device suppliers have been supplying DDs for more than a decade and will find it easy to expand their capabilities with EDDL.

Conclusion and Recommendation

At the time it was created, FDT/DTM was a good idea, especially to compensate for the diversity of fieldbuses and the highly divisive market at that time. The situation has changed dramatically over the past few years: first by the agreements at the fieldbus standards level, and now by the adoption of an international standard EDDL by all of the major process control-oriented fieldbus consortiums. FDT/DTM has done its work with the majority of its ideas being included into the scope of EDDL.

End users are tired of vendor squabbling over technical issues that do not affect their ability to do work, and tend to divide the market over inconsequential issues. FDT/DTM has now become one of those inconsequential issues that tend to inhibit the end user from freely selecting field instrumentation based on important issues of accuracy, availability, longevity, serviceability, delivery and price. For the 95 percent of end users interested in using one of the dominant three fieldbuses—HART, Profibus-PA, or Foundation Fieldbus— support of EDDL is an important issue. Support of FDT/DTM is no longer important and only serves to mask the important issues. Suppliers of both control systems and field instrumentation should concentrate all of the limited resources on developing and supporting EDDL, and no longer implement or enhance FDT/DTM beyond current levels. 

Dick Caro, president of CMC Associates, is the chair of ISA’s SP50 Fieldbus Committee. He can be reached at RCaro@cmc.us.