



FDI

Field Device Integration Technology

Table of Contents

.....	1
1 Introduction	3
2 FDI Technology.....	3
2.1 FDI Package	3
2.2 Harmonization of EDDL.....	4
3 FDI host	5
3.1 FDI Host – Client-Server architecture	6
3.2 Standalone FDI Host.....	8
3.3 FDT based FDI host.....	9
3.4 Hierarchical networks – nested communication	9
4 Migration and reusability of existing solutions.....	11
4.1 EDD.....	11
4.2 DTM.....	12
5 Tools and standard components for increasing interoperability	13
5.1 Integrated Development Environment (IDE)	13
5.2 Standard host components.....	14
6 The benefits	15

1 Introduction

Through device integration, functions and information from devices can be made accessible at a higher level (in a central location). Both, simple and complex devices need integration!

Efficient and economically viable device integration requires multiprotocol, standardized technology so that device information can be made available across different manufacturers. In the past, the development of such uniform technology was inhibited by too many different interests from organizations and automation manufacturers, meaning different technical solutions were created. The current solutions – EDDL (Electronic Device Description Language) in various formats and FDT (Field Device Technology) – have their strengths and weaknesses, but also overlap to a large extent and thus lead to additional expense for users, manufacturers and in standardization. With FDI, a technology has been developed that combines the advantages of FDT with those of EDDL in a single, scalable solution. FDI takes account of the various tasks over the entire lifecycle for both simple and the most complex devices, including configuration, commissioning, diagnosis and calibration.

Globally leading control system and device manufacturers, such as ABB, Emerson, Endress+Hauser, Honeywell, Invensys, Siemens and Yokogawa, along with the major associations FDT Group, Fieldbus Foundation, HART Communication Foundation, OPC Foundation, PROFIBUS & PROFINET International, are supporting and driving forward the development of the FDI Technology together.

2 FDI Technology

2.1 FDI Package

The core of FDI Technology is the scalable FDI Package. The FDI Package is a collection of files: The Electronic Device Description (EDD), includes the device definition (Def), business logic (BL) and user interface description (UID). It is based on Electronic Device Description Language (EDDL, IEC 61804-3). The optional user interface plug-in (UIP) offers the advantages of freely programmable user interfaces familiar from FDT, based on Windows Presentation Foundation (WPF). The device manufacturers define via the FDI Device Package which data, functions and user interfaces are stored on the FDI Server.

The device definition describes the field device data and the internal structure (e.g. blocks). The business logic primarily ensures that the device data remain consistent, e.g. refresh data when unit is changed. The dynamic dependencies, in particular, play a part here, e.g. options and ranges that depend on prior selection of other settings like only show temperature units if temperature value is chosen, or only display custom settings if custom probe is selected. User interface descriptions and user interface plug-ins define the field device user interfaces. Product documentation, protocol-specific files, such as GSD or CFF, etc. will all be added to the FDI Package as attachments.

FDI has defined a single protocol independent encoded file format for the EDD part of the FDI Package.

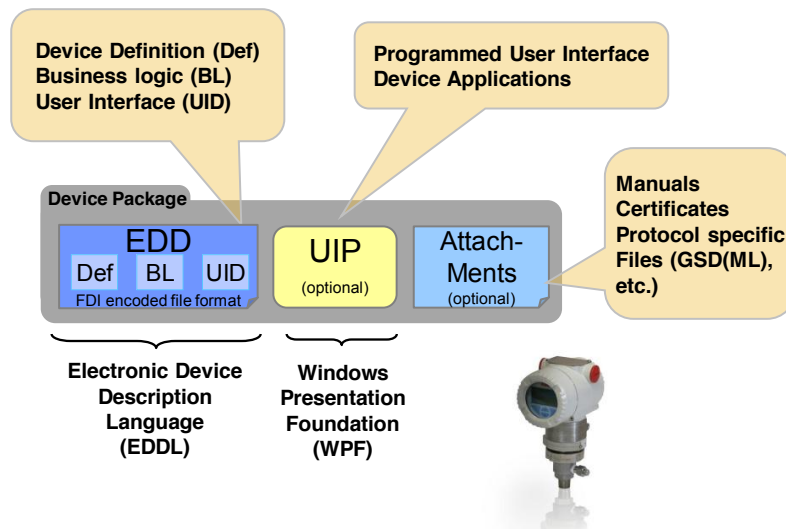


Figure 1: FDI Device Package

2.2 Harmonization of EDDL

The Electronic Device Description Language (EDDL) as a basis for FDI is specified in the international standard IEC 61804-3. Furthermore, the standard defines in profiles which constructs from the overall language scope and which library functions are permitted to be used for the HART, Foundation Fieldbus and PROFIBUS protocols. Accordingly, the protocols in question use a subset of the language standard.

To a certain extent, this is due to differences in the protocols or the underlying device models. However, a much more substantial component of the differences cannot be traced to protocol-specific requirements, but is a consequence of the history of the language, which, until 2006, existed in independent specifications of the fieldbus organizations. For device manufacturers, this means that for a device type with the protocol types HART, PROFIBUS and Foundation Fieldbus, three different EDDs also have to be created instead of one core EDD with three separated communication descriptions.

In the FDI technology, EDDL is largely harmonized and standardized across the protocols. A uniform EDDL is the foundation for uniform multiprotocol FDI Package development tools (FDI IDE), and uniform host components, such as EDD Engine (interpreter) and the client-side components UID Renderer and UIP Hosting. The result is sustainable strengthening of the key factors interoperability and quality. At the same time, cost savings can be achieved for device and system manufacturers, fieldbus organizations and, last but not least, end users.

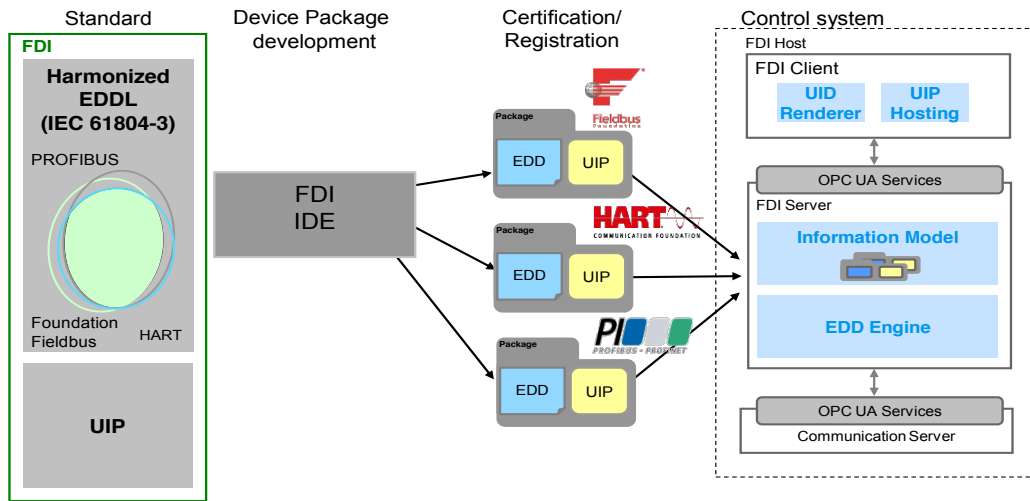


Figure 4: FDI workflow

3 FDI host

A FDI host could be device management software as part of a process control or asset management system, a device configuration tool on a laptop or a hand held field communicator. The FDI architecture allows for different kind of host implementations including a FDT based host system that ensures interoperability with FDT.

In any case a FDI host always supports all features of a FDI Device Package.

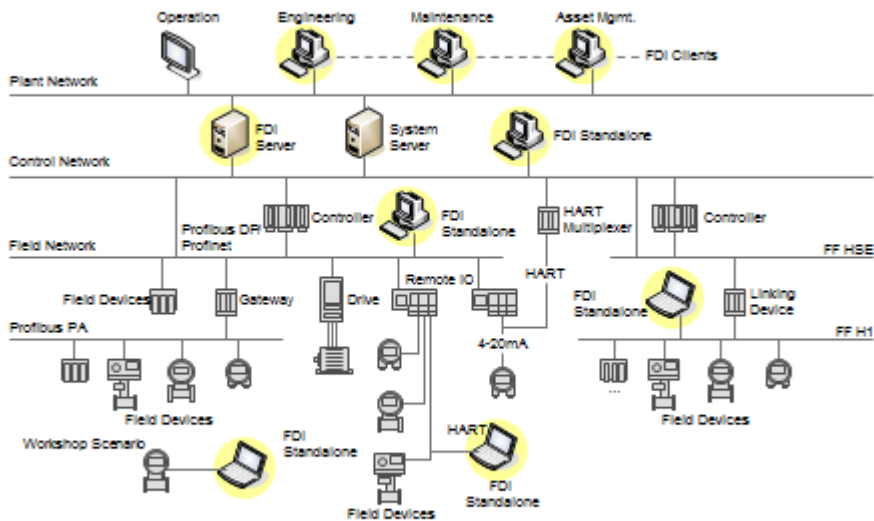


Figure 2: FDI host systems in various applications

3.1 FDI Host – Client-Server architecture

In a client-server architecture, a server provides services which are accessed by various clients (often distributed). The FDI architecture is based on the information model used in the OPC Unified Architecture (OPC UA), which provides advantages such as platform independence. The FDI Server imports FDI Device Packages into its internal device catalog. This makes version management of FDI Packages much easier as they are managed centrally within the FDI Server. As FDI Packages do not require registration in the sense of a software installation, there are no unpleasant side effects.

The representation of device instances in the FDI Server takes place in the information model. The information model maps the communication topology of the automation system by representing the entire communication infrastructure and the field devices as objects. This is also where the data, functions and user interfaces of the devices are stored. If an FDI Client wishes to work with a device, it accesses the information model and, for example, loads the user interface of the device in order to display it on the client side, in a similar manner to a web browser. By interpreting the EDD in its EDD Engine the FDI Server always ensures that the device data remain consistent. If OPC UA communication is used between client and server, OPC UA authentication and encryption mechanisms take effect and prevent unauthorized access. Other (non-FDI) OPCUA clients, e.g. MES applications can also access the device parameters in the information model without a device-specific user interface.

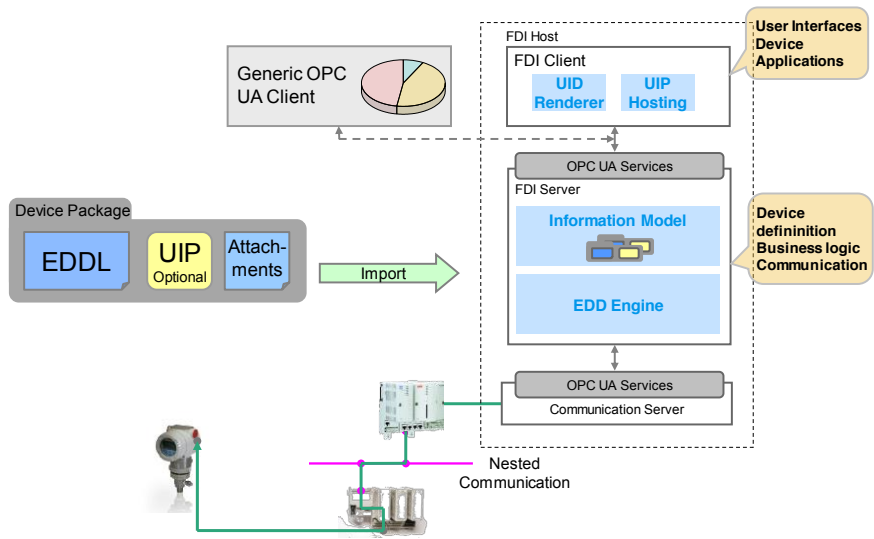


Figure 2: FDI host – client server architecture

3.2 Standalone FDI Host

FDI hosts do not have to implement the client server architecture. The FDI architecture allows also for the implementation of standalone tools.

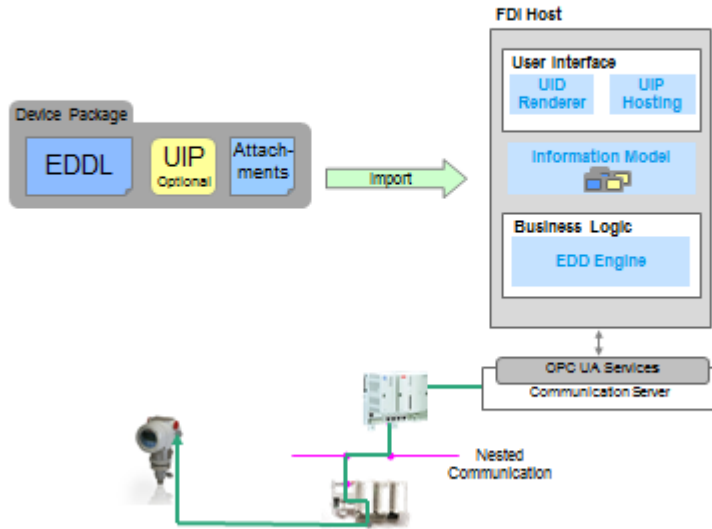


Figure 2: FDI host – standalone tool

3.3 FDT based FDI host

FDI reused FDT 2.0 interfaces for the FDI UIP. This creates the prerequisite that FDI Packages can be processed in two system architectures – one purely FDI host (figure 2) and one FDT-based FDI host (figure 5).

In the FDT-based architecture, from the viewpoint of the FDI Package, the FDI DTM acts like an FDI host; for FDT framework applications, it acts as a DTM. For many FDT Frame manufacturers, this opens up an economically attractive migration route to FDI. The FDT 2.0 Frame is retained without changes and simply supplemented with an FDI DTM. It is expected that FDI DTMs will be offered on the market in the same way as existing EDD DTMs, meaning each FDT Frame manufacturer will not have to develop the component themselves.

The interoperability with FDT allows all system and tool manufacturers to support FDI, meaning they can process FDI Packages. This will primarily benefit device manufacturers who currently have to provide a DTM and a EDD for one device. Instead device manufacturers deliver only a FDI Package. Over the longer term, this will lead to reductions in the number of device drivers per device type, and thus to significant savings in product development and maintenance. In the end, the end users benefit from the improved interoperability and smaller range of versions.

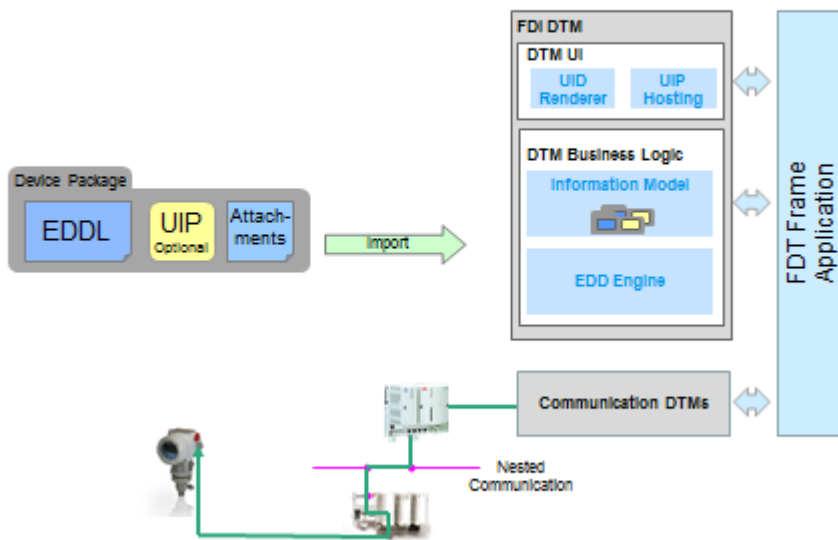


Figure 5: FDT based FDI host

3.4 Hierarchical networks – nested communication

Another FDT concept that is used in FDI is nested communication, i.e. the open integration of gateways, and the integration of communication drivers via communication servers.

In FDI, every device is mapped through an FDI Package and integrated into an FDI host. This also applies to communication devices. All operations and services that are necessary for communication are described and provided in a standardized format as part of an FDI Communication Package via EDDL code. The FDI Server takes care of the execution and management of all tasks. Devices that are used for access to networks require access to

hardware resources. This is implemented by the FDI Communication Server.

The FDI communications concept allows communication with devices in heterogeneous hierarchical networks and the use of any communication hardware.

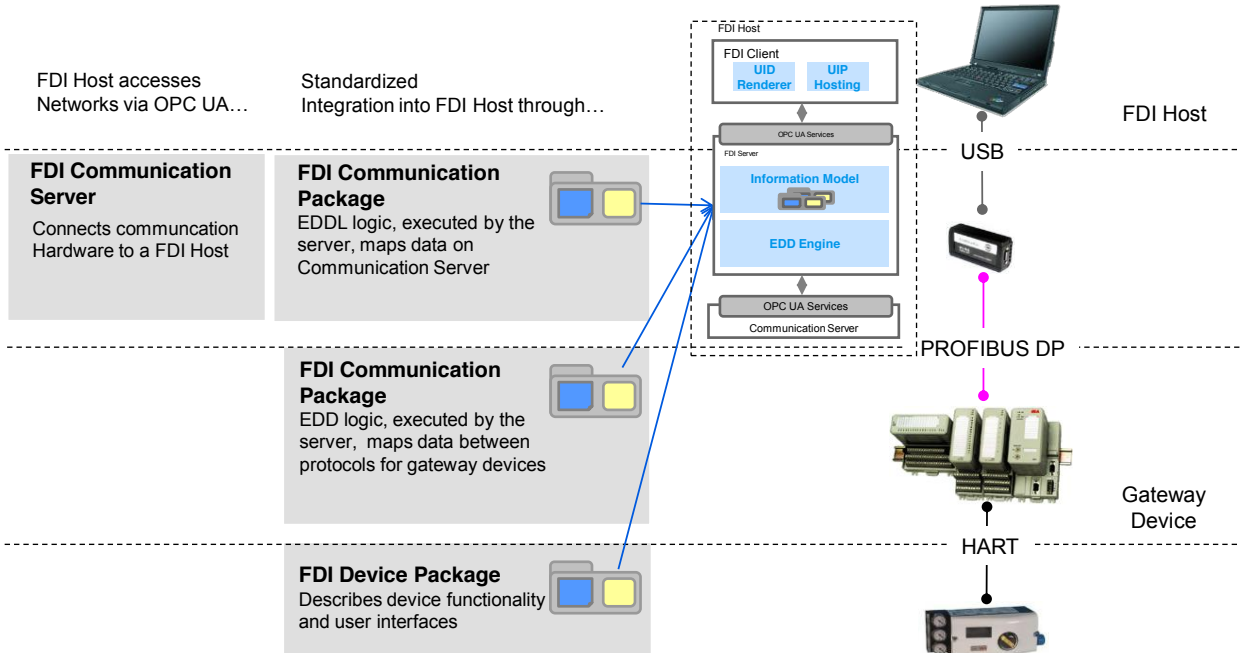


Figure 3: Nested Communication

Figure 5: FDI-FDT Interoperability

4 Migration and reusability of existing solutions

FDI is a new integration technology that is intended in the long term to replace the existing EDDL and FDT technologies. Of course, the installed base will have to be supported during the transitional period. In the process industry, this could mean periods of more than 10 years. Tools and system manufacturers are used to handle life cycle topics and provide concepts for maintaining the system hardware and software. In case of system software upgrade it is always possible to migrate from DTM or EDD to FDI without changing the devices, because the EDD, DTM and the new FDI Package exist only on the computer, not in the device.

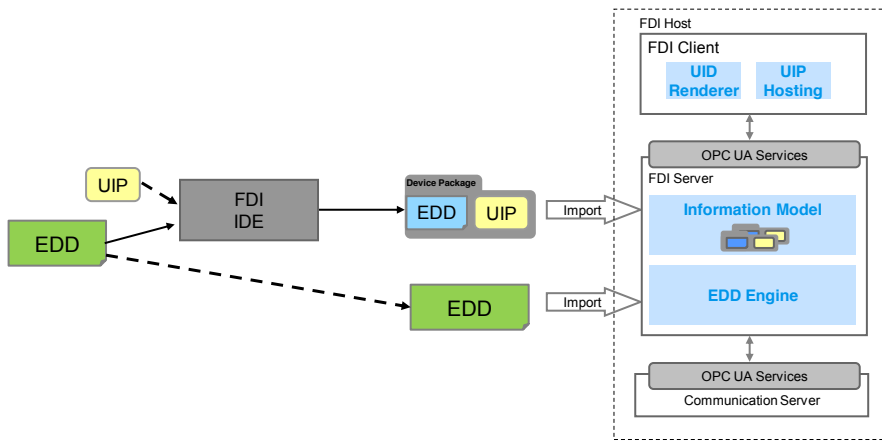
Furthermore, it should be made possible for device manufacturers to create FDI Packages efficiently and economically for new as well as for existing device. Specifically, it should be possible for existing EDD sources or DTMs to be reused.

The FDI Technology supports both methods.

4.1 EDD

The installed base of EDD (Electronic Device Descriptions) is supported by two means:

1. The FDI Package development tool (IDE, Integrated Development Environment) allows existing EDD sources to be converted into the harmonized EDD and used in a device package in combination with a UIP.
2. The multiprotocol EDD Engine (EDD interpreter) is backward compatible with existing EDD formats. Existing EDDs can therefore be processed directly in an FDI host.



6

Figure 6: EDD Migration

4.2 DTM

Similar to the migration concept for EDDs, the migration of DTMs will be made possible:

1. The FDI Package development tool (IDE) allows existing EDD sources to be converted into the harmonized EDD and used in a device package in combination with a UIP.
2. Existing DTM software can be reused for creating UIPs and then used as a component of the FDI Package.
3. FDT frame applications can process device packages with the aid of an FDI DTM. As the multiprotocol EDD Engine (EDD interpreter) is also backward compatible with existing EDD formats, existing EDDs can also be processed using the FDI DTM.
4. Existing DTMs can still be processed in the FDT frame application.

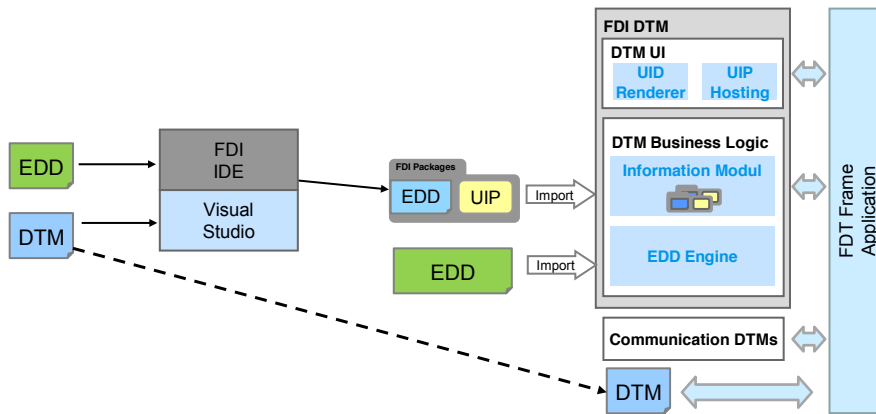


Figure 7: DTM Migration

5 Tools and standard components for increasing interoperability

To support the devices and system development, as well as increasing the interoperability of FDI products, multiprotocol software tools and standard host components are provided by the fieldbus organizations.

5.1 Integrated Development Environment (IDE)

The Integrated Development Environment (IDE) helps device manufacturers create device packages for FF, HART, PROFIBUS and PROFINET devices. Essentially, the tool has four components:

- EDDs are created with the help of an editor and converted to the encoded file format by means of “tokenizing” (the binary coding of an EDD).
- The encoded EDD, the UIP developed using Visual Studio, and the attachments are combined to form an FDI Package.
- A runtime environment (reference host) runs the FDI Packages for tests and debugging.
- The test engine is used to automatically run FDI Package conformance tests.

The FDI Packages that a device manufacturer creates in this manner are certified and registered by Fieldbus Foundation, HART Communication Foundation and PROFIBUS & PROFINET International, together with the respective device hardware.

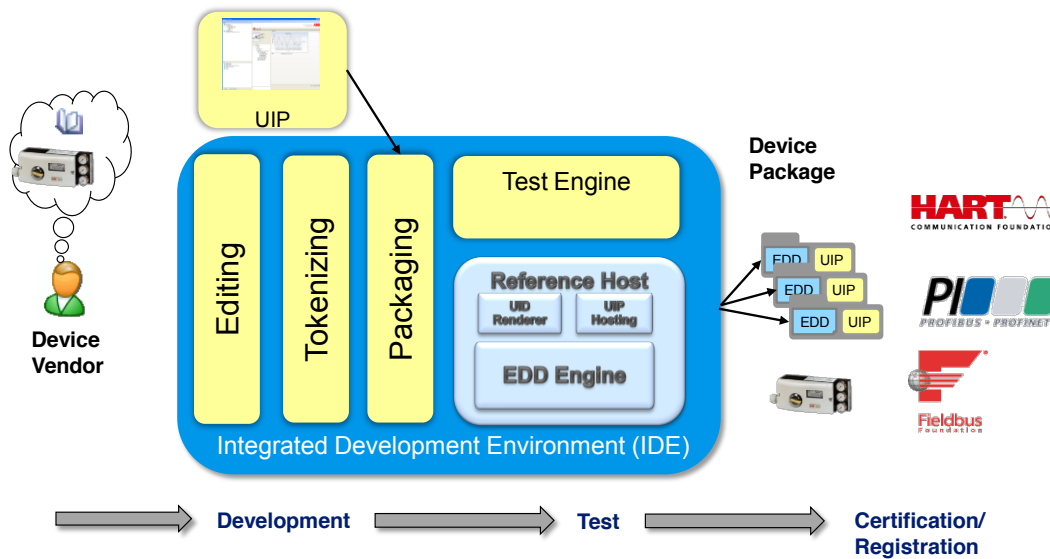


Figure 8: FDI Integrated Development Environment

5.2 Standard host components

The interpreter components available today for specific protocols differ significantly in their functional capabilities, quality and behavior towards the host system. They also only support the fieldbus-specific EDD version.

The way out of this dilemma is to develop uniform, multiprotocol standard FDI host components. EDD Engine (interpreter), UID Renderer and UIP Hosting components ensure that an FDI Device Package behaves in the same way in various systems. The EDD Engine supports the entire language scope of EDD in a multiprotocol manner, in accordance with IEC 61804-3, and is backward compatible with existing EDD formats. This means that, in future, system manufacturers no longer need to integrate three interpreter components, but only one. This saves time and effort, and aims to contribute to improving the quality and interoperability.

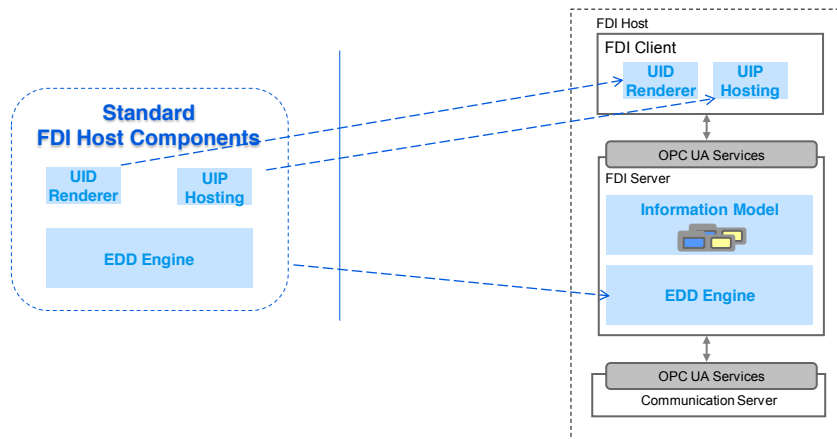


Figure 9: FDI standard host components

6 The benefits

FDI combines the tried-and-tested concepts of both EDDL and FDT and thus provides benefits for control system manufacturers, device manufacturers and users.

For control system manufacturers, the client-server architecture simplifies the use of device data and functions in powerful, distributed control systems. In addition, transparent access to device data and functions facilitates the integration of other applications (e.g. connection of MES). Other benefits are clear: the central management of data prevents inconsistencies and the automatic loading of user interfaces by the client means client-side installation is no longer required.

For device manufacturers, FDI reduces effort and saves costs because, in future, only a FDI Device Package has to be created for one device, instead of the current EDD variants and DTM. Another advantage is the scalability of the FDI Device Package. Simple devices get along with a simple device package. By their nature, complex devices require a more complex device package.

An Integrated Development Environment and Standard Host components ensure interoperability and cost efficient development of FDI Device Packages and host systems.

For the customer, the main benefit of FDI lies in the standardized integration of field devices through a future-proof standard that ensures unrestricted interoperability of device packages from a wide variety of device manufacturers with FDI systems (FDI hosts) from a wide variety of control system manufacturers.