

DIN EN 62769-3

ICS 35.240.50

**Feldgeräteintegration (FDI) –
Teil 3: FDI-Server
(IEC 62769-3:2015);
Englische Fassung EN 62769-3:2015**

Field Device Integration (FDI) –
Part 3: FDI Server
(IEC 62769-3:2015);
English version EN 62769-3:2015

Intégration des appareils de terrain (FDI) –
Partie 3: Serveur FDI
(IEC 62769-3:2015);
Version anglaise EN 62769-3:2015

Gesamtumfang 64 Seiten

DKE Deutsche Kommission Elektrotechnik Elektronik Informationstechnik in DIN und VDE

Anwendungsbeginn

Anwendungsbeginn für die von CENELEC am 2015-06-24 angenommene Europäische Norm als DIN-Norm ist 2016-03-01.

Nationales Vorwort

Vorausgegangener Norm-Entwurf: E DIN EN 62769-3:2013-01.

Für dieses Dokument ist das nationale Arbeitsgremium K 956 „Feldbus“ der DKE Deutsche Kommission Elektrotechnik Elektronik Informationstechnik in DIN und VDE (www.dke.de) zuständig.

Die enthaltene IEC-Publikation wurde vom SC 65E „Devices and integration in enterprise systems“ erarbeitet.

Das IEC-Komitee hat entschieden, dass der Inhalt dieser Publikation bis zu dem Datum (stability date) unverändert bleiben soll, das auf der IEC-Website unter „<http://webstore.iec.ch>“ zu dieser Publikation angegeben ist. Zu diesem Zeitpunkt wird entsprechend der Entscheidung des Komitees die Publikation

- bestätigt,
- zurückgezogen,
- durch eine Folgeausgabe ersetzt oder
- geändert.

Für den Fall einer undatierten Verweisung im normativen Text (Verweisung auf ein Dokument ohne Angabe des Ausgabedatums und ohne Hinweis auf eine Abschnittsnummer, eine Tabelle, ein Bild usw.) bezieht sich die Verweisung auf die jeweils aktuellste Ausgabe des in Bezug genommenen Dokuments.

Für den Fall einer datierten Verweisung im normativen Text bezieht sich die Verweisung immer auf die in Bezug genommene Ausgabe des Dokuments.

Der Zusammenhang der zitierten Dokumente mit den entsprechenden Deutschen Dokumenten ergibt sich, soweit ein Zusammenhang besteht, grundsätzlich über die Nummer der entsprechenden IEC-Publikation. Beispiel: IEC 60068 ist als EN 60068 als Europäische Norm durch CENELEC übernommen und als DIN EN 60068 ins Deutsche Normenwerk aufgenommen.

Das Präsidium des DIN hat mit Präsidialbeschluss 1/2004 festgelegt, dass DIN-Normen, deren Inhalt sich auf internationale Arbeitsergebnisse der Informationsverarbeitung gründet, unter bestimmten Bedingungen allein in englischer Sprache veröffentlicht werden dürfen. Diese Bedingungen sind für die vorliegende Norm erfüllt.

Da sich die Benutzer der vorliegenden Norm der englischen Sprache als Fachsprache bedienen, wird die Englische Fassung der EN 62769-3 veröffentlicht. Zu deren Abschnitt 1, der den Anwendungsbereich festlegt, und Abschnitt 3, der die Begriffe festlegt, wurde eine Übersetzung angefertigt und als informativer Nationaler Anhang NA der vorliegenden Norm hinzugefügt. Für die meisten der verwendeten Begriffe existieren keine gebräuchlichen deutschen Benennungen, da sich die deutschen Anwender in der Regel ebenfalls der englischen Benennungen bedienen. Diese Norm steht nicht in unmittelbarem Zusammenhang mit Rechtsvorschriften und ist nicht als Sicherheitsnorm anzusehen.

Das Original-Dokument enthält Bilder in Farbe, die in der Papierversion in einer Graustufen-Darstellung wiedergegeben werden. Elektronische Versionen dieses Dokuments enthalten die Bilder in der originalen Farbdarstellung.

Nationaler Anhang NA (informativ)

1 Anwendungsbereich^{N1)}

Dieser Teil von IEC 62769 legt den FDI-Server fest. Die gesamtheitliche FDI-Architektur ist in Bild 1 dargestellt. Die im Anwendungsbereich dieses Dokumentes liegenden Architekturkomponenten sind in diesem Bild hervorgehoben.

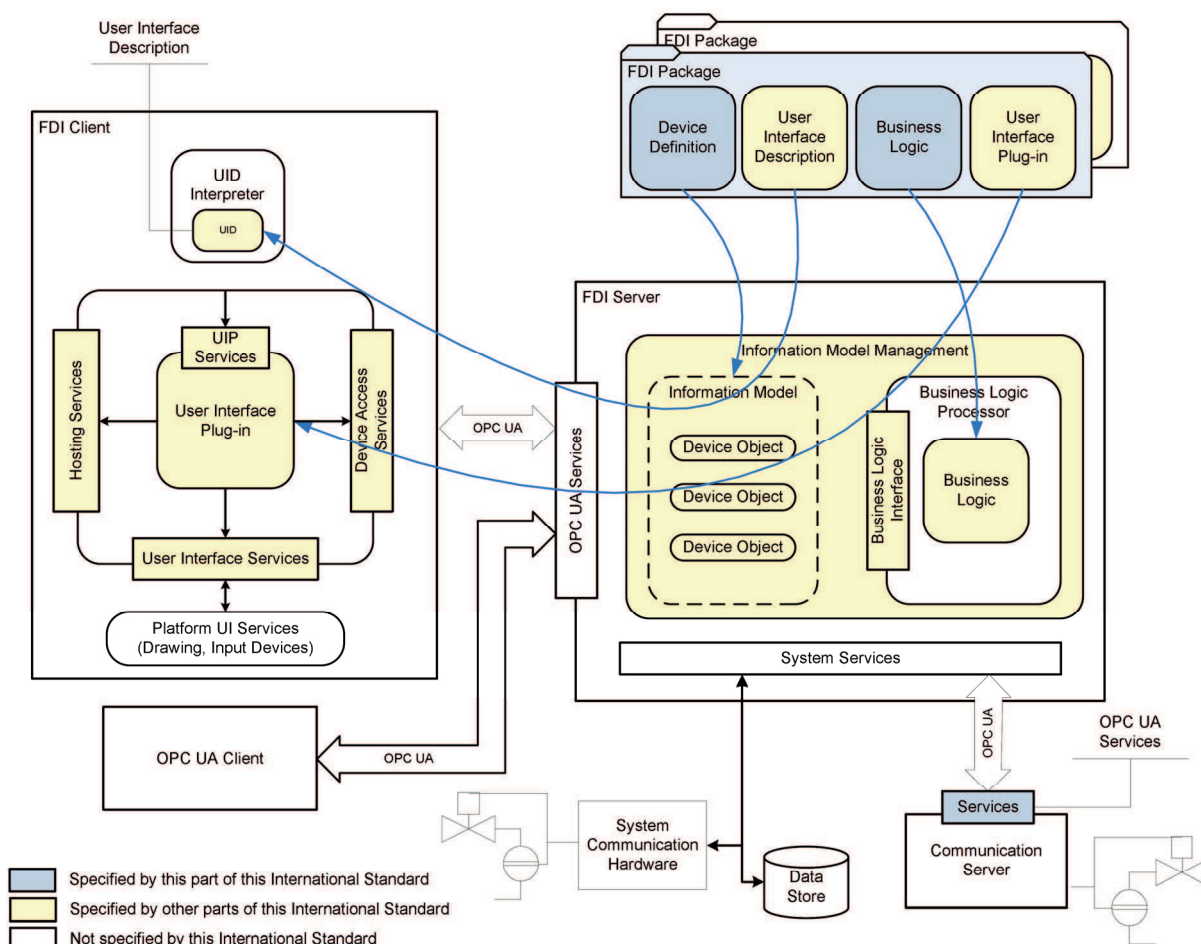


Bild 1 – FDI-Architektur

3 Begriffe, Abkürzungen und Konventionen^{N2)}

3.1 Begriffe

Für die Anwendung dieses Dokuments gelten die Begriffe nach IEC 62769-1 und die folgenden Begriffe.

^{N1)} Nationale Fußnote: Die Nummerierung der IEC 62769-3:2015 wurde beibehalten.

^{N2)} Nationale Fußnote: Die deutschen Übersetzungen haben Empfehlungscharakter und können nur die ungefähre Begrifflichkeit in Deutsch wiedergeben. Unter Experten wird empfohlen, ausschließlich die englischen Fachbegriffe zu verwenden, um die Gefahr von Missverständnissen zu reduzieren.

3.1.1

Aktionsproxy

(en: Actions Proxy)

interne Einheit des FDI-Servers, die alle EDD-Methoden zusammenfasst, welche in einer Definition der EDD-Aktion festgelegt sind

English Version

Field Device Integration (FDI) - Part 3: FDI Server
(IEC 62769-3:2015)

Intégration des appareils de terrain (FDI) - Partie 3: Serveur
FDI
(IEC 62769-3:2015)

Feldgeräteintegration (FDI) - Teil 3: FDI-Server
(IEC 62769-3:2015)

This European Standard was approved by CENELEC on 2015-06-24. CENELEC members are bound to comply with the CEN/CENELEC Internal Regulations which stipulate the conditions for giving this European Standard the status of a national standard without any alteration.

Up-to-date lists and bibliographical references concerning such national standards may be obtained on application to the CEN-CENELEC Management Centre or to any CENELEC member.

This European Standard exists in three official versions (English, French, German). A version in any other language made by translation under the responsibility of a CENELEC member into its own language and notified to the CEN-CENELEC Management Centre has the same status as the official versions.

CENELEC members are the national electrotechnical committees of Austria, Belgium, Bulgaria, Croatia, Cyprus, the Czech Republic, Denmark, Estonia, Finland, Former Yugoslav Republic of Macedonia, France, Germany, Greece, Hungary, Iceland, Ireland, Italy, Latvia, Lithuania, Luxembourg, Malta, the Netherlands, Norway, Poland, Portugal, Romania, Slovakia, Slovenia, Spain, Sweden, Switzerland, Turkey and the United Kingdom.



European Committee for Electrotechnical Standardization
Comité Européen de Normalisation Electrotechnique
Europäisches Komitee für Elektrotechnische Normung

CEN-CENELEC Management Centre: Avenue Marnix 17, B-1000 Brussels

European foreword

The text of document 65E/346/CDV, future edition 1 of IEC 62769-3, prepared by SC 65E "Devices and integration in enterprise systems" of IEC/TC 65 "Industrial-process measurement, control and automation" was submitted to the IEC-CENELEC parallel vote and approved by CENELEC as EN 62769-3:2015.

The following dates are fixed:

- latest date by which the document has to be (dop) 2016-03-24
implemented at national level by
publication of an identical national
standard or by endorsement
- latest date by which the national (dow) 2018-06-24
standards conflicting with the
document have to be withdrawn

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. CENELEC [and/or CEN] shall not be held responsible for identifying any or all such patent rights.

Endorsement notice

The text of the International Standard IEC 62769-3:2015 was approved by CENELEC as a European Standard without any modification.

In the official version, for Bibliography, the following notes have to be added for the standards indicated:

IEC 61804-5	NOTE	Harmonized as EN 61804-5 ¹⁾
IEC 62769-6	NOTE	Harmonized as EN 62769-6

1) To be published.

CONTENTS

INTRODUCTION.....	6
1 Scope.....	7
2 Normative references	7
3 Terms, definitions, abbreviated terms and acronyms	8
3.1 Terms and definitions	8
3.2 Abbreviated terms and acronyms.....	8
4 Overview	8
5 Information Model.....	9
5.1 General.....	9
5.2 Online/Offline	10
5.2.1 Overview	10
5.2.2 Transfer to device	10
5.2.3 Transfer from device	10
5.3 Access privileges	11
5.4 Private Parameters	11
5.5 Locking.....	11
5.6 EditContext	12
5.6.1 Concept and usage model	12
5.6.2 Services	13
5.6.3 NodeIds	14
5.6.4 Reading.....	14
5.6.5 Writing.....	14
5.6.6 Writing dominant and dependent Variables	14
5.6.7 Actions (EDD METHODS)	15
5.6.8 UIDs	16
5.6.9 Synchronization	16
5.7 Reading	16
5.7.1 General	16
5.7.2 Reading offline variables	17
5.7.3 Reading online variables	17
5.8 Writing.....	19
5.8.1 General	19
5.8.2 Write offline variables.....	19
5.8.3 Writing online variables	20
5.8.4 Writing to an EditContext.....	22
5.9 Subscription.....	23
5.9.1 General	23
5.9.2 Subscription of offline variables.....	23
5.9.3 Subscription of online variables	24
5.10 Device topology	26
5.10.1 General	26
5.10.2 Connection Points	26
5.10.3 Topology management.....	27
5.10.4 Topology scanning	30
5.10.5 Use of SCAN function	31

5.10.6	Validation of defined topology	31
5.11	User Interface Elements	32
5.11.1	User Interface Descriptions	32
5.11.2	User Interface Plug-ins	33
5.12	Actions	33
5.12.1	FDI Server – FDI Client interaction	33
5.12.2	Action state machine	36
5.12.3	Actions Proxies	37
5.12.4	Actions, EDD Actions and Actions Proxies	38
6	OPC UA services	39
6.1	OPC UA profiles	39
6.2	Service error information	39
6.2.1	Overview	39
6.2.2	OPC UA services and their response	40
6.2.3	Mappings of EDDL response codes to OPC UA service response	40
6.3	Parameter value update during write service request	41
6.4	Localization	41
6.5	Audit events	41
7	Communication	42
7.1	Notation	42
7.2	General	42
7.2.1	Concepts	42
7.2.2	Terms	44
7.3	Communication Service processing	45
7.3.1	Communication Service invocation	45
7.3.2	Analyze communication path	45
7.3.3	Manage communication relations	46
7.3.4	Communication service request mapping	46
7.3.5	Communication service request propagation	47
7.3.6	Communication error handling	48
7.4	FDI Communication Server specific handling	48
7.4.1	Discovery	48
7.4.2	Information Model synchronization	49
8	Parallel Execution within the FDI Server	49
8.1	Motivation	49
8.2	Internal structure of the EDD interpreter	49
8.3	Rules for running an EDD entity	50
Annex A (informative)	FDI Server functional structure	51
A.1	FDI functional elements	51
A.2	FDI Server extension	52
Annex B (informative)	Access privileges and user roles	54
B.1	User roles and usage case	54
B.2	Private data usage	55
Annex C (informative)	Parallel execution within the FDI Server – Examples	56
C.1	Simple example for a synchronous execution	56
C.2	Example for a concurrent execution	56
C.3	Deadlock detection in concurrent execution	58

Bibliography	59
Annex ZA (normative) Normative references to international publications with their corresponding European publications	60
Figure 1 – FDI architecture diagram	7
Figure 2 – Locking services.....	12
Figure 3 – EditContext models	13
Figure 4 – EditContext for EDD Methods	16
Figure 5 – Offline variable read	17
Figure 6 – Online variable read	18
Figure 7 – Offline variable write immediate	20
Figure 8 – Online variable write immediate	21
Figure 9 – Write with EditContext	22
Figure 10 – Offline variable subscription.....	24
Figure 11 – Online variable subscription.....	25
Figure 12 – Topology with Network objects (non-normative).....	26
Figure 13 – Add Device to topology.....	28
Figure 14 – Remove Device from topology	29
Figure 15 – Scan topology	30
Figure 16 – Action execution.....	35
Figure 17 – Action state machine	36
Figure 18 – System communication integration example	42
Figure 19 – FDI Communication Server integration example	43
Figure 20 – Gateway integration example.....	44
Figure 21 – Message propagation example scenario.....	47
Figure A.1 – Functional components of an FDI Server	51
Figure A.2 – FDI Server extensions.....	53
Figure B.1 – User roles and access privileges	54
Figure C.1 – Synchronous execution of two triggers	56
Figure C.2 – Concurrent execution of two triggers (step1).....	56
Figure C.3 – Concurrent execution of two triggers (step 2).....	57
Figure C.4 – Concurrent execution of two triggers (step 3).....	57
Figure C.5 – Concurrent execution of two triggers (step 4).....	58
Figure C.6 – Concurrent execution of two triggers	58
Table 1 – Action states	36
Table 2 – Action state transitions	37
Table 3 – EDD Action types and the EDD constructs that use them.....	38
Table 4 – OPC UA severity bits and EDDL response codes TYPE.....	40

INTRODUCTION

The International Electrotechnical Commission (IEC) draws attention to the fact that it is claimed that compliance with this document may involve the use of patents concerning

- a) method for the Supplying and Installation of Device-Specific Functionalities, see Patent Family DE10357276;
- b) method and device for accessing a functional module of automation system, see Patent Family EP2182418;
- c) methods and apparatus to reduce memory requirements for process control system software applications, see Patent Family US2013232186;
- d) extensible device object model, see Patent Family US12/893,680.

IEC takes no position concerning the evidence, validity and scope of this patent right.

The holders of these patent rights have assured the IEC that he/she is willing to negotiate licences either free of charge or under reasonable and non-discriminatory terms and conditions with applicants throughout the world. In this respect, the statement of the holder of this patent right is registered with IEC. Information may be obtained from:

- a) ABB Research Ltd
Claes Ryttoft
Affolterstrasse 4
Zurich, 8050
Switzerland
- b) Phoenix Contact GmbH & Co KG
Intellectual Property, Licenses & Standards
Flachsmarktstrasse 8, 32825 Blomberg
Germany
- c) Fisher Controls International LLC
John Dilger, Emerson Process Management LLLP
301 S. 1st Avenue, Marshalltown, Iowa 50158
USA
- d) Rockwell Automation Technologies, Inc.
1 Allen-Bradley Drive
Mayfield Heights, Ohio 44124
USA

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights other than those identified above. IEC shall not be held responsible for identifying any or all such patent rights.

ISO (www.iso.org/patents) and IEC (<http://patents.iec.ch>) maintain on-line data bases of patents relevant to their standards. Users are encouraged to consult the data bases for the most up to date information concerning patents.

IEC 61804-3¹, *Function block (FB) for process control and Electronic Device Description Language (EDDL) – Part 3: EDDL syntax and semantics*

IEC 61804-4², *Function blocks (FB) for process control and Electronic Device Description Language (EDDL) – Part 4: EDD interpretation*

IEC 62541 (all parts), *OPC unified architecture*

IEC 62541-4, *OPC unified architecture –Part 4: Services*

IEC 62541-7, *OPC unified architecture – Part 7: Profiles*

IEC 62769-1, *Field Device Integration – Part 1: Overview*

NOTE IEC 62769-1 is technically identical to FDI-2021.

IEC 62769-2, *Field Device Integration – Part 2: FDI Client*

NOTE IEC 62769-2 is technically identical to FDI-2022.

IEC 62769-4, *Field Device Integration – Part 4: FDI Packages*

NOTE IEC 62769-4 is technically identical to FDI-2024.

IEC 62769-5, *Field Device Integration – Part 5: FDI Information Model*

NOTE IEC 62769-5 is technically identical to FDI-2025.

IEC 62769-7, *Field Device Integration – Part 7: FDI Communication Devices*

NOTE IEC 62769-7 is technically identical to FDI-2027.

3 Terms, definitions, abbreviated terms and acronyms

3.1 Terms and definitions

For the purposes of this document, the terms and definitions given in IEC 62769-1 as well as the following apply.

3.1.1

Actions Proxy

internal FDI Server entity that encapsulates all the EDD Methods specified in an EDD Action definition

3.2 Abbreviated terms and acronyms

For the purposes of this document, the abbreviated terms and acronyms given in IEC 62769-1 apply.

4 Overview

The structure for an FDI Server is shown in Figure 1.

¹ To be published.

² To be published.

FDI Servers that support connectivity with third-party FDI Clients shall support OPC UA. A vendor can provide both an FDI Server and one or more FDI Clients. In this case, the FDI Clients can communicate with the FDI Server through proprietary protocols.

An FDI Server communicates with devices via Native Communication (see 7.2.1) and/or Communication Devices (see IEC 62769-7).

An FDI Server provides information to FDI Clients through an Information Model (see IEC 62769-5) as follows.

- The Information Model includes information about Device Types and Device Instances. The information for a Device Instance includes offline data (engineering data), as well as online data (values from the physical device).
- The Information Model is created using information from FDI Packages. However, not all of the information in an FDI Package is reflected in the Information Model.
- Referential integrity of the Information Model is maintained using information from FDI Packages.
- FDI Packages can contain Attachments that contain device manuals and protocol specific information (see IEC 62769-4). Those Attachments, including device manuals and protocol specific support files, are exposed via the Information Model.
- FDI Device Packages contain information about device types (see IEC 62769-4). Each device type defined in a package is mapped to a distinct DeviceType node in the Information Model.
- FDI Profile Packages are used to provide interaction with devices for which an FDI Device Package does not exist (see IEC 62769-4).
- Multiple revisions of an FDI Package generate distinct DeviceType nodes in the Information Model (see IEC 62769-4).

FDI Packages contain digital signatures that allow an FDI Server to authenticate their contents (see IEC 62769-4). An FDI Server shall not use an FDI Package if the digital signature provided by the FDI Package is invalid.

An FDI Server shall verify the FDI Technology Version (see IEC 62769-1) of any FDI Package it uses to ensure the FDI Package is compatible with the FDI Server.

5 Information Model

5.1 General

The FDI Server shall use the Device Definition of an FDI Package to maintain the Information Model.

The Device Definition can contain conditional expressions. Conditional expressions are used when a certain aspect of the Device Definition is not static but rather is dependent on the state of the device. Whenever the online or offline values of a Device Instance are modified, the FDI Server shall re-evaluate the relevant conditional expressions and modify the Information Model accordingly.

The evaluation of conditional expressions can invalidate variables in the Information Model. The FDI Server shall change the AccessLevel attribute of invalidated variables such that they are neither readable nor writable and the status of these variables shall be set to bad. Read and write service requests for invalidated variables shall return a failure.

The Device Definition can specify relationships between variables in a device. These relationships can impact the value of variables in the Information Model.

The FDI Server shall generate DataChange Notifications to any FDI Clients that are subscribing to Information Model elements that have changed.

FDI Packages provide Business Logic that is used by the FDI Server to maintain the integrity of the Information Model. The Business Logic specified in an FDI Package can invoke built-in functions that shall be implemented by the FDI Server. The built-in functions that shall be implemented by the FDI Server are specified in IEC 61804.

5.2 Online/Offline

5.2.1 Overview

The Information Model maintained by the FDI Server contains online and offline values. The online values reflect values in a physical component/device. The offline values reflect values stored in a configuration database.

The offline values are updated through write service requests from an FDI Client or Business Logic executed by the FDI Server. The offline values are not updated when the FDI Server reads data from the device or writes data to the device.

The online values in the Information Model are not updated through write service requests. Successful write service requests through the Information Model result in value changes in the physical devices. The online values in the Information Model will then be updated as a result of read service requests or subscriptions.

FDI Servers can provide a server-specific mechanism for creating Device Instances without the presence of physical hardware. The FDI Server creates these instances using information in FDI Packages. All read/write requests for online values for Device Instances with no physical device shall return an error.

The transfer of information between the offline values and the physical device is supported through the TransferToDevice and TransferFromDevice methods in the Information Model. These Methods shall implement the download and upload procedures, respectively, as specified in IEC 61804-4. When no implementation is provided based on IEC 61804-4, then these Methods shall return Bad_NotSupported, as per IEC 62541-4.

The Device shall have been locked prior to invoking these methods, as specified in IEC 62769-5.

5.2.2 Transfer to device

The TransferToDevice method shall implement the download procedure as specified in IEC 61804-4. This transfers the offline values to the physical device.

As a general rule, the FDI Server should not change the Online variable node when writing a value to the device. The Online variable node should be updated only in the process of read operations or subscriptions. Notwithstanding, as specified in IEC 62769-5, the FDI Server will reset any cached Value for the target Nodes in the Information Model so that they will be re-read next time they are requested.

The status information returned for each variable included in the write service request is used to compose the TransferResult, as specified in IEC 62769-5.

5.2.3 Transfer from device

The TransferFromDevice method shall implement the upload procedure as specified in IEC 61804-4. This transfers the values from the physical device to the offline values.

If any read operations from the device fail during upload, the corresponding offline value shall not be modified.

The status information returned for each variable included in the read service request is used to compose the TransferResult, as specified in IEC 62769-5.

5.3 Access privileges

Systems implement security and access policies based on a number of characteristics such as user role and plant area. FDI Servers use these policies, along with information in FDI Packages, to determine the access privileges granted to the user.

The elements of an FDI Package can be associated with one or more usage attributes. The FDI Server uses these attributes to set the UserAccessLevel attribute of Variables and the UserExecutable attribute of Methods. The usage attributes in an FDI Package are simply hints to be used by the FDI Server, i.e., they may be disregarded or overridden by the FDI Server. See also Annex B.

5.4 Private Parameters

The Parameters and Actions specified in an FDI Package may be declared private. Private Parameters and Actions shall not be browsable; they shall only be accessible through references from other elements of an FDI Package.

More specifically, the FDI Server shall support private Parameters and Actions as follows.

- The FDI Server shall create nodes in the Information Model for the private Parameters and Actions.
- The FDI Server shall not include information about private Parameters and Actions in a response to a Browse, BrowseNext, QueryFirst, or QueryNext service request.
- The FDI Server shall return the NodeIds of private Parameters and Actions when the name of a private Parameter or Action is passed to TranslateBrowsePathsToNodeIds.
- The FDI Server shall process a read/write service request for a private Parameter in the same way as it does for public (browsable) Parameters (see 5.7 and 5.8).
- The FDI Server shall execute private Actions in the same way as it does public (browsable) Actions (see 5.12).

An example of private parameters is parameters that should only be modified through an Action. These parameters should not be visible to FDI Clients to prevent direct access. FDI Clients invoke Actions to access these private parameters.

5.5 Locking

The FDI Server provides locking services to grant FDI Clients exclusive access to Device and Network elements in the Information Model. The locking services consist of a set of Methods and status information. The methods, and their behavior, are specified in IEC 62769-5.

The following behavior shall be implemented by the FDI Server to support locks.

- Locking applies to both online and offline nodes.
- Once locked by one FDI Client, any attempt to write to a Parameter or to execute an Action by another FDI Client shall be rejected.
- Locking is not required for read services.
- Parameters that are locked by one FDI Client can still be read by other FDI Clients, i.e., read requests on a Parameter that is locked are not rejected.

Internal use of the locking mechanism for maintaining the Information Model integrity is FDI Server vendor specific.

Figure 2 illustrates a locking sequence with multiple service invocations during the locked state.

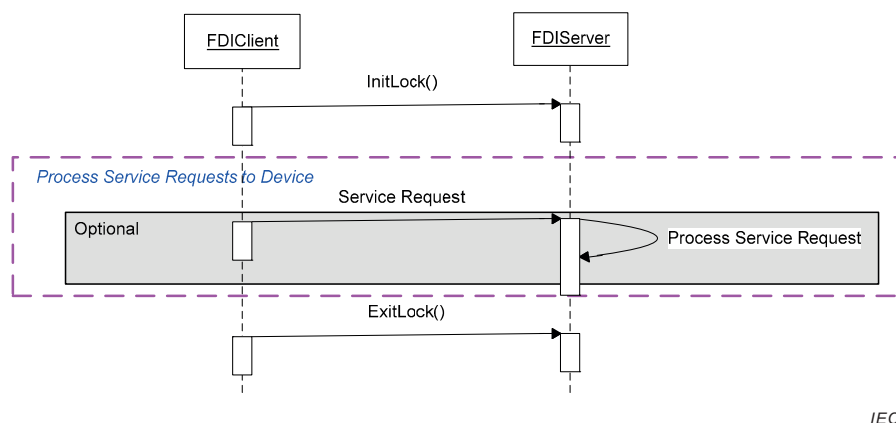


Figure 2 – Locking services

A service request that requires locking shall fail either partially or completely if no lock has been acquired by the FDI Client via InitLock prior to requesting the service. The FDI Client has to release the lock via ExitLock after all service requests have been completed.

NOTE A write operation will partially fail, i.e., it will return a status code for each variable in the set of variables to be written since some may belong to devices that are locked and some to devices that are not locked.

FDI Servers may queue InitLock requests until a service for which a lock has been created completes and the lock has been released. However, such an optimization is not part of the standard behavior required of an FDI Server.

5.6 EditContext

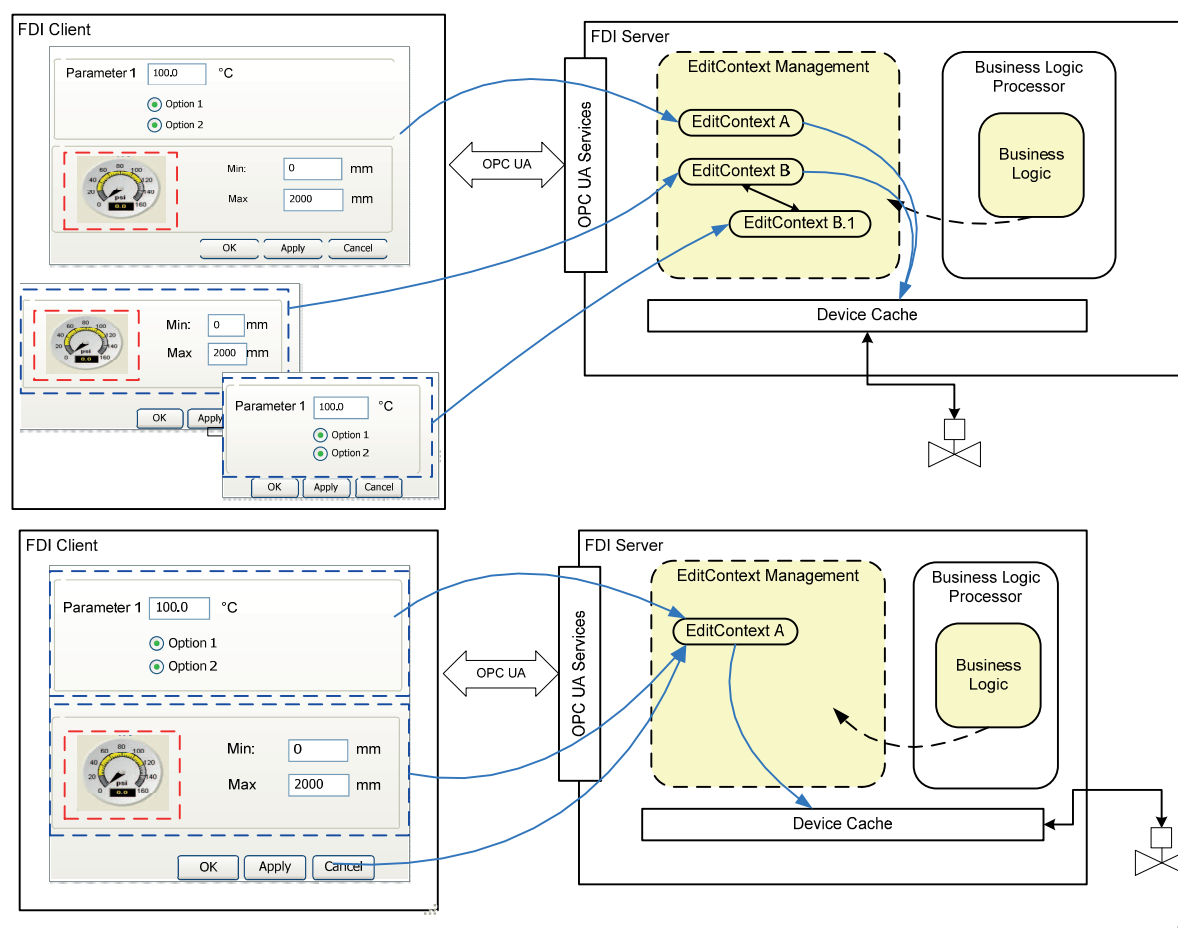
5.6.1 Concept and usage model

The FDI Server provides the EditContext model to interact with Clients during their editing task. The concept is closely related to UIDs and fulfills the needs for Server-driven UI dialogs based on EDDL rules.

An EditContext can be used to make changes to Variable Values visible to the Server without applying them to the online or offline representation of a Device. The Server will apply business logic associated to the edited Variable which – in some cases – causes changes to other Variable Values (e.g. if an engineering unit is changed) or the UID (e.g. a Variable becomes invisible). Thus the Client can use an EditContext to modify (edit) Parameters like engineering units, ranges and more, verify any side effects, and re-adjust the settings before applying the changes.

An FDI Server may implement different EditContext strategies:

- A single EditContext instance for all dialogues of an FDI Client.
- Multiple EditContext instances.
- Hierarchical EditContext instances.



IEC

Figure 3 – EditContext models

Figure 3 shows two possible Server strategies and how the Client can adapt. In the lower scenario the Server provides a single EditContext instance for all dialogs. Here, the Client groups all dialogs and exposes a single set of buttons to Apply and Cancel, because it always concerns all edits.

In the upper scenario, the Server provides multiple EditContext instances, one of them as child of another one. Each instance can be addressed separately. If the changes in a child instance are applied, they are transferred to the parent. If the changes in a root instance are applied, they are transferred to the Device.

Parent-child dependencies:

- A change for a Variable in the parent overwrites an edited Value for the same Variable in the Client.
- The parent cannot be discarded before the child is discarded but the edited Values in the parent instance can be applied or reset.

5.6.2 Services

A set of Services is provided to the FDI Client to maintain EditContext instances (see IEC 62769-5 for a detailed description of these Services):

- GetContext – This Service is used to request an EditContext instance. The Client specifies certain characteristics for the Server to decide which EditContext instance to return. Depending on its internal strategy, the Server returns the same instance or new instances.

- **RegisterNodes** – The FDI Client has to register all Nodes of the Information Model that shall be maintained in an EditContext. It is possible to register Nodes of the online and of the offline representation of a Device. The result is new NodeIds that the Client shall use when calling Services to read, write, subscribe to Variables or to invoke Actions.
- **Apply** – Transfer the modified (edited) Variable Values to the parent (either a parent EditContext instance or the Device). If the same Variable has been edited in the parent instance it will be overwritten with a call of the Apply Service for the child.
- **Reset** – Clears all modifications. A Reset of already applied modifications is not possible.
- **Discard** – Deletes an EditContext instance (and its children). Edited Values that have not been applied are discarded. Once deleted, all registered NodeIds will be invalid. If such NodeIds are still subscribed, the Client is notified with proper StatusCodes.

The Client first calls `GetEditContext` to acquire an EditContext instance. It will then register the Nodes it wants to be part of it. The registration returns new NodeIds which can then be used for reading, writing or subscribing Variables and for calling Methods.

The Client can call `GetEditContext` multiple times, for instance when it opens an additional edit window or for a completely separate dialog (diagnosis in parallel to configuration). It is up to the Server strategy whether it returns a new instance or the same instance. The Client is expected to adapt its user interface to the EditContext strategy of the Server. See Figure 3 for how Clients may position the Apply and Cancel buttons so that the User clearly understands which changes s/he applies or discards.

5.6.3 NodeIds

`RegisterNode` returns two NodeIds for each registered Node: a `ContextNodeId` and a `DeviceNodeId`. The Client uses these NodeIds when calling OPC UA Services to read, write and subscribe or call a Method.

Using the `ContextNodeId` addresses the Value in the EditContext instance. Using the `DeviceNodeId` addresses the Value in the Device.

5.6.4 Reading

Reading or subscribing a Variable with the `ContextNodeId` will return the edited Value from the EditContext instance. If no edited Value exists, the Value from the parent instance or the Device (online or offline) will be returned.

The `StatusCode` indicates whether the Value originates from the Device (`StatusCode Good` defined in IEC 62541) or from an EditContext instance (`StatusCode Good_Edited` defined in IEC 62769-5).

Reading or subscribing a Variable with the `DeviceNodeId` will return the Value from the Device (online or offline).

5.6.5 Writing

Writing to a Variable with the `ContextNodeId` modifies the Value in the EditContext instance.

Writing to a Variable with the `DeviceNodeId` modifies the Value in the Device (online or offline). Any edited Values for this Variable in the addressed EditContext instance or its parents will be reset.

5.6.6 Writing dominant and dependent Variables

Dependencies between dominant and dependent Variables are only evaluated when writing to the online Device, not in the EditContext. The following rules support FDI Clients when working with such Variables in the EditContext:

a) Change to the dominant Variable:

1) A change is made in Online EditContext:

- The Client monitoring Variables in the EditContext will get
 - Good_Edited for dominant Variable
 - Uncertain_DominantValueChanged for dependent Variable.
- If dependent and dominant Variables have been modified in EditContext they all shall have status Good_Edited.
 - It is recommended to disallow writes to dependent Variables until after the dominant Variable is applied to the Device.

2) A change is made in Online EditContext and then to the online Device before the change in the EditContext is applied.

- The Value in the EditContext will be cleared.
- The Client monitoring the dominant value and any dependent Variables in the EditContext will get the StatusCode “Good” with no sub-status.

3) Offline is the same as Online except as follows:

- The dependent Variables will remain writable before the dominant Variable is applied to the Device.

b) Change to a dependent Variable:

1) A change is made in Online EditContext:

- The Client monitoring Variables in the EditContext will get
 - Good_Edited for the changed dependent Variable
 - Good_DependentValueChanged for the dominant Variable.
- If dependent and dominant Variables have been modified in EditContext they all shall have status Good_Edited.
 - It is recommended to disallow writes to the dominant Variable until after the dependent Variable is applied to the Device.

2) A dependent Variable is changed in the Online Device before the change in the EditContext is applied

- The Value in the EditContext will be cleared.
- The Client monitoring Variables in the EditContext will get
 - Good for the cleared dependent Variable and for the dominant Variable.

3) The dominant Variable is changed in the Online Device before the change of the dependent Variable in the EditContext is applied

- The Value of the dependent Variable in the EditContext will be cleared.
- The Client monitoring the dominant value and any dependent Variables in the EditContext will get the StatusCode “Good” with no sub-status.

4) Offline is the same as Online except as follows:

- Dominant and dependent Variables are individually writable

5.6.7 Actions (EDD METHODS)

Before invoking Actions, the Client has to register the ActionSet Node of the Device. The NodeId of this Node has to be specified when calling InvokeAction.

Calling InvokeAction with the ContextNodeId of the ActionSet Node associates it with the proper EditContext instance. The Server will implicitly create an EditContext instance for the invoked Action. This is illustrated in Figure 4.

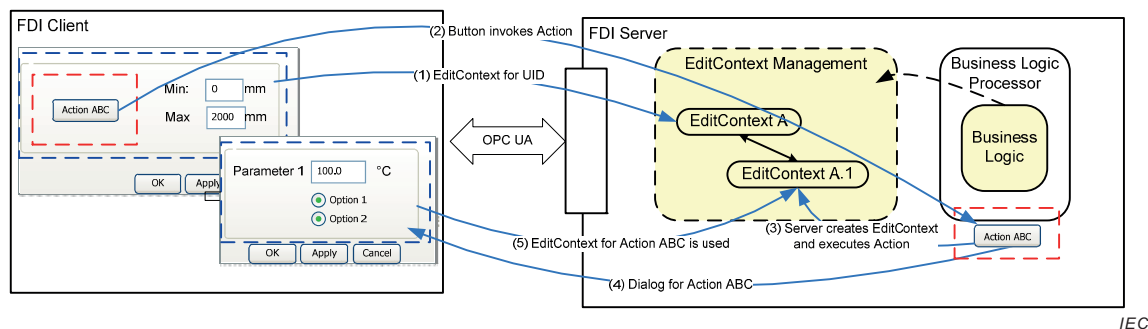


Figure 4 – EditContext for EDD Methods

The EDD METHOD represented by the Action uses builtins to modify Values in the EditContext or the Device, to synchronize changes with the underlying cache or to discard them.

If the Action execution fails, the EditContext for the Action is discarded.

5.6.8 UIDs

The UID Interpreter in the Client calls GetEditContext before it calls up a top-level UID. Additional EditContexts for dialogs may be instantiated by the Server and passed to the Client inside each UID document. See IEC 62769-2 for the UID Schema and the handling of an EditContext in the UID Interpreter.

5.6.9 Synchronization

A Lock has to be created before the first Value is written to an EditContext.

Locking is also required when writing directly to the Device.

5.7 Reading

5.7.1 General

The Read service specified in IEC 62541-4 can be used to read a single value or multiple values from a single device or multiple devices. If a Read service request specifies multiple values are to be read, the values are read in the order they appear in the service request.

All values that are returned to the FDI Client as result of a Read service request shall be unscaled.

A failure encountered while reading a single value shall not abort the read process; all values shall be read. Each value returned contains a status indicating success or failure. Standard OPC UA service status information is returned by the FDI Server as a result of the service calls as specified in 6.2.

An FDI Package can define read actions that are executed by the FDI Server during read service requests. These actions shall not require user interaction; they are strictly intended to be used for Business Logic processing. Any read action that eventually requires user interaction will not perform the built-in but will return an error if possible. The following read actions may be defined in an FDI Package:

- Pre-read Actions
- Post-read Actions