

DIN EN IEC 61804-3



ICS 35.060; 35.240.50

Einsprüche bis 2021-01-13
Vorgesehen als Ersatz für
DIN EN 61804-3:2016-04

Entwurf

**Funktionsbausteine für die Prozessautomation und elektronische
Gerätebeschreibungssprache –
Teil 3: Syntax und Semantik der elektronischen
Gerätebeschreibungssprache (EDDL)
(IEC 65E/631/CDV:2018);
Englische Fassung prEN IEC 61804-3:2018**

Function Blocks (FB) for process control and Electronic Device Description
Language (EDDL) –
Part 3: EDDL syntax and semantics
(IEC 65E/631/CDV:2018);
English version prEN IEC 61804-3:2018

Blocs Fonctionnels (FB) pour les procédés industriels et le Langage de Description
Electronique de Produit (EDDL) –
Partie 3: Sémantique et syntaxe EDDL
(IEC 65E/631/CDV:2018);
Version anglaise prEN IEC 61804-3:2018

Anwendungswarnvermerk

Dieser Norm-Entwurf mit Erscheinungsdatum 2020-11-13 wird der Öffentlichkeit zur Prüfung und
Stellungnahme vorgelegt.

Weil die beabsichtigte Norm von der vorliegenden Fassung abweichen kann, ist die Anwendung dieses
Entwurfs besonders zu vereinbaren.

Stellungnahmen werden erbeten

- vorzugsweise online im Norm-Entwurfs-Portal von DIN unter www.din.de/go/entwuerfe bzw. für Norm-Entwürfe der DKE auch im Norm-Entwurfs-Portal der DKE unter www.entwuerfe.normenbibliothek.de, sofern dort wiedergegeben;
- oder als Datei per E-Mail an dke@vde.com möglichst in Form einer Tabelle. Die Vorlage dieser Tabelle kann im Internet unter www.din.de/go/stellungnahmen-norm-entwuerfe oder für Stellungnahmen zu Norm-Entwürfen der DKE unter www.dke.de/stellungnahme abgerufen werden;
- oder in Papierform an die DKE Deutsche Kommission Elektrotechnik Elektronik Informationstechnik in DIN und VDE oder Stresemannallee 15, 60596 Frankfurt am Main.

Die Empfänger dieses Norm-Entwurfs werden gebeten, mit ihren Kommentaren jegliche relevanten
Patentrechte, die sie kennen, mitzuteilen und unterstützende Dokumentationen zur Verfügung zu stellen.

Gesamtumfang 382 Seiten

DKE Deutsche Kommission Elektrotechnik Elektronik Informationstechnik in DIN und VDE

Nationales Vorwort

Die englische Fassung des europäischen Dokuments prEN IEC 61804-3:2018 „Funktionsbausteine für die Prozessautomation und elektronische Gerätebeschreibungssprache – Teil 3: Elektronische Gerätebeschreibungssprache (EDDL)“ (Entwurf in der Umfrage) ist unverändert in diesen Norm-Entwurf übernommen worden.

Die Internationale Elektrotechnische Kommission (IEC) und das Europäische Komitee für Elektrotechnische Normung (CENELEC) haben vereinbart, dass ein auf IEC-Ebene erarbeiteter Entwurf für eine Internationale Norm zeitgleich (parallel) bei IEC und CENELEC zur Umfrage (CDV-Stadium) und Abstimmung als FDIS (en: Final Draft International Standard) bzw. Schluss-Entwurf für eine Europäische Norm gestellt wird, um eine Beschleunigung und Straffung der Normungsarbeit zu erreichen. Dem entsprechend ist das internationale Dokument IEC 65E/631/CDV:2018 „Function Blocks (FB) for process control and Electronic Device Description Language (EDDL) – Part 3: EDDL syntax and semantics“ unverändert in den Entwurf prEN IEC 61804-3:2018 übernommen worden.

Das internationale Dokument wurde vom SC 65E „Devices and integration in enterprise systems“ der Internationalen Elektrotechnischen Kommission (IEC) erarbeitet und unterliegt dem Copyright der IEC . Das Dokument wurde den nationalen Komitees zur Stellungnahme vorgelegt.

Bei der Abstimmung zu dem Europäischen Schluss-Entwurf bei CENELEC und dem Internationalen Schluss-Entwurf bei IEC [Final Draft International Standard (FDIS)] sind jeweils nur „JA/NEIN“-Entscheidungen möglich, wobei „NEIN“-Entscheidungen fundiert begründet werden müssen. Dokumente, die bei CENELEC als Europäische Norm angenommen und ratifiziert werden, sind unverändert als Deutsche Normen zu übernehmen.

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

Aktuelle Informationen zu diesem Dokument können über die Internetseiten von DKE (www.dke.de) und DIN (www.din.de) durch eine Suche nach der Dokumentennummer aufgerufen werden.

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 des vorliegenden Norm-Entwurfs der englischen Sprache als Fachsprache bedienen, wird die englische Fassung der IEC 65E/631/CDV veröffentlicht. Zu deren Abschnitt 1, der den Anwendungsbereich festlegt und deren Abschnitt 3, der die Begriffe festlegt, wurde eine Übersetzung angefertigt und als informativer Nationaler Anhang NA des vorliegenden Norm-Entwurfs hinzugefügt. Für einige der verwendeten Begriffe existieren keine gebräuchlichen deutschen Benennungen, da sich die deutschen Anwender in der Regel ebenfalls der englischen Benennungen bedienen. Dieser Norm-Entwurf steht nicht in unmittelbarem Zusammenhang mit Rechtsvorschriften und ist nicht als Sicherheitsnorm anzusehen.

Es ist vorgesehen, auch bei der entsprechenden zukünftigen Deutschen Norm auf die deutsche Sprachfassung zu verzichten und diese in der englischsprachigen Fassung zu veröffentlichen.

Es wird auf die Möglichkeit hingewiesen, dass einige Elemente dieses Dokuments Patentrechte berühren können. IEC ist nicht dafür verantwortlich, einige oder alle diesbezüglichen Patentrechte zu identifizieren.

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.

Änderungen

Gegenüber DIN EN 61804-3:2016-04 wurden folgende Änderungen vorgenommen:

- a) Kommunikationsprofile ISA100 und GPE wurden hinzugefügt;
- b) EDD Identifikationsinformation hat ein neues LAYOUT_TYPE Attribut;
- c) neue Konstruktion SEMANTIC_MAP wurden hinzugefügt;
- d) CLASS Attribut-Werte LOCAL_A und LOCAL_B wurden hinzugefügt;
- e) erweiterte LIST-Funktionalität zur Unterstützung der geräteverwalteten Listen.

Nationaler Anhang NA (informativ)

Auszugsweise Übersetzung

Reihenfolge und Inhalt der folgenden Abschnitte sind identisch mit denen in den jeweiligen Abschnitten der Englischen Fassung.

Einleitung

Die Gerätebeschreibungssprache (EDDL) füllt die Lücke zwischen dem in IEC 61804-2 festgelegten Spezifikationskonzept für Funktionsbausteine und der Produktimplementierung. Sie ermöglicht es den Herstellern, bei auf verschiedenen Technologien und Plattformen basierenden Geräten ein und dasselbe Beschreibungsverfahren anzuwenden. Diese Aspekte sind in Bild 1 dargestellt.

IEC 61804 besteht aus den folgenden Teilen unter dem allgemeinen Titel „Funktionsbausteine für die Prozessautomation und elektronische Gerätebeschreibungssprache“:

- Teil 2: Festlegung des Funktionsbausteinkonzepts;
- Teil 3: Elektronische Gerätebeschreibungssprache (EDDL);
- Teil 4: Interpretation von Gerätebeschreibungen;
- Teil 5: Bibliothek vorgefertigter Unterprogramme;
- Teil 6: Einhaltung der Anforderungen an integrierende Feldbusgeräte in ingenieurstechnischen Werkzeugen für Feldgeräte.

In den vorliegenden Teil von IEC 61804 wurden einige Teile der IEC/TS 61804-1:2003 integriert, die im Januar 2013 zurückgezogen wurde.

Die Gerätebeschreibungssprache kann auch zur Beschreibung von Produkteigenschaften in anderen Bereichen, wie z. B. der Industrieautomation, angewendet werden. Die Industrieautomation kann Geräte wie z. B. allgemeine digitale und analoge Eingabe/Ausgabe-Module, Bewegungssteuerungen, Mensch-Maschine-Schnittstellen, Sensoren, (echte) Regler, Kodierer, Hydraulikventile und programmierbare Regler umfassen.

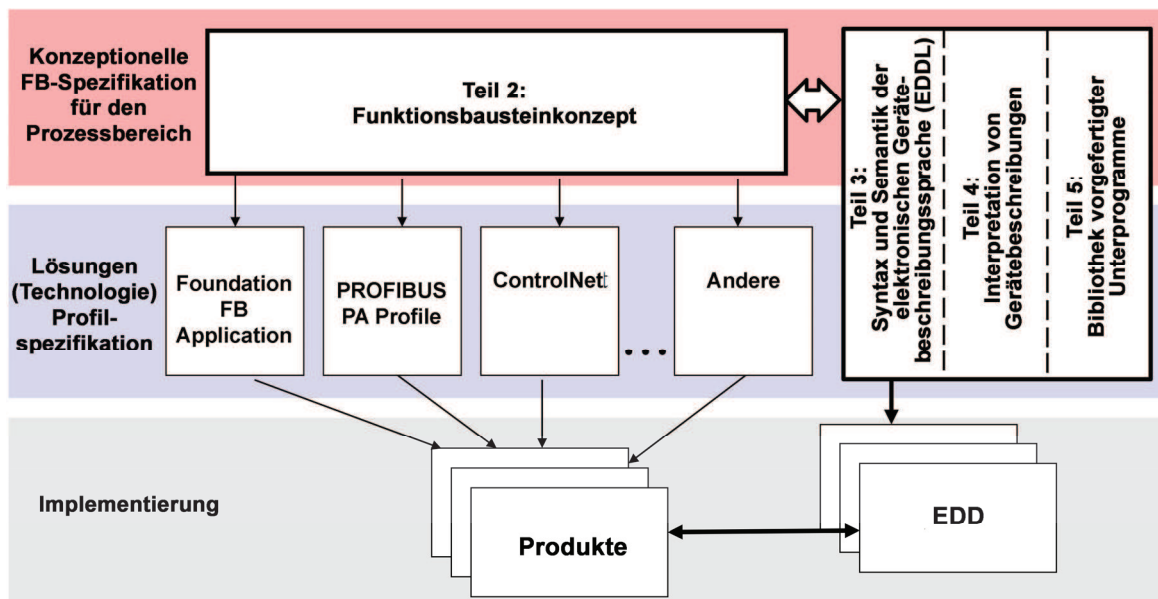


Bild 1 – Stellung der Norm IEC 61804 in Bezug auf andere Normen und Produkte

1 Anwendungsbereich

In diesem Teil von IEC 61804 wird die Technologie für die elektronische Gerätebeschreibungssprache (en: Electronic Device Description Language, EDDL) festgelegt, die die Integration von Einzelheiten realer Produkte mit Hilfe der Werkzeuge bzw. Tools des ingenieurstechnischen Lebenszyklus ermöglicht.

In diesem Teil von IEC 61804 wird EDDL als eine generische Sprache zur Beschreibung der Eigenschaften von Komponenten von Automationssystemen festgelegt. EDDL eignet sich zur Beschreibung von:

- Geräteparametern und ihren Abhängigkeiten;
- Gerätefunktionen, z. B. Simulationsmodus, Kalibrierung;
- grafischen Darstellungen, z. B. Menüs;
- Interaktionen mit Leitgeräten;
- grafischen Darstellungen:
 - erweiterten Benutzerschnittstellen,
 - grafischen Darstellungssystemen;
- permanenten Datenspeichern.

EDDL wird zur Erstellung einer elektronischen Gerätebeschreibung (EDD) verwendet, z. B. für konkrete Geräte, allgemein nutzbare Profile oder Bibliotheken. Diese EDD wird mit entsprechenden Tools zur Erzeugung eines interpretativen Codes angewendet, um den Umgang mit den Parametern, die Bedienung und die Überwachung von Komponenten von Automationssystemen, wie z. B. abgesetzten E/A, Steuer- und Regleinrichtungen, Sensoren und programmierbaren Reglern, zu unterstützen. Die Implementierung dieser Tools ist nicht Gegenstand dieser Norm.

In diesem Teil von IEC 61804 wird die semantische und lexikalische Struktur unabhängig von der Syntax festgelegt. In Anhang A wird eine spezielle Syntax definiert, das semantische Modell kann aber auch mit einer anderen Syntax angewendet werden.

Teil 4 der IEC 61804 spezifiziert die EDD-Interpretation für EDD-Anwendungen und EDDs zur Unterstützung der EDD-Interoperabilität.

Teil 5 der IEC 61804 spezifiziert die EDDL Builtin Bibliothek und stellt die Profile der verschiedenen Feldbusse zur Verfügung.

3 Begriffe, Abkürzungen und Akronyme

3.1 Begriffe

Für die Anwendung dieses Dokuments gelten die Begriffe nach IEC 60050-351, IEC 61804-2 und die folgenden Begriffe.

ISO und IEC stellen terminologische Datenbanken für die Verwendung in der Normung unter den folgenden Adressen bereit:

- IEC Electropedia: verfügbar unter <http://www.electropedia.org/>
- ISO Online Browsing Platform: verfügbar unter <http://www.iso.org/obp>

3.1.1

Builtin

(en: Builtin)

vordefiniertes Unterprogramm, das von der EDD-Anwendung z. B. für Kommunikations- und Anzeigezwecke ausgeführt wird

3.1.2

Gerät

(en: device)

unabhängige physische Einheit, die fähig ist, eine oder mehrere spezifische Funktionen in einem speziellen Kontext auszuführen und die durch ihre Schnittstellen abgegrenzt ist

[QUELLE: IEC 61499-1]

3.1.3

EDD-Anwendung

(en: EDD application)

Programm, das die EDD oder eine beliebige übersetzte Form der EDD verwendet und Funktionen wie die Kommunikationsdarstellung, die Datendarstellung, die graphische Darstellung usw. bereitstellt

3.1.4

EDDL-Prozessor

(en: EDDL processor)

Prozessor oder Programm, der/das die EDD in eine ausführbare Form übersetzt, die von einer EDD-Anwendung verarbeitet werden kann

3.1.5

EDDL-Profil

(en: EDDL profile)

Auswahl der unterstützten Elemente der lexikalischen EDDL-Struktur, einschließlich der Syntaxdefinitionen für eine Anzahl von spezifischen Konsortien

3.1.6

elektronische Gerätebeschreibungssprache

EDDL

(en: Electronic Device Description Language)

formale Sprache zur Beschreibung von Geräten und Kommunikationskomponenten

3.1.7

elektronische Gerätebeschreibung

EDD

(en: Electronic Device Description)

Datensammlung, welche die Geräteparameter, deren Abhängigkeiten und graphische Darstellung sowie eine Beschreibung der übermittelten Datensätze enthält

Anmerkung 1 zum Begriff: Die elektronische Gerätebeschreibung wird unter Verwendung der EDDL erstellt.

3.1.8

EDD-Compiler

(en: Electronic Device Description Compiler)

Compiler, der die EDD-Quelle in ein internes Format übersetzt, das vom EDD-Interpreter verwendet wird

3.1.9

EDD-Interpreter

EDDI

(en: Electronic Device Description Interpreter)

Interpreter, der die EDD-Quelle oder ein vom EDDL-Compiler vorgegebenes internes Format verwendet, um dem EDD-Anwender die EDD-Informationen bereitzustellen

3.1.10

Funktionsbaustein (Funktionsbausteininstanz)

(en: function block (function block instance))

funktionale Software-Einheit, die eine individuelle, mit einem Namen versehene Kopie einer Datenstruktur und zugeordnete Operationen umfasst, die durch einen entsprechenden FB-Typ festgelegt sind

[QUELLE: IEC 61804-2]

Anmerkung 1 zum Begriff: Typische Operationen eines FB schließen eine Veränderung der Datenwerte in seiner Datenstruktur ein.

3.1.11

Hardware

(en: hardware)

physische Einrichtung im Gegensatz zu Programmen, Prozeduren, Regeln und der zugehörigen Dokumentation

3.1.12

Vorprozessor

(en: preprocessor)

Teil der Laufzeitumgebung, der die Informationen in einer EDD umwandelt

3.1.13

Anweisungen für den Vorprozessor

(en: preprocessor directives)

Beschreibung der Bedingungen für die Filterung des EDD-Codes vor der Kompilierung oder Interpretation

Anmerkung 1 zum Begriff: Beispielsweise eine Anweisung für den Vorprozessor zur Festlegung von Namen für Konstanten oder zum Schreiben von Makros für ein vereinfachtes Lesen von Code.

3.1.14

Software

(en: software)

geistige Schöpfung, die die Programme, Prozeduren, Regeln und alle zugehörigen Dokumentationen umfasst, die zum Betrieb eines Systems gehören

[QUELLE: IEC 61499-1]

3.2 Abkürzungen und Akronyme

ASCII	Amerikanischer Standard-Code für den Informationsaustausch nach ISO/IEC 10646-1 (en: American Standard Code for Information Interchange according to ISO/IEC 10646-1)
CP	Kommunikationsprofil (en: Communication Profile)
CPF	Kommunikationsprofilfamilie (en: Communication Profile Family)
CS	Kommunikationsserver (en: Communication Server)
EDD	elektronische Gerätebeschreibung (en: Electronic Device Description)
EDDL	elektronische Gerätebeschreibungssprache (en: Electronic Device Description Language)
EUC	Erweiterter Unicode nach ISO/IEC 10646 (en: Extended Unicode according to ISO/IEC 10646)
FB	Funktionsbaustein (en: Function Block)
FF	Foundation Fieldbus (en: Foundation Fieldbus)
GPE	Allgemeine Protokoll-Erweiterungen (en: Generic Protocol Extension)
HART	Highway Addressable Remote Transducer (en: Highway Addressable Remote Transducer)
HMI	Mensch-Maschine Schnittstelle (en: Human Machine Interface)

E DIN EN IEC 61804-3:2020-12

I/O	Eingang/Ausgang (en: Input/Output)
ID	Identifizier (en: Identifier)
ISA100	ISA100 (en: ISA100_Wireless™)
OSI	Open Systems Interconnection (en: Open Systems Interconnection)
PI	PROFIBUS und PROFINET International (en: PROFIBUS and PROFINET International)
PB	PROFIBUS (en: PROFIBUS)
PDU	Protocol Data Unit (en: Protocol Data Unit)
PN	PROFINET (en: PROFINET)
PNO	PROFIBUS Nutzerorganisation (en: PROFIBUS Nutzerorganisation)
QM	Qualitätsmanagement (en: Quality Management)

CONTENTS

1	Scope	6
2	Normative references	6
3	Terms, definitions, abbreviated terms and acronyms	7
3.1	Terms and definitions	7
3.2	Abbreviated terms and acronyms.....	9
4	Conformance statement.....	9
5	EDD and EDDL model	10
5.1	Overview of EDD and EDDL	10
5.2	EDD architecture	10
5.3	Concepts of EDD.....	10
5.4	Principles of the EDD development process	10
5.4.1	General	10
5.4.2	EDD source generation.....	11
5.4.3	EDD preprocessing.....	11
5.4.4	EDD compilation	11
5.5	Interrelations between the lexical structure and formal definitions	11
5.6	Builtins	11
5.7	Profiles.....	12
6	Conventions	12
6.1	General	12
6.2	Conventions for basic construct description.....	12
6.3	Conventions for lexical structure.....	13
7	Overview	14
7.1	EDDL features.....	14
7.2	Syntax representation.....	14
7.3	EDD language elements	14
7.4	Basic construction elements	15
7.4.1	General	15
7.4.2	AXIS.....	15
7.4.3	BLOB.....	15
7.4.4	BLOCK_A.....	15
7.4.5	BLOCK_B.....	16
7.4.6	CHART	16
7.4.7	COLLECTION.....	16
7.4.8	COMMAND.....	16
7.4.9	COMPONENT.....	17
7.4.10	COMPONENT_FOLDER.....	17
7.4.11	COMPONENT_REFERENCE.....	18
7.4.12	COMPONENT_RELATION.....	18
7.4.13	EDIT_DISPLAY	18
7.4.14	FILE	18
7.4.15	GRAPH.....	19
7.4.16	GRID	19
7.4.17	IMAGE.....	19
7.4.18	IMPORT	19

7.4.19	INTERFACE	19
7.4.20	LIKE	19
7.4.21	LIST	20
7.4.22	MENU	20
7.4.23	METHOD	21
7.4.24	PLUGIN	21
7.4.25	RECORD	21
7.4.26	REFERENCE_ARRAY	21
7.4.27	Relations	22
7.4.28	RESPONSE_CODES	23
7.4.29	SOURCE	23
7.4.30	TEMPLATE	23
7.4.31	VALUE_ARRAY	23
7.4.32	VARIABLE	24
7.4.33	VARIABLE_LIST	24
7.4.34	WAVEFORM	24
7.5	Common attributes	25
7.6	Special elements	25
7.7	Rules for instances	25
7.8	Rules for a list of VARIABLES	26
8	EDD identification information	26
8.1	General structure	26
8.2	DD_REVISION	27
8.3	DEVICE_REVISION	27
8.4	DEVICE_TYPE	28
8.5	EDD_PROFILE	28
8.6	EDD_VERSION	29
8.7	LAYOUT_TYPE	29
8.8	MANUFACTURER	29
8.9	MANUFACTURER_EXT	30
9	Data modelling	30
9.1	VARIABLE	30
9.1.1	General structure	30
9.1.2	CLASS	32
9.1.3	CONSTANT_UNIT	35
9.1.4	DEFAULT_VALUE	36
9.1.5	INITIAL_VALUE	36
9.1.6	POST_EDIT_ACTIONS	37
9.1.7	POST_READ_ACTIONS	37
9.1.8	POST_RQSTUPDATE_ACTIONS	38
9.1.9	POST_USERCHANGE_ACTIONS	38
9.1.10	POST_WRITE_ACTIONS	39
9.1.11	PRE_EDIT_ACTIONS	39
9.1.12	PRE_READ_ACTIONS	39
9.1.13	PRE_WRITE_ACTIONS	40
9.1.14	REFRESH_ACTIONS	40
9.1.15	TYPE	40
9.2	REFERENCE_ARRAY	59
9.2.1	General structure	59

9.2.2	ELEMENTS	62
9.2.3	item-type	62
9.3	VALUE_ARRAY	62
9.3.1	General structure	62
9.3.2	NUMBER_OF_ELEMENTS	63
9.3.3	TYPE	64
9.4	LIST	64
9.4.1	General structure	64
9.4.2	CAPACITY	66
9.4.3	COUNT	66
9.4.4	TYPE	67
9.5	COLLECTION	67
9.5.1	General structure	67
9.5.2	item-type	68
9.6	FILE	69
9.6.1	General structure	69
9.6.2	IDENTITY	70
9.6.3	ON_UPDATE_ACTIONS	71
9.6.4	SHARED	71
9.7	Relations	72
9.7.1	REFRESH	72
9.7.2	SEMANTIC_MAP	73
9.7.3	UNIT	75
9.7.4	WRITE_AS_ONE	75
9.8	BLOB	76
9.8.1	General structure	76
9.8.2	IDENTITY	77
10	Communication	77
10.1	COMMAND	77
10.1.1	General structure	77
10.1.2	Command addressing	79
10.1.3	API	79
10.1.4	BLOCK_B	80
10.1.5	HEADER	80
10.1.6	INDEX	80
10.1.7	NUMBER	80
10.1.8	OPERATION	81
10.1.9	SLOT	82
10.1.10	SUB_SLOT	82
10.1.11	TRANSACTION	82
10.2	RECORD	89
10.3	RESPONSE_CODES	90
10.4	VARIABLE_LIST	90
11	User interface	91
11.1	MENU	91
11.1.1	General structure	91
11.1.2	ACCESS	92
11.1.3	EXIT_ACTIONS	93
11.1.4	INIT_ACTIONS	93

11.1.5	ITEMS	93
11.1.6	POST_EDIT_ACTIONS.....	95
11.1.7	POST_READ_ACTIONS	96
11.1.8	POST_WRITE_ACTIONS	96
11.1.9	PRE_EDIT_ACTIONS	96
11.1.10	PRE_READ_ACTIONS	97
11.1.11	PRE_WRITE_ACTIONS.....	97
11.1.12	STYLE	97
11.1.13	Sequence diagrams for actions.....	98
11.2	EDIT_DISPLAY	100
11.2.1	General structure.....	100
11.2.2	EDIT_ITEMS	101
11.2.3	DISPLAY_ITEMS	102
11.2.4	POST_EDIT_ACTIONS.....	102
11.2.5	PRE_EDIT_ACTIONS	102
11.3	IMAGE	103
11.3.1	General structure.....	103
11.3.2	LINK	104
11.3.3	PATH.....	105
11.4	GRID	105
11.4.1	General structure.....	105
11.4.2	ORIENTATION	108
11.4.3	VECTORS	108
11.5	CHART	109
11.5.1	General structure.....	109
11.5.2	CYCLE_TIME	117
11.5.3	LENGTH	118
11.5.4	TYPE	118
11.5.5	Sequence diagram for actions	121
11.5.6	CHART Legend/Help rules.....	121
11.6	SOURCE	123
11.6.1	General structure.....	123
11.6.2	EXIT_ACTIONS	125
11.6.3	INIT_ACTIONS	125
11.6.4	REFRESH_ACTIONS	125
11.6.5	Y_AXIS.....	126
11.7	AXIS.....	126
11.7.1	General structure.....	126
11.7.2	MIN_VALUE, MAX_VALUE	127
11.7.3	SCALING	128
11.7.4	CONSTANT_UNIT	128
11.8	GRAPH	128
11.8.1	General structure.....	128
11.8.2	CYCLE_TIME	137
11.8.3	X_AXIS.....	138
11.8.4	Sequence diagram for actions	138
11.9	WAVEFORM.....	139
11.9.1	General structure.....	139
11.9.2	EXIT_ACTIONS	139

11.9.3	INIT_ACTIONS	140
11.9.4	KEY_POINTS	141
11.9.5	REFRESH_ACTIONS	142
11.9.6	TYPE	142
11.9.7	Y_AXIS	145
11.10	PLUGIN	145
11.10.1	General structure	145
11.10.2	UUID	146
12	Business Logic	146
12.1	METHOD	146
12.1.1	General structure	146
12.1.2	ACCESS	148
12.1.3	CLASS	148
12.1.4	TYPE	149
13	Device modelling	150
13.1	BLOCK_A	150
13.1.1	General structure	150
13.1.2	AXIS_ITEMS	152
13.1.3	CHARACTERISTICS	153
13.1.4	CHARTS	153
13.1.5	CHART_ITEMS	153
13.1.6	COLLECTION_ITEMS	154
13.1.7	EDIT_DISPLAY_ITEMS	154
13.1.8	FILES	154
13.1.9	FILE_ITEMS	155
13.1.10	GRAPHS	155
13.1.11	GRAPH_ITEMS	155
13.1.12	GRIDS	156
13.1.13	GRID_ITEMS	156
13.1.14	IMAGE_ITEMS	156
13.1.15	ITEM_ARRAY_ITEMS	157
13.1.16	LISTS	157
13.1.17	LIST_ITEMS	157
13.1.18	LOCAL_PARAMETERS	157
13.1.19	MENUS	158
13.1.20	MENU_ITEMS	158
13.1.21	METHODS	158
13.1.22	METHOD_ITEMS	159
13.1.23	PARAMETERS	159
13.1.24	PARAMETER_LISTS	159
13.1.25	PLUGINS	160
13.1.26	PLUGIN_ITEMS	160
13.1.27	REFRESH_ITEMS	161
13.1.28	SEMANTIC_MAP_ITEMS	161
13.1.29	SOURCE_ITEMS	161
13.1.30	UNIT_ITEMS	161
13.1.31	WAVEFORM_ITEMS	162
13.1.32	WRITE_AS_ONE_ITEMS	162
13.2	BLOCK_B	162

13.2.1	General structure.....	162
13.2.2	NUMBER.....	163
13.2.3	TYPE.....	163
13.3	LIKE.....	164
13.4	IMPORT.....	164
13.4.1	General structure.....	164
13.5	TEMPLATE.....	168
13.5.1	General structure.....	168
13.5.2	DEFAULT_VALUES.....	169
13.6	COMPONENT.....	169
13.6.1	General structure.....	169
13.6.2	BYTE_ORDER.....	170
13.6.3	CAN_DELETE.....	171
13.6.4	CHECK_CONFIGURATION.....	171
13.6.5	COMPONENT_RELATIONS.....	171
13.6.6	CONNECTION_POINT.....	172
13.6.7	DETECT.....	172
13.6.8	EDD.....	172
13.6.9	INITIAL_VALUES.....	173
13.6.10	REDUNDANCY.....	173
13.6.11	SCAN.....	173
13.6.12	SCAN_LIST.....	174
13.6.13	PRODUCT_URI.....	174
13.7	COMPONENT_FOLDER.....	174
13.8	COMPONENT_REFERENCE.....	175
13.9	COMPONENT_RELATION.....	176
13.9.1	General structure.....	176
13.9.2	COMPONENTS.....	177
13.9.3	RELATION_TYPE.....	178
13.9.4	ADDRESSING.....	179
13.9.5	MAXIMUM_NUMBER.....	179
13.9.6	MINIMUM_NUMBER.....	179
13.9.7	REQUIRED_INTERFACE.....	180
13.10	INTERFACE.....	180
13.10.1	General structure.....	180
13.10.2	DECLARATION.....	180
14	Common attributes.....	181
14.1	CLASSIFICATION.....	181
14.2	COMPONENT_PARENT.....	183
14.3	COMPONENT_PATH.....	183
14.4	DEFINITION.....	183
14.5	EMPHASIS.....	184
14.6	HANDLING.....	185
14.7	HEIGHT.....	185
14.8	HELP.....	186
14.9	LABEL.....	187
14.10	LINE_COLOR.....	187
14.11	LINE_TYPE.....	188
14.12	MEMBERS.....	

14.13 PRIVATE	190
14.14 PROTOCOL	190
14.15 RESPONSE_CODES	191
14.16 SUPPLIED_INTERFACE	191
14.17 VALIDITY	192
14.18 VISIBILITY	192
14.19 WIDTH	193
14.20 WRITE_MODE	193
15 Redefinitions	193
15.1 Overview	193
15.2 AXIS	196
15.3 BLOB	196
15.4 BLOCK_A	197
15.5 BLOCK_B	199
15.6 CHART	199
15.7 COLLECTION	199
15.8 COMMAND	200
15.9 COMPONENT	200
15.10 COMPONENT_FOLDER	201
15.11 COMPONENT_REFERENCE	201
15.12 COMPONENT_RELATION	202
15.13 EDIT_DISPLAY	202
15.14 FILE	203
15.15 GRAPH	203
15.16 GRID	204
15.17 IMAGE	204
15.18 INTERFACE	205
15.19 LIST	205
15.20 MENU	205
15.21 METHOD	206
15.22 PLUGIN	207
15.23 RECORD	207
15.24 REFERENCE_ARRAY	207
15.25 RESPONSE_CODES	208
15.26 SOURCE	208
15.27 TEMPLATE	209
15.28 VALUE_ARRAY	209
15.29 VARIABLE	210
15.30 VARIABLE_LIST	211
15.31 WAVEFORM	212
16 Referencing	212
16.1 Referencing an EDD instance	212
16.2 Referencing AXIS of a GRAPH, SOURCE, WAVEFORM	213
16.3 Referencing bits of a BIT_ENUMERATED_VARIABLE	213
16.4 Referencing members of a CHART	213
16.5 Referencing members of a COLLECTION	214
16.6 Referencing COMPONENT instances	214
16.7 Referencing COMPONENT types	214
16.8 Referencing members of a FILE	214

16.9	Referencing members of a GRAPH.....	215
16.10	Referencing elements of a LIST.....	215
16.11	Referencing members of a RECORD	216
16.12	Referencing elements of a REFERENCE_ARRAY	216
16.13	Referencing members of a SOURCE	216
16.14	Referencing elements of a VALUE_ARRAY	217
16.15	Referencing members of a VARIABLE_LISTS	217
16.16	Referencing BLOCK_A CHARACTERISTICS	217
16.17	Referencing elements of BLOCK_A LOCAL_PARAMETERS.....	218
16.18	Referencing elements of BLOCK_A PARAMETERS.....	218
16.19	Referencing elements of BLOCK_A PARAMETER_LISTS	218
16.20	Referencing CHARACTERISTICS of specific BLOCK_A instance	218
16.21	Referencing CHARTS of specific BLOCK_A instance	219
16.22	Referencing FILES of specific BLOCK_A instance.....	219
16.23	Referencing GRAPHS of specific BLOCK_A instance	220
16.24	Referencing GRIDS of specific BLOCK_A instance.....	220
16.25	Referencing LISTS of specific BLOCK_A instance	220
16.26	Referencing LOCAL_PARAMETERS of specific BLOCK_A instance	221
16.27	Referencing MENUS of specific BLOCK_A instance	221
16.28	Referencing METHODS of specific BLOCK_A instance	222
16.29	Referencing PARAMETERS of specific BLOCK_A instance	222
16.30	Referencing PLUGINS of specific BLOCK_A instance	222
17	Strings.....	223
17.1	Specifying a string as a string literal	223
17.2	Specifying a string as a string variable	223
17.3	Specifying a string as an enumeration value.....	223
17.4	Specifying a string as a dictionary reference	224
17.5	Referencing HELP and LABEL attributes of EDD instances	224
17.6	String operations	224
18	Conditional expression	225
18.1	General structure	225
18.2	IF Conditional.....	225
18.3	SELECT Conditional.....	226
19	Expression	227
19.1	General structure	227
19.2	Primary expressions	227
19.3	Unary expressions.....	230
19.4	Binary expressions	230
19.4.1	General structure.....	230
19.4.2	Multiplicative operators.....	231
19.4.3	Additive operators	231
19.4.4	Shift operators	231
19.4.5	Relational operators	232
19.4.6	Equality operators	232
19.4.7	Bitwise AND operator (&).....	232
19.4.8	Bitwise XOR operator (^)	232
19.4.9	Bitwise OR operator ()	232
19.4.10	Logical AND operator (&&)	232
19.4.11	Logical OR operator ()	233

19.4.12 Conditional evaluation	233
20 Text dictionary	233
Annex A (normative) EDDL formal definition	234
A.1 EDDL preprocessor	234
A.1.1 General structure	234
A.1.2 Directives	234
A.1.3 Predefined macros	236
A.1.4 NEWLINE characters	237
A.1.5 Comments	237
A.2 Conventions	237
A.2.1 Integer constants	237
A.2.2 Floating-point constants	237
A.2.3 String literals	238
A.2.4 Using language and country codes in string literals	238
A.3 Operators	239
A.4 Keywords	243
A.5 Symbols and symbol scope	247
A.6 Terminals	247
A.7 Formal EDDL syntax	248
A.7.1 General	248
A.7.2 EDD identification information	248
A.7.3 AXIS	250
A.7.4 BLOCK_A and BLOCK_B	251
A.7.5 CHART	255
A.7.6 COLLECTION	256
A.7.7 COMMAND	257
A.7.8 COMPONENT	260
A.7.9 COMPONENT_FOLDER	263
A.7.10 COMPONENT_REFERENCE	263
A.7.11 COMPONENT_RELATION	264
A.7.12 EDIT_DISPLAY	266
A.7.13 FILE	267
A.7.14 GRAPH	267
A.7.15 GRID	267
A.7.16 IMAGE	268
A.7.17 INTERFACE	269
A.7.18 LIST	269
A.7.19 IMPORT	270
A.7.20 LIKE	272
A.7.21 MENU	274
A.7.22 METHOD	275
A.7.23 RECORD	277
A.7.24 REFERENCE_ARRAY	277
A.7.25 Relations	278
A.7.26 RESPONSE_CODES	280
A.7.27 SOURCE	281
A.7.28 TEMPLATE	281
A.7.29 VALUE_ARRAY	282
A.7.30 VARIABLE	282

A.7.31	VARIABLE_LIST	293
A.7.32	WAVEFORM	293
A.7.33	Common attributes	295
A.7.34	Expression	299
A.7.35	C-Grammar	301
A.7.36	Redefinition	305
A.7.37	References	330
A.7.38	PLUGIN	332
A.7.39	BLOB	333
A.7.40	SEMANTIC_MAP	333
A.8	Formal dictionary syntax	334
Annex B (normative)	Data types	335
B.1	METHOD DEFINITION data types	335
B.2	VARIABLE TYPE data types	335
B.2.1	General	335
B.2.2	Coding of data DATE	337
B.2.3	Coding of data DATE_AND_TIME	338
B.2.4	Coding of data DURATION	338
B.2.5	Coding of data TIME	339
B.2.6	Coding of data TIME_VALUE	340
B.2.7	Coding of PACKED_ASCII (6-BIT ASCII) DATA FORMAT	340
B.2.8	Coding of BOOLEAN	341
B.2.9	Coding of BIT_ENUMERATED	341
Annex C (informative)	EDD example	343
C.1	EDD example of a temperature transmitter	343
C.2	EDD example	343
Figure 1	– Position of IEC 61804 in relation to other standards and products	5
Figure 2	– EDD generation process	11
Figure 3	– BLOCK_A	15
Figure 4	– CHART	16
Figure 5	– COLLECTION	16
Figure 6	– COMMAND	17
Figure 7	– COMPONENT	17
Figure 8	– COMPONENT_FOLDER	18
Figure 9	– COMPONENT_REFERENCE	18
Figure 10	– COMPONENT_RELATION	18
Figure 11	– EDIT_DISPLAY	18
Figure 12	– FILE	19
Figure 13	– GRAPH	19
Figure 14	– GRID	19
Figure 15	– IMAGE	19
Figure 16	– LIKE	20
Figure 17	– LIST	20
Figure 18	– MENU	21
Figure 19	– RECORD	21

Figure 20 – REFERENCE_ARRAY	22
Figure 21 – REFRESH	22
Figure 22 – SEMANTIC_MAP	22
Figure 23 – UNIT	23
Figure 24 – WRITE_AS_ONE	23
Figure 25 – SOURCE	23
Figure 26 – VALUE_ARRAY	24
Figure 27 – VARIABLE	24
Figure 28 – VARIABLE_LIST	24
Figure 29 – WAVEFORM	25
Figure 30 – REFERENCE_ARRAY example	61
Figure 31 – Abstract COMMAND definition example	62
Figure 32 – EDD examples of SEMANTIC_MAP	74
Figure 33 – Example of an item mask	85
Figure 34 – Example of a single item mask	86
Figure 35 – Mapping example with a single item mask	86
Figure 36 – Example of multiple item masks	87
Figure 37 – Mapping example with a multiple item mask	87
Figure 38 – INFO qualifier	88
Figure 39 – INDEX qualifier	88
Figure 40 – INFO and INDEX qualifier	88
Figure 41 – Examples of MENU STYLES	98
Figure 42 – MENU activation	100
Figure 43 – EDD example of an IMAGE	104
Figure 44 – EDD example of an IMAGE with the LINK attribute	104
Figure 45 – EDD example of a GRID	108
Figure 46 – Result of the EDD example	108
Figure 47 – Example of a chart with one curve in a dialog	111
Figure 48 – Example of a chart with two SOURCES	112
Figure 49 – Displaying example of a chart with two SOURCES	113
Figure 50 – Example of a chart with three horizontal bars	115
Figure 51 – Displaying example of a chart with three horizontal bars	115
Figure 52 – Example of a chart in a dialog	117
Figure 53 – Trend CHART Types examples	120
Figure 54 – Meter CHART Types examples	120
Figure 55 – CHART activation	121
Figure 56 – Legend / help for SOURCES with single and multiple Variables – Y_ AXIS not defined	122
Figure 57 – Legend / Help for SOURCES Arrangement of multiple SOURCES with shared Y_AXIS	123
Figure 58 – A graph and the visual elements	130
Figure 59 – Example of a graph	133
Figure 60 – Multiple used axes	134
Figure 61 – EDD with device-supported zooming and scrolling	137

Figure 62 – GRAPH activation	139
Figure 63 – EDDL import mechanisms	165
Figure 64 – Example of TEMPLATE usage.....	169
Figure 65 – REQUIRED_RANGES usage.....	178
Figure 66 – NEXT_COMPONENT usage.....	179
Figure C.1 – Example of an operator screen using EDD.....	343
Table 1 – Format for the basic construct table.....	12
Table 2 – Contents of the Basic Construct table.....	12
Table 3 – Attribute usage descriptions	13
Table 4 – Rules for lexical convention.....	13
Table 5 – EDD sources	14
Table 6 – EDD identification information attributes	26
Table 7 – DD_REVISION attribute	27
Table 8 – DEVICE_REVISION attribute.....	28
Table 9 – DEVICE_TYPE attribute	28
Table 10 – EDD_PROFILE attribute	28
Table 11 – EDD_VERSION attribute	29
Table 12 – LAYOUT_TYPE attribute	29
Table 13 – MANUFACTURER attribute	30
Table 14 – MANUFACTURER_EXT attribute.....	30
Table 15 – VARIABLE attributes	31
Table 16 – CLASS attributes.....	33
Table 17 – CONSTANT_UNIT attribute	36
Table 18 – POST_EDIT_ACTIONS, PRE_EDIT_ACTIONS, POST_READ_ACTIONS, PRE_READ_ACTIONS, POST_WRITE_ACTIONS, PRE_WRITE_ACTIONS, REFRESH_ACTIONS attributes	37
Table 19 – POST_USERCHANGE_ACTIONS, POST_RQSTUPDATE_ACTIONS attributes	38
Table 20 – TYPE overview	41
Table 21 – MIN_VALUE , MAX_VALUE attribute	42
Table 22 – TYPE attributes for DOUBLE, FLOAT, INTEGER, UNSIGNED_INTEGER.....	43
Table 23 – Default Display Format Conversion Type	44
Table 24 – Default Width/Precision Based on VARIABLE and Format Conversion Type	44
Table 25 – TYPE attributes for DATE, DATE_AND_TIME, DURATION, TIME, TIME_VALUE.....	48
Table 26 – TYPE attributes for BIT_ENUMERATED	49
Table 27 – size attribute	50
Table 28 – status–class attributes	50
Table 29 – ALL, AO, DV, TV attributes	52
Table 30 – TYPE attributes for ENUMERATED	53
Table 31 – TYPE attributes for INDEX	54
Table 32 – TYPE attributes for ASCII, EUC, PACKED_ASCII, PASSWORD, VISIBLE	55
Table 33 – String handling	56

Table 34 – TYPE attributes for OCTET	57
Table 35 – TYPE attributes for BITSTRING.....	58
Table 36 – REFERENCE_ARRAY attributes	59
Table 37 – ELEMENTS attribute	62
Table 38 – VALUE_ARRAY attributes	63
Table 39 – NUMBER_OF_ELEMENTS attribute	63
Table 40 – TYPE attribute.....	64
Table 41 – LIST attributes.....	65
Table 42 – CAPACITY attribute.....	66
Table 43 – COUNT attribute.....	66
Table 44 – TYPE attribute.....	67
Table 45 – COLLECTION attributes	68
Table 46 – item-type	68
Table 47 – FILE attributes.....	70
Table 48 – IDENTITY attribute	71
Table 49 – ON_UPDATE_ACTIONS attribute.....	71
Table 50 – SHARED attribute.....	71
Table 51 – REFRESH overview.....	72
Table 52 – REFRESH attributes.....	72
Table 53 – SEMANTIC_MAP overview	74
Table 54 – SEMANTIC_MAP attributes	74
Table 55 – UNIT overview.....	75
Table 56 – UNIT attributes	75
Table 57 – WRITE_AS_ONE overview	76
Table 58 – WRITE_AS_ONE attribute	76
Table 59 – BLOB attributes	77
Table 60 – IDENTITY attribute	77
Table 61 – COMMAND attributes	78
Table 62 – API attribute	79
Table 63 – BLOCK_B attribute	80
Table 64 – HEADER attribute.....	80
Table 65 – INDEX attribute	80
Table 66 – NUMBER attribute	81
Table 67 – OPERATION attribute.....	81
Table 68 – PROFIBUS and PROFINET communication mapping.....	81
Table 69 – SLOT attribute.....	82
Table 70 – SUB_SLOT attribute	82
Table 71 – TRANSACTION attributes.....	83
Table 72 – REPLY and REQUEST attributes.....	84
Table 73 – POST_RQSTRECEIVE_ACTIONS attributes	89
Table 74 – RECORD attributes	89
Table 75 – RESPONSE_CODES overview	90
Table 76 – RESPONSE_CODES attributes	90

Table 77 – VARIABLE_LIST attributes	91
Table 78 – MENU attributes	92
Table 79 – ACCESS attribute	92
Table 80 – MENU ITEMS	94
Table 81 – MENU ITEMS qualifiers	94
Table 82 – EXIT_ACTIONS, INIT_ACTIONS, POST_EDIT_ACTIONS, PRE_EDIT_ACTIONS, POST_READ_ACTIONS, PRE_READ_ACTIONS, POST_WRITE_ACTIONS, PRE_WRITE_ACTIONS attributes	96
Table 83 – STYLE attribute	98
Table 84 – EDIT_DISPLAY attributes	101
Table 85 – EDIT_ITEMS attribute	101
Table 86 – DISPLAY_ITEM attributes	102
Table 87 – POST_EDIT_ACTIONS, PRE_EDIT_ACTIONS attribute	103
Table 88 – IMAGE attributes	103
Table 89 – LINK attribute	104
Table 90 – PATH attribute	105
Table 91 – Image formats	105
Table 92 – GRID attributes	106
Table 93 – ORIENTATION attribute	108
Table 94 – VECTORS attribute	109
Table 95 – CHART attributes	110
Table 96 – CYCLE_TIME attribute	117
Table 97 – LENGTH attribute	118
Table 98 – TYPE attribute	119
Table 99 – SOURCE attributes	124
Table 100 – Y_AXIS attribute	126
Table 101 – AXIS attributes	127
Table 102 – MIN_VALUE, MAX_VALUE attribute	127
Table 103 – SCALING attribute	128
Table 104 – GRAPH attributes	130
Table 105 – CYCLE_TIME attribute	138
Table 106 – X_AXIS attribute	138
Table 107 – WAVEFORM attributes	140
Table 108 – EXIT_ACTIONS, INIT_ACTIONS, REFRESH_ACTIONS attribute	141
Table 109 – KEY_POINTS attribute	141
Table 110 – X_VALUES, Y_VALUES attribute	142
Table 111 – TYPE attribute	143
Table 112 – XY attribute	143
Table 113 – YT attribute	144
Table 114 – HORIZONTAL attribute	144
Table 115 – VERTICAL attribute	145
Table 116 – Y_AXIS attribute	145
Table 117 – PLUGIN attributes	146
Table 118 – UUID attribute	146

Table 119 – METHOD attributes	147
Table 120 – Parameter types	147
Table 121 – ACCESS attributes	148
Table 122 – CLASS attributes	148
Table 123 – TYPE attributes	150
Table 124 – BLOCK_A attributes	151
Table 125 – AXIS_ITEMS attribute.....	153
Table 126 – CHARACTERISTIC attribute.....	153
Table 127 – CHARTS attribute	153
Table 128 – CHART_ITEMS attribute.....	154
Table 129 – COLLECTION_ITEMS attribute.....	154
Table 130 – EDIT_DISPLAY_ITEMS attribute	154
Table 131 – FILES attribute	154
Table 132 – FILE_ITEMS attribute	155
Table 133 – GRAPHS attribute.....	155
Table 134 – GRAPH_ITEMS attribute	155
Table 135 – GRIDS attribute	156
Table 136 – GRID_ITEMS attribute	156
Table 137 – IMAGE_ITEMS attribute	156
Table 138 – ITEM_ARRAY_ITEMS attribute.....	157
Table 139 – LISTS attribute	157
Table 140 – LIST_ITEMS attribute	157
Table 141 – MENUS attribute.....	158
Table 142 – MENU_ITEMS attribute.....	158
Table 143 – METHODS attribute	159
Table 144 – METHOD_ITEMS attribute.....	159
Table 145 – PARAMETER attributes	159
Table 146 – PARAMETER_LISTS attributes.....	160
Table 147 – PLUGINS attribute	160
Table 148 – PLUGIN_ITEMS attribute.....	160
Table 149 – REFRESH_ITEMS attribute	161
Table 150 – SEMANTIC_MAP_ITEMS attribute.....	161
Table 151 – SOURCE_ITEMS attribute	161
Table 152 – UNIT_ITEMS attribute	162
Table 153 – WAVEFORM_ITEMS attribute.....	162
Table 154 – WRITE_AS_ONE_ITEMS attribute.....	162
Table 155 – BLOCK_B attributes	163
Table 156 – NUMBER attributes	163
Table 157 – TYPE attributes	163
Table 158 – LIKE overview	164
Table 159 – LIKE attributes.....	164
Table 160 – IMPORT overview.....	166
Table 161 – Importing Device Description	166

Table 162 – TEMPLATE attributes	168
Table 163 – DEFAULT_VALUES attribute	169
Table 164 – COMPONENT attributes	170
Table 165 – BYTE_ORDER attribute	171
Table 166 – CAN_DELETE attribute	171
Table 167 – CHECK_CONFIGURATION attribute	171
Table 168 – COMPONENT_RELATIONS attribute.....	172
Table 169 – CONNECTION_POINT attribute.....	172
Table 170 – DETECT attribute	172
Table 171 – EDD attribute.....	172
Table 172 – INITIAL_VALUES attribute.....	173
Table 173 – REDUNDANCY attribute	173
Table 174 – SCAN attribute	174
Table 175 – SCAN_LIST attribute	174
Table 176 – PRODUCT_URI attribute	174
Table 177 – COMPONENT_FOLDER attributes	175
Table 178 – COMPONENT_REFERENCE attributes	175
Table 179 – COMPONENT_RELATION attributes	176
Table 180 – COMPONENTS attribute.....	177
Table 181 – RELATION_TYPE attribute	178
Table 182 – ADDRESSING attribute	179
Table 183 – MAXIMUM_NUMBER attribute.....	179
Table 184 – MINIMUM_NUMBER attribute	179
Table 185 – REQUIRED_INTERFACE attribute.....	180
Table 186 – INTERFACE attributes.....	180
Table 187 – DECLARATION attribute.....	181
Table 188 – CLASSIFICATION attributes	181
Table 189 – COMPONENT_PARENT attribute	183
Table 190 – COMPONENT_PATH attribute.....	183
Table 191 – DEFINITION attribute	184
Table 192 – EMPHASIS attribute	184
Table 193 – HANDLING attribute	185
Table 194 – HEIGHT/WIDTH attribute.....	185
Table 195 – HELP attribute	186
Table 196 – LABEL attribute	187
Table 197 – LINE_COLOR attribute	187
Table 198 – Standard RGB Color Values	188
Table 199 – LINE_TYPE attribute	189
Table 200 – MEMBERS attributes	189
Table 201 – PRIVATE attributes	190
Table 202 – PROTOCOL attribute.....	191
Table 203 – RESPONSE_CODES attribute	191
Table 204 – SUPPLIED_INTERFACE attribute.....	192

Table 205 – VALIDITY attributes	192
Table 206 – VISIBILITY attributes	192
Table 207 – WRITE_MODE attributes	193
Table 208 – Redefinition attributes	196
Table 209 – Redefinition rules for AXIS attributes	196
Table 210 – Redefinition rules for BLOB attributes	196
Table 211 – Redefinition rules for BLOCK_A attributes	197
Table 212 – Redefinition rules for BLOCK_B attributes	199
Table 213 – Redefinition rules for CHART attributes	199
Table 214 – Redefinition rules for COLLECTION attributes	200
Table 215 – Redefinition rules for COMMAND attributes	200
Table 216 – Redefinition rules for COMPONENT attributes	201
Table 217 – Redefinition rules for COMPONENT_FOLDER attributes	201
Table 218 – Redefinition rules for COMPONENT_REFERENCE attributes	202
Table 219 – Redefinition rules for COMPONENT_RELATION attributes	202
Table 220 – Redefinition rules for EDIT_DISPLAY attributes	203
Table 221 – Redefinition rules for FILE attributes	203
Table 222 – Redefinition rules for GRAPH attributes	203
Table 223 – Redefinition rules for GRID attributes	204
Table 224 – Redefinition rules for IMAGE attributes	204
Table 225 – Redefinition rules for INTERFACE attributes	205
Table 226 – Redefinition rules for LIST attributes	205
Table 227 – Redefinition rules for MENU attributes	206
Table 228 – Redefinition rules for METHOD attributes	206
Table 229 – Redefinition rules for PLUGIN attributes	207
Table 230 – Redefinition rules for RECORD attributes	207
Table 231 – Redefinition rules for REFERENCE_ARRAY attributes	208
Table 232 – Redefinition rules for RESPONSE_CODES attributes	208
Table 233 – Redefinition rules for SOURCE attributes	208
Table 234 – Redefinition rules for TEMPLATE attributes	209
Table 235 – Redefinition rules for VALUE_ARRAY attributes	209
Table 236 – Redefinition rules for VARIABLE attributes	210
Table 237 – Redefinition rules for VARIABLE_LIST attributes	212
Table 238 – Redefinition rules for WAVEFORM attributes	212
Table 239 – Referencing an EDD instance	213
Table 240 – Referencing AXIS of a GRAPH. SOURCE, WAVEFORM	213
Table 241 – Referencing elements of VARIABLE	213
Table 242 – Referencing members of CHART	214
Table 243 – Referencing members of COLLECTION	214
Table 244 – Referencing a COMPONENT instance	214
Table 245 – Referencing a COMPONENT type	215
Table 246 – Referencing members of FILE	215
Table 247 – Referencing members of GRAPH	215

Table 248 – Referencing elements of LIST	215
Table 249 – Referencing elements of RECORD	216
Table 250 – Referencing members of REFERENCE_ARRAY	216
Table 251 – Referencing members of SOURCE	217
Table 252 – Referencing elements of VALUE_ARRAY	217
Table 253 – Referencing members of VARIABLE_LISTS	217
Table 254 – Referencing BLOCK_A CHARACTERISTICS	217
Table 255 – Referencing members of BLOCK_A LOCAL_PARAMETER	218
Table 256 – Referencing members of a BLOCK_A PARAMETERS	218
Table 257 – Referencing members of BLOCK_A PARAMETER_LISTS	218
Table 258 – Referencing CHARACTERISTICS of specific BLOCK_A instance	219
Table 259 – Referencing CHARTS of specific BLOCK_A instance	219
Table 260 – Referencing FILES of specific BLOCK_A instance	219
Table 261 – Referencing GRAPHS of specific BLOCK_A instance	220
Table 262 – Referencing GRIDS of specific BLOCK_A instance	220
Table 263 – Referencing LISTS of specific BLOCK_A instance	221
Table 264 – Referencing LOCAL_PARAMETERS of specific BLOCK_A instance	221
Table 265 – Referencing MENUS of specific BLOCK_A instance	221
Table 266 – Referencing METHODS of specific BLOCK_A instance	222
Table 267 – Referencing PARAMETERS of specific BLOCK_A instance	222
Table 268 – Referencing PLUGINS of specific BLOCK_A instance	223
Table 269 – String as a string literal	223
Table 270 – String as a string variable	223
Table 271 – String as an enumeration value	224
Table 272 – String as a dictionary reference	224
Table 273 – Referencing HELP and LABEL attributes of EDD instances	224
Table 274 – String operation	225
Table 275 – IF conditional	225
Table 276 – SELECT conditional	226
Table 277 – Primary expressions	227
Table 278 – Attribute values of VARIABLES	228
Table 279 – AXIS attribute values	229
Table 280 – BLOB attribute values	230
Table 281 – LIST attribute values	230
Table 282 – ARRAY attribute values	230
Table 283 – Unary expressions	230
Table 284 – Multiplicative operators	231
Table 285 – Additive operators	231
Table 286 – Shift operators	231
Table 287 – Relational operators	232
Table 288 – Equality operators	232
Table 289 – Text dictionary attributes	233
Table A.1 – Conventions for integer constants	237

Table A.2 – Using escape sequences in string literals	238
Table A.3 – Language code examples for string literals	239
Table A.4 – Precedence and associativity for EDDL operators	239
Table A.5 – Operations for VARIABLES or METHOD local variables	241
Table A.6 – EDDL keywords	243
Table B.1 – METHOD DEFINITION data types	335
Table B.2 – VARIABLE TYPEs	335
Table B.3 – DATE coding	338
Table B.4 – DATE_AND_TIME coding	338
Table B.5 – DURATION coding	338
Table B.6 – DURATION coding for ISA100	339
Table B.7 – TIME coding	339
Table B.8 – TIME coding for ISA100	339
Table B.9 – TIME_VALUE coding (four octets)	340
Table B.10 – TIME_VALUE coding (eight octets)	340
Table B.11 – PACKED_ASCII coding	341
Table B.12 – BOOLEAN coding	341
Table B.13 – BIT_ENUMERATED coding	341
Table B.14 – BIT_ENUMERATED coding for FF	342

INTERNATIONAL ELECTROTECHNICAL COMMISSION

**FUNCTION BLOCKS (FB) FOR PROCESS CONTROL AND
ELECTRONIC DEVICE DESCRIPTION LANGUAGE (EDDL) –**

Part 3: EDDL syntax and semantics

FOREWORD

- 1) The International Electrotechnical Commission (IEC) is a worldwide organization for standardization comprising all national electrotechnical committees (IEC National Committees). The object of IEC is to promote international co-operation on all questions concerning standardization in the electrical and electronic fields. To this end and in addition to other activities, IEC publishes International Standards, Technical Specifications, Technical Reports, Publicly Available Specifications (PAS) and Guides (hereafter referred to as “IEC Publication(s)”). Their preparation is entrusted to technical committees; any IEC National Committee interested in the subject dealt with may participate in this preparatory work. International, governmental and non-governmental organizations liaising with IEC also participate in this preparation. IEC collaborates closely with the International Organization for Standardization (ISO) in accordance with conditions determined by agreement between the two organizations.
 - 2) The formal decisions or agreements of IEC on technical matters express, as nearly as possible, an international consensus of opinion on the relevant subjects since each technical committee has representation from all interested IEC National Committees.
 - 3) IEC Publications have the form of recommendations for international use and are accepted by IEC National Committees in that sense. While all reasonable efforts are made to ensure that the technical content of IEC Publications is accurate, IEC cannot be held responsible for the way in which they are used or for any misinterpretation by any end user.
 - 4) In order to promote international uniformity, IEC National Committees undertake to apply IEC Publications transparently to the maximum extent possible in their national and regional publications. Any divergence between any IEC Publication and the corresponding national or regional publication shall be clearly indicated in the latter.
 - 5) IEC itself does not provide any attestation of conformity. Independent certification bodies provide conformity assessment services and, in some areas, access to IEC marks of conformity. IEC is not responsible for any services carried out by independent certification bodies.
 - 6) All users should ensure that they have the latest edition of this publication.
 - 7) No liability shall attach to IEC or its directors, employees, servants or agents including individual experts and members of its technical committees and IEC National Committees for any personal injury, property damage or other damage of any nature whatsoever, whether direct or indirect, or for costs (including legal fees) and expenses arising out of the publication, use of, or reliance upon, this IEC Publication or any other IEC Publications.
 - 8) Attention is drawn to the Normative references cited in this publication. Use of the referenced publications is indispensable for the correct application of this publication.
 - 9) Attention is drawn to the possibility that some of the elements of this IEC Publication may be the subject of patent rights. IEC shall not be held responsible for identifying any or all such patent rights.
- International Standard IEC 61804-3 has been prepared by subcommittee 65E: Devices and integration in enterprise systems, of IEC technical committee 65: Industrial-process measurement, control and automation.
- This fourth edition cancels and replaces the third edition published in 2015. This edition constitutes a technical revision.
- This edition was developed by merging material from multiple variants of existing EDDL specifications including those from FieldComm Group (Foundation™ Fieldbus¹, HART®²),

¹ FOUNDATION™ Fieldbus is the trademark of FieldComm Group. This information is given for the convenience of users of this document and does not constitute an endorsement by IEC of the product named. Equivalent products may be used if they can be shown to lead to the same results.

² HART® is the registered trademark of FieldComm Group. This information is given for the convenience of users of this document and does not constitute an endorsement by IEC of the product named. Equivalent products may be used if they can be shown to lead to the same results.

PROFIBUS^{TM3} Nutzerorganisation e.V. (PNO), and ISA100_Wireless^{TM4} Compliance Institute (ISA100 WCI). Anywhere there may be a profile deviation are now indicated in the context where the related deviation is found. As a result, the formatting and numbering of this edition may be different from any of the individual specifications from which this edition was derived.

This edition includes the following significant technical changes with respect to the previous edition:

- Communication profiles ISA100 and GPE were added.
- EDD Identification Information has a new LAYOUT_TYPE attribute.
- New construct SEMANTIC_MAP was added.
- CLASS attribute values LOCAL_A and LOCAL_B were added.
- Extended LIST functionality to support device managed lists.

The text of this standard is based on the following documents:

FDIS	Report on voting
65E/XX/FDIS	65E/XX/RVD

Full information on the voting for the approval of this standard can be found in the report on voting indicated in the above table.

This publication has been drafted in accordance with the ISO/IEC Directives, Part 2.

Headings ending with '(void)' are used to retain the numbering of previous editions.

A list of all parts in IEC 61804 series, published under the general title *Function blocks (FB) for process control and Electronic Device Description Language (EDDL)*, can be found on IEC website.

Future standards in this series will carry the new general title as cited above. Titles of existing standards in this series will be updated at the time of the next edition.

The committee has decided that the contents of this publication will remain unchanged until the stability date indicated on IEC web site under "<http://webstore.iec.ch>" in the data related to the specific publication. At this date, the publication will be

- reconfirmed,
- withdrawn,
- replaced by a revised edition, or
- amended.

The National Committees are requested to note that for this publication the stability date is 2023.

³ PROFIBUS and PROFINET are the trademarks of the PROFIBUS Nutzerorganisation e.V. This information is given for the convenience of users of this document and does not constitute an endorsement by IEC of the product named. Equivalent products may be used if they can be shown to lead to the same results.

⁴ ISA100_WirelessTM is the trademark of ISA100 Wireless Compliance Institute. This information is given for the convenience of users of this document and does not constitute an endorsement by IEC of the product named. Equivalent products may be used if they can be shown to lead to the same results.

84 THIS TEXT IS INCLUDED FOR THE INFORMATION OF THE NATIONAL COMMITTEES AND WILL BE
85 DELETED AT THE PUBLICATION STAGE.

86

IMPORTANT – The 'colour inside' logo on the cover page of this publication indicates that it contains colours which are considered to be useful for the correct understanding of its contents. Users should therefore print this document using a colour printer.

87

88

INTRODUCTION

The EDDL fills the gap between the conceptual function block specification of IEC 61804-2 and a product implementation. It allows the manufacturers to use the same description method for devices based on different technologies and platforms. Figure 1 shows these aspects.

IEC 61804 has the general title "Function blocks (FB) for process control and Electronic Device Description Language (EDDL)" and consists of the following parts:

Part 2: FB concept

Part 3: EDDL syntax and semantics

Part 4: EDD interpretation

Part 5: EDDL Builtin library

Part 6: Meeting the requirements for integrating fieldbus devices in engineering tools for field devices

This part of IEC 61804 has integrated some parts of IEC/TS 61804-1:2003, which was withdrawn in January 2013.

The EDDL may also be used for the description of product properties in other domains such as industrial automation. Industrial automation may include devices such as generic digital and analog input/output modules, motion controllers, human-machine interfaces, sensors, closed-loop controllers, encoders, hydraulic valves, and programmable controllers.

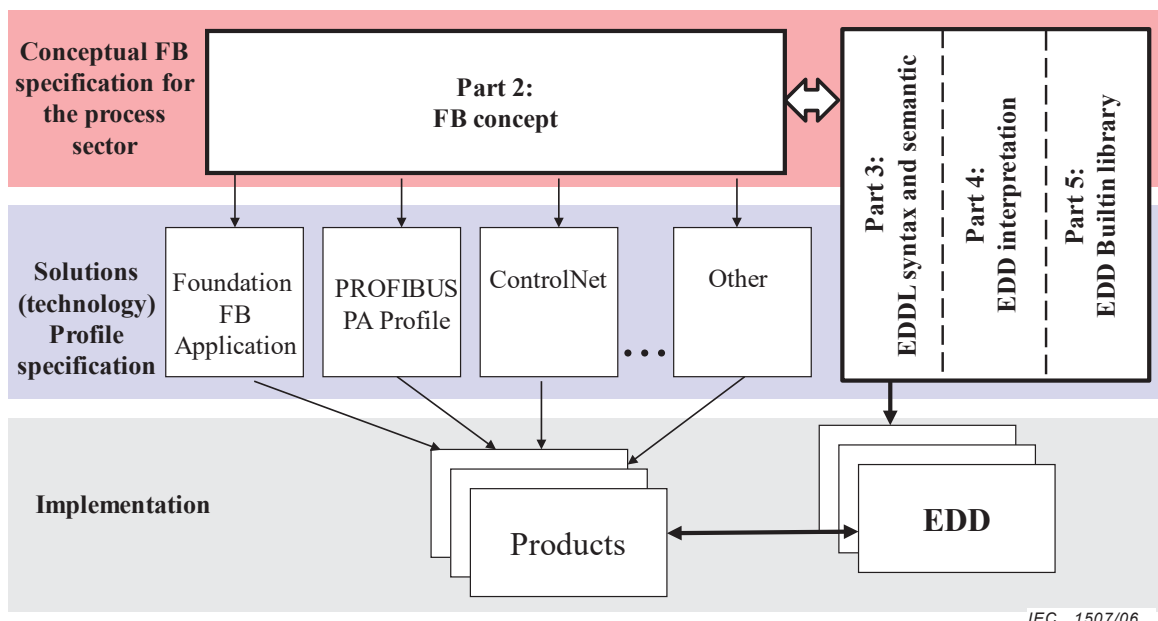


Figure 1 – Position of IEC 61804 in relation to other standards and products

FUNCTION BLOCKS (FB) FOR PROCESS CONTROL AND ELECTRONIC DEVICE DESCRIPTION LANGUAGE (EDDL) –

Part 3: EDDL syntax and semantics

1 Scope

This part of IEC 61804 specifies the Electronic Device Description Language (EDDL) technology, which enables the integration of real product details using the tools of the engineering life cycle.

This part of IEC 61804 specifies EDDL as a generic language for describing the properties of automation system components. EDDL is capable of describing

- device parameters and their dependencies;
- device functions, for example, simulation mode, calibration;
- graphical representations, for example, menus;
- interactions with control devices;
- graphical representations:
 - enhanced user interface,
 - graphing system;
- persistent data store.

EDDL is used to create Electronic Device Description (EDD) for e.g. concrete devices, common usable profiles or libraries. This EDD is used with appropriate tools to generate an interpretative code to support parameter handling, operation, and monitoring of automation system components such as remote I/Os, controllers, sensors, and programmable controllers. Tool implementation is outside the scope of this standard.

This part of IEC 61804 specifies the semantic and lexical structure in a syntax-independent manner. A specific syntax is defined in Annex A, but it is possible to use the semantic model also with different syntaxes.

Part 4 of IEC 61804 specifies EDD interpretation for EDD applications and EDDs to support EDD interoperability.

Part 5 of IEC 61804 specifies the EDDL Builtin library and provides the profiles of the various fieldbuses.

2 Normative references

The following documents are referred to in the text in such a way that some or all of their content constitutes requirements of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

IEC 61804-4, *Devices and integration in enterprise systems – Function blocks (FB) for process control and EDDL – Part 4: EDD Interpretation*

IEC 61804-5, *Devices and integration in enterprise systems – Function blocks (FB) for process control and EDDL – Part 5: EDDL Builtin library*

ISO/IEC 9899:1999, *Programming Languages - C*

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

ISO/IEC 2375, *Information technology – Procedure for registration of escape sequences and coded character sets*

- 154 ISO/IEC 7498-1, *Information technology – Open Systems Interconnection – Basic Reference*
155 *Model: The Basic Model*
- 156 ISO/IEC 8859-1, *Information technology – 8-bit single-byte coded graphic character sets –*
157 *Part 1: Latin alphabet No. 1*
- 158 ISO/IEC 9834-8, *Information technology — Open Systems Interconnection — Procedures for*
159 *the operation of OSI Registration Authorities: Generation and registration of Universally Unique*
160 *Identifiers (UUIDs) and their use as ASN.1 Object Identifier components* ISO/IEC 9899,
161 *Information technology – Programming languages – C*
- 162 ISO/IEC 10646-1, *Information technology – Universal Multiple-Octet Coded Character Set*
163 *(UCS) – Part 1: Architecture and Basic Multilingual Plane*
- 164 ISO/IEC 10918, *Information technology — Digital compression and coding of continuous-tone*
165 *still images, lossless*
- 166 ISO/IEC 15948, *Information technology — Computer graphics and image processing —*
167 *Portable Network Graphics (PNG)*
- 168 ISO 639 (all parts), *Codes for the representation of names of languages*
- 169 ISO 3166-1, *Codes for the representation of names of countries and their subdivisions – Part*
170 *1: Country codes*
- 171 ISO 8859-1, *Information technology — 8-bit single-byte coded graphic character sets — Part*
172 *1: Latin alphabet No. 1*
- 173 IEEE 754, *IEEE Standard for Floating-Point Arithmetic*
- 174 RFC 3629, *UTF-8, User Datagram Protocol*, available at <http://www.ietf.org/rfc/rfc0768.txt>
- 175 W3C Cascading Style Sheets Level 2 Specification <http://www.w3.org/TR/CSS2>
- 176

177 **3 Terms, definitions, abbreviated terms and acronyms**

178 **3.1 Terms and definitions**

179 For the purposes of this document, the terms and definitions given in IEC 60050-351, in
180 IEC 61804-2, as well as the following terms and definitions apply. ISO and IEC maintain
181 terminological databases for use in standardization at the following addresses:

- 182 • ISO Online browsing platform: available at <https://www.iso.org/obp>
- 183 • IEC Electropedia: available at <http://www.electropedia.org/>

184 **3.1.1** 185 **Builtin**

186 predefined subroutine executed by the EDD application, e.g. for communication and display
187 purposes

188 **3.1.2** 189 **device**

190 independent physical entity capable of performing one or more specified functions in a
191 particular context and delimited by its interfaces

192 [SOURCE: IEC 61499-1]

193 **3.1.3** 194 **EDD application**

195 program using the EDD, or any translated form, which offers functionality such as
196 communication representation, data representation, graphical representation, etc.



- 197 **3.1.4**
198 **EDDL processor**
199 processor or program, which translates the EDD into an executable form that can be
200 processed by an EDD application
- 201 **3.1.5**
202 **EDDL profile**
203 selection of the supported elements of the EDDL lexical structure including the syntax
204 definitions for a number of specific consortia
- 205 **3.1.6**
206 **Electronic Device Description Language**
207 **EDDL**
208 formal language for describing devices and communication components
- 209 **3.1.7**
210 **Electronic Device Description**
211 **EDD**
212 data collection containing the device parameter(s), its(their) dependencies, its(their) graphical
213 representation and a description of the data sets which are transferred
- 214 Note 1 to entry: The Electronic Device Description is created using the EDDL.
- 215 **3.1.8**
216 **Electronic Device Description compiler**
217 compiler which translates the EDD source in an internal format that is used by the EDD
218 Interpreter
- 219 **3.1.9**
220 **Electronic Device Description Interpreter**
221 **EDDI**
222 Interpreter which uses the EDD source or an internal format that is given by the EDDL
223 compiler to provide the EDD information to the EDD user
- 224 **3.1.10**
225 **function block**
226 **function block instance**
227 software functional unit comprising an individual, named copy of a data structure and
228 associated operations specified by a corresponding FB type
- 229 Note 1 to entry: Typical operations of an FB include modification of the values of the data in its associated data
230 structure.
- 231 [SOURCE: IEC 61804-2]
- 232 **3.1.11**
233 **hardware**
234 physical equipment, as opposed to programs, procedures, rules and associated
235 documentation
- 236 **3.1.12**
237 **preprocessor**
238 part of the run-time environment that transforms the information given in an EDD
- 239 **3.1.13**
240 **preprocessor directives**
241 description of conditions for filtering the EDD code before compilation or interpretation
- 242 Note 1 to entry: For example, a preprocessor directive providing the facility to define names for constants or to
243 write macros to make code easier to read.
- 244 **3.1.14**
245 **software**
246 intellectual creation comprising the programs, procedures, rules and any associated
247 documentation pertaining to the operation of a system

248 [SOURCE: IEC 61499-1]

249 3.2 Abbreviated terms and acronyms

ASCII	American Standard Code for Information Interchange according to ISO/IEC 10646-1
CP	Communication Profile
CPF	Communication Profile Family
CS	Communication Server
EDD	Electronic Device Description
EDDL	Electronic Device Description Language
EUC	Extended Unicode according to ISO/IEC 10646
FB	Function Block
FF	Fieldbus Foundation ⁵
GPE	Generic Protocol Extension
HART	Highway Addressable Remote Transducer
HMI	Human Machine Interface
I/O	Input/Output
ID	Identifier
ISA100	ISA100_Wireless ^{TM 6}
OSI	Open Systems Interconnection
PI	PROFIBUS and PROFINET ⁷ International
PB	PROFIBUS ⁷
PDU	Protocol Data Unit
PN	PROFINET ⁷
PNO	PROFIBUS Nutzerorganisation
QM	Quality Management

250 4 Conformance statement

251 A statement of conformance to a profile of this part of IEC 61804 shall be stated⁸ as

- 252 • compliance to IEC 61804-3 <Type>

253 where the Type within the angle brackets < > is optional and the angle brackets are not to be
254 included.

255 The Type may be, for example, the name of the communication protocol abbreviations used
256 such as CS, FF, GPE, HART, ISA100, PB, or PN, or in an encoded way according to the
257 Communication Profiles specified in IEC 61784-1 or in IEC 61784-2.

258 Product standards shall not include any conformity assessment aspects (including QM
259 provisions), either normative or informative, other than provisions for product testing
260 (evaluation and examination).

261 **5 EDD and EDDL model**

262 **5.1 Overview of EDD and EDDL**

263 An Electronic Device Description (EDD) contains all device parameter(s) of an automation
264 system component. The technology used to describe an EDD is called Electronic Device
265 Description Language (EDDL). The EDDL provides a set of scalable basic language elements
266 to handle simple, complex or modular devices. The EDDL is a descriptive language based on
267 an ASCII format with clear separation between data and program.

268 Data in a text field, which is marked with a country code like Japanese, may use a multi-byte
269 code.

270 **5.2 EDD architecture**

271 From the viewpoint of the ISO/OSI model (ISO/IEC 7498-1), an EDD is above Layer 7.
272 However, the EDD application uses the communication system to transfer its information. An
273 EDD contains constructs that support mapping to a supporting communication system.

274 The device manufacturer defines the objects, which are reflected by the logical representation
275 of the objects within an EDD application. For that reason, EDDL has language elements,
276 which map the EDD data to the data representation of the communication system, so that the
277 user of an EDD does not need to know the physical location (address) of a device object.

278 EDD describes the management of information to be displayed to the user. The specific
279 representation of such visualization is not part of EDD or EDDL definitions.

280 **5.3 Concepts of EDD**

281 The manufacturer of a device or of an automation system component describes the properties
282 of the device by using the EDDL. The resulting EDD contains information such as:

- 283 • description of the device parameters;
- 284 • description of parameter dependencies;
- 285 • logical grouping of the device parameters;
- 286 • selection and execution of supported device functions;
- 287 • business logic (programmed functions);
- 288 • description of the transferred data sets.

289 Depending on the required usage, the EDD may be physically located

- 290 • in a device;
- 291 • in an external data storage medium such as a compact disk, floppy or a server;
- 292 • partially distributed in the device and an external storage medium.

293 EDD supports text strings (common terms, phrases etc.) in more than one language (English,
294 German, French, etc.). Text strings may be stored in separate dictionaries. There may be
295 more than one dictionary for one EDD.

296 An EDD implementation includes sufficient information about the target device to match a
297 specific EDD to a specific device. Manufacturer, device type, and revision are examples of
298 information that can be used to match a device to a specific EDD.

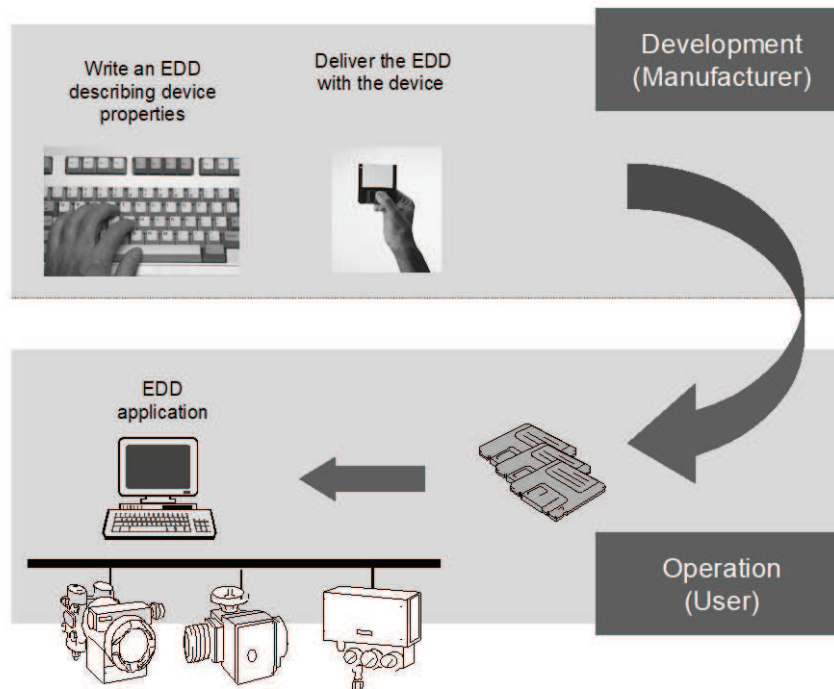
299 **5.4 Principles of the EDD development process**

300 **5.4.1 General**

301 The creation of an EDD is a three-stage process: EDD source generation, EDD preprocessing
302 and EDD compilation.

5.4.2 EDD source generation

The EDD generation process is shown in Figure 2.



IEC 1511/06

Figure 2 – EDD generation process

The device manufacturer writes an appropriate EDD for his device and delivers both (EDD and device). If the automation system supports the EDD method, a system integration can be made by the user.

The EDDs for devices may be embedded in the device memory or delivered using separate storage media or downloaded from an appropriate network server. The EDD is 'interpreted' or ' browsed' by an EDD-application. EDDs are normally stored as source files or preprocessed files.

5.4.3 EDD preprocessing

In the preprocessing stage an EDD preprocessor generates a consistent EDD representation suitable for final compilation.

Preprocessing supports, for example, substitution of definitions and inclusion of external text. The output of the preprocessor is a complete EDD without any preprocessor directive.

5.4.4 EDD compilation

The EDD compilation stage produces an EDD application internal representation from a preprocessed EDD to be used in the EDD application.

5.5 Interrelations between the lexical structure and formal definitions

The lexical structure of EDDL and its elements are described in Clauses 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20. Formal definitions and syntax for each EDDL element are given in Annex A. The lexical structure and its formal description use the same name.

NOTE Instead of the specified formal definitions and syntax in Annex A, another specification may be developed as an additional option.

5.6 Builtins

Builtins are predefined subroutines which are executed in the EDD application.

The library of Builtins is defined in IEC 61804-5.

5.7 Profiles

EDDL is a unified specification of existing legacy EDD concepts. Concrete EDD applications use a subset of the EDDL specification. The elements of the specification applicable to various profiles are described in tables throughout the document, along with the description of the element.

In addition to EDD profiles, implementing consortia also publish “Device Profiles”, which are used to support the interchangeability of compliant devices. These Device Profiles may be described using the EDDL specified in this document.

6 Conventions

6.1 General

In IEC 61804 series all keywords are written in uppercase letters. An EDDL element is written in lowercase letters to address the semantic of the whole element.

The EDDL is generally described using lexical structures in which the elements and the presence of fields are specified.

6.2 Conventions for basic construct description

The following conventions apply for the basic construct definition.

- Table 1 defines the format for the Builtin overview table.
- Table 2 defines the contents of the columns of Table 1.
- Table 3 defines the usage of attributes of basic constructs or their subattributes.
- EDDL basic constructs and attributes referenced in this document are written in upper case letters.

Table 1 – Format for the basic construct table

Reference	Basic construct	Attribute	Communication Profiles

Table 2 – Contents of the Basic Construct table

Column	Text	Meaning
Reference		Reference to the respective subclause that specifies the attribute of the basic construct.
Basic construct		Name of the basic construct.
Communication Profiles	✓	The basic construct is supported by the respective Communication Profile.
	x	The basic construct is not supported by the respective Communication Profile.

Table 3 – Attribute usage descriptions

Usage	Attribute	Description
m	yyy	The presence of this attribute is mandatory.
o	xxx	The presence of this attribute is optional.
s	z1	The presence of this attribute is selectable with other attributes, which are also marked with “s” in the “usage” column. Only one of the selectable attributes (z1 or z2) is present.
s	z2	The presence of this attribute is selectable with other attributes, which are also marked with “s” in the “usage” column. Only one of the selectable attributes (z1 or z2) is present.
c	uuu	The presence of this attribute is constrained and it is only present if the constraint is met.
L	Vvv	The presence of this attribute is limited to be used only in specified cases.

The characters in the “usage” column have the following meanings.

m: this attribute is mandatory and shall be present.

o: this attribute is optional and need not be present.

s: this attribute is a selection. One, and only one, of the fields marked with s (z1 or z2) shall be present.

c: this attribute is constrained. The constraint is described in the “description” column.

In the “attribute” column, if more than one attribute exists, which have the same usage, the attributes are sorted in alphabetical order.

6.3 Conventions for lexical structure

The lexical structures in this document are specified using the rules defined in Table 4.

Table 4 – Rules for lexical convention

Lexical structure element	Description
bold	Concrete syntax literals are in bold .
<i>italics</i>	Syntax elements are in <i>italics</i> .
:=	Definition of a syntax element.
<cond>	The preceding syntax element may be conditionally specified.
[]	Grouping of syntax elements
*	The preceding syntax element may appear zero or more times. The syntax element can be a group of syntax elements enclosed in square brackets.
+	The preceding syntax element may appear one or more times. The syntax element can be a group of syntax elements enclosed in square brackets.
?	The preceding syntax element may appear zero or one time. The syntax element can be a group of syntax elements enclosed in square brackets.
	Alternative syntax elements
expression	An expression
<data type>-expression	An expression that can be resolved to a specified data type.
<data type>-constant	A constant value of the specified data type.
<data type>-value	A value of the specified data type.

Table 5 Describes the applicable EDD sources for lexical structure element definitions.

372

Table 5 – EDD sources

Lexical structure element	Applicable source						
	Constant (Literal)	Referenced attribute	Enumerated()	Dictionary reference (only string)	Concatenation ('+', only string)	Expression incl. type converting	Condition (IF, SELECT)
<data type>-constant	x	x	x	x			
<data type>-value	x	x	x	x	x		
<data type>-value<cond>	x	x	x	x	x		x
<data type>-expression	x	x	x	x	x	x	
<data type>-expression<cond>	x	x	x	x	x	x	x

373

374 7 Overview

375 7.1 EDDL features

376 The EDDL is a structured and interpretative language for describing device properties. Also
 377 the interactions between the devices and the EDD run-time environment are incorporated in
 378 the EDDL. The EDDL provides a set of language elements for these purposes.

379 For a specific EDD implementation it is not necessary to use all of the elements provided by
 380 the language.

381 Compatible subsets of EDDL are permitted and may be specified using profiles (for example,
 382 choice of constructs, number of recursions, and selection of options). EDD developers are
 383 required to identify within each EDD details as to which profile has been used.

384 7.2 Syntax representation

385 The lexical structure is converted into specific syntaxes specified in Annex A by the element
 386 name (for example, VARIABLE in the lexical structure equates to the keyword "VARIABLE" in
 387 Annex A).

388 Beside the defined syntax in this part of IEC 61804, it is possible to use other syntax
 389 definitions, which may be added in the future to allow other features and representations for
 390 future progress.

391 7.3 EDD language elements

392 The language is structured with the following language elements:

- 393 • identification element;
- 394 • basic construction elements;
- 395 • special elements.

396 Frequently used text strings, for example, help text and multi-lingual lists of labels should be
 397 separated in a text dictionary (see 20).

398 Identification information is the first entry in every EDD file. The identification information
 399 uniquely identifies the version of EDDL used, together with the specific device type, model
 400 codes and revision details covered by the EDD file.

7.4 Basic construction elements

7.4.1 General

These basic constructs have been specified to support descriptions of devices used within industrial control applications, together with their properties and functionality.

Some constructs have similar names and functions while differing in their detailed specification. This additional variety has been included to ensure compatibility with several existing description languages. Appropriate mapping cross-references are given by profiles.

Each of the basic constructs has a set of attributes associated with it. Attributes can also have subattributes, which refine the definition of the attribute and hence the definition of the construct itself.

The definition of an attribute may be static or dynamic. A static attribute definition never changes, while a dynamic attribute definition may change in order to accommodate parameter value changes.

7.4.2 AXIS

AXIS describes the axis of a CHART or GRAPH (see 11.7).

7.4.3 BLOB

BLOB describes a Binary Large Object used for transferring binary data to or from a device (see 9.8).

7.4.4 BLOCK_A

BLOCK_A is a logical grouping of CHARACTERISTICS, PARAMETERS, PARAMETER_LISTS, and ITEM_LISTS, see Figure 3. To access one item of BLOCK_A, the instance of the block should be used (see 13.1 and Figure 3).

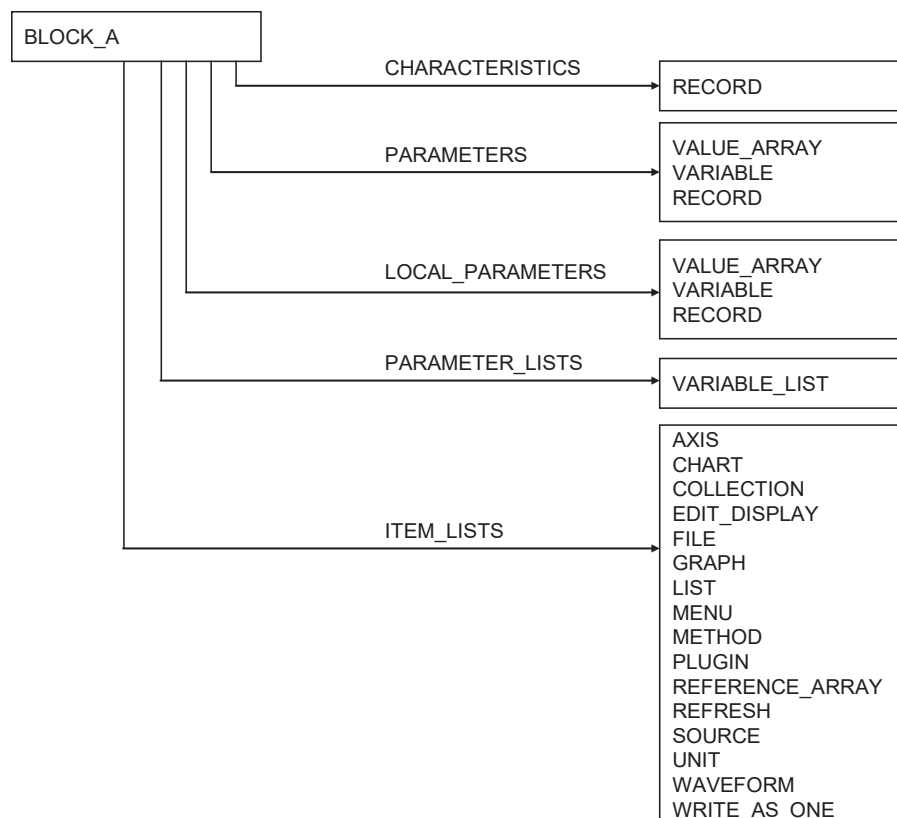


Figure 3 – BLOCK_A

IEC 1512/06

7.4.5 BLOCK_B

BLOCK_B is used to structure variables in instances of logical block types. For accessing a VARIABLE within a BLOCK_B construct, the basic construct COMMAND is used to provide the relative addressing (see 13.2).

7.4.6 CHART

CHART describes a chart used to display data from a device in the EDD application (see 11.5 and Figure 4).



Figure 4 – CHART

7.4.7 COLLECTION

COLLECTION describes logical groupings of data (see 9.5 and Figure 5).

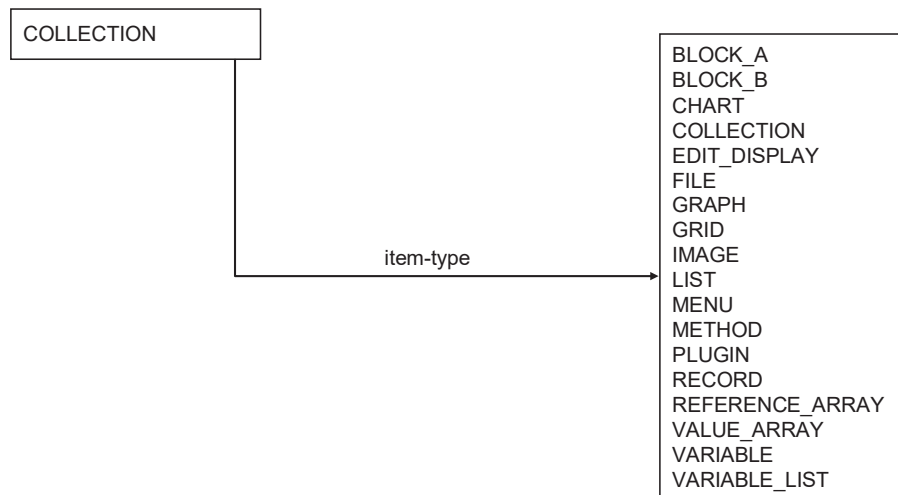
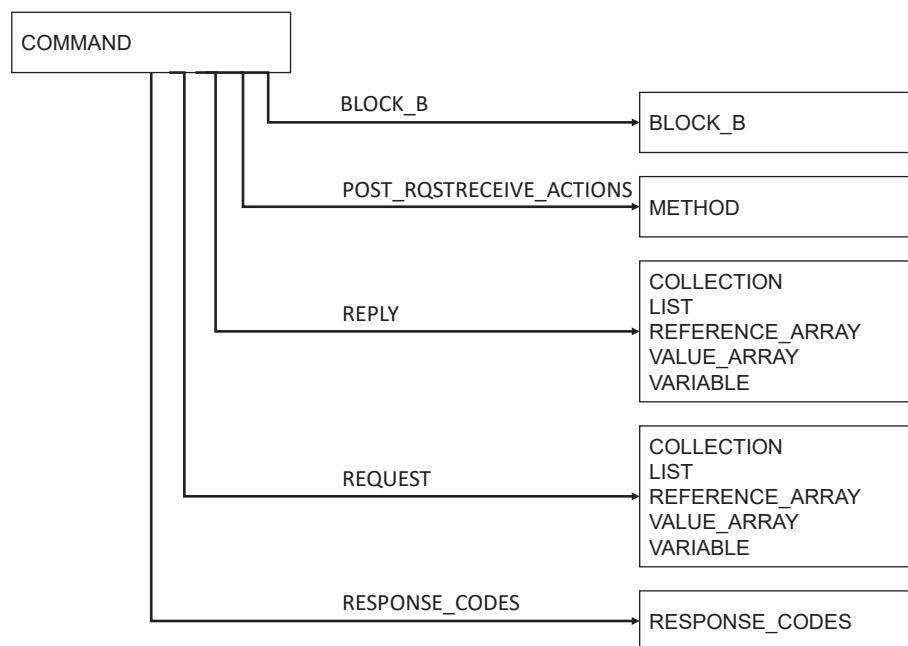


Figure 5 – COLLECTION

IEC 1514/06

7.4.8 COMMAND

COMMAND describes the structure and the addressing of the variables in the device (see 10.1 and Figure 6).



441
442

443

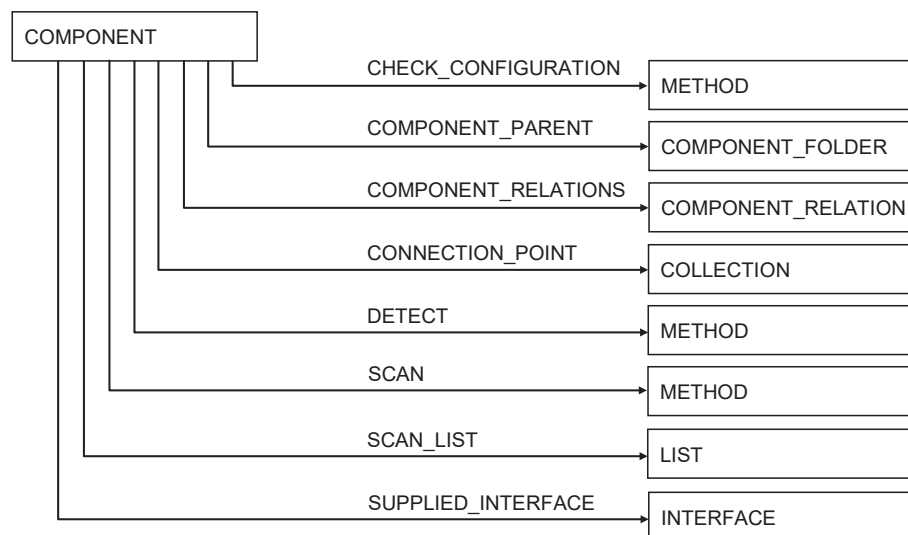
444

Figure 6 – COMMAND

IEC 1515/06

7.4.9 COMPONENT

COMPONENT describes the configuration behavior of the device that is described in the EDD (see 13.6 and Figure 7).



448

449

Figure 7 – COMPONENT

7.4.10 COMPONENT_FOLDER

COMPONENT_FOLDER describes a folder in a catalog. Folders allow groupings of components in a catalog (see 13.7 and Figure 8).

452



Figure 8 – COMPONENT_FOLDER

7.4.11 COMPONENT_REFERENCE

COMPONENT_REFERENCE refers to a COMPONENT or a COMPONENT_FOLDER (see 13.8 and Figure 9).



Figure 9 – COMPONENT_REFERENCE

7.4.12 COMPONENT_RELATION

COMPONENT_RELATION specifies a relation between COMPONENTs, e.g. a module and one or more other modules (see 13.9 and Figure 10).

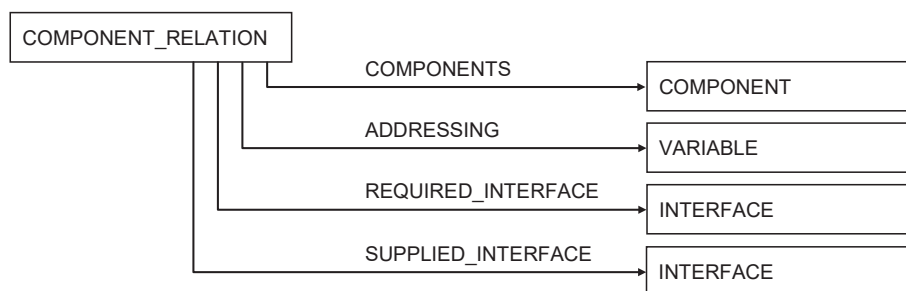


Figure 10 – COMPONENT_RELATION

7.4.13 EDIT_DISPLAY

EDIT_DISPLAY describes how the data will be represented to a user by a display device (see 11.2 and Figure 11).

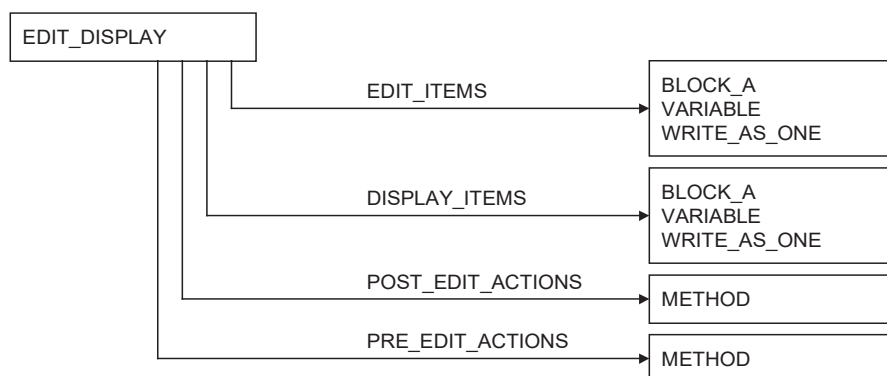


Figure 11 – EDIT_DISPLAY

IEC 1517/06

7.4.14 FILE

FILE describes an area of persistent storage (see 9.6 and Figure 12).

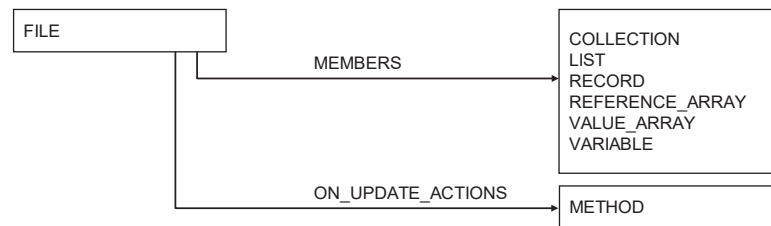


Figure 12 – FILE

IEC 1518/06

7.4.15 GRAPH

GRAPH describes a graph used to display data from a device in the EDD application (see 11.7.4 and Figure 13).

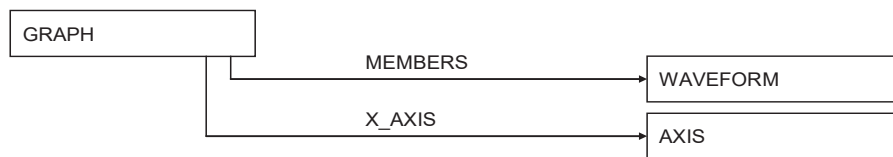


Figure 13 – GRAPH

IEC 1519/06

7.4.16 GRID

GRID describes a matrix used to display data from a device in the EDD application (see 11.4 and Figure 14).



Figure 14 – GRID

IEC 1520/06

7.4.17 IMAGE

IMAGE describes a visual image (see 11.3 and Figure 15).



Figure 15 – IMAGE

IEC 1521/06

7.4.18 IMPORT

IMPORT is used to import an EDD and to modify its existing definitions (see 13.4).

7.4.19 INTERFACE

INTERFACE is used to specify a set of basic constructs of components that are accessible by other components (see 13.10).

7.4.20 LIKE

LIKE creates a new instance of an existing instance of a basic construct. The attributes of the new instance may be redefined, added or deleted (see 13.3 and Figure 16).

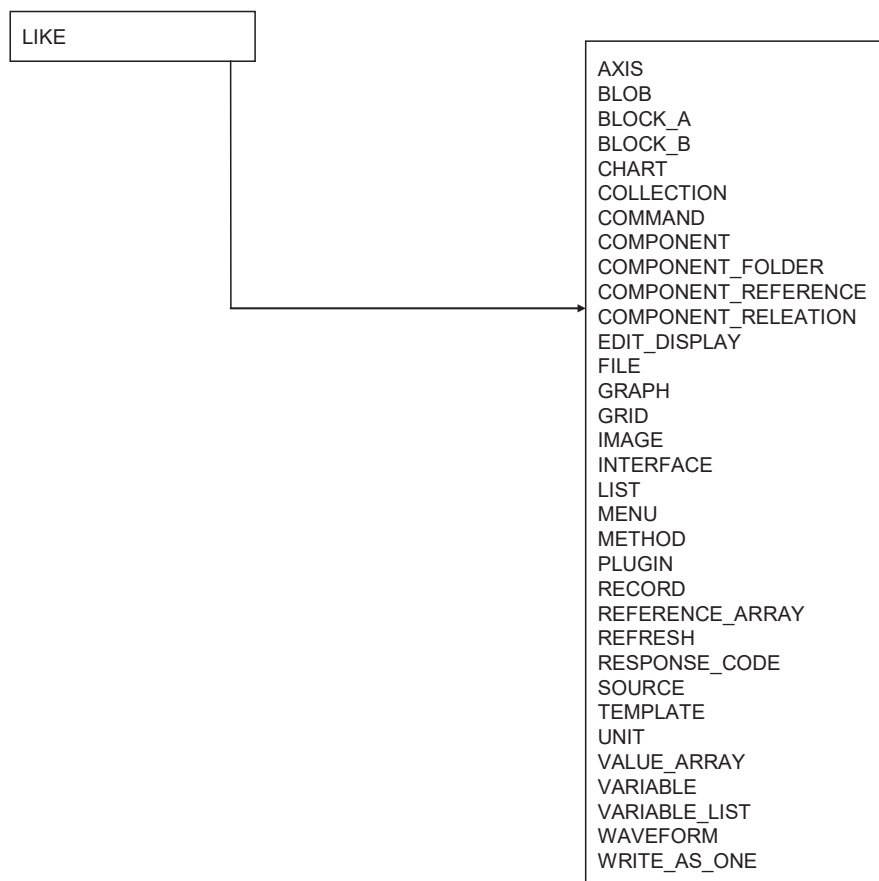


Figure 16 – LIKE

IEC 1522/06

7.4.21 LIST

LIST describes a sequence of elements (see 9.4 and Figure 17).



Figure 17 – LIST

IEC 1523/06

7.4.22 MENU

MENU describes how the data will be presented to a user by an EDD application (see 11.1 and Figure 18).

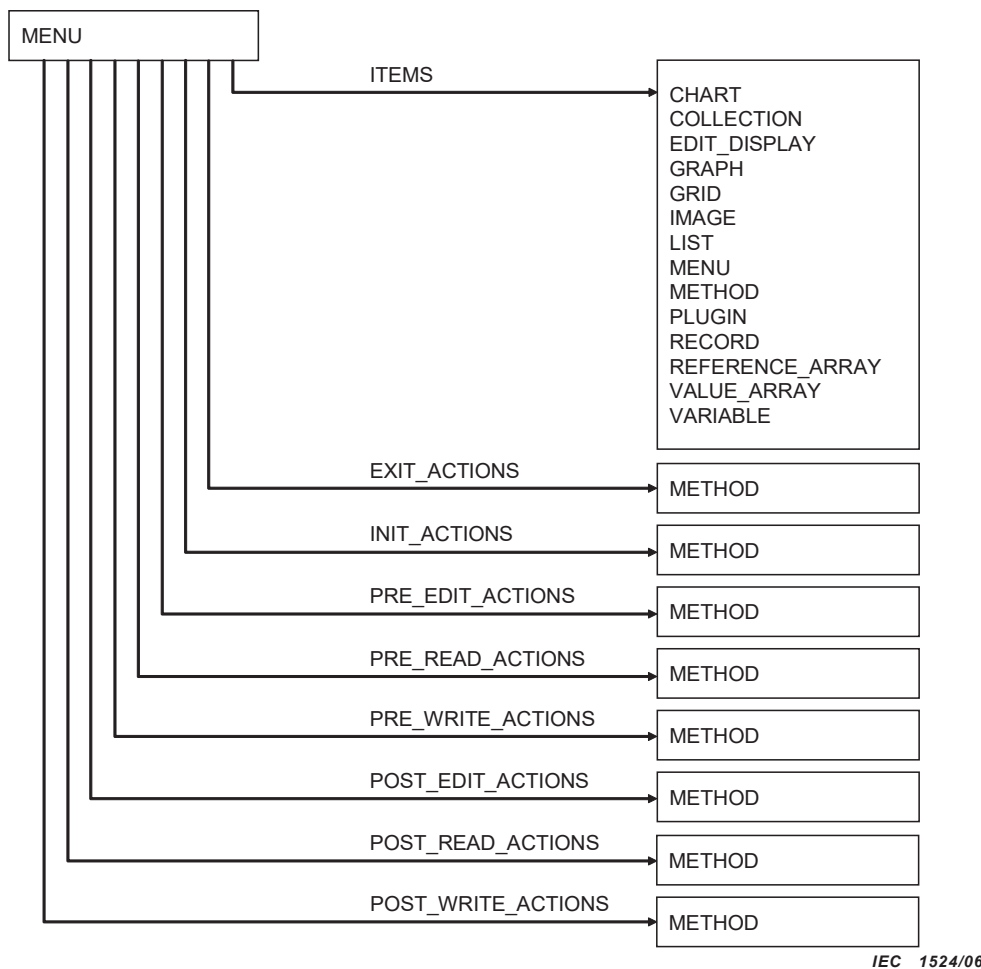


Figure 18 – MENU

7.4.23 METHOD

METHOD describes the execution of a complex sequence of event interactions that take place between system devices such as display units, configurators and devices (see 12.1).

7.4.24 PLUGIN

PLUGIN describes a User Interface Plug-in (UIP) as specified in 11.10.

7.4.25 RECORD

RECORD describes the data contained in the device (see 10.2 and Figure 19).

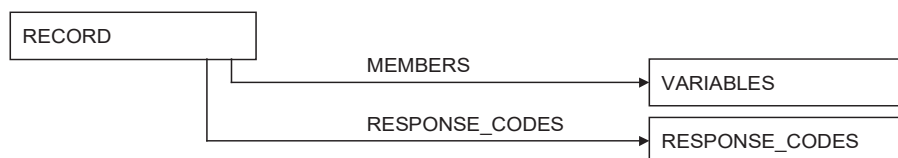
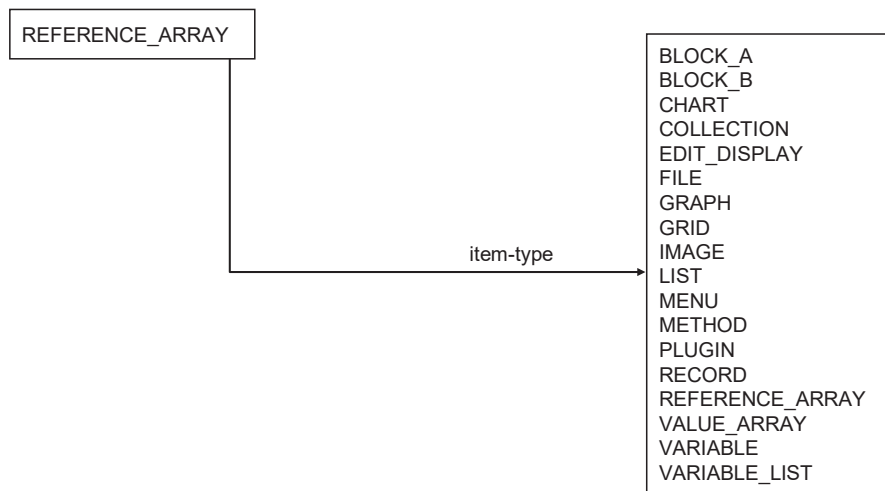


Figure 19 – RECORD

7.4.26 REFERENCE_ARRAY

REFERENCE_ARRAY is a set of EDD items of the same EDDL item type (for example, VARIABLE or MENU). An item can be referenced in the EDD via the REFERENCE_ARRAY name and the index associated with the item (see 9.2 and Figure 20).



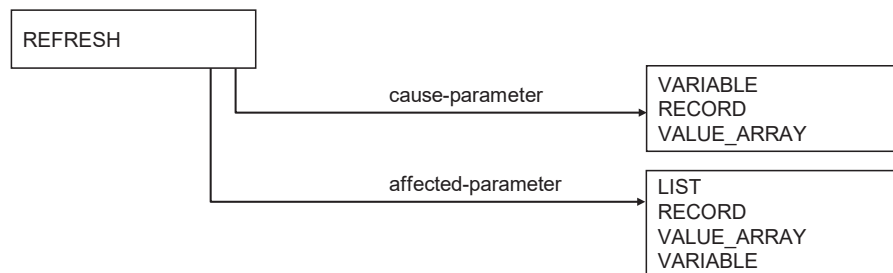
IEC 1527/06

Figure 20 – REFERENCE_ARRAY

7.4.27 Relations

7.4.27.1 REFRESH

REFRESH describes relationships between LISTS, RECORDs, VALUE_ARRAYs and VARIABLES (see 9.7.1 and Figure 21).



IEC 1528/06

Figure 21 – REFRESH

7.4.27.2 SEMANTIC_MAP

SEMANTIC_MAP describes relationships between standardized semantic identifiers and data items (see 9.7.2 and Figure 22). Semantic identifiers are locale independent strings typically specified in international standards like the IEC Common Data Dictionary (CDD) based on IEC 61360 and IEC 61987.

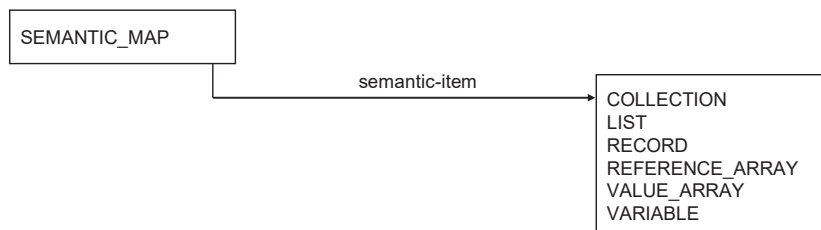


Figure 22 – SEMANTIC_MAP

7.4.27.3 UNIT

UNIT describes relationships between AXISs, VALUE_ARRAYs and VARIABLES (see 9.7.2 and Figure 23).

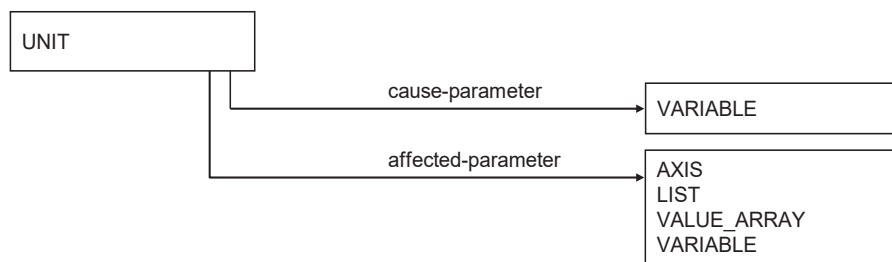


Figure 23 – UNIT

IEC 1529/06

7.4.27.4 WRITE_AS_ONE

WRITE_AS_ONE describes relationships between VARIABLES, RECORDs, and VALUE_ARRAYs (see 9.7.4 and Figure 24).



Figure 24 – WRITE_AS_ONE

IEC 1530/06

7.4.28 RESPONSE_CODES

RESPONSE_CODES specifies the list of response code elements that are used by the EDD application to inform the user about the communication result, e.g. error, warning (see 10.3).

7.4.29 SOURCE

SOURCE describes a source of data values for a CHART (see 11.6 and Figure 25).

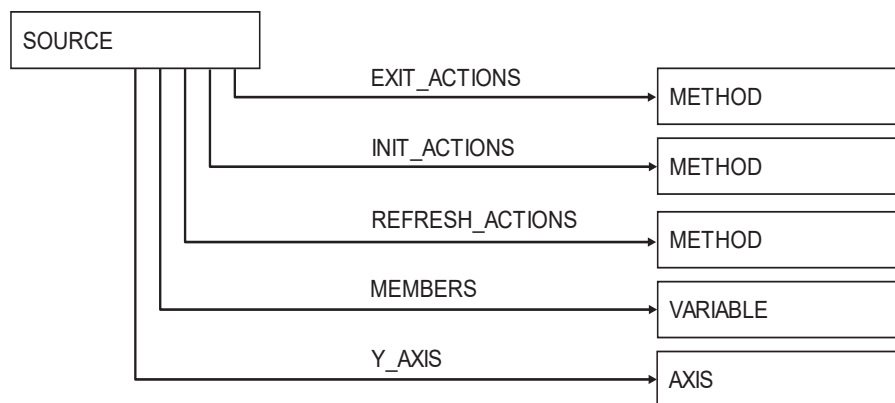


Figure 25 – SOURCE

IEC 1531/06

7.4.30 TEMPLATE

TEMPLATE describes a set of parameter values that are used for offline configuration by the EDD application (see 13.5).

7.4.31 VALUE_ARRAY

VALUE_ARRAY describes logical groups of EDDL variables (see 9.3 and Figure 26).

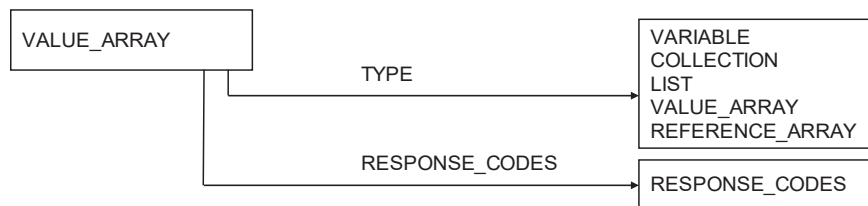


Figure 26 – VALUE_ARRAY

IEC 1532/06

7.4.32 VARIABLE

VARIABLE describes the data contained in the device (see 9.1 and Figure 27).

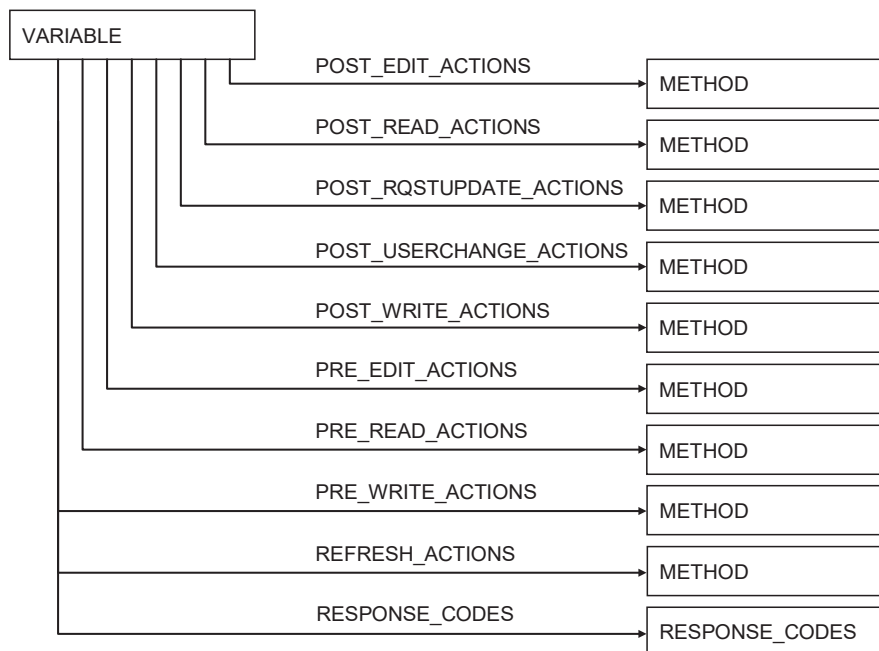


Figure 27 – VARIABLE

IEC 1533/06

7.4.33 VARIABLE_LIST

VARIABLE_LIST describes logical groupings of data contained in the device that may be communicated as a list (see 10.4 and Figure 28).

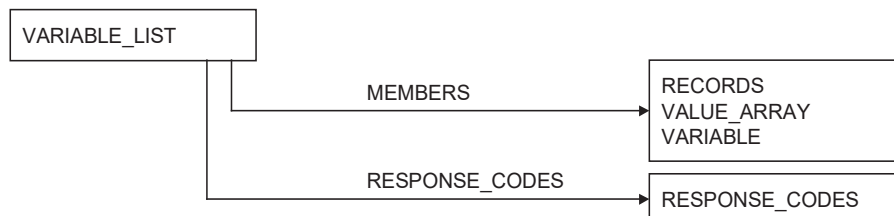


Figure 28 – VARIABLE_LIST

IEC 1534/06

7.4.34 WAVEFORM

WAVEFORM describes a data set that may be displayed by a GRAPH (see 11.9 and Figure 29).