



BSI Standards Publication

Communication networks and systems for power utility automation

Part 8-2: Specific Communication Service Mapping (SCSM) -
Mapping to Extensible Messaging Presence Protocol (XMPP)

National foreword

This British Standard is the UK implementation of EN IEC 61850-8-2:2019. It is identical to IEC 61850-8-2:2018.

The UK participation in its preparation was entrusted to Technical Committee PEL/57, Power systems management and associated information exchange.

A list of organizations represented on this committee can be obtained on request to its secretary.

This publication does not purport to include all the necessary provisions of a contract. Users are responsible for its correct application.

© The British Standards Institution 2019
Published by BSI Standards Limited 2019

ISBN 978 0 580 90794 4

ICS 33.200

Compliance with a British Standard cannot confer immunity from legal obligations.

This British Standard was published under the authority of the Standards Policy and Strategy Committee on 31 March 2019.

Amendments/corrigenda issued since publication

Date	Text affected
------	---------------

English Version

Communication networks and systems for power utility
automation - Part 8-2: Specific Communication Service Mapping
(SCSM) - Mapping to Extensible Messaging Presence Protocol
(XMPP)
(IEC 61850-8-2:2018)

Réseaux et systèmes de communication pour
l'automatisation des systèmes électriques - Partie 8-2:
Mapping des services de communication spécifiques
(SCSM) - Mapping avec le protocole XMPP (Extensible
Messaging Presence Protocol)
(IEC 61850-8-2:2018)

Kommunikationsnetze und -systeme für die
Automatisierung der elektrischen Energieversorgung - Teil
8-2: Spezifische Abbildung von Kommunikationsdiensten
(SCSM) - Abbildungen auf erweiterbares Messaging
Presence Protocol (XMPP)
(IEC 61850-8-2:2018)

This European Standard was approved by CENELEC on 2019-01-18. 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, Serbia, 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: Rue de la Science 23, B-1040 Brussels

European foreword

The text of document 57/2020/FDIS, future edition 1 of IEC 61850-8-2, prepared by IEC/TC 57 "Power systems management and associated information exchange" was submitted to the IEC-CENELEC parallel vote and approved by CENELEC as EN IEC 61850-8-2:2019.

The following dates are fixed:

- latest date by which the document has to be implemented at national level by publication of an identical national standard or by endorsement (dop) 2019-10-18
- latest date by which the national standards conflicting with the document have to be withdrawn (dow) 2022-01-18

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

This document has been prepared under a mandate given to CENELEC by the European Commission and the European Free Trade Association.

Endorsement notice

The text of the International Standard IEC 61850-8-2:2018 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 61850-7-1	NOTE	Harmonized as EN 61850-7-1
IEC 61850-7-410	NOTE	Harmonized as EN 61850-7-410
IEC 61850-7-420	NOTE	Harmonized as EN 61850-7-420

Annex ZA

(normative)

Normative references to international publications with their corresponding European publications

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.

NOTE 1 Where an International Publication has been modified by common modifications, indicated by (mod), the relevant EN/HD applies.

NOTE 2 Up-to-date information on the latest versions of the European Standards listed in this annex is available here: www.cenelec.eu.

<u>Publication</u>	<u>Year</u>	<u>Title</u>	<u>EN/HD</u>	<u>Year</u>
IEC 61850	series	Communication networks and systems for power utility automation	EN 61850	series
IEC/TS 61850-2	-	Communication networks and systems in substations - Part 2: Glossary	-	-
IEC 61850-5	-	Communication networks and systems for power utility automation - Part 5: Communication requirements for functions and device models	EN 61850-5	-
IEC 61850-6	-	Communication networks and systems for power utility automation - Part 6: Configuration description language for communication in electrical substations related to IEDs	EN 61850-6	-
IEC 61850-7-1	-	Communication networks and systems for power utility automation - Part 7-1: Basic communication structure - Principles and models	EN 61850-7-1	-
IEC 61850-7-2	2010	Communication networks and systems for power utility automation - Part 7-2: Basic information and communication structure - Abstract communication service interface (ACSI)	EN 61850-7-2	2010
+ A1	2018		+A1	2018
IEC 61850-7-3	-	Communication networks and systems for power utility automation - Part 7-3: Basic communication structure - Common data classes	EN 61850-7-3	-
IEC 61850-7-4	-	Communication networks and systems for power utility automation - Part 7-4: Basic communication structure - Compatible logical node classes and data object classes	EN 61850-7-4	-
IEC 61850-8-1	2011	Communication networks and systems for power utility automation - Part 8-1: Specific communication service mapping (SCSM) - Mappings to MMS (ISO 9506-1 and ISO 9506-2) and to ISO/IEC 8802-3	EN 61850-8-1	2011

+ A1	2018		-	-
IEC 62351	series	Power systems management and associated information exchange - Data and communications security	EN 62351	series
IEC 62351-4	2018	Power systems management and associated information exchange - Data and communications security - Part 4: Profiles including MMS and derivatives	EN IEC 62351-4	2018
IEC 62351-6	-	Power systems management and associated information exchange - Data and communications security - Part 6: Security for IEC 61850	-	-
IEC/TR 62357-200	-	Power systems management and associated information exchange - Part 200: Guidelines for migration from Internet Protocol version 4 (IPv4) to Internet Protocol version 6 (IPv6)	-	-
ISO/IEC 7498-1	1994	Information technology - Open Systems Interconnection - Basic reference model: The basic model	-	-
ISO/IEC 8824-1	2015	Information technology - Abstract Syntax Notation One (ASN.1): Specification of basic notation	-	-
ISO/IEC 8825-4	2015	Information technology - ASN.1 encoding rules: XML Encoding Rules (XER)	-	-
ISO 9506	series	Industrial automation systems - Manufacturing Message Specification	-	series
ISO 9506-1	2003	Industrial automation systems - Manufacturing Message Specification - Part 1: Service definition	-	-
ISO 9506-2	2003	Industrial automation systems - Manufacturing Message Specification - Part 2: Protocol specification	-	-
IEEE C37.111	1999	IEEE Standard for Common Format for Transient Data Exchange (COMTRADE) for Power Systems	-	-
IEEE 754	1985	IEEE Standard for Binary Floating-Point Arithmetic	-	-
RFC 768	-	User datagram protocol, IETF, available at http://www.ietf.org	-	-
RFC 791	-	Internet Protocol - DARPA Internet Program Protocol Specification	-	-
RFC 792	-	Internet control message protocol - DARPA internet program - Protocol specification, IETF, available at http://www.ietf.org	-	-
RFC 793	-	Transmission Control Protocol - DARPA Internet Program Protocol Specification	-	-
RFC 826	-	An Ethernet address resolution protocol or converting network protocol addresses to 48.bit Ethernet address for transmission on Ethernet hardware, IETF, available at http://www.ietf.org	-	-
RFC 919	-	Broadcasting internet datagrams, IETF, available at http://www.ietf.org	-	-
RFC 922	-	Broadcasting internet datagrams in the presence of subnets, IETF, available at http://www.ietf.org	-	-

RFC 950	-	Internet standard subnetting procedure, IETF, available at http://www.ietf.org	-	-
RFC 1112	-	Host extensions for IP multicasting, IETF, available at http://www.ietf.org	-	-
RFC 2460	-	Internet Protocol, Version 6 (IPv6) Specification	-	-
RFC 3629	-	UTF-8, User Datagram Protocol	-	-
RFC 4422	-	Simple authentication and security layer (SASL), available at http://www.ietf.org	-	-
RFC 5246	2008	The Transport Layer Security (TLS) Protocol Version 1.2	-	-
RFC 5905	-	Network Time Protocol Version 4: Protocol and Algorithms Specification	-	-
RFC 6120	-	Extensible messaging and presence protocol (XMPP): Core, available at http://www.ietf.org	-	-
RFC 6121	-	Extensible messaging and presence protocol (XMPP): Instant messaging and presence, available at http://www.ietf.org	-	-
RFC 6122	-	Extensible messaging and presence protocol (XMPP): Address format, available at http://www.ietf.org	-	-
XEP-0198	-	Stream management	-	-

NOTE This specification defines an XMPP protocol extension for active management of an XML stream between two XMPP entities, including features for stanza acknowledgements and stream resumption.

XEP-0199 - XMPP Ping

NOTE This specification defines an XMPP protocol extension for sending application-level pings over XML streams. Such pings can be sent from a client to a server, from one server to another, or end-to-end.

This page deliberately left blank

CONTENTS

FOREWORD.....	13
INTRODUCTION.....	15
1 Scope.....	16
1.1 General.....	16
1.2 Namespace name and version	16
1.3 Code Component distribution	17
2 Normative references	17
3 Terms and definitions	19
4 Abbreviated terms	21
5 Overview	22
5.1 General.....	22
5.2 Mapping of client/server services	23
5.2.1 General	23
5.2.2 XML payloads.....	24
5.2.3 Implementation agreements.....	25
5.2.4 XMPP	25
5.3 Time sync services	26
6 Usage of XMPP	26
6.1 Principles.....	26
6.2 Connection establishment.....	26
6.2.1 General	26
6.2.2 Usage of TLS and SASL	26
6.2.3 Stream Compression	27
6.3 Mapping of ACSI services	27
6.4 Usage of XMPP presence	28
6.5 Usage of the Roster	28
6.6 XMPP extensions.....	28
6.6.1 Usage of XMPP PING – XEP 0199	28
6.6.2 Usage of Stream Management – XEP 0198	28
6.7 Implementation agreements – XMPP PING – XEP 0199.....	28
7 End-to-end security	29
8 Payload description	29
8.1 XSD overview	29
8.2 Objects of IEC 61850.....	29
8.2.1 General	29
8.2.2 Logical Node (LN).....	30
8.2.3 Mapping of references of Logical Nodes to VariableAccessSpecifications	33
8.2.4 Mapping of DataObjects reference to VariableAccessSpecifications	34
8.2.5 Mapping of DataAttributes (DataAttr) reference to VariableAccessSpecifications	35
8.2.6 Usage of alternate access for DataObjects and DataAttributes references	35
8.3 Mapping of IEC 61850-7-2 data attributes	40
8.3.1 BasicTypes.....	40
8.3.2 Additional definitions of BasicType	41
8.3.3 Common ACSITypes	43

8.3.4	Mapping of quality common data attribute type specified in IEC 61850-7-2	54
8.4	General mapping of data values within XML payloads	55
8.5	Extended behaviour for optimization of bandwidth	56
9	Server class model	56
9.1	General	56
9.2	GetServerDirectory	57
9.2.1	General	57
10	Association model	60
10.1	Association relation to communication profiles	60
10.2	Two party association model for client/server communication profile	61
10.2.1	Establishment of a secured end-to-end association	61
10.2.2	Association services	61
11	Logical device model	66
11.1	General	66
11.2	Response-	68
11.3	Extended behaviour	69
12	Logical Node model	70
12.1	General	70
12.2	GetLogicalNodeDirectory	71
12.2.1	General	71
12.2.2	Response-	73
12.2.3	Extended behaviour	73
12.3	GetAllDataValues	76
12.3.1	General	76
13	DataObject, DataAttribute, SubDataAttribute model	79
13.1	General	79
13.2	GetDataValues	79
13.3	SetDataValues	81
13.4	GetDataDirectory	82
13.4.1	General	82
13.4.2	Response-	84
13.4.3	Extended behaviour	85
13.5	GetDataDefinition	87
14	Data set class model	87
14.1	General	87
14.2	GetDataSetValues	88
14.2.1	General	88
14.2.2	Response-	90
14.3	SetDataSetValues	91
14.3.1	General	91
14.3.2	Response-	92
14.4	CreateDataSet	93
14.4.1	General	93
14.4.2	Response-	95
14.5	DeleteDataSet	96
14.5.1	General	96
14.5.2	Errors	97
14.6	GetDataSetDirectory	98

14.6.1	General	98
14.6.2	Response-	100
15	ServiceTracking model	101
16	Setting group control class model	104
16.1	Setting group control block definition	104
16.2	Setting group control class services	104
16.2.1	SelectActiveSG	104
16.2.2	SelectEditSG	106
16.2.3	SetEditSGValue	106
16.2.4	ConfirmEditSGValues	107
16.2.5	GetEditSGValue	108
16.2.6	GetSGCBValues	109
17	Reporting and logging class model	110
17.1	Report model – Report control blocks	110
17.1.1	Buffered report control block	110
17.1.2	Unbuffered report control block	112
17.2	Reporting services	113
17.2.1	Report service	113
17.2.2	GetBRCBValues	116
17.2.3	SetBRCBValues	119
17.2.4	GetURCBValues	122
17.2.5	SetURCBValues	122
17.3	Log model	122
17.3.1	Overview	122
17.3.2	Description of LCB attributes	124
17.3.3	Mapping of log and log control services	124
17.3.4	Conformance	130
18	Mapping of the generic substation event model (GSE) – Generic object oriented substation event (GOOSE)	130
18.1	GOOSE control definition	130
18.2	Specialization for Layer 2 GoCB	131
18.3	Specialization for Routable GOOSE	131
18.4	GOOSE services	131
18.4.1	General	131
18.4.2	GetGoCBValues	131
18.4.3	SetGoCBValues	131
19	Transmission of sampled values class model	131
19.1	Sampled value control block	131
19.1.1	General	131
19.1.2	Specialization for Layer 2 Sampled value	131
19.1.3	Specialization for Routable Sampled value	132
19.1.4	Specialization for Unicast Sampled value	132
19.2	Sampled value services	132
19.2.1	General	132
19.2.2	GetMSVCBValues	132
19.2.3	SetMSVCBValues	132
20	Control class model	132
20.1	General	132

20.2	Overview of control services mapping	134
20.3	Select	134
20.4	SelectWithValue	136
20.4.1	SelectWithValue service parameter mapping	136
20.4.2	General mapping of the SelectWithValue service	136
20.4.3	SelectWithValue response–	139
20.5	Cancel	139
20.5.1	Cancel service parameter mapping	139
20.5.2	General mapping of the Cancel service	140
20.5.3	Cancel response–	141
20.6	Operate	142
20.6.1	Operate service parameter mapping	142
20.6.2	General mapping of the Operate service	142
20.6.3	Operate response–	144
20.6.4	CommandTermination service parameter mapping	144
20.6.5	General mapping of the CommandTermination service	144
20.7	TimeActivatedOperate	148
20.7.1	TimeActivatedOperate service parameter mapping	148
20.7.2	Mapping of the TimeActivatedOperate service	148
20.8	TimeActivatedOperateTermination service	148
20.9	AdditionalCauseDiagnosis in negative control service responses	149
20.10	Tracking of control services	152
20.10.1	General	152
20.10.2	Mapping of the Control service tracking (CTS)	152
21	Time and time synchronization model	153
22	Naming conventions	153
23	File transfer	153
23.1	File transfer model	153
23.2	File services	155
23.2.1	GetFile	155
23.2.2	SetFile	161
23.2.3	DeleteFile	163
23.2.4	GetFileAttributeValues	164
24	Conformance	166
24.1	Notation	166
24.2	PICS	166
24.2.1	Profile conformance	166
24.2.2	XML Payload conformance	168
24.3	PICS Statement	189
24.3.1	General	189
24.3.2	Substation configuration language	189
25	Substation Configuration Language (SCL)	189
Annex A (normative)	Communication stack	190
A.1	Overview	190
A.1.1	General	190
A.1.2	XMPP communication profiles	191
A.1.3	Non-XMPP communication profiles	191
A.2	Communication stack	192

A.2.1	Overview of the protocol usage.....	192
A.2.2	Client/server services and communication profiles.....	192
A.2.3	Time sync.....	194
Annex B	(informative) Deployment of XMPP infrastructure.....	196
B.1	General.....	196
B.2	Deployment of XMPP within one XMPP domain.....	196
B.2.1	Use case facility.....	196
B.2.2	Use case hierarchy within a facility.....	197
B.3	Deployment of XMPP while interconnecting more than one XMPP domain.....	198
B.3.1	Interconnection of XMPP Domain.....	198
B.3.2	Definition of a federation communication between XMPP domains.....	204
B.3.3	Interconnection of Domain with federation.....	205
B.4	Communication path outage and recovery.....	209
Annex C	(informative) Security for DER integration based on XMPP.....	210
C.1	General.....	210
C.2	Assumptions and boundary conditions.....	211
C.3	Derivation of security requirements.....	211
C.4	Mapping of security options to XMPP based integration of DER.....	212
C.5	Sequence diagrams.....	213
C.5.1	General.....	213
C.5.2	XMPP and Stream opening.....	213
C.5.3	Stream establishment, ROSTER and presence.....	214
C.5.4	Communication outage.....	215
C.5.5	Request Response (Clear Transfer).....	218
Annex D	(normative) Mapping of services and errors over XMPP stanzas.....	220
Annex E	(informative) Intentional deviations from IEC 61850-8-1 SCSM.....	223
Annex F	(informative) SCL conformance.....	224
Annex G	(normative) XML schema definitions for the XML payload.....	225
G.1	General.....	225
G.2	XML schema of the Virtual API for IEC 61850-8-2.....	225
G.3	XML schema of the applicative payload for IEC 61850-8-2.....	225
G.4	Extension of IEC 62351-4 when E2E security is turned off.....	252
Bibliography		253
Figure 1	– Overview of functionality and profiles.....	22
Figure 2	– Example of XML Payload.....	25
Figure 3	– Generic structure of client/server ACSI services.....	27
Figure 4	– Algorithm for logical node mapping.....	30
Figure 5	– Ordered list of functional constraints.....	31
Figure 6	– Example of Logical Node type description.....	32
Figure 7	– List of the flattened Named Variables corresponding to an LN.....	33
Figure 8	– XML mapping of a LNReference with direct access.....	34
Figure 9	– XML mapping of a LNReference with alternate access.....	34
Figure 10	– Direct XML mapping of a FCD.....	35
Figure 11	– Direct XML mapping of a FCDA.....	35
Figure 12	– Alternate access without array element.....	36

Figure 13 – Alternate access with array element	38
Figure 14 – Alternate access with flattened variable and array element	39
Figure 15 – XML structure of GetServerDirectory-Request (LD)	58
Figure 16 – XML structure of GetServerDirectory-Response (LD)	58
Figure 17 – XML structure of GetServerDirectory-Request (FILE)	60
Figure 18 – XML structure of GetServerDirectory-Response (FILE).....	60
Figure 19 – XML structure of Associate-Request.....	63
Figure 20 – XML structure of Associate-Response	63
Figure 21 – XML structure of GetLogicalDeviceDirectory-Request	68
Figure 22 – XML structure of GetLogicalDeviceDirectory-Response	68
Figure 23 – XML structure of extended GetLogicalDeviceDirectory-Request	70
Figure 24 – XML structure of extended GetLogicalDeviceDirectory-Response.....	70
Figure 25 – XML structure of GetLogicalNodeDirectory-Request.....	72
Figure 26 – XML structure of GetLogicalNodeDirectory-Response	73
Figure 27 – XML structure of extended GetLogicalNodeDirectory-Request (step 1).....	75
Figure 28 – XML structure of extended GetLogicalNodeDirectory-Response (step 1).....	75
Figure 29 – XML structure of extended GetLogicalNodeDirectory-Request (step 2).....	75
Figure 30 – XML structure of extended GetLogicalNodeDirectory-Response (step 2)	76
Figure 31 – XML structure of GetAllDataValues-Request	78
Figure 32 – XML structure of GetAllDataValues-Response.....	79
Figure 33 – XML structure of GetDataValues-Request	80
Figure 34 – XML structure of GetDataValues-Response.....	81
Figure 35 – XML structure of SetDataValues-Request.....	82
Figure 36 – XML structure of SetDataValues-Response	82
Figure 37 – XML structure of GetDataDirectory-Request.....	83
Figure 38 – XML structure of GetDataDirectory-Response	84
Figure 39 – XML structure of extended GetDataDirectory-Request.....	86
Figure 40 – XML structure of extended GetDataDirectory-Response	87
Figure 41 – Mapping of reference to persistent data set within logical device.....	87
Figure 42 – Mapping of reference to persistent data set outside logical device	88
Figure 43 – Mapping of reference to non-persistent data set.....	88
Figure 44 – XML structure of GetDataSetValues-Request	89
Figure 45 – XML structure of GetDataSetValues-Response	90
Figure 46 – XML structure of SetDataSetValues-Request	92
Figure 47 – XML structure of SetDataSetValues-Response.....	92
Figure 48 – XML structure of CreateDataSet-Request.....	94
Figure 49 – XML structure of CreateDataSet-Response	95
Figure 50 – XML structure of DeleteDataSet-Request	97
Figure 51 – XML structure of DeleteDataSet-Response	97
Figure 52 – XML structure of GetDataSetDirectory-Request.....	99
Figure 53 – XML structure of GetDataSetDirectory-Response	100
Figure 54 – XML structure of SelectActiveSG-Request.....	105
Figure 55 – XML structure of SelectActiveSG-Response+	105

Figure 56 – XML structure of SelectActiveSG-Response-.....	105
Figure 57 – XML structure of SelectEditSG-Request	106
Figure 58 – XML structure of SetEditSGValue-Request.....	107
Figure 59 – XML structure of ConfirmEditSGValues	108
Figure 60 – XML structure of GetEditSGValue-Request	108
Figure 61 – XML structure of GetEditSGValue-Response.....	109
Figure 62 – XML structure of GetSGCBValues-Request.....	109
Figure 63 – XML structure of GetSGCBValues-Response	110
Figure 64 – XML structure of Report	115
Figure 65 – XML structure of GetBRCBValues-Request	117
Figure 66 – XML structure of GetBRCBValues-Response	119
Figure 67 – XML structure of SetBRCBValues-Request	121
Figure 68 – XML structure of SetBRCBValues-Response.....	122
Figure 69 – Relationship of LCB attributes to IEC 61850-7-2 log definitions	123
Figure 70 – XML structure of QueryLogByTime-Request.....	126
Figure 71 – XML structure of QueryLogByTime-Response	127
Figure 72 – XML structure of QueryLogAfter-Request	129
Figure 73 – XML structure of Select-Request.....	135
Figure 74 – XML structure of Select-Response	136
Figure 75 – XML structure of SelectWithValue-Request	138
Figure 76 – XML structure of SelectWithValue-Response+	139
Figure 77 – XML structure of SelectWithValue-Response-	139
Figure 78 – XML structure of CommandTermination Request+.....	146
Figure 79 – XML structure of CommandTermination Request-.....	148
Figure 80 – XML structure of InformationReport with AdditionalCauseDiagnosis	150
Figure 81 – Mapping of ACSI GetFile to FileOpen, FileRead, FileClose	156
Figure 82 – XML example of FileOpen Request	158
Figure 83 – XML example of FileOpen Response+.....	158
Figure 84 – XML example of FileRead Request (first)	158
Figure 85 – XML example of FileRead Response+ (first).....	159
Figure 86 – XML example of FileRead Request (second).....	159
Figure 87 – XML example of FileRead Response+ (second)	159
Figure 88 – XML example of FileClose Request.....	159
Figure 89 – XML example of FileClose Response	159
Figure 90 – Mapping of ACSI SetFile service	161
Figure 91 – XML example of ObtainFile Request	162
Figure 92 – XML example of ObtainFile Response	162
Figure 93 – XML example of DeleteFile Request.....	164
Figure 94 – XML example of DeleteFile Response	164
Figure 95 – XML example of GetFileAttributeValues Request	165
Figure 96 – XML example of GetFileAttributeValues Response	166
Figure 97 – VariableSpecification for LDevice/MHA11.HA.phsAHar(7).cVal.mag.f.....	178

Figure 98 – Shorter VariableSpecification for LDevice/MHAI1.HA.phsAHar(7).cVal.mag.f	179
Figure 99 – Non conformant VariableSpecification I	180
Figure 100 – Non conformant VariableSpecification II	181
Figure 101 – VariableSpecification for LDevice/MHAI1.HA.phsAHar(7) [MX]	182
Figure 102 – Shorter VariableSpecification for LDevice/MHAI1.HA.phsAHar(7) [MX]	183
Figure A.1 – Overview of functionality and profiles	191
Figure A.2 – OSI reference model and profiles	192
Figure B.1 – Facility domain	196
Figure B.2 – Hierarchical Aggregation at facility	197
Figure B.3 – DER Management System at facility	198
Figure B.4 – Facility Management integration at DSO	199
Figure B.5 – Multiple facilities at DSO	200
Figure B.6 – VPP and contracted DERs	201
Figure B.7 – indirect control using VPP JIDs	202
Figure B.8 – VPP direct control using VPP JIDs	202
Figure B.9 – VPP direct control using DSO JIDs	203
Figure B.10 – DSO Indirect control with VPP JIDs	203
Figure B.11 – DSO direct control	204
Figure B.12 – Concept of federation in XMPP	205
Figure B.13 – Federation DSO – Facility	206
Figure B.14 – Use of federation with VPP	206
Figure B.15 – communication with VPP JIDs – indirect control	207
Figure B.16 – VPP communication with VPP JIDs – direct control	207
Figure B.17 – VPP communication with DSO JIDs	208
Figure B.18 – DSO Communication with VPP JIDs	209
Figure C.1 – Base system for discussion of IT security requirements	210
Figure C.2 – XMPP Stream establishment – IEC 61850 Server to the XMPP Server	214
Figure C.3 – XMPP Stream establishment – IEC 61850 Client to the XMPP Server	214
Figure C.4 – Stream establishment, roster and presence	215
Figure C.5 – Communication outage – Loss of link	216
Figure C.6 – Communication outage – Presence unavailable	217
Figure C.7 – Request response	218
Figure C.8 – Request – Abort	219
Table 1 – Services requiring client/server Communication Profile	23
Table 2 – Mapping of ACSI classes on MMS concepts	30
Table 3 – Mapping of ACSI BasicTypes	40
Table 4 – PhyComAddr structure for Layer 2 communication	44
Table 5 – PhyComAddr for UPD/IP communication	45
Table 6 – GetNameList conflicting IEC 61850 objectClass and objectScope	47
Table 7 – Service error mappings for ACSI services using GetNameList	47
Table 8 – Read service error mappings	48

Table 9 – Write service error mappings	49
Table 10 – GetFileAttributeValues service error mappings	50
Table 11 – Encoding of IEC 61850-7-2 TimeQuality	51
Table 12 – Encoding of the TriggerConditions	52
Table 13 – Encoding of the ReasonForInclusionInReport	52
Table 14 – Encoding of the ReasonForInclusionInLog	53
Table 15 – Encoding of the RCBReportOptions	53
Table 16 – Encoding of the SVMMessageOptions	53
Table 17 – Encoding of the CheckConditions	54
Table 18 – Encoding of IEC 61850-7-2 quality	55
Table 19 – Examples of data values encoding	56
Table 20 – Mapping of ACSI GetServerDirectory (LOGICAL DEVICE)	57
Table 21 – Mapping of ACSI GetServerDirectory (FILE)	59
Table 22 – Association model versus communication profiles	60
Table 23 – Mapping of ACSI Associate service	62
Table 24 – Description of Associate request elements	62
Table 25 – Description of Associate response elements	63
Table 26 – Associate ACSI service error mappings	64
Table 27 – Mapping of ACSI Release service	66
Table 28 – Release service error mappings	66
Table 29 – Mapping of ACSI GetLogicalDeviceDirectory	67
Table 30 – Extended mapping of ACSI GetLogicalDeviceDirectory	69
Table 31 – Objectclasses for GetLogicalNodeDirectory service	70
Table 32 – Mapping of ACSI GetLogicalNodeDirectory	72
Table 33 – Extended mapping of ACSI GetLogicalNodeDirectory	74
Table 34 – Mapping of ACSI GetAllDataValues	77
Table 35 – Mapping of GetDataValues service parameters	80
Table 36 – Mapping of SetDataValues service parameters	81
Table 37 – Mapping of GetDataDirectory service parameters	83
Table 38 – GetDataDirectory service error mappings	85
Table 39 – Extended mapping of ACSI GetDataDirectory	86
Table 40 – Mapping of GetDataSetValues service parameters	88
Table 41 – GetDataSetValues error mappings	91
Table 42 – Mapping of SetDataSetValues service parameters	91
Table 43 – SetDataSetValues error mappings	93
Table 44 – Mapping of CreateDataSet service parameters	94
Table 45 – CreateDataSet service error mappings	96
Table 46 – Mapping of DeleteDataSet service parameters	96
Table 47 – DeleteDataSet service error mappings	98
Table 48 – Mapping of GetDataSetDirectory service parameters	99
Table 49 – GetDataSetDirectory service error mappings	101
Table 50 – Mapping of ACSI ServiceType values	101
Table 51 – Mapping of ACSI errorCode values	103

Table 52 – Mapping of CDC LTS.....	103
Table 53 – Mapping of CDC GTS.....	104
Table 54 – Mapping of SGCB.....	104
Table 55 – BRCB structure	111
Table 56 – URCB structure	112
Table 57 – Order of AccessResults for Report	113
Table 58 – Mapping of GetBRCBValues service parameters	116
Table 59 – Mapping of SetBRCBValues service parameters	120
Table 60 – LCB structure	123
Table 61 – Mapping of QueryLogByTime service parameters	125
Table 62 – ServiceError mappings for Log services	129
Table 63 – Mapping of QueryLogAfter-Request parameters	129
Table 64 – Log conformance requirements	130
Table 65 –TypeDescription definition for GoCB structure	130
Table 66 – Controllable service parameters	133
Table 67 – Mapping of IEC 61850-7-2 control model to control components.....	133
Table 68 – Mapping of control services	134
Table 69 – Mapping of Select parameters	135
Table 70 – SelectWithValue service parameter mapping	136
Table 71 – Mapping of SelectWithValue parameters	137
Table 72 – SelectWithValue, Oper and Cancel DataAccessError specification	139
Table 73 – Cancel service parameter mapping.....	140
Table 74 – Mapping of the Cancel service.....	141
Table 75 – Operate service parameter mapping	142
Table 76 – Mapping of the Operate service	143
Table 77 – Mapping of the CommandTermination service	145
Table 78 – Definition of LastApplError variable structure.....	149
Table 79 – Mapping of ACSI AddCause values	152
Table 80 – Mapping of CDC CTS	153
Table 81 – Mapping of ACSI file class to MMS file object.....	153
Table 82 – Reserved file suffixes	154
Table 83 – Mapping of ACSI GetFile service	157
Table 84 – GetFile service error mappings.....	158
Table 85 – Mappings of ACSI ServiceErrors to FileOpen Service Errors	160
Table 86 – Mappings of ACSI ServiceErrors to FileRead Service Errors	160
Table 87 – Mappings of ACSI ServiceErrors to FileClose Service Errors	161
Table 88 – Mapping of ACSI SetFile parameters	162
Table 89 – Mappings of ACSI ServiceErrors to ObtainFile Service Errors	163
Table 90 – Mapping of ACSI DeleteFile service	163
Table 91 – Mappings of ACSI ServiceErrors to DeleteFile Service Errors.....	164
Table 92 – Mapping of ACSI GetFileAttributeValues parameters	165
Table 93 – PICS for A-Profile support	167
Table 94 – PICS for Time Sync A-Profile support.....	167

Table 95 – PICS for T-Profile support	168
Table 96 – MMS InitiateRequest general parameters	168
Table 97 – MMS InitiateResponse general parameters	169
Table 98 – MMS service supported conformance table	169
Table 99 – MMS Parameter CBB	172
Table 100 – GetNameList conformance statement	173
Table 101 – GetCapabilityList conformance statement	173
Table 102 – GetDomainAttributes conformance statement	174
Table 103 – Status conformance statement	174
Table 104 – Cancel conformance statement	175
Table 105 – Identify conformance statement	175
Table 106 – AlternateAccess conformance statement	176
Table 107 – AlternateAccessSelection conformance statement	176
Table 108 – VariableAccessSpecification conformance statement	183
Table 109 – VariableSpecification conformance statement	184
Table 110 – Read conformance statement	184
Table 111 – Write conformance statement	184
Table 112 – InformationReport conformance statement	185
Table 113 – GetVariableAccessAttributes conformance statement	185
Table 114 – DefineNamedVariableList conformance statement	185
Table 115 – GetNamedVariableListAttributes conformance statement	186
Table 116 – DeleteNamedVariableList conformance statement	186
Table 117 – ReadJournal conformance statement	186
Table 118 – EntryContent conformance statement	187
Table 119 – FileDirectory conformance statement	188
Table 120 – FileOpen conformance statement	188
Table 121 – FileRead conformance statement	188
Table 122 – FileClose conformance statement	189
Table 123 – Allowed P-Type definitions for client/server addressing	189
Table A.1 – Service and protocols for client/server communication A-Profile	193
Table A.2 – Service and protocols for client/server XMPP T-Profile	193
Table A.3 – Time sync A-Profile	194
Table A.4 – Time sync T-Profile	195
Table D.1 – ACSI services mapping over XMPP stanzas	220
Table D.2 – Error mapping over XMPP stanzas	222
Table F.1 – SCL conformance degrees	224
Table F.2 – Supported ACSI services for SCL.2	224

INTERNATIONAL ELECTROTECHNICAL COMMISSION

**COMMUNICATION NETWORKS AND
SYSTEMS FOR POWER UTILITY AUTOMATION –****Part 8-2: Specific Communication Service Mapping (SCSM) –
Mapping to Extensible Messaging Presence Protocol (XMPP)**

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 the 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 61850-8-2 has been prepared by IEC technical committee 57: Power systems management and associated information exchange.

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

FDIS	Report on voting
57/2020/FDIS	57/2039/RVD

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

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

This IEC standard includes Code Components i.e components that are intended to be directly processed by a computer. Such content is any text found between the markers <CODE BEGINS> and <CODE ENDS>, or otherwise is clearly labeled in this standard as a Code Component.

The purchase of this IEC standard carries a copyright license for the purchaser to sell software containing Code Components from this standard to end users either directly or via distributors, subject to IEC software licensing conditions, which can be found at: www.iec.ch/CCv1.

A list of all parts in the IEC 61850 series, published under the general title *Communication networks and systems for power utility automation*, can be found on the IEC website.

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

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

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.

INTRODUCTION

This part of IEC 61850 is part of a set of specifications which details layered utility communication architecture.

The usage of the IEC 61850 communication standard is largely spreading over all the domains connected to the smart grid, pushing the usage of technologies adapted to the connection of a very large number of applications and devices across the intra/inter-net. The involved domains use already well established protocols typically for exchanging data with IT level applications like resource planning, asset and maintenance management, etc. Therefore, it becomes imperative to provide an integration strategy that allows the integration of IEC 61850 into these various disparate protocols and information.

In this context, this part of IEC 61850 describes a specific communication service mapping (SCSM) over the Extensible Messaging and Presence Protocol (XMPP), providing detailed information on how to create and exchange concrete communication messages that implement abstract services and models specified in IEC 61850-7-4, IEC 61850-7-3, and IEC 61850-7-2.

This mapping is intended to be utilized between all kinds of utility Distributed Energy Resource devices and their related management systems, in particular over public networks.

NOTE This part of IEC 61850 does not provide tutorial material. For this purpose, IEC 61850-5 and IEC 61850-7-1 can be read in conjunction with IEC 61850-7-2.

COMMUNICATION NETWORKS AND SYSTEMS FOR POWER UTILITY AUTOMATION –

Part 8-2: Specific Communication Service Mapping (SCSM) – Mapping to Extensible Messaging Presence Protocol (XMPP)

1 Scope

1.1 General

This part of IEC 61850 specifies a method of exchanging data through any kinds of network, including public networks. Among the various kinds of services specified in IEC 61850-7-2, only the client/server and time synchronization services are considered so far.

NOTE Client/server services of GOOSE and SMV models are mapped as well (see Table 1).

For the client/server services, the principle is to map the objects and services of the ACSI (Abstract Communication Service Interface defined in IEC 61850-7-2) to XML messages transported over XMPP. The mapping description includes mainly three aspects:

- The usage of the XMPP protocol itself, describing in details which features are really used and how they are used by the mapping (see Clause 6).
- How to achieve end-to-end secured communications (see Clause 7).
- The description of the XML payloads corresponding to each ACSI service thanks in particular to the XML Schema and XML message examples (starting at Clause 9).

NOTE 1 This document does not address the detailed usage of the XMPP protocol.

NOTE 2 This document does not address system management services.

NOTE 3 For the information of people familiar with the mapping defined in IEC 61850-8-1, the XML messages defined in the present document are derived from those defined in IEC 61850-8-1 but with an XML encoding instead of a binary one. In this way implementing gateways between IEC 61850-8-1 and IEC 61850-8-2 is very straightforward in both directions. However reading IEC 61850-8-1 is not necessary to understand the present document except when it is used in conjunction with one of the GOOSE mappings described in IEC 61850-8-1.

1.2 Namespace name and version

This new section is mandatory for any IEC 61850 namespace (as defined by IEC 61850-7-1).

The parameters which identify this release of the SCSM_8_2 namespace `xmlns="http://www.iec.ch/61850/2018/SCSM_8_2"` are:

- Namespace Version: 2018
- Namespace Revision: A
- Namespace Release: 1
- Namespace release date: 2018-12

Edition	Publication date	Webstore	Namespace
Edition 1.0	2018-12	IEC 61850-8-2:2018	IEC 61850-8-2:2018

1.3 Code Component distribution

The Code Components included in this IEC standard are also available as electronic machine readable file at:

http://www.iec.ch/tc57/supportdocuments/IEC_61850-8-2.2018_ed1.0.XSD.2018A1.full.zip

The Code Component(s) included in this IEC standard are potentially subject to maintenance works and users shall select the latest release in the repository located at: <https://www.iec.ch/tc57/supportdocuments>.

The latest version/release of the document will be found by selecting the file IEC 61850-8-2.2018_ed1.0.XSD.{VersionStateInfo}.full.zip with the filed VersionStateInfo of the highest value.

In case of any differences between the downloadable code mentioned above and the IEC pdf published content, the downloadable code(s) is(are) the valid one; it may be subject to updates. See history files.

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 61850 (all parts), *Communication networks and systems for power utility automation*

IEC TS 61850-2, *Communication networks and systems in substations – Part 2: Glossary*

IEC 61850-5, *Communication networks and systems for power utility automation – Part 5: Communication requirements for functions and device models*

IEC 61850-6, *Communication networks and systems for power utility automation – Part 6: Configuration description language for communication in electrical substations related to IEDs*

IEC 61850-7-1, *Communication networks and systems for power utility automation – Part 7-1: Basic communication structure – Principles and models*

IEC 61850-7-2:2010, *Communication networks and systems for power utility automation – Part 7-2: Basic information and communication structure – Abstract communication service interface (ACSI)*

IEC 61850-7-2:2010/AMD1:2018

IEC 61850-7-3, *Communication networks and systems for power utility automation – Part 7-3: Basic communication structure – Common data classes*

IEC 61850-7-4, *Communication networks and systems for power utility automation – Part 7-4: Basic communication structure – Compatible logical node classes and data object classes*

IEC 61850-8-1:2011, *Communication networks and systems for power utility automation – Part 8-1: Specific communication service mapping (SCSM) – Mappings to MMS (ISO 9506-1 and ISO 9506-2) and to ISO/IEC 8802-3*

IEC 61850-8-1:2011/AMD1:2018

IEC 62351 (all parts), *Power systems management and associated information exchange – Data and communications security*

IEC 62351-4:2018, *Power systems management and associated information exchange – Data and communications security – Part 4: Profiles including MMS and derivatives*¹

IEC 62351-6, *Power systems management and associated information exchange – Data and communications security – Part 6: Security for IEC 61850*²

IEC TR 62357-200, *Power systems management and associated information exchange – Part 200: Guidelines for migration from Internet Protocol version 4 (IPv4) to Internet Protocol version 6 (IPv6)*

ISO/IEC 7498-1:1994, *Information technology – Open systems interconnection – Basic reference model: The basic model*

ISO/IEC 8824-1:2015 [ITU-T X.680:2015], *Information technology – Abstract Syntax Notation One (ASN.1): Specification of basic notation*

ISO/IEC 8825-4:2015 [ITU-T X.693:2015], *Information technology – ASN.1 encoding rules: XML Encoding Rules (XER)*

ISO 9506 (all parts), *Industrial automation systems – Manufacturing message specification*

ISO 9506-1:2003, *Industrial automation systems – Manufacturing message specification – Part 1: Service definition*

ISO 9506-2:2003, *Industrial automation systems – Manufacturing message specification – Part 2: Protocol specification*

IEEE C37.111:1999, *IEEE standard common format for transient data exchange (COMTRADE) for power systems*

IEEE 754:1985, *IEEE Standard for binary floating-point arithmetic*

RFC 768, *User datagram protocol*, IETF, available at <http://www.ietf.org>

RFC 791, *Internet protocol – DARPA Internet Program – Protocol specification*, IETF, available at <http://www.ietf.org>

RFC 792, *Internet control message protocol – DARPA internet program – Protocol specification*, IETF, available at <http://www.ietf.org>

RFC 793, *Transmission control protocol – DARPA internet program – Protocol specification*, IETF, available at <http://www.ietf.org>

RFC 826, *An Ethernet address resolution protocol or converting network protocol addresses to 48.bit Ethernet address for transmission on Ethernet hardware*, IETF, available at <http://www.ietf.org>

RFC 919, *Broadcasting internet datagrams*, IETF, available at <http://www.ietf.org>

¹ Under preparation. Stage at the time of publication: IEC/DECFDIS 62351-4:2018.

² At the time of publication, a new edition was under preparation.

RFC 922, *Broadcasting internet datagrams in the presence of subnets*, IETF, available at <http://www.ietf.org>

RFC 950, *Internet standard subnetting procedure*, IETF, available at <http://www.ietf.org>

RFC 1112, *Host extensions for IP multicasting*, IETF, available at <http://www.ietf.org>

RFC 2460, *Internet protocol, Version 6 (IPv6) specification*, IETF, available at <http://www.ietf.org>

RFC 3629, *UTF-8, a transformation format of ISO 1646 – IETF*, available at <http://www.ietf.org>

RFC 4422, *Simple authentication and security layer (SASL)*, available at <http://www.ietf.org>

RFC 5246:2008, *The TLS protocol version 1.2*, available at <http://www.ietf.org>

RFC 5905, *Network time protocol (NTP) Version 4: Protocol and algorithms specification*, IETF, available at <http://www.ietf.org>

RFC 6120, *Extensible messaging and presence protocol (XMPP): Core*, available at <http://www.ietf.org>

RFC 6121, *Extensible messaging and presence protocol (XMPP): Instant messaging and presence*, available at <http://www.ietf.org>

RFC 6122, *Extensible messaging and presence protocol (XMPP): Address format*, available at <http://www.ietf.org>

XEP-0198, *Stream management*

NOTE This specification defines an XMPP protocol extension for active management of an XML stream between two XMPP entities, including features for stanza acknowledgements and stream resumption.

XEP-0199, *XMPP Ping*

NOTE This specification defines an XMPP protocol extension for sending application-level pings over XML streams. Such pings can be sent from a client to a server, from one server to another, or end-to-end.

3 Terms and definitions

For the purposes of this document, the terms and definitions given in IEC TS 61850-2 and the following apply.

ISO and IEC maintain terminological databases for use in standardization at the following addresses:

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

3.1

class

description of a set of objects that share the same attributes, services, relationships, and semantics

[SOURCE: IEC 61850-7-2:2010, 3.1]

3.2

client

entity that requests a service from a server and that receives unsolicited messages from a server

[SOURCE: IEC 61850-7-2:2010, 3.2]

3.3

device

entity that performs control, actuating and/or sensing functions and interfaces to other such entities within an automation system

Note 1 to entry: Devices alone do not perform energy transport functions.

[SOURCE: IEC 61850-7-2:2010, 3.3]

3.4

logical device

entity that represents a set of typical substation functions

[SOURCE: IEC 61850-7-2:2010, 3.6]

3.5

logical node

entity that represents a typical substation function

[SOURCE: IEC 61850-7-2:2010, 3.7]

3.6

physical device

entity that represents the physical parts of a device (hardware and operating system, etc.)

Note 1 to entry: Physical devices host logical devices.

[SOURCE: IEC 61850-7-2:2010, 3.8]

3.7

application and transport profiles

A-Profile and T-Profile

set of protocols for a specific purpose

3.8

data attribute

named element of data object which has a specific type

3.9

data object

meaningful structured information of applications located in an automation device which can be read or written

Note 1 to entry: This definition is in accordance with the DataObject definition of IEC 61850-7-2. The MMS use of DataObject refers to data types and potential instantiations of MMS variables.

3.10

server

entity that provides services to clients or issues unsolicited messages

3.11**private network**

network used by a unique entity which masters all the data flows and whose performance is guaranteed in terms of bandwidth, throughput, transmission delay, availability, etc.

Note 1 to entry: If these conditions are not fulfilled, the network is considered as public.

Note 2 to entry: A private network may be based on a public or shared infrastructure, as soon as the level of services can be guaranteed.

3.12**smart grid**

application of computer intelligence and networking abilities to the electricity delivery system so it monitors, protects and automatically optimizes the operation of its interconnected elements, from the central and distributed generator through the high-voltage network and distribution system, to industrial users and building automation systems, to energy storage installations and to end-use consumers and their thermostats, electric vehicles, appliances and other household devices

4 Abbreviated terms

ACSI	Abstract Communication Service Interface
BRCB	Buffered Report Control Block
CDC	Common Data class
DER	Distributed Energy Resource
DNS	Domain Name System
DSO	Distribution System Operator
E2E	End-to-end
FC	Functional Constraint
FCD	Functionally Constrained Data
IED	Intelligent Electronic Device
IETF	Internet Engineering Task Force
IP	Internet Protocol
JID	Jabber Identifier
LAN	Local Area Network
LCB	Log Control Block
LD	Logical Device
LN	Logical Node
MMS	Manufacturing Message Specification (ISO 9506)
OSI	Open Systems Interconnection
PDU	Protocol Data Unit
PICS	Protocol Implementation Conformance Statement
PIXIT	Protocol Implementation Extra Information
RBAC	Role-Based Access Control
RFC	Request for Comments
SBO	Select Before Operate
SCL	substation automation System Configuration Language (IEC 61850-6)
SCSM	Specific Communication Service Mapping
SCSM 8-1	Specific Communication Service Mapping as defined by IEC 61850-8-1
SCSM 8-2	Specific Communication Service Mapping as defined by IEC 61850-8-2
SG	Setting Group

SNTP	Simple Network Time Protocol
SV	Sampled Values
TCP	Transmission Control Protocol
UCA	Utility Communication Architecture
URCB	Unbuffered Report Control Block
UTC	Coordinated universal time
VMD	Virtual Manufacturing Device
VPP	Virtual Power Plant
WAN	Wide Area Network
XER	XML Encoding Rules
XMPP	Extensible Messaging Presence Protocol
XSD	XML Schema Definition

5 Overview

5.1 General

The purpose of this document is to provide detailed instructions/specifications as to the mechanisms and rules required to implement the core ACSI services specified in IEC 61850-7-2, while making use of XMPP, as well as the usage of SNTP for time synchronization (see Figure 1).

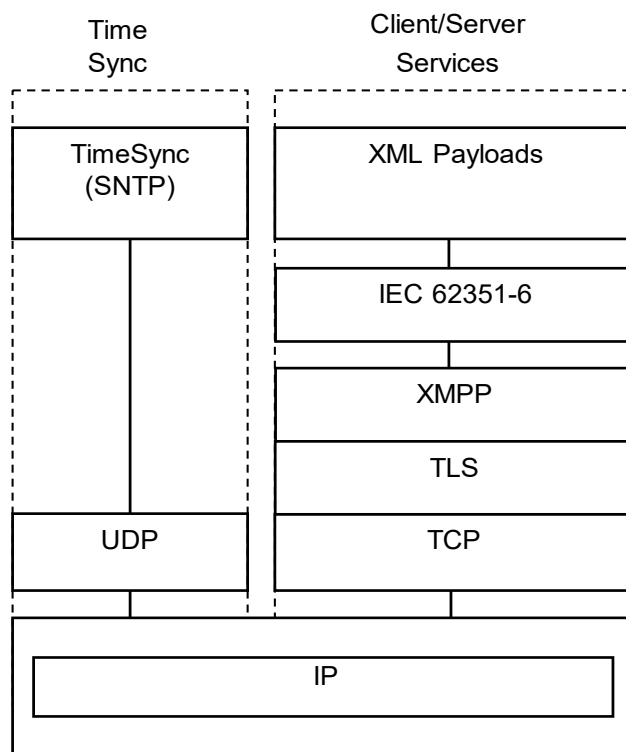


Figure 1 – Overview of functionality and profiles

Further details pertaining to the use and actual specification of each profile may be found in Clause 6.

NOTE The stack profiles described here start at the IP layer. Which specifications must be used for the layers below IP is out of the scope of this SCSM.

5.2 Mapping of client/server services

5.2.1 General

The client/server communication profile shall be used for any implementation claiming conformance to this document and declaring support for one of the IEC 61850-7-2 services shown in Table 1.

Table 1 – Services requiring client/server Communication Profile

IEC 61850-7-2 model	IEC 61850-7-2 service
Server	GetServerDirectory
Association	Associate
	Abort
	Release
Logical Device	GetLogicalDeviceDirectory
Logical Node	GetLogicalNodeDirectory
	GetAllDataValues
Data	GetDataValues
	SetDataValues
	GetDataDirectory
	GetDataDefinition
Data Set	GetDataSetValues
	SetDataSetValues
	CreateDataSet
	DeleteDataSet
	GetDataSetDirectory
Setting Group Control Block	SelectActiveSG
	SelectEditSG
	SetEditSGValue
	ConfirmEditSGValues
	GetEditSGValue
	GetSGCBValues
Report Control Block	Report
	GetBRCBValues
	SetBRCBValues
	GetURCBValues
	SetURCBValues
LOG Control Block	GetLCBValues
	SetLCBValues
	GetLogStatusValues
	QueryLogByTime
	QueryLogAfter
GOOSE ^a	GetGoCBValues
	SetGoCBValues
Multicast Sampled Values ^a	GetMSVCBValues
	SetMSVCBValues
Control	Select
	SelectWithValue
	Cancel
	Operate
	CommandTermination
	TimeActivatedOperate
FILE transfer	GetFile
	SetFile
	DeleteFile
	GetFileAttributeValues
^a The services SendGOOSEMessage, GetGoReference, GetGOOSEElementNumber, SendMSVMessage, GetMsvReference and GetMSVElementNumber are not mapped within the present document because an XMPP mapping would not fulfill the required performances.	

The request and response messages of the client/server ACSI services shall be represented by XML messages transported between the caller application and the called application over the XMPP protocol. Per the XMPP protocol, an XML stream has to be first established between the two applications and an XMPP server. Then the XML messages representing the ACSI services shall be embedded within XML constructs defined by XMPP and called stanzas. Each stanza shall be sent by the caller within its XMPP stream, then routed by the XMPP server and finally delivered to the called application through its own XMPP stream.

NOTE See A.2.2 for a formal description of the communication layers involved in this profile.

5.2.2 XML payloads

When building the XML messages transported over XMPP, implementations shall be aware that each XML message is made up of three separate and nested parts (see Figure 2):

- The service PDU representing the ACSI service itself is the deeper nested level. It is represented in the XML messages by an element such as <confirmed-RequestPDU> or <confirmed-ResponsePDU> for respectively a service request and a service response. A major goal of the present document is to describe how to build these applicative payloads. Their structure shall be formally defined by the XML schema provided in Annex G.3.
- The service PDU is then wrapped within end-to-end security elements, holding security related information such as certificates, session keys and other parameters. This information is used to establish and manage an end-to-end secured channel between the caller and called applications.
- The association context element is the direct child element of the XMPP stanza, wrapping the service PDU and the security elements. It transports an identifier used to create and manage the association between the applications. Figure 2 presents an example of XML payload corresponding to an ACSI GetDataValues request, assuming that encryption of the applicative payload is not activated.

The two latter parts mentioned above are specified in IEC 62351-4.

<i><Element(s) for management of association context></i>	Association context
<i><Elements for management of end-to-end security></i>	Security elements
<pre> < > [...]</pre>	
<pre> <confirmed-RequestPDU> <invokeID>2676121</invokeID> <ConfirmedServiceRequest> <read specificationWithResult="false" > <variableAccessSpecification> <listOfVariable> <SEQUENCE> <variableSpecification> <name> <domain-specific> <domainId>IEDNameLDINst</domainId> <itemId>LLN0\$ST\$Beh\$stVal</itemId> </domain-specific> </name> </variableSpecification> </SEQUENCE> </listOfVariable> </variableAccessSpecification> </read> </ConfirmedServiceRequest> </confirmed-RequestPDU></pre>	Service PDU (applicative payload)
<pre> </ > [...]</pre>	
<i></Elements for management of end-to-end security></i>	Security elements
<i></Element(s) for management of association context ></i>	Association context

Figure 2 – Example of XML Payload

5.2.3 Implementation agreements

Support of the end-to-end security mechanism mentioned above is mandatory for any implementation conforming to the present document. However, for a given deployment the end-to-end security feature may be turned off. In that case the behavior of both client and server applications shall be as follows:

- Turning-off the end-to-end security mechanism is a dedicated configuration of both IEC 61850 applications, client and server. This configuration is out of the scope of the present document.
- When the two communicating applications are properly configured for operating without end-to-end security, the exchanged payloads shall not include any security elements and the service PDU shall be directly wrapped within the association context part (see G.4).
- The usage of the XMPP stanzas and the management of the association context shall remain identical to the case when end-to-end security is applied and shall be therefore as described in IEC 62351-4.
- In case the client and server entities are configured differently, one using the end-to-end security and the other not, no interaction is possible. Any request for initiating a communication shall fail.

5.2.4 XMPP

XMPP is a protocol defined at the Internet Engineering Task Force (IETF) by RFC 6120, RFC 6121 and RFC 6122. Beside the core specifications, the XMPP Standards Foundation develops extensions called XEPs.

For the present SCSM the role of XMPP is to transport the XML payloads between two IEC 61850 applications

Clause 6 gives details about how the present mapping uses XMPP including both the core specifications and the possible XEPs.

5.3 Time sync services

The time synchronization services specified in IEC 61850-7-2 shall be mapped on the IETF Simple Network Time Protocol (SNTP).

NOTE 1 Using SNTP on public networks may require particular engineering techniques for ensuring appropriate cyber-security (e.g. usage of redundant clocks, discovery of trusted clocks). Defining or recommending such techniques goes beyond the scope of this document.

NOTE 2 See A.2.3 for a formal description of the communication layers involved in this profile.

6 Usage of XMPP

6.1 Principles

XMPP is based on a distributed architecture of XMPP servers. From an XMPP point of view, an application complying with the present SCSM shall act as an XMPP client connected to one or several XMPP servers.

NOTE The XMPP servers can be operated either by one of the actors involved in the IEC 61850 interactions at applicative level or by a third party provider.

Each IED shall have an account on the XMPP server to which it is connected and the resource part of the JID used to identify a given connection shall be fixed. In order that the same connection can be used by any IEC protocol based on XMPP, the generic value IEC shall be used.

Example of a full JID: myIED@DSO.org/IEC.

6.2 Connection establishment

6.2.1 General

For establishing a connection to the XMPP server, an IED shall perform the following steps:

- Initiate a TCP connection to the server
- Open an XML stream
- Negotiate stream features
- Authenticate with the XMPP server
- Select a resource
- Request the roster and send initial presence

These different steps are specified in details by XMPP (RFC 6120). Additionally the rules specified in Subclauses 6.2.2 to 6.2.3 shall apply.

6.2.2 Usage of TLS and SASL

TLS shall be used in any case and shall be implemented as specified in IEC 62351-6. However several cases shall be considered:

1. XMPP over TLS where the TLS negotiation is part of the XMPP Stream negotiation;
2. XMPP over TLS where the TLS session is a tunnel over which the XMPP Stream is established;

3. XMPP over TLS where the TLS negotiation is part of the XMPP Stream negotiation; however within an existing TLS tunnel over which the XMPP Stream is established (combination of 1 and 2).

SASL shall be used in any case for authentication of the XMPP endpoints. However several cases shall be considered, depending on how TLS is used:

- SASL EXTERNAL mechanism (where authentication is given by the certificate exchanged during the <stream:feature/> <starttls/> negotiation) may be achieved for 1 and 3, as far as the TLS certificate used during the negotiation includes the JID identity.
- SASL PLAIN mechanism (a simple clear text password mechanism) may also be deployed for 1 and 3, and it becomes mandatory as soon as the TLS certificate used during the negotiation does not include the JID identity.
- SASL PLAIN mechanism shall be used as the only option for 2 as the certificate used in this case by TLS would not be available at the XMPP component level.

One of the proposed option for both TLS and SASL shall be selected and deployed, depending on the transport infrastructure, local regulation and/or local policy.

6.2.3 Stream Compression

XMPP Stream Compression may be used (see XEP 0138), but note that it will be not efficient for stanzas involved in an end-to-end security session with encryption.

6.3 Mapping of ACSI services

Most of the client/server services are solicited services defined by three elements: the Request, the positive Response and the negative Response (see Figure 3).

Parameter name
Request
<request parameters>
Response+
<response parameters>
Response-
<service errors>

Figure 3 – Generic structure of client/server ACSI services

Solicited services shall be implemented by <iq> stanzas. The Request message shall be implemented by an <iq> stanza with the attribute “type” equal to “set” or “get” and the Response+ message shall be implemented by an <iq> stanza with attribute “type” set to “result”. Response- messages shall be transmitted as an <iq/> stanza with type=“result” or as a <message/> with type=“normal”.

Unsolicited services shall be implemented as <message/> stanzas of type “normal”.

NOTE The payloads representing the ACSI services, i.e. the Service PDUs specified by the present document, are embedded within elements used for management of end-to-end security and association context (see Figure 2). The two latter kinds of elements are specified by IEC 62351-4. The errors encountered during the management of these elements is specified in IEC 62351-4 as well.

Errors occurring at XMPP level shall use the error mechanism defined by XMPP.

Table D.1 defines for each ACSI service which kind of stanza and which value of the “type” attribute shall be used. Note that the mapping of the Associate service (last three rows of the table) is rather described in IEC 62351-4.

The various error cases (Response- and errors raised at XMPP level) are described in 8.3.3.4 and Table D.2.

6.4 Usage of XMPP presence

Two-party associations shall be monitored by using the presence information of the IEC 61850 entities (i.e. the IEC 61850 client and the IEC 61850 server).

During the Associate request, the IEC 61850 client shall subscribe to the presence of the IEC 61850 server (if not already done before). The server may also subscribe to the client presence. An entity detecting the unavailable presence status of its associated peer shall generate a local Abort.

The establishment of TCP connections and XMPP streams between an IEC 61850 communicating entity and an XMPP server is not dedicated to one association and therefore not part of it. Both shall exist before the Associate service is called.

6.5 Usage of the Roster

As recommended in RFC 6121, an XMPP client (and then an IEC 61850 client or server) shall request its Roster at the end of the stream creation.

Then, for security consideration, the XMPP client shall check the consistency of its Roster and eventually update it accordingly.

6.6 XMPP extensions

6.6.1 Usage of XMPP PING – XEP 0199

XMPP servers may use various methods for checking the presence of the connected clients, typically TCP keepalive which detects only broken TCP connections or XMPP PING (XEP 0199), which detects problems both on TCP and XMPP levels. As a minimum requirement an XMPP client shall be able to answer to XMPP PING stanzas as defined in XEP 0199.

NOTE Other usages of this XEP are outside the scope of this SCSM, for example entities may also send XMPP pings for checking the availability of a peer, either IEC 61850 entity or XMPP server.

6.6.2 Usage of Stream Management – XEP 0198

When they want to improve the network reliability, XMPP entities may enable the stream management in conformance with the XEP 0198. The term XMPP entity covers here not only the XMPP clients but also the XMPP servers.

An implementation declaring support for XEP 0198 shall manage the acknowledgements as defined by this XEP, for checking that the number of stanzas sent by one entity corresponds to the number of stanzas handled by the receiving entity.

It is recommended that an entity resume the stream and resend the non handled stanzas when it detects that these two numbers are different, or when the XMPP stream has been broken. For a minimum functional level, the resumption may be replaced by a comprehensive error raised to the upper stack layers.

NOTE The strategy for exchanging the acks (periodically, every stanza or every N stanza), depends on the scenarios.

As resumption scenarios can imply the resending of already handled stanzas, all entities (even those not supporting the XEP 0198) shall check the stanza "id" attribute in order to ignore the duplicates.

6.7 Implementation agreements – XMPP PING – XEP 0199

The delay between two XMPP PING depends on the XMPP server configuration. The recommended default value is 60 s.

7 End-to-end security

The end-to-end security mechanism used in the present SCSM is specified by IEC 62351-4. IEC 62351-4 applies to both the present SCSM and IEC 61850-8-1 and gives the principles and the abstract definitions of the security elements exchanged between the caller and the called endpoints.

The end-to-end security session is created together with the creation of the ACSI association during the initial exchange of XMPP messages corresponding to the ACSI Associate. Within these messages, the payload corresponding to the ACSI Associate service is completed by security elements required for establishing the end-to-end session: tokens holding the necessary cryptographic parameters like for example digital certificates, as well as signatures.

During this handshake, the mutual authentication of the two endpoints is performed based on the exchanged digital certificates and signatures. At the same time a Diffie Hellman key exchange takes place, providing the session keys necessary to secure the subsequent data exchanges by using symmetric cryptographic algorithms for integrity protection and data confidentiality during data transfer.

The details of the mechanism including in particular the available options (e.g. only signing or also encrypting the payload) are defined in IEC 62351-4.

8 Payload description

8.1 XSD overview

The XML payloads transported over XMPP consist of several parts which are not all defined within the present document. The first levels of child elements within the XMPP stanzas are in fact defined in IEC 62351-4 and are mainly related to the management of the end-to-end security. The present clause describes only the applicative part of the payload which is formally defined by the XML schema presented in annex G.3.

The MMSpdu complexType is the entry point within the XSD. It proposes a choice of different envelops for wrapping the various ACSI services. The most commonly used envelops are <confirmed-RequestPDU/> and <confirmed-ResponsePDU/> for mapping ACSI confirmed services and <unconfirmed-PDU/> for mapping ACSI unconfirmed services. Within the envelops for confirmed ACSI services, the complexType ConfirmedServiceRequest (and also ConfirmedServiceResponse) proposes a choice of different service elements (e.g. <read/>, <write/>, <getNameList/>). The detail of the available envelops and service elements is described in Table D.1. How to map the different ACSI services to these envelops and service elements is described in the detailed mapping of each ACSI service starting at Clause 9.

Note that the end-to-end security mechanism may imply to encrypt the applicative payload. In that case the element defined by IEC 62351-4 for containing the applicative payload, would not be of type MMSpdu. It would contain instead the encrypted form of the equivalent value of type MMSpdu which would be used for mapping the same ACSI service when no encryption is used.

8.2 Objects of IEC 61850

8.2.1 General

For mapping the objects defined in IEC 61850-7-2, the present SCSM uses some concepts already used by IEC 61850-8-1 and inherited from the protocol MMS. Their names appear in the XML payloads but it is not necessary to understand these concepts in details. In general it will be enough to know how the ACSI class are mapped on these concepts (see Table 2).

Table 2 – Mapping of ACSI classes on MMS concepts

ACSI Classes	MMS Concepts
Server	VMD
Logical Device	Domain
Logical Node	NamedVariable
Data Object	NamedVariable
All Control Block classes	NamedVariable
DataSet	NamedVariableList
Log	Journal
Files	Files

8.2.2 Logical Node (LN)

The construction of the content of a Logical Node is defined in IEC 61850-7-2, IEC 61850-7-3 and IEC 61850-7-4.

This SCSM maps each Logical Node instance to a Named Variable (concept inherited from MMS), whose name is the name of the Logical Node instance and which have a hierarchical complex structure consisting in multiples levels of components.

The type description of this structure can be created by the algorithm in Figure 4.

```

For each functional constraint (c) found in Figure 53

    For each DataObject (d) in the LN

        If the FCD generated from d,c is not empty, then add a component of ComponentName (d) of the
        ComponentType determined by the FCD to structure (s).

    End For

If the structure (s) is not empty then add a component of ComponentName (c) of the
ComponentType (s) to the TypeDescription being generated.

End For

```

Figure 4 – Algorithm for logical node mapping

The order of the resulted components is recommended to be as shown in Figure 5.

³ The recommended ordering provides compatibility with existing UCA 2.0 implementations

MX
ST
CO
CF
DC
SP
SG
RP
LG
BR
GO
GS (deprecated)
SV
SE
MS
US (deprecated)
EX
SR
OR
BL
RG
RS

Figure 5 – Ordered list of functional constraints

NOTE 1 FCs are defined as follows: RP – unbuffered report control blocks; LG – log control blocks; BR – buffered report control blocks; GO – GOOSE control blocks; GS – deprecated GSSE control blocks; SV – substituted values; SE- Setting group editing; MS – Multicast Sampled Values control block (from IEC 61850-9-2); US – deprecated Unicast Sampled Values control block; EX – name space for model extension; SR – service tracking; CO – Control service parameters; OR – oper received; BL for blocking; RG for Routable GOOSE control blocks; and RS for Routable Multicast Sampled Values control blocks.

NOTE 2 The FC “XX” defined in IEC 61850-7-2 as the wildcard FC for services only is not mapped to the MMS SC5M. Therefore any request that uses FC=“XX” will be acknowledged negatively.

If there is no DataObject (d) that instantiates a specific functional constraint, that functional constraint shall not appear as part of the NamedVariable TypeDescription.

Within a component representing a functional constraint, the order of the sub-components is determined by the order of the names of the instantiated compatible data classes as SCL DOs (see IEC 61850-6) whose data attributes have the specific functional constraint.

Within a component representing IEC 61850-7-4 DataObjects, the order of the sub-components is determined by the order of the names of the attributes of the common data classes as defined in IEC 61850-7-3 or IEC extensions to it for IEDs claiming conformance to those standards, and the SCL shall reflect the same order.

Figure 6 is the graphical illustration of a hierarchical type description of a logical node instance, as produced by the algorithm of Figure 4.

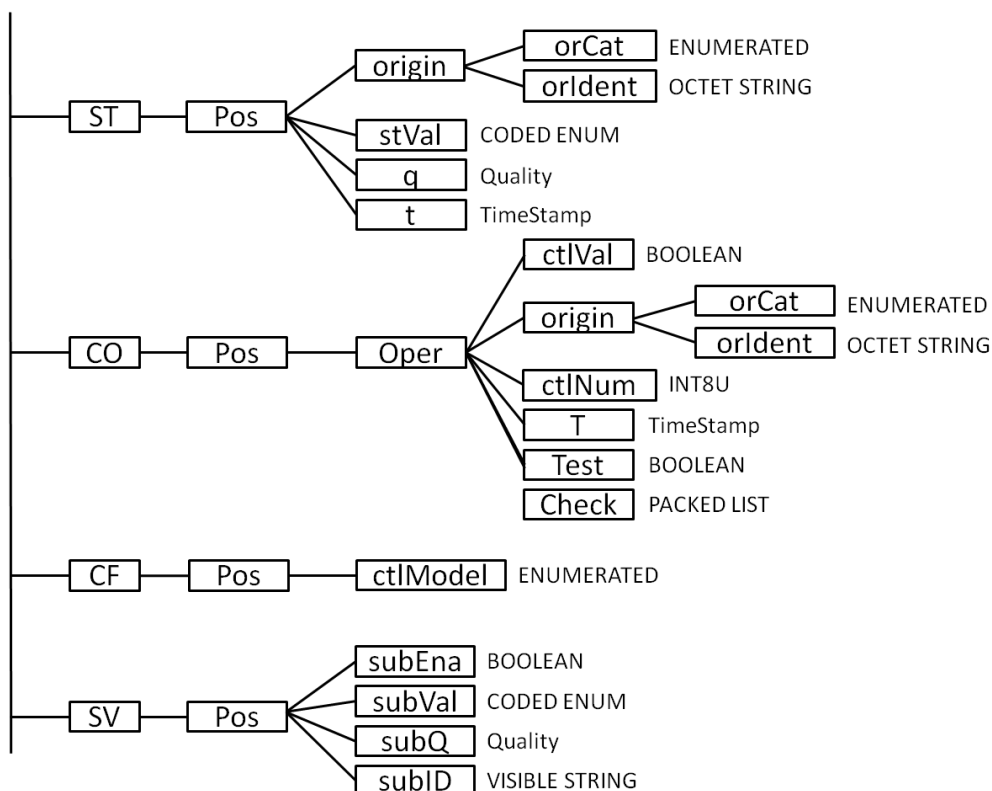


Figure 6 – Example of Logical Node type description

Additionally to this mapping to one Named Variable with a hierarchical structure, a Logical Node can also be seen as a collection of separated Named Variables, one for each component (node or leaf) in the hierarchical structure. The name of the Named Variable corresponding to a given component of the tree is created through the concatenation of the component names present in the path to the given component, plus the name of the given component itself, with all names separated by “\$”. The nesting of the concatenation shall be constrained by the maximum identifier size: 64 characters. Figure 7 presents the list of the flattened Named Variables derived from the LN structure of Figure 6.

```

CSWI1
CSWI1$ST
CSWI1$ST$Pos
CSWI1$ST$Pos$origin$orCat
CSWI1$ST$Pos$origin$orIdent
CSWI1$ST$Pos$stVal
CSWI1$ST$Pos$q
CSWI1$ST$Pos$t
CSWI1$CO
CSWI1$CO$Pos
CSWI1$CO$Pos$Oper$ctlVal
CSWI1$CO$Pos$Oper$origin$orCat
CSWI1$CO$Pos$Oper$origin$orIdent
CSWI1$CO$Pos$Oper$ctlNum
CSWI1$CO$Pos$Oper$T
CSWI1$CO$Pos$Oper$Test
CSWI1$CO$Pos$Oper$Check
CSWI1$CF
CSWI1$CF$Pos
CSWI1$CF$Pos$ctlModel
CSWI1$SV
CSWI1$SV$Pos
CSWI1$SV$Pos$subEna
CSWI1$SV$Pos$subVal
CSWI1$SV$Pos$subQ
CSWI1$SV$Pos$subID

```

Figure 7 – List of the flattened Named Variables corresponding to an LN

The type description of a flattened Named Variable corresponding to a given component is represented by the sub-tree starting at this component within of the overall LN structure.

8.2.3 Mapping of references of Logical Nodes to VariableAccessSpecifications

ACSI services parameters representing references to logical nodes are mapped within the XML payloads on XML elements of type VariableAccessSpecification. The VariableSpecification of such a mapping shall include an ObjectName type. If a LogicalDevice name is present within the LNReference, the child element within the ObjectName shall be domain-Specific. If an “@” is present, the child element shall be aa-specific. If neither is present, the child element shall be vmd-specific.

Within a domain-specific element, the domainId shall be the name of the LogicalDevice and the ItemId shall be the name of the LogicalNode. Figure 8 is a mapping example for the reference LDName/LNName = CONTROL/LLN0.

```

<variableAccessSpecification>
  <listOfVariable>
    <SEQUENCE>
      <variableSpecification>
        <name>
          <domain-specific>
            <domainId> CONTROL </domainId>
            <itemId> LLN0 </itemId>
          </domain-specific>
        </name>
      </variableSpecification>
    </SEQUENCE>
  </listOfVariable>
</variableAccessSpecification>

```

Figure 8 – XML mapping of a LNReference with direct access

An alternate specification shall be supported when the LNReference is constrained to one specific FC (in the case of GetAllDataValues see 12.3.1). Figure 9 is an example of VariableAccessSpecification where an alternateAccess element is used for encoding the functional constraint ST.

```

<variableAccessSpecification>
  <listOfVariable>
    <SEQUENCE>
      <variableSpecification>
        <name>
          <domain-specific>
            <domainId> CONTROL </domainId>
            <itemId> GGIO1 </itemId>
          </domain-specific>
        </name>
      </variableSpecification>
      <alternateAccess>
        <CHOICE>
          <unnamed>
            <selectAlternateAccess>
              <accessSelection>
                <component> ST </component>
              </accessSelection>
            </selectAlternateAccess>
          </unnamed>
        </CHOICE>
      </alternateAccess>
    </SEQUENCE>
  </listOfVariable>
</variableAccessSpecification>

```

Figure 9 – XML mapping of a LNReference with alternate access

8.2.4 Mapping of DataObjects reference to VariableAccessSpecifications

Logical Nodes, according to IEC 61850-7-2, consist of one or more DataObjects. The DataObjects themselves can be constructed with SubDataObjects which can themselves contain SubDataObjects. Flattened named variables can be associated to DataObjects and SubDataObjects whose name is based upon the hierarchically named component of the data found within the Logical Node. Each level of hierarchy will be delimited through the use of a "\$".

EXAMPLES: <LNVariableName>\$<FC>\$<LNDataObjectName1> (e.g. XCBR1\$ST\$Pos)

<LNVariableName>\$<FC>\$<LNDataObjectName1>\$<SubDataObjectName1> (e.g. MMXU1\$MX\$phsA)

The preferred way is to map such references as a VariableSpecification within a VariableAccessSpecification, and to encode the flattened named variable within the itemId element (see example in Figure 10).

```

<variableAccessSpecification>
  <listOfVariable>
    <SEQUENCE>
      <variableSpecification>
        <name>
          <domain-specific>
            <domainId> CONTROL </domainId>
            <itemId> XCBR1$ST$Pos </itemId>
          </domain-specific>
        </name>
      </variableSpecification>
    </SEQUENCE>
  </listOfVariable>
</variableAccessSpecification>

```

Figure 10 – Direct XML mapping of a FCD

Any reference of a SubDataObject that contains an array shall use an AlternateAccessSpecification in the variableAccessSpecifications as soon as one of the array element is addressed (see 8.2.6).

8.2.5 Mapping of DataAttributes (DataAttr) reference to VariableAccessSpecifications

The data attributes DataAttr of the DataObjects map in a similar manner to the DataObjects. However, the attribute name is also included within the hierarchy (see examples below and Figure 11).

EXAMPLES: <LNVariableName>\${FC}\${LNDDataName1}\${AttributeNames1}> (e.g. XCBR1\$ST\$Pos\$stVal)

<LNVariableName>\${FC}\${LNDDataName1}\${AttributeNames1}\${subDataAttributeName1}>

(e.g. XBCR1\$ST\$Pos\$origin\$orCat)

```

<variableAccessSpecification>
  <listOfVariable>
    <SEQUENCE>
      <variableSpecification>
        <name>
          <domain-specific>
            <domainId> CONTROL </domainId>
            <itemId> XCBR1$ST$Pos$stVal </itemId>
          </domain-specific>
        </name>
      </variableSpecification>
    </SEQUENCE>
  </listOfVariable>
</variableAccessSpecification>

```

Figure 11 – Direct XML mapping of a FCDA

Any reference of a DataAttribute, SubDataAttribute that contains an array shall use an AlternateAccessSpecification in the variableAccessSpecifications as soon as one of the array element is addressed (see 8.2.6).

8.2.6 Usage of alternate access for DataObjects and DataAttributes references

A reference shall use an alternate access as soon as an array element is addressed. However the alternate access may be used in other cases. The alternate access consists in adding an

element `<alternateAccess/>` after the element `<variableSpecification/>`. See also clauses regarding AlternateAccess conformance in 24.2.2.2.1.

The character '\$' is not allowed within values of `<component/>` elements in the alternate access part.

It is recommended to always use the choice `<unnamed/>` within the alternate access part.

The examples below illustrate the usage of such an alternate access in the case respectively of:

Figure 12: a nested component without any array.

NOTE This reference might have been encoded without alternate access with `itemId= LLN0$ST$Beh$stVal`

```

<variableSpecification>
  <name>
    <domain-specific>
      <domainId>LDevice</domainId>
      <itemId>LLN0</itemId>
    </domain-specific>
  </name>
</variableSpecification>
<alternateAccess>
  <CHOICE>
    <unnamed>
      <selectAlternateAccess>
        <accessSelection>
          <component>ST</component>
        </accessSelection>
        <alternateAccess>
          <CHOICE>
            <unnamed>
              <selectAlternateAccess>
                <accessSelection>
                  <component>Beh</component>
                </accessSelection>
                <alternateAccess>
                  <CHOICE>
                    <unnamed>
                      <selectAlternateAccess>
                        <accessSelection>
                          <component>stVal</component>
                        </accessSelection>
                        <alternateAccess/>
                      </selectAlternateAccess>
                    </unnamed>
                  </CHOICE>
                </alternateAccess>
              </selectAlternateAccess>
            </unnamed>
          </CHOICE>
        </selectAlternateAccess>
      </unnamed>
    </CHOICE>
  </alternateAccess>
</selectAlternateAccess>
</unnamed>
</CHOICE>
</alternateAccess>
</selectAlternateAccess>
</unnamed>
</CHOICE>
</alternateAccess>

```

Figure 12 – Alternate access without array element

Figure 13: a nested component within an array element. Note that the component path before the index may have been encoded as a flattened variable within the <variableSpecification/> element (see Figure 14).

```

<variableSpecification>
  <name>
    <domain-specific>
      <domainId>LDevice</domainId>
      <itemId>MHAI1</itemId>
    </domain-specific>
  </name>
</variableSpecification>
<alternateAccess>
  <CHOICE>
    <unnamed>
      <selectAlternateAccess>
        <accessSelection>
          <component>MX</component>
        </accessSelection>
        <alternateAccess>
          <CHOICE>
            <unnamed>
              <selectAlternateAccess>
                <accessSelection>
                  <component>HA</component>
                </accessSelection>
                <alternateAccess>
                  <CHOICE>
                    <unnamed>
                      <selectAlternateAccess>
                        <accessSelection>
                          <component>phsAHar</component>
                        </accessSelection>
                        <alternateAccess>
                          <CHOICE>
                            <unnamed>
                              <selectAlternateAccess>
                                <accessSelection>
                                  <index>7</index>
                                </accessSelection>
                                <alternateAccess>
                                  <CHOICE>
                                    <unnamed>
                                      <selectAlternateAccess>
                                        <accessSelection>
                                          <component>cVal</component>
                                        </accessSelection>
                                        <alternateAccess>
                                          <CHOICE>
                                            <unnamed>
                                              <selectAlternateAccess>
                                                <accessSelection>
                                                  <component>mag</component>
                                                </accessSelection>
                                                <alternateAccess>
                                                  <CHOICE>
                                                    <unnamed>
                                                      <selectAlternateAccess>
                                                        <accessSelection>
                                                          <component>f</component>
                                                        </accessSelection>
                                                        <alternateAccess>

```

```

</selectAlternateAccess>
</unnamed>
</CHOICE>
</alternateAccess>
</selectAlternateAccess>
</unnamed>
</CHOICE>
</alternateAccess>
</selectAlternateAccess>
</unnamed>
</CHOICE>
</alternateAccess>
</selectAlternateAccess>
</unnamed>
</CHOICE>
</alternateAccess>
</selectAlternateAccess>
</unnamed>
</CHOICE>
</alternateAccess>
</selectAlternateAccess>
</unnamed>
</CHOICE>
</alternateAccess>
</selectAlternateAccess>
</unnamed>
</CHOICE>
</alternateAccess>

```

Figure 13 – Alternate access with array element

```

<variableSpecification>
  <name>
    <domain-specific>
      <domainId>LDevice</domainId>
      <itemId>MHAI1$MX$HA$phsAHar</itemId>
    </domain-specific>
  </name>
</variableSpecification>
<alternateAccess>
  <CHOICE>
    <unnamed>
      <selectAlternateAccess>
        <accessSelection>
          <index>7</index>
        </accessSelection>
        <alternateAccess>
          <CHOICE>
            <unnamed>
              <selectAlternateAccess>
                <accessSelection>
                  <component>cVal</component>
                </accessSelection>
                <alternateAccess>
                  <CHOICE>
                    <unnamed>
                      <selectAlternateAccess>
                        <accessSelection>
                          <component>mag</component>
                        </accessSelection>
                        <alternateAccess>
                          <CHOICE>
                            <unnamed>
                              <selectAlternateAccess>
                                <accessSelection>
                                  <component>f</component>
                                </accessSelection>
                                <alternateAccess/>
                              </selectAlternateAccess>
                            </unnamed>
                          </CHOICE>
                        </alternateAccess>
                      </selectAlternateAccess>
                    </unnamed>
                  </CHOICE>
                </alternateAccess>
              </selectAlternateAccess>
            </unnamed>
          </CHOICE>
        </selectAlternateAccess>
      </unnamed>
    </CHOICE>
  </alternateAccess>
</alternateAccess>

```

Figure 14 – Alternate access with flattened variable and array element

8.3 Mapping of IEC 61850-7-2 data attributes

8.3.1 BasicTypes

Within the XMLpayload, the BasicTypes of IEC 61850-7-2 are mapped on the XML elements described in Table 3. These elements are those defined in XSD file for the complexType Data (see Annex G).

Table 3 – Mapping of ACSI BasicTypes

IEC 61850-7-2 BasicType name	XML element name	XML type used for the value	Comment
BOOLEAN	<boolean/>	BOOLEAN	Example: <boolean>true</boolean>
INT8	<integer/>	INTEGER	INTEGER is defined in XSD as a restriction of xsd:integer Example: <integer>-99</integer>
INT16	<integer/>		
INT32	<integer/>		
INT64	<integer/>		
INT8U	<unsigned/>		INTEGER is defined in XSD as a restriction of xsd:integer Example: <unsigned>99</unsigned>
INT16U	<unsigned/>		
INT24U	<unsigned/>		
INT32U	<unsigned/>		
FLOAT32	<floating-point/>	OCTET_STRING	OCTET_STRING representing 4 octets. Range of values and precision as specified by IEEE 754 single precision floating point. NOTE 1 The value of Not-A-Number (NaN) is valid. However, the processing of a value indicating Not-A-Number (NaN) is a local issue. NOTE 2 OCTET_STRING is a restriction of xsd:string for encoding hexadecimal values
ENUMERATED	<integer/>	INTEGER	INTEGER is defined in XSD as a restriction of xsd:integer Used here for an ordered set of values, defined where type is used. See 8.3.2.2. Example: Beh.stVal = Test (3) would be encoded as <integer>3</integer>
CODED ENUM	<bit-string/>	BIT_STRING	BIT_STRING is an xsd:string type, restricted to zero or more “0” and “1” characters (see 8.3.2.1). Used here for an ordered set of values, defined where type is used. See also 8.3.2.3. Example: dbPos is a 2 bit length bit string <bit-string>01</bit-string>
OCTET STRING	<octet-string/>	OCTET_STRING	OCTET_STRING is a restriction of xsd:string for encoding hexadecimal values. Max length shall be defined where type is used. See also 8.3.2.4. Example: DPC or Ident is a 64 octet string. An example with 7 octets would be encoded as <octet-string>53544154494F4E</octet-string>
VISIBLE STRING	<visible-string/>	VisibleString	VisibleString is an xsd:string restricted to visible characters. Max length shall be defined where type is used. See 8.3.2.5. Example: RptID of a ReportControl is a Visible String129 <visible-string>CONTROL/GGIO1\$BR\$simpleBReport05</visible-string>

IEC 61850-7-2 BasicType name	XML element name	XML type used for the value	Comment
UNICODE STRING	<mMSSString/>	MMSSString	Defined in XSD as a mixed content of text and XML elements encoding control characters. Max length shall be defined where type is used. See 8.3.2.6. Example: <mMSSString>ābcdēfgh<nul/>ījkŁμñópq<soh/>r\$tùvwxyz</mMSSString>
For the purposes of the rest of the clauses in this document, the precision of unsigned / integer shall be determined by the IEC 61850-7-2 or IEC 61850-7-3 unless otherwise specified.			

8.3.2 Additional definitions of BasicType

8.3.2.1 General statement for BIT STRING

Within this standard, the reference to a BIT STRING shall indicate the use of an element with name="bit-string" and type="BIT_STRING" as defined by the normative XSD (see Annex G).

BIT_STRING is a restriction of xsd:string type, restricted to zero or more "0" and "1" characters.

NOTE Characters of a BIT_STRING are however referred as bits in the present document.

Bit(0) of the BIT STRING is mapped to the left most character in the XML String.

The amount of bits conveyed by a sent variable length bit-string shall always be equal to the maximum length of the defined variable length bit-string at the sender side, so that forward semantic compatibility can be maintained. Zero (0) length bit-string are not allowed in the context of IEC 61850-8-2.

8.3.2.2 ENUMERATED

The IEC 61850-7-2 ENUMERATED shall map to an <integer/> element with an xsd:integer value. Values that are 0 or greater shall be reserved for values standardized within IEC 61850. Values that are less than 0 shall be considered private values except for the IEC 61850-7-3 Multiplier enumeration where the enumeration value delivers the multiplier value to use. The range of the allowed values is specified in IEC 61850-7-2, IEC 61850-7-3 and IEC 61850-7-4.

The first ordered value, of the standardized enumerated values, shall be assigned the value of zero (0).

Enumerated values outside the standardized range of values shall be assumed to be value extensions and shall not cause a protocol error.

For IEC 61850 enumerated values, where the actual numeric values are specified, the integer values within the context of IEC 61850-8-2 shall be the same as the numeric values specified.

8.3.2.3 CODED ENUM

The IEC 61850-7-2 CODED ENUM is mapped on a <bit-string/> element with a BIT_STRING type value, as defined in the normative XSD (see 8.3.2.1). The size of the string shall be the number of bits required to represent the maximum enumerate value defined in IEC 61850-7-2. It shall represent the unsigned integer encoding of the enumerated values in the order specified in IEC 61850-7-2.

For the purposes of IEC 61850-8-2, CODED ENUMs that appear within a PACKED LIST or an array of other CODED ENUMs are treated as special cases.

8.3.2.4 OCTET STRING

The IEC 61850-7-2 OCTET STRING shall map to an <octet-string/> element with an OCTET_STRING type value. OCTET_STRING is a restriction of xsd:string for encoding hexadecimal values, i.e. based on the character set [0-9][A-F][a-f] but including possibly extra space characters. The maximum size shall be as specified in IEC 61850-7-2 and IEC 61850-7-3 and do not include the extra spaces added for readability. The NULL string has a length 0.

NOTE The term octet is used within the present document for designating the two characters of an OCTET_STRING representing the hexadecimal value of a binary octet.

8.3.2.5 VISIBLE STRING

The IEC 61850-7-2 VISIBLE STRING shall map to a <visible-string/> element with a VisibleString type value. VisibleString is defined in the XSD as a an xsd:string restricted to visible characters, i.e. the range from Space (ascii code 32) to Tilde (ascii code 126). The maximum size shall be as specified in IEC 61850-7-2 and IEC 61850-7-3. The NULL string has a length 0.

The character set shall be constrained to ISO646String.

8.3.2.6 UNICODE STRING

The IEC 61850-7-2 UNICODE STRING shall map to a <mMSString/> element with a mMSString complexType value. mMSString complexType may contain a mix of text (UTF8 encoding) and control characters encoded as explicit XML elements like <nul/> or <soh/>. The maximum size shall be the number of Unicode characters as specified in IEC 61850-7-2 and IEC 61850-7-3. The serialization of a Unicode character occurs as specified in RFC 3629, and may involve up to 4 octets. The NULL string has a length 0.

8.3.2.7 Array

An array shall map to an <array/> XML element with one more child XML elements.

If the elements of the array are of a BasicType, the child XML elements shall be directly those described in Table 3 with one XML element corresponding to each array element.

Otherwise, the child elements shall be <structure/> elements which can recursively include BasicType elements and/or <structure/> elements.

See in XSD file of Annex G for the definition of the elements “array” within the complexType “Data”.

Example: encoding of an array of 2 points

```
<array>
  <structure>
    <floating-point>3f800000</floating-point>
    <floating-point>40000000</floating-point>
    <floating-point>40400000</floating-point>
  </structure>
</structure>
```

```

        <floating-point>40800000</floating-point>
        <floating-point>40a00000</floating-point>
        <floating-point>40c00000</floating-point>
    </structure>
</array>

```

8.3.3 Common ACSITypes

8.3.3.1 ObjectName

This type only appears in the context of ObjectReference. The allowed character set within instances of ObjectName is defined in IEC 61850-7-2.

8.3.3.2 ObjectReference

8.3.3.2.1 General

The IEC 61850-7-2 Object reference mapping depends on the context where the object reference is defined:

- Within the scope of control block attributes: see 8.3.3.2.2.
- Within the scope of data attributes (Tracking CDCs,ORG): see 8.3.3.2.3.
- Within the scope of DataRef of reports: see 8.3.3.2.4.

In all cases, the ObjectReference is mapped as a VISIBLE STRING with a maximum length of 129 octets.

8.3.3.2.2 Object Reference within Control Blocks

The value shall be constructed as:

- AA-specific scope objects: @<Object Name>
- Domain specific: <Domain Name>/<Object Name>
- VMD-specific: /<Object Name>

The maximum Object Name and Domain Name size shall be 64.

The allowed character set is and the same as ObjectName plus the “\$”, “/” and “@” characters.

8.3.3.2.3 Object Reference within CDCs (Tracking, ORG).

The value shall be constructed as specified in IEC 61850- 7-2.

The allowed character set is defined in IEC 61850-7-2. The value shall be constructed as:

- Reference to a Logical Device:
 <LDName>
- Reference to a Logical Node:
 <LDName>/<LNName>
- Reference to a DataSet:
 <LDName>/<LNName>.<DataSetName>
- Reference to a DataObject: