

DIN EN 61850-8-2



ICS 33.040.40

Einsprüche bis 2018-06-20

Entwurf

**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 57/1833/CDV:2017);
Englische Fassung prEN 61850-8-2:2017**

Communication networks and systems for power utility automation –
Part 8-2: Specific communication service mapping (SCSM) –
Mapping to Extensible Messaging Presence Protocol (XMPP)
(IEC 57/1833/CDV:2017);
English version prEN 61850-8-2:2017

Anwendungswarnvermerk

Dieser Norm-Entwurf mit Erscheinungsdatum 2018-04-20 wird der Öffentlichkeit zur Prüfung und Stellungnahme vorgelegt.

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

Stellungnahmen werden erbeten

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

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

Gesamtumfang 253 Seiten

DKE Deutsche Kommission Elektrotechnik Elektronik Informationstechnik in DIN und VDE

Nationales Vorwort

Die Englische Fassung des europäischen Dokuments prEN 61850-8-2:2017 „Kommunikationsnetze und -systeme für die Automatisierung der elektrischen Energieversorgung – Teil 8-2: Spezifische Abbildung von Kommunikationsdiensten (SCSM) – Abbildungen auf erweiterbares Messaging Presence Protocol (XMPP)“ (Entwurf in der Umfrage) ist unverändert in diesen Norm-Entwurf übernommen worden.

Die Internationale Elektrotechnische Kommission (IEC) und das Europäische Komitee für Elektrotechnische Normung (CENELEC) haben vereinbart, dass ein auf IEC-Ebene erarbeiteter Entwurf für eine Internationale Norm zeitgleich (parallel) bei IEC und CENELEC zur Umfrage (CDV-Stadium) und Abstimmung als FDIS (en: Final Draft International Standard) bzw. Schluss-Entwurf für eine Europäische Norm gestellt wird, um eine Beschleunigung und Straffung der Normungsarbeit zu erreichen. Dem entsprechend ist das internationale Dokument IEC 57/1833/CDV:2017 „Communication networks and systems for power utility automation – Part 8-2: Specific communication service mapping (SCSM) – Mapping to Extensible Messaging Presence Protocol (XMPP)“ unverändert in den Entwurf prEN 62325-451-6:2017 übernommen worden.

Das internationale Dokument wurde vom TC 57 „Power systems management and associated information exchange“ der Internationalen Elektrotechnischen Kommission (IEC) erarbeitet.

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

Für dieses Dokument ist das nationale Arbeitsgremium K 952 „Netzleittechnik“ der DKE Deutsche Kommission Elektrotechnik Elektronik Informationstechnik in DIN und VDE (www.dke.de) zuständig.

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

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

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

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

Da sich die Benutzer des vorliegenden Normentwurfs der englischen Sprache als Fachsprache bedienen, wird die Englische Fassung der prEN 61850-8-2 veröffentlicht. Zu deren Abschnitt 1, der den Anwendungsbereich festlegt, und Abschnitt 3, der die Begriffe festlegt, wurde eine Übersetzung angefertigt und als Nationaler Anhang NA des vorliegenden Normentwurfs hinzugefügt. Für einige der verwendeten Begriffe existieren keine gebräuchlichen deutschen Benennungen, da sich die deutschen Anwender in der Regel ebenfalls der englischen Benennungen bedienen. Dieser Normentwurf steht nicht in unmittelbarem Zusammenhang mit Rechtsvorschriften und ist nicht als Sicherheitsnorm anzusehen.

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

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

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

Nationaler Anhang NA (informativ)

Auszugsweise Übersetzung^{N1)}

1 Anwendungsbereich

Dieser Teil von IEC 61850 legt ein Verfahren für den Austausch von Daten über Netzwerke beliebiger Art, einschließlich öffentlicher Netzwerke, fest. Von den verschiedenen in IEC 61850-7-2 festgelegten Arten von Diensten werden zunächst nur die Client/Server-Dienste und die Zeitsynchronisierungsdienste betrachtet.

Bei den Client/Server-Diensten werden die Objekte und Dienste der in IEC 61850-7-2 definierten Schnittstelle für abstrakte Kommunikationsdienste (en: abstract communication service interface; ACSI) auf XML-Nachrichten abgebildet, die über XMPP übertragen werden. Die Beschreibung der Abbildung umfasst im Wesentlichen die folgenden drei Aspekte:

- die Verwendung des Protokolls XMPP, wobei ausführlich beschrieben wird, welche Funktionen bei der Abbildung auf welche Art und Weise zum Einsatz kommen (siehe Abschnitt 0)
- eine Beschreibung der Bedingungen sicherer Ende-zu-Ende-Kommunikation (siehe Abschnitt 0)
- eine Beschreibung der dem jeweiligen ACSI-Dienst entsprechenden XML-Nutzdaten, und zwar unter Verwendung des XML-Schemas und von Beispielen für XML-Nachrichten (ab Abschnitt 0)

ANMERKUNG 1 Eine ausführliche Beschreibung der Verwendung des Protokolls XMPP wird in diesem Dokument nicht vorgenommen.

ANMERKUNG 2 Dienste für das Systemmanagement werden in diesem Dokument nicht behandelt.

ANMERKUNG 3 Im Hinblick auf Anwender, die mit der in IEC 61850-8-1 definierten Abbildung vertraut sind, sei angemerkt, dass die im vorliegenden Dokument definierten XML-Nachrichten von den in IEC 61850-8-1 definierten Nachrichten abgeleitet sind, wobei anstelle einer binären Kodierung hier allerdings eine XML-Kodierung verwendet wird. Die Implementierung von Gateways zwischen IEC 61850-8-1 und IEC 61850-8-2 ist somit in beiden Richtungen unkompliziert. Für das Verständnis des vorliegenden Dokuments ist es allerdings nicht erforderlich, auch IEC 61850-8-1 zu lesen.

3 Begriffe

Für die Anwendung dieses Dokuments gelten die Begriffe nach IEC 61850-2 und die folgenden Begriffe.

3.1

(n)-Schicht

jedwede bestimmte Schicht

[ISO/IEC 7498-1, Definition 3.1]

3.2

(n)-Protokoll-Dateneinheit

in einem (n)-Protokoll festgelegte und aus (n)-Protokoll-Steuerinformationen und gegebenenfalls (n)-Benutzerdaten bestehende Dateneinheit

[ISO/IEC 7498-1, Definition 5.6.1.3]

^{N1)} Nationale Fußnote: Die Nummerierung aus IEC 57/1833/CDV:2017-03 wurde beibehalten.

3.3**(n)-Protokoll**

Menge von Regeln und Formaten (Semantik und Syntax), die das Kommunikationsverhalten von (N)-Instanzen bei der Ausführung von (n)-Funktionen bestimmen

[ISO/IEC 7498-1, Definition 5.2.1.9]

3.4**Klasse**

Beschreibung einer Menge von Objekten, welche die gleichen Attribute, Dienste, Beziehungen und Semantik aufweisen

[IEC 61850-7-2, Definition 3.1]

3.5**Client**

Einheit, die bei einem Server einen Dienst anfordert und unaufgeforderte Nachrichten von einem Server empfängt

[IEC 61850-7-2, Definition 3.2]

3.6**Gerät**

Einheit, die Steuer-, Stell- und/oder Sensorfunktionen ausführt und über Schnittstellen zu anderen derartigen Einheiten in einem Automatisierungssystem verfügt

[IEC 61850-7-2, Definition 3.3]

ANMERKUNG Geräte allein führen keine Funktionen zum Energietransport aus.

3.7**logisches Gerät**

Einheit, die eine Menge von typischen Stationsfunktionen repräsentiert

[IEC 61850-7-2, Definition 3.6]

3.8**logischer Knoten**

Einheit, die eine typische Stationsfunktion repräsentiert

[IEC 61850-7-2, Definition 3.7]

3.9**physisches Gerät**

Einheit, welche die physischen Teile eines Geräts repräsentiert (Hardware und Betriebssystem usw.)

[IEC 61850-7-2, Definition 3.8]

ANMERKUNG Physische Geräte fungieren als Host für logische Geräte.

3.10**Anwendungs- und Transportprofile****A-Profil und T-Profil**

Menge von Protokollen für einen bestimmten Zweck

3.11**Datenattribut**

benanntes Datenobjektelement eines bestimmten Typs

3.12

Datenattribut

bedeutungstragende, strukturierte Informationen von Anwendungen in einem Automatisierungsgerät, die gelesen oder geschrieben werden können

ANMERKUNG Diese Definition stimmt mit der Definition von DataObject in IEC 61850-7-2 überein. Die MMS-Verwendung von DataObject bezieht sich auf Datentypen und potenzielle Instanziierungen von MMS-Variablen.

3.13

Server

Einheit, die Clients Dienste bereitstellt oder unaufgeforderte Nachrichten ausgibt

3.14

privates/öffentliches Netzwerk

Ein Netzwerk gilt als privates Netzwerk, wenn die das Netzwerk nutzende eindeutige Einheit als Master den gesamten Datenfluss leitet und die Leistung aus Sicht der das Netzwerk nutzenden Einheit hinsichtlich Bandbreite, Durchsatz, Laufzeit, Verfügbarkeit usw. sichergestellt ist.

Andernfalls handelt es sich um ein öffentliches Netzwerk.

ANMERKUNG Ein privates Netzwerk kann auf einer öffentlichen oder gemeinsam genutzten Infrastruktur (en: shared infrastructure) aufbauen, wenn das Niveau der Dienste sichergestellt werden kann.

3.15

Smart Grid

intelligentes Elektrizitätsversorgungssystem

Anwendung von künstlicher Intelligenz und von Netzwerkfähigkeiten auf ein Stromversorgungssystem, so dass der Betrieb der miteinander verbundenen Elemente von der zentralen und dezentralen Energieerzeugung über das Hochspannungsnetz und -verteilungssystem bis zu industriellen Anwendern und Gebäudeautomationssystemen, Energiespeicheranlagen und Temperaturreglern, elektrischen Fahrzeugen und Einrichtungen sowie sonstigen Haushaltsgeräten beim Endverbraucher überwacht, geschützt und automatisch optimiert werden kann

– Leerseite –

CONTENTS

FOREWORD.....	14
INTRODUCTION.....	17
1 Scope.....	18
2 Normative references	18
3 Terms and definitions	21
4 Abbreviations	22
5 Overview	23
5.1 General.....	23
5.2 Mapping of client/server services.....	24
5.2.1 General	24
5.2.2 XML payloads	26
5.2.3 XMPP	27
5.3 Time sync services.....	27
6 Usage of XMPP	28
6.1 Principles.....	28
6.2 Connection establishment	28
6.2.1 General	28
6.2.2 Usage of TLS and SASL.....	28
6.2.3 Stream Compression.....	29
6.3 Mapping of solicited services.....	29
6.4 Mapping of unsolicited services	30
6.5 Usage of XMPP presence.....	30
6.6 Usage of the Roster	30
6.7 XMPP extensions	30
6.7.1 Usage of XMPP PING – XEP 0199	30
6.7.2 Usage of Stream Management – XEP 0198	31
6.8 Implementation agreements	31
6.8.1 XMPP PING – XEP 0199.....	31
7 Association context management.....	31
8 End-to-end security	32
9 Payload description	32
9.1 XSD overview	32
9.2 Objects of IEC 61850	33
9.2.1 General	33
9.2.2 Logical Node (LN)	33
9.2.3 Mapping of references of Logical Nodes to VariableAccessSpecifications	36
9.2.4 Mapping of DataObjects reference to VariableAccessSpecifications	37
9.2.5 Mapping of DataAttributes (DataAttr) reference to VariableAccessSpecifications	38
9.2.6 Usage of alternate access for DataObjects and DataAttributes references	38
9.3 Mapping of IEC 61850-7-2 data attributes	42
9.3.1 BasicTypes	42
9.3.2 Additional definitions of BasicType	43
9.3.2.1 General	

9.3.2.2	ENUMERATED	43
9.3.2.3	CODED ENUM	43
9.3.2.4	OCTET STRING	43
9.3.2.5	VISIBLE STRING	43
9.3.2.6	UNICODE STRING	44
9.3.2.7	Array	44
9.3.3	Common ACSITypes	44
9.3.3.1	ObjectName	44
9.3.3.2	ObjectReference	44
9.3.3.3	PhyComAddr	46
9.3.3.4	ServiceError	47
9.3.3.5	EntryID	52
9.3.3.6	PACKED LIST	52
9.3.3.7	Timestamp	52
9.3.3.8	EntryTime	53
9.3.3.9	TriggerConditions	53
9.3.3.10	Reason for inclusion in reports (ReasonForInclusionInReport)	54
9.3.3.11	Reason for inclusion in logs (ReasonForInclusionInLog)	54
9.3.3.12	RCBReportOptions	54
9.3.3.13	LCBLogEntryOptions	55
9.3.3.14	SVMessageOptions	55
9.3.3.15	CheckConditions	55
9.3.3.16	Originator	55
9.3.3.17	Currency	55
9.3.4	Mapping of quality common data attribute type specified in IEC 61850-7-2	56
9.4	General mapping of data values within XML payloads	56
9.5	Extended behavior for optimization of bandwidth	57
10	Server class model	57
10.1	General	57
10.2	GetServerDirectory	57
10.2.1	General	57
11	Association model	60
11.1	Association relation to communication profiles	60
11.2	Two party association model for client/server communication profile	61
11.2.1	Establishment of a secured end-to-end association	61
11.2.2	Association services	61
11.2.2.1	Associate	61
11.2.2.2	Abort	64
11.2.2.3	Release	64
12	Logical device model	65
12.1	General	65
12.2	Response-	67
12.3	Extended behavior	67
13	Logical Node model	68
13.1	General	68
13.2	GetLogicalNodeDirectory	68
13.2.1	General	68

13.2.2	Response-	70
13.2.3	Extended behavior	70
13.3	GetAllDataValues	72
13.3.1	General	72
14	DataObject, DataAttribute, SubDataAttribute model	75
14.1	General	75
14.2	GetDataValues	75
14.3	SetDataValues	76
14.4	GetDataDirectory	77
14.4.1	General	77
14.4.2	Response-	79
14.4.3	Extended behavior	79
	14.4.3.1 General	79
	14.4.3.2 Response-	81
14.5	GetDataDefinition	81
15	Data set class model	81
15.1	General	81
15.2	GetDataSetValues	82
15.2.1	General	82
15.2.2	Response-	83
15.3	DataSetValues	84
15.4	CreateDataSet	84
15.4.1	General	84
15.4.2	Response-	85
15.5	DeleteDataSet	86
15.5.1	General	86
15.5.2	Errors	87
15.6	DataSetDirectory	88
15.6.1	General	88
15.6.2	Response-	89
16	ServiceTracking model	90
17	Setting group control class model	93
17.1	Setting group control block definition	93
17.2	Setting group control class services	93
17.2.1	SelectActiveSG	93
17.2.2	SelectEditSG	94
17.2.3	SetEditSGValue	95
17.2.4	ConfirmEditSGValues	96
17.2.5	GetEditSGValue	96
17.2.6	GetSGCBValues	97
18	Reporting and logging class model	98
18.1	Report model – Report control blocks	98
18.1.1	Buffered report control block	98
18.1.2	Unbuffered report control block	99
18.2	Reporting services	100
18.2.1	Report service	100
18.2.2	GetBRCBValues	103
18.2.3	SetBRCBValues	105

18.2.4	GetURCBValues	107
18.2.5	SetURCBValues.....	107
18.3	Log model.....	108
18.3.1	General	108
18.3.1.1	Description of LCB attributes.....	109
18.3.2	Mapping of log and log control services	110
18.3.2.1	GetLCBValues	110
18.3.2.2	SetLCBValues	110
18.3.2.3	GetLogStatusValues	110
18.3.2.4	QueryLogByTime	110
18.3.2.5	QueryLogAfter	114
18.3.3	Conformance	115
19	Mapping of the generic substation event model (GSE).....	115
19.1	Generic object oriented substation event (GOOSE).....	115
19.1.1	GOOSE control definition	115
19.1.2	Specialization for Layer 2 GoCB	116
19.1.3	Specialization for Routable GOOSE.....	116
19.1.4	GOOSE services.....	116
19.1.4.1	General	116
19.1.4.2	GetGoCBValues	117
19.1.4.3	SetGoCBValues.....	117
20	Transmission of sampled values class model	117
20.1	Sampled value	117
20.1.1	Sampled value control block.....	117
20.1.1.1	General	117
20.1.1.2	Specialization for Layer 2 Sampled value	117
20.1.1.3	Specialization for Routable Sampled value	117
20.1.1.4	Specialization for Unicast Sampled value	117
20.1.2	Sampled value services.....	117
20.1.2.1	General	117
20.1.2.2	GetMSVCBValues.....	118
20.1.2.3	SetMSVCBValues	118
21	Control class model	118
21.1	General.....	118
21.2	Overview of control services mapping.....	120
21.3	Select	121
21.4	SelectWithValue.....	122
21.4.1	SelectWithValue service parameter mapping	122
21.4.2	General mapping of the SelectWithValue service	122
21.4.3	SelectWithValue response–.....	124
21.5	Cancel	125
21.5.1	Cancel service parameter mapping.....	125
21.5.2	General mapping of the Cancel service	125
21.5.3	Cancel response–	126
21.6	Operate	127
21.6.1	Operate service parameter mapping	127
21.6.2	General mapping of the Operate service.....	127
21.6.3	Operate response–	128
21.7	CommandTermination	129

21.7.1	CommandTermination service parameter mapping	129
21.7.2	General mapping of the CommandTermination service	129
21.8	TimeActivatedOperate	132
21.8.1	TimeActivatedOperate service parameter mapping	132
21.8.2	Mapping of the TimeActivatedOperate service	132
21.9	TimeActivatedOperateTermination service	132
21.10	AdditionalCauseDiagnosis in negative control service responses	132
21.11	Tracking of control services	135
21.11.1	General	135
21.11.2	Mapping of the Control service tracking (CTS)	135
22	Time and time synchronization model	136
23	Naming conventions	136
24	File transfer	136
24.1	File transfer model	136
24.2	File services	138
24.2.1	GetFile	138
24.2.1.1	General	138
24.2.1.2	Response-	141
24.2.2	SetFile	143
24.2.2.1	General	143
24.2.2.2	Response-	144
24.2.3	DeleteFile	145
24.2.3.1	General	145
24.2.3.2	Response-	146
24.2.4	GetFileAttributeValues	146
25	Conformance	148
25.1	Notation	148
25.2	PICS	148
25.2.1	Profile conformance	148
25.2.2	XML Payload conformance	149
25.2.2.1	Environmental services	149
25.2.2.2	Variable access conformance	156
25.2.2.3	Journal management services	166
25.2.2.4	File management services	167
25.3	PICS Statement	168
25.3.1	General	168
25.3.2	Substation configuration language	168
26	Substation Configuration Language (SCL)	169
Annex A (informative)	Communication stack	170
A.1	Overview	170
A.1.1	General	170
A.1.2	XMPP communication profiles	171
A.1.3	Non-XMPP communication profiles	171
A.2	Communication stack	171
A.2.1	Overview of the protocol usage	171
A.2.2	Client/server services and communication profiles	172
A.2.2.1	Client/server services	172
A.2.2.2	A-Profile	172

A.2.2.3	XMPP T-Profile	173
A.2.3	Time sync	173
A.2.3.1	Conformance statement	173
A.2.3.2	A-Profile	173
A.2.3.3	T-Profile	174
Annex B (informative)	Deployment of XMPP infrastructure	175
B.1	Deployment of XMPP within one XMPP domain	175
B.1.1	Use case facility	175
B.1.2	Use case hierarchy within a facility	175
B.2	Deployment of XMPP while interconnecting more than one XMPP domain	177
B.2.1	Interconnection of XMPP Domain	177
B.2.1.1	Use case Facility at DSO	177
B.2.1.2	Use case Multiple facilities at DSO	178
B.2.1.3	Use case VPP communications	179
B.2.1.4	Use case VPP communication to DERs@VPP	181
B.2.1.5	Use case VPP communication to DERs@DSO	182
B.2.1.6	Use case DSO communication to DERs@VPP	182
B.2.1.7	Use case DSO communication to DERs@DSO	183
B.2.1.8	Multiple JIDs	183
B.2.2	Definition of a federation communication between XMPP domains	183
B.2.3	Interconnection of Domain with federation	185
B.2.3.1	Use case Facility at DSO	185
B.2.3.2	Use case VPP communications	185
B.2.3.3	Use case VPP communication to DER@VPPs	186
B.2.3.4	Use case VPP communication to DERs@DSO	187
B.2.3.5	Use case DSO communication to DERs@VPP	187
B.3	Communication path outage and recovery	188
Annex C (informative)	Security for DER integration based on XMPP	189
C.1	General	189
C.2	Assumptions and Boundary Conditions	190
C.3	Derivation of Security Requirements	190
C.4	Mapping of security options to XMPP based integration of DER	191
C.4.1	General	191
C.4.2	Mapping of end-to-end security concept for SCSM 8-2	192
C.5	Sequence diagrams	193
C.5.1	General	193
C.5.2	XMPP and Stream opening	193
C.5.3	Stream establishment, ROSTER and Presence	194
C.5.4	Associate	195
C.5.5	Communication outage	199
C.5.6	Request Response (Clear Transfer)	201
C.5.7	Report Message (Encrypted Transfer)	203
C.5.8	Release (Clear Transfer)	204
Annex D (normative)	Mapping of services and errors over XMPP stanzas	206
Annex E (informative)	Implementation of gateways with IEC 61850-8-1 SCSM	209
Default values		209
Annex F (informative)	SCL conformance	211
Annex G (normative)	XML schema definitions for the XML payload	212

Figure 1 – Overview of functionality and profiles	24
Figure 2 – Example of XML Payload	27
Figure 3 – Generic structure of client/server ACSI services	29
Figure 4 – Example of <iq> stanza request	29
Figure 5 – Example of <iq> stanza response	29
Figure 6 – Example of <message> stanza	30
Figure 7 – Example of AssociationContext element	32
Figure 8 – Algorithm for logical node mapping	34
Figure 9 – Ordered list of functional constraints	34
Figure 10 – Example of Logical Node type description	35
Figure 11 – List of the flattened Named Variables corresponding to an LN	36
Figure 12 – XML mapping of a LNReference with direct access	37
Figure 13 – XML mapping of a LNReference with alternate access	37
Figure 14 – Direct XML mapping of a FCD	38
Figure 15 – Direct XML mapping of a FCDA	38
Figure 16 – Alternate access without array element	39
Figure 17 – Alternate access with array element	41
Figure 18 – Alternate access with flattened variable and array element	41
Figure 19 – Example of Association context error	47
Figure 20 – XML structure of GetServerDirectory-Request (LD)	58
Figure 21 – XML structure of GetServerDirectory-Response (LD)	59
Figure 22 – XML structure of GetServerDirectory-Request (FILE)	60
Figure 23 – XML structure of GetServerDirectory-Response (FILE)	60
Figure 24 – XML structure of Associate-Request	63
Figure 25 – XML structure of Associate-Response	63
Figure 26 – XML structure of GetLogicalDeviceDirectory-Request	66
Figure 27 – XML structure of GetLogicalDeviceDirectory-Response	66
Figure 28 – XML structure of extended GetLogicalDeviceDirectory-Request	68
Figure 29 – XML structure of extended GetLogicalDeviceDirectory-Response	68
Figure 30 – XML structure of GetLogicalNodeDirectory-Request	70
Figure 31 – XML structure of GetLogicalNodeDirectory-Response	70
Figure 32 – XML structure of extended GetLogicalNodeDirectory-Request (step 1)	71
Figure 33 – XML structure of extended GetLogicalNodeDirectory-Response (step 1)	72
Figure 34 – XML structure of extended GetLogicalNodeDirectory-Request (step 2)	72
Figure 35 – XML structure of extended GetLogicalNodeDirectory-Response (step 2)	72
Figure 36 – XML structure of GetAllDataValues-Request	74
Figure 37 – XML structure of GetAllDataValues-Response	75
Figure 38 – XML structure of GetDataValues-Request	76
Figure 39 – XML structure of GetDataValues-Response	76
Figure 40 – XML structure of SetDataValues-Request	77
Figure 41 – XML structure of SetDataValues-Response	77
Figure 42 – XML structure of GetDataDirectory-Request	78

Figure 43 – XML structure of GetDataDirectory-Response	79
Figure 44 – XML structure of extended GetDataDirectory-Request	81
Figure 45 – XML structure of extended GetDataDirectory-Response	81
Figure 46 – Mapping of reference to persistent data set within logical device	82
Figure 47 – Mapping of reference to persistent data set outside logical device	82
Figure 48 – Mapping of reference to non-persistent data set	82
Figure 49 – XML structure of GetDataSetValues-Request	83
Figure 50 – XML structure of GetDataSetValues-Response	83
Figure 51 – XML structure of CreateDataSet-Request	85
Figure 52 – XML structure of CreateDataSet-Response	85
Figure 53 – XML structure of DeleteDataSet-Request	87
Figure 54 – XML structure of DeleteDataSet-Response	87
Figure 55 – XML structure of GetDataSetDirectory-Request	89
Figure 56 – XML structure of GetDataSetDirectory-Response	89
Figure 57 – XML structure of SelectActiveSG-Request	94
Figure 58 – XML structure of SelectActiveSG-Response+	94
Figure 59 – XML structure of SelectActiveSG-Response-	94
Figure 60 – XML structure of SelectEditSG-Request	95
Figure 61 – XML structure of SetEditSGValue-Request	95
Figure 62 – XML structure of SelectEditSG-Request	96
Figure 63 – XML structure of GetEditSGValue-Request	97
Figure 64 – XML structure of GetEditSGValue-Response	97
Figure 65 – XML structure of GetSGCBValues-Request	97
Figure 66 – XML structure of GetSGCBValues-Response	98
Figure 67 – XML structure of Report	102
Figure 68 – XML structure of GetBRCBValues-Request	104
Figure 69 – XML structure of GetBRCBValues-Response	105
Figure 70 – XML structure of SetBRCBValues-Request	107
Figure 71 – XML structure of SetBRCBValues-Response	107
Figure 72 – Relationship of LCB attributes to IEC 61850-7-2 log definitions	108
Figure 73 – XML structure of QueryLogByTime-Request	111
Figure 74 – XML structure of QueryLogByTime-Response	113
Figure 75 – XML structure of QueryLogAfter-Request	115
Figure 76 – XML structure of Select-Request	122
Figure 77 – XML structure of Select-Response	122
Figure 78 – XML structure of SelectWithValue-Request	124
Figure 79 – XML structure of SelectWithValue-Response+	124
Figure 80 – XML structure of SelectWithValue-Response-	125
Figure 81 – XML structure of CommandTermination Request+	131
Figure 82 – XML structure of CommandTermination Request-	132
Figure 83 – XML structure of InformationReport with AdditionalCauseDiagnosis	133
Figure 84 – Mapping of ACSI GetFile to FileOpen, FileRead, FileClose	138
Figure 85 – XML example of FileOpen Request	140

Figure 86 – XML example of FileOpen Response+	140
Figure 87 – XML example of FileRead Request (first)	140
Figure 88 – XML example of FileRead Response+ (first)	141
Figure 89 – XML example of FileRead Request (second)	141
Figure 90 – XML example of FileRead Response+ (second)	141
Figure 91 – XML example of FileClose Request	141
Figure 92 – XML example of FileClose Response	141
Figure 93 – Mapping of ACSI SetFile service	143
Figure 94 – XML example of ObtainFile Request	144
Figure 95 – XML example of ObtainFile Response	144
Figure 96 – XML example of DeleteFile Request	146
Figure 97 – XML example of DeleteFile Response	146
Figure 98 – XML example of GetFileAttributeValues Request	147
Figure 99 – XML example of GetFileAttributeValues Response	148
Figure 100 – VariableSpecification for LDevice/MHAI1.HA.phsAHar(7).cVal.mag.f	159
Figure 101 – Shorter VariableSpecification for LDevice/MHAI1.HA.phsAHar(7).cVal.mag.f	160
Figure 102 – Non conformant VariableSpecification I	161
Figure 103 – Non conformant VariableSpecification II	162
Figure 104 – VariableSpecification for LDevice/MHAI1.HA.phsAHar(7) [MX]	163
Figure 105 – Shorter VariableSpecification for LDevice/MHAI1.HA.phsAHar(7) [MX]	163
Figure A.1 – Overview of functionality and profiles	170
Figure A.2 – OSI reference model and profiles	171
Figure B.1 – Facility domain	175
Figure B.2 – Hierarchical Aggregation at facility	176
Figure B.3 – DER Management System at facility	176
Figure B.4 – Facility Management integration at DSO	177
Figure B.5 – Multiple facilities at DSO	178
Figure B.6 – VPP and contracted DERs	179
Figure B.7 – indirect control using VPP JIDs	181
Figure B.8 – VPP direct control using VPP JIDs	181
Figure B.9 – VPP direct control using DSO JIDs	182
Figure B.10– DSO Indirect control with VPP JIDs	182
Figure B.11 – DSO direct control	183
Figure B.12 – Concept of federation in XMPP	184
Figure B.13 – Federation DSO - Facility	185
Figure B.14 – Use of federation with VPP	185
Figure B.15 – communication with VPP JIDs – indirect control	186
Figure B.16 – VPP communication with VPP JIDs – direct control	186
Figure B.17 – VPP communication with DSO JIDs	187
Figure B.18 – DSO Communication with VPP JIDs	187
Figure C.1– Base System for Discussion of IT-Security Requirements	189
Figure C.2 – End-to-end security profiles	192

Figure C.3 – Initial handshake.....	193
Figure C.4 – Request / Response messages for A+/AE+ profiles.....	193
Table 1 – Services requiring client/server Communication Profile.....	24
Table 2 – Mapping of ACSI classes on MMS concepts	33
Table 3 – Mapping of ACSI BasicTypes	42
Table 4 – PhyComAddr structure.....	46
Table 5 – PhyComAddr for UPD/IP communication	46
Table 6 – GetNameList conflicting IEC 61850 objectClass and objectScope	48
Table 7 –Service error mappings for ACSI services using GetNameList	48
Table 8 – Read service error mappings	49
Table 9 – Write service error mappings	50
Table 10 – GetFileAttributeValues service error mappings	51
Table 11 – Encoding of IEC 61850-7-2 TimeQuality	53
Table 12 – Encoding of the TriggerConditions	53
Table 13 – Encoding of the ReasonForInclusionInReport	54
Table 14 – Encoding of the ReasonForInclusionInLog.....	54
Table 15 – Encoding of the RCBReportOptions	54
Table 16 – Encoding of the SVMMessageOptions.....	55
Table 17 – Encoding of the CheckConditions.....	55
Table 18 – Encoding of IEC 61850-7-2 quality	56
Table 19 – Examples of data values encoding	57
Table 20 – Mapping of ACSI GetServerDirectory (LOGICAL DEVICE).....	58
Table 21 – Mapping of ACSI GetServerDirectory (FILE).....	59
Table 22 – Association model versus communication profiles	61
Table 23 – Mapping of ACSI Associate service.....	61
Table 24 – Description of Associate request elements	62
Table 25 – Description of Associate response elements.....	62
Table 26 – Associate ACSI service error mappings.....	64
Table 27 – Mapping of ACSI Release service	64
Table 28 – Release service error mappings.....	65
Table 29 – Mapping of ACSI GeLogicalDeviceDirectory	65
Table 30 – Extended mapping of ACSI GeLogicalDeviceDirectory.....	67
Table 31 – Objectclasses for GetLogicalNodeDirectory service	68
Table 32 – Mapping of ACSI GeLogicalNodeDirectory	69
Table 33 – Extended mapping of ACSI GeLogicalNodeDirectory	71
Table 34 – Mapping of ACSI GetAllDataValues.....	73
Table 35 – Mapping of GetDataValues service parameters	75
Table 36 – Mapping of SetDataValues service parameters.....	76
Table 37 – Mapping of GetDataDirectory service parameters.....	78
Table 38 – GetDataDirectory service error mappings	79
Table 39 – Extended mapping of ACSI GetDataDirectory.....	80
Table 40 – Mapping of GetDataSetValues service parameters	82

Table 41 – GetDataSetValues error mappings	84
Table 42 – Mapping of CreateDataSet service parameters	84
Table 43 – CreateDataSet service error mappings	86
Table 44 – Mapping of DeleteDataSet service parameters	86
Table 45 – DeleteDataSet service error mappings	88
Table 46 – Mapping of GetDataSetDirectory service parameters	88
Table 47 – GetDataSetDirectory service error mappings	90
Table 48 – Mapping of ACSI ServiceType values	90
Table 49 – Mapping of ACSI errorCode values	92
Table 50 – Mapping of CDC LTS	92
Table 51 – Mapping of CDC GTS	93
Table 52 – Mapping of SGCB	93
Table 53 – BRCB structure	98
Table 54 –URCB structure	99
Table 55 – Order of AccessResults for Report	100
Table 56 – Mapping of GetBRCBValues service parameters	103
Table 57 – Mapping of SetBRCBValues service parameters	106
Table 58 – LCB structure	109
Table 59 – Mapping of QueryLogByTime service parameters	111
Table 60 – ServiceError mappings for Log services	114
Table 61 – Mapping of QueryLogAfter-Request parameters	115
Table 62 – Log conformance requirements	115
Table 63 –TypeDescription definition for GoCB structure	116
Table 64 – Controllable service parameters	118
Table 65 – Mapping of IEC 61850-7-2 control model to control components	119
Table 66 – Mapping of control services	120
Table 67 – Mapping of Select parameters	121
Table 68 – SelectWithValue service parameter mapping	122
Table 69 – Mapping of SelectWithValue parameters	123
Table 70 – SelectWithValue, Oper and Cancel DataAccessError specification	125
Table 71 – Cancel service parameter mapping	125
Table 72 – Mapping of the Cancel service	126
Table 73 – Operate service parameter mapping	127
Table 74 – Mapping of the Operate service	128
Table 75 – Mapping of the CommandTermination service	130
Table 76 – Definition of LastApplError variable structure	132
Table 77 – Mapping of ACSI AddCause values	135
Table 78 – Mapping of CDC CTS	136
Table 79 – Mapping of ACSI file class to MMS file object	136
Table 80 – Reserved file suffixes	137
Table 81 – Mapping of ACSI GetFile service	139
Table 82 – GetFile service error mappings	140
Table 83 – Mappings of ACSI ServiceErrors to FileOpen Service Errors	142

Table 84 – Mappings of ACSI ServiceErrors to FileRead Service Errors	142
Table 85 – Mappings of ACSI ServiceErrors to FileClose Service Errors	142
Table 86 – Mapping of ACSI SetFile parameters.....	144
Table 87 – Mappings of ACSI ServiceErrors to ObtainFile Service Errors	145
Table 88 – Mapping of ACSI DeleteFile service	145
Table 89 – Mappings of ACSI ServiceErrors to DeleteFile Service Errors	146
Table 90 – Mapping of ACSI GetFileAttributeValues parameters	147
Table 91 – PICS for A-Profile support.....	148
Table 92 – PICS for Time Sync A-Profile support.....	149
Table 93 – PICS for T-Profile support.....	149
Table 94 – MMS InitiateRequest general parameters	149
Table 95 – MMS InitiateResponse general parameters	150
Table 96 – MMS service supported conformance table	151
Table 97 – MMS Parameter CBB.....	153
Table 98 – GetNameList conformance statement.....	154
Table 99 – GetCapabilityList conformance statement.....	154
Table 100 – GetDomainAttributes conformance statement	155
Table 101 – Status conformance statement	155
Table 102 – Cancel conformance statement	156
Table 103 – Identify conformance statement.....	156
Table 104 – AlternateAccess conformance statement	157
Table 105 – AlternateAccessSelection conformance statement	157
Table 106 – VariableAccessSpecification conformance statement.....	163
Table 107 – VariableSpecification conformance statement.....	164
Table 108 – Read conformance statement.....	164
Table 109 – Write conformance statement.....	164
Table 110 – InformationReport conformance statement	164
Table 111 – GetVariableAccessAttributes conformance statement	165
Table 112 – DefineNamedVariableList conformance statement	165
Table 113 – GetNamedVariableListAttributes conformance statement	165
Table 114 – DeleteNamedVariableList conformance statement	166
Table 115 – ReadJournal conformance statement.....	166
Table 116 – EntryContent conformance statement.....	167
Table 117 – FileDirectory conformance statement	167
Table 118 – FileOpen conformance statement.....	168
Table 119 – FileRead conformance statement	168
Table 120 – FileClose conformance statement	168
Table 121 – Allowed P-Type definitions for client/server addressing.....	169
Table A.1 – Service and protocols for client/server communication A-Profile	172
Table A.2 – Service and protocols for client/server XMPP T-Profile.....	173
Table A.3 – Time sync A-Profile	173
Table A.4– Time sync T-Profile	174
Table D.1 – ACSI services mapping over XMPP stanzas	206

Table D.2 – Error mapping over XMPP stanzas	208
Table E.1 – Elements defined in IEC 61850-8-1 as optional with a default value.....	209
Table F.1 – SCL conformance degrees.....	211
Table F.2 – Supported ACSI services for SCL.2	211

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
XX/XX/FDIS	XX/XX/RVD

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

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

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 directly to end users and to end users via distributors, subject to IEC software licensing conditions, which can be found at: <http://www.iec.ch/CCv1>.

The Code Components included in this IEC standard are also available as electronic machine readable file at: <http://www.iec.ch/yyy/zzz>.

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

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

The National Committees are requested to note that for this publication the stability date is **20XX**.

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

Document history

Any person intervening in the present document is invited to complete the table below before sending the document elsewhere. The purpose is to allow all actors to see all changes introduced and the intervening persons.

Any important message to IEC editors should also be included in the table below.

Name of intervening person	Document received		Brief description of the changes introduced	Document sent	
	From	Date		To	Date
Margareta Noeth	Project leader	2016-11-30	CDV document (English version)	CO	2016-12-01

88 This table will be removed by IEC editors before FDIS circulation (in case of IS) or before final
89 publication (in case of TS or TR).
90

91

92

93 INTRODUCTION

94 This document is part of a set of specifications which details layered utility communication
95 architecture.

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

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

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

110 NOTE This part of IEC 61850 does not provide tutorial material. It is recommended that IEC 61850-5 and
111 IEC 61850-7-1 be read in conjunction with IEC 61850-7-2.

112

113

114

COMMUNICATION NETWORKS AND SYSTEMS FOR POWER UTILITY AUTOMATION –

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

Scope

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 the IEC 61850-7-2 only the client/server and time synchronization services are considered so far.

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 chapter 6).
- How to achieve end-to-end secured communications (see chapter 8).
- The description of the XML payloads corresponding to each ACSI service thanks in particular to the XML Schema and XML message examples (starting at chapter 10).

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

NOTE 2 This document doesn't 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 the IEC 61850-8-1 document is not necessary to understand the present one.

Normative references

The following referenced documents are indispensable for the application 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 61850-2, *Communication networks and systems in substations – Part 2: Glossary*

IEC 61850-5, *Communication networks and systems in substations – 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 – Part 7-1: Principles and models*

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

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

- 160 IEC 61850-7-4, *Communication networks and systems for power utility automation – Part 7-4:*
161 *Basic communication structure – Compatible logical node classes and data object classes*
- 162 IEC 61850-7-410, *Communication networks and systems for power utility automation – Part 7-*
163 *410: Hydroelectric power plants – Communication for monitoring and control*
- 164 IEC 61850-7-420, *Communication networks and systems for power utility automation – Part 7-*
165 *420: Basic communication structure – Distributed energy resources logical nodes*
- 166 IEC 61850-8-1, *Communication networks and systems for Power Utility Automation – Part 8-1:*
167 *Specific communication service mapping (SCSM) – Mappings to MMS (ISO 9506-1 and ISO*
168 *9506-2) and to ISO/IEC 8802-3*
- 169 IEC 62351 (all parts), *Data and communication security*
- 170 IEC TR 62357-200:2015, *Power systems management and associated information exchange -*
171 *Part 200: Guidelines for migration from Internet Protocol version 4 (IPv4) to Internet Protocol*
172 *version 6 (IPv6)*
- 173 ISO/IEC 7498-1:1994, *Information technology – Open Systems Interconnection – Basic*
174 *Reference Model: The Basic Model*
- 175 ISO/IEC 7498-3:1997, *Information technology – Open Systems Interconnection – Basic*
176 *Reference Model: Naming and addressing*
- 177 ISO/IEC 8824-1:2008, *Information technology – Abstract Syntax Notation One (ASN. 1):*
178 *Specification of basic notation*
- 179 ISO/IEC 8825-1:2008, *Information technology – ASN.1 encoding rules: Specification of XML*
180 *Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules*
181 *(DER)*
- 182 ITU X.693: SERIES X: *DATA NETWORKS AND OPEN SYSTEM COMMUNICATIONS - OSI*
183 *networking and system aspects – Abstract Syntax Notation One (ASN.1) - Information*
184 *technology – ASN.1 encoding rules: XML Encoding Rules (XER)*
- 185 ISO 9506 series, *Industrial automation systems – Manufacturing Message Specification*
- 186 ISO 9506-1:2003, *Industrial automation systems – Manufacturing Message Specification –*
187 *Part 1: Service definition*
- 188 ISO 9506-2:2003, *Industrial automation systems – Manufacturing Message Specification –*
189 *Part 2: Protocol specification*
- 190 IEEE C37.111:1999, *IEEE Standard Common Format for Transient Data Exchange*
191 *(COMTRADE) for Power Systems*
- 192 IEEE 754:1985, *IEEE Standard for Binary Floating-Point Arithmetic*
- 193 RFC 768, *User Datagram Protocol, IETF, available at <http://www.ietf.org>*
- 194 RFC 791, *Internet Protocol – DARPA Internet Program – Protocol Specification, IETF,*
195 *available at <http://www.ietf.org>*
- 196 RFC 792, *Internet Control Message Protocol – DARPA Internet Program – Protocol*
197 *Specification, IETF, available at <http://www.ietf.org>*
- 198 RFC 793, *Transmission Control Procedure – DARPA Internet Program – Protocol*
199 *Specification, IETF, available at <http://www.ietf.org>*
- 200 RFC 826, *An Ethernet Address Resolution Protocol or Converting Network Protocol*
201 *Addresses to 48.bit Ethernet Address for Transmission on Ethernet Hardware, IETF, available*
202 *at <http://www.ietf.org>*

- 203 RFC 919, *Broadcasting Internet Datagrams*, IETF, available at <http://www.ietf.org>
- 204 RFC 922, *Broadcasting Internet Datagrams in the presence of subnets*, IETF, available at
205 <http://www.ietf.org>
- 206 RFC 950, *Internet Standard Subnetting Procedure*, IETF, available at <http://www.ietf.org>
- 207 RFC 1112, *Host Extensions for IP Multicasting*, IETF, available at <http://www.ietf.org>
- 208 RFC 2460, *Internet Protocol, Version 6 (IPv6) Specification*, IETF, available at
209 <http://www.ietf.org>
- 210 RFC 3629, *UTF-8, a transformation format of ISO 1646* – IETF, available at
211 <http://www.ietf.org>
- 212 RFC 4422, *Simple Authentication and Security Layer (SASL)*, available at <http://www.ietf.org>
- 213 RFC 5246:2008, *The TLS Protocol Version 1.2*, available at <http://www.ietf.org>
- 214 RFC 5905, *Network Time Protocol (NTP) Version 4: Protocol and Algorithms Specification*,
215 IETF, available at <http://www.ietf.org>
- 216 RFC 6120, *Extensible Messaging and Presence Protocol (XMPP): Core*, available at
217 <http://www.ietf.org>
- 218 RFC 6121, *Extensible Messaging and Presence Protocol (XMPP): Instant Messaging and*
219 *Presence*, available at <http://www.ietf.org>
- 220 RFC 6122, *Extensible Messaging and Presence Protocol (XMPP): Address Format*, available
221 at <http://www.ietf.org>
- 222 XEP-0198, *Stream Management (This specification defines an XMPP protocol extension for*
223 *active management of an XML stream between two XMPP entities, including features for*
224 *stanza acknowledgements and stream resumption)*
- 225 XEP-0199, *XMPP Ping (This specification defines an XMPP protocol extension for sending*
226 *application-level pings over XML streams. Such pings can be sent from a client to a server,*
227 *from one server to another, or end-to-end)*

228

229

Terms and definitions

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

3.1**(n)-layer**

any specific layer

[ISO/IEC 7498-1, definition 3.1]

3.2**(n)-protocol data unit**

unit of data specified in an (n)-protocol and consisting of (n)-protocol-control-information and possibly (n)-user-data

[ISO/IEC 7498-1, definition 5.6.1.3]

3.3**(n)-protocol**

set of rules and formats (semantic and syntactic) which determines the communication behavior of (N)-entities in the performance of (n)-functions

[ISO/IEC 7498-1, definition 5.2.1.9]

3.4**class**

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

[IEC 61850-7-2, definition 3.1]

3.5**client**

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

[IEC 61850-7-2, definition 3.2]

3.6**device**

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

[IEC 61850-7-2, definition 3.3]

NOTE Devices alone do not perform energy transport functions.

3.7**logical device**

entity that represents a set of typical substation functions

[IEC 61850-7-2, definition 3.6]

3.8**logical node**

entity that represents a typical substation function

[IEC 61850-7-2, definition 3.7]

3.9**physical device**

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

[IEC 61850-7-2, definition 3.8]

275 NOTE Physical devices host logical devices.

276 3.10

277 application and transport profiles

278 A-Profile and T-Profile

279 set of protocols for a specific purpose

280 3.11

281 data attribute

282 named element of data object which has a specific type

283 3.12

284 data object

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

287 NOTE This definition is in accordance with the DataObject definition of IEC 61850-7-2. The MMS use of DataObject
288 refers to data types and potential instantiations of MMS variables.

289 3.13

290 server

291 entity that provides services to clients or issues unsolicited messages

292 3.14

293 private/public network

294 A network is considered as private if the unique entity using the network masters all the data
295 flows and if the performance seen by the entity using the network is guaranteed in terms of
296 bandwidth, throughput, transmission delay, availability, etc.

297 Otherwise, the network is considered as public.

298 NOTE A private network may be based on a public or shared infrastructure, as soon as the level of services can be
299 guaranteed.

300 3.15

301 smart grid

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

308 Abbreviations

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
FC	Functional Constraint
FCD	Functionally Constrained Data
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 Base Access
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
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

309

310 **Overview**311 **5.1 General**

312 The purpose of IEC 61850-8-2 is to provide detailed instructions/specifications as to the
313 mechanisms and rules required to implement the core ACSI services specified in IEC 61850-
314 7-2, while making use of XMPP, as well as the usage of SNTP for time synchronization (see
315 Figure 1).

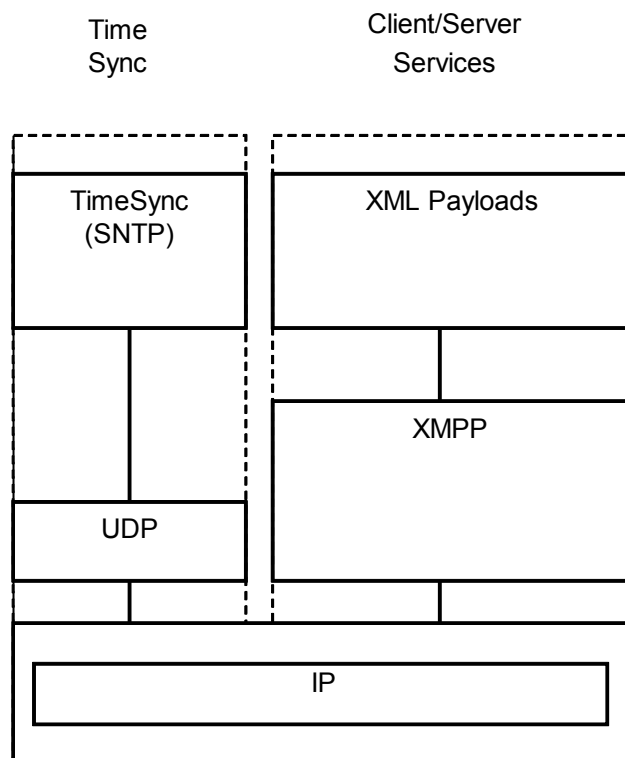


Figure 1 – Overview of functionality and profiles

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 standard 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

IEC 61850-7-2 model	IEC 61850-7-2 service
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	GetGoCBValues
	SetGoCBValues
Sampled Values (multicast/unicast)	GetMSVCBValues
	SetMSVCBValues
	GetUSVCBValues
	SetUSVCBValues
Control	Select
	SelectWithValue
	Cancel
	Operate
	CommandTermination
	TimeActivatedOperate
FILE transfer	GetFile
	SetFile
	DeleteFile
	GetFileAttributeValues

329

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

337 One of the major goals of this document is to define the structure of the XML messages
 338 transported as the payload of the XMPP stanzas.

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

5.2.2 XML payloads

The XML payloads shall be formally defined by the XML schema provided in Annex G. However when building such payloads, 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.

- The service PDU is then wrapped within the 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 the ID attribute used to create and manage the association between the applications. Figure 2 below presents an example of XML payload corresponding to an ACSI GetDataValues request. The end-to-end security profile used here can ensure end-to-end authentication and session integrity (profile A+).

<code><AssociationContext xmlns="http://www.iec.ch/61850/2016/SCSM 8 2" id="32:78"></code>	
<code><E2E-security-handshake></code> <code><data></code> <code><clear></code> <code><clearToken2></code> <code>[...]</code> <code></clearToken2></code> <code><handshakeClearTransfer-PDUs></code> <code><mmsServices></code>	Security elements
<code><confirmed-RequestPDU></code> <code><invokeID>2676121</invokeID></code> <code><ConfirmedServiceRequest></code> <code><read></code> <code><specificationWithResult></code> <code><false></code> <code></specificationWithResult></code> <code><variableAccessSpecification></code> <code><listOfVariable></code> <code><SEQUENCE></code> <code><variableSpecification></code> <code><name></code> <code><domain-specific></code> <code><domainId>IEDNameLDINst</domainId></code> <code><itemId>LLN0\$ST\$Beh\$stVal</itemId></code> <code></domain-specific></code> <code></name></code> <code></variableSpecification></code> <code></SEQUENCE></code> <code></listOfVariable></code> <code></variableAccessSpecification></code> <code></read></code> <code></ConfirmedServiceRequest></code> <code></confirmed-RequestPDU></code>	Service PDU
<code></mmsServices></code> <code></handshakeClearTransfer-PDUs></code> <code><authenticator></code> <code>[...]</code> <code></authenticator></code> <code></clear></code> <code></data></code> <code></E2E-security-handshake></code>	Security elements
<code></AssociationContext></code>	

Figure 2 – Example of XML Payload

5.2.3 XMPP

XMPP is a protocol defined at the Internet Engineering Task Force (IETF) by the 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 is using 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.

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 may 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 must 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 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 the following clauses shall apply.

6.2.2 Usage of TLS and SASL

TLS shall be used in any case. 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) can 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) can 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 TSL 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 (profile AE+, see 8).

6.3 Mapping of solicited 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 below) .

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

Figure 3 – Generic structure of client/server ACSI services

All solicited services shall be implemented by <iq> stanzas. The request message is implemented by an <iq> stanza with the attribute “type” equal to “set” or “get” (see example in Figure 5) and the response message is implemented by an <iq> stanza with attribute “type” set to “result” (see example in Figure 5). The direct child element of the <iq> stanza, AssociationContext, represents the association established between the client and the server and it contains the XML payload representing the ACSI service parameters and the security elements used for end-to-end security.

Table D.1 defines for each ACSI service which kind of stanza and which value of the “type” attribute shall be used.

```
<iq type="get" id="446-016" from="IEDclient@XMPPserver/IEC" to="IEDserver@XMPPserver/IEC">
  <AssociationContext xmlns="http://www.iec.ch/61850/2016/SCSM_8_2" id="32:78">
    .....
  </AssociationContext>
</iq>
```

Figure 4 – Example of <iq> stanza request

```
<iq type="result" id="446-016" from="IEDserver@XMPPserver/IEC" to="IEDclient@XMPPserver/IEC">
  <AssociationContext xmlns="http://www.iec.ch/61850/2016/SCSM_8_2" id="32:78">
    .....
  </AssociationContext>
</iq>
```

Figure 5 – Example of <iq> stanza response

Errors occurring in upper levels of the communication stack (i.e. within the IEC 61850 application that builds or consumes the Service PDU or within the end-to-end security layer)

shall be transmitted as a normal <iq/> response with type="result" or as a <message/> with type="normal" and the appropriate XML payload as defined in 9.3.3.4 and Table D.2.

Errors occurring in lower levels (XMPP or management of the AssociationContext) shall use the error mechanism defined by XMPP (see also 9.3.3.4 and Table D.2)

6.4 Mapping of unsolicited services

Unsolicited services shall be implemented as <message/> stanzas of type "normal" (see Table D.1). The direct child element of the stanza, AssociationContext, represents the association established between the client and the server and it contains the XML payload representing the ACSI service parameters and the security elements used for end-to-end security.

```
<message type="normal" id="20175-011" from="IEDserver@XMPPserver/IEC"
to="IEDclient@XMPPserver/IEC">
  <AssociationContext xmlns="http://www.iec.ch/61850/2016/SCSM_8_2" id="32:78">
    .....
  </AssociationContext>
</message>
```

Figure 6 – Example of <message> stanza

Errors occurring in upper levels of the communication stack (i.e. within the IEC 61850 application that builds or consumes the Service PDU or within the end-to-end security layer) shall be transmitted as a <message/> with type="normal" and the appropriate XML payload as defined in 9.3.3.4 and Table D.2.

Errors occurring in lower levels (XMPP or management of the AssociationContext) shall use the error mechanism defined by XMPP (see also 9.3.3.4 and Table D.2)

6.5 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.

NOTE 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 must exist before the Associate service is called.

6.6 Usage of the Roster

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

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

6.7 XMPP extensions

6.7.1 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.

6.7.2 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 can 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 to resend already handled stanzas, all entities (even those not supporting the XEP 0198) shall check the stanza "id" attribute in order to ignore the duplicates.

6.8 Implementation agreements**6.8.1 XMPP PING – XEP 0199**

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

Association context management

The present SCSM uses the concept of two-party application association as defined by IEC 61850-7-2. This concept allows to convey the requests and responses for both confirmed and unconfirmed services.

The AssociationId as defined by IEC 61850-7-2 is generated during the Associate service. The caller generates a unique identifier (unique in its scope) and send it within the Associate request. Then the called entity generates also a unique identifier in its scope, appends it to the client identifier and returns both identifiers in the Associate response. This pair of identifiers forms the AssociationId which shall then be part of all exchanged messages between the two associated entities.

NOTE 1 During the Association request and its related errors, the partial AssociationId generated only by the caller application is used.

NOTE 2 Prior to establishing an association between two application entities, an underlying transport connection has to be established, i.e. each endpoint must be connected to its XMPP server. If the two endpoints doesn't belong to the same XMPP domain, the two related XMPP server shall be federated.

Once the association is established, both endpoints (caller and called) shall monitor the status of the communication. In case an endpoint detects a communication outage, a local abort shall be raised up to the IEC 61850 application level:

- If a loss of link is detected by an endpoint at the XMPP client level (e.g. broken TCP connection between XMPP client and XMPP server), then a local Abort shall be raised for all Associations using this link, i.e. all the Associations where this endpoint is involved with the same JID.
- If the XMPP client of an endpoint is informed by the XMPP server that a JID involved in an association is "unavailable" (through the presence mechanism, see 6.5), then a local Abort shall be raised but only for the Associations involving this given JID.

The mapping of the AssociationId within the XML payload is described in clause 9. Further details regarding the management of the association are also described in chapter 11 and annex C.5.

End-to-end security

The end-to-end security mechanism used in the present SCSM is specified by IEC 62351-4 (work in progress). 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.

This clause describes the principles of this end-to-end security mechanism. Clause 9.1, Annex C and more specifically the XSD included in Annex G describe how to map the end-to-end security mechanism for the present SCSM.

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 key necessary to secure the subsequent data exchanges by using symmetric cryptographic algorithms.

There are in fact two possible security profiles which can be negotiated during the initial handshake:

- The A+ profile provides end-to-end authentication and ensures message integrity during association establishment and for the subsequent data transfer phase bound to the initial authentication. Then the XML payloads exchanged during the session includes a security token for holding the information necessary for maintaining the secured session as well as a MAC for ensuring the integrity of both the token and the service PDU representing the ACSI service parameters.
- The AE+ profile provides the same protection as the A+ one. In addition, it provides confidentiality during the data transfer phase by encrypting the service PDU.

Payload description

9.1 XSD overview

The XML schema included in Annex G defines formally the payload transported by the XMPP stanzas.

The entry point in this schema is the definition of the element <AssociationContext/> which shall be the first child element of any <iq/> or <message/> stanza used by the present SCSM. The namespace http://www.iec.ch/61850/2016/SCSM_8_2 shall be also declared within this element (see Figure 7). The mandatory attribute “id” shall map the AssociationId defined by IEC 61850-7-2. As such, the values of this attribute shall be managed as defined in 7 for the AssociationId. Note that the general format of this attribute shall be two decimal identifiers generated respectively by the caller and the called entities separated by a colon.

```
<iq type="get" id="446-016" from="IEDclient@XMPPserver/IEC" to="IEDserver@XMPPserver/IEC">
  <AssociationContext xmlns="http://www.iec.ch/61850/2016/SCSM_8_2" id="32:78">
    [...]
  </AssociationContext>
</iq>
```

Figure 7 – Example of AssociationContext element

The content of the <AssociationContext/> element is defined at the first nested level by an XML element <E2E-security-handshake/>. The type definition of this element and all the nested definitions below are referring to XML complexTypes and simpleTypes in order to define the entire payload.

The upper levels definitions relate to end-to-end security and are direct mappings of the ASN.1 elements defined by IEC 62351-4 for holding security elements (i.e. tokens, signatures and authenticators) and ACSI services PDUs.

<E2E-security-handshake/> is defined as a choice of various elements that shall be used respectively for:

- Transporting the request and response messages for creating the association and the end-to-end security session
- Transporting errors raised by the end-to-end security mechanism
- Transporting the payload of the ACSI services once the association is established

Within the definitions used for transfer of data when the association is established, the complexType Handshake-DataTransfer gives the choice for embedding the ACSI service PDU either in clear (case of the A+ profile) or encrypted (for the AE+ profile). In the latter case the ACSI service PDU is finally defined as an element <encrypted-MMSpdu/> with an opaque OCTET STRING value (see definition of OCTET STRING in 9.3.2.4). In the former case, the ACSI service PDU is completely detailed within the element <mmsServices/> using the complexType EE-MMSpdu.

NOTE In case of AE+ profile, the OCTET STRING within the <encrypted-MMSpdu/> element shall contain the encrypted form of the equivalent value of type EE-MMSpdu which would be used for mapping the same ACSI service in A+ profile case.

The EE-MMSpdu complexType proposes then 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 10.

9.2 Objects of IEC 61850

9.2.1 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

9.2.2 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.

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

646 The type description of this structure can be created by the algorithm in Figure 8.

For each functional constraint (c) found in Figure 9¹

IEC 813/11

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 8 – Algorithm for logical node mapping

647

648 The order of the resulted components is recommended to be as shown in Figure 9.

649

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

IEC 814/11

650

Figure 9 – Ordered list of functional constraints

651

¹ The recommended ordering provides compatibility with existing UCA 2.0 implementations

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 – GSSE control blocks; SV – substituted values; SE – Setting group editing; MS – Multicast Sampled Values control block (from IEC 61850-9-2); US – Unicast Sampled Values control block (from IEC 61850-9-1); 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 SCSM. 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.

NOTE 3 To maintain backward compatibility with previous version, then the order of the instantiated compatible data classes as SCL DOs shall meet the requirements of that previous version.

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 10 is the graphical illustration of a hierarchical type description of a logical node instance, as produced by the algorithm of Figure 8.

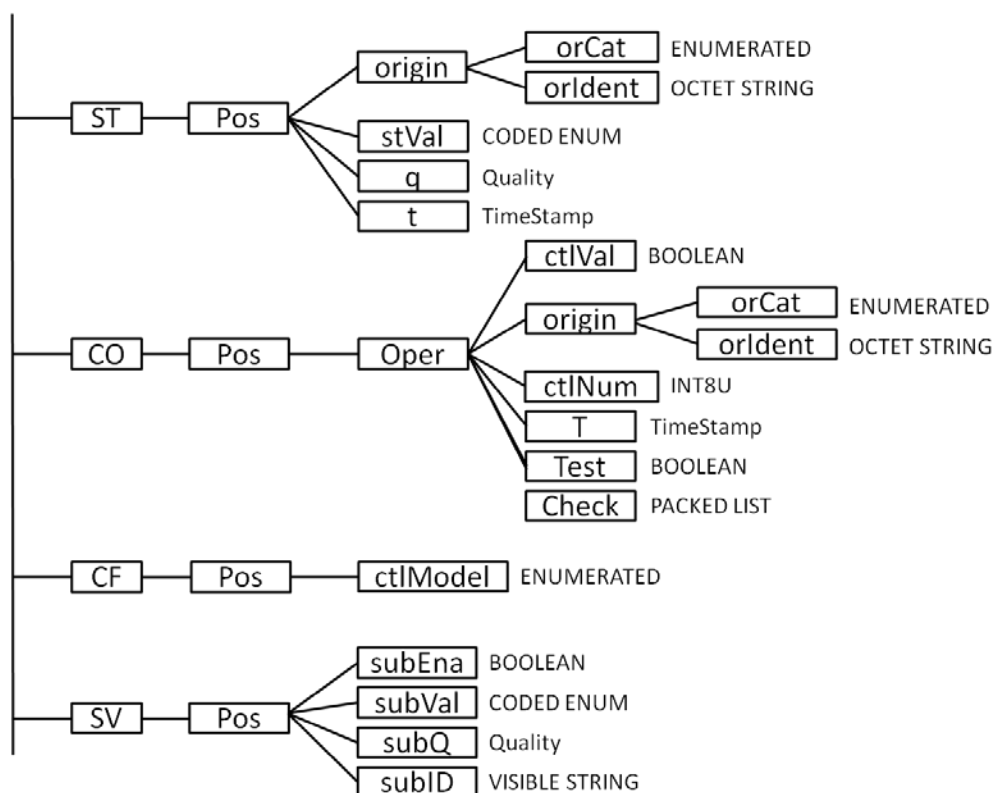


Figure 10 – Example of Logical Node type description

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

683 constrained by the maximum identifier size : 64 characters. Figure 11 presents the list of the
 684 flattened Named Variables derived from the LN structure of Figure 10.

```

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
  
```

685

686 **Figure 11 – List of the flattened Named Variables corresponding to an LN**

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

689 9.2.3 Mapping of references of Logical Nodes to VariableAccessSpecifications

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

696 Within a domain-specific element, the domainId shall be the name of the LogicalDevice and
 697 the ItemId shall be the name of the LogicalNode. Figure 12 is a mapping example for the
 698 reference LDName/LNName = CONTROL/LLN0.

```

699      <variableAccessSpecification>
700      <listOfVariable>
701      <SEQUENCE>
702      <variableSpecification>
703      <name>
704      <domain-specific>
705      <domainId> CONTROL </domainId>
706      <itemId> LLN0 </itemId>
707      </domain-specific>
708      </name>
709      </variableSpecification>
710      </SEQUENCE>
711      </listOfVariable>
712      </variableAccessSpecification>

```

Figure 12 – XML mapping of a LNReference with direct access

An alternate specification shall be supported when the LNReference is constrained to one specific FC (case of GetAllDataValues see 13.3.1). Figure 13 is an example of VariableAccessSpecification where an alternateAccess element is used for encoding the functional constraint ST.

```

718      <variableAccessSpecification>
719      <listOfVariable>
720      <SEQUENCE>
721      <variableSpecification>
722      <name>
723      <domain-specific>
724      <domainId>CONTROL</domainId>
725      <itemId>GGIO1</itemId>
726      </domain-specific>
727      </name>
728      </variableSpecification>
729      <alternateAccess>
730      <unnamed>
731      <selectAlternateAccess>
732      <accessSelection>
733      <component>ST</component>
734      </accessSelection>
735      </alternateAccess/>
736      </selectAlternateAccess>
737      </unnamed>
738      </alternateAccess>
739      </SEQUENCE>
740      </listOfVariable>
741      </variableAccessSpecification>

```

Figure 13 – XML mapping of a LNReference with alternate access

9.2.4 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\$A\$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 14).

```

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

```

Figure 14 – 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 9.2.6).

9.2.5 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 15).

EXAMPLES: <LNVariableName>\$<FC>\$<LNDataName1>\$<AttributeName1> (e.g. XCBR1\$ST\$Pos\$stVal)

<LNVariableName>\$<FC>\$<LNDataName1>\$<AttributeName1>\$<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 15 – 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 9.2.6).

9.2.6 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

803 element <alternateAccess/> after the element <variableSpecification/>. See also clauses
804 regarding AlternateAccess conformance in 25.2.2.2.1.

805 NOTE 1 The character '\$' is not allowed within values of <component/> elements in the alternate access part.

806 NOTE 2 It is recommended to always use the choice <unnamed/> within the alternate access part.

807

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

810 Figure 16: a nested component without any array.

811 NOTE This reference may have been encoded without alternate access with itemId= LLN0\$ST\$Beh\$stVal

```

812         <variableSpecification>
813             <name>
814                 <domain-specific>
815                     <domainId>LDevice</domainId>
816                     <itemId>LLN0</itemId>
817                 </domain-specific>
818             </name>
819         </variableSpecification>
820         <alternateAccess>
821             <unnamed>
822                 <selectAlternateAccess>
823                     <accessSelection>
824                         <component>ST</component>
825                     </accessSelection>
826                 </alternateAccess>
827                 <unnamed>
828                     <selectAlternateAccess>
829                         <accessSelection>
830                             <component>Beh</component>
831                         </accessSelection>
832                     </alternateAccess>
833                     <unnamed>
834                         <selectAlternateAccess>
835                             <accessSelection>
836                                 <component>stVal</component>
837                             </accessSelection>
838                         </alternateAccess>
839                     </selectAlternateAccess>
840                 </unnamed>
841             </alternateAccess>
842         </selectAlternateAccess>
843     </unnamed>
844 </alternateAccess>
845 </selectAlternateAccess>
846 </unnamed>
847 </alternateAccess>

```

848 **Figure 16 – Alternate access without array element**

849 Figure 17: a nested component within an array element. Note that the component path before
850 the index may have been encoded as a flattened variable within the <variableSpecification/>
851 element (see Figure 18).

```

852 <variableSpecification>
853     <name>
854         <domain-specific>
855             <domainId>LDevice</domainId>
856             <itemId>MHAI1</itemId>
857         </domain-specific>

```

```

858     </name>
859 </variableSpecification>
860 <alternateAccess>
861     <unnamed>
862         <selectAlternateAccess>
863             <accessSelection>
864                 <component>MX</component>
865             </accessSelection>
866         </alternateAccess>
867         <unnamed>
868             <selectAlternateAccess>
869                 <accessSelection>
870                     <component>HA</component>
871                 </accessSelection>
872             </alternateAccess>
873             <unnamed>
874                 <selectAlternateAccess>
875                     <accessSelection>
876                         <component>phsAHar</component>
877                     </accessSelection>
878                 </alternateAccess>
879                 <unnamed>
880                     <selectAlternateAccess>
881                         <accessSelection>
882                             <index>7</index>
883                         </accessSelection>
884                     </alternateAccess>
885                     <unnamed>
886                         <selectAlternateAccess>
887                             <accessSelection>
888                                 <component>cVal</component>
889                             </accessSelection>
890                         </alternateAccess>
891                         <unnamed>
892                             <selectAlternateAccess>
893                                 <accessSelection>
894                                     <component>mag</component>
895                                 </accessSelection>
896                             </alternateAccess>
897                             <unnamed>
898                                 <selectAlternateAccess>
899                                     <accessSelection>
900                                         <component>f</component>
901                                     </accessSelection>
902                                 </alternateAccess>
903                             </selectAlternateAccess>
904                         </unnamed>
905                     </alternateAccess>
906                 </selectAlternateAccess>
907             </unnamed>
908         </alternateAccess>
909     </selectAlternateAccess>
910 </unnamed>
911 </alternateAccess>
912 </selectAlternateAccess>
913 </unnamed>
914 </alternateAccess>
915 </selectAlternateAccess>
916 </unnamed>
917 </alternateAccess>
918 </selectAlternateAccess>
919 </unnamed>

```

```

920     </alternateAccess>
921   </selectAlternateAccess>
922 </unnamed>
923 </alternateAccess>

```

Figure 17 – Alternate access with array element

```

925 <variableSpecification>
926   <name>
927     <domain-specific>
928       <domainId>LDevice</domainId>
929       <itemId>MHAI1$MX$HA$phsAHar</itemId>
930     </domain-specific>
931   </name>
932 </variableSpecification>
933 <alternateAccess>
934   <unnamed>
935     <selectAlternateAccess>
936       <accessSelection>
937         <index>7</index>
938       </accessSelection>
939       <alternateAccess>
940         <unnamed>
941           <selectAlternateAccess>
942             <accessSelection>
943               <component>cVal</component>
944             </accessSelection>
945             <alternateAccess>
946               <unnamed>
947                 <selectAlternateAccess>
948                   <accessSelection>
949                     <component>mag</component>
950                   </accessSelection>
951                   <alternateAccess>
952                     <unnamed>
953                       <selectAlternateAccess>
954                         <accessSelection>
955                           <component>f</component>
956                         </accessSelection>
957                       </alternateAccess>
958                     </selectAlternateAccess>
959                   </unnamed>
960                 </alternateAccess>
961               </selectAlternateAccess>
962             </unnamed>
963           </alternateAccess>
964         </selectAlternateAccess>
965       </unnamed>
966     </alternateAccess>
967   </selectAlternateAccess>
968 </unnamed>
969 </alternateAccess>

```

Figure 18 – Alternate access with flattened variable and array element

972 **9.3 Mapping of IEC 61850-7-2 data attributes**973 **9.3.1 BasicTypes**

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

977 **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 9.3.2.2. 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 9.3.2.1). Used here for an ordered set of values, defined where type is used. See also 9.3.2.3. 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 9.3.2.4. 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 9.3.2.5. Example: RptID of a ReportControl is a Visible String129 <visible-string>CONTROL/GGIO1\$BR\$simpleBReport05</visible-string>
UNICODE STRING	<mMSString/>	MMSString	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 9.3.2.6. Example: <mMSString>ābcdēfgh<nul/>ījkĚμñópq<soh/>r\$ťùvwxyz</mMSString>

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.

978

979 9.3.2 Additional definitions of BasicType

980 9.3.2.1 General

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

983 BIT_STRING is a restriction of xsd:string type, restricted to zero or more "0" and "1"
984 characters. Bit(0) is mapped to the most significant bit of the value.

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

989 9.3.2.2 ENUMERATED

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

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

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

1000 NOTE For IEC 61850 enumerated values, where the actual numeric values are specified, the integer values within
1001 the context of IEC 61850-8-2 shall be the same as the numeric values specified.

1002 9.3.2.3 CODED ENUM

1003 The IEC 61850-7-2 CODED ENUM is mapped on a <bit-string/> element with a BIT_STRING
1004 type value, as defined in the normative XSD (see 9.3.2.1). The size of the string shall be the
1005 number of bits required to represent the maximum enumerate value defined in IEC 61850-7-2.
1006 It shall represent the unsigned integer encoding of the enumerated values in the order
1007 specified in IEC 61850-7-2.

1008 For the purposes of IEC 61850-8-2, CODED ENUMs that appear within a PACKED LIST or an
1009 array of other CODED ENUMs are treated as special cases.

1010 9.3.2.4 OCTET STRING

1011 The IEC 61850-7-2 OCTET STRING shall map to an <octet-string/> element with an
1012 OCTET_STRING type value. OCTET_STRING is a restriction of xsd:string for encoding
1013 hexadecimal values, i.e. based on the character set [0-9][A-F][a-f] but including possibly
1014 extra space characters. The maximum size shall be as specified in IEC 61850-7-2 and
1015 IEC 61850-7-3 and do not include the extra spaces added for readability.

1016 9.3.2.5 VISIBLE STRING

1017 The IEC 61850-7-2 VISIBLE STRING shall map to a <visible-string/> element with a
1018 VisibleString type value. VisibleString is defined in the XSD as a an xsd:string restricted to
1019 visible characters, i.e. the range from Space (ascii code 32) to Tilde (ascii code 126). The
1020 maximum size shall be as specified in IEC 61850-7-2 and IEC 61850-7-3.

1021 The character set shall be constrained to ISO646String.

9.3.2.6 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.

9.3.2.7 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>
```

9.3.3 Common ACSITypes**9.3.3.1 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.

9.3.3.2 ObjectReference**9.3.3.2.1 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 9.3.3.2.2.
- Within the scope of data attributes (Tracking CDCs,ORG): see 9.3.3.2.3.
- Within the scope of DataRef of reports: see 9.3.3.2.4.

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

9.3.3.2.2 Object Reference within Control Blocks

The value shall be constructed as:

- AA-specific scope objects: @<Object Name>

1067 • Domain specific: <Domain Name>/<Object Name>

1068 • VMD-specific: /<Object Name>

1069 The maximum Object Name and Domain Name size shall be 64.

1070 The allowed character set is and the same as ObjectName plus the “\$”, “/” and “@”
1071 characters.

1072 9.3.3.2.3 Object Reference within CDCs (Tracking, ORG).

1073 The value shall be constructed as specified in IEC 61850- 7-2.

1074 The allowed character set is defined in IEC 61850-7-2. The value shall be constructed as:

1075 • Reference to a Logical Device:

1076 <LDName>

1077 • Reference to a Logical Node:

1078 <LDName>/<LNName>

1079 • Reference to a DataSet:

1080 <LDName>/<LNName>.<DataSetName>

1081 • Reference to a DataObject:

1082 <LDName>/<LNName>.<DataObjectName>[.<SubDataObjectName>[. ...]]

1083 • Reference to a DataAttribute:

1084 <LDName>/<LNName>.<DataObjectName>[.<SubDataObjectName>[.
1085 ...]].<DataAttributeName>[(<NumArrayElement>)][.<SubDataAttributeName>[. ...]]

1086 • Reference to a Control Block

1087 <LDName>/<LNName>.<CBName>

1088 9.3.3.2.4 Object Reference within DataRef of reports.

1089 The value shall be constructed as:

1090 <Domain Name>/<Object Name>

1091 The allowed character set is defined in IEC 61850-7-2.

1092 The value shall be constructed as:

1093 • Reference to a FCD DataObject:

1094 <LDName>/<LNName>.<FC>.<DataObjectName>[.<SubDataObjectName>[(
1095 rrayElement>)][.\$...]]

1096 Example: The datRef representing the ACSI reference LDevice/MHA11.HA.phsAHar(7) functionally constrained MX
1097 will be LD1/MMHA1\$MX\$HA\$phsAHar(7)

1098 • Reference to a FCDA

1099 <LDName>/<LNName>.<FC>.<DataObjectName>[.<SubDataObjectName>[(
1100 rrayElement>)][.\$...]] <DataAttributeName> [(
1101 \$<SubDataAttributeName>[\$...]]

1102 Example: The datRef representing the ACSI reference LDevice/MHA11.HA.phsAHar(7).cVal.mag.f will be
1103 LD1/MMHA1\$MX\$HA\$phsAHar(7)\$cVal\$mag\$f

1104