

INTERNATIONAL STANDARD



OPC unified architecture –
Part 7: Profiles

IECNORM.COM : Click to view the full PDF of IEC 62541-7:2020 RLV



THIS PUBLICATION IS COPYRIGHT PROTECTED

Copyright © 2020 IEC, Geneva, Switzerland

All rights reserved. Unless otherwise specified, no part of this publication may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm, without permission in writing from either IEC or IEC's member National Committee in the country of the requester. If you have any questions about IEC copyright or have an enquiry about obtaining additional rights to this publication, please contact the address below or your local IEC member National Committee for further information.

IEC Central Office
3, rue de Varembe
CH-1211 Geneva 20
Switzerland

Tel.: +41 22 919 02 11
info@iec.ch
www.iec.ch

About the IEC

The International Electrotechnical Commission (IEC) is the leading global organization that prepares and publishes International Standards for all electrical, electronic and related technologies.

About IEC publications

The technical content of IEC publications is kept under constant review by the IEC. Please make sure that you have the latest edition, a corrigendum or an amendment might have been published.

IEC publications search - webstore.iec.ch/advsearchform

The advanced search enables to find IEC publications by a variety of criteria (reference number, text, technical committee,...). It also gives information on projects, replaced and withdrawn publications.

IEC Just Published - webstore.iec.ch/justpublished

Stay up to date on all new IEC publications. Just Published details all new publications released. Available online and once a month by email.

IEC Customer Service Centre - webstore.iec.ch/csc

If you wish to give us your feedback on this publication or need further assistance, please contact the Customer Service Centre: sales@iec.ch.

Electropedia - www.electropedia.org

The world's leading online dictionary on electrotechnology, containing more than 22 000 terminological entries in English and French, with equivalent terms in 16 additional languages. Also known as the International Electrotechnical Vocabulary (IEV) online.

IEC Glossary - std.iec.ch/glossary

67 000 electrotechnical terminology entries in English and French extracted from the Terms and Definitions clause of IEC publications issued since 2002. Some entries have been collected from earlier publications of IEC TC 37, 77, 86 and CISPR.

IECNORM.COM : Click to view the full text of IEC 60364-7:2020 RVV

INTERNATIONAL STANDARD



**OPC unified architecture –
Part 7: Profiles**

INTERNATIONAL
ELECTROTECHNICAL
COMMISSION

ICS 25.040.40; 35.100.05

ISBN 978-2-8322-8562-6

Warning! Make sure that you obtained this publication from an authorized distributor.

CONTENTS

FOREWORD	12
1 Scope	15
2 Normative references	15
3 Terms, definitions, and ~~conventions~~ abbreviated terms	16
3.1 Terms and definitions	16
3.2 Abbreviated terms	17
4 Overview	17
4.1 General	17
4.2 ConformanceUnit	18
4.3 Profiles	18
4.4 Profile Categories	19
5 Conformance Units	19
5.1 Overview	19
5.2 Services	20
5.3 Transport and communication related features	30
5.4 Information Model and AddressSpace related features	42
5.5 Miscellaneous	62
6 Profiles	64
6.1 Overview	64
6.2 Profile list	64
6.3 Conventions for Profile definitions	71
6.4 Profile versioning	71
6.5 Applications	71
6.6 Profile tables	73
6.6.1 General	73
6.6.2 Core Server Facet	73
6.6.3 Core 2017 Server Facet	73
6.6.4 Sessionless Server Facet	74
6.6.5 Reverse Connect Server Facet	75
6.6.6 Base Server Behaviour Facet	75
6.6.7 Request State Change Server Facet	75
6.6.8 Subnet Discovery Server Facet	75
6.6.9 Global Certificate Management Server Facet	76
6.6.10 Authorization Service Server Facet	76
6.6.11 KeyCredential Service Server Facet	76
6.6.12 Attribute WriteMask Server Facet	76
6.6.13 File Access Server Facet	77
6.6.14 Documentation Server Facet	77
6.6.15 Embedded DataChange Subscription Server Facet	77
6.6.16 Standard DataChange Subscription Server Facet	78
6.6.17 Standard DataChange Subscription 2017 Server Facet	78
6.6.18 Enhanced DataChange Subscription Server Facet	78
6.6.19 Enhanced DataChange Subscription 2017 Server Facet	78
6.6.20 Durable Subscription Server Facet	79
6.6.21 Data Access Server Facet	79
6.6.22 ComplexType Server Facet	79

6.6.23	ComplexType 2017 Server Facet	80
6.6.24	Standard Event Subscription Server Facet	80
6.6.25	Address Space Notifier Server Facet	81
6.6.26	A & C Base Condition Server Facet	81
6.6.27	A & C Refresh2 Server Facet	82
6.6.28	A & C Address Space Instance Server Facet	82
6.6.29	A & C Enable Server Facet	82
6.6.30	A & C AlarmMetrics Server Facet	82
6.6.31	A & C Alarm Server Facet	83
6.6.32	A & C Acknowledgeable Alarm Server Facet	83
6.6.33	A & C Exclusive Alarming Server Facet	84
6.6.34	A & C Non-Exclusive Alarming Server Facet	84
6.6.35	A & C Previous Instances Server Facet	84
6.6.36	A & C Dialog Server Facet	85
6.6.37	A & C CertificateExpiration Server Facet	85
6.6.38	A & E Wrapper Facet	85
6.6.39	Method Server Facet	86
6.6.40	Auditing Server Facet	86
6.6.41	Node Management Server Facet	87
6.6.42	User Role Base Server Facet	87
6.6.43	User Role Management Server Facet	87
6.6.44	State Machine Server Facet	88
6.6.45	Client Redundancy Server Facet	88
6.6.46	Redundancy Transparent Server Facet	88
6.6.47	Redundancy Visible Server Facet	89
6.6.48	Historical Raw Data Server Facet	89
6.6.49	Historical Aggregate Server Facet	89
6.6.50	Historical Data AtTime Server Facet	90
6.6.51	Historical Access Modified Data Server Facet	91
6.6.52	Historical Annotation Server Facet	91
6.6.53	Historical Data Insert Server Facet	91
6.6.54	Historical Data Update Server Facet	91
6.6.55	Historical Data Replace Server Facet	92
6.6.56	Historical Data Delete Server Facet	92
6.6.57	Historical Access Structured Data Server Facet	92
6.6.58	Base Historical Event Server Facet	92
6.6.59	Historical Event Update Server Facet	93
6.6.60	Historical Event Replace Server Facet	93
6.6.61	Historical Event Insert Server Facet	93
6.6.62	Historical Event Delete Server Facet	93
6.6.63	Aggregate Subscription Server Facet	94
6.6.64	Nano Embedded Device Server Profile	95
6.6.65	Nano Embedded Device 2017 Server Profile	95
6.6.66	Micro Embedded Device Server Profile	95
6.6.67	Micro Embedded Device 2017 Server Profile	95
6.6.68	Embedded UA Server Profile	95
6.6.69	Embedded 2017 UA Server Profile	96
6.6.70	Standard UA Server Profile	96
6.6.71	Standard 2017 UA Server Profile	96

6.6.72	Core Client Facet.....	97
6.6.73	Core 2017 Client Facet	97
6.6.74	Sessionless Client Facet	98
6.6.75	Reverse Connect Client Facet	98
6.6.76	Base Client Behaviour Facet.....	98
6.6.77	Discovery Client Facet.....	98
6.6.78	Subnet Discovery Client Facet.....	99
6.6.79	Global Discovery Client Facet.....	99
6.6.80	Global Certificate Management Client Facet	99
6.6.81	KeyCredential Service Client Facet.....	99
6.6.82	Access Token Request Client Facet	100
6.6.83	AddressSpace Lookup Client Facet	100
6.6.84	Request State Change Client Facet	100
6.6.85	File Access Client Facet	101
6.6.86	Entry-Level Support 2015 Client Facet.....	101
6.6.87	Multi-Server Client Connection Facet.....	101
6.6.88	Documentation – Client	101
6.6.89	Attribute Read Client Facet.....	102
6.6.90	Attribute Write Client Facet.....	102
6.6.91	DataChange Subscriber Client Facet	102
6.6.92	Durable Subscription Client Facet.....	103
6.6.93	DataAccess Client Facet.....	103
6.6.94	Event Subscriber Client Facet.....	104
6.6.95	Base Event Processing Client Facet	104
6.6.96	Notifier and Source Hierarchy Client Facet	105
6.6.97	A & C Base Condition Client Facet	105
6.6.98	A & C Refresh2 Client Facet.....	105
6.6.99	A & C Address Space Instance Client Facet	106
6.6.100	A & C Enable Client Facet	106
6.6.101	A & C Alarm Metrics Client Facet.....	106
6.6.102	A & C Alarm Client Facet.....	106
6.6.103	A & C Exclusive Alarming Client Facet.....	107
6.6.104	A & C Non-Exclusive Alarming Client Facet	107
6.6.105	A & C Previous Instances Client Facet.....	108
6.6.106	A & C Dialog Client Facet	108
6.6.107	A & C CertificateExpiration Client Facet.....	108
6.6.108	A & E Proxy Facet	109
6.6.109	Method Client Facet.....	110
6.6.110	Auditing Client Facet	110
6.6.111	Node Management Client Facet.....	110
6.6.112	Advanced Type Programming Client Facet	110
6.6.113	User Role Management Client Facet.....	111
6.6.114	State Machine Client Facet.....	111
6.6.115	Diagnostic Client Facet.....	111
6.6.116	Redundant Client Facet	112
6.6.117	Redundancy Switch Client Facet	112
6.6.118	Historical Access Client Facet	112
6.6.119	Historical Data AtTime Client Facet	112
6.6.120	Historical Aggregate Client Facet.....	112

6.6.121	Historical Annotation Client Facet	114
6.6.122	Historical Access Modified Data Client Facet	114
6.6.123	Historical Data Insert Client Facet	114
6.6.124	Historical Data Update Client Facet	114
6.6.125	Historical Data Replace Client Facet	114
6.6.126	Historical Data Delete Client Facet	115
6.6.127	Historical Access Client Server Timestamp Facet	115
6.6.128	Historical Structured Data Access Client Facet	115
6.6.129	Historical Structured Data AtTime Client Facet	115
6.6.130	Historical Structured Data Modified Client Facet	116
6.6.131	Historical Structured Data Insert Client Facet	116
6.6.132	Historical Structured Data Update Client Facet	116
6.6.133	Historical Structured Data Replace Client Facet	116
6.6.134	Historical Structured Data Delete Client Facet	116
6.6.135	Historical Events Client Facet	117
6.6.136	Historical Event Insert Client Facet	117
6.6.137	Historical Event Update Client Facet	117
6.6.138	Historical Event Replace Client Facet	117
6.6.139	Historical Event Delete Client Facet	118
6.6.140	Aggregate Subscriber Client Facet	118
6.6.141	Standard UA Client Profile	119
6.6.142	Standard UA Client 2017 Profile	119
6.6.143	UA-TCP UA-SC UA-Binary	120
6.6.144	HTTPS UA-Binary	120
6.6.145	HTTPS UA-XML	121
~~SOAP HTTP WS SC UA XML~~		
~~SOAP HTTP WS SC UA Binary~~		
~~SOAP HTTP WS SC UA XML UA Binary~~		
6.6.146	HTTPS UA-JSON	121
6.6.147	WSS UA-SC UA-Binary	122
6.6.148	WSS UA-JSON	122
6.6.149	Security User Access Control Full	122
6.6.150	Security User Access Control Base	123
6.6.151	Security Time Synchronization	123
6.6.152	Best Practice – Audit Events	123
6.6.153	Best Practice – Alarm Handling	123
6.6.154	Best Practice – Random Numbers	124
6.6.155	Best Practice – Timeouts	124
6.6.156	Best Practice – Administrative Access	124
6.6.157	Best Practice – Strict Message Handling	124
6.6.158	Best Practice – Audit Events Client	125
~~TransportSecurity – TLS 1.0~~		
~~TransportSecurity – TLS 1.1~~		
6.6.159	TransportSecurity – TLS 1.2	126
6.6.160	TransportSecurity – TLS 1.2 with PFS	126
6.6.161	SecurityPolicy – None	126
6.6.162	SecurityPolicy – Basic128Rsa15	127
6.6.163	SecurityPolicy – Basic256	127
6.6.164	SecurityPolicy [A] – Aes128-Sha256-RsaOaep	128

6.6.165	SecurityPolicy [B] – Basic256Sha256	128
6.6.166	SecurityPolicy – Aes256-Sha256-RsaPss	129
6.6.167	User Token – Anonymous Facet	129
6.6.168	User Token – User Name Password Server Facet	130
6.6.169	User Token – X509 Certificate Server Facet	130
6.6.170	User Token – Issued Token Server Facet	130
6.6.171	User Token – Issued Token Windows Server Facet	130
6.6.172	User Token – JWT Server Facet	131
6.6.173	User Token – User Name Password Client Facet	131
6.6.174	User Token – X509 Certificate Client Facet	131
6.6.175	User Token – Issued Token Client Facet	131
6.6.176	User Token – Issued Token Windows Client Facet	131
6.6.177	User Token – JWT Client Facet	132
6.6.178	Global Discovery Server Profile	132
6.6.179	Global Discovery Server 2017 Profile	132
6.6.180	Global Discovery and Certificate Management Server	132
6.6.181	Global Discovery and Certificate Mgmt 2017 Server	133
6.6.182	Global Certificate Management Client Profile	133
6.6.183	Global Certificate Management Client 2017 Profile	133
6.6.184	Global Service Authorization Request Server Facet	133
6.6.185	Global Service KeyCredential Pull Facet	134
6.6.186	Global Service KeyCredential Push Facet	134
	Bibliography	135
	Figure 1 – Profile – ConformanceUnit – TestCases	18
	Figure 2 – HMI Client sample	72
	Figure 3 – Embedded Server sample	72
	Figure 4 – Standard UA Server sample	73
	Table 1 – Profile Categories	19
	Table 2 – Conformance Groups	20
	Table 3 – Discovery Services	21
	Table 4 – Session Services	22
	Table 5 – Node Management Services	24
	Table 6 – View Services	24
	Table 7 – Attribute Services	25
	Table 8 – Method Services	26
	Table 9 – Monitored Item Services	27
	Table 10 – Subscription Services	29
	Table 11 – Security	31
	Table 12 – Protocol and Encoding	42
	Table 13 – Base Information	43
	Table 14 – Address Space Model	46
	Table 15 – Data Access	48
	Table 16 – Alarms and Conditions	49
	Table 17 – Historical Access	52

Table 18 – Aggregates	55
Table 19 – Auditing	61
Table 20 – Redundancy	61
Table 21 – Global Discovery Server	62
Table 22 – Miscellaneous	63
Table 23 – Profile list	65
Table 24 – Core 2017 Server Facet	74
Table 25 – Sessionless Server Facet	74
Table 26 – Reverse Connect Server Facet	75
Table 27 – Base Server Behaviour Facet	75
Table 28 – Request State Change Server Facet	75
Table 29 – Subnet Discovery Server Facet	75
Table 30 – Global Certificate Management Server Facet	76
Table 31 – Authorization Service Server Facet	76
Table 32 – KeyCredential Service Server Facet	76
Table 33 – Attribute WriteMask Server Facet	76
Table 34 – File Access Server Facet	77
Table 35 – Documentation Server Facet	77
Table 36 – Embedded DataChange Subscription Server Facet	77
Table 37 – Standard DataChange Subscription 2017 Server Facet	78
Table 38 – Enhanced DataChange Subscription 2017 Server Facet	79
Table 39 – Durable Subscription Server Facet	79
Table 40 – Data Access Server Facet	79
Table 41 – ComplexType 2017 Server Facet	80
Table 42 – Standard Event Subscription Server Facet	81
Table 43 – Address Space Notifier Server Facet	81
Table 44 – A & C Base Condition Server Facet	82
Table 45 – A & C Refresh2 Server Facet	82
Table 46 – A & C Address Space Instance Server Facet	82
Table 47 – A & C Enable Server Facet	82
Table 48 – A & C AlarmMetrics Server Facet	83
Table 49 – A & C Alarm Server Facet	83
Table 50 – A & C Acknowledgeable Alarm Server Facet	84
Table 51 – A & C Exclusive Alarming Server Facet	84
Table 52 – A & C Non-Exclusive Alarming Server Facet	84
Table 53 – A & C Previous Instances Server Facet	85
Table 54 – A & C Dialog Server Facet	85
Table 55 – A & C CertificateExpiration Server Facet	85
Table 56 – A & E Wrapper Facet	86
Table 57 – Method Server Facet	86
Table 58 – Auditing Server Facet	87
Table 59 – Node Management Server Facet	87
Table 60 – User Role Base Server Facet	87

Table 61 – User Role Management Server Facet	88
Table 62 – State Machine Server Facet	88
Table 63 – Client Redundancy Server Facet	88
Table 64 – Redundancy Transparent Server Facet.....	88
Table 65 – Redundancy Visible Server Facet.....	89
Table 66 – Historical Raw Data Server Facet.....	89
Table 67 – Historical Aggregate Server Facet.....	90
Table 68 – Historical Data AtTime Server Facet.....	91
Table 69 – Historical Access Modified Data Server Facet	91
Table 70 – Historical Annotation Server Facet	91
Table 71 – Historical Data Insert Server Facet.....	91
Table 72 – Historical Data Update Server Facet.....	92
Table 73 – Historical Data Replace Server Facet.....	92
Table 74 – Historical Data Delete Server Facet.....	92
Table 75 – Historical Access Structured Data Server Facet.....	92
Table 76 – Base Historical Event Server Facet	93
Table 77 – Historical Event Update Server Facet.....	93
Table 78 – Historical Event Replace Server Facet	93
Table 79 – Historical Event Insert Server Facet	93
Table 80 – Historical Event Delete Server Facet	93
Table 81 – Aggregate Subscription Server Facet	94
Table 82 – Nano Embedded Device 2017 Server Profile	95
Table 83 – Micro Embedded Device 2017 Server Profile.....	95
Table 84 – Embedded 2017 UA Server Profile	96
Table 85 – Standard 2017 UA Server Profile.....	97
Table 86 – Core 2017 Client Facet.....	97
Table 87 – Sessionless Client Facet	98
Table 88 – Reverse Connect Client Facet	98
Table 89 – Base Client Behaviour Facet	98
Table 90 – Discovery Client Facet.....	99
Table 91 – Subnet Discovery Client Facet.....	99
Table 92 – Global Discovery Client Facet	99
Table 93 – Global Certificate Management Client Facet.....	99
Table 94 – KeyCredential Service Client Facet	100
Table 95 – Access Token Request Client Facet	100
Table 96 – AddressSpace Lookup Client Facet.....	100
Table 97 – Request State Change Client Facet.....	100
Table 98 – File Access Client Facet	101
Table 99 – Entry Level Support 2015 Client Facet	101
Table 100 – Multi-Server Client Connection Facet	101
Table 101 – Documentation – Client	102
Table 102 – Attribute Read Client Facet.....	102
Table 103 – Attribute Write Client Facet.....	102

Table 104 – DataChange Subscriber Client Facet	103
Table 105 – Durable Subscription Client Facet	103
Table 106 – DataAccess Client Facet	104
Table 107 – Event Subscriber Client Facet	104
Table 108 – Base Event Processing Client Facet	105
Table 109 – Notifier and Source Hierarchy Client Facet	105
Table 110 – A & C Base Condition Client Facet	105
Table 111 – A & C Refresh2 Client Facet	106
Table 112 – A & C Address Space Instance Client Facet	106
Table 113 – A & C Enable Client Facet	106
Table 114 – A & C AlarmMetrics Client Facet	106
Table 115 – A & C Alarm Client Facet	107
Table 116 – A & C Exclusive Alarming Client Facet	107
Table 117 – A & C Non-Exclusive Alarming Client Facet	108
Table 118 – A & C Previous Instances Client Facet	108
Table 119 – A & C Dialog Client Facet	108
Table 120 – A & C CertificateExpiration Client Facet	108
Table 121 – A & E Proxy Facet	109
Table 122 – Method Client Facet	110
Table 123 – Auditing Client Facet	110
Table 124 – Node Management Client Facet	110
Table 125 – Advanced Type Programming Client Facet	111
Table 126 – User Role Management Client Facet	111
Table 127 – State Machine Client Facet	111
Table 128 – Diagnostic Client Facet	111
Table 129 – Redundant Client Facet	112
Table 130 – Redundancy Switch Client Facet	112
Table 131 – Historical Access Client Facet	112
Table 132 – Historical Data AtTime Client Facet	112
Table 133 – Historical Aggregate Client Facet	113
Table 134 – Historical Annotation Client Facet	114
Table 135 – Historical Access Modified Data Client Facet	114
Table 136 – Historical Data Insert Client Facet	114
Table 137 – Historical Data Update Client Facet	114
Table 138 – Historical Data Replace Client Facet	115
Table 139 – Historical Data Delete Client Facet	115
Table 140 – Historical Access Client Server Timestamp Facet	115
Table 141 – Historical Structured Data Access Client Facet	115
Table 142 – Historical Structured Data AtTime Client Facet	115
Table 143 – Historical Structured Data Modified Client Facet	116
Table 144 – Historical Structured Data Insert Client Facet	116
Table 145 – Historical Structured Data Update Client Facet	116
Table 146 – Historical Structured Data Replace Client Facet	116

Table 147 – Historical Structured Data Delete Client Facet	117
Table 148 – Historical Events Client Facet	117
Table 149 – Historical Event Insert Client Facet	117
Table 150 – Historical Event Update Client Facet	117
Table 151 – Historical Event Replace Client Facet	117
Table 152 – Historical Event Delete Client Facet	118
Table 153 – Aggregate Subscriber Client Facet	118
Table 154 – Standard UA Client 2017 Profile	120
Table 155 – UA-TCP UA-SC UA-Binary	120
Table 156 – HTTPS UA-Binary	120
Table 157 – HTTPS UA-XML	121
Table 158 – HTTPS UA-JSON	121
Table 159 – WSS UA-SC UA-Binary	122
Table 160 – WSS UA-JSON	122
Table 161 – Security User Access Control Full	123
Table 162 – Security User Access Control Base	123
Table 163 – Security Time Synchronization	123
Table 164 – Best Practice – Audit Events	123
Table 165 – Best Practice – Alarm Handling	124
Table 166 – Best Practice – Random Numbers	124
Table 167 – Best Practice – Timeouts	124
Table 168 – Best Practice – Administrative Access	124
Table 169 – Best Practice – Strict Message Handling	124
Table 170 – Best Practice – Audit Events Client	125
Table 171 – TransportSecurity – TLS 1.2	126
Table 172 – TransportSecurity – TLS 1.2 with PFS	126
Table 173 – SecurityPolicy – None	127
Table 174 – SecurityPolicy [A] – Aes128-Sha256-RsaOaep	128
Table 175 – SecurityPolicy [B] – Basic256Sha256	129
Table 176 – SecurityPolicy – Aes256-Sha256-RsaPss	129
Table 177 – User Token – Anonymous Facet	130
Table 178 – User Token – User Name Password Server Facet	130
Table 179 – User Token – X509 Certificate Server Facet	130
Table 180 – User Token – Issued Token Server Facet	130
Table 181 – User Token – Issued Token Windows Server Facet	130
Table 182 – User Token – JWT Server Facet	131
Table 183 – User Token – User Name Password Client Facet	131
Table 184 – User Token – X509 Certificate Client Facet	131
Table 185 – User Token – Issued Token Client Facet	131
Table 186 – User Token – Issued Token Windows Client Facet	132
Table 187 – User Token – JWT Client Facet	132
Table 188 – Global Discovery Server 2017 Profile	132
Table 189 – Global Discovery and Certificate Mgmt 2017 Server	133

Table 190 – Global Certificate Management Client 2017 Profile	133
Table 191 – Global Service Authorization Request Server Facet.....	133
Table 192 – Global Service KeyCredential Pull Facet	134
Table 193 – Global Service KeyCredential Push Facet.....	134

IECNORM.COM : Click to view the full PDF of IEC 62541-7:2020 RLV

INTERNATIONAL ELECTROTECHNICAL COMMISSION

OPC UNIFIED ARCHITECTURE –

Part 7: Profiles

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.

This redline version of the official IEC Standard allows the user to identify the changes made to the previous edition. A vertical bar appears in the margin wherever a change has been made. Additions are in green text, deletions are in strikethrough red text.

International Standard IEC 62541-7 has been prepared by subcommittee 65E: Devices and integration in enterprise systems, of IEC technical committee 65: Industrial-process measurement, control and automation.

This third edition cancels and replaces the second edition published in 2015. This edition constitutes a technical revision.

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

a) new functional Profiles:

- profiles for global discovery and global certificate management;
- profiles for global KeyCredential management and global access token management;
- facet for durable subscriptions;
- standard UA Client Profile;
- profiles for administration of user roles and permissions.

b) new transport Profiles:

- HTTPS with JSON encoding;
- secure WebSockets (WSS) with binary or JSON encoding;
- reverse connectivity.

c) new security Profiles:

- transportSecurity – TLS 1.2 with PFS (with perfect forward secrecy);
- securityPolicy [A] – Aes128-Sha256-RsaOaep (replaces Base128Rsa15);
- securityPolicy – Aes256-Sha256-RsaPss adds perfect forward secrecy for UA TCP;
- user Token JWT (Jason Web Token).

d) deprecated Security Profiles (due to broken algorithms):

- securityPolicy – Basic128Rsa15 (broken algorithm Sha1);
- securityPolicy – Basic256 (broken algorithm Sha1);
- transportSecurity – TLS 1.0 (broken algorithm RC4);
- transportSecurity – TLS 1.1 (broken algorithm RC4).

e) deprecated Transport (missing support on most platforms):

- SOAP/HTTP with WS-SecureConversation (all encodings).

The text of this International Standard is based on the following documents:

FDIS	Report on voting
65E/707/FDIS	65E/725/RVD

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

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

Throughout this document and the other parts of the IEC 62541 series, certain document conventions are used:

Italics are used to denote a defined term or definition that appears in the "Terms and definition" clause in one of the parts of the IEC 62541 series.

Italics are also used to denote the name of a service input or output parameter or the name of a structure or element of a structure that are usually defined in tables.

The *italicized terms and names* are, with a few exceptions, written in camel-case (the practice of writing compound words or phrases in which the elements are joined without spaces, with each element's initial letter capitalized within the compound). For example the defined term is *AddressSpace* instead of Address Space. This makes it easier to understand that there is a single definition for *AddressSpace*, not separate definitions for Address and Space.

A list of all parts of the IEC 62541 series, published under the general title *OPC Unified Architecture*, can be found on the IEC website.

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

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

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

OPC UNIFIED ARCHITECTURE –

Part 7: Profiles

1 Scope

This part of IEC 62541 ~~describes~~ defines the OPC Unified Architecture (OPC UA) *Profiles*. The *Profiles* in this document are used to segregate features with regard to testing of OPC UA products and the nature of the testing (tool based or lab based). This includes the testing performed by the OPC Foundation provided OPC UA CTT (a self-test tool) and by the OPC Foundation provided Independent certification test labs. This could equally as well refer to test tools provided by another organization or a test lab provided by another organization. What is important is the concept of automated tool-based testing versus lab-based testing. The scope of this standard includes defining functionality that can only be tested in a lab and defining the grouping of functionality that is to be used when testing OPC UA products either in a lab or using automated tools. The definition of actual *TestCases* is not within the scope of this document, but the general categories of *TestCases* are within the scope of this document.

Most OPC UA applications will conform to several, but not all, of the *Profiles*.

2 Normative references

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

IEC TR 62541-1, *OPC Unified Architecture – Part 1: Overview and Concepts*

IEC TR 62541-2, *OPC Unified Architecture – Part 2: Security Model*

IEC 62541-3, *OPC Unified Architecture – Part 3: Address Space Model*

IEC 62541-4, *OPC Unified Architecture – Part 4: Services*

IEC 62541-5, *OPC Unified Architecture – Part 5: Information Model*

IEC 62541-6, *OPC Unified Architecture – Part 6: Mappings*

IEC 62541-8, *OPC Unified Architecture – Part 8: Data Access*

IEC 62541-9, *OPC Unified Architecture – Part 9: Alarms and Conditions*

IEC 62541-11⁴, *OPC Unified Architecture – Part 11: Historical Access*

IEC 62541-12, *OPC Unified Architecture – Part 12: Discovery and Global Services*

IEC 62541-13⁴, *OPC Unified Architecture – Part 13: Aggregates*

⁴ ~~To be published.~~

Compliance Part 8 UA Server: OPC Test Lab Specification: Part 8 – UA Server
<http://www.opcfoundation.org/Test/Part8/>

Compliance Part 9 UA Client: OPC Test Lab Specification: Part 9 – UA Client
<http://www.opcfoundation.org/Test/Part9/>

3 Terms, definitions, and ~~conventions~~ abbreviated terms

3.1 Terms and definitions

For the purposes of this document, the terms and definitions given in IEC TR 62541-1, IEC TR 62541-2, IEC 62541-3, IEC 62541-4, IEC 62541-6, and IEC 62541-8 and the following apply.

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

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

NOTE An overview of the terms defined in this document and their interaction can be viewed in Figure 1.

3.1.1

application

software program that executes or implements some aspect of OPC UA

Note 1 to entry: The application could run on any machine and perform any function. The application could be software or it could be a hardware application, the only requirement is that it implements OPC UA.

3.1.2

ConformanceUnit

specific set of OPC UA features that can be tested as a single entity

Note 1 to entry: A ConformanceUnit can cover a group of services, portions of services or information models. ~~For additional detail see Clause 5.~~

3.1.3

ConformanceGroup

group of ConformanceUnits that is given a name

Note 1 to entry: This grouping is only to assist in organizing ConformanceUnits. Typical ConformanceGroups include groups for each of the service sets in OPC UA and each of the Information Model standards.

3.1.4

Facet

Profile dedicated to a specific feature that a Server or Client may require

Note 1 to entry: Facets are typically combined to form higher-level Profiles. The use of the term Facet in the title of a Profile indicates that the given Profile is not a standalone Profile.

3.1.5

FullFeatured Profile

Profile that defines all features necessary to build a functional OPC UA Application

Note 1 to entry: A FullFeatured Profile in particular adds definitions of the transport and security requirements.

3.1.6

ProfileCategory

category which arranges Profiles into application classes, such as Server or Client

Note 1 to entry: These categories help determine the type of Application that a given Profile would be used for. For additional details see 4.4.

3.1.7

TestCase

technical description of a set of steps required to test a particular function or information model

Note 1 to entry: TestCases provide sufficient details to allow a developer to implement them in code. TestCases also provide a detailed summary of the expected result(s) from the execution of the implemented code and any precondition(s) that ~~must~~ shall be established before the TestCase can be executed.

3.1.8

TestLab

facility that is designated to provide testing services

Note 1 to entry: These services include but are not limited to personal that directly perform testing, automated testing and a formal repeatable process. The OPC Foundation has provided detailed standard describing OPC UA TestLabs and the testing they are to provided (see Compliance Part 8 UA Server, Compliance Part 9 UA Client).

3.2 Abbreviated terms

DA	data access
HA	historical access
HMI	human machine interface
NIST	National Institute of Standard and Technology
PKI	Public Key Infrastructure
RSA	Rivest-Shamir-Adleman
UA	Unified Architecture

4 Overview

4.1 General

The IEC 62541 series, which is the OPC Unified architecture ~~multipart~~ standard, describes a number of *Services* and a variety of information models. These *Services* and information models can be referred to as features of a *Server* or *Client*. *Servers* and *Clients* need to be able to describe which features they support and wish to have certified. This document provides a grouping of these features. The individual features are grouped into *ConformanceUnits* which are further grouped into *Profiles*. Figure 1 provides an overview of the interactions between *Profiles*, *ConformanceUnits* and *TestCases*. The large arrows indicate the components that are used to construct the parent. For example a *Profile* is constructed from *Profiles* and *ConformanceUnits*. The figure also illustrates a feature of the OPC UA Compliance Test Tool (CTT), in that it will test if a requested *Profile* passes all *ConformanceUnits*. It will also test all other *ConformanceUnits* and report any other *Profiles* that pass conformance testing. The individual *TestCases* are defined in separate documents (see Compliance Part 8 UA Server and Compliance Part 9 UA Client). The *TestCases* are related back to the appropriate *ConformanceUnits* defined in this document. This relationship is also displayed by the OPC UA Compliance Test Tool.

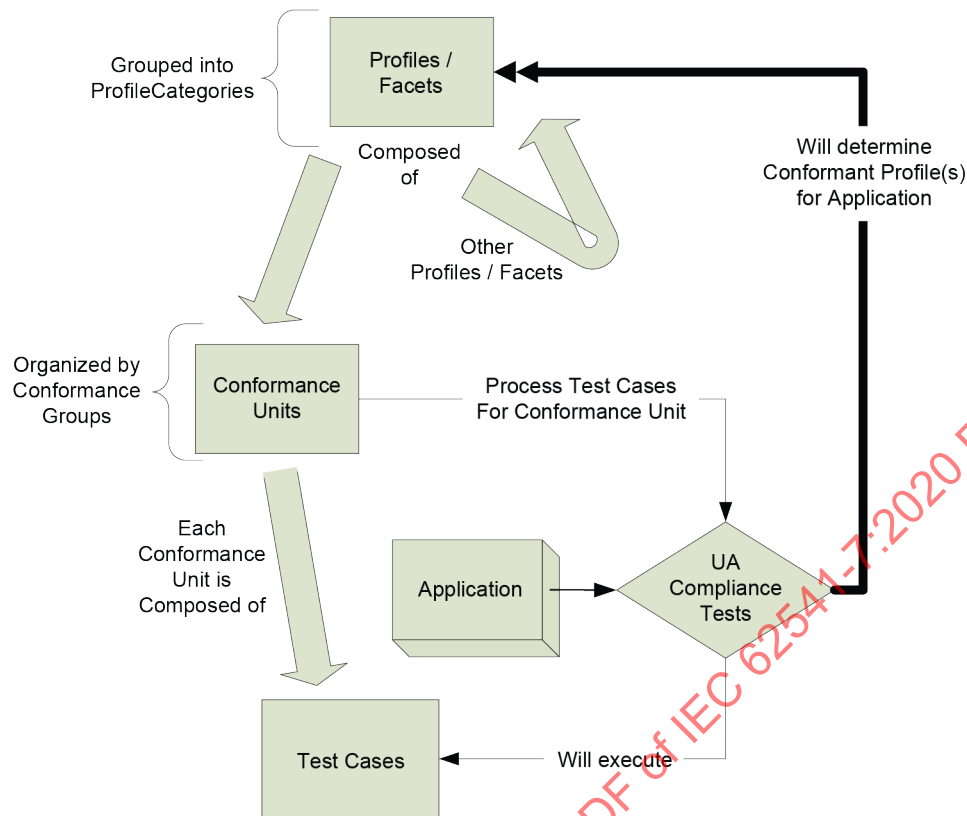


Figure 1 – Profile – ConformanceUnit – TestCases

4.2 ConformanceUnit

Each *ConformanceUnit* represents a specific set of features (e.g. a group of services, portions of services or information models) that can be tested as a single entity. *ConformanceUnits* are the building blocks of a *Profile*. Each *ConformanceUnit* can also be used as a test category. For each *ConformanceUnit*, there would be a number of *TestCases* that test the functionality described by the *ConformanceUnit*. The description of a *ConformanceUnit* is intended to provide enough information to illustrate the required functionality, but in many cases to obtain a complete understanding of the *ConformanceUnit* the reader may be required to also examine the appropriate part of IEC-62541 OPC UA. Additional Information regarding testing of a *ConformanceUnit* are provided in the Compliance Part 8 UA Server or Compliance Part 9 UA Client test standards.

The same features do not appear in more than one *ConformanceUnit*.

4.3 Profiles

A *Profile* is a named aggregation of *ConformanceUnits* and other *Profiles*. To support a *Profile*, an application has to support the *ConformanceUnits* and all aggregated *Profiles*. The definition of *Profiles* is an ongoing activity, in that it is expected that new *Profiles* will be added in the future.

An OPC UA Application will typically support multiple *Profiles*.

Multiple *Profiles* may include the same *ConformanceUnit*.

Testing of a *Profile* consists of testing the individual *ConformanceUnits* that comprise the *Profile*.

Profiles are named based on naming conventions (see 6.3 for details).

4.4 Profile Categories

Profiles are grouped into categories to help vendors and end users understand the applicability of a *Profile*. A *Profile* can be assigned to more than one category.

Table 1 contains the list of currently defined *ProfileCategories*.

Table 1 – Profile Categories

Category	Description
Client	Profiles of this category specify functions of an OPC UA Client. ~~The URI of such Profiles can be part of a Software Certificate passed in the ActivateSession request.~~
Global Directory Service	Profiles of this category specify functions for global discovery and certificate management.
Security	Profiles of this category specify security related functions. Security policies are part of this category. The URI of security policies has to be part of an Endpoint Description returned from the GetEndpoints service. Profiles of this category apply to Clients and Servers.
Server	Profiles of this category specify functions of an OPC UA Server. The URI of such Profiles can be ~~part of a Software Certificate returned with the CreateSession service response and~~ exposed in the Server capabilities.
Transport	Profiles of this category specify specific protocol mappings. The URI of such Profiles has to be part of an Endpoint Description. These Profiles apply to Clients and Servers.

5 Conformance Units

5.1 Overview

A *ConformanceUnit* represents an individually testable entity. For improved clarity, the large list of *ConformanceUnits* is arranged into named *ConformanceGroups*. These groups reflect the *Service Sets* in IEC 62541-4 and the OPC UA information models. Table 2 lists the *ConformanceGroups*. These groups and the *ConformanceUnits* that they describe are detailed in the subclauses of Clause 5, starting with 5.2. *ConformanceGroups* have no impact on testing; they are used only for organizational reasons, i.e. to simplify the readability of this document.

Table 2 – Conformance Groups

Group	Description
Address Space Model	Defines <i>ConformanceUnits</i> for various features of the OPC UA <i>AddressSpace</i> .
Aggregates	All <i>ConformanceUnits</i> that are related to Aggregates, including individual <i>ConformanceUnits</i> for each supported Aggregate as described in IEC 62541-13.
Alarms and Conditions	All <i>ConformanceUnits</i> that are associated with the OPC UA Information Model for <i>Conditions</i> , acknowledgeable <i>Conditions</i> , confirmations and <i>Alarms</i> as specified in IEC 62541-9.
Attribute Services	Includes <i>ConformanceUnits</i> to read or write current or historical <i>Attribute</i> values.
Auditing	User level security includes support for security audit trails, with traceability between <i>Client</i> and <i>Server</i> audit logs.
Base Information	All information elements as defined in IEC 62541-5.
Data Access	<i>ConformanceUnits</i> specific to <i>Clients</i> and <i>Servers</i> that deal with the representation and use of automation data as specified in IEC 62541-8.
Discovery Services	<i>ConformanceUnits</i> which focus on <i>Server</i> Endpoint <i>Discovery</i> .
GDS	<i>ConformanceUnits</i> for a GDS. Includes units for global discovery and global certificate management.
Historical Access	Access to archived data of <i>Node Attribute</i> values or Events.
Method Services	Methods represent the function calls of <i>Objects</i> . Methods are invoked and return only after completion (successful or unsuccessful).
Miscellaneous	This group contains <i>ConformanceUnits</i> that cover miscellaneous subjects, such as recommended behaviours, documentation etc. These <i>ConformanceUnits</i> typically do not fit into any of the other groups.
Monitored Item Services	<i>Clients</i> define <i>MonitoredItems</i> to subscribe to data and Events. Each <i>MonitoredItem</i> identifies the item to be monitored and the <i>Subscription</i> to use to send <i>Notifications</i> .
Node Management Services	Bundles <i>ConformanceUnits</i> for all <i>Services</i> to add and delete OPC UA <i>AddressSpace</i> <i>Nodes</i> and <i>References</i> .
Protocol and Encoding	Covers all transport and encoding combinations that are specified in IEC 62541-6.
Query Services	~~A Query may be used to provide advanced filtering and return a subset of data.~~
Redundancy	The design of OPC UA ensures that vendors can create redundant <i>Clients</i> and redundant <i>Servers</i> in a consistent manner. Redundancy may be used for high availability, fault tolerance and load balancing.
Security	Security related <i>ConformanceUnits</i> that can be profiled this includes all aspects of security.
Session Services	An (OPC UA) <i>Session</i> is an application layer connection.
Subscription Services	<i>Subscriptions</i> are used to report <i>Notifications</i> to the <i>Client</i> .
View Services	<i>Clients</i> use the View <i>Service</i> Set to navigate through the OPC UA <i>AddressSpace</i> or through a View (a subset) of the OPC UA <i>AddressSpace</i> .

5.2 Services

Table 3 to Table 10 describe *ConformanceUnits* for the *Services* specified in IEC 62541-4. The tables correlate with the *Service Sets*.

A single *ConformanceUnit* can reference several *Services* (e.g. *CreateSession*, *ActivateSession* and *CloseSession*), but can also refer to individual aspects of *Services* (e.g. the use of *ActivateSession* to impersonate a new user).

Each table includes a listing of the *Profile Category* to which a *ConformanceUnit* belongs, the title and description of the *ConformanceUnit* ~~and a column that indicates if the *ConformanceUnit* is derived from another *ConformanceUnit*. A *ConformanceUnit* that is derived from another *ConformanceUnit* includes all of the same tests as its parent plus one or more additional *TestCases*.~~ In some cases, a *ConformanceUnit* will be derived from another *ConformanceUnit*. This parent unit will then be specified in the description of each derived unit. In such cases, the derived units inherit all of the tests of its parent plus one or more additional *TestCases*. These *TestCases* can only further restrict the existing *TestCases*. An

example would be one in which the number of connections is tested, where the *TestCase* of the parent required at least one connection and the derived *ConformanceUnit* would require a *TestCase* for at least five connections.

The *Discovery Service Set* is composed of multiple *ConformanceUnits* (see Table 3). All *Servers* provide some aspects of this functionality; see *Profiles* categorized as *Server Profiles* for details. *Clients* may support some aspects of this functionality; see *Profiles* categorized as *Client Profiles* for details.

Table 3 – Discovery Services

Category	Title	Description	Derived
Server	Discovery Get Endpoints	Support the GetEndpoints <i>Service</i> to obtain all Endpoints of the <i>Server</i> . This includes filtering based on <i>Profiles</i> .	
Server	Discovery Get Endpoints SessionLess	Support at least one endpoint for issuing SessionLess <i>Services</i> . Support obtaining such endpoints by accepting the Transport URI as a filter to the GetEndpoints <i>Service</i> with the query string "SL" appended to the Transport URI. E.g. "http://opcfoundation.org/UA-Profile/Transport/https-uajson?SL"	
Server	Discovery Find Servers Self	Support the FindServers <i>Service</i> only for itself.	
Server	Discovery Register	Call the RegisterServer <i>Service</i> to register itself (OPC UA <i>Server</i>) with an external <i>Discovery Service</i> via a secure channel with a SecurityMode other than "None" NONE.	
Server	Discovery Register2	Call the RegisterServer2 <i>Service</i> to register with an external <i>Discovery Service</i> via a Secure Channel with a SecurityMode other than "None". This includes passing a list of short capability identifiers. The identifiers and their use are specified in IEC 62541-12.	
Server	Discovery Server Announcement using mDNS	Provide mDNS functionality to announce a <i>Server</i> with its capabilities. The capability identifiers and the use of mDNS records for the purpose of OPC UA <i>Discovery</i> is specified in IEC 62541-12. Note that this functionality is only required for <i>Servers</i> that do not register with an LDS. The capability identifiers and their use in mDNS records are specified in IEC 62541-12.	
Server	Discovery Configuration	Allow configuration of the <i>Discovery Server</i> URL where the <i>Server</i> will register itself. Allow complete disabling of registration with a <i>Discovery Server</i> .	
Client	Discovery Client Find Servers Basic	Uses the FindServers <i>Service</i> to obtain all <i>Servers</i> installed on a given platform.	
Client	Discovery Client Find Servers with URI	Use FindServers <i>Service</i> to obtain URLs for specific <i>Server</i> URIs.	
Client	Discovery Client Find Servers Dynamic	Detect new <i>Servers</i> after an initial FindServers <i>Service</i> call.	
Client	Discovery Client Find Servers on Network	Support one of the options to locate <i>Servers</i> on the network.	
Client	Discovery Client Find Servers on Network using LDS-ME	Use FindServersOnNetwork <i>Service</i> to obtain URLs for specific <i>Server</i> URIs. Note that this <i>Service</i> is available via the Local <i>Discovery Server</i> with multicast extension (LDS-ME).	

Category	Title	Description	Derived
Client	Discovery Client Find Servers on Network using mDNS	Use mDNS based <i>Service Discovery</i> to locate <i>Servers</i> on the same multicast network. The contents of mDNS records for OPC UA <i>Discovery</i> are described in IEC 62541-12. Note that this functionality is only required for <i>Clients</i> when there is no Local <i>Discovery Server</i> with multicast extension (LDS-ME). The capability identifiers and their use in mDNS records are specified in IEC 62541-12.	
Client	Discovery Client Find Servers in GDS	Use the <i>QueryServers Method</i> on the GDS Directory <i>Object</i> to locate <i>Servers</i> that meet filter criteria specified in the request. This <i>Method</i> is specified in IEC 62541-12.	
Client	Discovery Client Find Applications in GDS	Use the <i>QueryApplications Method</i> on the GDS Directory <i>Object</i> to locate Applications that meet filter criteria specified in the request. This <i>Method</i> is specified in IEC 62541-12.	
Client	Discovery Client Get Endpoints Basic	Uses the <i>GetEndpoints Service</i> to obtain all Endpoints for a given <i>Server URI</i> .	
Client	Discovery Client Get Endpoints SessionLess	Uses the <i>GetEndpoints Service</i> with a filter to obtain Endpoints that can be used for <i>SessionLess Service</i> invocation. The filter is the Transport URI extended with the query string "SL". E.g. http://opcfoundation.org/UA/Profile/Transport/https-uajson?SL .	
Client	Discovery Client Get Endpoints Dynamic	Detect changes to the Endpoints after an initial <i>GetEndpoints Service</i> call.	
Client	Discovery Client Configure Endpoint	Allow specification of an Endpoint without going through the <i>Discovery Service Set</i> .	

The *Session Service Set* is composed of multiple *ConformanceUnits* (see Table 4). The *CreateSession*, *ActivateSession*, and *CloseSession* services are supported as a single unit. All *Servers* and *Clients* provide this functionality.

Table 4 – Session Services

Category	Title	Description	Derived
Server	Session General Service Behaviour	Implement basic <i>Service</i> behaviour. This includes in particular: – checking the authentication token – returning the requestHandle in responses – returning available diagnostic information as requested with the 'returnDiagnostics' parameter – respecting a timeoutHint	
Server	Session Base	Support the <i>Session Service Set</i> (<i>CreateSession</i> , <i>ActivateSession</i> , <i>CloseSession</i>) except the use of <i>ActivateSession</i> to change the <i>Session</i> user. This includes correct handling of all parameters that are provided. Note that for the <i>CreateSession</i> and <i>ActivateSession</i> services, if the <i>SecurityMode</i> = None then: 1) The Application <i>Certificate</i> and Nonce are optional. 2) The signatures are null/empty. The details of this are described in IEC 62541-4.	
Server	Session Change User	Support the use of <i>ActivateSession</i> to change the <i>Session</i> user.	
Server	Session Cancel	Support the <i>Cancel Service</i> to cancel outstanding requests.	
Server	Session Minimum 1	Support minimum 1 <i>Session</i> (total).	
Server	Session Minimum 2 Parallel	Support minimum 2 parallel <i>Sessions</i> (total for all <i>Clients</i>).	

Category	Title	Description	Derived
Server	Session Minimum 50 Parallel	Support minimum 50 parallel Sessions (total for all <i>Clients</i>).	
Server	Session Sessionless Invocation	Defines the support of the SessionlessInvoke Service defined in IEC 62541-4 to process any of the Services (like Read/Write, Browse, or Call) that are designated for Sessionless Invocation.	
Client	Session Client General Service Behaviour	Implement basic Service behaviour. This includes in particular: – including the proper authentication token of the Session – creating a requestHandle if needed – requesting diagnostic information with the 'returnDiagnostics' parameter – evaluate the serviceResult and operational results	
Client	Session Client Base	Use the Session Service Set (CreateSession, ActivateSession, and CloseSession) except the use of ActivateSession to change the Session user. This includes correct handling of all parameters that are provided Note that for the CreateSession and ActivateSession services, if the SecurityMode = None then: 1) The Application Certificate and Nonce are optional. 2) The signatures are null/empty.	
Client	Session Client Multiple Connections	Support unlimited connections (client side) with multiple Servers. Any limit on numbers of connections is from server side. May have a memory-based limit, but not a software constraint limit.	
Client	Session Client Renew NodeIds	This ConformanceUnit applies to Clients that allow persisting NodeIds. Verify that the Namespace Table has not changed for NodeIds that the Client has persisted and is going to re-use beyond a Session lifetime. If changes occurred the Client has to recalculate the Namespace Indices of the respective NodeIds.	
Client	Session Client Impersonate	Uses ActivateSession to change the Session user (impersonation).	
Client	Session Client KeepAlive	Make periodic requests to keep the Session alive.	
Client	Session Client Detect Shutdown	Read or monitor the ServerStatus/State Variable to recognize a potential shutdown of the Server and clean up resources.	
Client	Session Client Cancel	Use the Cancel Service to cancel outstanding requests.	
Client	Session Client Auto Reconnect	Automatic Client reconnect including: – ActivateSession with new SecureChannel if SecureChannel is no longer valid but Session is still valid – Creation of a new Session only if Session is no longer valid	
Client	~~Client Entry-Level Support~~	~~The Client is able to interoperate with Servers with lowest level functionality. This includes the ability to operate with a single Session, a pre-knowledge of the OPC-UA Types (the Server may not expose them in the AddressSpace), and the ability to use Read vs. Subscriptions for monitoring.~~ ~~There may be further restrictions provided by the Server via the Server capabilities.~~	
Client	Session Client Single Session	The Client shall interoperate with Servers that only support one Session.	
Client	Session Client SessionLess Service Calls	Defines the use of the SessionlessInvoke Service defined in IEC 62541-4 to request one of the Services (like Read or Browse) that are allowed for Sessionless Invocation. IEC 62541-6 specifies which transports may be used and how.	

The Node Management Service Set is composed of multiple *ConformanceUnits* (see Table 5). Servers may provide some aspects of this functionality; see *Profiles* categorized as *Server Profiles* for details. Clients may support some aspects of this functionality; see *Profiles* categorized as *Client Profiles* for details.

Table 5 – Node Management Services

Category	Title	Description	Derived
Server	Node Management Add Node	Support the AddNodes <i>Service</i> to add one or more <i>Nodes</i> into the OPC UA <i>AddressSpace</i> .	
Server	Node Management Delete Node	Support the DeleteNodes <i>Service</i> to delete one or more <i>Nodes</i> from the OPC UA <i>AddressSpace</i> .	
Server	Node Management Add Ref	Support the AddReferences <i>Service</i> to add one or more <i>References</i> to one or more <i>Nodes</i> in the OPC UA <i>AddressSpace</i> .	
Server	Node Management Delete Ref	Support the DeleteReferences <i>Service</i> to delete one or more <i>References</i> of a <i>Node</i> in the OPC UA <i>AddressSpace</i> .	
Client	Node Management Client	Uses <i>Node Management Services</i> to add or delete <i>Nodes</i> and to add or delete <i>References</i> in <i>Server's</i> OPC UA <i>AddressSpace</i> .	

The View *Service Set* is composed of a multiple *ConformanceUnits* (see Table 6). All *Servers* support some aspects of this conformance group. *Clients* may support some aspects of this functionality; see *Profiles* categorized as *Client Profiles* for details.

Table 6 – View Services

Category	Title	Description	Derived
Server	View Basic	Support the View <i>Service Set</i> (Browse, BrowseNext).	
Server	View TranslateBrowsePath	Support TranslateBrowsePathsToNodeIds <i>Service</i> .	
Server	View RegisterNodes	Support the RegisterNodes and UnregisterNodes <i>Services</i> as a way to optimize access to repeatedly used <i>Nodes</i> in the <i>Server's</i> OPC UA <i>AddressSpace</i> .	
Server	View Minimum Continuation Point 01	Support minimum 1 continuation point per <i>Session</i> .	
Server	View Minimum Continuation Point 05	Support minimum 5 continuation points per <i>Session</i> . This number has to be supported for at least half of the minimum required sessions.	
Client	View Client Basic Browse	Uses Browse and BrowseNext <i>Services</i> to navigate through the <i>Server's</i> OPC UA <i>AddressSpace</i> . Make use of the referenceTypeId and the nodeClassMask to specify the needed <i>References</i> .	
Client	View Client Remote Nodes Browse	The <i>Client</i> can browse to nodes that have an extended NodeId that reference a <i>Server</i> different than the originating <i>Server</i> . This includes automatic connection to the remote <i>Server</i> . It is acceptable that the <i>Server</i> configuration information be pre-configured on the <i>Client</i> and / or that the user is prompted to connect.	
Client	View Client Basic ResultSet Filtering	Makes use of the resultMask parameter to optimize the result set to be returned by the <i>Server</i> .	
Client	View Client TranslateBrowsePath	Uses the TranslateBrowsePathsToNodeIds <i>Service</i> to identify the NodeIds for <i>Nodes</i> where a starting <i>Node</i> and a BrowsePath is known. Makes use of bulk operations rather than multiple calls whenever possible.	
Client	View Client Remote Nodes Translate Browse	The <i>Client</i> can translate browse paths that include nodes with extended NodeId that reference a <i>Server</i> different than the originating <i>Server</i> and return them as part of the TranslateBrowsePathsToNodeIds <i>Service</i> . It is acceptable that the <i>Server</i> configuration information be pre-configured on the <i>Client</i> .	
Client	View Client RegisterNodes	Uses the RegisterNodes <i>Service</i> to optimize access for <i>Nodes</i> that are used repeatedly. Use UnregisterNodes when <i>Nodes</i> are not used anymore.	

The *Attribute Service Set* is composed of multiple *ConformanceUnits* (see Table 7). The majority of the *Attribute* service set is a core functionality of OPC UA and as such is supported by most *Servers*. Most *Clients* will also support some aspects of the *Attribute Service Set*.

Table 7 – Attribute Services

Category	Title	Description	Derived
Server	Attribute Read	Supports the Read <i>Service</i> to read one or more <i>Attributes</i> of one or more <i>Nodes</i> . This includes support of the <i>IndexRange</i> parameter to read a single element or a range of elements when the <i>Attribute</i> value is an array.	
Server	Attribute Read Complex	Supports reading and encoding Values with structured <i>DataTypes</i> .	
Server	Attribute Write Values	Supports writing to values to one or more <i>Attributes</i> of one or more <i>Nodes</i> .	
Server	Attribute Write Complex	Supports writing and decoding Values with structured <i>DataTypes</i> .	
Server	Attribute Write StatusCode & Timestamp	Supports writing of <i>StatusCode</i> and <i>Timestamps</i> along with the Value.	
Server	Attribute Write Index	Supports the <i>IndexRange</i> to write a single element or a range of elements when the <i>Attribute</i> value is an array and partial updates are allowed for this array.	
Server	Attribute Alternate Encoding	Supports alternate Data Encoding when reading value <i>Attributes</i> . By default, every <i>Server</i> has to support the Data Encoding of the currently used <i>Stack Profile</i> (i.e. binary with UA Binary Encoding and XML with XML Encoding). This <i>ConformanceUnit</i> – when supported – specifies that the other Data Encoding is supported in addition.	
Server	Attribute Historical Read	Supports the <i>HistoryRead Service</i> . The details of what aspects of this service are used are listed in additional <i>ConformanceUnits</i> , but at least one of <i>ReadRaw</i> , <i>ReadProcessed</i> , <i>ReadModified</i> , <i>ReadAtTime</i> or <i>ReadEvents</i> ~~must~~ shall be supported.	
Server	Attribute Historical Update	Supports the <i>HistoryUpdate service</i> . The details of the supported features of this service are described by additional <i>ConformanceUnits</i> , but at least one of the following ~~must~~ shall be supported: <i>InsertData</i> , <i>InsertEvents</i> , <i>ReplaceData</i> , <i>ReplaceEvents</i> , <i>UpdateData</i> , <i>UpdateEvents</i> , <i>DeleteData</i> , <i>DeleteEvents</i> or <i>DeleteAtTime</i> .	
Client	Attribute Client Read Base	Use the Read <i>Service</i> to read one or more <i>Attributes</i> of one or more <i>Nodes</i> . This includes use of an <i>IndexRange</i> to select a single element or a range of elements when the <i>Attribute</i> value is an array. <i>Clients</i> shall use bulk operations whenever possible to reduce the number of <i>Service</i> invocations.	
Client	Attribute Client Remote Nodes Attribute Access	The <i>Client</i> can retrieve attributes of nodes that have an extended <i>NodeID</i> that reference a <i>Server</i> different than the originating <i>Server</i> . This requires a connection to the remote <i>Server</i> for access (not necessarily displayed as a connection). It is acceptable that the <i>Server</i> configuration information be pre-configured on the <i>Client</i> .	
Client	Attribute Client Read with proper Encoding	This <i>ConformanceUnit</i> refers to the ability of a ~~Server Client~~ to ~~support more than one Data Encoding for Attribute values.~~ <i>Clients</i> can discover the available encodings and ~~can explicitly~~ choose a specific one when calling the Read <i>Service</i> .	
Client	Attribute Client Read Complex	Read and decode Values with structured <i>DataTypes</i> .	

Category	Title	Description	Derived
Client	Attribute Client Write Base	Use the Write Service to write values to one or more Attributes of one or more Nodes. This includes use of an IndexRange to select a single element or a range of elements when the Attribute value is an array. Clients shall use bulk operations whenever possible to reduce the number of Service invocations.	
Client	Attribute Client Write Complex	Write and Encode Values with structured DataTypes.	
Client	Attribute Client Write Quality & Timestamp	Use the Write Service to also write StatusCode and/or Timestamps along with a Value.	
Client	Attribute Client Historical Read	The Client makes use of the HistoryRead service. The details of which aspect of this service are used are provided by additional ConformanceUnits, but at least one or more of the following is used ReadRaw, ReadAtTime, ReadProcessed, ReadModified or ReadEvents.	
Client	Attribute Client Historical Updates	The Client makes use of the HistoryUpdate service. The details of this usage are provided by additional ConformanceUnits, but at least one or more of the following must shall be provided InsertData, InsertEvents, ReplaceData, ReplaceEvents, UpdateData, UpdateEvents, DeleteData or DeleteEvents or DeleteAtTime.	

The *Method Service Set* is composed of *ConformanceUnits* (see Table 8). The primary *ConformanceUnits* provide support for the call functionality. Servers may provide some aspects of this functionality; see *Profiles* categorized as *Server Profiles* for details. Clients may support some aspects of this functionality; see *Profiles* categorized as *Client Profiles* for details.

Table 8 – Method Services

Category	Title	Description	Derived
Server	Method Call	Support the Call Service to call (invoke) a Method which includes support for Method Parameters.	
Client	Method Client Call	Use the Call Service to call one or several Methods.	

The *MonitoredItem Service Set* is composed of multiple *ConformanceUnits* (see Table 9). Servers may provide some aspects of this functionality; see *Profiles* categorized as *Server Profiles* for details. Clients may support some aspects of this functionality; see *Profiles* categorized as *Client Profiles* for details.

Table 9 – Monitored Item Services

Category	Title	Description	Derived
Server	Monitor Basic	Support the following <i>MonitoredItem</i> Services: CreateMonitoredItems, ModifyMonitoredItems, DeleteMonitoredItems and SetMonitoringMode.	
Server	Monitor Value Change	Support creation of <i>MonitoredItems</i> for <i>Attribute</i> value changes. This includes support of the <i>IndexRange</i> to select a single element or a range of elements when the <i>Attribute</i> value is an array.	
Server	Monitor Complex Value	Supports monitoring and encoding Values with structured <i>DataTypes</i> .	
Server	Monitored Items Deadband Filter	Supports an absolute Deadband filter as a <i>DataChangeFilter</i> for numeric data types.	
Server	Monitor Aggregate Filter	Support for Aggregate filters for <i>MonitoredItems</i> . The result of this <i>ConformanceUnit</i> includes a list of Aggregates that are supported as part of the <i>Profile Certificate</i> .	
Server	Monitor Alternate Encoding	Support alternate encoding when monitoring value <i>Attributes</i> . By default, every <i>Server</i> has to support the encoding of the currently used <i>Stack Profile</i> (i.e. binary with UA Binary Encoding and XML with XML Encoding). This <i>ConformanceUnit</i> – when supported – specifies that the other encoding is supported in addition.	
Server	Monitor Items 2	Support at least 2 <i>MonitoredItems</i> per <i>Subscription</i> where the size of each <i>MonitoredItem</i> is at least equal to size of <i>Double</i> .	
Server	Monitor Items 10	Support at least 10 <i>MonitoredItems</i> per <i>Subscription</i> where the size of each <i>MonitoredItem</i> is at least equal to size of <i>Double</i> .	
Server	Monitor Items 100	Support at least 100 <i>MonitoredItems</i> per <i>Subscription</i> . This number has to be supported for at least half of the required <i>Subscriptions</i> for half of the required <i>Sessions</i> .	
Server	Monitor Items 500	Support at least 500 <i>MonitoredItems</i> per <i>Subscription</i> . This number has to be supported for at least half of the required <i>Subscriptions</i> for half of the required <i>Sessions</i> .	
Server	Monitor QueueSize_1	This <i>ConformanceUnit</i> does not require queuing when multiple value changes occur during a "publish period". I.e. the latest change will be sent in the <i>Notification</i> .	
Server	Monitor MinQueueSize_02	Support at least 2 queue entries for <i>MonitoredItems</i> . <i>Servers</i> often will adapt the queue size to the number of currently <i>MonitoredItems</i> . However, it is expected that <i>Servers</i> support this minimum queue size for at least one third of the supported <i>MonitoredItems</i> .	
Server	Monitor MinQueueSize_05	Support at least 5 queue entries for <i>MonitoredItems</i> . <i>Servers</i> often will adapt the queue size to the number of currently <i>MonitoredItems</i> . However, it is expected that <i>Servers</i> support this minimum queue size for at least one third of the supported <i>MonitoredItems</i> .	
Server	Monitor QueueSize_ServerMax	This <i>ConformanceUnit</i> is for events. When the <i>Client</i> requests <i>queuesize</i> =MAXUInt32 the <i>Server</i> is to return the maximum queue size that it can support for event notifications as the revisedQueueSize.	
Server	Monitor Triggering	Support the SetTriggering <i>Service</i> to create and/or delete triggering links for a triggering item.	

Category	Title	Description	Derived
Server	Monitor Events	Support creation of <i>MonitoredItems</i> for an "EventNotifier Attribute" for the purpose of <i>Event Notification</i> . The subscription includes supporting a filter that includes SimpleAttribute Operands and a select list of Operators. The list of Operators includes: Equals, IsNull, GreaterThan, LessThan, GreaterThanorEqual, LessThanorEqual, Like, Not, Between, InList, And, Or, Cast, BitwiseAnd, BitwiseOr.	
Server	Monitor Complex Event Filter	~~Support for complex Event filters, where complex is defined as supporting the complex filter operator (TypeOf).~~ Support for the 'TypeOf' complex Event filter operator.	
Client	Monitor Client Value Change	Use the <i>MonitoredItem Service Set</i> to register items for changes in <i>Attribute</i> value. Use <i>CreateMonitoredItems</i> to register the <i>Node/Attribute</i> tuple. Set proper sampling interval, Deadband filter and queuing mode. Use disabling / enabling instead of deleting and re-creating a <i>MonitoredItem</i> . Use bulk operations rather than individual service requests to reduce communication overhead.	
Client	Monitor Client Complex Value	Monitor and decode Values with structured DataTypes.	
Client	Monitor Client Deadband Filter	Uses Absolute Deadband filters for subscriptions.	
Client	Monitor Client by Index	Use the <i>IndexRange</i> to select a single element or a range of elements when the <i>Attribute</i> value is an array.	
Client	Monitor Client Aggregate Filter	Uses Aggregate filters for Subscriptions.	
Client	Monitor Client Events	Use the <i>MonitoredItem Service Set</i> to create <i>MonitoredItems</i> for Event notifications.	
Client	Monitor Client Event Filter	Use the <i>Event</i> filter when calling <i>CreateMonitoredItems</i> to filter the desired Events and to select the columns to be provided for each <i>Event Notification</i> .	
Client	Monitor Client Complex Event Filter	Uses of the 'TypeOf' complex Event filters operator.	
Client	Monitor Client Modify	Use <i>ModifyMonitoredItems Service</i> to change the configuration setting. Use <i>SetMonitoringMode Service</i> to disable/enable sampling and/or publishing.	
Client	Monitor Client Trigger	Use the <i>Triggering Model</i> if certain items are to be reported only if some other item triggers. Use proper monitoring mode for these items. Use <i>SetTriggering Service</i> to link these items to the trigger item.	

The *Subscription Service Set* is composed of multiple *ConformanceUnits* (see Table 10). *Servers* may provide some aspects of this functionality; see *Profiles* categorized as *Server Profiles* for details. *Clients* may support some aspects of this functionality; see *Profiles* categorized as *Client Profiles* for details.

Table 10 – Subscription Services

Category	Title	Description	Derived
Server	Subscription Basic	Support the following <i>Subscription Services</i> : CreateSubscription, ModifySubscription, DeleteSubscriptions, Publish, Republish and SetPublishingMode.	
Server	Subscription Minimum 1	Support at least 1 <i>Subscription</i> per <i>Session</i> . This number has to be supported for all of the minimum required sessions.	
Server	Subscription Minimum 02	Support at least 2 Subscriptions per <i>Session</i> . This number has to be supported for at least half of the minimum required sessions.	
Server	Subscription Minimum 05	Support at least 5 Subscriptions per <i>Session</i> . This number has to be supported for at least half of the minimum required sessions.	
Server	Subscription Publish Min 02	Support at least 2 Publish <i>Service</i> requests per <i>Session</i> . ~~This number has to be supported for all of the minimum required sessions. Support of republish is optional and no notification retransmission queue has to be provided however the republish service must be provided and will return the appropriate operation level results.~~ This number has to be supported for all of the minimum required sessions. Support of a NotificationMessage retransmission queue is not required; if not available, the Republish <i>Service</i> returns Bad_MessageNotAvailable.	
Server	Subscription Publish Min 05	Support at least 5 Publish <i>Service</i> requests per <i>Session</i> . This number has to be supported for at least half of the minimum required sessions. Support, as a minimum, the number of Publish requests per session as the size of the NotificationMessage retransmission queue for Republish.	
Server	Subscription Publish Min 10	Support at least 10 Publish <i>Service</i> requests per <i>Session</i> . This number has to be supported for at least half of the minimum required sessions. Support, as a minimum, the number of Publish requests per session as the size of the NotificationMessage retransmission queue for Republish.	
Server	Subscription Publish Discard Policy	Respect the specified policy for discarding Publish <i>Service</i> requests. If the maximum number of Publish <i>Service</i> requests has been queued and a new Publish <i>Service</i> request arrives, the "oldest" Publish request has to be discarded by returning the proper error.	
Server	Subscription Transfer	Support TransferSubscriptions <i>Service</i> to transfer a <i>Subscription</i> from one <i>Session</i> to another.	
Server	Subscription Durable	Support setting Subscriptions in durable mode. This mode requires that collected data and events are stored and delivered even if a <i>Client</i> was disconnected for a longer time or the <i>Server</i> was restarted. Support one of the " <i>Subscription Durable StorageLevel nnn</i> " <i>ConformanceUnits</i> .	
Server	Subscription Durable StorageLevel Small	Support at least 20 monitored items with a queue size of 10 000 for each item and where the size of each <i>MonitoredItem</i> is at least equal to size of Double. This requires storage capacity for 200 000 values of <i>DataType</i> Double.	
Server	Subscription Durable StorageLevel Medium	Support at least 100 monitored items with a queue size of 50 000 for each item and where the size of each <i>MonitoredItem</i> is at least equal to size of Double. This requires storage capacity for 5 million values of <i>DataType</i> Double.	

Category	Title	Description	Derived
Server	Subscription Durable StorageLevel High	Support at least 2 000 monitored items with a queue size of 200 000 for each item and where the size of each <i>MonitoredItem</i> is at least equal to size of Double. This requires storage capacity for 400 million values of <i>DataType</i> Double.	
Client	Subscription Client Basic	Use the <i>Subscription</i> and <i>MonitoredItem</i> Service Set as an efficient means to detect changes of <i>Attribute</i> values and / or to receive <i>Event</i> occurrences. Set appropriate intervals for publishing, keep alive notifications and total <i>Subscription</i> lifetime. Supply a sufficient number of Publish requests to the <i>Server</i> so that <i>Notifications</i> can be sent whenever a publish timer expires. Acknowledge received <i>Notifications</i> with subsequent Publish requests.	
Client	Subscription Client Fallback	The <i>Client</i> shall interoperate with <i>Servers</i> that do not support <i>Subscriptions</i> , or have exhausted <i>Subscription</i> limits, for Monitoring by using Read <i>Service</i> .	
Client	Subscription Client Republish	Evaluate the sequence number in <i>Notifications</i> to detect lost <i>Notifications</i> . Use Republish to request missing <i>Notifications</i> .	
Client	Subscription Client Modify	Allow modification of the <i>Subscription</i> configuration using the ModifySubscription <i>Service</i> .	
Client	Subscription Client TransferSubscriptions	The <i>Client</i> supports transferring <i>Subscription</i> from other <i>Clients</i> . This <i>ConformanceUnit</i> is used as part of redundant <i>Clients</i> .	
Client	Subscription Client Multiple	Use multiple <i>Subscriptions</i> to reduce the payload of individual <i>Notifications</i> .	
Client	Subscription Client Publish Configurable	Send multiple Publish <i>Service</i> requests to assure that the <i>Server</i> is always able to send <i>Notifications</i> . The number of parallel Publish <i>Service</i> requests per <i>Session</i> shall be configurable.	
Client	Subscription Client Durable	Use durable <i>Subscriptions</i> .	

5.3 Transport and communication related features

Table 11 describes security related *ConformanceUnits*. All of these *ConformanceUnits* apply equally to both *Clients* and *Servers*, where a *Client* uses the related security unit and a *Server* supports the use of it. These items are defined in detail in IEC 62541-6. It is recommended that a *Server* and *Client* support as many of these options as possible in order to achieve increased levels of interoperability. It is the task of an administrator to determine which of these *ConformanceUnits* are exposed in a given deployed *Server* or *Client* application.

Table 11 – Security

Category	Title	Description	Derived
Security	Security-Certificate-Validation	A certificate will be validated as specified in IEC 62541-4. This includes among others structure and signature examination. Allowing for some validation errors to be suppressed by administration directive.	
Security	Security-None	A suite of algorithms that does NOT provide any security settings: -> SymmetricSignatureAlgorithm – Not Used -> SymmetricEncryptionAlgorithm – Not Used -> AsymmetricSignatureAlgorithm – Not Used -> SymmetricKeyWrapAlgorithm – Not Used -> AsymmetricEncryptionAlgorithm – Not Used -> KeyDerivationAlgorithm – Not Used -> DerivedSignatureKeyLength – 0 The use of this suite of algorithms must be able to be enabled or disabled by an administrator.	
Security	Security-None-CreateSession-ActivateSession	When SecurityPolicy=None, the CreateSession and ActivateSession service allow for a NULL/empty signature and do not require Application Certificates or a Nonce.	
Security	Security-Basic-128Rsa15	A suite of algorithms that uses RSA15 as Key-Wrap algorithm and 128-Bit encryption algorithms: -> SymmetricSignatureAlgorithm – HmacSha1 – (http://www.w3.org/2000/09/xmldsig#hmac-sha1). -> SymmetricEncryptionAlgorithm – Aes128 – (http://www.w3.org/2001/04/xmenc#aes128-ecb). -> AsymmetricSignatureAlgorithm – RsaSha1 – (http://www.w3.org/2000/09/xmldsig#rsa-sha1). -> AsymmetricKeyWrapAlgorithm – KwRsa15 – (http://www.w3.org/2001/04/xmenc#rsa-1_5). -> AsymmetricEncryptionAlgorithm – Rsa15 – (http://www.w3.org/2001/04/xmenc#rsa-1_5). -> KeyDerivationAlgorithm – PSha1 – (http://docs.oasis-open.org/ws-sx/ws-secureconversation/200512/dk/p_sha1). -> DerivedSignatureKeyLength – 128. -> MinAsymmetricKeyLength – 1024 -> MaxAsymmetricKeyLength – 2048 -> CertificateSignatureAlgorithm – Sha1	
Security	Security-Basic-256	A suite of algorithms that are for 256-Bit encryption, algorithms include: -> SymmetricSignatureAlgorithm – HmacSha1 – (http://www.w3.org/2000/09/xmldsig#hmac-sha1). -> SymmetricEncryptionAlgorithm – Aes256 – (http://www.w3.org/2001/04/xmenc#aes256-ecb). -> AsymmetricSignatureAlgorithm – RsaSha1 – (http://www.w3.org/2000/09/xmldsig#rsa-sha1). -> AsymmetricKeyWrapAlgorithm – KwRsaOaep – (http://www.w3.org/2001/04/xmenc#rsa-oaep-mgf1p). -> AsymmetricEncryptionAlgorithm – RsaOaep – (http://www.w3.org/2001/04/xmenc#rsa-oaep). -> KeyDerivationAlgorithm – PSha1 – (http://docs.oasis-open.org/ws-sx/ws-secureconversation/200512/dk/p_sha1). -> DerivedSignatureKeyLength – 192. -> MinAsymmetricKeyLength – 1024 -> MaxAsymmetricKeyLength – 2048 -> CertificateSignatureAlgorithm – Sha1	

Category	Title	Description	Derived
Security	Security Basic-256-Sha256	A suite of algorithms that are for 256-Bit encryption, algorithms include: -> SymmetricSignatureAlgorithm - Hmac_Sha256 (http://www.w3.org/2000/09/xmldsig#hmac-sha256). -> SymmetricEncryptionAlgorithm - Aes256_CBC (http://www.w3.org/2001/04/xmenc#aes256-ecb). -> AsymmetricSignatureAlgorithm - Rsa_Sha256 (http://www.w3.org/2000/09/xmldsig#rsa-sha256). -> AsymmetricKeyWrapAlgorithm — KwRsaOaep (http://www.w3.org/2001/04/xmenc#rsa-oaep-mgf1p). -> AsymmetricEncryptionAlgorithm — Rsa_Oaep (http://www.w3.org/2001/04/xmenc#rsa-oaep). -> KeyDerivationAlgorithm — PSHA256 (http://docs.oasis-open.org/ws-sx/ws-secureconversation/200512/dk/p_sha256). -> DerivedSignatureKeyLength — 256 -> MinAsymmetricKeyLength — 2048 -> MaxAsymmetricKeyLength — 4096 -> CertificateSignatureAlgorithm — Sha256 Support for this security profile may require support for a second application instance certificate, with a larger keysize. Applications shall support multiple Application Instance Certificates if required by supported Security Policies and use the certificate that is required for a given security endpoint.	
Security	Security TLS-General	This <i>ConformanceUnit</i> indicates that at least one of the transport security <i>Profiles</i> for TLS is supported by this application. It is used in TLS transport <i>Profiles</i> , but the choice of transport security profile is optional. The actual used security profile will default to the most secure one.	
Security	Security TLS-1.1	The connection is established using TLS 1.1. The application needs to be configured to prevent TLS 1.0 connections, unless the TLS 1.0 connection is using TLS_RSA_WITH_RC4_128_SHA as described in <i>ConformanceUnit</i> "Security TLS_RSA_WITH_RC4_128_SHA".	
Security	Security TLS_RSA_WITH_RC4_128_SHA	The connection is established using TLS_RSA_WITH_RC4_128_SHA. The application needs to be configured to prevent the use of AES-based protocol suites (TLS 1.0).	
Security	Security TLS_RSA_WITH_AES_256_CBC_SHA256	The connection is established using TLS_RSA_WITH_AES_256_CBC_SHA256. That has a MinAsymmetricKeyLength — 2048, MaxAsymmetricKeyLength — 4096, AsymmetricSignatureAlgorithm — RSA_SHA256. (TLS 1.2)	
Security	Security Encryption-Required	Encryption is required using the algorithms provide in the security algorithm suite.	
Security	Security Signing-Required	Signing is required using the algorithms provide in the security algorithm suite.	
Security	Security Time-Synch-Configuration	Application supports configuring acceptable clock skew.	
Security	Security Time-Synch-NTP / OS Based support	Application supports time synchronization, either via an implementation of Network Time Protocol (NTP), or via features of a standard operating system.	
Security	Security Time-Synch-UA based support	An application makes use of the responses header timestamp provided by a configured well know source, such as a <i>Discovery Server</i> to synchronize the time on the application and that this time synchronization occurs periodically. Use of this TimeSyncing can be configured.	

Category	Title	Description	Derived
Security	Security Administration	Allow configuration of the following Security related items: * select the allowed User identification policy or policies (User Name/Password or X509 or Kerberos or Anonymous). * enable/disable the security policy "None" or other security policies. * enable/disable endpoints with MessageSecurityMode SIGN or SIGNANDENCRYPT. * set the permitted certification authorities. * define how to react to unknown Certificates.	
Security	Security Administration XML Schema	Support the OPC UA defined XML schema for importing and exporting security configuration information. This schema is defined in IEC 62541-6.	
Security	Security Certificate Administration	Allow a site administrator to be able to assign a site specific ApplicationInstanceCertificate and if desired to configure a site specific Certificate Authority (CA).	
Security	Security Default ApplicationInstanceCertificate	An application, when installed, has a default ApplicationInstanceCertificate that is valid. The default ApplicationInstanceCertificate shall either be created as part of the installation or installation instructions explicitly describe the process to create and apply a default ApplicationInstanceCertificate to the application.	
Security	Security No Application Authentication	The Server supports being able to be configured for no application authentication, just User authentication and normal encryption/signing: - Configure server to accept all certificates - Certificates are just used for message security (signing and encryption) - Users level is used for authentication	
Security	Best Practice Audit Events	Subscriptions for Audit Events are restricted to authorized personnel. A Server may also reject a Subscription for Audit Events that is not over a Secure Channel if one is available.	
Security	Best Practice Alarm Handling	A Server should restrict critical alarm functionality to users that have the appropriate rights to perform these actions. This would include disabling or alarms, shelving of alarms and generation of dialog messages. It would also include other security related functionality such maintaining appropriate timeouts for shelving and dialogs and preventing an overload of dialog messages.	
Security	Best Practice Random Numbers	All random numbers that are required for security use appropriate cryptographic library based random number generators.	
Security	Best Practice Timeouts	The user is able to configure reasonable timeouts for Secure Channels, Sessions and Subscriptions to limit denial of service and resource consumption issues (see IEC TR 62541-2 for additional details).	
Security	Best Practice Administrative Access	The Server and Client allow for appropriate restriction of access to administrative personnel. This includes multiple levels of administrative access on platforms that support multiple administrative roles (such as Windows or Linux).	
Security	Best Practice Strict Message Handling	The application assures that messages that are illegally or incorrectly formed are rejected with appropriate error codes or appropriate actions as specified in IEC 62541-4 and IEC 62541-6.	
Security	Best Practice Audit Events Client	Audit tracking system connects to a Server using a Secure Channel and under the appropriate administrative rights to allow access to Audit Events.	
Security	Security User Name Password	The Server supports User Name/Password combination(s). Encryption of the password with the algorithm provided in the UserNameIdentityToken is required if no message encryption is used.	

Category	Title	Description	Derived
Security	Security User X509	The ~~Server~~ supports a public/private key pair for user identity. The use of this feature must be able to be enabled or disabled by an administrator.	
Security	Security User IssuedToken Kerberos	The ~~Server~~ supports a Kerberos Server token for User Identity. The use of this feature must be able to be enabled or disabled by an Administrator. Specific encryption of the IssuedToken is required if no message encryption is used. The use of this token is defined in Kerberos Token Documentation.	
Security	Security User IssuedToken Kerberos Windows	The ~~Server~~ supports the Windows implementation of Kerberos Tokens. This <i>ConformanceUnit</i> only applies if the "Security User IssuedToken Kerberos" is supported.	
Security	Security User Anonymous	The ~~Server~~ provides support for Anonymous access. The use of this feature must be able to be enabled or disabled by an Administrator. By default Anonymous access shall be disabled.	
Security	Security User IssuedToken Kerberos Client	A ~~Client~~ uses a Kerberos Server token. Specific encryption of the issuedToken is required. If no message encryption is used. The use of this token is defined by the Kerberos documentation.	
Security	Security User IssuedToken Kerberos Windows Client	A ~~Client~~ uses the Windows implementation of Kerberos tokens. This <i>ConformanceUnit</i> only applies if the "Security User IssuedToken Kerberos Client" is supported.	
Security	Security User Name Password Client	A ~~Client~~ uses a User Name/Password combination. Encryption of the password with the algorithm provided in the UserNameldentityToken is required if no message encryption is used.	
Security	Security User X509 Client	A ~~Client~~ uses a public/private key pair for user identity. This includes all validation and trust issues associated with a certificate.	

Category	Title	Description
Security	Security User Name Password	The ~~Server~~ supports User Name/Password combination(s). The token will be encrypted if required by the security policy of the User Token Policy or by the security policy of the endpoint. An unencrypted token either requires message encryption or means outside the scope of OPC UA to secure the identity token so that it cannot be retrieved by sniffing the communication. One option would be a secure transport like a VPN.
Security	Security User X509	The ~~Server~~ supports a public/private key pair for user identity. The use of this feature shall be able to be enabled or disabled by an administrator.
Security	Security User IssuedToken Kerberos	The ~~Server~~ supports a Kerberos Server token for User Identity. The use of this feature shall be able to be enabled or disabled by an Administrator. The use of this token is defined in Kerberos Token Documentation. The token will be encrypted if required by the security policy of the User Token Policy or by the security policy of the endpoint. An unencrypted token either requires message encryption or means outside the scope of OPC UA to secure the identity token so that it cannot be retrieved by sniffing the communication. One option would be a secure transport like a VPN.
Security	Security User IssuedToken Kerberos Windows	The ~~Server~~ supports the Windows implementation of Kerberos Tokens. This <i>ConformanceUnit</i> only applies if the "Security User IssuedToken Kerberos" is supported. The token will be encrypted if required by the security policy of the User Token Policy or by the security policy of the endpoint. An unencrypted token either requires message encryption or means outside the scope of OPC UA to secure the identity token so that it cannot be retrieved by sniffing the communication. One option would be a secure transport like a VPN.

Category	Title	Description
Security	Security User JWT IssuedToken	The <i>Server</i> supports a JSON Web Token (JWT) for user identity. IEC 62541-6 describes OAuth2 and JWTs in more detail. The use of this feature shall be able to be enabled or disabled by an administrator. The token will be encrypted if required by the security policy of the User Token Policy or by the security policy of the endpoint. An unencrypted token either requires message encryption or means outside the scope of OPC UA to secure the identity token so that it cannot be retrieved by sniffing the communication. One option would be a secure transport like a VPN.
Security	Security User Anonymous	The <i>Server</i> provides support for Anonymous access. The use of this feature shall be able to be enabled or disabled by an Administrator. By default, Anonymous access shall be disabled.
Security	Security User Name Password <i>Client</i>	A <i>Client</i> uses a User Name/Password combination. The token will be encrypted if required by the security policy of the User Token Policy or by the security policy of the endpoint. An unencrypted token either requires message encryption or means outside the scope of OPC UA to secure the identity token so that it cannot be retrieved by sniffing the communication. One option would be a secure transport like a VPN.
Security	Security User X509 <i>Client</i>	A <i>Client</i> uses a public/private key pair for user identity. This includes all validation and trust issues associated with a certificate.
Security	Security User IssuedToken Kerberos <i>Client</i>	A <i>Client</i> uses a Kerberos <i>Server</i> token. The use of this token is defined by the Kerberos documentation. The token will be encrypted if required by the security policy of the User Token Policy or by the security policy of the endpoint. An unencrypted token either requires message encryption or means outside the scope of OPC UA to secure the identity token so that it cannot be retrieved by sniffing the communication. One option would be a secure transport like a VPN.
Security	Security User IssuedToken Kerberos Windows <i>Client</i>	A <i>Client</i> uses the Windows implementation of Kerberos tokens. This <i>ConformanceUnit</i> only applies if the "Security User IssuedToken Kerberos <i>Client</i>" is supported. The token will be encrypted if required by the security policy of the User Token Policy or by the security policy of the endpoint. An unencrypted token either requires message encryption or means outside the scope of OPC UA to secure the identity token so that it cannot be retrieved by sniffing the communication. One option would be a secure transport like a VPN.
Security	Security User JWT IssuedToken <i>Client</i>	A <i>Client</i> uses a JSON Web Token (JWT) for user identity. IEC 62541-6 describes OAuth2 and JWTs in more detail. The token will be encrypted if required by the security policy of the User Token Policy or by the security policy of the endpoint. An unencrypted token either requires message encryption or means outside the scope of OPC UA to secure the identity token so that it cannot be retrieved by sniffing the communication. One option would be a secure transport like a VPN.
Security	Security Invalid user token	<i>Servers</i> shall take proper measures to protect against attacks on user identity tokens. Such an attack is assumed if repeated connection attempts with invalid user identity tokens happen. See <i>ActivateSession Service</i> in IEC 62541-4.
Security	Security User JWT Token Policy	The <i>Server</i> supports one or more Endpoints with a <i>UserTokenPolicy</i> that includes a <i>JWT IssuerEndpointUrl</i> as defined in IEC 62541-6. For JWT the <i>issuerEndpointUrl</i> is a JSON object that includes all parameters that define the <i>AuthorizationService</i>. As part of the JWT Token Policy, the <i>Server</i> shall support at least one of the following Authority Profile Conformance Units. The URIs defined in the <i>ConformanceUnit</i> shall be exposed in the <i>authorityProfileURI</i> field of the JWT Token Policy.

Category	Title	Description
Security	Security User JWT Token Policy <i>Client</i>	The Client understands and uses the Authorization Service definition inside the JWT UserTokenPolicy returned with GetEndpoints. It shall support at least one of the following Authority Profile Conformance Units. The URIs defined in the ConformanceUnit are in the authorityProfileURI field of the JWT Token Policy exposed in Server Endpoints.
Security	OAuth2 Authority <i>Profile</i>	This unit indicates support of OAuth2 over HTTPS to request access tokens. The URI for the interactions with this authority is "http://opcfoundation.org/UA/Authorization#OAuth2"
Security	OPC UA Authority <i>Profile</i>	This unit indicates support of the OPC UA <i>Methods</i> defined in IEC 62541-12 to request access tokens. The URI for the interactions with this authority is "http://opcfoundation.org/UA/Authorization#OPCUA"
Security	Azure Identity Provider Authority <i>Profile</i>	This unit indicates support of the Azure identity provider to request access tokens. The URI for the interactions with this authority is "http://opcfoundation.org/UA/Authorization#Azure"
Security	Security <i>Certificate</i> Validation	A certificate will be validated as specified in IEC 62541-4. This includes, among others, structure and signature examination. Allowing for some validation errors to be suppressed by administration directive.
Security	Security Default ApplicationInstance Certificate	An application, when installed, has a default ApplicationInstanceCertificate that is valid. The default ApplicationInstanceCertificate shall either be created as part of the installation or installation instructions explicitly describe the process to create and apply a default ApplicationInstanceCertificate to the application.
Security	Security – No Application Authentication	The <i>Server</i> supports being able to be configured for no application authentication, just User authentication and normal encryption/signing: – Configure <i>Server</i> to accept all certificates – <i>Certificates</i> are just used for message security (signing and encryption) – Users level is used for authentication
Security	Security Policy Required	Support at least Security Policy [A] and Security Policy [B]. Support of multiple Security Policies – even obsolete ones – is recommended. This will provide best interoperability and allows the end user to choose the required level of security. Obsolete Security Policies shall not be enabled/usable without administrative intervention.
Security	Security None CreateSession ActivateSession	When SecurityPolicy = None, the CreateSession and ActivateSession service allow for a NULL/empty signature and do not require Application <i>Certificates</i> or a Nonce.
Security	Security None CreateSession ActivateSession 1.0	The <i>Client</i> can connect to <i>Servers</i> that require a certificate being passed on Session establishment. The <i>Client</i> in this case will first try without a <i>Certificate</i> and if this fails present a <i>Certificate</i> .
Security	Security TLS General	This <i>ConformanceUnit</i> indicates that at least one of the transport security <i>Profiles</i> for TLS is supported by this application. It is used in TLS transport <i>Profiles</i> , but the choice of transport security profile is optional. The actual used security profile will default to the most secure one.
Security	Security TLS_RSA with AES_256_CBC_SHA256	The connection is established using TLS_RSA_WITH_AES_256_CBC_SHA256. That has a MinAsymmetricKeyLength – 2048, MaxAsymmetricKeyLength – 4096, AsymmetricSignatureAlgorithm – RSA_SHA256. (TLS 1.2)
Security	Security TLS_DHE_RSA with AES_nnn_CBC_SHA256	The connection is established using TLS_DHE_RSA with AES_128_CBC_SHA256 or TLS_DHE_RSA with AES_256_CBC_SHA256. That has a MinAsymmetricKeyLength – 2048, MaxAsymmetricKeyLength – 4096, CertificateSignatureAlgorithm – RSA_SHA256. (TLS 1.2). <i>Clients</i> and <i>Servers</i> have to support both algorithms.
Security	Security Encryption Required	Encryption is required using the algorithms provided in the security algorithm suite.

Category	Title	Description
Security	Security Signing Required	Signing is required using the algorithms provided in the security algorithm suite.
Security	Security Time Synch – Configuration	Application supports configuring acceptable clock skew.
Security	Security Time Synch – NTP / OS Based support	Application supports time synchronization, either via an implementation of Network Time Protocol (NTP), or via features of a standard operating system.
Security	Security Time Synch – UA based support	An application makes use of the responses header timestamp provided by a configured well know source, such as a <i>Discovery Server</i> to synchronize the time on the application and that this time synchronization occurs periodically. Use of this TimeSyncing can be configured.
Security	Security Administration	Allow configuration of the following Security related items (when they apply). * select the allowed User identification policy or policies (e.g. Username/Password or X509). * enable/disable the security policy "None" or other security policies. * enable/disable endpoints with MessageSecurityMode SIGN or SIGNANDENCRYPT. * set the permitted certification authorities. * define how to react to unknown Certificates. * allow accepting any valid Certificate
Security	Security Administration – XML Schema	Support the OPC UA defined XML schema for importing and exporting security configuration information. This schema is defined in IEC 62541-6.
Security	Security <i>Certificate</i> Administration	Allow a site administrator to be able to assign a site specific ApplicationInstanceCertificate and if desired to configure a site-specific <i>Certificate Authority (CA)</i> .
Security	Security Role <i>Server</i> Base	Support the User Authorization Information Model defined in UA Parts 3 and 5 – like Roles – and the RolePermissions and UserRolePermissions <i>Attributes</i> .
Security	Security Role Well Known	Support the well-known Roles "ConfigureAdmin" and "SecurityAdmin" with suggested permissions defined in IEC 62541-3.
Security	Security Role <i>Server</i> IdentityManagement	Allow authorized users to add and/or remove Identities from Roles with the appropriate <i>Methods</i> .
Security	Security Role <i>Server</i> Management	Allow authorized users to create new Roles and/or remove Roles with the appropriate <i>Methods</i> .
Security	Security Role <i>Server</i> Restrict Applications	Support adding applications to a Role with the appropriate <i>Methods</i> so that only these applications can use this Role.
Security	Security Role <i>Server</i> Restrict Endpoints	Support adding Endpoints to a Role with the appropriate <i>Methods</i> . With this restriction a Role is only applied when a <i>Client</i> connects via one of these Endpoints.
Security	Security Role <i>Server</i> DefaultRolePermissions	Allow authorized users to set the DefaultRolePermissions <i>Property</i> for certain NameSpaces. DefaultRolePermissions are applied if no RolePermissions are associated with a <i>Node</i> .
Security	Security Role <i>Server</i> RolePermissions	Allow authorized users to set the RolePermissions <i>Attribute</i> on <i>Nodes</i> .
Security	Security Role <i>Server</i> Authorization	Restrict access based on the configured Roles and permissions.
Security	Security Role <i>Client</i> Base	Understand and use the User Authorization Information Model defined in IEC 62541-5 and the RolePermissions <i>Attribute</i> .
Security	Security Role <i>Client</i> Management	Support creating new Roles and adding Identities as well as remove Roles or Identities using the appropriate <i>Methods</i> .
Security	Security Role <i>Client</i> Restrict Applications	Use the appropriate <i>Methods</i> to add applications to a Role so that only these applications can use this Role.
Security	Security Role <i>Client</i> Restrict Endpoints	Use the appropriate <i>Methods</i> to add Endpoints to a Role. With this restriction a Role is only applied when a <i>Client</i> connects via one of these Endpoints.
Security	Security Role <i>Client</i> DefaultRolePermissions	Ability to set the DefaultRolePermissions <i>Property</i> for certain NameSpaces. DefaultRolePermissions are applied if no RolePermissions are associated with a <i>Node</i> .

Category	Title	Description
Security	Security Role <i>Client</i> RolePermissions	Support setting the RolePermissions <i>Attribute</i> on <i>Nodes</i> .
Security	Pull Model for Global <i>Certificate</i> and TrustList Management	Use the <i>Certificate</i> Management <i>Services</i> of IEC 62541-12 for the Pull model to manage Application Instance <i>Certificates</i> and Trust Lists including Revocation Lists.
Security	Push Model for Global <i>Certificate</i> and TrustList Management	Support the <i>Certificate</i> Management <i>Services</i> of IEC 62541-12 for the Push model to manage Application Instance <i>Certificates</i> and Trust Lists including Revocation Lists.
Security	Pull Model for KeyCredential <i>Service</i>	Use the <i>Methods</i> on an instance of the KeyCredentialServiceType (Pull model) to obtain KeyCredentials as specified in IEC 62541-12.
Security	Push Model for KeyCredential <i>Service</i>	Support the KeyCredential <i>Services</i> Push model of IEC 62541-12 to obtain KeyCredentials. This includes support of one or more instances of the KeyCredentialConfigurationType and the <i>Methods</i> to update or delete credentials.
Security	Authorization Service Configuration Server	Support the Object Types defined in IEC 62541-12 to allow configuration of information needed to accept Access Tokens when presented by the Client during session establishment. Access Tokens are issued by Authorization <i>Services</i> .
Security	Authorization Service Client	Use the RequestAccessToken Method defined in IEC 62541-12.
Security	SymmetricSignatureAlgorithm_None	This algorithm does not apply.
Security	SymmetricSignatureAlgorithm_HMAC-SHA1	A keyed hash which is defined in https://tools.ietf.org/html/rfc2104 . The hash algorithm is SHA1 and is described in https://tools.ietf.org/html/rfc3174 . The URI is http://www.w3.org/2000/09/xmldsig#hmac-sha1 . No known exploits exist when using SHA1 with a keyed hash, however, SHA1 was broken in 2017 so use of this algorithm is not recommended.
Security	SymmetricSignatureAlgorithm_HMAC-SHA2-256	A keyed hash used for message authentication which is defined in https://tools.ietf.org/html/rfc2104 . The hash algorithm is SHA2 with 256 bits and described in https://tools.ietf.org/html/rfc4634
Security	SymmetricEncryptionAlgorithm_None	This algorithm does not apply.
Security	SymmetricEncryptionAlgorithm_AES128-CBC	The AES encryption algorithm which is defined in http://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.197.pdf . Multiple blocks encrypted using the CBC mode described in http://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-38a.pdf . The key size is 128 bits. The block size is 16 bytes. The URI is http://www.w3.org/2001/04/xmlenc#aes128-cbc .
Security	SymmetricEncryptionAlgorithm_AES256-CBC	The AES encryption algorithm which is defined in http://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.197.pdf . Multiple blocks encrypted using the CBC mode described in http://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-38a.pdf . The key size is 256 bits. The block size is 16 bytes. The URI is http://www.w3.org/2001/04/xmlenc#aes256-cbc .

Category	Title	Description
Security	SymmetricEncryptionAlgorithm_AES128-CTR	The AES encryption algorithm which is defined in http://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.197.pdf. Multiple blocks encrypted using the CTR mode described in http://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-38a.pdf. The counter block format is defined in https://tools.ietf.org/html/rfc3686. The key size is 128 bits. The block size is 16 bytes. The input nonce length is 4 bytes. The URI is http://opcfoundation.org/UA/security/aes128-ctr.
Security	SymmetricEncryptionAlgorithm_AES256-CTR	The AES encryption algorithm which is defined in http://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.197.pdf. The key size is 256 bits. The block size is 16 bytes. Multiple blocks encrypted using the CTR mode described in http://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-38a.pdf. The counter block format is defined in https://tools.ietf.org/html/rfc3686. The key size is 128 bits. The block size is 16 bytes. The input nonce length is 4 bytes. The URI is http://opcfoundation.org/UA/security/aes256-ctr.
Security	AsymmetricSignatureAlgorithm_None	This algorithm does not apply.
Security	AsymmetricSignatureAlgorithm_RSA-PKCS15-SHA1	The RSA signature algorithm which is defined in https://tools.ietf.org/html/rfc3447. The RSASSA-PKCS1-v1_5 scheme is used. The hash algorithm is SHA1 and is described in https://tools.ietf.org/html/rfc3174. The URI is http://www.w3.org/2000/09/xmldsig#rsa-sha1. SHA1 was broken in 2017 so this algorithm should not be used.
Security	AsymmetricSignatureAlgorithm_RSA-PKCS15-SHA2-256	The RSA signature algorithm which is defined in https://tools.ietf.org/html/rfc3447. The RSASSA-PKCS1-v1_5 scheme is used. The hash algorithm is SHA2 with 256bits and is described in https://tools.ietf.org/html/rfc6234. The URI is http://www.w3.org/2001/04/xmldsig-more#rsa-sha256.
Security	AsymmetricSignatureAlgorithm_RSA-PSS - SHA2-256	The RSA signature algorithm which is defined in https://tools.ietf.org/html/rfc3447. The RSASSA-PSS scheme is used. The hash algorithm is SHA2 with 256bits and is described in https://tools.ietf.org/html/rfc6234. The mask generation algorithm also uses SHA2 with 256 bits. The salt length is 32 bytes. The URI is http://opcfoundation.org/UA/security/rsa-pss-sha2-256.
Security	AsymmetricEncryptionAlgorithm_None	This algorithm does not apply.
Security	AsymmetricEncryptionAlgorithm_RSA-PKCS15	The RSA encryption algorithm which is defined in https://tools.ietf.org/html/rfc3447. The RSAES-PKCS1-v1_5 scheme is used. The URI is http://www.w3.org/2001/04/xmlenc#rsa-1_5. The RSAES-PKCS1-v1_5 scheme has known weaknesses and is not recommended.

Category	Title	Description
Security	AsymmetricEncryptionAlgorithm_RSA-OAEP-SHA1	The RSA encryption algorithm which is defined in https://tools.ietf.org/html/rfc3447. The RSAES-OAEP scheme is used. The hash algorithm is SHA1 and is described in https://tools.ietf.org/html/rfc6234. The mask generation algorithm also uses SHA1. The URI is http://www.w3.org/2001/04/xmlenc#rsa-oaep. No known exploits exist when using SHA1 with RSAES-OAEP, however, SHA1 was broken in 2017 so use of this algorithm is not recommended.
Security	AsymmetricEncryptionAlgorithm_RSA-OAEP-SHA2-256	The RSA encryption algorithm which is defined in https://tools.ietf.org/html/rfc3447. The RSAES-OAEP scheme is used. The hash algorithm is SHA2 with 256 bits and is described in https://tools.ietf.org/html/rfc6234. The mask generation algorithm also uses SHA2 with 256 bits. The URI is http://opcfoundation.org/UA/security/rsa-oaep-sha2-256.
Security	KeyDerivationAlgorithm_None	This algorithm does not apply.
Security	KeyDerivationAlgorithm_P-SHA1	The P_SHA-1 pseudo-random function defined in https://tools.ietf.org/html/rfc4346. The URI is http://docs.oasis-open.org/ws-sx/ws-secureconversation/200512/dk/p_sha1. No known exploits exist when using SHA1 with P-SHA-1, however, SHA1 was broken in 2017 so use of this algorithm is not recommended.
Security	KeyDerivationAlgorithm_P-SHA2-256	The P_SHA256 pseudo-random function defined in https://tools.ietf.org/html/rfc5246. The URI is http://docs.oasis-open.org/ws-sx/ws-secureconversation/200512/dk/p_sha256.
Security	CertificateSignatureAlgorithm_None	This algorithm does not apply.
Security	CertificateSignatureAlgorithm_RSA-PKCS15-SHA2-256	The RSA signature algorithm which is defined in https://tools.ietf.org/html/rfc3447. The RSASSA-PKCS1-v1_5 scheme is used. The hash algorithm is SHA2 with 256bits and is described in https://tools.ietf.org/html/rfc6234. The SHA2 algorithm with 384 or 512 bits may be used instead of SHA2 with 256 bits. The URI is http://www.w3.org/2001/04/xmldsig-more#rsa-sha256.
Security	CertificateSignatureAlgorithm_RSA-PKCS15-SHA1	The RSA signature algorithm which is defined in https://tools.ietf.org/html/rfc3447. The RSASSA-PKCS1-v1_5 scheme is used. The hash algorithm is SHA1 and is described in https://tools.ietf.org/html/rfc3174. The URI is http://www.w3.org/2000/09/xmldsig#rsa-sha1. SHA1 was broken in 2017 so this algorithm should not be used. The SHA2 algorithm with 244, 256, 384 or 512 bits may be used instead of SHA1. The SHA2 algorithm is described in https://tools.ietf.org/html/rfc6234.
Security	SecurityPolicy_None_Limits	DerivedSignatureKeyLength: 0

Category	Title	Description
Security	Aes128-Sha256-RsaOaep_Limits	-> DerivedSignatureKeyLength: 256 bits -> MinAsymmetricKeyLength: 2048 bits -> MaxAsymmetricKeyLength: 4096 bits -> SecureChannelNonceLength: 32 bytes
Security	Basic256Sha256_Limits	-> DerivedSignatureKeyLength: 256 bits -> MinAsymmetricKeyLength: 2048 bits -> MaxAsymmetricKeyLength: 4096 bits -> SecureChannelNonceLength: 32 bytes
Security	Aes256-Sha256-RsaPss_Limits	-> DerivedSignatureKeyLength: 256 bits -> MinAsymmetricKeyLength: 2048 bits -> MaxAsymmetricKeyLength: 4096 bits -> SecureChannelNonceLength: 32 bytes
Security	Basic128Rsa15_Limits	-> DerivedSignatureKeyLength: 128 bits -> MinAsymmetricKeyLength: 1024 bits -> MaxAsymmetricKeyLength: 2048 bits -> SecureChannelNonceLength: 16 bytes
Security	Basic256_Limits	-> DerivedSignatureKeyLength: 192 bits -> MinAsymmetricKeyLength: 1024 bits -> MaxAsymmetricKeyLength: 2048 bits -> SecureChannelNonceLength: 32 bytes

Table 12 describes protocol and encoding related features that can be profiled. These features are defined in detail in IEC 62541-6. It is recommended that *Servers* and *Clients* support as many of these options as possible for greatest interoperability.

Table 12 – Protocol and Encoding

Category	Title	Description	Derived
<i>Server</i>	Protocol Configuration	Allow administration of the Endpoints and the port number used by the Endpoints.	
Transport	Protocol TCP Binary UA Security	Support the UA TCP transport protocol with UA Binary Encoding and with UA Secure Conversation.	
Transport	Protocol HTTPS with UA Binary	Support the HTTPS protocol with UA Binary Encoding.	
Transport	Protocol HTTPS with Soap	Support the HTTPS protocol with Soap-based Xml Encoding.	
Transport	Protocol Soap Xml WS Security	Support "SOAP/HTTP" transport with XML Encoding and with WS Secure Conversation.	
Transport	Protocol Soap Binary WS Security	Support "SOAP/HTTP" transport with UA Binary Encoding and with WS Secure Conversation.	

Category	Title	Description
<i>Server</i>	Protocol Reverse Connect <i>Server</i>	Support reverse connectivity by sending a ReverseHello message to a Client. The reverse connect procedure can be applied to several transports as specified in IEC 62541-6 and shall be supported for all of these that are available in a Server.
<i>Server</i>	Protocol Configuration	Allow administration of the Endpoints and the port number used by the Endpoints.
<i>Client</i>	Protocol Reverse Connect <i>Client</i>	Support reverse connectivity by accepting Reverse Hello messages from Servers and establish a Secure Channel if the URI of the Server is accepted by Client or Client user. The reverse connect procedure can be applied to several transports as specified in IEC 62541-6 and shall be supported for all of these transports that are supported by the Client.
Transport	Protocol UA TCP	Support the UA TCP transport protocol as defined in IEC 62541-6.
Transport	Protocol HTTPS	Support the HTTPS protocol as defined in IEC 62541-6.
Transport	Protocol Web Sockets	Support the WebSocket protocol (WSS) with at least one of the sub-protocols defined in IEC 62541-6.
Transport	UA Secure Conversation	Support UA Secure Conversation specified in IEC 62541-6.
Transport	UA Binary Encoding	Support UA Binary Encoding. Values of these data types are encoded in compact binary formats, contiguously and without tagging. I.e. the receiver is assumed to understand the structure it is decoding.
Transport	UA SOAP-XML Encoding	Support Soap V1.2 based Xml Encoding as defined in IEC 62541-6. The XML elements include all information necessary to convert it back into OPC UA structures of any language.
Transport	JSON Reversible Encoding	Support reversible JSON Encoding as defined in IEC 62541-6. The JSON object includes all information necessary to convert it back into OPC UA structures of any language.

5.4 Information Model and AddressSpace related features

Table 13 describes base features related items that can be profiled. For additional information about these items, please refer to IEC 62541-3, IEC 62541-5 and IEC 62541-6. *Servers* with a larger resource capacity would support most of this functionality, but smaller resource constraint *Server* may omit some of this functionality. Many *Clients* would utilize some of this functionality and more robust *Clients* would utilize most of this functionality.

Table 13 – Base Information

Category	Title	Description	Derived
Server	Base Info Core Structure	The <i>Server</i> supports the <i>Server Object</i> , <i>ServerCapabilities</i> and supports the OPC UA <i>AddressSpace</i> structure.	
Server	Base Info Server Capabilities	The <i>Server</i> supports publishing of the <i>Server</i> limitation in the <i>ServerCapabilities</i> , including <i>MaxArrayLength</i> , <i>MaxStringLength</i> , <i>MaxNodePerRead</i> , <i>MaxNodesPerWrite</i> , <i>MaxNodesPerSubscription</i> and <i>MaxNodesPerBrowse</i> .	
Server	Base Info Progress Events	The <i>Server</i> exposes if generation of Progress events for long running service calls such as <i>HistoryRead</i> or <i>Query</i> is supported. If it is listed as supported in <i>ServerCapabilities</i> , then the actual events are verified.	
Server	Base Info Diagnostics	~~The <i>Server</i> supports Diagnostic Objects and Variables.~~ The <i>Server</i> supports the collection of diagnostic information. The <i>EnabledFlag</i> in the <i>ServerDiagnostics Object</i> can be set TRUE and in that case all static and dynamic <i>Objects</i> and <i>Variables</i> for diagnostic data as defined in IEC 62541-5 are supported.	
Server	Base Info System Status	The <i>Server</i> supports generating <i>SystemStatusChangeEvent</i> Type indicating shutdown of the <i>Server</i> (<i>SourceNode=Server</i>).	
Server	Base Info Estimated Return Time	<i>Server</i> supports the <i>EstimatedReturnTime Property</i> . It indicates the time at which the <i>Server</i> is expected to have a <i>ServerStatus.State</i> of <i>RUNNING_0</i> . <i>Clients</i> can use this information to govern the reconnect logic.	
Server	Base Info System Status Underlying System	The <i>Server</i> supports generating <i>SystemStatusChangeEvent</i> Type indicating changes to an Underlying System (<i>SourceNode = Server</i>). This event can also be used to indicate that the OPC UA <i>Server</i> has underlying systems.	
Server	Base Info Device Failure	The <i>Server</i> supports generating <i>DeviceFailureEvent</i> Type indicating changes to individual devices in an underlying system.	
Server	Base Info GetMonitoredItems <i>Method</i>	The <i>Server</i> supports obtaining subscription information via <i>GetMonitoredItems Method</i> on the <i>Server</i> object.	
Server	Base Info ResendData <i>Method</i>	Support the standard <i>Method ResendData</i> (defined in IEC 62541-5) to get the latest value of the monitored items of a <i>Subscription</i> .	
Server	Base Info Type System	The <i>Server</i> exposes a <i>Type System</i> with <i>DataTypes</i> , <i>ReferenceTypes</i> , <i>ObjectTypes</i> and <i>VariableTypes</i> including all of the OPC UA (namespace 0) types that are used by the <i>Server</i> , as defined in IEC 62541-5. Items that are defined in Namespace 0 but are defined in other specification parts are tested as part of the other information models.	

Category	Title	Description	Derived
Server	Base Info Custom Type System	~~The Server supports defining user defined <i>ObjectTypes</i>, <i>VariableTypes</i>, <i>ReferenceType</i> and <i>DataTypes</i>. Supporting this conformance unit does not require that a <i>Server</i> exposes the OPC-UA <i>Object</i>, <i>Variable</i>, <i>Reference</i>, or <i>Data Types</i>, unless the <i>Server</i> implements <i>User types</i>. If <i>User types</i> are defined then the full type-hierarchy has to be exposed as well.~~ The <i>Server</i> supports custom types (i.e. types that are derived from well-known <i>ObjectTypes</i> , <i>VariableTypes</i> , <i>ReferenceTypes</i> or <i>DataTypes</i>). Supporting this conformance unit requires that the custom types with their full inheritance tree are exposed in the <i>AddressSpace</i> .	
Server	Base Info Model Change	The <i>Server</i> supports <i>ModelChange Event</i> and <i>NodeVersion Property</i> for all <i>Nodes</i> that the server allows <i>Model changes</i> for.	
Server	Base Info Placeholder Modelling Rules	The <i>Server</i> supports defining custom <i>Object</i> or <i>Variables</i> that include the use of <i>OptionalPlaceholder</i> or <i>MandatoryPlaceholder</i> modelling rules.	
Server	Base Info SemanticChange	The <i>Server</i> supports <i>SemanticChangeEvent</i> for some <i>Properties</i> . This includes setting the <i>SemanticChange Bit</i> in the status when a semantic change occurs, such as a change in the engineering unit associated with a value.	
Server	Base Info EventQueueOverflow EventType	The <i>Server</i> supports the <i>EventQueueOverflowEventType</i> as defined in IEC 62541-4.	
Server	Base Info OptionSet	The <i>Server</i> supports the <i>VariableType OptionSet</i> .	
Server	Base Info ValueAsText	The <i>Server</i> supports the <i>Property ValueAsText</i> for enumerated <i>DataTypes</i> .	
Server	Base Info Engineering Units	The <i>Server</i> supports defining <i>Variables</i> that include the <i>Engineering Units Property</i> . This property makes use of the <i>EUInformation</i> data structure. This structure by default represents the UN/CEFACT "Codes for Units of Measurement". If a different EU representation is required then the <i>EUInformation.namespaceUri</i> will indicate the alternate namespace.	
Server	Base Info Selection List	The <i>Server</i> supports <i>Variables</i> of the <i>SelectionListType VariableType</i> .	
Server	Base Info FileType Base	The <i>Server</i> supports the <i>FileType Object</i> (see IEC 62541-5). File writing may be restricted.	
Server	Base Info FileType Write	The <i>Server</i> supports the <i>FileType Object</i> , including writing of files. Also included is the support of user access control on <i>FileType Object</i> .	
Server	Base Info RequestServerStateChange Method	The <i>Server</i> supports the <i>RequestServerStateChange Method</i> .	

Category	Title	Description	Derived
Server	Base Info State Machine Instance	Support instances of the <i>StateMachineType</i> or a sub-type in the <i>AddressSpace</i> . Generate Events when significant state changes occur. At least one <i>GeneratesEvent Reference</i> exists to define the <i>Event(s)</i> triggered on state changes.	
Server	Base Info Finite State Machine Instance	Support instances of the <i>FiniteStateMachineType</i> or a sub-type in the <i>AddressSpace</i> .	
Server	Base Info Available States and Transitions	Support the Properties <i>AvailableStates</i> and <i>AvailableTransitions</i> defined for the <i>FiniteStateMachineType</i> .	
Client	Base Info <i>Client</i> Basic	The <i>Client</i> uses the defined OPC UA <i>AddressSpace</i> . Access or provide access to <i>Server</i> information like the <i>Server's</i> state, <i>BuildInfo</i> , capabilities, <i>Namespace Table</i> and <i>Type Model</i> .	
Client	Base Info <i>Client</i> Honour Operation Limits	The <i>Client</i> shall honour <i>Server</i> limits described in <i>ServerCapabilities Object</i> of <i>Server</i> .	
Client	Base Info <i>Event</i> Processing	The <i>Client</i> is able to subscribe for and process base OPC UA Events.	
Client	Base Info <i>Client</i> System Status	The <i>Client</i> makes use of <i>SystemStatusChangeEvent</i> to detect server shutdowns.	
Client	Base Info <i>Client</i> Estimated Return Time	<i>Client</i> uses the <i>EstimatedReturnTime Property</i> to govern the reconnect logic.	
Client	Base Info <i>Client</i> System Status Underlying System	The <i>Client</i> makes use of <i>SystemStatusChangeEvent</i> to detect ~~server shutdowns~~ changes to an Underlying System (<i>SourceNode</i> = <i>Server</i>).	
Client	Base Info <i>Client</i> Device Failure	The <i>Client</i> makes use of <i>DeviceFailureEvent</i> to detect failed devices in underlying systems	
Client	Base Info <i>Client</i> Progress Events	The <i>Client</i> makes use of <i>ProgressEvents</i> , including checking for their support.	
Client	Base Info <i>Client</i> Diagnostics	The <i>Client</i> provides interactive or programmatic access to the <i>Server's</i> diagnostic information.	
Client	Base Info <i>Client</i> Type Programming	The <i>Client</i> programmatically process instances of <i>Objects</i> or <i>Variables</i> by using their type definitions. This includes custom <i>DataTypes</i> , <i>ObjectTypes</i> and <i>VariableTypes</i> .	
Client	Base Info <i>Client</i> Type Pre-Knowledge	The <i>Client</i> shall interoperate with <i>Servers</i> that do not expose OPC UA <i>Types</i> in <i>AddressSpace</i> .	
Client	Base Info <i>Client</i> Remote Nodes	The <i>Client</i> can access Nodes that have an extended <i>NodeId</i> that reference a <i>Server</i> different then the originating <i>Server</i> . It is acceptable that the <i>Server</i> configuration information be pre-configured on the <i>Client</i> .	
Client	Base Info <i>Client</i> Change Events	The <i>Client</i> processes <i>ModelChangeEvents</i> to detect changes in the <i>Server's</i> OPC UA <i>AddressSpace</i> and take appropriate action for a given change.	
Client	Base Info <i>Client</i> GetMonitoredItems Method	The <i>Client</i> makes use of <i>GetMonitoredItems Method</i> to recover for communication interruptions and/or to recover subscription information.	

Category	Title	Description	Derived
<i>Client</i>	Base Data <i>Client</i> ResendData Method	The <i>Client</i> makes use of ResendData Method to fetch the last value of the data monitored items.	
<i>Client</i>	Base Info <i>Client</i> Selection List	The <i>Client</i> uses and understands Variables of the SelectionListType VariableType.	
<i>Client</i>	Base Info <i>Client</i> FileType Base	The <i>Client</i> can access a FileType Object to transfer a file from the Server to the <i>Client</i> . This includes large files.	
<i>Client</i>	Base Info <i>Client</i> FileType Write	The <i>Client</i> can access a FileType Object to transfer a file from the <i>Client</i> to the Server. This includes large files.	
<i>Client</i>	Base Info <i>Client</i> RequestServerStateChange	The <i>Client</i> can invoke the RequestServerStateChange Method.	
<i>Client</i>	Base Info <i>Client</i> State Machine Instance	Use instances of the StateMachineType or a sub-type. Monitor either the CurrentState component Variable of the instance or the Events triggered as effect of state changes. Use Methods when defined for the StateMachineType to affect the state.	
<i>Client</i>	Base Info <i>Client</i> Finite State Machine Instance	Use instances of the FinitStateMachineType or a sub-type. Monitor either the CurrentState component Variable of the instance or the Events triggered as effect of state changes.	
<i>Client</i>	Base Info <i>Client</i> Available States and Transitions	Use the Properties AvailableStates and AvailableTransitions when exposed by a Server.	

Table 14 describes Address Space Model information related items that can be profiled. The details of these model items are defined in IEC 62541-3 and IEC 62541-5. This includes *Server Facets* that describe what a *Server* exposes and *Client Facets* that describe what a *Client* consumes.

Table 14 – Address Space Model

Category	Title	Description	Derived
Server	Address Space Base	Support the <i>NodeClasses</i> with their <i>Attributes</i> and ~~behaviour~~ <i>References</i> as defined in IEC 62541-3. This includes for instance: <i>Object</i> , <i>ObjectType</i> , <i>Variable</i> , <i>VariableType</i> , <i>References</i> and <i>DataTypes</i> .	
Server	Address Space Atomicity	Support setting the NonatomicRead and NonatomicWrite flags in the AccessLevelEx Attribute for <i>Variable Nodes</i> to indicate whether Read or Write operations can be performed in atomic manner. If the flags are set to '1', atomicity cannot be assured.	
Server	Address Space Full Array Only	Support setting the WriteFullArrayOnly flag in the AccessLevelEx Attribute for <i>Variable Nodes</i> of non-scalar data types to indicate whether write operations for an array can be performed with an IndexRange.	
Server	Address Space Events	Support OPC UA AddressSpace elements for generating <i>Event</i> notifications. This includes at least one <i>Node</i> with an <i>EventNotifier Attribute</i> set to True (<i>Server Node</i>).	
Server	Address Space Complex ~~DataTypes~~ Data Dictionary	~~Support StructuredDataTypes with a Data Dictionary.~~ Support structured DataTypes with a Data Dictionary. Note that V1.04 of IEC 62541-3 specifies a simplified approach using the new DataTypeDefinition Attribute. The "Address Space DataTypeDefinition Attribute" Conformance Unit requires support of the DataTypeDefinition Attribute. Support of a DataDictionary will be deprecated in one of the next OPC UA versions.	
Server	Address Space DataTypeDefinition Attribute	Support structured DataTypes and expose the meta data and encoding information with a StructureDefinitionType via the DataTypeDefinition Attribute.	

Category	Title	Description	Derived
Server	Address Space Method	Support <i>Method Nodes</i> .	
Server	Address Space Notifier Hierarchy	Supports using the HasNotifier reference to build a hierarchy of <i>Object Nodes</i> that are notifiers with other notifier <i>Object Nodes</i> .	
Server	Address Space Source Hierarchy	Supports hierarchies of event sources where each hierarchy roots in an <i>Object Node</i> that is a notifier. The HasEventSource Reference is used to relate the <i>Nodes</i> within a hierarchy. If <i>Conditions</i> are supported, the hierarchy shall include HasCondition References.	
Server	Address Space WriteMask	Supports WriteMask indicating the write access availability for all attributes, including not supported attributes.	
Server	Address Space UserWriteMask	Supports UserWriteMask indicating the write access availability for all attributes for the given user, including not supported attributes. Support includes at least two levels of users.	
Server	Address Space UserWriteMask Multilevel	Supports UserWriteMask indicating the write access availability for all attributes for the given user, including not supported attributes. This includes supporting multiple levels of access control for all nodes in the system.	
Server	Address Space User Access Level Full	Implements User Access Level security, this includes supporting multiple levels of access control for <i>Variable</i> nodes in the system. This includes an indication of read, write, Historical read and Historical write access to the Value Attribute.	
Server	Address Space User Access Level Base	Implements User Access Level Security for <i>Variable</i> nodes, this includes at least two users in the system. This includes an indication of read, write, historical read and Historical write access to the value attribute	
Client	Address Space Client Base	Uses and understands the <i>NodeClasses</i> with their <i>Attributes</i> and behaviour as defined in IEC 62541-3. This includes for instance: <i>Object</i> , <i>ObjectType</i> , <i>Variable</i> , <i>VariableType</i> , <i>References</i> and <i>DataTypes</i> . This includes treating BrowseNames and String NodeIds as case sensitive.	
Client	Address Space Client Atomicity	Access the NonatomicRead or NonatomicWrite flags in the AccessLevelEx Attribute of <i>Variable Nodes</i> to determine whether Read or Write operations can be performed in atomic manner. This information will typically be shown to a user for further action.	
Client	Address Space Client Full Array Only	Access the WriteFullArrayOnly flag in the AccessLevelEx Attribute of <i>Variable Nodes</i> with non-scalar data types to determine whether writing to an array with an IndexRange is allowed.	
Client	Address Space Client Complex Data Types Data Dictionary	Uses and understands arbitrary structured <i>DataTypes</i> via Data Dictionary. Note that V1.04 of IEC 62541-3 specifies a simplified approach using the new DataTypeDefinition Attribute. The "Address Space Client DataTypeDefinition Attribute" Conformance Unit requires support of the DataTypeDefinition Attribute.	
Client	Address Space Client DataTypeDefinition Attribute	Uses and understands arbitrary structured <i>DataTypes</i> where the meta data and encoding information are exposed with the StructureDefinitionType via the DataTypeDefinition Attribute.	
Client	Address Space Client Notifier Hierarchy	Uses hierarchy of <i>Object Nodes</i> that are notifiers to detect specific areas where the <i>Client</i> can subscribe for Events.	
Client	Address Space Client Source Hierarchy	Detect and use the hierarchy of event sources exposed for specific <i>Object Nodes</i> that are event notifiers.	

Table 15 describes Data Access information model related items that can be profiled. The details of this model are defined in IEC 62541-8. Servers ~~could~~ may expose this information model and Clients ~~could~~ may utilize this information model.

Table 15 – Data Access

Category	Title	Description	Derived
Server	Data Access DataItems	Provide <i>Variables</i> of <i>DataItem</i> Type or one of its subtypes. Support the <i>StatusCodes</i> specified in IEC 62541-8. Support of optional <i>Properties</i> (e.g. "InstrumentRange") shall be verified during certification testing and will be shown in the <i>Certificate</i> .	
Server	Data Access AnalogItems	Support <i>AnalogItem</i> Type <i>Variables</i> with corresponding <i>Properties</i> . The support of optional <i>properties</i> will be listed.	
Server	Data Access PercentDeadband	Support PercentDeadband filter when monitoring <i>AnalogItem</i> Type <i>Variables</i> .	
Server	Data Access Semantic Changes	Support semantic changes of <i>AnalogItem</i> Type items (<i>EURange Property</i> and/or <i>EngineeringUnits Property</i>). Support semantic change <i>StatusCode</i> bits where appropriate.	
Server	Data Access TwoState	Support <i>TwoStateDiscrete</i> Type <i>Variables</i> with corresponding <i>Properties</i> .	
Server	Data Access MultiState	Support <i>MultiStateDiscrete</i> Type <i>Variables</i> with corresponding <i>Properties</i> .	
Server	Data Access MultiStateValueDiscrete	Support <i>MultiStateValueDiscrete</i> Type <i>Variables</i> with corresponding <i>Properties</i> .	
Server	Data Access ArrayItemType	Provide <i>Variables</i> of <i>ArrayItem</i> Type or one of its subtypes (<i>YArrayItem</i> Type, <i>XYArrayItem</i> Type, <i>ImageArray</i> Type, <i>CubeArray</i> Type and <i>NDimensionArray</i> Type). The supported subtypes will be listed. Support for this type includes supporting all of the mandatory <i>properties</i> including <i>AxisInformation</i> .	
Server	Data Access Complex Number	Supports the Complex Number data type. This data type is available for any variable types that do not have other explicit restrictions.	
Server	Data Access DoubleComplex Number	Supports the DoubleComplex Number data type. This data type is available for any variable types that do not have other explicit restrictions.	
Client	Data Access Client Basic	Understand the <i>DataAccess Variable</i> Types. Make use of the standard <i>Properties</i> if applicable.	
Client	Data Access Client AnalogItems	Understand <i>AnalogItem</i> Type <i>Variables</i> with corresponding <i>Properties</i> .	
Client	Data Access Client TwoState	Understand <i>TwoStateDiscrete</i> Type <i>Variables</i> with corresponding <i>Properties</i> .	
Client	Data Access Client MultiState	Understand <i>MultiStateDiscrete</i> Type <i>Variables</i> with corresponding <i>Properties</i> .	
Client	Data Access Client MultiStateValueDiscrete	Understand <i>MultiStateValueDiscrete</i> Type <i>Variables</i> with corresponding <i>Properties</i> .	
Client	Data Access Client Deadband	Use PercentDeadband to filter value changes of <i>AnalogItem</i> Type <i>Variables</i> .	
Client	Data Access Client SemanticChange	Recognize the semantic change bit in the <i>StatusCode</i> while monitoring items and take proper action. Typically, the <i>Client</i> has to re-read <i>Properties</i> that define type-specific semantic like the <i>EURange</i> and <i>EngineeringUnits</i> <i>Properties</i> .	

Table 16 describes *Alarm* and *Conditions* information model related items that can be profiled. The details of this model are defined in IEC 62541-9. *Servers* that deal with *Alarm* and *Conditions* would expose this information model and *Clients* that process *Alarms* and *Conditions* would utilize this information model.

Table 16 – Alarms and Conditions

Category	Title	Description	Derived
Server	A & C Basic	Supports <i>Alarm</i> & <i>Condition</i> model <i>ConditionType</i> .	
Server	A & C Enable	Supports Enable and Disable Methods.	
Server	A & C Refresh	Supports <i>ConditionRefresh Method</i> and the concept of a refresh.	
Server	A & C Refresh2	Supports <i>ConditionRefresh2 Method</i> and the concept of a monitored item-based refresh.	
Server	A & C Instances	Support exposing of A&C <i>Condition</i> s instances in the <i>AddressSpace</i> .	
Server	A & C ConditionClasses	Supports multiple <i>Condition</i> classes for grouping and filtering of <i>Alarms</i> .	
Server	A & C <i>Condition</i> Sub-Classes	Support assigning multiple <i>Condition</i> sub-classes for grouping and filtering of <i>Alarms</i> .	
Server	A & C Acknowledge	~~Support Acknowledge, includes Acknowledge Method, Acknowledgeable Type.~~ Supports the Acknowledge concept, Acknowledge <i>Method</i> , and AcknowledgeableCondition Type.	
Server	A & C Confirm	~~Support confirming Conditions, includes Confirm Method.~~ Supports the concept of Confirm and the Confirm <i>Method</i> .	
Server	A & C Comment	~~Support Comments, includes AddComment Method.~~ Supports the concept of Comments and the AddComment <i>Method</i> .	
Server	A & C <i>Alarm</i>	Supports the mandatory features of the AlarmCondition Type.	
Server	A & C <i>Alarm</i> Metrics	~~Support for Basic Alarm functionality, including active, inactive states.~~ Support the collection of alarm metrics data as defined in IEC 62541-9. This implies one or more instances of the AlarmMetrics Type.	
Server	A & C Branch	~~Support for Alarm Branches which includes previous Condition Instances, i.e. conditions instance other than the current condition that still requires some operator action, such as acknowledgement or a dialog.~~ Support for branching of <i>Condition</i> Types and any subtypes, such as AcknowledgeableConditionType and AlarmConditionType etc.	
Server	A & C Shelving	Support the concept of shelving ~~mode, including~~ and the TimedShelve, OneShotShelve and Unshelve Methods.	
Server	A & C Suppression	Support the SuppressedState.	
Server	A & C Suppression by Operator	Support the Suppress and UnSuppress <i>Methods</i> to allow an operator control over the SuppressedState.	
Server	A & C Silencing	Support the concept of silencing and the Silence Method.	
Server	A & C Out Of Service	Support the OutOfService state and the OutOfService Method.	
Server	A & C On-Off Delay	Support the OnDelay and OffDelay Properties to eliminate nuisance <i>Alarms</i> .	
Server	A & C Re-Alarming	Support the ReAlarmTime and ReAlarmRepeatCount Properties that define automatic re-annunciation of <i>Alarms</i> in certain conditions.	
Server	A & C First in Group <i>Alarm</i>	Support the "FirstInGroup" elements for an <i>Alarm</i> , indicating which <i>Alarm</i> of a group was the trigger.	
Server	A & C Audible Sound	Support the AudibleSound <i>Property</i> . This <i>Property</i> contains the sound file that is to be played if an audible <i>Alarm</i> is to be generated.	
Server	A & C Exclusive Level	Supports Exclusive Level <i>Alarm</i> type.	
Server	A & C Exclusive Limit	Supports Exclusive Limit <i>Alarms</i> . A <i>Server</i> that supports this ~~must~~ shall support at least one of the sub-types: Level, Deviation or RateofChange.	

Category	Title	Description	Derived
Server	A & C Exclusive Deviation	Supports Exclusive Deviation <i>Alarm</i> type.	
Server	A & C Exclusive RateOfChange	Supports Exclusive RateOfChange <i>Alarm</i> type.	
Server	A & C Non-Exclusive Limit	Supports Non-Exclusive Limit <i>Alarms</i> . A <i>Server</i> that supports this ~~must~~ shall support at least one of the sub-types: Level, Deviation or RateOfChange.	
Server	A & C Non-Exclusive Level	Supports Non-Exclusive Level <i>Alarm</i> type.	
Server	A & C Non-Exclusive Deviation	Supports Non-Exclusive Deviation <i>Alarm</i> type.	
Server	A & C Non-Exclusive RateOfChange	Supports Non-Exclusive RateOfChange <i>Alarm</i> type.	
Server	A & C Discrete	Supports Discrete <i>Alarm</i> types.	
Server	A & C OffNormal	Supports OffNormalAlarmType.	
Server	A & C SystemOffNormal	Supports SystemOffNormalAlarmType.	
Server	A & C Trip	Supports Trip <i>Alarm</i> type.	
Server	A & C Discrepancy	Supports Discrepancy <i>Alarm</i> type.	
Server	A & C Dialog	Supports DialogConditionType including Respond <i>Method</i> .	
Server	A & C CertificateExpiration	Supports CertificateExpirationAlarmType.	
Server	A & E Wrapper Mapping	The <i>Server</i> uses the COM A&E mapping specified in the annex of IEC 62541-9 to map OPC-COM Events to A&C Events. This includes <i>Condition</i> Class mapping.	
Client	A & C Basic <i>Client</i>	Uses the <i>Alarm</i> & <i>Condition</i> model ConditionType.	
Client	A & C Enable <i>Client</i>	Uses Enable and Disable <i>Methods</i> .	
Client	A & C Refresh <i>Client</i>	Uses ConditionRefresh <i>Method</i> and the concept of a refresh.	
Client	A & C Refresh2 <i>Client</i>	Uses ConditionRefresh2 <i>Method</i> and the concept of a monitored item-based refresh.	
Client	A & C Instances <i>Client</i>	Uses A&C ~~Conditions that~~ <i>Condition</i> instances when they are exposed in the <i>AddressSpace</i> .	
Client	A & C ConditionClasses <i>Client</i>	Uses <i>Condition</i> classes to group <i>Alarms</i> .	
Client	A & C <i>Condition</i> Sub-Classes <i>Client</i>	Uses <i>Condition</i> sub-classes to group or filter <i>Alarms</i> .	
Client	A & C Acknowledge <i>Client</i>	~~Uses Acknowledge, including Acknowledge Method, Acknowledgeable type.~~ Understands the Acknowledge concept and the AcknowledgeableCondition Type, and uses the Acknowledge <i>Method</i> if requested.	
Client	A & C Confirm <i>Client</i>	~~Uses confirming Conditions, including Confirm Method.~~ Understands the concept of confirming <i>Conditions</i> and uses the Confirm <i>Method</i> .	
Client	A & C Comment <i>Client</i>	~~Uses Comments, including AddComment Method.~~ Understands the concept of Comments and uses the AddComment <i>Method</i> .	
Client	A & C <i>Alarm</i> <i>Client</i>	Understands the concept of <i>Alarms</i> and uses the mandatory features of the AlarmCondition Type.	
Client	A & C <i>Alarm</i> Metrics <i>Client</i>	~~Uses Basic Alarm functionality, including active, inactive states.~~ Understand and use <i>Alarm</i> metrics data as defined in IEC 62541-9. This implies discovery of instances of the AlarmMetricsType that can exist anywhere in the HasNotifier hierarchy.	
Client	A & C Branch <i>Client</i>	~~Uses Alarm Branches which included previous Condition instances, i.e. conditions instance other than the current condition that still requires some action, such as acknowledgement or confirmation.~~ Can make use of and process <i>Condition</i> Branches, including all actions associated with previous <i>Condition</i> instances.	

Category	Title	Description	Derived
<i>Client</i>	A & C Shelving <i>Client</i>	~~Uses the shelving model, including the TimedShelve, OneShotShelve and Unshelve Methods.~~ Understand the shelving model and use the TimedShelve, OneShotShelve and Unshelve Methods.	
<i>Client</i>	A & C Suppression <i>Client</i>	Understand the SuppressedState model.	
<i>Client</i>	A & C Suppression by Operator <i>Client</i>	Use the Suppress and UnSuppress <i>Methods</i> to allow an operator control over the SuppressedState.	
<i>Client</i>	A & C Silencing <i>Client</i>	Understand the SilencedState model and use the Silence Method.	
<i>Client</i>	A & C Out Of Service <i>Client</i>	Understand the OutOfServiceState model and use the OutOfService Method.	
<i>Client</i>	A & C On-Off Delay <i>Client</i>	Uses the OnDelay and OffDelay Properties to eliminate nuisance <i>Alarms</i> .	
<i>Client</i>	A & C Re-Alarming <i>Client</i>	Understand and use the ReAlarmTime and ReAlarmRepeatCount Properties. Configure the ReAlarmTime <i>Property</i> for automatic re-annunciation of an <i>Alarm</i> . Note that configuration is only possible for <i>Servers</i> that expose <i>Alarm</i> instances.	
<i>Client</i>	A & C First in Group <i>Alarm Client</i>	Use the "FirstInGroup" elements for an <i>Alarm</i> to determine which <i>Alarm</i> of a group was the trigger.	
<i>Client</i>	A & C Audible Sound <i>Client</i>	Use the AudibleSound <i>Property</i> and – if reported – play the sound file.	
<i>Client</i>	A & C Exclusive Level <i>Client</i>	Uses Exclusive Level <i>Alarms</i> ~~as defined.~~	
<i>Client</i>	A & C Exclusive Limit <i>Client</i>	Uses Exclusive Limit <i>Alarms</i> . Requires that at least one of the sub-types be used.	
<i>Client</i>	A & C Exclusive Deviation <i>Client</i>	Uses Exclusive Deviation <i>Alarms</i> .	
<i>Client</i>	A & C Exclusive RateOfChange <i>Client</i>	Uses Exclusive RateOfChange <i>Alarms</i> .	
<i>Client</i>	A & C Non-Exclusive Level <i>Client</i>	Uses Non-Exclusive Level <i>Alarms</i> .	
<i>Client</i>	A & C Non-Exclusive Limit <i>Client</i>	Uses Non-Exclusive Limit <i>Alarms</i> . Requires that at least one of the sub-types be used.	
<i>Client</i>	A & C Non-Exclusive Deviation <i>Client</i>	Uses Non-Exclusive Deviation <i>Alarms</i> .	
<i>Client</i>	A & C Non-Exclusive RateOfChange <i>Client</i>	Uses Non-Exclusive RateOfChange <i>Alarms</i> .	
<i>Client</i>	A & C Discrete <i>Client</i>	Uses Discrete <i>Alarm</i> types.	
<i>Client</i>	A & C OffNormal <i>Client</i>	Uses OffNormalAlarmType.	
<i>Client</i>	A & C SystemOffNormal <i>Client</i>	Uses SystemOffNormalAlarmType.	
<i>Client</i>	A & C Trip <i>Client</i>	Uses TripAlarmType.	
<i>Client</i>	A & C Discrepancy <i>Client</i>	Uses Discrepancy <i>Alarm</i> type.	
<i>Client</i>	A & C Dialog <i>Client</i>	Uses DialogConditionType including Respond <i>Method</i> .	
<i>Client</i>	A & C CertificateExpiration <i>Client</i>	Uses CertificateExpirationAlarmType.	

Table 17 describes Historical Data Access information model related items that can be profiled. The details of this model are defined in IEC 62541-11. *Servers* that support some level of historical data would expose this information model and *Clients* that utilize historical data would utilize this information model.

Table 17 – Historical Access

Category	Title	Description	Derived
Server	Historical Access Read Raw	General support for basic historical access, reading raw data using the ReadRawModifiedDetails structure. Where the time range is specified using a start time, stop time and number of values (a minimum of two of the three parameters ~~must~~ shall be provided) and the ReadModified flag is set to False.	
Server	Historical Access Data Max Nodes Read Continuation Point	Supports enough continuation points to cover the number of supported points indicated in the MaxNodesPerHistoryReadData Server OperationLimits ~~parameter~~ <i>Property</i> for historical data access.	
Server	Historical Access Time Instance	Supports reading historical data at a specified instance in time using the ReadAtTimeDetails structure.	
Server	Historical Access Aggregates	Supports reading one or more Aggregates of historical values of <i>Variables</i> using the ReadProcessedDetails structure. At least one of the Aggregates described in IEC 62541-13 ~~must shall~~ be supported. ~~The complete list will be shown in the Software Certificate.~~	
Server	Historical Access Insert Value	Supports inserting historical values of <i>Variables</i> .	
Server	Historical Access Delete Value	Supports deleting historical values of <i>Variables</i> .	
Server	Historical Access Update Value	Supports updating historical values of <i>Variables</i> .	
Server	Historical Access Replace Value	Supports replacing historical values of <i>Variables</i> .	
Server	Historical Access Modified Values	Supports maintaining old values for historical data that have been updated and the retrieval of these values using the ReadRawModifiedDetails structure (ReadModified flag set to true).	
Server	Historical Access Annotations	Supports the entry and retrieval of Annotations for historical data. The retrieval is accomplished using the standard historical read raw functionality (ReadRawModifiedDetails). The entry uses the standard historical update (UpdateStructureDataDetails) functionality.	
Server	Historical Access ServerTimestamp	Supports providing a ServerTimestamp (as well as the default SourceTimestamp).	
Server	Historical Access Structured Data Read Raw	Supports ReadRawModified historical access for structured data. Supporting the structure for an annotation is not considered supporting generic structured data.	
Server	Historical Access Structured Data Time Instance	Supports historical access for structured data. Supporting ReadAtTimeDetails for structured data. Supporting the structure for an annotation is not considered supporting generic structured data.	
Server	Historical Access Structured Data Insert	Supports historical access for structured data. Inserting Structured data. Supporting the structure for an annotation is not considered supporting generic structured data.	
Server	Historical Access Structured Data Delete	Supports historical access for structured data. Delete of existing data. Supporting the structure for an annotation is not considered supporting generic structured data.	
Server	Historical Access Structured Data Update	Supports historical access for structured data. Updates of existing data. Supporting the structure for an annotation is not considered supporting generic structured data.	
Server	Historical Access Structured Data Replace	Supports replacing structured historical data. Supporting the structure for an annotation is not considered supporting generic structured data.	
Server	Historical Access Structured Data Read Modified	Supports maintaining old values for historical structured data that have been updated and the retrieval of these values. Using the ReadRawModifiedDetails structure (ReadModified flag set to true) for structured data. Supporting the structure for an annotation is not considered supporting generic structured data.	

Category	Title	Description	Derived
Server	Historical Access Events	Supports the retrieval of historical Events using the ReadEventDetails structure. This includes support for simple filtering of Events. The <i>Event</i> fields that are stored are server specific, but at least the mandatory fields of BaseEventType are required.	
Server	Historical Access Event Max Events Read Continuation Point	Supports enough continuation points to cover the number of supported <i>Event</i> reads indicated in the MaxNodesPerHistoryReadEvents <i>Server</i> OperationLimits ~~parameter~~ <i>Property</i> for Historical <i>Event</i> access.	
Server	Historical Access Insert <i>Event</i>	Supports inserting historical Events.	
Server	Historical Access Update <i>Event</i>	Supports updating historical Events.	
Server	Historical Access Replace <i>Event</i>	Supports replacing historical Events.	
Server	Historical Access Delete <i>Event</i>	Supports deleting of historical Events.	
Client	Historical Access Client Browse	Uses the View <i>Service</i> Set to discover <i>Nodes</i> with historical data.	
Client	Historical Access Client Read Raw	Uses the HistoryRead <i>Service</i> to read raw historical data using the ReadRawModifiedDetails Structure (ReadModified Flag set to False).	
Client	Historical Access Client Read Modified	Uses the HistoryRead <i>Service</i> to read modified historical data using the ReadRawModifiedDetails Structure (ReadModified Flag set to True).	
Client	Historical Access Client Read Aggregates	Uses the HistoryRead <i>Service</i> to read Aggregated historical data. This includes using at least one of the Aggregates defined in IEC 62541-13. ~~The complete list of Aggregates used by the Client is included in the results of this ConformanceUnit.~~	
Client	Historical Access Client Structure Data Raw	Uses the HistoryRead <i>Service</i> to read raw historical data using the ReadRawModifiedDetails Structure (ReadModified Flag set to False) for structured data.	
Client	Historical Access Client Structure Data Read Modified	Uses the HistoryRead <i>Service</i> to read modified structured historical data using the ReadRawModifiedDetails Structure (ReadModified Flag set to True).	
Client	Historical Access Client Structure Data Insert	Uses the HistoryUpdate <i>Service</i> to insert historical data values for structured data.	
Client	Historical Access Client Structure Data Delete	Uses the HistoryUpdate <i>Service</i> to delete historical data values for structured data.	
Client	Historical Access Client Structure Data Update	Uses the HistoryUpdate <i>Service</i> to update historical data values for structured data.	
Client	Historical Access Client Structure Data Replace	Uses the HistoryUpdate <i>Service</i> to replace historical data values for structured data.	
Client	Historical Access Client Structure Data Time Instance	Reads historical data at a specified instance in time for structured data. Using the ReadAtTimeDetails structure.	
Client	Historical Access Client Read Events	Uses the HistoryRead <i>Service</i> to read historical <i>Event</i> data using the ReadEventDetails Structure.	
Client	Historical Access Client Event Inserts	Uses the HistoryUpdate <i>Service</i> to insert historical Events.	
Client	Historical Access Client Event Updates	Uses the HistoryUpdate <i>Service</i> to update historical Events.	
Client	Historical Access Client Event Replaces	Uses the HistoryUpdate <i>Service</i> to replace historical Events.	

Category	Title	Description	Derived
<i>Client</i>	Historical Access <i>Client Event Deletes</i>	Uses the HistoryUpdate <i>Service</i> to delete historical Events.	
<i>Client</i>	Historical Access <i>Client Data Insert</i>	Uses the HistoryUpdate <i>Service</i> to insert historical data values.	
<i>Client</i>	Historical Access <i>Client Data Delete</i>	Uses the HistoryUpdate <i>Service</i> to delete historical data values.	
<i>Client</i>	Historical Access <i>Client Data Update</i>	Uses the HistoryUpdate <i>Service</i> to update historical data values.	
<i>Client</i>	Historical Access <i>Client Data Replace</i>	Uses the HistoryUpdate <i>Service</i> to replace historical data values.	
<i>Client</i>	Historical Access <i>Client Annotations</i>	Enters and retrieves Annotations of historical data. The retrieval is accomplished using the standard historical read raw functionality (ReadRawModifiedDetails). The entry uses the standard Historical Update (UpdateStructureDataDetails) functionality.	
<i>Client</i>	Historical Access <i>Client Time Instance</i>	Reads historical data at a specified instance in time using the ReadAtTimeDetails structure.	
<i>Client</i>	Historical Access <i>Client Server Timestamp</i>	Uses the ServerTimestamp (as well as the default SourceTimestamp), if it is provided by the <i>Server</i> .	

Table 18 describes Aggregate related items that can be profiled. *Servers* that support the Aggregates would expose this functionality and *Clients* that utilize Aggregates would implement some of this functionality.

Table 18 – Aggregates

Category	Title	Description	Derived
Server	Aggregate Master Configuration	Supports ~~at least one master~~ an AggregateConfigurationType Object as part of the ~~Server configuration~~ HistoricalServerCapabilities (defined in IEC 62541-11).	
Server	Aggregate ~~optional~~ Historical Configuration	Supports at least one ~~optional~~ AggregateConfigurationType Object. ~~Optional~~ AggregateConfigurationType Objects occur ~~at different levels from the master~~ AggregateConfigurationType Object as part of an HistoricalConfiguration Object, allowing Variable specific configurations.	
Server	Aggregate – Interpolative	Supports the Interpolative Aggregate for Historical access.	
Server	Aggregate – Average	Supports the Average Aggregate for Historical access.	
Server	Aggregate – TimeAverage	Supports the TimeAverage Aggregate for Historical access.	
Server	Aggregate – TimeAverage2	Supports the TimeAverage2 Aggregate for Historical access.	
Server	Aggregate – Total	Supports the Total Aggregate for Historical access.	
Server	Aggregate – Total2	Supports the Total2 Aggregate for Historical access.	
Server	Aggregate – Minimum	Supports the Minimum Aggregate for Historical access.	
Server	Aggregate – MinimumActualTime	Supports the MinimumActualTime Aggregate for Historical access.	
Server	Aggregate – Minimum2	Supports the Minimum2 Aggregate for Historical access.	
Server	Aggregate – MinimumActualTime2	Supports the MinimumActualTime2 Aggregate for Historical access.	
Server	Aggregate – Maximum	Supports the Maximum Aggregate for Historical access.	
Server	Aggregate – MaximumActualTime	Supports the MaximumActualTime Aggregate for Historical access.	
Server	Aggregate – Maximum2	Supports the Maximum2 Aggregate for Historical access.	
Server	Aggregate – MaximumActualTime2	Supports the MaximumActualTime2 Aggregate for Historical access.	
Server	Aggregate – Range	Supports the Range Aggregate for Historical access.	
Server	Aggregate – Range2	Supports the Range2 Aggregate for Historical access.	
Server	Aggregate – Count	Supports the Count Aggregate for Historical access.	
Server	Aggregate – DurationInStateZero	Supports the DurationInStateZero Aggregate for Historical access.	
Server	Aggregate – DurationInStateNonZero	Supports the DurationInStateNonZero Aggregate for Historical access.	
Server	Aggregate – NumberOfTransitions	Supports the NumberOfTransitions Aggregate for Historical access.	
Server	Aggregate – Start	Supports the Start Aggregate for Historical access.	

Category	Title	Description	Derived
Server	Aggregate – StartBound	Supports the StartBound Aggregate for Historical access.	
Server	Aggregate – End	Supports the End Aggregate for Historical access.	
Server	Aggregate – EndBound	Supports the EndBound Aggregate for Historical access.	
Server	Aggregate – Delta	Supports the Delta Aggregate for Historical access.	
Server	Aggregate – DeltaBounds	Supports the DeltaBounds Aggregate for Historical access.	
Server	Aggregate – DurationGood	Supports the DurationGood Aggregate for Historical access.	
Server	Aggregate – DurationBad	Supports the DurationBad Aggregate for Historical access.	
Server	Aggregate – PercentGood	Supports the PercentGood Aggregate for Historical access.	
Server	Aggregate – PercentBad	Supports the PercentBad Aggregate for Historical access.	
Server	Aggregate – WorstQuality	Supports the WorstQuality Aggregate for Historical access.	
Server	Aggregate – WorstQuality2	Supports the WorstQuality2 Aggregate for Historical access.	
Server	Aggregate – AnnotationCount	Supports the AnnotationCount Aggregate for Historical access.	
Server	Aggregate – StandardDeviationSample	Supports the StandardDeviationSample Aggregate for Historical access.	
Server	Aggregate – VarianceSample	Supports the VarianceSample Aggregate for Historical access.	
Server	Aggregate – StandardDeviationPopulation	Supports the StandardDeviationPopulation for Historical access.	
Server	Aggregate – VariancePopulation	Supports the VariancePopulation for Historical access.	
Server	Aggregate – Custom	The <i>Server</i> supports custom Aggregates for Historical access that do not have standard tests defined. These Aggregates are list as untested by this <i>ConformanceUnit</i> .	
Server	Aggregate <i>Subscription</i> – Filter	Supports Aggregate subscription filters which requires at least one of the defined Aggregates is supported as defined in IEC 62541-13.	
Server	Aggregate <i>Subscription</i> – Interpolative	Supports subscription filter for the Interpolative Aggregate.	
Server	Aggregate <i>Subscription</i> – Average	Supports subscription filter for the Average Aggregate.	
Server	Aggregate <i>Subscription</i> – TimeAverage	Supports subscription filter for the TimeAverage Aggregate.	
Server	Aggregate <i>Subscription</i> – TimeAverage2	Supports subscription filter for the TimeAverage2 Aggregate.	
Server	Aggregate <i>Subscription</i> – Total	Supports subscription filter for the Total Aggregate.	
Server	Aggregate <i>Subscription</i> – Total2	Supports subscription filter for the Total2 Aggregate.	
Server	Aggregate <i>Subscription</i> – Minimum	Supports subscription filter for the Minimum Aggregate.	

Category	Title	Description	Derived
Server	Aggregate <i>Subscription</i> – MinimumActualTime	Supports subscription filter for the MinimumActualTime Aggregate.	
Server	Aggregate <i>Subscription</i> – Minimum2	Supports subscription filter for the Minimum2 Aggregate.	
Server	Aggregate <i>Subscription</i> – MinimumActualTime2	Supports subscription filter for the MinimumActualTime2 Aggregate.	
Server	Aggregate <i>Subscription</i> – Maximum	Supports subscription filter for the Maximum Aggregate.	
Server	Aggregate <i>Subscription</i> – MaximumActualTime	Supports subscription filter for the MaximumActualTime Aggregate.	
Server	Aggregate <i>Subscription</i> – Maximum2	Supports subscription filter for the Maximum2 Aggregate.	
Server	Aggregate <i>Subscription</i> – MaximumActualTime2	Supports subscription filter for the MaximumActualTime2 Aggregate.	
Server	Aggregate <i>Subscription</i> – Range	Supports subscription filter for the Range Aggregate.	
Server	Aggregate <i>Subscription</i> – Range2	Supports subscription filter for the Range2 Aggregate.	
Server	Aggregate <i>Subscription</i> – Count	Supports subscription filter for the Count Aggregate.	
Server	Aggregate <i>Subscription</i> – DurationInStateZero	Supports subscription filter for the DurationInStateZero Aggregate.	
Server	Aggregate <i>Subscription</i> – DurationInStateNonZero	Supports subscription filter for the DurationInStateNonZero Aggregate.	
Server	Aggregate <i>Subscription</i> – NumberOfTransitions	Supports subscription filter for the NumberOfTransitions Aggregate.	
Server	Aggregate <i>Subscription</i> – Start	Supports subscription filter for the Start Aggregate.	
Server	Aggregate <i>Subscription</i> – StartBound	Supports subscription filter for the StartBound Aggregate.	
Server	Aggregate <i>Subscription</i> – End	Supports subscription filter for the End Aggregate.	
Server	Aggregate <i>Subscription</i> – EndBound	Supports subscription filter for the EndBound Aggregate.	
Server	Aggregate <i>Subscription</i> – Delta	Supports subscription filter for the Delta Aggregate.	
Server	Aggregate <i>Subscription</i> – DeltaBounds	Supports subscription filter for the DeltaBounds Aggregate.	
Server	Aggregate <i>Subscription</i> – DurationGood	Supports subscription filter for the DurationGood Aggregate.	
Server	Aggregate <i>Subscription</i> – DurationBad	Supports subscription filter for the DurationBad Aggregate.	
Server	Aggregate <i>Subscription</i> – PercentGood	Supports subscription filter for the PercentGood Aggregate.	
Server	Aggregate <i>Subscription</i> – PercentBad	Supports subscription filter for the PercentBad Aggregate.	
Server	Aggregate <i>Subscription</i> – WorstQuality	Supports subscription filter for the WorstQuality Aggregate.	
Server	Aggregate <i>Subscription</i> – WorstQuality2	Supports subscription filter for the WorstQuality2 Aggregate.	
Server	Aggregate <i>Subscription</i> – AnnotationCount	Supports subscription filter for the AnnotationCount Aggregate.	
Server	Aggregate <i>Subscription</i> – StandardDeviationSample	Supports subscription filter for the StandardDeviationSample Aggregate.	
Server	Aggregate <i>Subscription</i> – VarianceSample	Supports subscription filter for the VarianceSample Aggregate.	

Category	Title	Description	Derived
Server	Aggregate <i>Subscription</i> – StandardDeviationPopulation	Supports subscription filter for the StandardDeviationPopulation Aggregate.	
Server	Aggregate <i>Subscription</i> – VariancePopulation	Supports subscription filter for the VariancePopulation Aggregate.	
Server	Aggregate <i>Subscription</i> – Custom	The <i>Server</i> supports subscribing to custom Aggregates that do not have standard tests defined. These Aggregates are listed as untested by this <i>ConformanceUnit</i> .	
Client	Aggregate – <i>Client</i> Usage	Uses Historical access to Aggregate which requires at least one of the defined Aggregates is supported as defined in IEC 62541-13.	
Client	Aggregate – <i>Client</i> Interpolative	Uses Historical access to the Interpolative Aggregate.	
Client	Aggregate – <i>Client</i> Average	Uses Historical access to the Average Aggregate.	
Client	Aggregate – <i>Client</i> TimeAverage	Uses Historical access to the TimeAverage Aggregate.	
Client	Aggregate – <i>Client</i> TimeAverage2	Uses Historical access to the TimeAverage2 Aggregate.	
Client	Aggregate – <i>Client</i> Total	Uses Historical access to the Total Aggregate.	
Client	Aggregate – <i>Client</i> Total2	Uses Historical access to the Total2 Aggregate.	
Client	Aggregate – <i>Client</i> Minimum	Uses Historical access to the Minimum Aggregate.	
Client	Aggregate – <i>Client</i> MinimumActualTime	Uses Historical access to the MinimumActualTime Aggregate.	
Client	Aggregate – <i>Client</i> Minimum2	Uses Historical access to the Minimum2 Aggregate.	
Client	Aggregate – <i>Client</i> MinimumActualTime2	Uses Historical access to the MinimumActualTime2 Aggregate.	
Client	Aggregate – <i>Client</i> Maximum	Uses Historical access to the Maximum Aggregate.	
Client	Aggregate – <i>Client</i> MaximumActualTime	Uses Historical access to the MaximumActualTime Aggregate.	
Client	Aggregate – <i>Client</i> Maximum2	Uses Historical access to the Maximum2 Aggregate.	
Client	Aggregate – <i>Client</i> MaximumActualTime2	Uses Historical access to the MaximumActualTime2 Aggregate.	
Client	Aggregate – <i>Client</i> Range	Uses Historical access to the Range Aggregate.	
Client	Aggregate – <i>Client</i> Range2	Uses Historical access to the Range2 Aggregate.	
Client	Aggregate – <i>Client</i> Count	Uses Historical access to the Count Aggregate.	
Client	Aggregate – <i>Client</i> DurationInStateZero	Uses Historical access to the DurationInStateZero Aggregate.	
Client	Aggregate – <i>Client</i> DurationInStateNonZero	Uses Historical access to the DurationInStateNonZero Aggregate.	
Client	Aggregate – <i>Client</i> NumberOfTransitions	Uses Historical access to the NumberOfTransitions Aggregate.	
Client	Aggregate – <i>Client</i> Start	Uses Historical access to the Start Aggregate.	

Category	Title	Description	Derived
<i>Client</i>	Aggregate – <i>Client</i> StartBound	Uses Historical access to the StartBound Aggregate.	
<i>Client</i>	Aggregate – <i>Client</i> End	Uses Historical access to the End Aggregate.	
<i>Client</i>	Aggregate – <i>Client</i> EndBound	Uses Historical access to the EndBound Aggregate.	
<i>Client</i>	Aggregate – <i>Client</i> Delta	Uses Historical access to the Delta Aggregate.	
<i>Client</i>	Aggregate – <i>Client</i> DeltaBounds	Uses Historical access to the DeltaBounds Aggregate.	
<i>Client</i>	Aggregate – <i>Client</i> DurationGood	Uses Historical access to the DurationGood Aggregate.	
<i>Client</i>	Aggregate – <i>Client</i> DurationBad	Uses Historical access to the DurationBad Aggregate.	
<i>Client</i>	Aggregate – <i>Client</i> PercentGood	Uses Historical access to the PercentGood Aggregate.	
<i>Client</i>	Aggregate – <i>Client</i> PercentBad	Uses Historical access to the PercentBad Aggregate.	
<i>Client</i>	Aggregate – <i>Client</i> WorstQuality	Uses Historical access to the WorstQuality Aggregate.	
<i>Client</i>	Aggregate – <i>Client</i> WorstQuality2	Uses Historical access to the WorstQuality2 Aggregate.	
<i>Client</i>	Aggregate – <i>Client</i> AnnotationCount	Uses Historical access to the AnnotationCount Aggregate.	
<i>Client</i>	Aggregate – <i>Client</i> StandardDeviationSample	Uses Historical access to the StandardDeviationSample Aggregate.	
<i>Client</i>	Aggregate – <i>Client</i> VarianceSample	Uses Historical access to the VarianceSample Aggregate.	
<i>Client</i>	Aggregate – <i>Client</i> StandardDeviationPopulation	Uses Historical access to the StandardDeviationPopulation Aggregate.	
<i>Client</i>	Aggregate – <i>Client</i> VariancePopulation	Uses Historical access to the VariancePopulation Aggregate.	
<i>Client</i>	Aggregate – <i>Client</i> Custom Aggregates	The <i>Client</i> can make use of all custom Aggregates in the list of Aggregates, via Historical access, exposed by the <i>Server</i> . This includes displaying or utilizing the data in some manner.	
<i>Client</i>	Aggregate Subscription – <i>Client</i> Filter	Subscribes for data using Aggregate filters which requires at least one of the Aggregates defined in IEC 62541-13 is supported.	
<i>Client</i>	Aggregate Subscription – <i>Client</i> Interpolative	Subscribes for data using the Interpolative Aggregate filter.	
<i>Client</i>	Aggregate Subscription – <i>Client</i> Average	Subscribes for data using the Average Aggregate filter.	
<i>Client</i>	Aggregate Subscription – <i>Client</i> TimeAverage	Subscribes for data using the TimeAverage Aggregate filter.	
<i>Client</i>	Aggregate Subscription – <i>Client</i> TimeAverage2	Subscribes for data using the TimeAverage2 Aggregate filter.	
<i>Client</i>	Aggregate Subscription – <i>Client</i> Total	Subscribes for data using the Total Aggregate filter.	
<i>Client</i>	Aggregate Subscription – <i>Client</i> Total2	Subscribes for data using the Total2 Aggregate filter.	
<i>Client</i>	Aggregate Subscription – <i>Client</i> Minimum	Subscribes for data using the Minimum Aggregate filter.	

Category	Title	Description	Derived
Client	Aggregate Subscription – Client MinimumActualTime	Subscribes for data using the MinimumActualTime Aggregate filter.	
Client	Aggregate Subscription – Client Minimum2	Subscribes for data using the Minimum2 Aggregate filter.	
Client	Aggregate Subscription – Client MinimumActualTime2	Subscribes for data using the MinimumActualTime2 Aggregate filter.	
Client	Aggregate Subscription – Client Maximum	Subscribes for data using the Maximum Aggregate filter.	
Client	Aggregate Subscription – Client MaximumActualTime	Subscribes for data using the MaximumActualTime Aggregate filter.	
Client	Aggregate Subscription – Client MaximumActualTime2	Subscribes for data using the MaximumActualTime2 Aggregate filter.	
Client	Aggregate Subscription – Client Maximum2	Subscribes for data using the Maximum2 Aggregate filter.	
Client	Aggregate Subscription – Client Range	Subscribes for data using the Range Aggregate filter.	
Client	Aggregate Subscription – Client Range2	Subscribes for data using the Range2 Aggregate filter.	
Client	Aggregate Subscription – Client Count	Subscribes for data using the Count Aggregate filter.	
Client	Aggregate Subscription – Client DurationInStateZero	Subscribes for data using the DurationInStateZero Aggregate filter.	
Client	Aggregate Subscription – Client DurationInStateNonZero	Subscribes for data using the DurationInStateNonZero Aggregate filter.	
Client	Aggregate Subscription – Client NumberOfTransitions	Subscribes for data using the NumberOfTransitions Aggregate filter.	
Client	Aggregate Subscription – Client Start	Subscribes for data using the Start Aggregate filter.	
Client	Aggregate Subscription – Client StartBound	Subscribes for data using the StartBound Aggregate filter.	
Client	Aggregate Subscription – Client End	Subscribes for data using the End Aggregate filter.	
Client	Aggregate Subscription – Client EndBound	Subscribes for data using the EndBound Aggregate filter.	
Client	Aggregate Subscription – Client Delta	Subscribes for data using the Delta Aggregate filter.	
Client	Aggregate Subscription – Client DeltaBounds	Subscribes for data using the DeltaBounds Aggregate filter.	
Client	Aggregate Subscription – Client DurationGood	Subscribes for data using the DurationGood Aggregate filter.	
Client	Aggregate Subscription – Client DurationBad	Subscribes for data using the DurationBad Aggregate filter.	
Client	Aggregate Subscription – Client PercentGood	Subscribes for data using the PercentGood Aggregate filter.	
Client	Aggregate Subscription – Client PercentBad	Subscribes for data using the PercentBad Aggregate filter.	
Client	Aggregate Subscription – Client WorstQuality	Subscribes for data using the WorstQuality Aggregate filter.	
Client	Aggregate Subscription – Client WorstQuality2	Subscribes for data using the WorstQuality2 Aggregate filter.	
Client	Aggregate Subscription – Client AnnotationCount	Subscribes for data using the AnnotationCount Aggregate filter.	
Client	Aggregate Subscription – Client ~~StandardDevSample~~ StandardDeviationSample	Subscribes for data using the StandardDeviationSample Aggregate filter.	

Category	Title	Description	Derived
Client	Aggregate Subscription – Client VarianceSample	Subscribes for data using the VarianceSample Aggregate filter.	
Client	Aggregate Subscription – Client StandardDevPopulationStandardDeviationPopulation	Subscribes for data using the StandardDeviationPopulation Aggregate filter.	
Client	Aggregate Subscription – Client VariancePopulation	Subscribes for data using the VariancePopulation Aggregate filter.	
Client	Aggregate Subscription – Client Custom Aggregates	The <i>Client</i> supports subscribing to all custom Aggregates in the list of Aggregates exposed by the <i>Server</i> . This includes displaying or utilizing the data in some manner.	

Table 19 describes auditing related items that can be profiled. Most full function *Servers* would support these features, although some resource constrained *Servers* may not provide this functionality. *Clients* that are security aware or are used to support security logging would support these features.

Table 19 – Auditing

Category	Title	Description	Derived
Server	Auditing Base	Support AuditEvents. The list of supported AuditEvents shall be verified during certification testing and will be shown in the ~~Software Certificate~~ certification test result. Base AuditEvents are defined in IEC 62541-3 and in IEC 62541-5.	
Client	Auditing Client Audit ID	<i>Client</i> supports generating AuditEvents ids and providing them to <i>Servers</i> .	
Client	Auditing Client Subscribes	The <i>Client</i> supports subscribing for AuditEvents and storing / processing them in a secure manner.	

Table 20 describes Redundancy related items that are profiled. *Servers* that support redundancy would support appropriate *ConformanceUnits* based on the type of redundancy they support. *Clients* that are capable of handling redundancy would support the appropriate *ConformanceUnits* based on the type of redundancy they support.

Table 20 – Redundancy

Category	Title	Description	Derived
Server	Redundancy Server	Supports <i>Server</i> based redundancy.	
Server	Redundancy Server Transparent	Supports transparent <i>Server</i> redundancy.	
Client	Redundancy Client	<i>Client</i> supports <i>Client</i> redundancy. <i>Clients</i> that support <i>Client</i> redundancy can failover to another <i>Client</i> (requires some out of band communication).	
Client	Redundancy Client Switch	<i>Clients</i> supporting this <i>ConformanceUnit</i> monitor the redundancy status for non-transparent redundancy <i>Servers</i> and switch to the backup <i>Server</i> when they recognize a change in server status.	

Table 21 describes items for a Global *Discovery Server* (GDS). *Servers* that act as a GDS would support these *ConformanceUnits*.

Table 21 – Global Discovery Server

Category	Title	Description
Global Directory Service	GDS Application Directory	Supports the Directory <i>Object</i> with all Methods like RegisterApplication and QueryServers.
Global Directory Service	GDS Query Applications	Supports the QueryApplications <i>Method</i> on the Directory <i>Object</i> specified in IEC 62541-12.
Global Directory Service	GDS LDS-ME Connectivity	The GDS can be configured to use specific LDS-ME installations for semi-automatic application registration for all <i>Servers</i> on a subnet.
Global Directory Service	GDS <i>Certificate</i> Manager Pull Model	This Conformance Unit requires support of the complete Information Model and <i>Services</i> for <i>Certificate</i> management including the Pull Model as specified in IEC 62541-12.
Global Directory Service	GDS <i>Certificate</i> Manager Push Model	This Conformance Unit requires use of the complete Information model and <i>Services</i> for the <i>Certificate</i> management Push Model as specified in IEC 62541-12.
Global Directory Service	GDS Key Credential Service Pull Model	This Conformance Unit requires support of the complete Information Model and <i>Services</i> for KeyCredential Pull Management as specified in IEC 62541-12.
Global Directory Service	GDS Key Credential Service Push Model	This Conformance Unit requires use of the complete Information model and <i>Services</i> for KeyCredential Push Management as specified in IEC 62541-12.
Global Directory Service	GDS Authorization Service Server	This Conformance Unit requires support of AuthorizationServiceType Objects as specified in IEC 62541-12. UA Clients use the RequestAccessToken Method on these Objects to request an Access Token from an Identity Provider.

5.5 Miscellaneous

Table 22 describes miscellaneous *ConformanceUnits*.

~~Each table includes a listing of the Profile Category to which a ConformanceUnit belongs, the title and description of the ConformanceUnit and a column that indicates if the ConformanceUnit is derived from another ConformanceUnit. A ConformanceUnit that is derived from another ConformanceUnit includes all of the same tests as its parent plus one or more additional TestCases. These TestCases can only further restrict the existing TestCases.~~

Table 22 – Miscellaneous

Category	Title	Description	Derived
~~Client, Server~~	Documentation – Supported <i>Profiles</i>	The documentation includes a description of the profiles supported by the product. This description includes the level of Certification testing the product has passed.	
~~Client, Server~~	Documentation – Multiple Languages	The documentation is available in multiple languages. The results of this conformance unit include the list of supported languages.	
~~Client, Server~~	Documentation – Users Guide	The application includes documentation that describes the available functionality provided by the application. For <i>Servers</i> it includes a summary of all functionality provided by the <i>Server</i> .	
~~Client, Server~~	Documentation – On-line	The documentation provided by the application is available in electronic format as part of the application. The electronic documentation, could be a WEB page, installed document or CD/DVD, but in all case it can be accessed from the application or from a link installed with the application.	
~~Client, Server~~	Documentation – Installation	The application includes installation instructions that are sufficient to easily install the application. This includes descriptions of any and all possible configuration items. Instructions for loading or configuring security related items such as Application Instance <i>Certificates</i> .	
~~Client, Server~~	Documentation – Trouble Shooting Guide	The application includes documentation that describes typical problems a user may encounter and actions that the user could perform to resolve the problem. It could also describe tip, tricks or other actions that could help a user diagnose or fix a problem. It could also describe tools or other items that can be used in diagnosing or repairing problems. The actual Trouble Shooting Guide can be part of other documentation, but should be complete enough to provide useful information to a novice user.	
<i>Client</i>	Documentation <i>Client</i> – Supported <i>Profiles</i>	The documentation includes a description of the profiles supported by the product. This description includes any software certificates that describes the level of Certification testing the product has passed.	
<i>Client</i>	Documentation <i>Client</i> – Multiple Languages	The documentation is available in multiple languages. The results of this conformance unit include the list of supported languages.	
<i>Client</i>	Documentation <i>Client</i> – Users Guide	The application includes documentation that describes the available functionality provided by the application. For client applications, this includes any operator restrictions or general functionality that the client application makes use of.	
<i>Client</i>	Documentation <i>Client</i> – On-line	The documentation provided by the application is available in electronic format as part of the application. The electronic documentation could be a WEB page, installed document or CD/DVD, but in all cases it can be accessed from the application or from a link installed with the application.	
<i>Client</i>	Documentation <i>Client</i> – Installation	The application includes installation instructions that are sufficient to easily install the application. This includes descriptions of any and all possible configuration items. Instructions for loading or configuring security related items such as Application Instance <i>Certificates</i> .	
<i>Client</i>	Documentation <i>Client</i> – Trouble Shooting Guide	The application includes documentation that describes typical problems a user may encounter and actions that the user could perform to resolve the problem. It could also describe tips, tricks or other actions that could help a user diagnose or fix a problem. It could also describe tools or other items that can be used in diagnosing or repairing problems. The actual Trouble Shooting Guide can be part of other documentation, but should be complete enough to provide useful information to a novice user.	
Security	Best Practice – Timeouts	The user is able to configure reasonable timeouts for Secure Channels, sessions and subscriptions to limit Denial of <i>Service</i> and resource consumption issues (see IEC 62541-2 for additional details).	
Security	Best Practice – Strict Message Handling	The application assures that messages that are illegally or incorrectly formed are rejected with appropriate error code or appropriate actions as specified in IEC 62541-4 and IEC 62541-6.	

Category	Title	Description	Derived
Security	Best Practice – Random Numbers	All random numbers that are required for security use appropriate cryptographic library based random number generators.	
Security	Best Practice – Administrative Access	The <i>Server</i> and <i>Client</i> allow for appropriate restriction of access to administrative personnel. This includes multiple levels of administrative access on platforms that support multiple administrative roles (such as Windows or Linux).	
Security	Best Practice – Alarm Handling	A <i>Server</i> should restrict critical alarm functionality to users that have the appropriate rights to perform these actions. This would include disabling or alarms, shelving of alarms and generation of dialog messages. It would also include other security related functionality such maintaining appropriate timeouts for shelving and dialogs and preventing an overload of dialog messages.	
Security	Best Practice – Audit Events	Subscriptions for Audit Events are restricted to authorized personnel. A <i>Server</i> may also reject a <i>Subscription</i> for Audit Events that is not over a Secure Channel if one is available.	
Security	Best Practice – Audit Events <i>Client</i>	Audit tracking system connects to a <i>Server</i> using a Secure Channel and under the appropriate administrative rights to allow access to Audit Events.	

6 Profiles

6.1 Overview

This clause includes a listing of the categories that a *Profile* can be grouped into, a list of named *Profiles* and the detailed listing of each *Profile* including directly defined *ConformanceUnits* and any sub *Profiles* that are included in the *Profile*.

6.2 Profile list

Table 23 lists *Profiles*. The *Profile* table is ordered by *Profile* category and then alphabetically by the name of the *Profile*. The table includes a list of categories the *Profile* is associated with and a URI. The URI is used to uniquely identify a *Profile*. The URI shall be able to be used to access the information provided in this document with regard to the given *Profile* in an on-line display. ~~This URI is also included in the SoftwareCertificate associated with the Profile. The URI is case sensitive.~~

An application (*Client* or *Server*) shall implement all of the *ConformanceUnits* in a *Profile* in order to be compliant with the *Profile*. Some *Profiles* contain optional *ConformanceUnits*. An optional *ConformanceUnit* means that an application has the option to not support the *ConformanceUnit*. However, if supported, the application shall pass all tests associated with the *ConformanceUnit*. For example, some *ConformanceUnits* require specific information model items to be available. They are, therefore, listed as optional in order to allow for the information model items to be omitted. If a *Server* desires to be listed as supporting the optional *ConformanceUnit* then it shall include any required information model items in the configuration provided for certification testing. ~~The support for optional ConformanceUnits is described in the certificate that is generated by the associated testing. Optional ConformanceUnits are clearly identified in this document and as part of the SoftwareCertificate that describes the Profiles supported by a product. The SoftwareCertificate must show all optional ConformanceUnits and if they are support. Any on-line displays that list the Profiles a product supports must also include the optional ConformanceUnits.~~ The test result that is generated by the certification testing lists all optional *ConformanceUnits* and whether they are supported or not by the tested UA application. Some *ConformanceUnits* also include lists of supported DataTypes or optional Subtypes; the lists are handled in the same manner as optional *ConformanceUnits*. All reporting requirements for optional *ConformanceUnits* also apply to these lists of supported DataTypes or Subtypes.

Table 23 – Profile list

Profile	Related Category	URI
Core 2017 Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/Core2017Facet
Sessionless Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/SessionLess
Reverse Connect Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/ReverseConnect
Base Server Behaviour Facet	Server	http://opcfoundation.org/UA-Profile/Server/Behaviour
Request State Change Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/RequestStateChange
Subnet Discovery Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/SubnetDiscovery
Global Certificate Management Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/GlobalCertificateManagement
Authorization Service Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/AuthorizationServiceConfiguration
KeyCredential Service Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/KeyCredentialManagement
Attribute WriteMask Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/AttributeWriteMask
File Access Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/FileAccess
Documentation Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/Documentation
Embedded DataChange Subscription Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/EmbeddedDataChangeSubscription
Standard DataChange Subscription 2017 Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/StandardDataChangeSubscription2017
Enhanced DataChange Subscription 2017 Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/EnhancedDataChangeSubscription2017
Durable Subscription Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/DurableSubscription
Data Access Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/DataAccess
ComplexType 2017 Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/ComplexTypes2017
Standard Event Subscription Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/StandardEventSubscription
Address Space Notifier Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/AddressSpaceNotifier
A & C Base Condition Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/ACBaseCondition
A & C Refresh2 Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/ACRefresh2
A & C Address Space Instance Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/ACAddressSpaceInstance
A & C Enable Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/ACEnable
A & C AlarmMetrics Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/ACAlarmMetrics
A & C Alarm Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/ACAlarm
A & C Acknowledgeable Alarm Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/ACAckAlarm
A & C Exclusive Alarming Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/ACExclusiveAlarming
A & C Non-Exclusive Alarming Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/ACNon-ExclusiveAlarming
A & C Previous Instances Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/ACPreviousInstances

Profile	Related Category	URI
A & C Dialog Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/ACDialog
A & C CertificateExpiration Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/ACCertificateExpiration
A & E Wrapper Facet	Server	http://opcfoundation.org/UA-Profile/Server/AEWrapper
Method Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/Methods
Auditing Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/Auditing
Node Management Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/NodeManagement
User Role Base Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/UserRoleBase
User Role Management Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/UserRoleManagement
State Machine Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/StateMachine
Client Redundancy Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/ClientRedundancy
Redundancy Transparent Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/TransparentRedundancy
Redundancy Visible Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/VisibleRedundancy
Historical Raw Data Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/HistoricalRawData
Historical Aggregate Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/AggregateHistorical
Historical Data AtTime Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/HistoricalDataAtTime
Historical Access Modified Data Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/HistoricalModifiedData
Historical Annotation Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/HistoricalAnnotation
Historical Data Insert Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/HistoricalDataInsert
Historical Data Update Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/HistoricalDataUpdate
Historical Data Replace Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/HistoricalDataReplace
Historical Data Delete Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/HistoricalDataDelete
Historical Access Structured Data Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/HistoricalStructuredData
Base Historical Event Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/BaseHistoricalEvent
Historical Event Update Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/HistoricalEventUpdate
Historical Event Replace Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/HistoricalEventReplace
Historical Event Insert Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/HistoricalEventInsert
Historical Event Delete Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/HistoricalEventDelete
Aggregate Subscription Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/AggregateSubscription
Nano Embedded Device 2017 Server Profile	Server	http://opcfoundation.org/UA-Profile/Server/NanoEmbeddedDevice2017

Profile	Related Category	URI
Micro Embedded Device 2017 Server Profile	Server	http://opcfoundation.org/UA-Profile/Server/MicroEmbeddedDevice2017
Embedded 2017 UA Server Profile	Server	http://opcfoundation.org/UA-Profile/Server/EmbeddedUA2017
Standard 2017 UA Server Profile	Server	http://opcfoundation.org/UA-Profile/Server/StandardUA2017
Core 2017 Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/Core2017
Sessionless Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/SessionLess
Reverse Connect Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/ReverseConnect
Base Client Behaviour Facet	Client	http://opcfoundation.org/UA-Profile/Client/Behaviour
Discovery Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/Discovery
Subnet Discovery Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/SubnetDiscovery
Global Discovery Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/GlobalDiscovery
Global Certificate Management Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/GlobalCertificateManagement
KeyCredential Service Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/KeyCredentialManagement
Access Token Request Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/AccessTokenRequest
AddressSpace Lookup Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/AddressSpaceLookup
Request State Change Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/RequestStateChange
File Access Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/FileAccess
Entry Level Support 2015 Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/EntryLevelSupport2015
Multi-Server Client Connection Facet	Client	http://opcfoundation.org/UA-Profile/Client/MultiServer
Documentation – Client	Client	http://opcfoundation.org/UA-Profile/Client/Documentation
Attribute Read Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/AttributeRead
Attribute Write Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/AttributeWrite
DataChange Subscriber Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/DataChangeSubscriber
Durable Subscription Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/DurableSubscription
DataAccess Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/DataAccess
Event Subscriber Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/EventSubscriber
Base Event Processing Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/BaseEventProcessing
Notifier and Source Hierarchy Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/NotifierAndSourceHierarchy
A & C Base Condition Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/ACBaseCondition
A & C Refresh2 Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/ACRefresh2
A & C Address Space Instance Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/ACAddressSpaceInstance
A & C Enable Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/ACEnable
A & C AlarmMetrics Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/ACAlarmMetrics
A & C Alarm Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/ACAlarm

Profile	Related Category	URI
A & C Exclusive Alarming Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/ACExclusiveAlarming
A & C Non-Exclusive Alarming Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/ACNon-ExclusiveAlarming
A & C Previous Instances Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/ACPreviousInstances
A & C Dialog Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/ACDialog
A & C CertificateExpiration Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/ACCertificateExpiration
A & E Proxy Facet	Client	http://opcfoundation.org/UA-Profile/Client/AEProxy
Method Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/Method
Auditing Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/Auditing
Node Management Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/NodeManagement
Advanced Type Programming Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/TypeProgramming
User Role Management Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/UserRoleManagement
State Machine Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/StateMachine
Diagnostic Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/Diagnostic
Redundant Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/Redundancy
Redundancy Switch Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/RedundancySwitch
Historical Access Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/HistoricalAccess
Historical Data AtTime Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/HistoricalAccessAtTime
Historical Aggregate Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/HistoricalAccessAggregate
Historical Annotation Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/HistoricalAnnotation
Historical Access Modified Data Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/HistoricalAccessModifiedData
Historical Data Insert Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/HistoricalInsertData
Historical Data Update Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/HistoricalUpdateData
Historical Data Replace Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/HistoricalReplaceData
Historical Data Delete Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/HistoricalDeleteData
Historical Access Client Server Timestamp Facet	Client	http://opcfoundation.org/UA-Profile/Client/HistoricalServerTimeStamp
Historical Structured Data Access Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/HistoricalAccessStructuredData
Historical Structured Data AtTime Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/HistoricalAtTimeStructuredData
Historical Structured Data Modified Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/HistoricalModifiedStructuredData
Historical Structured Data Insert Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/HistoricalInsertStructuredData

Profile	Related Category	URI
Historical Structured Data Update Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/HistoricalUpdateStructuredData
Historical Structured Data Replace Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/HistoricalReplaceStructuredData
Historical Structured Data Delete Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/HistoricalDeleteStructuredData
Historical Events Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/HistoricalEvents
Historical Event Insert Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/HistoricalInsertEvents
Historical Event Update Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/HistoricalUpdateEvents
Historical Event Replace Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/HistoricalReplaceEvents
Historical Event Delete Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/HistoricalDeleteEvents
Aggregate Subscriber Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/AggregateSubscriber
Standard UA Client Profile	Client	http://opcfoundation.org/UA-Profile/Client/Standard
Standard UA Client 2017 Profile	Client	http://opcfoundation.org/UA-Profile/Client/Standard2017
UA-TCP UA-SC UA-Binary	Transport	http://opcfoundation.org/UA-Profile/Transport/uatcp-uasc-uabinary
HTTPS UA-Binary	Transport	http://opcfoundation.org/UA-Profile/Transport/https-uabinary
HTTPS UA-XML	Transport	http://opcfoundation.org/UA-Profile/Transport/https-uasoapxml
HTTPS UA-JSON	Transport	http://opcfoundation.org/UA-Profile/Transport/https-uajson
WSS UA-SC UA-Binary	Transport	http://opcfoundation.org/UA-Profile/Transport/wss-uasc-uabinary
WSS UA-JSON	Transport	http://opcfoundation.org/UA-Profile/Transport/wss-uajson
Security User Access Control Full	Security, Server	http://opcfoundation.org/UA-Profile/Security/UserAccessFull
Security User Access Control Base	Security, Server	http://opcfoundation.org/UA-Profile/Security/UserAccessBase
Security Time Synchronization	Security	http://opcfoundation.org/UA-Profile/Security/TimeSync
Best Practice – Audit Events	Security, Server	http://opcfoundation.org/UA-Profile/Security/BestPracticeAuditEvents
Best Practice – Alarm Handling	Security, Server	http://opcfoundation.org/UA-Profile/Security/BestPracticeAlarmHandling
Best Practice – Program Access	Security, Server	http://opcfoundation.org/UA-Profile/Security/BestPracticeProgramAccess
Best Practice – Random Numbers	Security	http://opcfoundation.org/UA-Profile/Security/BestPracticeRandomNumbers
Best Practice – Timeouts	Security	http://opcfoundation.org/UA-Profile/Security/BestPracticeTimeouts
Best Practice – Administrative Access	Security	http://opcfoundation.org/UA-Profile/Security/BestPracticeAdministrativeAccess
Best Practice – Strict Message Handling	Security, Server	http://opcfoundation.org/UA-Profile/Security/BestPracticeStrictMessage
Best Practice – Alarm Handling Client	Client, Security	http://opcfoundation.org/UA-Profile/Security/BestPracticeAlarmHandlingClient
Best Practice – Audit Events Client	Client, Security	http://opcfoundation.org/UA-Profile/Security/BestPracticeAuditEventsClient

Profile	Related Category	URI
TransportSecurity—TLS 1.0	Security	http://opcfoundation.org/UA-Profile/TransportSecurity/TLS-1-0
TransportSecurity—TLS 1.1	Security	http://opcfoundation.org/UA-Profile/TransportSecurity/TLS-1-1
TransportSecurity – TLS 1.2	Security	http://opcfoundation.org/UA-Profile/TransportSecurity/TLS-1-2
TransportSecurity – TLS 1.2 with PFS	Security	http://opcfoundation.org/UA-Profile/TransportSecurity/TLS-1-2-PFS
SOAP-HTTP-WS-SC-UA-XML	Transport	http://opcfoundation.org/UA-Profile/Transport/soaphttp-wssc-uaxml
SOAP-HTTP-WS-SC-UA-Binary	Transport	http://opcfoundation.org/UA-Profile/Transport/soaphttp-wssc-uabinary
SOAP-HTTP-WS-SC-UA-XML-UA-Binary	Transport	http://opcfoundation.org/UA-Profile/Transport/soaphttp-wssc-uaxml-uabinary
SecurityPolicy – None	Security	http://opcfoundation.org/UA/SecurityPolicy#None
SecurityPolicy—Basic128Rsa15	Security	http://opcfoundation.org/UA/SecurityPolicy#Basic128Rsa15
SecurityPolicy	Security	http://opcfoundation.org/UA/SecurityPolicy#Basic256
SecurityPolicy [A] – Aes128-Sha256-RsaOaep	Security	http://opcfoundation.org/UA/SecurityPolicy#Aes128_Sha256_RsaOaep
SecurityPolicy [B] – Basic256Sha256	Security	http://opcfoundation.org/UA/SecurityPolicy#Basic256Sha256
SecurityPolicy – Aes256-Sha256-RsaPss	Security	http://opcfoundation.org/UA/SecurityPolicy#Aes256_Sha256_RsaPss
User Token – Anonymous Facet	Security	http://opcfoundation.org/UA-Profile/Security/UserToken/Anonymous
User Token – User Name Password Server Facet	Security, Server	http://opcfoundation.org/UA-Profile/Security/UserToken/Server/UserNamePassword
User Token – X509 Certificate Server Facet	Security, Server	http://opcfoundation.org/UA-Profile/Security/UserToken/Server/X509Certificate
User Token – Issued Token Server Facet	Security, Server	http://opcfoundation.org/UA-Profile/Security/UserToken/Server/IssuedToken
User Token – Issued Token Windows Server Facet	Security, Server	http://opcfoundation.org/UA-Profile/Security/UserToken/Server/IssuedTokenWindows
User Token – JWT Server Facet	Security	http://opcfoundation.org/UA-Profile/Security/UserToken/Server/JsonWebToken
User Token – User Name Password Client Facet	Client, Security	http://opcfoundation.org/UA-Profile/Security/UserToken/Client/UserNamePassword
User Token – X509 Certificate Client Facet	Client, Security	http://opcfoundation.org/UA-Profile/Security/UserToken/Client/X509Certificate
User Token – Issued Token Client Facet	Client, Security	http://opcfoundation.org/UA-Profile/Security/UserToken/Client/IssuedToken
User Token – Issued Token Windows Client Facet	Client, Security	http://opcfoundation.org/UA-Profile/Security/UserToken/Client/IssuedTokenWindows
User Token – JWT Client Facet	Security	http://opcfoundation.org/UA-Profile/Security/UserToken/Client/JsonWebToken
Global Discovery Server 2017 Profile	Global Directory Service, Server	http://opcfoundation.org/UA-Profile/Server/GlobalDiscovery2017
Global Discovery and Certificate Mgmt 2017 Server	Global Directory Service, Server	http://opcfoundation.org/UA-Profile/Server/GlobalDiscoveryAndCertificateManagement2017

Profile	Related Category	URI
Global Certificate Management Client 2017 Profile	Client, Global Directory Service	http://opcfoundation.org/UA-Profile/Client/GlobalCertificateManagement2017
Global Service Authorization Request Server Facet	Global Directory Service	http://opcfoundation.org/UA-Profile/Server/GlobalServiceAuthorization
Global Service KeyCredential Pull Facet	Global Directory Service	http://opcfoundation.org/UA-Profile/Server/GlobalServiceKeyCredentials
Global Service KeyCredential Push Facet	Global Directory Service	http://opcfoundation.org/UA-Profile/Client/GlobalServiceKeyCredentials

The contents of each of the listed *Profiles* will be described in a ~~tabular form in a separate section~~ subclause of 6.6 and where the title is equal to the *Profile* name. Each table may contain references to additional *Profiles* and or *ConformanceUnits*. If a *Profile* is referenced it means that it is completely included. The *ConformanceUnits* are referenced using their name and conformance group. For the details of the *ConformanceUnit* the reader should examine the *ConformanceUnit* details in ~~the appropriate conformance group section~~ clause 5.

6.3 Conventions for Profile definitions

Profiles have the following naming conventions:

- *Profiles* intended for OPC UA Servers contain the term *Server* in their titles,
- *Profiles* intended for OPC UA Clients contain the term *Client* in their titles
- The term Facet in the title of a *Profile* indicates that this *Profile* is expected to be part of another larger *Profile* or concerns a specific aspect of OPC UA. *Profiles* with the term Facet in their title are expected to be combined with other *Profiles* to define the complete functionality of an OPC UA Server or Client.

6.4 Profile versioning

Versioning of *Profile* is accomplished with a naming convention. Whenever a profile is revised, the year of the new revision is added to the name. Example:

Version 1	Core Server Facet
Version 2	Core 2017 Server Facet

6.5 Applications

A vendor that is developing a UA application, whether it is a *Server* application or a *Client* application, shall review the list of available *Profiles*. From this list the vendor shall select the *Profiles* that include the functionality required by the application. Typically this will be multiple *Profiles*. Conformance to a single *Profile* may not yield a complete application. In most cases multiple *Profiles* are needed to yield a useful application. All *Servers* and *Clients* shall support at least a core *Profile* (Core Server Facet or Core Client Facet) and at least one Transport *Profile*

For example an HMI *Client* application may choose to support the "Core Client Facet", the "UA-TCP UA-SC UA-Binary" *Profile*, the "Data Access Client Facet", the "DataChange Subscriber Client Facet" and the "Attribute Write Client Facet". If the *Client* is to be *TestLab* tested then it would also support "Base Client Behaviour" *Profile*. This list of *Profiles* would allow the *Client* to communicate with an OPC UA Server using UA-TCP/UA Security/UA binary. It would be able to subscribe for data, write to data and would support the DA data

model. It would also follow the best practice guideline for behaviour. Figure 2 illustrates the *Profile* hierarchy that this application may contain. This figure is only an illustration and the represented *Profiles* may change.

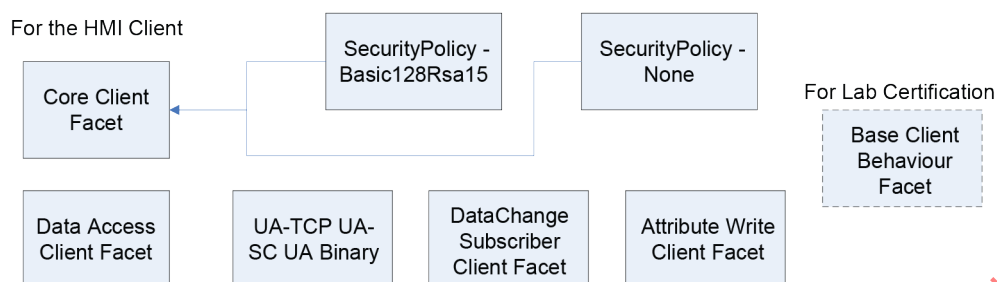


Figure 2 – HMI Client sample

All *Clients* should take into account the types of *Servers* and *Server Profiles* that they are targeted to support. Some *Servers* might not support *Subscriptions* and *Clients* should be able to fall back to Read *Services*.

A special case is a generic *Client* that is designed to communicate with a large number of *Servers* and therefore able to perform a broad range of functionality. "Standard UA *Client Profile*" has been defined for this kind of *Clients*.

Many *Clients*, however, will be specialized and do not need all of the functionality in the "Standard UA *Client Profile*" and thus would only support the limited set of functionality they require. A trend *Client*, for example, would only need functionality to subscribe to or read data.

Another example is an embedded device OPC UA *Server* application that may choose to support "Embedded UA *Server Profile*" and the "DataAccess *Server Facet Profile*". This device would be a resource constrained device that would support UA-TCP, UA-Security, UA Binary encoding, data subscriptions and the DA data model. It may not support the optional attribute write. Figure 3 illustrates the hierarchy that this application may contain. This figure is just an illustration and the represented *Profiles* may change.

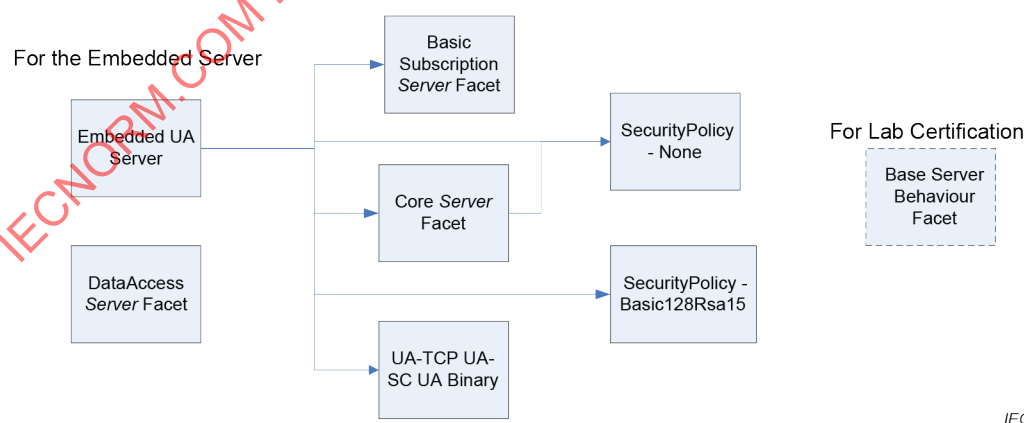


Figure 3 – Embedded Server sample

Another simple system *Server* application may choose to support: "Standard UA *Server Profile*" and the "DataAccess *Server Facet Profile*". If the *Server* is to be lab tested then it would also support "Base *Server Behaviour Profile*". This device would be a mid-level OPC UA *Server* that would support all that the embedded *Server* in the previous example supported and it would add support for an enhance level of the subscription service and

support for writes. Figure 4 illustrates the hierarchy that this application may contain. This figure is just an illustration and the represented *Profile* may change.

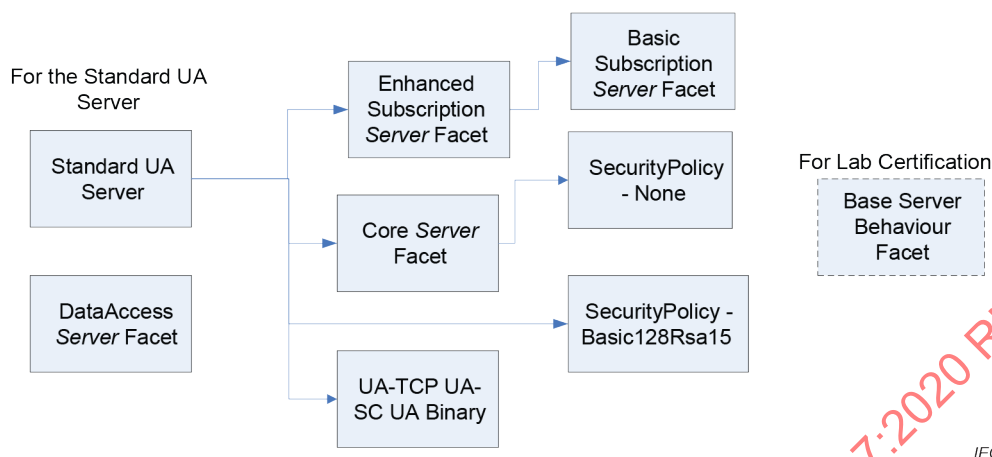


Figure 4 – Standard UA Server sample

If the example HMI *Client* were to connect to either of the example *Servers*, it may have to adjust its behavior based on the *Profile* reported by the respective *Servers*. If the HMI *Client* were communicating with the embedded device it would not be able to perform any write operations. It may also have to limit the number of subscriptions or sessions based on the performance limits of the *Server*. If the HMI *Client* is connected to the Standard *Server* it would be able to open additional windows, have higher limits on performance related items and it would be able to allow writes.

6.6 Profile tables

6.6.1 Introduction General

Subclauses 6.6.2 to 6.6.186 describe *Profiles* in a tabular format.

Each table contains three columns. The first column is a description of the conformance group that the *ConformanceUnit* is part of. This allows the reader to easily find the *ConformanceUnit*. This column may also state "*Profile*" in which case the listed item is not a *ConformanceUnit*, but an included *Profile*. The second column is a brief description of the *ConformanceUnit* or included *Profile*. The last column indicates if the *ConformanceUnit* is optional or required.

6.6.2 Core Server Facet

Core Server Facet has been deprecated as it is superseded with Core 2017 Server Facet in v1.04.

6.6.3 Core 2017 Server Facet

Table 24 describes the details of the Core 2017 *Server Facet*. This Facet defines the core functionality required for any UA *Server* implementation. The core functionality includes the ability to discover endpoints, establish secure communication channels, create Sessions, browse the *AddressSpace* and read and/or write to *Attributes* of *Nodes*. The key requirements are: support for a single *Session*, support for the *Server* and *Server Capabilities Object*, all mandatory *Attributes* for *Nodes* in the *AddressSpace* and authentication with *UserName* and *Password*. ~~Support for a type system is not required nor does the Server need to support encryption and signing of user identity tokens (This assumes the Server also supports a transport that provides security.) This Facet has been extended with additional Base Information ConformanceUnits. They are optional to provide backward compatibility. In the future the ConformanceUnit "Base Info Server Capabilities" will become required, and so it is~~

~~highly recommended that all Servers support it.~~ For broad applicability, it is recommended that Servers support multiple transport and security *Profiles*. This Facet supersedes the "Core Server Facet".

Table 24 – Core 2017 Server Facet

Group	Conformance Unit / Profile Title	Optional
<i>Profile</i>	SecurityPolicy – None	False
<i>Profile</i>	User Token – User Name Password Server Facet	False
Address Space Model	Address Space Atomicity	False
Address Space Model	Address Space Base	False
Address Space Model	Address Space Full Array Only	False
Attribute Services	Attribute Read	False
Attribute Services	Attribute Write Index	True
Attribute Services	Attribute Write Values	True
Base Information	Base Info Core Structure	False
Base Information	Base Info Estimated Return Time	True
Base Information	Base Info OptionSet	True
Base Information	Base Info Placeholder Modelling Rules	True
Base Information	Base Info Selection List	True
Base Information	Base Info Server Capabilities	True
Base Information	Base Info ValueAsText	True
Discovery Services	Discovery Find Servers Self	False
Discovery Services	Discovery Get Endpoints	False
Security	~~Security – No Application Authentication~~	~~True~~
Security	Security Administration	True
Security	Security Role Server Authorization	True
Session Services	Session Base	False
Session Services	Session General Service Behaviour	False
Session Services	Session Minimum 1	False
View Services	View Basic	False
View Services	View Minimum Continuation Point 01	False
View Services	View RegisterNodes	False
View Services	View TranslateBrowsePath	False

6.6.4 Sessionless Server Facet

Table 25 describes the details of the Sessionless Server Facet. Defines the use of Sessionless Service invocation in a Server.

Table 25 – Sessionless Server Facet

Group	Conformance Unit / Profile Title	Optional
Discovery Services	Discovery Get Endpoints SessionLess	False
Session Services	Session Sessionless Invocation	False

6.6.5 Reverse Connect Server Facet

Table 26 describes the details of the Reverse Connect *Server* Facet. This Facet defines support of reverse connectivity in a *Server*. Usually, a connection is opened by the *Client* before starting the UA-specific handshake. This will fail, however, when *Servers* are behind firewalls with no open ports to connect to. In the reverse connectivity scenario, the *Server* opens the connection and starts with a ReverseHello message requesting that the *Client* establish a Secure Channel using this connection.

Table 26 – Reverse Connect Server Facet

Group	Conformance Unit / Profile Title	Optional
Protocol and Encoding	Protocol Reverse Connect <i>Server</i>	False

6.6.6 Base Server Behaviour Facet

Table 27 describes the details of the Base *Server* Behaviour Facet. This Facet defines best practices for the configuration and management of *Servers* when they are deployed in a production environment. It provides the ability to enable or disable certain protocols, ~~to set the security level~~ and to configure the *Discovery Server* and specify where this *Server* shall be registered.

Table 27 – Base Server Behaviour Facet

Group	Conformance Unit / Profile Title	Optional
<i>Discovery Services</i>	<i>Discovery</i> Configuration	False
Protocol and Encoding	Protocol Configuration	False
Security	Security Administration	False
Security	Security Administration – XML Schema	False
Security	Security <i>Certificate</i> Administration	False

6.6.7 Request State Change Server Facet

Table 28 describes the details of the Request State Change *Server* Facet. This Facet specifies the support of the RequestServerStateChange *Method*.

Table 28 – Request State Change Server Facet

Group	Conformance Unit / Profile Title	Optional
Base Information	Base Info RequestServerStateChange <i>Method</i>	False

6.6.8 Subnet Discovery Server Facet

Table 29 describes the details of the Subnet *Discovery Server* Facet. Support of this Facet enables discovery of the *Server* on a subnet using mDNS. This functionality is only applicable when *Servers* do not register with an LDS.

Table 29 – Subnet Discovery Server Facet

Group	Conformance Unit / Profile Title	Optional
<i>Discovery Services</i>	<i>Discovery Server</i> Announcement using mDNS	False

6.6.9 Global Certificate Management Server Facet

Table 30 describes the details of the Global *Certificate* Management *Server* Facet. This Facet defines the capability to interact with a Global *Certificate* Management *Server* to obtain an initial or renewed *Certificate* and Trust Lists.

Table 30 – Global Certificate Management Server Facet

Group	Conformance Unit / <i>Profile</i> Title	Optional
Security	Push Model for Global <i>Certificate</i> and TrustList Management	False

6.6.10 Authorization Service Server Facet

Table 31 describes the details of the Authorization *Service* *Server* Facet. This Facet defines the support for configuring the necessary information to validate access tokens when presented by a Client during session establishment. Access Tokens are issued by Authorization *Services*.

Table 31 – Authorization Service Server Facet

Group	Conformance Unit / <i>Profile</i> Title	Optional
Security	Authorization Service Configuration Server	False

6.6.11 KeyCredential Service Server Facet

Table 32 describes the details of the KeyCredential *Service* *Server* Facet. This Facet defines the capability to interact with a KeyCredential *Service* to obtain KeyCredentials. For example KeyCredentials are needed to access an Authorization *Service* or a Broker. The KeyCredential *Service* is typically part of a system-wide tool, like a GDS that also manages Applications, Access Tokens, and *Certificates*.

Table 32 – KeyCredential Service Server Facet

Group	Conformance Unit / <i>Profile</i> Title	Optional
Security	Push Model for KeyCredential <i>Service</i>	False

6.6.12 Attribute WriteMask Server Facet

Table 33 describes the details of the *Attribute* WriteMask *Server* Facet. This Facet defines the capability to update characteristics of individual *Nodes* in the *AddressSpace* by allowing writing to *Node* *Attributes*. It requires support for authenticating user access as well as providing information related to access rights in the *AddressSpace* and actually restricting the access rights as described.

Table 33 – Attribute WriteMask Server Facet

Group	Conformance Unit / <i>Profile</i> Title	Optional
<i>Profile</i>	Security User Access Control Base	False
Address Space Model	Address Space UserWriteMask	False
Address Space Model	Address Space UserWriteMask Multilevel	True
Address Space Model	Address Space WriteMask	False

6.6.13 File Access Server Facet

Table 34 describes the details of the File Access *Server* Facet. This Facet specifies the support of exposing File information via the defined FileType. This includes reading of file as well as optionally writing of file data.

Table 34 – File Access Server Facet

Group	Conformance Unit / Profile Title	Optional
Base Information	Base Info FileType Base	False
Base Information	Base Info FileType Write	True

6.6.14 Documentation Server Facet

Table 35 describes the details of the Documentation *Server* Facet. This Facet defines a list of user documentation that a server application should provide.

Table 35 – Documentation Server Facet

Group	Conformance Unit / Profile Title	Optional
Miscellaneous	Documentation – Installation	False
Miscellaneous	Documentation – Multiple Languages	True
Miscellaneous	Documentation – On-line	True
Miscellaneous	Documentation – Supported Profiles	True
Miscellaneous	Documentation – Trouble Shooting Guide	True
Miscellaneous	Documentation – Users Guide	False

6.6.15 Embedded DataChange Subscription Server Facet

Table 36 describes the details of the Embedded DataChange *Subscription Server* Facet. This Facet specifies the minimum level of support for data change notifications within subscriptions. It includes limits which minimize memory and processing overhead required to implement the Facet. This Facet includes functionality to create, modify and delete Subscriptions and to add, modify and remove Monitored Items. As a minimum for each Session, Servers shall support one Subscription with up to two items, ~~but, republish buffering is not required.~~ In addition, support for two parallel Publish requests is required. This Facet is geared for a platform such as the one provided by the Micro Embedded Device *Server Profile* in which memory is limited and needs to be managed.

Table 36 – Embedded DataChange Subscription Server Facet

Group	Conformance Unit / Profile Title	Optional
Monitored Item Services	Monitor Basic	False
Monitored Item Services	Monitor Items 2	False
Monitored Item Services	Monitor QueueSize_1	False
Monitored Item Services	Monitor Value Change	False
Subscription Services	Subscription Basic	False
Subscription Services	Subscription Minimum 1	False
Subscription Services	Subscription Publish Discard Policy	False
Subscription Services	Subscription Publish Min 02	False

6.6.16 Standard DataChange Subscription Server Facet

Standard DataChange Subscription Server Facet has been deprecated as it is superseded with Standard DataChange Subscription 2017 Server Facet in v1.04.

6.6.17 Standard DataChange Subscription 2017 Server Facet

Table 37 describes the details of the Standard DataChange *Subscription 2017 Server Facet*. ~~This Facet specifies the standard support of subscribing to data changes. This Facet extends features and limits defined by the Embedded Data Change Subscription Facet. As a minimum, Servers shall support 2 Subscriptions with at least 100 items for at least half of the required Sessions. The 100 items shall be supported for at least half of the required Subscriptions. Queuing with up to two queued entries is required. Support of five parallel Publish requests per Session is required. This Facet also requires the support of the triggering service. This Facet has been updated to include optional ConformanceUnits to allow for backward compatibility. These optional ConformanceUnits are highly recommended, in that in a future release they will be made mandatory.~~ This Facet specifies the standard support of subscribing to data changes and extends features and limits defined by the Embedded Data Change Subscription Facet. See *ConformanceUnits* for these limits. Note that the *Method Call Service* is only required for the *Methods* defined in this Facet. This Facet supersedes the "Standard DataChange Subscription Server Facet".

Table 37 – Standard DataChange Subscription 2017 Server Facet

Group	Conformance Unit / Profile Title	Optional
<i>Profile</i>	Embedded DataChange Subscription Server Facet	False
Base Information	Base Info GetMonitoredItems <i>Method</i>	~~True~~ False
Base Information	Base Info ResendData <i>Method</i>	False
<i>Method Services</i>	<i>Method Call</i>	~~True~~ False
Monitored Item <i>Services</i>	Monitor Items 10	False
Monitored Item <i>Services</i>	Monitor Items 100	False
Monitored Item <i>Services</i>	Monitor MinQueueSize_02	False
Monitored Item <i>Services</i>	Monitor Triggering	False
Monitored Item <i>Services</i>	Monitored Items Deadband Filter	False
<i>Subscription Services</i>	<i>Subscription</i> Minimum 02	False
<i>Subscription Services</i>	<i>Subscription</i> Publish Min 05	False

6.6.18 Enhanced DataChange Subscription Server Facet

Enhanced DataChange *Subscription Server Facet* has been deprecated as it is superseded with Enhanced DataChange Subscription 2017 Server Facet in v1.04.

6.6.19 Enhanced DataChange Subscription 2017 Server Facet

Table 38 describes the details of the Enhanced DataChange *Subscription 2017 Server Facet*. This Facet specifies an enhanced support of subscribing to data changes. It is part of the Standard UA *Server 2017 Profile*. This Facet increases the limits defined by the Standard Data Change *Subscription 2017 Server Facet*.

Table 38 – Enhanced DataChange Subscription 2017 Server Facet

Group	Conformance Unit / Profile Title	Optional
<i>Profile</i>	Standard DataChange <i>Subscription 2017 Server Facet</i>	False
Monitored Item <i>Services</i>	Monitor Items 500	False
Monitored Item <i>Services</i>	Monitor MinQueueSize_05	False
<i>Subscription Services</i>	<i>Subscription</i> Minimum 05	False
<i>Subscription Services</i>	<i>Subscription</i> Publish Min 10	False

6.6.20 Durable Subscription Server Facet

Table 39 describes the details of the Durable *Subscription Server Facet*. This Facet specifies support of durable storage of data and events even when *Clients* are disconnected. This Facet implies support of any of the DataChange or *Event Subscription Facets*.

Table 39 – Durable Subscription Server Facet

Group	Conformance Unit / Profile Title	Optional
<i>Subscription Services</i>	<i>Subscription</i> Durable	False
<i>Subscription Services</i>	<i>Subscription</i> Durable StorageLevel High	True
<i>Subscription Services</i>	<i>Subscription</i> Durable StorageLevel Medium	True
<i>Subscription Services</i>	<i>Subscription</i> Durable StorageLevel Small	True

6.6.21 Data Access Server Facet

Table 40 describes the details of the Data Access *Server Facet*. This Facet specifies the support for an Information Model used to provide industrial automation data. This model defines standard structures for analog and discrete data items and their quality of service. This Facet extends the Core *Server Facet* which includes support of the basic *AddressSpace* behaviour.

Table 40 – Data Access Server Facet

Group	Conformance Unit / Profile Title	Optional
Data Access	Data Access AnalogItems	True
Data Access	Data Access ArrayItemType	True
Data Access	Data Access Complex Number	True
Data Access	Data Access DataItems	False
Data Access	Data Access DoubleComplex Number	True
Data Access	Data Access MultiState	True
Data Access	Data Access MultiStateValueDiscrete	True
Data Access	Data Access PercentDeadband	True
Data Access	Data Access Semantic Changes	True
Data Access	Data Access TwoState	True

6.6.22 ComplexType Server Facet

ComplexType *Server Facet* has been deprecated as it is superseded with ComplexType 2017 *Server Facet* in v1.04.

6.6.23 ComplexType 2017 Server Facet

Table 41 describes the details of the ComplexType 2017 Server Facet. This Facet extends the Core Server Facet to include *Variables* with ~~Complex~~ structured data, i.e. data that are composed of multiple elements such as a structure and where the individual elements are exposed as component variables. Support of this Facet requires the implementation of structured DataTypes and *Variables* that make use of these DataTypes. The Read, Write and Subscriptions service set shall support the encoding and decoding of these structured DataTypes. As an option the Server can also support alternate encodings, such as an XML encoding when the binary protocol is currently used and vice-versa.

Table 41 – ComplexType 2017 Server Facet

Group	Conformance Unit / Profile Title	Optional
Address Space Model	Address Space ~~Complex~~ DataTypes DataTypeDefinition Attribute	False
Attribute Services	Attribute Alternate Encoding	True
Attribute Services	Attribute Read Complex	False
Attribute Services	Attribute Write Complex	False
Monitored Item Services	Monitor Alternate Encoding	True
Monitored Item Services	Monitor Complex Value	True

6.6.24 Standard Event Subscription Server Facet

Table 42 describes the details of the Standard Event Subscription Server Facet. This Facet specifies the standard support for subscribing to events and is intended to supplement any of the FullFeatured Profiles. Support of this Facet requires the implementation of *Event Types* representing the Events that the Server can report and their specific fields. It also requires at least the Server Object to have the *EventNotifier Attribute* set. It includes the Services to Create, Modify and Delete Subscriptions and to Add, Modify and Remove Monitored Items for Object Nodes with an "EventNotifier Attribute". Creating a monitoring item may include a filter that includes SimpleAttribute FilterOperands and a select list of Operators. The operators include: Equals, IsNull, GreaterThan, LessThan, GreaterThanOrEqual, LessThanOrEqual, Like, Not, Between, InList, And, Or, Cast, BitwiseAnd, BitwiseOr and TypeOf. Support of more complex filters is optional. This Facet has been updated to include several optional Base Information ConformanceUnits. These ConformanceUnits are optional to allow for backward compatibility, in the future these optional ConformanceUnits will become required, and so it is highly recommended that all servers support them.

Table 42 – Standard Event Subscription Server Facet

Group	Conformance Unit / <i>Profile</i> Title	Optional
Address Space Model	Address Space Events	False
Base Information	Base Info Device Failure	True
Base Information	Base Info EventQueueOverflow EventType	True
Base Information	Base Info Progress Events	True
Base Information	Base Info SemanticChange	True
Base Information	Base Info System Status	True
Base Information	Base Info System Status Underlying System	True
Monitored Item <i>Services</i>	Monitor Basic	False
Monitored Item <i>Services</i>	Monitor Complex <i>Event</i> Filter	True
Monitored Item <i>Services</i>	Monitor Events	False
Monitored Item <i>Services</i>	Monitor Items 10	False
Monitored Item <i>Services</i>	Monitor QueueSize_ServerMax	False
<i>Subscription Services</i>	<i>Subscription</i> Basic	False
<i>Subscription Services</i>	<i>Subscription</i> Minimum 02	False
<i>Subscription Services</i>	<i>Subscription</i> Publish Discard Policy	False
<i>Subscription Services</i>	<i>Subscription</i> Publish Min 05	False

6.6.25 Address Space Notifier Server Facet

Table 43 describes the details of the Address Space Notifier *Server* Facet. This Facet requires the support of a hierarchy of *Object Nodes* that are notifiers and *Nodes* that are event sources. The hierarchy is commonly used as a way to organize a plant into areas that can be managed by different operators.

Table 43 – Address Space Notifier Server Facet

Group	Conformance Unit / <i>Profile</i> Title	Optional
Address Space Model	Address Space Notifier Hierarchy	False
Address Space Model	Address Space Source Hierarchy	False

6.6.26 A & C Base Condition Server Facet

Table 44 describes the details of the A & C Base *Condition Server* Facet. This Facet requires basic support for *Conditions*. Information about *Conditions* is provided through *Event* notifications and thus this Facet builds upon the Standard *Event Subscription Server* Facet. *Conditions* that are in an "interesting" state (as defined by the *Server*) can be refreshed using the Refresh *Method*, which requires support for the *Method Server* Facet. Optionally the server may also provide support for *Condition* classes

Table 44 – A & C Base Condition Server Facet

Group	Conformance Unit / Profile Title	Optional
Profile	Method Server Facet	False
Profile	Standard Event Subscription Server Facet	False
Alarms and Conditions	A & C Basic	False
Alarms and Conditions	A & C Condition Sub-Classes	True
Alarms and Conditions	A & C ConditionClasses	True
Alarms and Conditions	A & C Refresh	False

6.6.27 A & C Refresh2 Server Facet

Table 45 describes the details of the A & C Refresh2 Server Facet. This Facet enhances the A & C Base Condition Server Facet with support of the ConditionRefresh2 Method.

Table 45 – A & C Refresh2 Server Facet

Group	Conformance Unit / Profile Title	Optional
Profile	A & C Base Condition Server Facet	False
Alarms and Conditions	A & C Refresh2	False

6.6.28 A & C Address Space Instance Server Facet

Table 46 describes the details of the A & C Address Space Instance Server Facet. This Facet specifies the support required for a Server to expose Alarms and Conditions in its AddressSpace. This includes the A & C AddressSpace information model.

Table 46 – A & C Address Space Instance Server Facet

Group	Conformance Unit / Profile Title	Optional
Alarms and Conditions	A & C Instances	False

6.6.29 A & C Enable Server Facet

Table 47 describes the details of the A & C Enable Server Facet. This Facet requires the enabling and disabling of Conditions. This Facet builds upon the A&C Base Condition Server Facet. Enabling and disabling also requires that instances of these ConditionTypes exist in the AddressSpace since the enable Method can only be invoked on an instance of the Condition

Table 47 – A & C Enable Server Facet

Group	Conformance Unit / Profile Title	Optional
Profile	A & C Base Condition Server Facet	False
Alarms and Conditions	A & C Enable	False
Alarms and Conditions	A & C Instances	False

6.6.30 A & C AlarmMetrics Server Facet

Table 48 describes the details of the A & C AlarmMetrics Server Facet. This Facet requires support for AlarmMetrics. AlarmMetrics expose status and potential issues in the alarm

system. A *Server* can provide these metrics at various levels (operator station, plant area, overall system etc.).

Table 48 – A & C AlarmMetrics Server Facet

Group	Conformance Unit / Profile Title	Optional
<i>Alarms and Conditions</i>	A & C <i>Alarm Metrics</i>	False

6.6.31 A & C Alarm Server Facet

Table 49 describes the details of the A & C *Alarm Server Facet*. This Facet requires support for *Alarms*. *Alarms* extend the *ConditionType* by adding an *Active* state which indicates when something in the system requires attention by an Operator. This Facet builds upon the A&C *Base Condition Server Facet*. This facet requires that discrete *AlarmTypes* be supported, it also allows for optional support of shelving, alarm comments and other discrete *AlarmTypes* such as Trip or Off-Normal.

Table 49 – A & C Alarm Server Facet

Group	Conformance Unit / Profile Title	Optional
<i>Profile</i>	A & C <i>Base Condition Server Facet</i>	False
<i>Alarms and Conditions</i>	A & C <i>Alarm</i>	False
<i>Alarms and Conditions</i>	A & C Audible Sound	True
<i>Alarms and Conditions</i>	A & C Comment	True
<i>Alarms and Conditions</i>	A & C Discrepancy	True
<i>Alarms and Conditions</i>	A & C Discrete	False
<i>Alarms and Conditions</i>	A & C First in Group <i>Alarm</i>	True
<i>Alarms and Conditions</i>	A & C OffNormal	True
<i>Alarms and Conditions</i>	A & C On-Off Delay	True
<i>Alarms and Conditions</i>	A & C Out Of Service	True
<i>Alarms and Conditions</i>	A & C Re-Alarming	True
<i>Alarms and Conditions</i>	A & C Shelving	True
<i>Alarms and Conditions</i>	A & C Silencing	True
<i>Alarms and Conditions</i>	A & C Suppression	True
<i>Alarms and Conditions</i>	A & C Suppression by Operator	True
<i>Alarms and Conditions</i>	A & C SystemOffNormal	True
<i>Alarms and Conditions</i>	A & C Trip	True

6.6.32 A & C Acknowledgeable Alarm Server Facet

Table 50 describes the details of the A & C Acknowledgeable *Alarm Server Facet*. This Facet requires support for Acknowledgement of active *Alarms*. This Facet builds upon the A & C *Alarm Server Facet*. Acknowledgement requires support of the *Acknowledge Method* and the Acknowledged state. Support of the Confirmed state and the Confirm *Method* is optional.

Table 50 – A & C Acknowledgeable Alarm Server Facet

Group	Conformance Unit / Profile Title	Optional
<i>Profile</i>	A & C Alarm Server Facet	False
<i>Alarms and Conditions</i>	A & C Acknowledge	False
<i>Alarms and Conditions</i>	A & C Confirm	True

6.6.33 A & C Exclusive Alarming Server Facet

Table 51 describes the details of the A & C Exclusive Alarming Server Facet. This Facet requires support for *Alarms* with multiple sub-states that identify different limit *Conditions*. This facet builds upon the A&C Alarm Server Facet. The term exclusive means only one sub-state can be active at a time. For example, a temperature exceeds the HighHigh limit the associated exclusive LevelAlarm will be in the HighHigh sub-state and not in the High sub-state. This Facet requires that a Server support at least one of the optional Alarm models: Limit, RateOfChange or Deviation.

Table 51 – A & C Exclusive Alarming Server Facet

Group	Conformance Unit / Profile Title	Optional
<i>Profile</i>	A & C Alarm Server Facet	False
<i>Alarms and Conditions</i>	A & C Exclusive Deviation	True
<i>Alarms and Conditions</i>	A & C Exclusive Level	True
<i>Alarms and Conditions</i>	A & C Exclusive Limit	False
<i>Alarms and Conditions</i>	A & C Exclusive RateOfChange	True

6.6.34 A & C Non-Exclusive Alarming Server Facet

Table 52 describes the details of the A & C Non-Exclusive Alarming Server Facet. This Facet requires support for *Alarms* with multiple sub-states that identify different limit *Conditions*. This Facet builds upon the A&C Alarm Server Facet. The term non-exclusive means more than one sub-state can be active at a time. For example, if a temperature exceeds the HighHigh limit the associated non-exclusive LevelAlarm will be in both the High and the HighHigh sub-state. This Facet requires that a server support at least one of the optional alarm models: Limit, RateOfChange or Deviation.

Table 52 – A & C Non-Exclusive Alarming Server Facet

Group	Conformance Unit / Profile Title	Optional
<i>Profile</i>	A & C Alarm Server Facet	False
<i>Alarms and Conditions</i>	A & C Non-Exclusive Deviation	True
<i>Alarms and Conditions</i>	A & C Non-Exclusive Level	True
<i>Alarms and Conditions</i>	A & C Non-Exclusive Limit	False
<i>Alarms and Conditions</i>	A & C Non-Exclusive RateOfChange	True

6.6.35 A & C Previous Instances Server Facet

Table 53 describes the details of the A & C Previous Instances Server Facet. This Facet requires support for *Conditions* with previous states that still require action on the part of the operator. This Facet builds upon the A&C Base Condition Server Facet. A common use case for this Facet is a safety critical system that requires that all *Alarms* be acknowledged even if it the original problem goes away and the Alarm returns to the inactive state. In these cases, the previous state with active Alarm is still reported by the Server until the Operator

acknowledges it. When a *Condition* has previous states it will produce events with different Branch identifiers. When previous state no longer needs attention the branch will disappear.

Table 53 – A & C Previous Instances Server Facet

Group	Conformance Unit / Profile Title	Optional
Profile	A & C Base Condition Server Facet	False
Alarms and Conditions	A & C Branch	False

6.6.36 A & C Dialog Server Facet

Table 54 describes the details of the A & C Dialog Server Facet. This Facet requires support of Dialog *Conditions*. This Facet builds upon the A & C BaseCondition Server Facet Dialogs are ConditionTypes used to request user input. They are typically used when a *Server* has entered some state that requires intervention by a *Client*. For example, a *Server* monitoring a paper machine indicates that a roll of paper has been wound and is ready for inspection. The *Server* would activate a Dialog *Condition* indicating to the user that an inspection is required. Once the inspection has taken place the user responds by informing the *Server* of an accepted or unaccepted inspection allowing the process to continue.

Table 54 – A & C Dialog Server Facet

Group	Conformance Unit / Profile Title	Optional
Profile	A & C Base Condition Server Facet	False
Alarms and Conditions	A & C Dialog	False

6.6.37 A & C CertificateExpiration Server Facet

Table 55 describes the details of the A & C CertificateExpiration Server Facet. This Facet requires support of the CertificateExpirationAlarmType. It is used to inform *Clients* when the *Server's Certificate* is within the defined expiration period.

Table 55 – A & C CertificateExpiration Server Facet

Group	Conformance Unit / Profile Title	Optional
Profile	A & C Base Condition Server Facet	False
Alarms and Conditions	A & C Acknowledge	False
Alarms and Conditions	A & C Alarm	False
Alarms and Conditions	A & C CertificateExpiration	False
Alarms and Conditions	A & C Comment	True
Alarms and Conditions	A & C Confirm	True
Alarms and Conditions	A & C Shelving	True

6.6.38 A & E Wrapper Facet

Table 56 describes the details of the A & E Wrapper Facet. This Facet specifies the requirements for a UA *Server* that wraps an OPC *Alarm & Event* (AE) *Server* (COM). This *Profile* identifies the sub-set of the UA *Alarm & Condition* model which is provided by the COM OPC AE specification. It is intended to provide guidance to developers who are creating servers that front end existing applications. It is important to note that some OPC A&E COM *Servers* may not support all of the functionality provided by an OPC UA A&C server, in these cases similar functionality maybe available via some non-OPC interface. For example if an A&E COM server does not support sending *Alarm Acknowledgement* messages to the system

that it is obtaining alarm information from, this functionality may be available via some out of scope features in the underlying *Alarm* system. Another possibility is that the underlying system does not require acknowledgements or automatically acknowledges the alarm.

Table 56 – A & E Wrapper Facet

Group	Conformance Unit / Profile Title	Optional
Address Space Model	Address Space Events	False
Address Space Model	Address Space Notifier Hierarchy	False
Address Space Model	Address Space Source Hierarchy	False
<i>Alarms and Conditions</i>	A & C Acknowledge	False
<i>Alarms and Conditions</i>	A & C Alarm	False
<i>Alarms and Conditions</i>	A & C Basic	False
<i>Alarms and Conditions</i>	A & C ConditionClasses	False
<i>Alarms and Conditions</i>	A & C Refresh	False
<i>Alarms and Conditions</i>	A & E Wrapper Mapping	False
Monitored Item Services	Monitor Basic	False
Monitored Item Services	Monitor Complex Event Filter	False
Monitored Item Services	Monitor Events	False
Monitored Item Services	Monitor Items 2	False
Monitored Item Services	Monitor QueueSize_ServerMax	False
Subscription Services	Subscription Basic	False
Subscription Services	Subscription Minimum 1	False
Subscription Services	Subscription Publish Discard Policy	False
Subscription Services	Subscription Publish Min 02	False

6.6.39 Method Server Facet

Table 57 describes the details of the *Method Server* Facet. This Facet specifies the support of *Method* invocation via the Call service. Methods are "lightweight" functions which are similar to the methods of a class found in any object-oriented programming language. A *Method* can have its scope bounded by an owning *Object* or an owning *ObjectType*. Methods with an *ObjectType* as their scope are similar to static methods in a class.

Table 57 – Method Server Facet

Group	Conformance Unit / Profile Title	Optional
Address Space Model	Address Space Method	False
Method Services	Method Call	False

6.6.40 Auditing Server Facet

Table 58 describes the details of the *Auditing Server* Facet. This Facet requires the support of Auditing which includes the Standard *Event Subscription Server* Facet. Support of this Facet requires that Audit Events be produced when a client performs some action to change the state of the server, such as changing the *AddressSpace*, inserting or updating a value etc. The auditEntryId passed by the *Client* is a field contained in every Audit Event and allows actions to be traced across multiple systems. The Audit Event Types and their fields ~~must~~ shall be exposed in the *Server's AddressSpace*

Table 58 – Auditing Server Facet

Group	Conformance Unit / Profile Title	Optional
Profile	Standard Event Subscription Server Facet	False
Auditing	Auditing Base	False

6.6.41 Node Management Server Facet

Table 59 describes the details of the *Node Management Server Facet*. This Facet requires the support of the *Services* that allow the *Client* to add, modify and delete *Nodes* in the *AddressSpace*. These *Services* provide an interface which can be used to configure *Servers*. This means all changes to the *AddressSpace* are expected to persist even after the *Client* has disconnected from the *Server*.

Table 59 – Node Management Server Facet

Group	Conformance Unit / Profile Title	Optional
Address Space Model	Address Space Base	False
Base Information	Base Info Model Change	False
Base Information	Base Info Type System	False
Node Management Services	Node Management Add Node	False
Node Management Services	Node Management Add Ref	False
Node Management Services	Node Management Delete Node	False
Node Management Services	Node Management Delete Ref	False

6.6.42 User Role Base Server Facet

Table 60 describes the details of the *User Role Base Server Facet*. This Facet defines support of the OPC UA Information Model to expose configured user roles and permissions.

Table 60 – User Role Base Server Facet

Group	Conformance Unit / Profile Title	Optional
Security	Security Role Server Base	False

6.6.43 User Role Management Server Facet

Table 61 describes the details of the *User Role Management Server Facet*. This Facet defines support of the OPC UA approach to manage user roles and permissions and to grant access to *Nodes* and *Services* based on the assigned roles and permissions.

Table 61 – User Role Management Server Facet

Group	Conformance Unit / Profile Title	Optional
Profile	User Role Base Server Facet	False
Security	Security Role Server DefaultRolePermissions	False
Security	Security Role Server IdentityManagement	False
Security	Security Role Server Management	False
Security	Security Role Server Restrict Applications	True
Security	Security Role Server Restrict Endpoints	True
Security	Security Role Server RolePermissions	True
Security	Security Role Well Known	False

6.6.44 State Machine Server Facet

Table 62 describes the details of the State Machine Server Facet. This Facet defines support of StateMachines based on the types in IEC 62541-5.

Table 62 – State Machine Server Facet

Group	Conformance Unit / Profile Title	Optional
Base Information	Base Info Available States and Transitions	True
Base Information	Base Info Finite State Machine Instance	True
Base Information	Base Info State Machine Instance	False

6.6.45 Client Redundancy Server Facet

Table 63 describes the details of the Client Redundancy Server Facet. This Facet defines the Server actions that are required for support of redundant Clients. Support of this Facet requires the implementation of the TransferSubscriptions Service which allows the transfer of Subscriptions from one Client's Session to another Client's Session.

Table 63 – Client Redundancy Server Facet

Group	Conformance Unit / Profile Title	Optional
Subscription Services	Subscription Transfer	False

6.6.46 Redundancy Transparent Server Facet

Table 64 describes the details of the Redundancy Transparent Server Facet. This Facet requires support for transparent redundancy. If Servers implement transparent redundancy then the failover from one Server to another is transparent to the Client such that the Client is unaware that a failover has occurred; the Client does not need to do anything at all to keep data flowing. This type of redundancy is usually a hardware solution.

Table 64 – Redundancy Transparent Server Facet

Group	Conformance Unit / Profile Title	Optional
Redundancy	Redundancy Server Transparent	False

6.6.47 Redundancy Visible Server Facet

Table 65 describes the details of the Redundancy Visible Server Facet. This Facet specifies the support for non-transparent redundancy. Failover for this type of redundancy requires the *Client* to monitor *Server* status and to switch to a backup *Server* if it detects a failure. The *Server* shall expose the methods of failover it supports (cold, warm or hot). The failover method tells the *Client* what it ~~must~~ shall do when connecting to a *Server* and when a failure occurs. Cold redundancy requires a *Client* to reconnect to a backup *Server* after the initial *Server* has failed. Warm redundancy allows a *Client* to connect to multiple *Servers*, but only one *Server* will be providing values. In hot redundancy multiple *Servers* are able to provide data and a *Client* can connect to multiple *Servers* for the data.

Table 65 – Redundancy Visible Server Facet

Group	Conformance Unit / Profile Title	Optional
Redundancy	Redundancy Server	False

6.6.48 Historical Raw Data Server Facet

Table 66 describes the details of the Historical Raw Data Server Facet. This Facet defines the basic functionality when supporting historical data access for raw data.

Table 66 – Historical Raw Data Server Facet

Group	Conformance Unit / Profile Title	Optional
Attribute Services	Attribute Historical Read	False
Historical Access	Historical Access Data Max Nodes Read Continuation Point	False
Historical Access	Historical Access Read Raw	False
Historical Access	Historical Access ServerTimestamp	True

6.6.49 Historical Aggregate Server Facet

Table 67 describes the details of the Historical Aggregate Server Facet. This Facet indicates that the server supports aggregate processing to produce derived values from raw historical data.

Table 67 – Historical Aggregate Server Facet

Group	Conformance Unit / Profile Title	Optional
Aggregates	Aggregate – AnnotationCount	True
Aggregates	Aggregate – Average	True
Aggregates	Aggregate – Count	True
Aggregates	Aggregate – Custom	True
Aggregates	Aggregate – Delta	True
Aggregates	Aggregate – DeltaBounds	True
Aggregates	Aggregate – DurationBad	True
Aggregates	Aggregate – DurationGood	True
Aggregates	Aggregate – DurationInStateNonZero	True
Aggregates	Aggregate – DurationInStateZero	True
Aggregates	Aggregate – End	True
Aggregates	Aggregate – EndBound	True
Aggregates	Aggregate – Interpolative	True
Aggregates	Aggregate – Maximum	True
Aggregates	Aggregate – Maximum2	True
Aggregates	Aggregate – MaximumActualTime	True
Aggregates	Aggregate – MaximumActualTime2	True
Aggregates	Aggregate – Minimum	True
Aggregates	Aggregate – Minimum2	True
Aggregates	Aggregate – MinimumActualTime	True
Aggregates	Aggregate – MinimumActualTime2	True
Aggregates	Aggregate – NumberOfTransitions	True
Aggregates	Aggregate – PercentBad	True
Aggregates	Aggregate – PercentGood	True
Aggregates	Aggregate – Range	True
Aggregates	Aggregate – Range2	True
Aggregates	Aggregate – StandardDeviationPopulation	True
Aggregates	Aggregate – StandardDeviationSample	True
Aggregates	Aggregate – Start	True
Aggregates	Aggregate – StartBound	True
Aggregates	Aggregate – TimeAverage	True
Aggregates	Aggregate – TimeAverage2	True
Aggregates	Aggregate – Total	True
Aggregates	Aggregate – Total2	True
Aggregates	Aggregate – VariancePopulation	True
Aggregates	Aggregate – VarianceSample	True
Aggregates	Aggregate – WorstQuality	True
Aggregates	Aggregate – WorstQuality2	True
Aggregates	Aggregate – ~~master~~ Historical Configuration	~~False~~ True
Aggregates	Aggregate – ~~optional~~ Master Configuration	~~True~~ False
Attribute Services	Attribute Historical Read	False
Historical Access	Historical Access Aggregates	False
Historical Access	Historical Access Data Max Nodes Read Continuation Point	False

6.6.50 Historical Data AtTime Server Facet

Table 68 describes the details of the Historical Data AtTime Server Facet. This Facet indicates that the historical Server supports reading data by specifying specific timestamps.

Table 68 – Historical Data AtTime Server Facet

Group	Conformance Unit / <i>Profile</i> Title	Optional
<i>Attribute Services</i>	<i>Attribute</i> Historical Read	False
Historical Access	Historical Access Data Max <i>Nodes</i> Read Continuation Point	False
Historical Access	Historical Access Time Instance	False

6.6.51 Historical Access Modified Data Server Facet

Table 69 describes the details of the Historical Access Modified Data *Server* Facet. This Facet defines support of reading modified historical values (values that were modified or inserted).

Table 69 – Historical Access Modified Data Server Facet

Group	Conformance Unit / <i>Profile</i> Title	Optional
<i>Attribute Services</i>	<i>Attribute</i> Historical Read	False
Historical Access	Historical Access Modified Values	False

6.6.52 Historical Annotation Server Facet

Table 70 describes the details of the Historical Annotation *Server* Facet. This Facet defines support for the storage and retrieval of annotations for historical data.

Table 70 – Historical Annotation Server Facet

Group	Conformance Unit / <i>Profile</i> Title	Optional
<i>Attribute Services</i>	<i>Attribute</i> Historical Read	False
<i>Attribute Services</i>	<i>Attribute</i> Historical Update	False
Historical Access	Historical Access Annotations	False

6.6.53 Historical Data Insert Server Facet

Table 71 describes the details of the Historical Data Insert *Server* Facet. This Facet includes Historical Data Insert functionality.

Table 71 – Historical Data Insert Server Facet

Group	Conformance Unit / <i>Profile</i> Title	Optional
<i>Attribute Services</i>	<i>Attribute</i> Historical Update	False
Historical Access	Historical Access Insert Value	False
Historical Access	Historical Access ServerTimestamp	True

6.6.54 Historical Data Update Server Facet

Table 72 describes the details of the Historical Data Update *Server* Facet. This Facet includes Historical Data Update functionality.

Table 72 – Historical Data Update Server Facet

Group	Conformance Unit / Profile Title	Optional
<i>Attribute Services</i>	<i>Attribute</i> Historical Update	False
Historical Access	Historical Access ServerTimestamp	True
Historical Access	Historical Access Update Value	False

6.6.55 Historical Data Replace Server Facet

Table 73 describes the details of the Historical Data Replace Server Facet. This Facet includes Historical Data Replace functionality.

Table 73 – Historical Data Replace Server Facet

Group	Conformance Unit / Profile Title	Optional
<i>Attribute Services</i>	<i>Attribute</i> Historical Update	False
Historical Access	Historical Access Replace Value	False
Historical Access	Historical Access ServerTimestamp	True

6.6.56 Historical Data Delete Server Facet

Table 74 describes the details of the Historical Data Delete Server Facet. This Facet includes Historical Data Delete functionality.

Table 74 – Historical Data Delete Server Facet

Group	Conformance Unit / Profile Title	Optional
<i>Attribute Services</i>	<i>Attribute</i> Historical Update	False
Historical Access	Historical Access Delete Value	False

6.6.57 Historical Access Structured Data Server Facet

Table 75 describes the details of the Historical Access Structured Data Server Facet. This Facet indicates that the Server supports storage and retrieval of structured values for all supported access types. If a listed access type is supported then the corresponding optional *ConformanceUnit* shall be supported.

Table 75 – Historical Access Structured Data Server Facet

Group	Conformance Unit / Profile Title	Optional
Historical Access	Historical Access Structured Data Delete	True
Historical Access	Historical Access Structured Data Insert	True
Historical Access	Historical Access Structured Data Read Modified	True
Historical Access	Historical Access Structured Data Read Raw	False
Historical Access	Historical Access Structured Data Replace	True
Historical Access	Historical Access Structured Data Time Instance	True
Historical Access	Historical Access Structured Data Update	True

6.6.58 Base Historical Event Server Facet

Table 76 describes the details of the Base Historical *Event Server* Facet. This Facet defines the server requirements to support basic Historical *Event* functionality, including simple filtering and general access.

Table 76 – Base Historical Event Server Facet

Group	Conformance Unit / Profile Title	Optional
<i>Attribute Services</i>	<i>Attribute</i> Historical Read	False
Historical Access	Historical Access <i>Event</i> Max Events Read Continuation Point	False
Historical Access	Historical Access Events	False

6.6.59 Historical Event Update Server Facet

Table 77 describes the details of the Historical *Event* Update Server Facet. This Facet includes Historical *Event* update access functionality.

Table 77 – Historical Event Update Server Facet

Group	Conformance Unit / Profile Title	Optional
<i>Attribute Services</i>	<i>Attribute</i> Historical Update	False
Historical Access	Historical Access Update <i>Event</i>	False

6.6.60 Historical Event Replace Server Facet

Table 78 describes the details of the Historical *Event* Replace Server Facet. This Facet includes Historical *Event* replace access functionality.

Table 78 – Historical Event Replace Server Facet

Group	Conformance Unit / Profile Title	Optional
<i>Attribute Services</i>	<i>Attribute</i> Historical Update	False
Historical Access	Historical Access Replace <i>Event</i>	False

6.6.61 Historical Event Insert Server Facet

Table 79 describes the details of the Historical *Event* Insert Server Facet. This Facet includes Historical *Event* insert access functionality.

Table 79 – Historical Event Insert Server Facet

Group	Conformance Unit / Profile Title	Optional
<i>Attribute Services</i>	<i>Attribute</i> Historical Update	False
Historical Access	Historical Access Insert <i>Event</i>	False

6.6.62 Historical Event Delete Server Facet

Table 80 describes the details of the Historical *Event* Delete Server Facet. This Facet includes Historical *Event* delete access functionality

Table 80 – Historical Event Delete Server Facet

Group	Conformance Unit / Profile Title	Optional
<i>Attribute Services</i>	<i>Attribute</i> Historical Update	False
Historical Access	Historical Access Delete <i>Event</i>	False

6.6.63 Aggregate Subscription Server Facet

Table 81 describes the details of the Aggregate *Subscription Server* Facet. This Facet defines the handling of the aggregate filter when subscribing for *Attribute* values.

Table 81 – Aggregate Subscription Server Facet

Group	Conformance Unit / Profile Title	Optional
<i>Profile</i>	Standard DataChange <i>Subscription Server</i> Facet	False
Aggregates	Aggregate <i>Subscription</i> – AnnotationCount	True
Aggregates	Aggregate <i>Subscription</i> – Average	True
Aggregates	Aggregate <i>Subscription</i> – Count	True
Aggregates	Aggregate <i>Subscription</i> – Custom	True
Aggregates	Aggregate <i>Subscription</i> – Delta	True
Aggregates	Aggregate <i>Subscription</i> – DeltaBounds	True
Aggregates	Aggregate <i>Subscription</i> – DurationBad	True
Aggregates	Aggregate <i>Subscription</i> – DurationGood	True
Aggregates	Aggregate <i>Subscription</i> – DurationInStateNonZero	True
Aggregates	Aggregate <i>Subscription</i> – DurationInStateZero	True
Aggregates	Aggregate <i>Subscription</i> – End	True
Aggregates	Aggregate <i>Subscription</i> – EndBound	True
Aggregates	Aggregate <i>Subscription</i> – Filter	False
Aggregates	Aggregate <i>Subscription</i> – Interpolative	True
Aggregates	Aggregate <i>Subscription</i> – Maximum	True
Aggregates	Aggregate <i>Subscription</i> – Maximum2	True
Aggregates	Aggregate <i>Subscription</i> – MaximumActualTime	True
Aggregates	Aggregate <i>Subscription</i> – MaximumActualTime2	True
Aggregates	Aggregate <i>Subscription</i> – Minimum	True
Aggregates	Aggregate <i>Subscription</i> – Minimum2	True
Aggregates	Aggregate <i>Subscription</i> – MinimumActualTime	True
Aggregates	Aggregate <i>Subscription</i> – MinimumActualTime2	True
Aggregates	Aggregate <i>Subscription</i> – NumberOfTransitions	True
Aggregates	Aggregate <i>Subscription</i> – PercentBad	True
Aggregates	Aggregate <i>Subscription</i> – PercentGood	True
Aggregates	Aggregate <i>Subscription</i> – Range	True
Aggregates	Aggregate <i>Subscription</i> – Range2	True
Aggregates	Aggregate <i>Subscription</i> – StandardDeviationPopulation	True
Aggregates	Aggregate <i>Subscription</i> – StandardDeviationSample	True
Aggregates	Aggregate <i>Subscription</i> – Start	True
Aggregates	Aggregate <i>Subscription</i> – StartBound	True
Aggregates	Aggregate <i>Subscription</i> – TimeAverage	True
Aggregates	Aggregate <i>Subscription</i> – TimeAverage2	True
Aggregates	Aggregate <i>Subscription</i> – Total	True
Aggregates	Aggregate <i>Subscription</i> – Total2	True
Aggregates	Aggregate <i>Subscription</i> – VariancePopulation	True
Aggregates	Aggregate <i>Subscription</i> – VarianceSample	True
Aggregates	Aggregate <i>Subscription</i> – WorstQuality	True
Aggregates	Aggregate <i>Subscription</i> – WorstQuality2	True
Monitored Item <i>Services</i>	Monitor Aggregate Filter	False

6.6.64 Nano Embedded Device Server Profile

Nano Embedded Device Server Profile has been deprecated as it is superseded with Nano Embedded Device 2017 Server Profile in v1.04.

6.6.65 Nano Embedded Device 2017 Server Profile

Table 82 describes the details of the Nano Embedded Device 2017 *Server Profile*. This *Profile* is a FullFeatured *Profile* intended for chip level devices with limited resources. This *Profile* is functionally equivalent to the Core *Server Facet* and defines the OPC UA TCP binary protocol as the required transport profile. The support of Diagnostic *Objects* and *Variables* is optional for this *Profile* despite it being defined as "mandatory" in IEC 62541-5. Support of Diagnostic *Objects* and *Variables* is mandatory in some higher-level *Profiles*. Exposing types in the *AddressSpace* is optional for this *Profile* except if custom types (i.e. types that are derived from well-known *ObjectTypes*, *VariableTypes*, *ReferenceType* or *DataTypes*) are used. Exposing all supported types in the *AddressSpace* is mandatory in some higher-level *Profiles*. This profile supersedes the "Nano Embedded Device *Server Profile*".

Table 82 – Nano Embedded Device 2017 Server Profile

Group	Conformance Unit / Profile Title	Optional
Profile	Core 2017 <i>Server Facet</i>	False
Profile	UA-TCP UA-SC UA-Binary	False
Base Information	Base Info Custom Type System	True
Base Information	Base Info Diagnostics	True

6.6.66 Micro Embedded Device Server Profile

Micro Embedded Device Server Profile has been deprecated as it is superseded with Micro Embedded Device 2017 Server Profile in v1.04.

6.6.67 Micro Embedded Device 2017 Server Profile

Table 83 describes the details of the Micro Embedded Device 2017 *Server Profile*. This *Profile* is a FullFeatured *Profile* intended for small devices with limited resources. This *Profile* builds upon the Nano Embedded Device *Server Profile*. The most important additions are: support for subscriptions via the Embedded Data Change *Subscription Server Facet* and support for at least two sessions. A complete Type System is not required; however, if the *Server* implements any non-UA types, then these types and their super-types ~~must~~ shall be exposed. This profile supersedes the "Micro Embedded Device *Server Profile*".

Table 83 – Micro Embedded Device 2017 Server Profile

Group	Conformance Unit / Profile Title	Optional
Profile	Embedded DataChange <i>Subscription Server Facet</i>	False
Profile	Nano Embedded Device 2017 <i>Server Profile</i>	False
~~Base Information~~	~~Base Info Custom Type System~~	~~False~~
Session Services	Session Minimum 2 Parallel	False

6.6.68 Embedded UA Server Profile

Embedded UA Server Profile has been deprecated as it is superseded with Embedded 2017 UA Server Profile in v1.04.

6.6.69 Embedded 2017 UA Server Profile

Table 84 describes the details of the Embedded 2017 UA Server Profile. This Profile is a FullFeatured Profile that is intended for devices with more than 50 MB of memory and a more powerful processor. This Profile builds upon the Micro Embedded Device Server Profile. The most important additions are: support for security via the SecurityPolicy — Basic128Rsa15 Facet, Policies and support for the Standard DataChange Subscription Server Facet. This Profile also requires that Servers expose all OPC-UA types that are used by the Server including their components and their super-types. This profile supersedes the "Embedded Device Server Profile".

Table 84 – Embedded 2017 UA Server Profile

Group	Conformance Unit / Profile Title	Optional
Profile	Micro Embedded Device 2017 Server Profile	False
Profile	SecurityPolicy — Basic128Rsa15	False
Profile	Standard DataChange Subscription 2017 Server Facet	False
Profile	User Token — X509 Certificate Server Facet	False
Base Information	Base Info Engineering Units	True
Base Information	Base Info Placeholder Modelling Rules	True
Base Information	Base Info Type System	False
Security	Security – No Application Authentication	True
Security	Security Default ApplicationInstance Certificate	False
Security	Security Policy Required	False

6.6.70 Standard UA Server Profile

Standard UA Server Profile has been deprecated as it is superseded with Standard 2017 UA Server Profile in v1.04.

6.6.71 Standard 2017 UA Server Profile

Table 85 describes the details of the Standard 2017 UA Server Profile. This Profile is a FullFeatured Profile that defines a minimum set of functionality required for PC based OPC UA servers. Such a server must provide the base AddressSpace structure with type nodes, instance nodes and diagnostic information. The Server must provide connection establishment through the OPC UA TCP binary protocol with security and the creation of at least 50 parallel sessions. It includes view services like browsing and the attribute services for reading and writing of current values. In addition, the monitoring of data changes is included with a minimum of 5 subscriptions for half of the required sessions (total 225) and a minimum of 500 monitored items for half of the subscriptions (total 56250). Compared to the embedded profiles, the Profile requires higher limits for Sessions, Subscriptions and Monitored Items. It also requires support of diagnostic information. This profile supersedes the "Standard UA Server Profile".

Table 85 – Standard 2017 UA Server Profile

Group	Conformance Unit / Profile Title	Optional
<i>Profile</i>	Embedded 2017 UA Server Profile	False
<i>Profile</i>	Enhanced DataChange Subscription 2017 Server Facet	False
<i>Profile</i>	User Token – X509 Certificate Server Facet	False
<i>Attribute Services</i>	Attribute Write StatusCode & Timestamp	True
Base Information	Base Info Diagnostics	False
<i>Discovery Services</i>	Discovery Register	False
<i>Discovery Services</i>	Discovery Register2	False
<i>Session Services</i>	Session Cancel	False
<i>Session Services</i>	Session Change User	True
<i>Session Services</i>	Session Minimum 50 Parallel	False
<i>View Services</i>	View Minimum Continuation Point 05	False

6.6.72 Core Client Facet

Core Client Facet has been deprecated as it is superseded with Core 2017 Client Facet in v1.04.

6.6.73 Core 2017 Client Facet

Table 86 describes the details of the Core 2017 Client Facet. This Facet defines the core functionality required for any *Client*. This Facet includes the core functions for Security and Session handling.

This Facet supersedes the Core Client Facet.

Table 86 – Core 2017 Client Facet

Group	Conformance Unit / Profile Title	Optional
<i>Profile</i>	SecurityPolicy – Basic128Rsa15	False
<i>Profile</i>	SecurityPolicy – None	False
<i>Profile</i>	User Token – User Name Password Client Facet	False
<i>Profile</i>	User Token – X509 Certificate Client Facet	False
Base Information	Base Info Client Estimated Return Time	True
Base Information	Base Info Client Selection List	True
Security	Security Administration	False
Security	Security Policy Required	False
<i>Session Services</i>	Session Client Auto Reconnect	False
<i>Session Services</i>	Session Client Base	False
<i>Session Services</i>	Session Client Cancel	True
<i>Session Services</i>	Session Client Detect Shutdown	False
<i>Session Services</i>	Session Client General Service Behaviour	False
<i>Session Services</i>	Session Client Impersonate	True
<i>Session Services</i>	Session Client KeepAlive	False
<i>Session Services</i>	Session Client Renew NodeIds	True

6.6.74 Sessionless Client Facet

Table 87 describes the details of the Sessionless *Client* Facet. Defines the use of Sessionless *Service* invocation in a *Client*.

Table 87 – Sessionless Client Facet

Group	Conformance Unit / Profile Title	Optional
Discovery Services	Discovery Client Get Endpoints SessionLess	True
Session Services	Session Client SessionLess Service Calls	False

6.6.75 Reverse Connect Client Facet

Table 88 describes the details of the Reverse Connect *Client* Facet. This Facet defines support of reverse connectivity in a *Client*. Usually, a connection is opened by the *Client* before starting the UA-specific handshake. This will fail, however, when *Servers* are behind firewalls. In the reverse connectivity scenario, the *Client* accepts a connection request and a ReverseHello message from a *Server* and establishes a Secure Channel using this connection.

Table 88 – Reverse Connect Client Facet

Group	Conformance Unit / Profile Title	Optional
Protocol and Encoding	Protocol Reverse Connect Client	False

6.6.76 Base Client Behaviour Facet

Table 89 describes the details of the Base *Client* Behaviour Facet. This Facet indicates that the *Client* supports behaviour that *Clients* shall follow for best use by operators and administrators. They include allowing configuration of an endpoint for a server without using the discovery service set; Support for manual security setting configuration and behaviour with regard to security issues; support for Automatic reconnection to a disconnected server. These behaviours can only be tested in a test lab. They are best practice guidelines.

Table 89 – Base Client Behaviour Facet

Group	Conformance Unit / Profile Title	Optional
Base Information	Base Info Client Remote Nodes	True
Discovery Services	Discovery Client Configure Endpoint	False
Security	Security Administration	False
Security	Security Administration – XML Schema	False
Security	Security Certificate Administration	False
Session Services	Session Client Auto Reconnect	True
Subscription Services	Subscription Client Multiple	False
Subscription Services	Subscription Client Publish Configurable	False

6.6.77 Discovery Client Facet

Table 90 describes the details of the *Discovery Client* Facet. This Facet defines the ability to discover *Servers* and their Endpoints.

Table 90 – Discovery Client Facet

Group	Conformance Unit / Profile Title	Optional
<i>Discovery Services</i>	<i>Discovery Client</i> Configure Endpoint	False
<i>Discovery Services</i>	<i>Discovery Client</i> Find Servers Basic	False
<i>Discovery Services</i>	<i>Discovery Client</i> Find Servers Dynamic	False
<i>Discovery Services</i>	<i>Discovery Client</i> Find Servers with URI	True
<i>Discovery Services</i>	<i>Discovery Client</i> Get Endpoints Basic	False
<i>Discovery Services</i>	<i>Discovery Client</i> Get Endpoints Dynamic	False

6.6.78 Subnet Discovery Client Facet

Table 91 describes the details of the Subnet *Discovery Client* Facet. Support of this Facet enables discovery of the *Server* on a subnet.

Table 91 – Subnet Discovery Client Facet

Group	Conformance Unit / Profile Title	Optional
<i>Discovery Services</i>	<i>Discovery Client</i> Find Servers on Network	False
<i>Discovery Services</i>	<i>Discovery Client</i> Find Servers on Network using LDS-ME	True
<i>Discovery Services</i>	<i>Discovery Client</i> Find Servers on Network using mDNS	True

6.6.79 Global Discovery Client Facet

Table 92 describes the details of the Global *Discovery Client* Facet. Support of this Facet enables system-wide discovery of *Servers* using a Global *Discovery Server* (GDS).

Table 92 – Global Discovery Client Facet

Group	Conformance Unit / Profile Title	Optional
<i>Discovery Services</i>	<i>Discovery Client</i> Find Applications in GDS	True
<i>Discovery Services</i>	<i>Discovery Client</i> Find Servers in GDS	False

6.6.80 Global Certificate Management Client Facet

Table 93 describes the details of the Global *Certificate Management Client* Facet. This Facet defines the capability to interact with a Global *Certificate Management Server* to obtain an initial or renewed *Certificate* and Trust Lists.

Table 93 – Global Certificate Management Client Facet

Group	Conformance Unit / Profile Title	Optional
Security	Pull Model for Global Certificate and TrustList Management	False

6.6.81 KeyCredential Service Client Facet

Table 94 describes the details of the KeyCredential *Service Client* Facet. This Facet defines the capability to interact with a KeyCredential *Service* to obtain KeyCredentials. For example KeyCredentials are needed to access an Authorization *Service* or a Broker. The KeyCredential *Service* is typically part of a system-wide tool, like a GDS that also manages Applications, Access Tokens, and *Certificates*.

Table 94 – KeyCredential Service Client Facet

Group	Conformance Unit / Profile Title	Optional
Security	Pull Model for KeyCredential Service	False

6.6.82 Access Token Request Client Facet

Table 95 describes the details of the Access Token Request *Client* Facet. A *Client* Facet for using the RequestAccessToken *Method* on an Authorization Server (defined in IEC 62541-12) to request such a token.

Table 95 – Access Token Request Client Facet

Group	Conformance Unit / Profile Title	Optional
Security	Authorization Service Client	False

6.6.83 AddressSpace Lookup Client Facet

Table 96 describes the details of the AddressSpace Lookup *Client* Facet. This Facet defines the ability to navigate through the AddressSpace and includes basic AddressSpace concepts, view and browse functionality and simple attribute read functionality.

Table 96 – AddressSpace Lookup Client Facet

Group	Conformance Unit / Profile Title	Optional
Address Space Model	Address Space <i>Client</i> Base	False
Attribute Services	Attribute <i>Client</i> Read Base	False
Attribute Services	Attribute <i>Client</i> Remote Nodes Attribute Access	True
Base Information	Base Info <i>Client</i> Basic	False
Base Information	Base Info <i>Client</i> Change Events	True
Base Information	Base Info <i>Client</i> GetMonitoredItems <i>Method</i>	True
Base Information	Base Info <i>Client</i> Progress Events	True
Base Information	Base Info <i>Client</i> System Status	True
View Services	View <i>Client</i> Basic Browse	False
View Services	View <i>Client</i> Basic ResultSet Filtering	False
View Services	View <i>Client</i> RegisterNodes	True
View Services	View <i>Client</i> Remote Nodes Browse	True
View Services	View <i>Client</i> Remote Nodes Translate Browse	True
View Services	View <i>Client</i> TranslateBrowsePath	True

6.6.84 Request State Change Client Facet

Table 97 describes the details of the Request State Change *Client* Facet. This Facet specifies the ability to invoke the RequestServerStateChange *Method*.

Table 97 – Request State Change Client Facet

Group	Conformance Unit / Profile Title	Optional
Base Information	Base Info <i>Client</i> RequestServerStateChange	False

6.6.85 File Access Client Facet

Table 98 describes the details of the File Access *Client* Facet. This Facet defines the ability to use File transfer via the defined FileType. This includes reading and optionally writing.

Table 98 – File Access Client Facet

Group	Conformance Unit / Profile Title	Optional
Base Information	Base Info <i>Client</i> FileType Base	False
Base Information	Base Info <i>Client</i> FileType Write	True

6.6.86 Entry-Level Support 2015 Client Facet

~~Table 75 describes the details of the Entry-Level Support *Client* Facet. This Facet defines the ability to interoperate with low-end Servers, e.g. Servers that support the Nano Embedded Profile (either by automatically adapting to the Server capabilities or through configuration). It implies respecting Server provided limits for Session, continuation points, Subscription, user authorization and locales.~~

Table 99 describes the details of the Entry Level Support 2015 *Client* Facet. This Facet defines the ability to interoperate with low-end Servers, in particular Servers that support the Nano Embedded Profile but in general Servers with defined limits.

Table 99 – Entry Level Support 2015 Client Facet

Group	Conformance Unit / Profile Title	Optional
~~Session Services~~	~~Client Entry-Level Support~~	~~False~~
Base Information	Base Info <i>Client</i> Honour Operation Limits	False
Base Information	Base Info <i>Client</i> Type Pre-Knowledge	False
<i>Session Services</i>	<i>Session-Client</i> Single <i>Session</i>	False
<i>Subscription Services</i>	<i>Subscription-Client</i> Fallback	False

6.6.87 Multi-Server Client Connection Facet

Table 100 describes the details of the Multi-Server *Client* Connection Facet. This Facet defines the ability for simultaneous access to multiple Servers.

Table 100 – Multi-Server Client Connection Facet

Group	Conformance Unit / Profile Title	Optional
<i>Session Services</i>	<i>Session-Client</i> Multiple Connections	False

6.6.88 Documentation – Client

Table 101 describes the details of the Documentation – *Client*. This Facet provides a list of user documentation that a *Client* application should provide.

Table 101 – Documentation – Client

Group	Conformance Unit / Profile Title	Optional
Miscellaneous	Documentation <i>Client</i> – Installation	False
Miscellaneous	Documentation <i>Client</i> – Multiple Languages	True
Miscellaneous	Documentation <i>Client</i> – On-line	True
Miscellaneous	Documentation <i>Client</i> – Supported Profiles	True
Miscellaneous	Documentation <i>Client</i> – Trouble Shooting Guide	True
Miscellaneous	Documentation <i>Client</i> – Users Guide	False

6.6.89 Attribute Read Client Facet

Table 102 describes the details of the *Attribute Read Client* Facet. This Facet defines the ability to read *Attribute* values of *Nodes*.

Table 102 – Attribute Read Client Facet

Group	Conformance Unit / Profile Title	Optional
Address Space Model	Address Space <i>Client</i> Atomicity	True
Address Space Model	Address Space <i>Client</i> Complex Data Dictionary	True
Address Space Model	Address Space <i>Client</i> DataTypeDefinition <i>Attribute</i>	True
Address Space Model	Address Space <i>Client</i> Full Array Only	True
<i>Attribute Services</i>	<i>Attribute Client</i> Read Base	False
<i>Attribute Services</i>	<i>Attribute Client</i> Read Complex	True
<i>Attribute Services</i>	<i>Attribute Client</i> Read with proper Encoding	True

6.6.90 Attribute Write Client Facet

Table 103 describes the details of the *Attribute Write Client* Facet. This Facet defines the ability to write *Attribute* values of *Nodes*.

Table 103 – Attribute Write Client Facet

Group	Conformance Unit / Profile Title	Optional
Address Space Model	Address Space <i>Client</i> Atomicity	True
Address Space Model	Address Space <i>Client</i> Complex Data Dictionary	True
Address Space Model	Address Space <i>Client</i> DataTypeDefinition <i>Attribute</i>	True
Address Space Model	Address Space <i>Client</i> Full Array Only	True
<i>Attribute Services</i>	<i>Attribute Client</i> Write Base	False
<i>Attribute Services</i>	<i>Attribute Client</i> Write Complex	True
<i>Attribute Services</i>	<i>Attribute Client</i> Write Quality & Timestamp	True

6.6.91 DataChange Subscriber Client Facet

Table 104 describes the details of the DataChange Subscriber *Client* Facet. This Facet defines the ability to monitor *Attribute* values for data change.

Table 104 – DataChange Subscriber Client Facet

Group	Conformance Unit / <i>Profile</i> Title	Optional
Address Space Model	Address Space <i>Client</i> Atomicity	True
Address Space Model	Address Space <i>Client</i> Complex Data Dictionary	True
Address Space Model	Address Space <i>Client</i> DataTypeDefinition <i>Attribute</i>	True
Address Space Model	Address Space <i>Client</i> Full Array Only	True
Base Information	Base Data <i>Client</i> ResendData <i>Method</i>	True
Base Information	Base Info <i>Client</i> GetMonitoredItems <i>Method</i>	True
Monitored Item <i>Services</i>	Monitor <i>Client</i> by Index	False
Monitored Item <i>Services</i>	Monitor <i>Client</i> Complex Value	True
Monitored Item <i>Services</i>	Monitor <i>Client</i> Deadband Filter	True
Monitored Item <i>Services</i>	Monitor <i>Client</i> Modify	True
Monitored Item <i>Services</i>	Monitor <i>Client</i> Trigger	True
Monitored Item <i>Services</i>	Monitor <i>Client</i> Value Change	False
<i>Subscription Services</i>	<i>Subscription Client</i> Basic	False
<i>Subscription Services</i>	<i>Subscription Client</i> Modify	True
<i>Subscription Services</i>	<i>Subscription Client</i> Multiple	True
<i>Subscription Services</i>	<i>Subscription Client</i> Republish	False

6.6.92 Durable Subscription Client Facet

Table 105 describes the details of the Durable *Subscription Client* Facet. This Facet specifies use of durable Subscriptions. It implies support of any of the DataChange or *Event* Subscriber Facets.

Table 105 – Durable Subscription Client Facet

Group	Conformance Unit / <i>Profile</i> Title	Optional
<i>Subscription Services</i>	<i>Subscription Client</i> Durable	False

6.6.93 DataAccess Client Facet

Table 106 describes the details of the DataAccess *Client* Facet. This Facet defines the ability to utilize the DataAccess Information Model, i.e., industrial automation data like analogue and discrete data items and their quality of service.

Table 106 – DataAccess Client Facet

Group	Conformance Unit / Profile Title	Optional
Address Space Model	Address Space <i>Client</i> Base	False
Address Space Model	Address Space <i>Client</i> Complex-DataTypes Data Dictionary	True
Attribute Services	Attribute <i>Client</i> Read Base	False
Attribute Services	Attribute <i>Client</i> Read Complex	True
Attribute Services	Attribute <i>Client</i> Read with proper Encoding	True
Data Access	Data Access <i>Client</i> AnalogItems	True
Data Access	Data Access <i>Client</i> Basic	False
Data Access	Data Access <i>Client</i> Deadband	True
Data Access	Data Access <i>Client</i> MultiState	True
Data Access	Data Access <i>Client</i> MultiStateValueDiscrete	True
Data Access	Data Access <i>Client</i> SemanticChange	True
Data Access	Data Access <i>Client</i> TwoState	True

6.6.94 Event Subscriber Client Facet

Table 107 describes the details of the *Event Subscriber Client* Facet. This Facet defines the ability to subscribe for *Event Notifications*. This includes basic *AddressSpace* concept and the browsing of it, adding events and event filters as monitored items and adding subscriptions.

Table 107 – Event Subscriber Client Facet

Group	Conformance Unit / Profile Title	Optional
Address Space Model	Address Space <i>Client</i> Base	False
Monitored Item Services	Monitor <i>Client</i> Complex Event Filter	True
Monitored Item Services	Monitor <i>Client</i> Event Filter	False
Monitored Item Services	Monitor <i>Client</i> Events	False
Monitored Item Services	Monitor <i>Client</i> Modify	True
Monitored Item Services	Monitor <i>Client</i> Trigger	True
Subscription Services	Subscription <i>Client</i> Basic	False
Subscription Services	Subscription <i>Client</i> Modify	True
Subscription Services	Subscription <i>Client</i> Multiple	True
Subscription Services	Subscription <i>Client</i> Republish	False
View Services	View <i>Client</i> Basic Browse	True
View Services	View <i>Client</i> TranslateBrowsePath	True

6.6.95 Base Event Processing Client Facet

Table 108 describes the details of the Base *Event Processing Client* Facet. This Facet defines the ability to subscribe for and process basic OPC UA Events. The *Client* has to support at least one of the Events in the Facet.

Table 108 – Base Event Processing Client Facet

Group	Conformance Unit / Profile Title	Optional
<i>Profile</i>	<i>Event Subscriber Client Facet</i>	False
Base Information	Base Info <i>Client</i> Change Events	True
Base Information	Base Info <i>Client</i> Device Failure	True
Base Information	Base Info <i>Client</i> Progress Events	True
Base Information	Base Info <i>Client</i> System Status	True
Base Information	Base Info <i>Client</i> System Status Underlying System	True
Base Information	Base Info <i>Event</i> Processing	False

6.6.96 Notifier and Source Hierarchy Client Facet

Table 109 describes the details of the Notifier and Source Hierarchy *Client* Facet. This Facet defines the ability to find and use a hierarchy of *Objects* that are event notifier and *Nodes* that are event sources in the *Server AddressSpace*.

Table 109 – Notifier and Source Hierarchy Client Facet

Group	Conformance Unit / Profile Title	Optional
<i>Profile</i>	<i>Event Subscriber Client Facet</i>	False
Address Space Model	Address Space <i>Client</i> Notifier Hierarchy	False
Address Space Model	Address Space <i>Client</i> Source Hierarchy	False
<i>Subscription Services</i>	<i>Subscription Client</i> Publish Configurable	False

6.6.97 A & C Base Condition Client Facet

Table 110 describes the details of the A & C Base *Condition Client* Facet. This Facet defines the ability to use the *Alarm* and *Condition* basic model. This includes the ability to subscribe for Events and to initiate a Refresh *Method*.

Table 110 – A & C Base Condition Client Facet

Group	Conformance Unit / Profile Title	Optional
<i>Profile</i>	<i>Event Subscriber Client Facet</i>	False
<i>Profile</i>	<i>Method Client Facet</i>	False
<i>Alarms and Conditions</i>	A & C Basic <i>Client</i>	False
<i>Alarms and Conditions</i>	A & C <i>Condition</i> Sub-Classes <i>Client</i>	True
<i>Alarms and Conditions</i>	A & C ConditionClasses <i>Client</i>	False
<i>Alarms and Conditions</i>	A & C Refresh <i>Client</i>	False

6.6.98 A & C Refresh2 Client Facet

Table 111 describes the details of the A & C Refresh2 *Client* Facet. This Facet enhances the A & C Base *Condition Server* Facet with the ability to initiate a ConditionRefresh2 *Method*.

Table 111 – A & C Refresh2 Client Facet

Group	Conformance Unit / Profile Title	Optional
<i>Profile</i>	A & C Base <i>Condition Client Facet</i>	False
<i>Alarms and Conditions</i>	A & C Refresh2 <i>Client</i>	False

6.6.99 A & C Address Space Instance Client Facet

Table 112 describes the details of the A & C Address Space Instance *Client Facet*. This Facet defines the ability to use *Condition* instances in the *AddressSpace*.

Table 112 – A & C Address Space Instance Client Facet

Group	Conformance Unit / Profile Title	Optional
<i>Alarms and Conditions</i>	A & C Instances <i>Client</i>	False

6.6.100 A & C Enable Client Facet

Table 113 describes the details of the A & C Enable *Client Facet*. This Facet defines the ability to enable and disable *Alarms*.

Table 113 – A & C Enable Client Facet

Group	Conformance Unit / Profile Title	Optional
<i>Profile</i>	A & C Base <i>Condition Client Facet</i>	False
<i>Alarms and Conditions</i>	A & C Enable <i>Client</i>	False

6.6.101 A & C AlarmMetrics Client Facet

Table 114 describes the details of the A & C AlarmMetrics *Client Facet*. This Facet defines the ability to use the AlarmMetrics model, i.e. understand and use the collected alarm metrics at any level in the HasNotifier hierarchy.

Table 114 – A & C AlarmMetrics Client Facet

Group	Conformance Unit / Profile Title	Optional
<i>Alarms and Conditions</i>	A & C Alarm Metrics <i>Client</i>	False

6.6.102 A & C Alarm Client Facet

Table 115 describes the details of the A & C Alarm *Client Facet*. This Facet defines the ability to use the alarming model (the AlarmType or any of the sub-types).

Table 115 – A & C Alarm Client Facet

Group	Conformance Unit / Profile Title	Optional
<i>Profile</i>	A & C Base <i>Condition Client</i> Facet	False
<i>Alarms and Conditions</i>	A & C Acknowledge <i>Client</i>	False
<i>Alarms and Conditions</i>	A & C Alarm <i>Client</i>	False
<i>Alarms and Conditions</i>	A & C Audible Sound <i>Client</i>	True
<i>Alarms and Conditions</i>	A & C Comment <i>Client</i>	True
<i>Alarms and Conditions</i>	A & C Confirm <i>Client</i>	True
<i>Alarms and Conditions</i>	A & C Discrepancy <i>Client</i>	True
<i>Alarms and Conditions</i>	A & C Discrete <i>Client</i>	False
<i>Alarms and Conditions</i>	A & C First in Group Alarm <i>Client</i>	True
<i>Alarms and Conditions</i>	A & C OffNormal <i>Client</i>	True
<i>Alarms and Conditions</i>	A & C On-Off Delay <i>Client</i>	True
<i>Alarms and Conditions</i>	A & C Out Of Service <i>Client</i>	True
<i>Alarms and Conditions</i>	A & C Re-Alarming <i>Client</i>	True
<i>Alarms and Conditions</i>	A & C Shelving <i>Client</i>	True
<i>Alarms and Conditions</i>	A & C Silencing <i>Client</i>	True
<i>Alarms and Conditions</i>	A & C Suppression by Operator <i>Client</i>	True
<i>Alarms and Conditions</i>	A & C Suppression <i>Client</i>	True
<i>Alarms and Conditions</i>	A & C SystemOffNormal <i>Client</i>	True
<i>Alarms and Conditions</i>	A & C Trip <i>Client</i>	True

6.6.103 A & C Exclusive Alarming Client Facet

Table 116 describes the details of the A & C Exclusive Alarming *Client* Facet. This Facet defines the ability to use the exclusive *Alarm* model. This includes understanding the various subtypes such as ExclusiveRateOfChangeAlarm, ExclusiveLevelAlarm and ExclusiveDeviationAlarm.

Table 116 – A & C Exclusive Alarming Client Facet

Group	Conformance Unit / Profile Title	Optional
<i>Profile</i>	A & C Alarm <i>Client</i> Facet	False
<i>Alarms and Conditions</i>	A & C Exclusive Deviation <i>Client</i>	True
<i>Alarms and Conditions</i>	A & C Exclusive Level <i>Client</i>	True
<i>Alarms and Conditions</i>	A & C Exclusive Limit <i>Client</i>	False
<i>Alarms and Conditions</i>	A & C Exclusive RateOfChange <i>Client</i>	True

6.6.104 A & C Non-Exclusive Alarming Client Facet

Table 117 describes the details of the A & C Non-Exclusive Alarming *Client* Facet. This Facet defines the ability to use the non-exclusive *Alarm* model. This includes understanding the various subtypes such as NonExclusiveRateOfChangeAlarm, NonExclusiveLevelAlarm and NonExclusiveDeviationAlarm.

Table 117 – A & C Non-Exclusive Alarming Client Facet

Group	Conformance Unit / Profile Title	Optional
Profile	A & C Alarm Client Facet	False
Alarms and Conditions	A & C Non-Exclusive Deviation Client	True
Alarms and Conditions	A & C Non-Exclusive Level Client	True
Alarms and Conditions	A & C Non-Exclusive Limit Client	False
Alarms and Conditions	A & C Non-Exclusive RateOfChange Client	True

6.6.105 A & C Previous Instances Client Facet

Table 118 describes the details of the A & C Previous Instances Client Facet. This Facet defines the ability to use previous instances of Alarms. This implies the ability to understand branchIds.

Table 118 – A & C Previous Instances Client Facet

Group	Conformance Unit / Profile Title	Optional
Profile	A & C Base Condition Client Facet	False
Alarms and Conditions	A & C Branch Client	False

6.6.106 A & C Dialog Client Facet

Table 119 describes the details of the A & C Dialog Client Facet. This Facet defines the ability to use the dialog model. This implies the support of Method invocation to respond to dialog messages.

Table 119 – A & C Dialog Client Facet

Group	Conformance Unit / Profile Title	Optional
Profile	A & C Base Condition Client Facet	False
Alarms and Conditions	A & C Dialog Client	False

6.6.107 A & C CertificateExpiration Client Facet

Table 120 describes the details of the A & C CertificateExpiration Client Facet. This Facet defines the ability to use the CertificateExpirationAlarmType.

Table 120 – A & C CertificateExpiration Client Facet

Group	Conformance Unit / Profile Title	Optional
Profile	A & C Base Condition Client Facet	False
Alarms and Conditions	A & C Acknowledge Client	True
Alarms and Conditions	A & C Alarm Client	False
Alarms and Conditions	A & C CertificateExpiration Client	False
Alarms and Conditions	A & C Comment Client	True
Alarms and Conditions	A & C Confirm Client	True
Alarms and Conditions	A & C Shelving Client	True

6.6.108 A & E Proxy Facet

Table 121 describes the details of the A & E Proxy Facet. This Facet describes the functionality used by a default A & E *Client* proxy. A *Client* exposes this Facet so that a *Server* may be able to better understand the commands that are being issued by the *Client*, since this Facet indicates that the *Client* is an A&E Com *Client*.

Table 121 – A & E Proxy Facet

Group	Conformance Unit / Profile Title	Optional
Address Space Model	Address Space <i>Client</i> Base	False
Alarms and Conditions	A & C Acknowledge <i>Client</i>	False
Alarms and Conditions	A & C Alarm <i>Client</i>	False
Alarms and Conditions	A & C Basic <i>Client</i>	False
Alarms and Conditions	A & C ConditionClasses <i>Client</i>	False
Alarms and Conditions	A & C Discrete <i>Client</i>	False
Alarms and Conditions	A & C Exclusive Deviation <i>Client</i>	False
Alarms and Conditions	A & C Exclusive Level <i>Client</i>	False
Alarms and Conditions	A & C Exclusive Limit <i>Client</i>	False
Alarms and Conditions	A & C Exclusive RateOfChange <i>Client</i>	False
Alarms and Conditions	A & C Instances <i>Client</i>	False
Alarms and Conditions	A & C Non-Exclusive Deviation <i>Client</i>	False
Alarms and Conditions	A & C Non-Exclusive Level <i>Client</i>	False
Alarms and Conditions	A & C Non-Exclusive Limit <i>Client</i>	False
Alarms and Conditions	A & C Non-Exclusive RateOfChange <i>Client</i>	False
Alarms and Conditions	A & C OffNormal <i>Client</i>	False
Alarms and Conditions	A & C Refresh <i>Client</i>	False
Alarms and Conditions	A & C SystemOffNormal <i>Client</i>	True
Alarms and Conditions	A & C Trip <i>Client</i>	False
Attribute Services	Attribute <i>Client</i> Read Base	False
Base Information	Base Info <i>Client</i> Basic	False
Base Information	Base Info <i>Client</i> Change Events	False
Discovery Services	Discovery <i>Client</i> Configure Endpoint	False
Discovery Services	Discovery <i>Client</i> Find Servers Basic	False
Discovery Services	Discovery <i>Client</i> Find Servers Dynamic	False
Discovery Services	Discovery <i>Client</i> Find Servers with URI	False
Discovery Services	Discovery <i>Client</i> Get Endpoints Basic	False
Discovery Services	Discovery <i>Client</i> Get Endpoints Dynamic	False
Method Services	Method <i>Client</i> Call	False
Monitored Item Services	Monitor <i>Client</i> Complex Event Filter	False
Monitored Item Services	Monitor <i>Client</i> Event Filter	False
Monitored Item Services	Monitor <i>Client</i> Events	False
Security	Security Administration	False
Security	Security Administration – XML Schema	False
Security	Security Certificate Administration	False
Session Services	Session <i>Client</i> Auto Reconnect	False
Subscription Services	Subscription <i>Client</i> Basic	False

Group	Conformance Unit / Profile Title	Optional
Subscription Services	Subscription Client Multiple	False
Subscription Services	Subscription Client Publish Configurable	False
Subscription Services	Subscription Client Republish	False
View Services	View Client Basic Browse	False
View Services	View Client Basic ResultSet Filtering	False
View Services	View Client TranslateBrowsePath	False

6.6.109 Method Client Facet

Table 122 describes the details of the *Method Client* Facet. This Facet defines the ability to call arbitrary Methods.

Table 122 – Method Client Facet

Group	Conformance Unit / Profile Title	Optional
Method Services	Method Client Call	False

6.6.110 Auditing Client Facet

Table 123 describes the details of the Auditing *Client* Facet. This Facet defines the ability to monitor Audit Events.

Table 123 – Auditing Client Facet

Group	Conformance Unit / Profile Title	Optional
Profile	Event Subscriber Client Facet	False
Auditing	Auditing Client Audit ID	False
Auditing	Auditing Client Subscribes	False

6.6.111 Node Management Client Facet

Table 124 describes the details of the *Node Management Client* Facet. This Facet defines the ability to configure the *AddressSpace* of an OPC UA Server through OPC UA *Node Management Service Set*.

Table 124 – Node Management Client Facet

Group	Conformance Unit / Profile Title	Optional
Address Space Model	Address Space Client Base	False
Node Management Services	Node Management Client	False

6.6.112 Advanced Type Programming Client Facet

Table 125 describes the details of the Advanced Type Programming *Client* Facet. This Facet defines the ability to use the type model and process the instance *AddressSpace* based on the type model. For example a client may contain generic displays that are based on a type, in that they contain a relative path from some main type. On call up this main type is matched to an instance and all of display items are resolved based on the provided type model.

Table 125 – Advanced Type Programming Client Facet

Group	Conformance Unit / Profile Title	Optional
Address Space Model	Address Space <i>Client</i> Base	False
Base Information	Base Info <i>Client</i> Basic	False
Base Information	Base Info <i>Client</i> Type Programming	False
View <i>Services</i>	View <i>Client</i> TranslateBrowsePath	False

6.6.113 User Role Management Client Facet

Table 126 describes the details of the User Role Management *Client Facet*. This *Facet* defines knowledge of the OPC UA *Information Model* for user roles and permissions and the use of the *Methods* to manage them.

Table 126 – User Role Management Client Facet

Group	Conformance Unit / Profile Title	Optional
Security	Security Role <i>Client</i> Base	False
Security	Security Role <i>Client</i> DefaultRolePermissions	False
Security	Security Role <i>Client</i> Management	False
Security	Security Role <i>Client</i> Restrict Applications	True
Security	Security Role <i>Client</i> Restrict Endpoints	True
Security	Security Role <i>Client</i> RolePermissions	False

6.6.114 State Machine Client Facet

Table 127 describes the details of the State Machine *Client Facet*. This *Facet* defines the ability to use state machines based on the *StateMachineType* or a sub-type.

Table 127 – State Machine Client Facet

Group	Conformance Unit / Profile Title	Optional
Base Information	Base Info Client Available States and Transitions	True
Base Information	Base Info <i>Client</i> Finite State Machine Instance	True
Base Information	Base Info <i>Client</i> State Machine Instance	False

6.6.115 Diagnostic Client Facet

Table 128 describes the details of the Diagnostic *Client Facet*. This *Facet* defines the ability to read and process diagnostic information that is part of the OPC UA information model.

Table 128 – Diagnostic Client Facet

Group	Conformance Unit / Profile Title	Optional
Address Space Model	Address Space <i>Client</i> Base	False
Base Information	Base Info <i>Client</i> Basic	False
Base Information	Base Info <i>Client</i> Diagnostics	False

6.6.116 Redundant Client Facet

Table 129 describes the details of the Redundant *Client* Facet. This Facet defines the ability to use the redundancy feature available for redundant *Clients*.

Table 129 – Redundant Client Facet

Group	Conformance Unit / <i>Profile</i> Title	Optional
Redundancy	Redundancy <i>Client</i>	False
<i>Subscription Services</i>	<i>Subscription Client</i> TransferSubscriptions	True

6.6.117 Redundancy Switch Client Facet

Table 130 describes the details of the Redundancy Switch *Client* Facet. A *Client* that supports this Facet supports monitoring the redundancy status for non-transparent redundant *Servers* and switching to the backup *Server* when they recognize a change.

Table 130 – Redundancy Switch Client Facet

Group	Conformance Unit / <i>Profile</i> Title	Optional
Redundancy	Redundancy <i>Client</i> Switch	False

6.6.118 Historical Access Client Facet

Table 131 describes the details of the Historical Access *Client* Facet. This Facet defines the ability to read, process, and update historical data.

Table 131 – Historical Access Client Facet

Group	Conformance Unit / <i>Profile</i> Title	Optional
<i>Attribute Services</i>	<i>Attribute Client</i> Historical Read	False
Historical Access	Historical Access <i>Client</i> Browse	False
Historical Access	Historical Access <i>Client</i> Read Raw	False

6.6.119 Historical Data AtTime Client Facet

Table 132 describes the details of the Historical Data AtTime *Client* Facet. This Facet defines the ability to access data at specific instances in time.

Table 132 – Historical Data AtTime Client Facet

Group	Conformance Unit / <i>Profile</i> Title	Optional
<i>Profile</i>	Historical Access <i>Client</i> Facet	False
Historical Access	Historical Access <i>Client</i> Time Instance	False

6.6.120 Historical Aggregate Client Facet

Table 133 describes the details of the Historical Aggregate *Client* Facet. This Facet defines the ability to read historical data by specifying the needed aggregate. This implies consideration of the list of aggregates supported by the *Server*.

Table 133 – Historical Aggregate Client Facet

Group	Conformance Unit / Profile Title	Optional
Aggregates	Aggregate – <i>Client</i> AnnotationCount	True
Aggregates	Aggregate – <i>Client</i> Average	True
Aggregates	Aggregate – <i>Client</i> Count	True
Aggregates	Aggregate – <i>Client</i> Custom Aggregates	True
Aggregates	Aggregate – <i>Client</i> Delta	True
Aggregates	Aggregate – <i>Client</i> DeltaBounds	True
Aggregates	Aggregate – <i>Client</i> DurationBad	True
Aggregates	Aggregate – <i>Client</i> DurationGood	True
Aggregates	Aggregate – <i>Client</i> DurationInStateNonZero	True
Aggregates	Aggregate – <i>Client</i> DurationInStateZero	True
Aggregates	Aggregate – <i>Client</i> End	True
Aggregates	Aggregate – <i>Client</i> EndBound	True
Aggregates	Aggregate – <i>Client</i> Interpolative	True
Aggregates	Aggregate – <i>Client</i> Maximum	True
Aggregates	Aggregate – <i>Client</i> Maximum2	True
Aggregates	Aggregate – <i>Client</i> MaximumActualTime	True
Aggregates	Aggregate – <i>Client</i> MaximumActualTime2	True
Aggregates	Aggregate – <i>Client</i> Minimum	True
Aggregates	Aggregate – <i>Client</i> Minimum2	True
Aggregates	Aggregate – <i>Client</i> MinimumActualTime	True
Aggregates	Aggregate – <i>Client</i> MinimumActualTime2	True
Aggregates	Aggregate – <i>Client</i> NumberOfTransitions	True
Aggregates	Aggregate – <i>Client</i> PercentBad	True
Aggregates	Aggregate – <i>Client</i> PercentGood	True
Aggregates	Aggregate – <i>Client</i> Range	True
Aggregates	Aggregate – <i>Client</i> Range2	True
Aggregates	Aggregate – <i>Client</i> StandardDeviationPopulation	True
Aggregates	Aggregate – <i>Client</i> StandardDeviationSample	True
Aggregates	Aggregate – <i>Client</i> Start	True
Aggregates	Aggregate – <i>Client</i> StartBound	True
Aggregates	Aggregate – <i>Client</i> TimeAverage	True
Aggregates	Aggregate – <i>Client</i> TimeAverage2	True
Aggregates	Aggregate – <i>Client</i> Total	True
Aggregates	Aggregate – <i>Client</i> Total2	True
Aggregates	Aggregate – <i>Client</i> Usage	False
Aggregates	Aggregate – <i>Client</i> VariancePopulation	True
Aggregates	Aggregate – <i>Client</i> VarianceSample	True
Aggregates	Aggregate – <i>Client</i> WorstQuality	True
Aggregates	Aggregate – <i>Client</i> WorstQuality2	True
Historical Access	Historical Access <i>Client</i> Read Aggregates	False

6.6.121 Historical Annotation Client Facet

Table 134 describes the details of the Historical Annotation *Client* Facet. This Facet defines the ability to retrieve and write annotations for historical data.

Table 134 – Historical Annotation Client Facet

Group	Conformance Unit / Profile Title	Optional
Profile	Historical Access <i>Client</i> Facet	False
Profile	Historical Data Update <i>Client</i> Facet	False
Historical Access	Historical Access <i>Client</i> Annotations	False

6.6.122 Historical Access Modified Data Client Facet

Table 135 describes the details of the Historical Access Modified Data *Client* Facet. This Facet defines the ability to access prior historical data (values that were modified or inserted).

Table 135 – Historical Access Modified Data Client Facet

Group	Conformance Unit / Profile Title	Optional
Profile	Historical Access <i>Client</i> Facet	False
Historical Access	Historical Access <i>Client</i> Read Modified	False

6.6.123 Historical Data Insert Client Facet

Table 136 describes the details of the Historical Data Insert *Client* Facet. This Facet defines the ability to insert historical data.

Table 136 – Historical Data Insert Client Facet

Group	Conformance Unit / Profile Title	Optional
Attribute Services	Attribute <i>Client</i> Historical Updates	False
Historical Access	Historical Access <i>Client</i> Data Insert	False

6.6.124 Historical Data Update Client Facet

Table 137 describes the details of the Historical Data Update *Client* Facet. This Facet defines the ability to update historical data.

Table 137 – Historical Data Update Client Facet

Group	Conformance Unit / Profile Title	Optional
Attribute Services	Attribute <i>Client</i> Historical Updates	False
Historical Access	Historical Access <i>Client</i> Data Update	False

6.6.125 Historical Data Replace Client Facet

Table 138 describes the details of the Historical Data Replace *Client* Facet. This Facet defines the ability to replace historical data.

Table 138 – Historical Data Replace Client Facet

Group	Conformance Unit / Profile Title	Optional
<i>Attribute Services</i>	<i>Attribute Client</i> Historical Updates	False
Historical Access	Historical Access <i>Client</i> Data Replace	False

6.6.126 Historical Data Delete Client Facet

Table 139 describes the details of the Historical Data Delete *Client* Facet. This Facet defines the ability to delete historical data.

Table 139 – Historical Data Delete Client Facet

Group	Conformance Unit / Profile Title	Optional
<i>Attribute Services</i>	<i>Attribute Client</i> Historical Updates	False
Historical Access	Historical Access <i>Client</i> Data Delete	False

6.6.127 Historical Access Client Server Timestamp Facet

Table 140 describes the details of the Historical Access *Client Server* Timestamp Facet. This Facet defines the ability to request and process *Server* timestamps, in addition to source timestamps.

Table 140 – Historical Access Client Server Timestamp Facet

Group	Conformance Unit / Profile Title	Optional
Historical Access	Historical Access <i>Client Server</i> Timestamp	False

6.6.128 Historical Structured Data Access Client Facet

Table 141 describes the details of the Historical Structured Data Access *Client* Facet. This Facet defines the ability to read structured values for historical nodes.

Table 141 – Historical Structured Data Access Client Facet

Group	Conformance Unit / Profile Title	Optional
<i>Profile</i>	Historical Access <i>Client</i> Facet	False
Historical Access	Historical Access <i>Client</i> Structure Data Raw	False

6.6.129 Historical Structured Data AtTime Client Facet

Table 142 describes the details of the Historical Structured Data AtTime *Client* Facet. This Facet defines the ability to read structured values for historical nodes at specific instances in time.

Table 142 – Historical Structured Data AtTime Client Facet

Group	Conformance Unit / Profile Title	Optional
<i>Profile</i>	Historical Data AtTime <i>Client</i> Facet	False
Historical Access	Historical Access <i>Client</i> Structure Data Time Instance	False

6.6.130 Historical Structured Data Modified Client Facet

Table 143 describes the details of the Historical Structured Data Modified *Client* Facet. This Facet defines the ability to read structured values for prior historical data (values that were modified or inserted).

Table 143 – Historical Structured Data Modified Client Facet

Group	Conformance Unit / Profile Title	Optional
Profile	Historical Access Modified Data <i>Client</i> Facet	False
Historical Access	Historical Access <i>Client</i> Structure Data Read Modified	False

6.6.131 Historical Structured Data Insert Client Facet

Table 144 describes the details of the Historical Structured Data Insert *Client* Facet. This Facet defines the ability to insert structured historical data.

Table 144 – Historical Structured Data Insert Client Facet

Group	Conformance Unit / Profile Title	Optional
Profile	Historical Data Insert <i>Client</i> Facet	False
Historical Access	Historical Access <i>Client</i> Structure Data Insert	False

6.6.132 Historical Structured Data Update Client Facet

Table 145 describes the details of the Historical Structured Data Update *Client* Facet. This Facet defines the ability to update structured historical data.

Table 145 – Historical Structured Data Update Client Facet

Group	Conformance Unit / Profile Title	Optional
Profile	Historical Data Update <i>Client</i> Facet	False
Historical Access	Historical Access <i>Client</i> Structure Data Update	False

6.6.133 Historical Structured Data Replace Client Facet

Table 146 describes the details of the Historical Structured Data Replace *Client* Facet. This Facet defines the ability to replace structured historical data.

Table 146 – Historical Structured Data Replace Client Facet

Group	Conformance Unit / Profile Title	Optional
Profile	Historical Data ~~Update~~ Replace <i>Client</i> Facet	False
Historical Access	Historical Access <i>Client</i> Structure Data Replace	False

6.6.134 Historical Structured Data Delete Client Facet

Table 147 describes the details of the Historical Structured Data Delete *Client* Facet. This Facet defines the ability to remove structured historical data.

Table 147 – Historical Structured Data Delete Client Facet

Group	Conformance Unit / Profile Title	Optional
<i>Profile</i>	Historical Data Delete <i>Client</i> Facet	False
Historical Access	Historical Access <i>Client</i> Structure Data Delete	False

6.6.135 Historical Events Client Facet

Table 148 describes the details of the Historical Events *Client* Facet. This Facet defines the ability to read Historical Events, including simple filtering.

Table 148 – Historical Events Client Facet

Group	Conformance Unit / Profile Title	Optional
<i>Attribute Services</i>	<i>Attribute Client</i> Historical Read	False
Historical Access	Historical Access <i>Client</i> Read Events	False

6.6.136 Historical Event Insert Client Facet

Table 149 describes the details of the Historical *Event* Insert *Client* Facet. This Facet defines the ability to insert historical events.

Table 149 – Historical Event Insert Client Facet

Group	Conformance Unit / Profile Title	Optional
<i>Attribute Services</i>	<i>Attribute Client</i> Historical Updates	False
Historical Access	Historical Access <i>Client</i> Event Inserts	False

6.6.137 Historical Event Update Client Facet

Table 150 describes the details of the Historical *Event* Update *Client* Facet. This Facet defines the ability to update historical events.

Table 150 – Historical Event Update Client Facet

Group	Conformance Unit / Profile Title	Optional
<i>Attribute Services</i>	<i>Attribute Client</i> Historical Updates	False
Historical Access	Historical Access <i>Client</i> Event Updates	False

6.6.138 Historical Event Replace Client Facet

Table 151 describes the details of the Historical *Event* Replace *Client* Facet. This Facet defines the ability to replace historical events.

Table 151 – Historical Event Replace Client Facet

Group	Conformance Unit / Profile Title	Optional
<i>Attribute Services</i>	<i>Attribute Client</i> Historical Updates	False
Historical Access	Historical Access <i>Client</i> Event Replaces	False

6.6.139 Historical Event Delete Client Facet

Table 152 describes the details of the Historical *Event Delete Client* Facet. This Facet defines the ability to delete Historical events.

Table 152 – Historical Event Delete Client Facet

Group	Conformance Unit / Profile Title	Optional
Attribute Services	Attribute Client Historical Updates	False
Historical Access	Historical Access Client Event Deletes	False

6.6.140 Aggregate Subscriber Client Facet

Table 153 describes the details of the Aggregate Subscriber *Client* Facet. This Facet defines the ability to use the aggregate filter when subscribing for *Attribute* values.

Table 153 – Aggregate Subscriber Client Facet

Group	Conformance Unit / Profile Title	Optional
Aggregates	Aggregate Subscription – Client AnnotationCount	True
Aggregates	Aggregate Subscription – Client Average	True
Aggregates	Aggregate Subscription – Client Count	True
Aggregates	Aggregate Subscription – Client Custom Aggregates	True
Aggregates	Aggregate Subscription – Client Delta	True
Aggregates	Aggregate Subscription – Client DeltaBounds	True
Aggregates	Aggregate Subscription – Client DurationBad	True
Aggregates	Aggregate Subscription – Client DurationGood	True
Aggregates	Aggregate Subscription – Client DurationInStateNonZero	True
Aggregates	Aggregate Subscription – Client DurationInStateZero	True
Aggregates	Aggregate Subscription – Client End	True
Aggregates	Aggregate Subscription – Client EndBound	True
Aggregates	Aggregate Subscription – Client Filter	False
Aggregates	Aggregate Subscription – Client Interpolative	True
Aggregates	Aggregate Subscription – Client Maximum	True
Aggregates	Aggregate Subscription – Client Maximum2	True
Aggregates	Aggregate Subscription – Client MaximumActualTime	True
Aggregates	Aggregate Subscription – Client MaximumActualTime2	True
Aggregates	Aggregate Subscription – Client Minimum	True
Aggregates	Aggregate Subscription – Client Minimum2	True
Aggregates	Aggregate Subscription – Client MinimumActualTime	True
Aggregates	Aggregate Subscription – Client MinimumActualTime2	True
Aggregates	Aggregate Subscription – Client NumberOfTransitions	True
Aggregates	Aggregate Subscription – Client PercentBad	True
Aggregates	Aggregate Subscription – Client PercentGood	True
Aggregates	Aggregate Subscription – Client Range	True
Aggregates	Aggregate Subscription – Client Range2	True
Aggregates	Aggregate Subscription – Client StandardDevPopulation StandardDeviationPopulation	True

Group	Conformance Unit / Profile Title	Optional
Aggregates	Aggregate <i>Subscription</i> – Client ~~StandardDevSample~~ StandardDeviationSample	True
Aggregates	Aggregate <i>Subscription</i> – Client Start	True
Aggregates	Aggregate <i>Subscription</i> – Client StartBound	True
Aggregates	Aggregate <i>Subscription</i> – Client TimeAverage	True
Aggregates	Aggregate <i>Subscription</i> – Client TimeAverage2	True
Aggregates	Aggregate <i>Subscription</i> – Client Total	True
Aggregates	Aggregate <i>Subscription</i> – Client Total2	True
Aggregates	Aggregate <i>Subscription</i> – Client VariancePopulation	True
Aggregates	Aggregate <i>Subscription</i> – Client VarianceSample	True
Aggregates	Aggregate <i>Subscription</i> – Client WorstQuality	True
Aggregates	Aggregate <i>Subscription</i> – Client WorstQuality2	True
Monitored Item <i>Services</i>	Monitor <i>Client</i> Aggregate Filter	False
Monitored Item <i>Services</i>	Monitor <i>Client</i> by Index	False
Monitored Item <i>Services</i>	Monitor <i>Client</i> Modify	True
Monitored Item <i>Services</i>	Monitor <i>Client</i> Value Change	False
<i>Subscription Services</i>	<i>Subscription Client</i> Basic	False
<i>Subscription Services</i>	<i>Subscription Client</i> Modify	True
<i>Subscription Services</i>	<i>Subscription Client</i> Multiple	True
<i>Subscription Services</i>	<i>Subscription Client</i> Republish	True

6.6.141 Standard UA Client Profile

Standard UA Client Profile has been deprecated as it is superseded with Standard UA Client 2017 Profile in v1.04.

6.6.142 Standard UA Client 2017 Profile

Table 154 describes the details of the Standard UA *Client* 2017 *Profile*. This *Profile* is a FullFeatured *Profile* that defines a minimum set of functionality required for generic OPC UA *Clients*. Such a *Client* shall be able to use local, subnet and global discovery. It shall be able to maintain a connection with a single *Session* (as required for nano embedded *Servers*). If *Subscriptions* are used, the *Client* shall respect the limits of *Servers* with limited resources. If a *Server* does not support *Subscriptions*, the *Client* shall provide read access as fallback. The *Client* shall provide connection establishment through the OPC UA TCP binary protocol with and without security.

This *Profile* supersedes the "Standard UA *Client Profile*".

Table 154 – Standard UA Client 2017 Profile

Group	Conformance Unit / Profile Title	Optional
Profile	AddressSpace Lookup Client Facet	False
Profile	Attribute Read Client Facet	False
Profile	Attribute Write Client Facet	False
Profile	Base Client Behaviour Facet	False
Profile	Core 2017 Client Facet	False
Profile	DataChange Subscriber Client Facet	False
Profile	Discovery Client Facet	False
Profile	Entry Level Support 2015 Client Facet	False
Profile	Global Certificate Management Client Facet	False
Profile	Global Discovery Client Facet	False
Profile	Method Client Facet	False
Profile	Subnet Discovery Client Facet	False
Profile	UA-TCP UA-SC UA-Binary	False
Profile	User Token – Anonymous Facet	False

6.6.143 UA-TCP UA-SC UA-Binary

Table 155 describes the details of the UA-TCP UA-SC UA-Binary. This transport Facet defines a combination of network protocol, security protocol and message encoding that is optimized for low resource consumption and high performance. It combines the simple TCP based network protocol UA-TCP 1.0 with the binary security protocol UA-SecureConversation 1.0 and the binary message encoding UA-Binary 1.0.

Table 155 – UA-TCP UA-SC UA-Binary

Group	Conformance Unit / Profile Title	Optional
Protocol and Encoding	Protocol UA TCP ~~Binary~~ UA Security	False
Protocol and Encoding	UA Binary Encoding	False
Protocol and Encoding	UA Secure Conversation	False

6.6.144 HTTPS UA-Binary

Table 156 describes the details of the HTTPS UA-Binary. This transport Facet defines a combination of network protocol, security protocol and message encoding that balances compatibility with widely used HTTPS transport and a compact UA-Binary encoded message for added performance. It is expected that this transport will be used to support installations where firewalls only permit HTTPS or where a WEB browser is used as *Client*. This transport requires that one of the TransportSecurity Profiles for TLS be provided.

Table 156 – HTTPS UA-Binary

Group	Conformance Unit / Profile Title	Optional
Protocol and Encoding	Protocol HTTPS ~~with UA Binary~~	False
Protocol and Encoding	UA Binary Encoding	False
Security	Security TLS General	False

6.6.145 HTTPS UA-XML

Table 157 describes the details of the HTTPS UA-XML. This transport Facet defines a combination of network protocol, security protocol and message encoding that uses HTTPS transport and a SOAP XML encoded message for use with standard SOAP V1.2 toolkits. This transport requires that one of the TransportSecurity *Profiles* for TLS be provided.

Table 157 – HTTPS UA-XML

Group	Conformance Unit / Profile Title	Optional
Protocol and Encoding	Protocol HTTPS ~~with Soap~~	False
Protocol and Encoding	UA SOAP-XML Encoding	False
Security	Security TLS General	False

6.6.146 HTTPS UA-JSON

Table 158 describes the details of the HTTPS UA-JSON. This transport Facet defines a combination of network protocol, security protocol and message encoding that uses HTTPS transport and a UA-JSON encoded message. This transport requires that one of the TransportSecurity *Profiles* for TLS be provided.

Table 158 – HTTPS UA-JSON

Group	Conformance Unit / Profile Title	Optional
Protocol and Encoding	JSON Reversible Encoding	False
Protocol and Encoding	Protocol HTTPS	False
Security	Security TLS General	False

~~6.5.113 SOAP HTTP WS-SC UA XML~~

~~Table 134 describes the details of the SOAP HTTP WS-SC UA XML. This transport Facet defines a combination of network protocol, security protocol and message encoding that provides maximum compatibility with enterprise class web service applications through the use of XML encoded SOAP messages. The performance of this transport profile will not be as good as the profiles with binary encoded messages. It requires support for SOAP 1.2, WS-Secure Conversation and the UA XML Encoding 1.0.~~

~~**Table 134 – SOAP HTTP WS-SC UA XML**~~

~~Group~~	~~Conformance Unit / Profile Title~~	~~Optional~~
~~Protocol and Encoding~~	~~Protocol Soap Xml WS Security~~	~~False~~

~~6.5.114 SOAP HTTP WS-SC UA Binary~~

~~Table 135 describes the details of the SOAP HTTP WS-SC UA Binary. This transport Facet defines a combination of network protocol, security protocol and message encoding that balances compatibility with enterprise class web service applications and performance through the use of SOAP message bodies that contain UA binary encoded messages. It requires support for SOAP 1.2, WS-Secure Conversation and the UA Binary Encoding 1.0.~~

~~**Table 135 – SOAP HTTP WS-SC UA Binary**~~

~~Group~~	~~Conformance Unit / Profile Title~~	~~Optional~~
~~Protocol and Encoding~~	~~Protocol Soap Binary WS Security~~	~~False~~

~~6.5.115 SOAP HTTP WS-SC UA XML-UA Binary~~

~~Table 136 describes the details of the SOAP HTTP WS-SC UA XML-UA Binary. This transport Facet combines the SOAP HTTP WS-SC UA Binary and SOAP HTTP WS-SC UA XML Facets. It is used by Servers that allow the Client to choose whether messages are encoded with XML or Binary. It requires support for SOAP 1.2, WS-Secure Conversation and the UA Binary Encoding 1.0 and the UA XML Encoding 1.0.~~

~~Table 136 – SOAP HTTP WS-SC UA XML-UA Binary~~

~~Group~~	~~Conformance Unit / Profile Title~~	~~Optional~~
~~Protocol and Encoding~~	~~Protocol Soap Binary WS Security~~	~~False~~
~~Protocol and Encoding~~	~~Protocol Soap Xml WS Security~~	~~False~~

6.6.147 WSS UA-SC UA-Binary

Table 159 describes the details of the WSS UA-SC UA-Binary. This transport Facet defines a combination of network protocol, security protocol and message encoding that uses WSS transport as a tunnel for UA-SecureConversation and UA-Binary encoded messages. Although transport security is available in WSS via TLS, additional message security can be used to assure end-to-end security.

Table 159 – WSS UA-SC UA-Binary

Group	Conformance Unit / Profile Title	Optional
Protocol and Encoding	Protocol Web Sockets	False
Protocol and Encoding	UA Binary Encoding	False
Protocol and Encoding	UA Secure Conversation	False
Security	Security TLS General	False

6.6.148 WSS UA-JSON

Table 160 describes the details of the WSS UA-JSON. This transport Facet defines a combination of network protocol, security protocol and message encoding that uses WSS transport with UA-JSON encoded messages.

Table 160 – WSS UA-JSON

Group	Conformance Unit / Profile Title	Optional
Protocol and Encoding	JSON Reversible Encoding	False
Protocol and Encoding	Protocol Web Sockets	False
Security	Security TLS General	False

6.6.149 Security User Access Control Full

Table 161 describes the details of the Security User Access Control Full. A Server that supports this profile supports restricting multiple levels of access to all Nodes in the AddressSpace based on the validated user.

Table 161 – Security User Access Control Full

Group	Conformance Unit / Profile Title	Optional
<i>Profile</i>	Security User Access Control Base	False
Address Space Model	Address Space User Access Level Full	False

6.6.150 Security User Access Control Base

Table 162 describes the details of the Security User Access Control Base. A *Server* that supports this profile supports restricting some level of access to some *Nodes* in the *AddressSpace* based on the validated user.

Table 162 – Security User Access Control Base

Group	Conformance Unit / Profile Title	Optional
Address Space Model	Address Space User Access Level Base	False
Security	Security User IssuedToken Kerberos	True
Security	Security User IssuedToken Kerberos Windows	True
Security	Security User Name Password	False
Security	Security User X509	True

6.6.151 Security Time Synchronization

Table 163 describes the details of the Security Time Synchronization. This Facet indicates that the application supports the minimum required level of time synchronization to ensure secure communication. One of the optional time synchronization conformance units ~~must~~ shall be supported.

Table 163 – Security Time Synchronization

Group	Conformance Unit / Profile Title	Optional
Security	Security Time Synch – Configuration	False
Security	Security Time Synch – NTP / OS Based support	True
Security	Security Time Synch – UA based support	True

6.6.152 Best Practice – Audit Events

Table 164 describes the details of the Best Practice – Audit Events. Subscriptions for Audit Events shall be restricted to authorized personnel.

Table 164 – Best Practice – Audit Events

Group	Conformance Unit / Profile Title	Optional
Miscellaneous	Best Practice – Audit Events	False

6.6.153 Best Practice – Alarm Handling

Table 165 describes the details of the Best Practice – *Alarm* Handling. A *Server* should restrict critical alarm handling functionality to users that have the appropriate rights to perform these actions.

Table 165 – Best Practice – Alarm Handling

Group	Conformance Unit / Profile Title	Optional
Miscellaneous	Best Practice – <i>Alarm</i> Handling	False

6.6.154 Best Practice – Random Numbers

Table 166 describes the details of the Best Practice – Random Numbers. All random numbers that are required for security should use appropriate cryptographic library based random number generators.

Table 166 – Best Practice – Random Numbers

Group	Conformance Unit / Profile Title	Optional
Miscellaneous	Best Practice – Random Numbers	False

6.6.155 Best Practice – Timeouts

Table 167 describes the details of the Best Practice – Timeouts. The administrator should be able to configure reasonable timeouts for Secure Channels, Sessions and Subscriptions. Setting these timeouts allows limiting Denial of Service attacks and overload issues.

Table 167 – Best Practice – Timeouts

Group	Conformance Unit / Profile Title	Optional
Miscellaneous	Best Practice – Timeouts	False

6.6.156 Best Practice – Administrative Access

Table 168 describes the details of the Best Practice – Administrative Access. The *Server* and *Client* allow restricting the use of certain *Services* and access to parts of the *AddressSpace* to administrative personnel. This includes multiple level of administrative access on platforms that support multiple administrative roles (such as Windows or Linux).

Table 168 – Best Practice – Administrative Access

Group	Conformance Unit / Profile Title	Optional
Miscellaneous	Best Practice – Administrative Access	False

6.6.157 Best Practice – Strict Message Handling

Table 169 describes the details of the Best Practice – Strict Message Handling. *Server* and *Client* reject messages that are incorrectly formed as specified in IEC 62541-4 and IEC 62541-6.

Table 169 – Best Practice – Strict Message Handling

Group	Conformance Unit / Profile Title	Optional
Miscellaneous	Best Practice – Strict Message Handling	False

6.6.158 Best Practice – Audit Events Client

Table 170 describes the details of the Best Practice – Audit Events *Client*. Audit Tracking system connect to a *Server* using a Secure Channel and under the appropriate authorization to allow access to Audit Events.

Table 170 – Best Practice – Audit Events Client

Group	Conformance Unit / Profile Title	Optional
Miscellaneous	Best Practice – Audit Events <i>Client</i>	False

6.5.132 TransportSecurity – TLS 1.0

Table 153 describes the details of the TransportSecurity – TLS 1.0 Profile. This Facet defines a transport security for configurations with medium-high security needs. It makes uses of TLS_RSA_WITH_RC4_128_SHA. This security profile is less secure than TLS 1.2.

As computing power increases, security algorithms are expected to expire. NIST provides guidelines for expected expiration dates for individual algorithms. These guidelines provide recommended dates at which the algorithm should be replaced or upgraded to a more secure algorithm. They do not indicate a failure of the algorithm. NIST already recommends users of this TransportSecurity should upgrade to TLS 1.2. This Policy is listed for systems that do not support TLS 1.1 or 1.2. It is recommended that *Servers* and *Client* support all security profiles and developers provide the recommended profile as a default. It is up to an administrator to configure the actual exposed TransportSecurity Profiles.

Table 153 – TransportSecurity – TLS 1.0

Group	Conformance Unit / Profile Title	Optional
Security	Security Level 4	False
Security	Security TLS_RSA_WITH_RC4_128_SHA	False

6.5.133 TransportSecurity – TLS 1.1

Table 154 describes the details of the TransportSecurity – TLS 1.1. This Facet defines a transport security for configurations with medium-high security needs. This security profile is less secure than TLS 1.2.

As computing power increases, security algorithms are expected to expire. NIST provides guidelines for expected expiration dates for individual algorithms. These guidelines provided recommended dates at which the algorithm should be replaced or upgraded to a more secure algorithm. They do not indicate a failure of the algorithm. NIST recommends users of this TransportSecurity should consider upgrading to TLS 1.2. This transport security is provided for systems that do not support TLS 1.2. It is recommended that *Servers* and *Client* support all security profiles and developers provide the recommended profile as a default. It is up to an administrator to configure the actual exposed TransportSecurity Profiles.

Table 154 – TransportSecurity – TLS 1.1

Group	Conformance Unit / Profile Title	Optional
Security	Security Level 2	False
Security	Security TLS 1.1	False

6.6.159 TransportSecurity – TLS 1.2

Table 171 describes the details of the **SecurityPolicy** **TransportSecurity** – TLS 1.2. This Facet defines a transport security for configurations with high security needs. It makes use of TLS 1.2 and uses TLS_RSA_WITH_AES_256_CBC_SHA256. As computing power increases, security algorithms are expected to expire. NIST provides guidelines for expected expiration dates for individual algorithms. These guidelines provide recommended dates at which the algorithm should be replaced or upgraded to a more secure algorithm. They do not indicate a failure of the algorithm. NIST has no recommendations for this TransportSecurity. It is recommended that *Servers* and *Clients* support all security profiles and developers provide the recommended profile as a default. It is up to an administrator to configure the actual exposed TransportSecurity *Profiles*.

Table 171 – TransportSecurity – TLS 1.2

Group	Conformance Unit / Profile Title	Optional
Security	Security Level 3	False
Security	Security TLS_RSA_WITH_ with AES_256_CBC_SHA256	False

6.6.160 TransportSecurity – TLS 1.2 with PFS

Table 172 describes the details of the TransportSecurity – TLS 1.2 with PFS. This Facet defines a transport security for configurations with high security needs and perfect forward secrecy (PFS). It makes use of TLS 1.2 and uses TLS_DHE_RSA_WITH_AES_128_CBC_SHA256 or TLS_DHE_RSA_WITH_AES_256_CBC_SHA256.

As computing power increases, security algorithms are expected to expire. NIST provides guidelines for expected expiration dates for individual algorithms. These guidelines provide recommended dates at which the algorithm should be replaced or upgraded to a more secure algorithm. They do not indicate a failure of the algorithm. NIST has no recommendations for this TransportSecurity. It is recommended that *Servers* and *Clients* support all security profiles and developers provide the recommended profile as a default. It is up to an administrator to configure the actual exposed TransportSecurity *Profiles*.

Table 172 – TransportSecurity – TLS 1.2 with PFS

Group	Conformance Unit / Profile Title	Optional
Security	Security TLS_DHE_RSA with AES_nnn_CBC_SHA256	False

6.6.161 SecurityPolicy – None

Table 173 describes the details of the SecurityPolicy – None. This security Facet defines a security policy used for configurations with the lowest security needs. This security policy can affect the behaviour of the *CreateSession* and *ActivateSession Services*. It also results in a SecureChannel which has no channel security. By default this security policy should be disabled if any other security policies are available.

Table 173 – SecurityPolicy – None

Group	Conformance Unit / Profile Title	Optional
Security	Security None	False
Security	AsymmetricEncryptionAlgorithm_None	False
Security	AsymmetricSignatureAlgorithm_None	False
Security	KeyDerivationAlgorithm_None	False
Security	Security None CreateSession ActivateSession	False
Security	Security None CreateSession ActivateSession 1.0	True
Security	SecurityPolicy_None_Limits	False
Security	SymmetricEncryptionAlgorithm_None	False
Security	SymmetricSignatureAlgorithm_None	False

6.6.162 SecurityPolicy – Basic128Rsa15

~~Table 150 describes the details of the SecurityPolicy – Basic128Rsa15. This security Facet defines a Security Policy for configurations with medium security. It requires a PKI infrastructure.~~

~~As computing power increases, SecurityPolicies are expected to expire. NIST provides guidelines for expected expiration dates for individual algorithms. These guidelines provided recommended dates at which the algorithm should be replaced or upgraded to a more secure algorithm. They do not indicate a failure of the algorithm. NIST recommends users of this SecurityPolicy should consider upgrading it for key lengths less than 2048 in 2010. NIST also recommends that this SecurityPolicy should be deprecated in 2012 for key lengths less than 2048. It is recommended that Servers and Client support all security profiles and developers provide the recommended profile as a default. It is up to an administrator to configure the actual exposed SecurityPolicies.~~

Table 150 – SecurityPolicy – Basic128Rsa15

Group	Conformance Unit / Profile Title	Optional
Security	Security Basic 128Rsa15	False
Security	Security Certificate Validation	False
Security	Security Encryption Required	False
Security	Security Level 1	False
Security	Security Signing Required	False

SecurityPolicy – Basic128Rsa15 has been deprecated in v1.04 since the hash algorithm Sha-1 is not considered secure anymore.

6.6.163 SecurityPolicy – Basic256

~~Table 151 describes the details of the SecurityPolicy – Basic256. This security Facet defines a Security Policy for configurations with medium to high security needs. It requires a PKI infrastructure.~~

~~As computing power increases, SecurityPolicies are expected to expire. NIST provides guidelines for expected expiration dates for individual algorithms. These guidelines provided recommended dates at which the algorithm should be replaced or upgraded to a more secure algorithm. They do not indicate a failure of the algorithm. NIST recommends users of this SecurityPolicy should consider upgrading it for key sizes less than 2048 in 2010. NIST also recommends that this SecurityPolicy should be deprecated in 2012 for key sizes less than 2048. It is recommended that Servers and Client support all security profiles and developers provide the recommended profile as a default. It is up to an administrator to configure the actual exposed SecurityPolicies.~~

Table 151 – SecurityPolicy – Basic256

Group	Conformance Unit / Profile Title	Optional
Security	Security-Basic-256	False
Security	Security Certificate Validation	False
Security	Security Encryption Required	False
Security	Security Level 2	False
Security	Security Signing Required	False

SecurityPolicy – Basic128Rsa15 has been deprecated in v1.04 since the hash algorithm Sha-1 is not considered secure anymore.

6.6.164 SecurityPolicy [A] – Aes128-Sha256-RsaOaep

Table 174 describes the details of the SecurityPolicy [A] – Aes128-Sha256-RsaOaep. This security Facet defines a security policy for configurations with average security needs. It requires a PKI infrastructure. As computing power increases, security policies are expected to expire. NIST provides guidelines for expected expiration dates for individual algorithms. These guidelines provide recommended dates at which the algorithm should be replaced or upgraded to a more secure algorithm. They do not indicate a failure of the algorithm. This security policy has no published end dates as of this time. It is recommended that *Servers* and *Clients* support all security profiles and support configurability of actual exposed and default security policies.

Table 174 – SecurityPolicy [A] – Aes128-Sha256-RsaOaep

Group	Conformance Unit / Profile Title	Optional
Security	Aes128-Sha256-RsaOaep_Limits	False
Security	AsymmetricEncryptionAlgorithm_RSA-OAEP-SHA1	False
Security	AsymmetricSignatureAlgorithm_RSA-PKCS15-SHA2-256	False
Security	CertificateSignatureAlgorithm_RSA-PKCS15-SHA2-256	False
Security	KeyDerivationAlgorithm_P-SHA2-256	False
Security	Security Certificate Validation	False
Security	Security Encryption Required	False
Security	Security Signing Required	False
Security	SymmetricEncryptionAlgorithm_AES128-CBC	False
Security	SymmetricSignatureAlgorithm_HMAC-SHA2-256	False

6.6.165 SecurityPolicy [B] – Basic256Sha256

Table 175 describes the details of the SecurityPolicy [B] – Basic256Sha256. This security Facet defines a security policy for configurations with high security needs. It requires a PKI infrastructure.

As computing power increases, security policies are expected to expire. NIST provides guidelines for expected expiration dates for individual algorithms. These guidelines provided recommended dates at which the algorithm should be replaced or upgraded to a more secure algorithm. They do not indicate a failure of the algorithm. This security policy has no published end dates as of this time. It is recommended that *Servers* and *Clients* support all security profiles and developers provide the recommended profile as a default. It is up to an administrator to configure the actual exposed security policies.

Table 175 – SecurityPolicy [B] – Basic256Sha256

Group	Conformance Unit / Profile Title	Optional
Security	Security-Basic-256-Sha256	False
Security	Security-Level-3	False
Security	AsymmetricEncryptionAlgorithm_RSA-OAEP-SHA1	False
Security	AsymmetricSignatureAlgorithm_RSA-PKCS15-SHA2-256	False
Security	Basic256Sha256_Limits	False
Security	CertificateSignatureAlgorithm_RSA-PKCS15-SHA2-256	False
Security	KeyDerivationAlgorithm_P-SHA2-256	False
Security	Security Certificate Validation	False
Security	Security Encryption Required	False
Security	Security Signing Required	False
Security	SymmetricEncryptionAlgorithm_AES256-CBC	False
Security	SymmetricSignatureAlgorithm_HMAC-SHA2-256	False

6.6.166 SecurityPolicy – Aes256-Sha256-RsaPss

Table 176 describes the details of the SecurityPolicy – Aes256-Sha256-RsaPss. This security Facet defines a security policy for configurations with a need for high security. It requires a PKI infrastructure. As computing power increases, security policies are expected to expire. NIST provides guidelines for expected expiration dates for individual algorithms. These guidelines provide recommended dates at which the algorithm should be replaced or upgraded to a more secure algorithm. They do not indicate a failure of the algorithm. This security policy has no published end dates as of this time. It is recommended that *Servers* and *Clients* support all security profiles and support configurability of actual exposed and default security policies.

Table 176 – SecurityPolicy – Aes256-Sha256-RsaPss

Group	Conformance Unit / Profile Title	Optional
Security	Aes256-Sha256-RsaPss_Limits	False
Security	AsymmetricEncryptionAlgorithm_RSA-OAEP-SHA2-256	False
Security	AsymmetricSignatureAlgorithm_RSA-PSS-SHA2-256	False
Security	CertificateSignatureAlgorithm_RSA-PKCS15-SHA2-256	False
Security	KeyDerivationAlgorithm_P-SHA2-256	False
Security	Security Certificate Validation	False
Security	Security Encryption Required	False
Security	Security Signing Required	False
Security	SymmetricEncryptionAlgorithm_AES256-CBC	False
Security	SymmetricSignatureAlgorithm_HMAC-SHA2-256	False

6.6.167 User Token – Anonymous Facet

Table 177 describes the details of the User Token – Anonymous Facet. This Facet indicates that anonymous User Tokens are supported.

Table 177 – User Token – Anonymous Facet

Group	Conformance Unit / Profile Title	Optional
Security	Security User Anonymous	False

6.6.168 User Token – User Name Password Server Facet

Table 178 describes the details of the User Token – User Name Password *Server Facet*. This Facet indicates that a user token that is comprised of a username and password is supported. This user token can affect the behaviour of the *ActivateSession Service*.

Table 178 – User Token – User Name Password Server Facet

Group	Conformance Unit / Profile Title	Optional
Security	Security Invalid user token	False
Security	Security User Name Password	False

6.6.169 User Token – X509 Certificate Server Facet

Table 179 describes the details of the User Token – X509 *Certificate Server Facet*. This Facet indicates that the use of an X509 certificates to identify users is supported.

Table 179 – User Token – X509 Certificate Server Facet

Group	Conformance Unit / Profile Title	Optional
Security	Security Invalid user token	False
Security	Security User X509	False

6.6.170 User Token – Issued Token Server Facet

Table 180 describes the details of the User Token – Issued Token *Server Facet*. This Facet indicates that a User Token that is comprised of an issued token is supported.

Table 180 – User Token – Issued Token Server Facet

Group	Conformance Unit / Profile Title	Optional
Security	Security Invalid user token	False
Security	Security User IssuedToken Kerberos	False

6.6.171 User Token – Issued Token Windows Server Facet

Table 181 describes the details of the User Token – Issued Token Windows *Server Facet*. This Facet further refines the User Token – Issued Token to indicate a windows implementation of Kerberos

Table 181 – User Token – Issued Token Windows Server Facet

Group	Conformance Unit / Profile Title	Optional
Profile	User Token – Issued Token <i>Server Facet</i>	False
Security	Security User IssuedToken Kerberos Windows	False

6.6.172 User Token – JWT Server Facet

Table 182 describes the details of the User Token – JWT *Server* Facet. This Facet defines support for JSON Web Tokens (JWT) to identify the user during *Session* setup. A JWT is the Access Token format which OPC UA requires when using OAuth2.

Table 182 – User Token – JWT Server Facet

Group	Conformance Unit / Profile Title	Optional
Security	Azure Identity Provider Authority <i>Profile</i>	True
Security	OAuth2 Authority <i>Profile</i>	True
Security	OPC UA Authority <i>Profile</i>	True
Security	Security Invalid user token	False
Security	Security User JWT IssuedToken	False
Security	Security User JWT Token Policy	False

6.6.173 User Token – User Name Password Client Facet

Table 183 describes the details of the User Token – User Name Password *Client* Facet. This Facet defines the ability to use a user token that is comprised of a username and password.

Table 183 – User Token – User Name Password Client Facet

Group	Conformance Unit / Profile Title	Optional
Security	Security User Name Password <i>Client</i>	False

6.6.174 User Token – X509 Certificate Client Facet

Table 184 describes the details of the User Token – X509 *Certificate Client* Facet. This Facet defines the ability to use an X509 certificates to identify users.

Table 184 – User Token – X509 Certificate Client Facet

Group	Conformance Unit / Profile Title	Optional
Security	Security User X509 <i>Client</i>	False

6.6.175 User Token – Issued Token Client Facet

Table 185 describes the details of the User Token – Issued Token *Client* Facet. This Facet defines the ability to use the User Token – Issued Token (Kerberos) to connect to a *Server*.

Table 185 – User Token – Issued Token Client Facet

Group	Conformance Unit / Profile Title	Optional
Security	Security User IssuedToken Kerberos <i>Client</i>	False

6.6.176 User Token – Issued Token Windows Client Facet

Table 186 describes the details of the User Token – Issued Token Windows *Client* Facet. This Facet defines the ability to use the User Token – Issued Token (Windows implementation of Kerberos) to connect to a *Server*

Table 186 – User Token – Issued Token Windows Client Facet

Group	Conformance Unit / Profile Title	Optional
Security	Security User IssuedToken Kerberos Windows <i>Client</i>	False

6.6.177 User Token – JWT Client Facet

Table 187 describes the details of the User Token – JWT *Client* Facet. This Facet defines the ability to use JSON Web Tokens (JWT) as user identification during *Session* setup. JWTs are used to request an access token from an external Authorization *Service*.

Table 187 – User Token – JWT Client Facet

Group	Conformance Unit / Profile Title	Optional
Security	Azure Identity Provider Authority <i>Profile</i>	True
Security	OAuth2 Authority <i>Profile</i>	True
Security	OPC UA Authority <i>Profile</i>	True
Security	Security User JWT IssuedToken <i>Client</i>	False
Security	Security User JWT Token Policy <i>Client</i>	False

6.6.178 Global Discovery Server Profile

Global Discovery Server Profile has been deprecated as it is superseded with Global Discovery Server 2017 Profile in v1.04.

6.6.179 Global Discovery Server 2017 Profile

Table 188 describes the details of the Global *Discovery* Server 2017 *Profile*. This *Profile* is a FullFeatured *Profile* that covers the necessary *Services* and Information Model of a UA *Server* that acts as a GDS.

This *Profile* supersedes the "Global *Discovery* Server *Profile*".

Table 188 – Global Discovery Server 2017 Profile

Group	Conformance Unit / Profile Title	Optional
<i>Profile</i>	Core 2017 <i>Server</i> Facet	False
<i>Profile</i>	<i>Method</i> <i>Server</i> Facet	False
<i>Profile</i>	Standard DataChange <i>Subscription</i> 2017 <i>Server</i> Facet	False
<i>Profile</i>	UA-TCP UA-SC UA-Binary	False
GDS	GDS Application Directory	False
GDS	GDS LDS-ME Connectivity	False
GDS	GDS Query Applications	False
Security	Security Default ApplicationInstance <i>Certificate</i>	False
Security	Security Policy Required	False
<i>Session Services</i>	<i>Session</i> Minimum 50 Parallel	False

6.6.180 Global Discovery and Certificate Management Server

Global Discovery and Certificate Management Server Profile has been deprecated as it is superseded with Global Discovery and Certificate Mgmt 2017 Server Profile in v1.04.

6.6.181 Global Discovery and Certificate Mgmt 2017 Server

Table 189 describes the details of the *Global Discovery* and *Certificate Mgmt 2017 Server*. This *Profile* is a *FullFeatured Profile* that covers the necessary *Services* and *Information Model* of a *UA Server* that acts as a *GDS* and a global *Certificate Manager*.

This *Profile* supersedes the "Global *Discovery* and *Certificate Management Server*".

Table 189 – Global Discovery and Certificate Mgmt 2017 Server

Group	Conformance Unit / Profile Title	Optional
Profile	Auditing Server Facet	False
Profile	File Access Server Facet	False
Profile	Global Discovery Server 2017 Profile	False
Profile	Standard Event Subscription Server Facet	False
GDS	GDS Certificate Manager Pull Model	False

6.6.182 Global Certificate Management Client Profile

Global Certificate Management Client Profile has been deprecated as it is superseded with Global Certificate Management Client 2017 Profile in v1.04.

6.6.183 Global Certificate Management Client 2017 Profile

Table 190 describes the details of the *Global Certificate Management Client 2017 Profile*. This *Profile* is a *FullFeatured Profile* that uses the *Push Model* for the management of *Certificates* and *Trust Lists*.

This *Profile* supersedes the "Global *Certificate Management Client Profile*".

Table 190 – Global Certificate Management Client 2017 Profile

Group	Conformance Unit / Profile Title	Optional
Profile	Core 2017 Client Facet	False
Profile	Discovery Client Facet	False
Profile	Entry Level Support 2015 Client Facet	False
Profile	File Access Client Facet	False
Profile	Method Client Facet	False
Profile	UA-TCP UA-SC UA-Binary	False
GDS	GDS Certificate Manager Push Model	False
Security	Security Default ApplicationInstance Certificate	False

6.6.184 Global Service Authorization Request Server Facet

Table 191 describes the details of the *Global Service Authorization Request Server Facet*. This Facet defines the capability of a *Server* (like a *GDS*) to provide access tokens to *OPC UA Clients* via an *Authorization Service* as defined in IEC 62541-12.

Table 191 – Global Service Authorization Request Server Facet

Group	Conformance Unit / Profile Title	Optional
GDS	GDS Authorization Service Server	False

6.6.185 Global Service KeyCredential Pull Facet

Table 192 describes the details of the Global Service KeyCredential Pull Facet. This Facet requires providing the *Information Model* for Pull Management as defined in IEC 62541-12. For example KeyCredentials are needed to access an Authorization *Service* or a Broker. OPC UA *Clients* use this Information Model to request and update KeyCredentials they need.

Table 192 – Global Service KeyCredential Pull Facet

Group	Conformance Unit / Profile Title	Optional
GDS	GDS Key Credential <i>Service</i> Pull Model	False

6.6.186 Global Service KeyCredential Push Facet

Table 193 describes the details of the Global Service KeyCredential Push Facet. This Facet requires the use of KeyCredential Push Management functions to set or update credentials in an OPC UA Server. For example KeyCredentials are needed to access an Authorization *Service* or a Broker. This OPC UA Server in turn has to provide the KeyCredentialConfigurationType *Objects* that represent required credentials.

Table 193 – Global Service KeyCredential Push Facet

Group	Conformance Unit / Profile Title	Optional
GDS	GDS Key Credential <i>Service</i> Push Model	False

IECNORM.COM : Click to view the full PDF of IEC 62541-7:2020 RLV

Bibliography

Test Specifications

Compliance Part 8 UA Server: *OPC Test Lab Specification – Part 8 – UA Server*

Compliance Part 9 UA Client: *OPC Test Lab Specification – Part 9 – UA Client*

IECNORM.COM : Click to view the full PDF of IEC 62541-7:2020 RLV

IECNORM.COM : Click to view the full PDF of IEC 62541-7:2020 RLV

INTERNATIONAL STANDARD

NORME INTERNATIONALE



**OPC unified architecture –
Part 7: Profiles**

**Architecture unifiée OPC –
Partie 7: Profils**

IECNORM.COM : Click to view the full PDF of IEC 62541-7:2020 RLV

CONTENTS

FOREWORD	12
1 Scope	15
2 Normative references	15
3 Terms, definitions, and abbreviated terms	16
3.1 Terms and definitions	16
3.2 Abbreviated terms	17
4 Overview	17
4.1 General	17
4.2 ConformanceUnit	18
4.3 Profiles	18
4.4 Profile Categories	19
5 Conformance Units	19
5.1 Overview	19
5.2 Services	20
5.3 Transport and communication related features	30
5.4 Information Model and AddressSpace related features	38
5.5 Miscellaneous	55
6 Profiles	57
6.1 Overview	57
6.2 Profile list	57
6.3 Conventions for Profile definitions	64
6.4 Profile versioning	64
6.5 Applications	64
6.6 Profile tables	66
6.6.1 General	66
6.6.2 Core Server Facet	66
6.6.3 Core 2017 Server Facet	66
6.6.4 Sessionless Server Facet	67
6.6.5 Reverse Connect Server Facet	67
6.6.6 Base Server Behaviour Facet	68
6.6.7 Request State Change Server Facet	68
6.6.8 Subnet Discovery Server Facet	68
6.6.9 Global Certificate Management Server Facet	68
6.6.10 Authorization Service Server Facet	69
6.6.11 KeyCredential Service Server Facet	69
6.6.12 Attribute WriteMask Server Facet	69
6.6.13 File Access Server Facet	69
6.6.14 Documentation Server Facet	70
6.6.15 Embedded DataChange Subscription Server Facet	70
6.6.16 Standard DataChange Subscription Server Facet	70
6.6.17 Standard DataChange Subscription 2017 Server Facet	71
6.6.18 Enhanced DataChange Subscription Server Facet	71
6.6.19 Enhanced DataChange Subscription 2017 Server Facet	71
6.6.20 Durable Subscription Server Facet	71
6.6.21 Data Access Server Facet	72
6.6.22 ComplexType Server Facet	72

6.6.23	ComplexType 2017 Server Facet	72
6.6.24	Standard Event Subscription Server Facet	73
6.6.25	Address Space Notifier Server Facet	74
6.6.26	A & C Base Condition Server Facet	74
6.6.27	A & C Refresh2 Server Facet	74
6.6.28	A & C Address Space Instance Server Facet	74
6.6.29	A & C Enable Server Facet	75
6.6.30	A & C AlarmMetrics Server Facet	75
6.6.31	A & C Alarm Server Facet	75
6.6.32	A & C Acknowledgeable Alarm Server Facet	76
6.6.33	A & C Exclusive Alarming Server Facet	76
6.6.34	A & C Non-Exclusive Alarming Server Facet	77
6.6.35	A & C Previous Instances Server Facet	77
6.6.36	A & C Dialog Server Facet	77
6.6.37	A & C CertificateExpiration Server Facet	78
6.6.38	A & E Wrapper Facet	78
6.6.39	Method Server Facet	79
6.6.40	Auditing Server Facet	79
6.6.41	Node Management Server Facet	80
6.6.42	User Role Base Server Facet	80
6.6.43	User Role Management Server Facet	80
6.6.44	State Machine Server Facet	81
6.6.45	Client Redundancy Server Facet	81
6.6.46	Redundancy Transparent Server Facet	81
6.6.47	Redundancy Visible Server Facet	82
6.6.48	Historical Raw Data Server Facet	82
6.6.49	Historical Aggregate Server Facet	82
6.6.50	Historical Data AtTime Server Facet	83
6.6.51	Historical Access Modified Data Server Facet	84
6.6.52	Historical Annotation Server Facet	84
6.6.53	Historical Data Insert Server Facet	84
6.6.54	Historical Data Update Server Facet	84
6.6.55	Historical Data Replace Server Facet	85
6.6.56	Historical Data Delete Server Facet	85
6.6.57	Historical Access Structured Data Server Facet	85
6.6.58	Base Historical Event Server Facet	85
6.6.59	Historical Event Update Server Facet	86
6.6.60	Historical Event Replace Server Facet	86
6.6.61	Historical Event Insert Server Facet	86
6.6.62	Historical Event Delete Server Facet	86
6.6.63	Aggregate Subscription Server Facet	87
6.6.64	Nano Embedded Device Server Profile	88
6.6.65	Nano Embedded Device 2017 Server Profile	88
6.6.66	Micro Embedded Device Server Profile	88
6.6.67	Micro Embedded Device 2017 Server Profile	88
6.6.68	Embedded UA Server Profile	88
6.6.69	Embedded 2017 UA Server Profile	89
6.6.70	Standard UA Server Profile	89
6.6.71	Standard 2017 UA Server Profile	89

6.6.72	Core Client Facet.....	90
6.6.73	Core 2017 Client Facet.....	90
6.6.74	Sessionless Client Facet	90
6.6.75	Reverse Connect Client Facet	90
6.6.76	Base Client Behaviour Facet.....	91
6.6.77	Discovery Client Facet.....	91
6.6.78	Subnet Discovery Client Facet.....	91
6.6.79	Global Discovery Client Facet.....	92
6.6.80	Global Certificate Management Client Facet	92
6.6.81	KeyCredential Service Client Facet.....	92
6.6.82	Access Token Request Client Facet	92
6.6.83	AddressSpace Lookup Client Facet	93
6.6.84	Request State Change Client Facet	93
6.6.85	File Access Client Facet	93
6.6.86	Entry Level Support 2015 Client Facet.....	94
6.6.87	Multi-Server Client Connection Facet.....	94
6.6.88	Documentation – Client	94
6.6.89	Attribute Read Client Facet.....	94
6.6.90	Attribute Write Client Facet.....	95
6.6.91	DataChange Subscriber Client Facet	95
6.6.92	Durable Subscription Client Facet.....	96
6.6.93	DataAccess Client Facet.....	96
6.6.94	Event Subscriber Client Facet.....	97
6.6.95	Base Event Processing Client Facet	97
6.6.96	Notifier and Source Hierarchy Client Facet	98
6.6.97	A & C Base Condition Client Facet	98
6.6.98	A & C Refresh2 Client Facet.....	98
6.6.99	A & C Address Space Instance Client Facet	99
6.6.100	A & C Enable Client Facet	99
6.6.101	A & C Alarm Metrics Client Facet.....	99
6.6.102	A & C Alarm Client Facet.....	99
6.6.103	A & C Exclusive Alarming Client Facet.....	100
6.6.104	A & C Non-Exclusive Alarming Client Facet	100
6.6.105	A & C Previous Instances Client Facet.....	101
6.6.106	A & C Dialog Client Facet	101
6.6.107	A & C CertificateExpiration Client Facet.....	101
6.6.108	A & E Proxy Facet	102
6.6.109	Method Client Facet.....	103
6.6.110	Auditing Client Facet	103
6.6.111	Node Management Client Facet.....	103
6.6.112	Advanced Type Programming Client Facet	103
6.6.113	User Role Management Client Facet.....	104
6.6.114	State Machine Client Facet.....	104
6.6.115	Diagnostic Client Facet.....	104
6.6.116	Redundant Client Facet	105
6.6.117	Redundancy Switch Client Facet	105
6.6.118	Historical Access Client Facet	105
6.6.119	Historical Data AtTime Client Facet	105
6.6.120	Historical Aggregate Client Facet.....	105

6.6.121	Historical Annotation Client Facet	107
6.6.122	Historical Access Modified Data Client Facet	107
6.6.123	Historical Data Insert Client Facet	107
6.6.124	Historical Data Update Client Facet	107
6.6.125	Historical Data Replace Client Facet	107
6.6.126	Historical Data Delete Client Facet	108
6.6.127	Historical Access Client Server Timestamp Facet	108
6.6.128	Historical Structured Data Access Client Facet	108
6.6.129	Historical Structured Data AtTime Client Facet	108
6.6.130	Historical Structured Data Modified Client Facet	109
6.6.131	Historical Structured Data Insert Client Facet	109
6.6.132	Historical Structured Data Update Client Facet	109
6.6.133	Historical Structured Data Replace Client Facet	109
6.6.134	Historical Structured Data Delete Client Facet	109
6.6.135	Historical Events Client Facet	110
6.6.136	Historical Event Insert Client Facet	110
6.6.137	Historical Event Update Client Facet	110
6.6.138	Historical Event Replace Client Facet	110
6.6.139	Historical Event Delete Client Facet	111
6.6.140	Aggregate Subscriber Client Facet	111
6.6.141	Standard UA Client Profile	112
6.6.142	Standard UA Client 2017 Profile	112
6.6.143	UA-TCP UA-SC UA-Binary	113
6.6.144	HTTPS UA-Binary	113
6.6.145	HTTPS UA-XML	114
6.6.146	HTTPS UA-JSON	114
6.6.147	WSS UA-SC UA-Binary	114
6.6.148	WSS UA-JSON	114
6.6.149	Security User Access Control Full	115
6.6.150	Security User Access Control Base	115
6.6.151	Security Time Synchronization	115
6.6.152	Best Practice – Audit Events	116
6.6.153	Best Practice – Alarm Handling	116
6.6.154	Best Practice – Random Numbers	116
6.6.155	Best Practice – Timeouts	116
6.6.156	Best Practice – Administrative Access	116
6.6.157	Best Practice – Strict Message Handling	117
6.6.158	Best Practice – Audit Events Client	117
6.6.159	TransportSecurity – TLS 1.2	117
6.6.160	TransportSecurity – TLS 1.2 with PFS	117
6.6.161	SecurityPolicy – None	118
6.6.162	SecurityPolicy – Basic128Rsa15	118
6.6.163	SecurityPolicy – Basic256	118
6.6.164	SecurityPolicy [A] – Aes128-Sha256-RsaOaep	118
6.6.165	SecurityPolicy [B] – Basic256Sha256	119
6.6.166	SecurityPolicy – Aes256-Sha256-RsaPss	119
6.6.167	User Token – Anonymous Facet	120
6.6.168	User Token – User Name Password Server Facet	120
6.6.169	User Token – X509 Certificate Server Facet	120

6.6.170	User Token – Issued Token Server Facet	121
6.6.171	User Token – Issued Token Windows Server Facet	121
6.6.172	User Token – JWT Server Facet	121
6.6.173	User Token – User Name Password Client Facet	121
6.6.174	User Token – X509 Certificate Client Facet	122
6.6.175	User Token – Issued Token Client Facet	122
6.6.176	User Token – Issued Token Windows Client Facet	122
6.6.177	User Token – JWT Client Facet	122
6.6.178	Global Discovery Server Profile	122
6.6.179	Global Discovery Server 2017 Profile	123
6.6.180	Global Discovery and Certificate Management Server	123
6.6.181	Global Discovery and Certificate Mgmt 2017 Server	123
6.6.182	Global Certificate Management Client Profile	123
6.6.183	Global Certificate Management Client 2017 Profile	123
6.6.184	Global Service Authorization Request Server Facet	124
6.6.185	Global Service KeyCredential Pull Facet	124
6.6.186	Global Service KeyCredential Push Facet	124
	Bibliography	125
	Figure 1 – Profile – ConformanceUnit – TestCases	18
	Figure 2 – HMI Client sample	64
	Figure 3 – Embedded Server sample	65
	Figure 4 – Standard UA Server sample	65
	Table 1 – Profile Categories	19
	Table 2 – Conformance Groups	20
	Table 3 – Discovery Services	21
	Table 4 – Session Services	22
	Table 5 – Node Management Services	23
	Table 6 – View Services	24
	Table 7 – Attribute Services	25
	Table 8 – Method Services	26
	Table 9 – Monitored Item Services	27
	Table 10 – Subscription Services	29
	Table 11 – Security	31
	Table 12 – Protocol and Encoding	38
	Table 13 – Base Information	39
	Table 14 – Address Space Model	41
	Table 15 – Data Access	42
	Table 16 – Alarms and Conditions	43
	Table 17 – Historical Access	46
	Table 18 – Aggregates	49
	Table 19 – Auditing	54
	Table 20 – Redundancy	54
	Table 21 – Global Discovery Server	55

Table 22 – Miscellaneous	56
Table 23 – Profile list	58
Table 24 – Core 2017 Server Facet	67
Table 25 – Sessionless Server Facet	67
Table 26 – Reverse Connect Server Facet	68
Table 27 – Base Server Behaviour Facet	68
Table 28 – Request State Change Server Facet	68
Table 29 – Subnet Discovery Server Facet	68
Table 30 – Global Certificate Management Server Facet	69
Table 31 – Authorization Service Server Facet	69
Table 32 – KeyCredential Service Server Facet	69
Table 33 – Attribute WriteMask Server Facet	69
Table 34 – File Access Server Facet	70
Table 35 – Documentation Server Facet	70
Table 36 – Embedded DataChange Subscription Server Facet	70
Table 37 – Standard DataChange Subscription 2017 Server Facet	71
Table 38 – Enhanced DataChange Subscription 2017 Server Facet	71
Table 39 – Durable Subscription Server Facet	72
Table 40 – Data Access Server Facet	72
Table 41 – ComplexType 2017 Server Facet	73
Table 42 – Standard Event Subscription Server Facet	73
Table 43 – Address Space Notifier Server Facet	74
Table 44 – A & C Base Condition Server Facet	74
Table 45 – A & C Refresh2 Server Facet	74
Table 46 – A & C Address Space Instance Server Facet	75
Table 47 – A & C Enable Server Facet	75
Table 48 – A & C AlarmMetrics Server Facet	75
Table 49 – A & C Alarm Server Facet	76
Table 50 – A & C Acknowledgeable Alarm Server Facet	76
Table 51 – A & C Exclusive Alarming Server Facet	77
Table 52 – A & C Non-Exclusive Alarming Server Facet	77
Table 53 – A & C Previous Instances Server Facet	77
Table 54 – A & C Dialog Server Facet	78
Table 55 – A & C CertificateExpiration Server Facet	78
Table 56 – A & E Wrapper Facet	79
Table 57 – Method Server Facet	79
Table 58 – Auditing Server Facet	80
Table 59 – Node Management Server Facet	80
Table 60 – User Role Base Server Facet	80
Table 61 – User Role Management Server Facet	81
Table 62 – State Machine Server Facet	81
Table 63 – Client Redundancy Server Facet	81
Table 64 – Redundancy Transparent Server Facet	81

Table 65 – Redundancy Visible Server Facet.....	82
Table 66 – Historical Raw Data Server Facet.....	82
Table 67 – Historical Aggregate Server Facet.....	83
Table 68 – Historical Data AtTime Server Facet.....	84
Table 69 – Historical Access Modified Data Server Facet	84
Table 70 – Historical Annotation Server Facet	84
Table 71 – Historical Data Insert Server Facet.....	84
Table 72 – Historical Data Update Server Facet.....	85
Table 73 – Historical Data Replace Server Facet.....	85
Table 74 – Historical Data Delete Server Facet.....	85
Table 75 – Historical Access Structured Data Server Facet.....	85
Table 76 – Base Historical Event Server Facet	86
Table 77 – Historical Event Update Server Facet.....	86
Table 78 – Historical Event Replace Server Facet	86
Table 79 – Historical Event Insert Server Facet	86
Table 80 – Historical Event Delete Server Facet	86
Table 81 – Aggregate Subscription Server Facet	87
Table 82 – Nano Embedded Device 2017 Server Profile.....	88
Table 83 – Micro Embedded Device 2017 Server Profile.....	88
Table 84 – Embedded 2017 UA Server Profile	89
Table 85 – Standard 2017 UA Server Profile.....	89
Table 86 – Core 2017 Client Facet.....	90
Table 87 – Sessionless Client Facet	90
Table 88 – Reverse Connect Client Facet.....	91
Table 89 – Base Client Behaviour Facet	91
Table 90 – Discovery Client Facet.....	91
Table 91 – Subnet Discovery Client Facet.....	92
Table 92 – Global Discovery Client Facet	92
Table 93 – Global Certificate Management Client Facet.....	92
Table 94 – KeyCredential Service Client Facet	92
Table 95 – Access Token Request Client Facet	93
Table 96 – AddressSpace Lookup Client Facet.....	93
Table 97 – Request State Change Client Facet.....	93
Table 98 – File Access Client Facet	93
Table 99 – Entry Level Support 2015 Client Facet	94
Table 100 – Multi-Server Client Connection Facet	94
Table 101 – Documentation – Client	94
Table 102 – Attribute Read Client Facet.....	95
Table 103 – Attribute Write Client Facet.....	95
Table 104 – DataChange Subscriber Client Facet.....	96
Table 105 – Durable Subscription Client Facet.....	96
Table 106 – DataAccess Client Facet	97
Table 107 – Event Subscriber Client Facet	97

Table 108 – Base Event Processing Client Facet	98
Table 109 – Notifier and Source Hierarchy Client Facet	98
Table 110 – A & C Base Condition Client Facet	98
Table 111 – A & C Refresh2 Client Facet	99
Table 112 – A & C Address Space Instance Client Facet	99
Table 113 – A & C Enable Client Facet	99
Table 114 – A & C AlarmMetrics Client Facet	99
Table 115 – A & C Alarm Client Facet	100
Table 116 – A & C Exclusive Alarming Client Facet	100
Table 117 – A & C Non-Exclusive Alarming Client Facet	101
Table 118 – A & C Previous Instances Client Facet	101
Table 119 – A & C Dialog Client Facet	101
Table 120 – A & C CertificateExpiration Client Facet	101
Table 121 – A & E Proxy Facet	102
Table 122 – Method Client Facet	103
Table 123 – Auditing Client Facet	103
Table 124 – Node Management Client Facet	103
Table 125 – Advanced Type Programming Client Facet	104
Table 126 – User Role Management Client Facet	104
Table 127 – State Machine Client Facet	104
Table 128 – Diagnostic Client Facet	104
Table 129 – Redundant Client Facet	105
Table 130 – Redundancy Switch Client Facet	105
Table 131 – Historical Access Client Facet	105
Table 132 – Historical Data AtTime Client Facet	105
Table 133 – Historical Aggregate Client Facet	106
Table 134 – Historical Annotation Client Facet	107
Table 135 – Historical Access Modified Data Client Facet	107
Table 136 – Historical Data Insert Client Facet	107
Table 137 – Historical Data Update Client Facet	107
Table 138 – Historical Data Replace Client Facet	108
Table 139 – Historical Data Delete Client Facet	108
Table 140 – Historical Access Client Server Timestamp Facet	108
Table 141 – Historical Structured Data Access Client Facet	108
Table 142 – Historical Structured Data AtTime Client Facet	108
Table 143 – Historical Structured Data Modified Client Facet	109
Table 144 – Historical Structured Data Insert Client Facet	109
Table 145 – Historical Structured Data Update Client Facet	109
Table 146 – Historical Structured Data Replace Client Facet	109
Table 147 – Historical Structured Data Delete Client Facet	110
Table 148 – Historical Events Client Facet	110
Table 149 – Historical Event Insert Client Facet	110
Table 150 – Historical Event Update Client Facet	110

Table 151 – Historical Event Replace Client Facet	110
Table 152 – Historical Event Delete Client Facet	111
Table 153 – Aggregate Subscriber Client Facet	111
Table 154 – Standard UA Client 2017 Profile	113
Table 155 – UA-TCP UA-SC UA-Binary	113
Table 156 – HTTPS UA-Binary.....	113
Table 157 – HTTPS UA-XML	114
Table 158 – HTTPS UA-JSON	114
Table 159 – WSS UA-SC UA-Binary	114
Table 160 – WSS UA-JSON.....	115
Table 161 – Security User Access Control Full	115
Table 162 – Security User Access Control Base	115
Table 163 – Security Time Synchronization	115
Table 164 – Best Practice – Audit Events	116
Table 165 – Best Practice – Alarm Handling	116
Table 166 – Best Practice – Random Numbers	116
Table 167 – Best Practice – Timeouts.....	116
Table 168 – Best Practice – Administrative Access	117
Table 169 – Best Practice – Strict Message Handling.....	117
Table 170 – Best Practice – Audit Events Client	117
Table 171 – TransportSecurity – TLS 1.2.....	117
Table 172 – TransportSecurity – TLS 1.2 with PFS	118
Table 173 – SecurityPolicy – None	118
Table 174 – SecurityPolicy [A] – Aes128-Sha256-RsaOaep	119
Table 175 – SecurityPolicy [B] – Basic256Sha256	119
Table 176 – SecurityPolicy – Aes256-Sha256-RsaPss	120
Table 177 – User Token – Anonymous Facet.....	120
Table 178 – User Token – User Name Password Server Facet	120
Table 179 – User Token – X509 Certificate Server Facet.....	120
Table 180 – User Token – Issued Token Server Facet.....	121
Table 181 – User Token – Issued Token Windows Server Facet	121
Table 182 – User Token – JWT Server Facet.....	121
Table 183 – User Token – User Name Password Client Facet.....	121
Table 184 – User Token – X509 Certificate Client Facet	122
Table 185 – User Token – Issued Token Client Facet	122
Table 186 – User Token – Issued Token Windows Client Facet	122
Table 187 – User Token – JWT Client Facet.....	122
Table 188 – Global Discovery Server 2017 Profile	123
Table 189 – Global Discovery and Certificate Mgmt 2017 Server	123
Table 190 – Global Certificate Management Client 2017 Profile	124
Table 191 – Global Service Authorization Request Server Facet.....	124

Table 192 – Global Service KeyCredential Pull Facet	124
Table 193 – Global Service KeyCredential Push Facet.....	124

IECNORM.COM : Click to view the full PDF of IEC 62541-7:2020 RLV

INTERNATIONAL ELECTROTECHNICAL COMMISSION

OPC UNIFIED ARCHITECTURE –

Part 7: Profiles

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 62541-7 has been prepared by subcommittee 65E: Devices and integration in enterprise systems, of IEC technical committee 65: Industrial-process measurement, control and automation.

This third edition cancels and replaces the second edition published in 2015. This edition constitutes a technical revision.

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

a) new functional Profiles:

- profiles for global discovery and global certificate management;
- profiles for global KeyCredential management and global access token management;
- facet for durable subscriptions;
- standard UA Client Profile;

- profiles for administration of user roles and permissions.
- b) new transport Profiles:
- HTTPS with JSON encoding;
 - secure WebSockets (WSS) with binary or JSON encoding;
 - reverse connectivity.
- c) new security Profiles:
- transportSecurity – TLS 1.2 with PFS (with perfect forward secrecy);
 - securityPolicy [A] – Aes128-Sha256-RsaOaep (replaces Base128Rsa15);
 - securityPolicy – Aes256-Sha256-RsaPss adds perfect forward secrecy for UA TCP);
 - user Token JWT (Jason Web Token).
- d) deprecated Security Profiles (due to broken algorithms):
- securityPolicy – Basic128Rsa15 (broken algorithm Sha1);
 - securityPolicy – Basic256 (broken algorithm Sha1);
 - transportSecurity – TLS 1.0 (broken algorithm RC4);
 - transportSecurity – TLS 1.1 (broken algorithm RC4).
- e) deprecated Transport (missing support on most platforms).
- SOAP/HTTP with WS-SecureConversation (all encodings).

The text of this International Standard is based on the following documents:

FDIS	Report on voting
65E/707/FDIS	65E/725/RVD

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

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

Throughout this document and the other parts of the IEC 62541 series, certain document conventions are used:

Italics are used to denote a defined term or definition that appears in the "Terms and definition" clause in one of the parts of the IEC 62541 series.

Italics are also used to denote the name of a service input or output parameter or the name of a structure or element of a structure that are usually defined in tables.

The *italicized terms and names* are, with a few exceptions, written in camel-case (the practice of writing compound words or phrases in which the elements are joined without spaces, with each element's initial letter capitalized within the compound). For example the defined term is *AddressSpace* instead of Address Space. This makes it easier to understand that there is a single definition for *AddressSpace*, not separate definitions for Address and Space.

A list of all parts of the IEC 62541 series, published under the general title *OPC Unified Architecture*, can be found on the IEC website.

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

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

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

IECNORM.COM : Click to view the full PDF of IEC 62541-7:2020

OPC UNIFIED ARCHITECTURE –

Part 7: Profiles

1 Scope

This part of IEC 62541 defines the OPC Unified Architecture (OPC UA) *Profiles*. The *Profiles* in this document are used to segregate features with regard to testing of OPC UA products and the nature of the testing (tool based or lab based). This includes the testing performed by the OPC Foundation provided OPC UA CTT (a self-test tool) and by the OPC Foundation provided Independent certification test labs. This could equally as well refer to test tools provided by another organization or a test lab provided by another organization. What is important is the concept of automated tool-based testing versus lab-based testing. The scope of this standard includes defining functionality that can only be tested in a lab and defining the grouping of functionality that is to be used when testing OPC UA products either in a lab or using automated tools. The definition of actual *TestCases* is not within the scope of this document, but the general categories of *TestCases* are within the scope of this document.

Most OPC UA applications will conform to several, but not all, of the *Profiles*.

2 Normative references

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

IEC TR 62541-1, *OPC Unified Architecture – Part 1: Overview and Concepts*

IEC TR 62541-2, *OPC Unified Architecture – Part 2: Security Model*

IEC 62541-3, *OPC Unified Architecture – Part 3: Address Space Model*

IEC 62541-4, *OPC Unified Architecture – Part 4: Services*

IEC 62541-5, *OPC Unified Architecture – Part 5: Information Model*

IEC 62541-6, *OPC Unified Architecture – Part 6: Mappings*

IEC 62541-8, *OPC Unified Architecture – Part 8: Data Access*

IEC 62541-9, *OPC Unified Architecture – Part 9: Alarms and Conditions*

IEC 62541-11, *OPC Unified Architecture – Part 11: Historical Access*

IEC 62541-12, *OPC Unified Architecture – Part 12: Discovery and Global Services*

IEC 62541-13, *OPC Unified Architecture – Part 13: Aggregates*

Compliance Part 8 UA Server: OPC Test Lab Specification: Part 8 – UA Server
<http://www.opcfoundation.org/Test/Part8/>

Compliance Part 9 UA Client: OPC Test Lab Specification: Part 9 – UA Client
<http://www.opcfoundation.org/Test/Part9/>

3 Terms, definitions, and abbreviated terms

3.1 Terms and definitions

For the purposes of this document, the terms and definitions given in IEC TR 62541-1, IEC TR 62541-2, IEC 62541-3, IEC 62541-4, IEC 62541-6, and IEC 62541-8 and the following apply.

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

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

NOTE An overview of the terms defined in this document and their interaction can be viewed in Figure 1.

3.1.1

application

software program that executes or implements some aspect of OPC UA

Note 1 to entry: The application could run on any machine and perform any function. The application could be software or it could be a hardware application, the only requirement is that it implements OPC UA.

3.1.2

ConformanceUnit

specific set of OPC UA features that can be tested as a single entity

Note 1 to entry: A ConformanceUnit can cover a group of services, portions of services or information models.

3.1.3

ConformanceGroup

group of ConformanceUnits that is given a name

Note 1 to entry: This grouping is only to assist in organizing ConformanceUnits. Typical ConformanceGroups include groups for each of the service sets in OPC UA and each of the Information Model standards.

3.1.4

Facet

Profile dedicated to a specific feature that a Server or Client may require

Note 1 to entry: Facets are typically combined to form higher-level Profiles. The use of the term Facet in the title of a Profile indicates that the given Profile is not a standalone Profile.

3.1.5

FullFeatured Profile

Profile that defines all features necessary to build a functional OPC UA Application

Note 1 to entry: A FullFeatured Profile in particular adds definitions of the transport and security requirements.

3.1.6

ProfileCategory

category which arranges Profiles into application classes, such as Server or Client

Note 1 to entry: These categories help determine the type of Application that a given Profile would be used for. For additional details see 4.4.

3.1.7

TestCase

technical description of a set of steps required to test a particular function or information model

Note 1 to entry: TestCases provide sufficient details to allow a developer to implement them in code. TestCases also provide a detailed summary of the expected result(s) from the execution of the implemented code and any precondition(s) that shall be established before the TestCase can be executed.

3.1.8

TestLab

facility that is designated to provide testing services

Note 1 to entry: These services include but are not limited to personal that directly perform testing, automated testing and a formal repeatable process. The OPC Foundation has provided detailed standard describing OPC UA TestLabs and the testing they are to provided (see Compliance Part 8 UA Server, Compliance Part 9 UA Client).

3.2 Abbreviated terms

DA	data access
HA	historical access
HMI	human machine interface
NIST	National Institute of Standard and Technology
PKI	Public Key Infrastructure
RSA	Rivest-Shamir-Adleman
UA	Unified Architecture

4 Overview

4.1 General

The IEC 62541 series, which is the OPC Unified architecture standard, describes a number of *Services* and a variety of information models. These *Services* and information models can be referred to as features of a *Server* or *Client*. *Servers* and *Clients* need to be able to describe which features they support and wish to have certified. This document provides a grouping of these features. The individual features are grouped into *ConformanceUnits* which are further grouped into *Profiles*. Figure 1 provides an overview of the interactions between *Profiles*, *ConformanceUnits* and *TestCases*. The large arrows indicate the components that are used to construct the parent. For example a *Profile* is constructed from *Profiles* and *ConformanceUnits*. The figure also illustrates a feature of the OPC UA Compliance Test Tool (CTT), in that it will test if a requested *Profile* passes all *ConformanceUnits*. It will also test all other *ConformanceUnits* and report any other *Profiles* that pass conformance testing. The individual *TestCases* are defined in separate documents (see Compliance Part 8 UA Server and Compliance Part 9 UA Client). The *TestCases* are related back to the appropriate *ConformanceUnits* defined in this document. This relationship is also displayed by the OPC UA Compliance Test Tool.

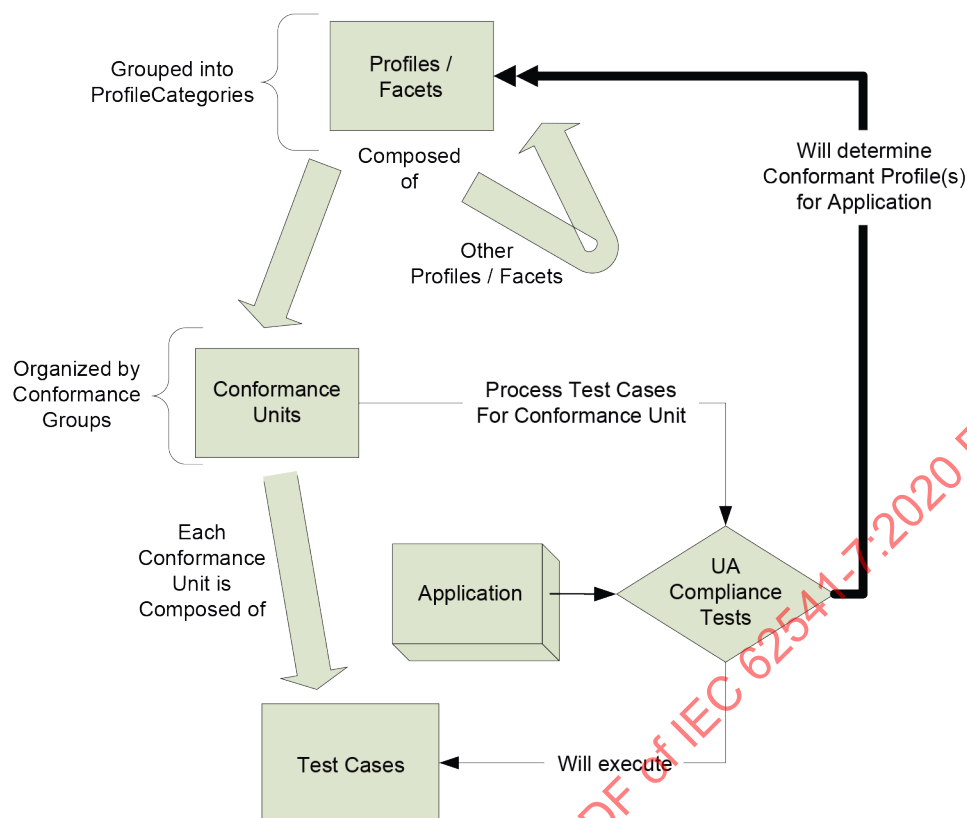


Figure 1 – Profile – ConformanceUnit – TestCases

4.2 ConformanceUnit

Each *ConformanceUnit* represents a specific set of features (e.g. a group of services, portions of services or information models) that can be tested as a single entity. *ConformanceUnits* are the building blocks of a *Profile*. Each *ConformanceUnit* can also be used as a test category. For each *ConformanceUnit*, there would be a number of *TestCases* that test the functionality described by the *ConformanceUnit*. The description of a *ConformanceUnit* is intended to provide enough information to illustrate the required functionality, but in many cases to obtain a complete understanding of the *ConformanceUnit* the reader may be required to also examine the appropriate part of OPC UA. Additional Information regarding testing of a *ConformanceUnit* are provided in the Compliance Part 8 UA Server or Compliance Part 9 UA Client test standards.

The same features do not appear in more than one *ConformanceUnit*.

4.3 Profiles

A *Profile* is a named aggregation of *ConformanceUnits* and other *Profiles*. To support a *Profile*, an application has to support the *ConformanceUnits* and all aggregated *Profiles*. The definition of *Profiles* is an ongoing activity, in that it is expected that new *Profiles* will be added in the future.

An OPC UA Application will typically support multiple *Profiles*.

Multiple *Profiles* may include the same *ConformanceUnit*.

Testing of a *Profile* consists of testing the individual *ConformanceUnits* that comprise the *Profile*.

Profiles are named based on naming conventions (see 6.3 for details).

4.4 Profile Categories

Profiles are grouped into categories to help vendors and end users understand the applicability of a *Profile*. A *Profile* can be assigned to more than one category.

Table 1 contains the list of currently defined *ProfileCategories*.

Table 1 – Profile Categories

Category	Description
Client	Profiles of this category specify functions of an OPC UA Client.
Global Directory Service	Profiles of this category specify functions for global discovery and certificate management.
Security	Profiles of this category specify security related functions. Security policies are part of this category. The URI of security policies has to be part of an Endpoint Description returned from the GetEndpoints service. Profiles of this category apply to Clients and Servers.
Server	Profiles of this category specify functions of an OPC UA Server. The URI of such Profiles can be exposed in the Server capabilities.
Transport	Profiles of this category specify specific protocol mappings. The URI of such Profiles has to be part of an Endpoint Description. These Profiles apply to Clients and Servers.

5 Conformance Units

5.1 Overview

A *ConformanceUnit* represents an individually testable entity. For improved clarity, the large list of *ConformanceUnits* is arranged into named *ConformanceGroups*. These groups reflect the *Service Sets* in IEC 62541-4, and the OPC UA information models. Table 2 lists the *ConformanceGroups*. These groups and the *ConformanceUnits* that they describe are detailed in the subclauses of Clause 5, starting with 5.2. *ConformanceGroups* have no impact on testing; they are used only for organizational reasons, i.e. to simplify the readability of this document.

Table 2 – Conformance Groups

Group	Description
Address Space Model	Defines <i>ConformanceUnits</i> for various features of the OPC UA <i>AddressSpace</i> .
Aggregates	All <i>ConformanceUnits</i> that are related to Aggregates, including individual <i>ConformanceUnits</i> for each supported Aggregate as described in IEC 62541-13.
Alarms and Conditions	All <i>ConformanceUnits</i> that are associated with the OPC UA Information Model for <i>Conditions</i> , acknowledgeable <i>Conditions</i> , confirmations and <i>Alarms</i> as specified in IEC 62541-9.
Attribute Services	Includes <i>ConformanceUnits</i> to read or write current or historical <i>Attribute</i> values.
Auditing	User level security includes support for security audit trails, with traceability between <i>Client</i> and <i>Server</i> audit logs.
Base Information	All information elements as defined in IEC 62541-5.
Data Access	<i>ConformanceUnits</i> specific to <i>Clients</i> and <i>Servers</i> that deal with the representation and use of automation data as specified in IEC 62541-8.
Discovery Services	<i>ConformanceUnits</i> which focus on <i>Server</i> Endpoint <i>Discovery</i> .
GDS	<i>ConformanceUnits</i> for a GDS. Includes units for global discovery and global certificate management.
Historical Access	Access to archived data of <i>Node Attribute</i> values or Events.
Method Services	Methods represent the function calls of <i>Objects</i> . Methods are invoked and return only after completion (successful or unsuccessful).
Miscellaneous	This group contains <i>ConformanceUnits</i> that cover miscellaneous subjects, such as recommended behaviours, documentation etc. These <i>ConformanceUnits</i> typically do not fit into any of the other groups.
Monitored Item Services	<i>Clients</i> define <i>MonitoredItems</i> to subscribe to data and Events. Each <i>MonitoredItem</i> identifies the item to be monitored and the <i>Subscription</i> to use to send <i>Notifications</i> .
Node Management Services	Bundles <i>ConformanceUnits</i> for all <i>Services</i> to add and delete OPC UA <i>AddressSpace Nodes</i> and <i>References</i> .
Protocol and Encoding	Covers all transport and encoding combinations that are specified in IEC 62541-6.
Redundancy	The design of OPC UA ensures that vendors can create redundant <i>Clients</i> and redundant <i>Servers</i> in a consistent manner. Redundancy may be used for high availability, fault tolerance and load balancing.
Security	Security related <i>ConformanceUnits</i> that can be profiled this includes all aspects of security.
Session Services	An (OPC UA) <i>Session</i> is an application layer connection.
Subscription Services	<i>Subscriptions</i> are used to report <i>Notifications</i> to the <i>Client</i> .
View Services	<i>Clients</i> use the View Service Set to navigate through the OPC UA <i>AddressSpace</i> or through a View (a subset) of the OPC UA <i>AddressSpace</i> .

5.2 Services

Table 3 to Table 10 describe *ConformanceUnits* for the *Services* specified in IEC 62541-4. The tables correlate with the *Service Sets*.

A single *ConformanceUnit* can reference several *Services* (e.g. CreateSession, ActivateSession and CloseSession), but can also refer to individual aspects of *Services* (e.g. the use of ActivateSession to impersonate a new user).

Each table includes a listing of the *Profile Category* to which a *ConformanceUnit* belongs, the title and description of the *ConformanceUnit*. In some cases, a *ConformanceUnit* will be derived from another *ConformanceUnit*. This parent unit will then be specified in the description of each derived unit. In such cases, the derived units inherit all of the tests of its parent plus one or more additional TestCases. These TestCases can only further restrict the existing TestCases. An example would be one in which the number of connections is tested, where the TestCase of the parent required at least one connection and the derived *ConformanceUnit* would require a TestCase for at least five connections.

The *Discovery Service Set* is composed of multiple *ConformanceUnits* (see Table 3). All *Servers* provide some aspects of this functionality; see *Profiles* categorized as *Server Profiles* for details. *Clients* may support some aspects of this functionality; see *Profiles* categorized as *Client Profiles* for details.

Table 3 – Discovery Services

Category	Title	Description
Server	Discovery Get Endpoints	Support the GetEndpoints <i>Service</i> to obtain all Endpoints of the <i>Server</i> . This includes filtering based on <i>Profiles</i> .
Server	Discovery Get Endpoints SessionLess	Support at least one endpoint for issuing SessionLess <i>Services</i> . Support obtaining such endpoints by accepting the Transport URI as a filter to the GetEndpoints <i>Service</i> with the query string "SL" appended to the Transport URI. E.g. "http://opcfoundation.org/UA-Profile/Transport/https-uajson?SL"
Server	Discovery Find Servers Self	Support the FindServers <i>Service</i> only for itself.
Server	Discovery Register	Call the RegisterServer <i>Service</i> to register itself (OPC UA <i>Server</i>) with an external <i>Discovery Service</i> via a secure channel with a SecurityMode other than NONE.
Server	Discovery Register2	Call the RegisterServer2 <i>Service</i> to register with an external <i>Discovery Service</i> via a Secure Channel with a SecurityMode other than "None". This includes passing a list of short capability identifiers. The identifiers and their use are specified in IEC 62541-12.
Server	Discovery Server Announcement using mDNS	Provide mDNS functionality to announce a <i>Server</i> with its capabilities. The capability identifiers and the use of mDNS records for the purpose of OPC UA <i>Discovery</i> is specified in IEC 62541-12. Note that this functionality is only required for <i>Servers</i> that do not register with an LDS. The capability identifiers and their use in mDNS records are specified in IEC 62541-12.
Server	Discovery Configuration	Allow configuration of the <i>Discovery Server</i> URL where the <i>Server</i> will register itself. Allow complete disabling of registration with a <i>Discovery Server</i> .
Client	Discovery Client Find Servers Basic	Uses the FindServers <i>Service</i> to obtain all <i>Servers</i> installed on a given platform.
Client	Discovery Client Find Servers with URI	Use FindServers <i>Service</i> to obtain URLs for specific <i>Server</i> URIs.
Client	Discovery Client Find Servers Dynamic	Detect new <i>Servers</i> after an initial FindServers <i>Service</i> call.
Client	Discovery Client Find Servers on Network	Support one of the options to locate <i>Servers</i> on the network.
Client	Discovery Client Find Servers on Network using LDS-ME	Use FindServersOnNetwork <i>Service</i> to obtain URLs for specific <i>Server</i> URIs. Note that this <i>Service</i> is available via the Local <i>Discovery Server</i> with multicast extension (LDS-ME).
Client	Discovery Client Find Servers on Network using mDNS	Use mDNS based <i>Service Discovery</i> to locate <i>Servers</i> on the same multicast network. The contents of mDNS records for OPC UA <i>Discovery</i> are described in IEC 62541-12. Note that this functionality is only required for <i>Clients</i> when there is no Local <i>Discovery Server</i> with multicast extension (LDS-ME). The capability identifiers and their use in mDNS records are specified in IEC 62541-12.
Client	Discovery Client Find Servers in GDS	Use the QueryServers <i>Method</i> on the GDS Directory <i>Object</i> to locate <i>Servers</i> that meet filter criteria specified in the request. This <i>Method</i> is specified in IEC 62541-12.
Client	Discovery Client Find Applications in GDS	Use the QueryApplications <i>Method</i> on the GDS Directory <i>Object</i> to locate Applications that meet filter criteria specified in the request. This <i>Method</i> is specified in IEC 62541-12.

Category	Title	Description
<i>Client</i>	<i>Discovery Client</i> Get Endpoints Basic	Uses the <i>GetEndpoints Service</i> to obtain all Endpoints for a given <i>Server</i> URI.
<i>Client</i>	<i>Discovery Client</i> Get Endpoints SessionLess	Uses the <i>GetEndpoints Service</i> with a filter to obtain Endpoints that can be used for <i>SessionLess Service</i> invocation. The filter is the Transport URI extended with the query string "SL". E.g. http://opcfoundation.org/UA-Profile/Transport/https-uajson?SL .
<i>Client</i>	<i>Discovery Client</i> Get Endpoints Dynamic	Detect changes to the Endpoints after an initial <i>GetEndpoints Service</i> call.
<i>Client</i>	<i>Discovery Client</i> Configure Endpoint	Allow specification of an Endpoint without going through the <i>Discovery Service</i> Set.

The *Session Service* Set is composed of multiple *ConformanceUnits* (see Table 4). The *CreateSession*, *ActivateSession*, and *CloseSession* services are supported as a single unit. All *Servers* and *Clients* provide this functionality.

Table 4 – Session Services

Category	Title	Description
<i>Server</i>	<i>Session</i> General <i>Service</i> Behaviour	Implement basic <i>Service</i> behaviour. This includes in particular: – checking the authentication token – returning the requestHandle in responses – returning available diagnostic information as requested with the 'returnDiagnostics' parameter – respecting a timeoutHint
<i>Server</i>	<i>Session</i> Base	Support the <i>Session Service</i> Set (<i>CreateSession</i> , <i>ActivateSession</i> , <i>CloseSession</i>) except the use of <i>ActivateSession</i> to change the <i>Session</i> user. This includes correct handling of all parameters that are provided. Note that for the <i>CreateSession</i> and <i>ActivateSession</i> services, if the <i>SecurityMode</i> = None then: 1) The Application <i>Certificate</i> and <i>Nonce</i> are optional. 2) The signatures are null/empty. The details of this are described in IEC 62541-4.
<i>Server</i>	<i>Session</i> Change User	Support the use of <i>ActivateSession</i> to change the <i>Session</i> user.
<i>Server</i>	<i>Session</i> Cancel	Support the <i>Cancel Service</i> to cancel outstanding requests.
<i>Server</i>	<i>Session</i> Minimum 1	Support minimum 1 <i>Session</i> (total).
<i>Server</i>	<i>Session</i> Minimum 2 Parallel	Support minimum 2 parallel <i>Sessions</i> (total for all <i>Clients</i>).
<i>Server</i>	<i>Session</i> Minimum 50 Parallel	Support minimum 50 parallel <i>Sessions</i> (total for all <i>Clients</i>).
<i>Server</i>	<i>Session</i> Sessionless Invocation	Defines the support of the <i>SessionlessInvoke</i> Service defined in IEC 62541-4 to process any of the <i>Services</i> (like <i>Read/Write</i> , <i>Browse</i> , or <i>Call</i>) that are designated for <i>Sessionless</i> Invocation.
<i>Client</i>	<i>Session Client</i> General <i>Service</i> Behaviour	Implement basic <i>Service</i> behaviour. This includes in particular: – including the proper authentication token of the <i>Session</i> – creating a requestHandle if needed – requesting diagnostic information with the 'returnDiagnostics' parameter – evaluate the serviceResult and operational results
<i>Client</i>	<i>Session Client</i> Base	Use the <i>Session Service</i> Set (<i>CreateSession</i> , <i>ActivateSession</i> , and <i>CloseSession</i>) except the use of <i>ActivateSession</i> to change the <i>Session</i> user. This includes correct handling of all parameters that are provided Note that for the <i>CreateSession</i> and <i>ActivateSession</i> services, if the <i>SecurityMode</i> = None then: 1) The Application <i>Certificate</i> and <i>Nonce</i> are optional. 2) The signatures are null/empty.

Category	Title	Description
<i>Client</i>	<i>Session Client</i> Multiple Connections	Support unlimited connections (client side) with multiple <i>Servers</i> . Any limit on numbers of connections is from server side. May have a memory-based limit, but not a software constraint limit.
<i>Client</i>	<i>Session Client</i> Renew NodeIds	This <i>ConformanceUnit</i> applies to <i>Clients</i> that allow persisting NodeIds. Verify that the Namespace Table has not changed for NodeIds that the <i>Client</i> has persisted and is going to re-use beyond a <i>Session</i> lifetime. If changes occurred the <i>Client</i> has to recalculate the Namespace Indices of the respective NodeIds.
<i>Client</i>	<i>Session Client</i> Impersonate	Uses <i>ActivateSession</i> to change the <i>Session</i> user (impersonation).
<i>Client</i>	<i>Session Client</i> KeepAlive	Make periodic requests to keep the <i>Session</i> alive.
<i>Client</i>	<i>Session Client</i> Detect Shutdown	Read or monitor the <i>ServerStatus/State</i> Variable to recognize a potential shutdown of the <i>Server</i> and clean up resources.
<i>Client</i>	<i>Session Client</i> Cancel	Use the <i>Cancel Service</i> to cancel outstanding requests.
<i>Client</i>	<i>Session Client</i> Auto Reconnect	Automatic <i>Client</i> reconnect including: – <i>ActivateSession</i> with new <i>SecureChannel</i> if <i>SecureChannel</i> is no longer valid but <i>Session</i> is still valid – Creation of a new <i>Session</i> only if <i>Session</i> is no longer valid
<i>Client</i>	<i>Session Client</i> Single <i>Session</i>	The <i>Client</i> shall interoperate with <i>Servers</i> that only support one <i>Session</i> .
<i>Client</i>	<i>Session Client</i> <i>SessionLess Service</i> Calls	Defines the use of the <i>SessionlessInvoke Service</i> defined in IEC 62541-4 to request one of the <i>Services</i> (like <i>Read</i> or <i>Browse</i>) that are allowed for <i>Sessionless</i> Invocation. IEC 62541-6 specifies which transports may be used and how.

The *Node Management Service* Set is composed of multiple *ConformanceUnits* (see Table 5). *Servers* may provide some aspects of this functionality; see *Profiles* categorized as *Server Profiles* for details. *Clients* may support some aspects of this functionality; see *Profiles* categorized as *Client Profiles* for details.

Table 5 – Node Management Services

Category	Title	Description
<i>Server</i>	<i>Node Management</i> Add <i>Node</i>	Support the <i>AddNodes Service</i> to add one or more <i>Nodes</i> into the OPC UA <i>AddressSpace</i> .
<i>Server</i>	<i>Node Management</i> Delete <i>Node</i>	Support the <i>DeleteNodes Service</i> to delete one or more <i>Nodes</i> from the OPC UA <i>AddressSpace</i> .
<i>Server</i>	<i>Node Management</i> Add Ref	Support the <i>AddReferences Service</i> to add one or more <i>References</i> to one or more <i>Nodes</i> in the OPC UA <i>AddressSpace</i> .
<i>Server</i>	<i>Node Management</i> Delete Ref	Support the <i>DeleteReferences Service</i> to delete one or more <i>References</i> of a <i>Node</i> in the OPC UA <i>AddressSpace</i> .
<i>Client</i>	<i>Node Management</i> <i>Client</i>	Uses <i>Node Management Services</i> to add or delete <i>Nodes</i> and to add or delete <i>References</i> in <i>Server's</i> OPC UA <i>AddressSpace</i> .

The *View Service* Set is composed of a multiple *ConformanceUnits* (see Table 6). All *Servers* support some aspects of this conformance group. *Clients* may support some aspects of this functionality; see *Profiles* categorized as *Client Profiles* for details.

Table 6 – View Services

Category	Title	Description
<i>Server</i>	View Basic	Support the View <i>Service</i> Set (Browse, BrowseNext).
<i>Server</i>	View TranslateBrowsePath	Support TranslateBrowsePathsToNodeIds <i>Service</i> .
<i>Server</i>	View RegisterNodes	Support the RegisterNodes and UnregisterNodes <i>Services</i> as a way to optimize access to repeatedly used <i>Nodes</i> in the <i>Server's</i> OPC UA <i>AddressSpace</i> .
<i>Server</i>	View Minimum Continuation Point 01	Support minimum 1 continuation point per <i>Session</i> .
<i>Server</i>	View Minimum Continuation Point 05	Support minimum 5 continuation points per <i>Session</i> . This number has to be supported for at least half of the minimum required sessions.
<i>Client</i>	View <i>Client</i> Basic Browse	Uses Browse and BrowseNext <i>Services</i> to navigate through the <i>Server's</i> OPC UA <i>AddressSpace</i> . Make use of the referenceTypeId and the nodeClassMask to specify the needed <i>References</i> .
<i>Client</i>	View <i>Client</i> Remote Nodes Browse	The <i>Client</i> can browse to nodes that have an extended NodeID that reference a <i>Server</i> different than the originating <i>Server</i> . This includes automatic connection to the remote <i>Server</i> . It is acceptable that the <i>Server</i> configuration information be pre-configured on the <i>Client</i> and / or that the user is prompted to connect.
<i>Client</i>	View <i>Client</i> Basic ResultSet Filtering	Makes use of the resultMask parameter to optimize the result set to be returned by the <i>Server</i> .
<i>Client</i>	View <i>Client</i> TranslateBrowsePath	Uses the TranslateBrowsePathsToNodeIds <i>Service</i> to identify the NodeIds for <i>Nodes</i> where a starting <i>Node</i> and a BrowsePath is known. Makes use of bulk operations rather than multiple calls whenever possible.
<i>Client</i>	View <i>Client</i> Remote Nodes Translate Browse	The <i>Client</i> can translate browse paths that include nodes with extended NodeID that reference a <i>Server</i> different than the originating <i>Server</i> and return them as part of the TranslateBrowsePathsToNodeIds <i>Service</i> . It is acceptable that the <i>Server</i> configuration information be pre-configured on the <i>Client</i> .
<i>Client</i>	View <i>Client</i> RegisterNodes	Uses the RegisterNodes <i>Service</i> to optimize access for <i>Nodes</i> that are used repeatedly. Use UnregisterNodes when <i>Nodes</i> are not used anymore.

The *Attribute Service* Set is composed of multiple *ConformanceUnits* (see Table 7). The majority of the *Attribute* service set is a core functionality of OPC UA and as such is supported by most *Servers*. Most *Clients* will also support some aspects of the *Attribute Service* Set.

Table 7 – Attribute Services

Category	Title	Description
Server	<i>Attribute Read</i>	Supports the Read <i>Service</i> to read one or more <i>Attributes</i> of one or more <i>Nodes</i> . This includes support of the <i>IndexRange</i> parameter to read a single element or a range of elements when the <i>Attribute</i> value is an array.
Server	<i>Attribute Read Complex</i>	Supports reading and encoding Values with structured DataTypes.
Server	<i>Attribute Write Values</i>	Supports writing to values to one or more <i>Attributes</i> of one or more <i>Nodes</i> .
Server	<i>Attribute Write Complex</i>	Supports writing and decoding Values with structured DataTypes.
Server	<i>Attribute Write StatusCode & Timestamp</i>	Supports writing of StatusCode and Timestamps along with the Value.
Server	<i>Attribute Write Index</i>	Supports the <i>IndexRange</i> to write a single element or a range of elements when the <i>Attribute</i> value is an array and partial updates are allowed for this array.
Server	<i>Attribute Alternate Encoding</i>	Supports alternate Data Encoding when reading value <i>Attributes</i> . By default, every <i>Server</i> has to support the Data Encoding of the currently used Stack <i>Profile</i> (i.e. binary with UA Binary Encoding and XML with XML Encoding). This <i>ConformanceUnit</i> – when supported – specifies that the other Data Encoding is supported in addition.
Server	<i>Attribute Historical Read</i>	Supports the HistoryRead <i>Service</i> . The details of what aspects of this service are used are listed in additional <i>ConformanceUnits</i> , but at least one of ReadRaw, ReadProcessed, ReadModified, ReadAtTime or ReadEvents shall be supported.
Server	<i>Attribute Historical Update</i>	Supports the HistoryUpdate service. The details of the supported features of this service are described by additional <i>ConformanceUnits</i> , but at least one of the following shall be supported: InsertData, InsertEvents, ReplaceData, ReplaceEvents, UpdateData, UpdateEvents, DeleteData, DeleteEvents or DeleteAtTime.
Client	<i>Attribute Client Read Base</i>	Use the Read <i>Service</i> to read one or more <i>Attributes</i> of one or more <i>Nodes</i> . This includes use of an <i>IndexRange</i> to select a single element or a range of elements when the <i>Attribute</i> value is an array. <i>Clients</i> shall use bulk operations whenever possible to reduce the number of <i>Service</i> invocations.
Client	<i>Attribute Client Remote Nodes Attribute Access</i>	The <i>Client</i> can retrieve attributes of nodes that have an extended NodeID that reference a <i>Server</i> different than the originating <i>Server</i> . This requires a connection to the remote <i>Server</i> for access (not necessarily displayed as a connection). It is acceptable that the <i>Server</i> configuration information be pre-configured on the <i>Client</i> .
Client	<i>Attribute Client Read with proper Encoding</i>	This <i>ConformanceUnit</i> refers to the ability of a <i>Client</i> to discover the available encodings and choose a specific one when calling the Read <i>Service</i> .
Client	<i>Attribute Client Read Complex</i>	Read and decode Values with structured DataTypes.
Client	<i>Attribute Client Write Base</i>	Use the Write <i>Service</i> to write values to one or more <i>Attributes</i> of one or more <i>Nodes</i> . This includes use of an <i>IndexRange</i> to select a single element or a range of elements when the <i>Attribute</i> value is an array. <i>Clients</i> shall use bulk operations whenever possible to reduce the number of <i>Service</i> invocations.
Client	<i>Attribute Client Write Complex</i>	Write and Encode Values with structured DataTypes.
Client	<i>Attribute Client Write Quality & Timestamp</i>	Use the Write <i>Service</i> to also write StatusCode and/or Timestamps along with a Value.
Client	<i>Attribute Client Historical Read</i>	The <i>Client</i> makes use of the HistoryRead service. The details of which aspect of this service are used are provided by additional <i>ConformanceUnits</i> , but at least one or more of the following is used ReadRaw, ReadAtTime, ReadProcessed, ReadModified or ReadEvents.

Category	Title	Description
<i>Client</i>	<i>Attribute Client</i> Historical Updates	The <i>Client</i> makes use of the HistoryUpdate service. The details of this usage are provided by additional <i>ConformanceUnits</i> , but at least one or more of the following shall be provided InsertData, InsertEvents, ReplaceData, ReplaceEvents, UpdateData, UpdateEvents, DeleteData or DeleteEvents or DeleteAtTime.

The *Method Service* Set is composed of *ConformanceUnits* (see Table 8). The primary *ConformanceUnits* provide support for the call functionality. *Servers* may provide some aspects of this functionality; see *Profiles* categorized as *Server Profiles* for details. *Clients* may support some aspects of this functionality; see *Profiles* categorized as *Client Profiles* for details.

Table 8 – Method Services

Category	Title	Description
<i>Server</i>	<i>Method</i> Call	Support the Call <i>Service</i> to call (invoke) a <i>Method</i> which includes support for <i>Method</i> Parameters.
<i>Client</i>	<i>Method Client</i> Call	Use the Call <i>Service</i> to call one or several <i>Methods</i> .

The *MonitoredItem Service* Set is composed of multiple *ConformanceUnits* (see Table 9). *Servers* may provide some aspects of this functionality; see *Profiles* categorized as *Server Profiles* for details. *Clients* may support some aspects of this functionality; see *Profiles* categorized as *Client Profiles* for details.

IECNORM.COM : Click to view the full PDF IEC 62541-7:2020 RLV

Table 9 – Monitored Item Services

Category	Title	Description
Server	Monitor Basic	Support the following <i>MonitoredItem Services</i> : CreateMonitoredItems, ModifyMonitoredItems, DeleteMonitoredItems and SetMonitoringMode.
Server	Monitor Value Change	Support creation of <i>MonitoredItems</i> for <i>Attribute</i> value changes. This includes support of the <i>IndexRange</i> to select a single element or a range of elements when the <i>Attribute</i> value is an array.
Server	Monitor Complex Value	Supports monitoring and encoding Values with structured <i>DataTypes</i> .
Server	Monitored Items Deadband Filter	Supports an absolute Deadband filter as a <i>DataChangeFilter</i> for numeric data types.
Server	Monitor Aggregate Filter	Support for Aggregate filters for <i>MonitoredItems</i> . The result of this <i>ConformanceUnit</i> includes a list of Aggregates that are supported as part of the <i>Profile Certificate</i> .
Server	Monitor Alternate Encoding	Support alternate encoding when monitoring value <i>Attributes</i> . By default, every <i>Server</i> has to support the encoding of the currently used <i>Stack Profile</i> (i.e. binary with UA Binary Encoding and XML with XML Encoding). This <i>ConformanceUnit</i> – when supported – specifies that the other encoding is supported in addition.
Server	Monitor Items 2	Support at least 2 <i>MonitoredItems</i> per <i>Subscription</i> where the size of each <i>MonitoredItem</i> is at least equal to size of Double.
Server	Monitor Items 10	Support at least 10 <i>MonitoredItems</i> per <i>Subscription</i> where the size of each <i>MonitoredItem</i> is at least equal to size of Double.
Server	Monitor Items 100	Support at least 100 <i>MonitoredItems</i> per <i>Subscription</i> . This number has to be supported for at least half of the required Subscriptions for half of the required Sessions.
Server	Monitor Items 500	Support at least 500 <i>MonitoredItems</i> per <i>Subscription</i> . This number has to be supported for at least half of the required Subscriptions for half of the required Sessions.
Server	Monitor QueueSize_1	This <i>ConformanceUnit</i> does not require queuing when multiple value changes occur during a "publish period". I.e. the latest change will be sent in the <i>Notification</i> .
Server	Monitor MinQueueSize_02	Support at least 2 queue entries for <i>MonitoredItems</i> . <i>Servers</i> often will adapt the queue size to the number of currently <i>MonitoredItems</i> . However, it is expected that <i>Servers</i> support this minimum queue size for at least one third of the supported <i>MonitoredItems</i> .
Server	Monitor MinQueueSize_05	Support at least 5 queue entries for <i>MonitoredItems</i> . <i>Servers</i> often will adapt the queue size to the number of currently <i>MonitoredItems</i> . However, it is expected that <i>Servers</i> support this minimum queue size for at least one third of the supported <i>MonitoredItems</i> .
Server	Monitor QueueSize_ServerMax	This <i>ConformanceUnit</i> is for events. When the <i>Client</i> requests queueSize=MAXUInt32 the <i>Server</i> is to return the maximum queue size that it can support for event notifications as the revisedQueueSize.
Server	Monitor Triggering	Support the SetTriggering <i>Service</i> to create and/or delete triggering links for a triggering item.
Server	Monitor Events	Support creation of <i>MonitoredItems</i> for an "EventNotifier Attribute" for the purpose of <i>Event Notification</i> . The subscription includes supporting a filter that includes SimpleAttribute Operands and a select list of Operators. The list of Operators includes: Equals, IsNull, GreaterThan, LessThan, GreaterThanorEqual, LessThanorEqual, Like, Not, Between, InList, And, Or, Cast, BitwiseAnd, BitwiseOr.
Server	Monitor Complex Event Filter	Support for the 'TypeOf' complex <i>Event</i> filter operator.

Category	Title	Description
<i>Client</i>	Monitor <i>Client</i> Value Change	Use the <i>MonitoredItem Service</i> Set to register items for changes in <i>Attribute</i> value. Use <i>CreateMonitoredItems</i> to register the <i>Node/Attribute</i> tuple. Set proper sampling interval, Deadband filter and queuing mode. Use disabling / enabling instead of deleting and re-creating a <i>MonitoredItem</i> . Use bulk operations rather than individual service requests to reduce communication overhead.
<i>Client</i>	Monitor <i>Client</i> Complex Value	Monitor and decode Values with structured DataTypes.
<i>Client</i>	Monitor <i>Client</i> Deadband Filter	Uses Absolute Deadband filters for subscriptions.
<i>Client</i>	Monitor <i>Client</i> by Index	Use the <i>IndexRange</i> to select a single element or a range of elements when the <i>Attribute</i> value is an array.
<i>Client</i>	Monitor <i>Client</i> Aggregate Filter	Uses Aggregate filters for Subscriptions.
<i>Client</i>	Monitor <i>Client</i> Events	Use the <i>MonitoredItem Service</i> Set to create <i>MonitoredItems</i> for <i>Event</i> notifications.
<i>Client</i>	Monitor <i>Client</i> Event Filter	Use the <i>Event</i> filter when calling <i>CreateMonitoredItems</i> to filter the desired Events and to select the columns to be provided for each <i>Event Notification</i> .
<i>Client</i>	Monitor <i>Client</i> Complex Event Filter	Use of the 'TypeOf' complex <i>Event</i> filter operator.
<i>Client</i>	Monitor <i>Client</i> Modify	Use <i>ModifyMonitoredItems Service</i> to change the configuration setting. Use <i>SetMonitoringMode Service</i> to disable/enable sampling and/or publishing.
<i>Client</i>	Monitor <i>Client</i> Trigger	Use the Triggering Model if certain items are to be reported only if some other item triggers. Use proper monitoring mode for these items. Use <i>SetTriggering Service</i> to link these items to the trigger item.

The *Subscription Service* Set is composed of multiple *ConformanceUnits* (see Table 10). *Servers* may provide some aspects of this functionality; see *Profiles* categorized as *Server Profiles* for details. *Clients* may support some aspects of this functionality; see *Profiles* categorized as *Client Profiles* for details.

Table 10 – Subscription Services

Category	Title	Description
Server	<i>Subscription Basic</i>	Support the following <i>Subscription Services</i> : CreateSubscription, ModifySubscription, DeleteSubscriptions, Publish, Republish and SetPublishingMode.
Server	<i>Subscription Minimum 1</i>	Support at least 1 <i>Subscription</i> per <i>Session</i> . This number has to be supported for all of the minimum required sessions.
Server	<i>Subscription Minimum 02</i>	Support at least 2 Subscriptions per <i>Session</i> . This number has to be supported for at least half of the minimum required sessions.
Server	<i>Subscription Minimum 05</i>	Support at least 5 Subscriptions per <i>Session</i> . This number has to be supported for at least half of the minimum required sessions.
Server	<i>Subscription Publish Min 02</i>	Support at least 2 Publish <i>Service</i> requests per <i>Session</i> . This number has to be supported for all of the minimum required sessions. Support of a NotificationMessage retransmission queue is not required; if not available, the Republish <i>Service</i> returns Bad_MessageNotAvailable.
Server	<i>Subscription Publish Min 05</i>	Support at least 5 Publish <i>Service</i> requests per <i>Session</i> . This number has to be supported for at least half of the minimum required sessions. Support, as a minimum, the number of Publish requests per session as the size of the NotificationMessage retransmission queue for Republish.
Server	<i>Subscription Publish Min 10</i>	Support at least 10 Publish <i>Service</i> requests per <i>Session</i> . This number has to be supported for at least half of the minimum required sessions. Support, as a minimum, the number of Publish requests per session as the size of the NotificationMessage retransmission queue for Republish.
Server	<i>Subscription Publish Discard Policy</i>	Respect the specified policy for discarding Publish <i>Service</i> requests. If the maximum number of Publish <i>Service</i> requests has been queued and a new Publish <i>Service</i> request arrives, the "oldest" Publish request has to be discarded by returning the proper error.
Server	<i>Subscription Transfer</i>	Support TransferSubscriptions <i>Service</i> to transfer a <i>Subscription</i> from one <i>Session</i> to another.
Server	<i>Subscription Durable</i>	Support setting Subscriptions in durable mode. This mode requires that collected data and events are stored and delivered even if a <i>Client</i> was disconnected for a longer time or the <i>Server</i> was restarted. Support one of the " <i>Subscription Durable StorageLevel nnn</i> " <i>ConformanceUnits</i> .
Server	<i>Subscription Durable StorageLevel Small</i>	Support at least 20 monitored items with a queue size of 10 000 for each item and where the size of each <i>MonitoredItem</i> is at least equal to size of Double. This requires storage capacity for 200 000 values of <i>DataType Double</i> .
Server	<i>Subscription Durable StorageLevel Medium</i>	Support at least 100 monitored items with a queue size of 50 000 for each item and where the size of each <i>MonitoredItem</i> is at least equal to size of Double. This requires storage capacity for 5 million values of <i>DataType Double</i> .
Server	<i>Subscription Durable StorageLevel High</i>	Support at least 2 000 monitored items with a queue size of 200 000 for each item and where the size of each <i>MonitoredItem</i> is at least equal to size of Double. This requires storage capacity for 400 million values of <i>DataType Double</i> .

Category	Title	Description
<i>Client</i>	<i>Subscription Client Basic</i>	Use the <i>Subscription</i> and <i>MonitoredItem Service</i> Set as an efficient means to detect changes of <i>Attribute</i> values and / or to receive <i>Event</i> occurrences. Set appropriate intervals for publishing, keep alive notifications and total <i>Subscription</i> lifetime. Supply a sufficient number of Publish requests to the <i>Server</i> so that <i>Notifications</i> can be sent whenever a publish timer expires. Acknowledge received <i>Notifications</i> with subsequent Publish requests.
<i>Client</i>	<i>Subscription Client Fallback</i>	The <i>Client</i> shall interoperate with <i>Servers</i> that do not support Subscriptions, or have exhausted <i>Subscription</i> limits, for Monitoring by using Read <i>Service</i> .
<i>Client</i>	<i>Subscription Client Republish</i>	Evaluate the sequence number in <i>Notifications</i> to detect lost <i>Notifications</i> . Use Republish to request missing <i>Notifications</i> .
<i>Client</i>	<i>Subscription Client Modify</i>	Allow modification of the <i>Subscription</i> configuration using the <i>ModifySubscription Service</i> .
<i>Client</i>	<i>Subscription Client TransferSubscriptions</i>	The <i>Client</i> supports transferring <i>Subscription</i> from other <i>Clients</i> . This <i>ConformanceUnit</i> is used as part of redundant <i>Clients</i> .
<i>Client</i>	<i>Subscription Client Multiple</i>	Use multiple Subscriptions to reduce the payload of individual <i>Notifications</i> .
<i>Client</i>	<i>Subscription Client Publish Configurable</i>	Send multiple Publish <i>Service</i> requests to assure that the <i>Server</i> is always able to send <i>Notifications</i> . The number of parallel Publish <i>Service</i> requests per <i>Session</i> shall be configurable.
<i>Client</i>	<i>Subscription Client Durable</i>	Use durable Subscriptions.

5.3 Transport and communication related features

Table 11 describes security related *ConformanceUnits*. All of these *ConformanceUnits* apply equally to both *Clients* and *Servers*, where a *Client* uses the related security unit and a *Server* supports the use of it. These items are defined in detail in IEC 62541-6. It is recommended that a *Server* and *Client* support as many of these options as possible in order to achieve increased levels of interoperability. It is the task of an administrator to determine which of these *ConformanceUnits* are exposed in a given deployed *Server* or *Client* application.

Table 11 – Security

Category	Title	Description
Security	Security User Name Password	The <i>Server</i> supports User Name/Password combination(s). The token will be encrypted if required by the security policy of the User Token Policy or by the security policy of the endpoint. An unencrypted token either requires message encryption or means outside the scope of OPC UA to secure the identity token so that it cannot be retrieved by sniffing the communication. One option would be a secure transport like a VPN.
Security	Security User X509	The <i>Server</i> supports a public/private key pair for user identity. The use of this feature shall be able to be enabled or disabled by an administrator.
Security	Security User IssuedToken Kerberos	The <i>Server</i> supports a Kerberos <i>Server</i> token for User Identity. The use of this feature shall be able to be enabled or disabled by an Administrator. The use of this token is defined in Kerberos Token Documentation. The token will be encrypted if required by the security policy of the User Token Policy or by the security policy of the endpoint. An unencrypted token either requires message encryption or means outside the scope of OPC UA to secure the identity token so that it cannot be retrieved by sniffing the communication. One option would be a secure transport like a VPN.
Security	Security User IssuedToken Kerberos Windows	The <i>Server</i> supports the Windows implementation of Kerberos Tokens. This <i>ConformanceUnit</i> only applies if the "Security User IssuedToken Kerberos" is supported. The token will be encrypted if required by the security policy of the User Token Policy or by the security policy of the endpoint. An unencrypted token either requires message encryption or means outside the scope of OPC UA to secure the identity token so that it cannot be retrieved by sniffing the communication. One option would be a secure transport like a VPN.
Security	Security User JWT IssuedToken	The <i>Server</i> supports a JSON Web Token (JWT) for user identity. IEC 62541-6 describes OAuth2 and JWTs in more detail. The use of this feature shall be able to be enabled or disabled by an administrator. The token will be encrypted if required by the security policy of the User Token Policy or by the security policy of the endpoint. An unencrypted token either requires message encryption or means outside the scope of OPC UA to secure the identity token so that it cannot be retrieved by sniffing the communication. One option would be a secure transport like a VPN.
Security	Security User Anonymous	The <i>Server</i> provides support for Anonymous access. The use of this feature shall be able to be enabled or disabled by an Administrator. By default, Anonymous access shall be disabled.
Security	Security User Name Password <i>Client</i>	A <i>Client</i> uses a User Name/Password combination. The token will be encrypted if required by the security policy of the User Token Policy or by the security policy of the endpoint. An unencrypted token either requires message encryption or means outside the scope of OPC UA to secure the identity token so that it cannot be retrieved by sniffing the communication. One option would be a secure transport like a VPN.
Security	Security User X509 <i>Client</i>	A <i>Client</i> uses a public/private key pair for user identity. This includes all validation and trust issues associated with a certificate.
Security	Security User IssuedToken Kerberos <i>Client</i>	A <i>Client</i> uses a Kerberos <i>Server</i> token. The use of this token is defined by the Kerberos documentation. The token will be encrypted if required by the security policy of the User Token Policy or by the security policy of the endpoint. An unencrypted token either requires message encryption or means outside the scope of OPC UA to secure the identity token so that it cannot be retrieved by sniffing the communication. One option would be a secure transport like a VPN.

Category	Title	Description
Security	Security User IssuedToken Kerberos Windows <i>Client</i>	A <i>Client</i> uses the Windows implementation of Kerberos tokens. This <i>ConformanceUnit</i> only applies if the "Security User IssuedToken Kerberos <i>Client</i>" is supported. The token will be encrypted if required by the security policy of the User Token Policy or by the security policy of the endpoint. An unencrypted token either requires message encryption or means outside the scope of OPC UA to secure the identity token so that it cannot be retrieved by sniffing the communication. One option would be a secure transport like a VPN.
Security	Security User JWT IssuedToken <i>Client</i>	A <i>Client</i> uses a JSON Web Token (JWT) for user identity. IEC 62541-6 describes OAuth2 and JWTs in more detail. The token will be encrypted if required by the security policy of the User Token Policy or by the security policy of the endpoint. An unencrypted token either requires message encryption or means outside the scope of OPC UA to secure the identity token so that it cannot be retrieved by sniffing the communication. One option would be a secure transport like a VPN.
Security	Security Invalid user token	<i>Servers</i> shall take proper measures to protect against attacks on user identity tokens. Such an attack is assumed if repeated connection attempts with invalid user identity tokens happen. See <i>ActivateSession Service</i> in IEC 62541-4.
Security	Security User JWT Token Policy	The Server supports one or more Endpoints with a UserTokenPolicy that includes a JWT IssuerEndpointUrl as defined in IEC 62541-6. For JWT the issuerEndpointUrl is a JSON object that includes all parameters that define the AuthorizationService. As part of the JWT Token Policy, the Server shall support at least one of the following Authority Profile Conformance Units. The URIs defined in the ConformanceUnit shall be exposed in the authorityProfileURI field of the JWT Token Policy.
Security	Security User JWT Token Policy <i>Client</i>	The Client understands and uses the Authorization Service definition inside the JWT UserTokenPolicy returned with GetEndpoints. It shall support at least one of the following Authority Profile Conformance Units. The URIs defined in the ConformanceUnit are in the authorityProfileURI field of the JWT Token Policy exposed in Server Endpoints.
Security	OAuth2 Authority <i>Profile</i>	This unit indicates support of OAuth2 over HTTPS to request access tokens. The URI for the interactions with this authority is "http://opcfoundation.org/UA/Authorization#OAuth2"
Security	OPC UA Authority <i>Profile</i>	This unit indicates support of the OPC UA <i>Methods</i> defined in IEC 62541-12 to request access tokens. The URI for the interactions with this authority is "http://opcfoundation.org/UA/Authorization#OPCUA"
Security	Azure Identity Provider Authority <i>Profile</i>	This unit indicates support of the Azure identity provider to request access tokens. The URI for the interactions with this authority is "http://opcfoundation.org/UA/Authorization#Azure"
Security	Security <i>Certificate</i> Validation	A certificate will be validated as specified in IEC 62541-4. This includes, among others, structure and signature examination. Allowing for some validation errors to be suppressed by administration directive.
Security	Security Default ApplicationInstance Certificate	An application, when installed, has a default ApplicationInstanceCertificate that is valid. The default ApplicationInstanceCertificate shall either be created as part of the installation or installation instructions explicitly describe the process to create and apply a default ApplicationInstanceCertificate to the application.
Security	Security – No Application Authentication	The <i>Server</i> supports being able to be configured for no application authentication, just User authentication and normal encryption/signing: – Configure <i>Server</i> to accept all certificates – <i>Certificates</i> are just used for message security (signing and encryption) – Users level is used for authentication

Category	Title	Description
Security	Security Policy Required	Support at least Security Policy [A] and Security Policy [B]. Support of multiple Security Policies – even obsolete ones – is recommended. This will provide best interoperability and allows the end user to choose the required level of security. Obsolete Security Policies shall not be enabled/usable without administrative intervention.
Security	Security None CreateSession ActivateSession	When SecurityPolicy = None, the CreateSession and ActivateSession service allow for a NULL/empty signature and do not require Application <i>Certificates</i> or a Nonce.
Security	Security None CreateSession ActivateSession 1.0	The <i>Client</i> can connect to <i>Servers</i> that require a certificate being passed on Session establishment. The <i>Client</i> in this case will first try without a <i>Certificate</i> and if this fails present a <i>Certificate</i> .
Security	Security TLS General	This <i>ConformanceUnit</i> indicates that at least one of the transport security <i>Profiles</i> for TLS is supported by this application. It is used in TLS transport <i>Profiles</i> , but the choice of transport security profile is optional. The actual used security profile will default to the most secure one.
Security	Security TLS_RSA with AES_256_CBC_SHA256	The connection is established using TLS_RSA_WITH_AES_256_CBC_SHA256. That has a MinAsymmetricKeyLength – 2048, MaxAsymmetricKeyLength – 4096, AsymmetricSignatureAlgorithm – RSA_SHA256 (TLS 1.2)
Security	Security TLS_DHE_RSA with AES_nnn_CBC_SHA256	The connection is established using TLS_DHE_RSA with AES_128_CBC_SHA256 or TLS_DHE_RSA with AES_256_CBC_SHA256. That has a MinAsymmetricKeyLength – 2048, MaxAsymmetricKeyLength – 4096, CertificateSignatureAlgorithm – RSA_SHA256. (TLS 1.2). <i>Clients</i> and <i>Servers</i> have to support both algorithms.
Security	Security Encryption Required	Encryption is required using the algorithms provided in the security algorithm suite.
Security	Security Signing Required	Signing is required using the algorithms provided in the security algorithm suite.
Security	Security Time Synch – Configuration	Application supports configuring acceptable clock skew.
Security	Security Time Synch – NTP / OS Based support	Application supports time synchronization, either via an implementation of Network Time Protocol (NTP), or via features of a standard operating system.
Security	Security Time Synch – UA based support	An application makes use of the responses header timestamp provided by a configured well know source, such as a <i>Discovery Server</i> to synchronize the time on the application and that this time synchronization occurs periodically. Use of this TimeSyncing can be configured.
Security	Security Administration	Allow configuration of the following Security related items (when they apply). * select the allowed User identification policy or policies (e.g. Username/Password or X509). * enable/disable the security policy "None" or other security policies. * enable/disable endpoints with MessageSecurityMode SIGN or SIGNANDENCRYPT. * set the permitted certification authorities. * define how to react to unknown Certificates. * allow accepting any valid Certificate
Security	Security Administration – XML Schema	Support the OPC UA defined XML schema for importing and exporting security configuration information. This schema is defined in IEC 62541-6.
Security	Security <i>Certificate</i> Administration	Allow a site administrator to be able to assign a site specific ApplicationInstanceCertificate and if desired to configure a site-specific <i>Certificate</i> Authority (CA).
Security	Security Role <i>Server</i> Base	Support the User Authorization Information Model defined in UA Parts 3 and 5 – like Roles – and the RolePermissions and UserRolePermissions <i>Attributes</i> .
Security	Security Role Well Known	Support the well-known Roles "ConfigureAdmin" and "SecurityAdmin" with suggested permissions defined in IEC 62541-3.
Security	Security Role <i>Server</i> IdentityManagement	Allow authorized users to add and/or remove Identities from Roles with the appropriate <i>Methods</i> .

Category	Title	Description
Security	Security Role <i>Server</i> Management	Allow authorized users to create new Roles and/or remove Roles with the appropriate <i>Methods</i> .
Security	Security Role <i>Server</i> Restrict Applications	Support adding applications to a Role with the appropriate <i>Methods</i> so that only these applications can use this Role.
Security	Security Role <i>Server</i> Restrict Endpoints	Support adding Endpoints to a Role with the appropriate <i>Methods</i> . With this restriction a Role is only applied when a <i>Client</i> connects via one of these Endpoints.
Security	Security Role <i>Server</i> DefaultRolePermissions	Allow authorized users to set the DefaultRolePermissions <i>Property</i> for certain NameSpaces. DefaultRolePermissions are applied if no RolePermissions are associated with a <i>Node</i> .
Security	Security Role <i>Server</i> RolePermissions	Allow authorized users to set the RolePermissions <i>Attribute</i> on <i>Nodes</i> .
Security	Security Role <i>Server</i> Authorization	Restrict access based on the configured Roles and permissions.
Security	Security Role <i>Client</i> Base	Understand and use the User Authorization Information Model defined in IEC 62541-5 and the RolePermissions <i>Attribute</i> .
Security	Security Role <i>Client</i> Management	Support creating new Roles and adding Identities as well as remove Roles or Identities using the appropriate <i>Methods</i> .
Security	Security Role <i>Client</i> Restrict Applications	Use the appropriate <i>Methods</i> to add applications to a Role so that only these applications can use this Role.
Security	Security Role <i>Client</i> Restrict Endpoints	Use the appropriate <i>Methods</i> to add Endpoints to a Role. With this restriction a Role is only applied when a <i>Client</i> connects via one of these Endpoints.
Security	Security Role <i>Client</i> DefaultRolePermissions	Ability to set the DefaultRolePermissions <i>Property</i> for certain NameSpaces. DefaultRolePermissions are applied if no RolePermissions are associated with a <i>Node</i> .
Security	Security Role <i>Client</i> RolePermissions	Support setting the RolePermissions <i>Attribute</i> on <i>Nodes</i> .
Security	Pull Model for Global <i>Certificate</i> and TrustList Management	Use the <i>Certificate</i> Management <i>Services</i> of IEC 62541-12 for the Pull model to manage Application Instance <i>Certificates</i> and Trust Lists including Revocation Lists.
Security	Push Model for Global <i>Certificate</i> and TrustList Management	Support the <i>Certificate</i> Management <i>Services</i> of IEC 62541-12 for the Push model to manage Application Instance <i>Certificates</i> and Trust Lists including Revocation Lists.
Security	Pull Model for KeyCredential <i>Service</i>	Use the <i>Methods</i> on an instance of the KeyCredentialServiceType (Pull model) to obtain KeyCredentials as specified in IEC 62541-12.
Security	Push Model for KeyCredential <i>Service</i>	Support the KeyCredential <i>Services</i> Push model of IEC 62541-12 to obtain KeyCredentials. This includes support of one or more instances of the KeyCredentialConfigurationType and the <i>Methods</i> to update or delete credentials.
Security	Authorization Service Configuration Server	Support the Object Types defined in IEC 62541-12 to allow configuration of information needed to accept Access Tokens when presented by the Client during session establishment. Access Tokens are issued by Authorization <i>Services</i> .
Security	Authorization Service Client	Use the RequestAccessToken Method defined in IEC 62541-12.
Security	SymmetricSignatureAlgorithm_None	This algorithm does not apply.
Security	SymmetricSignatureAlgorithm_HMAC-SHA1	A keyed hash which is defined in https://tools.ietf.org/html/rfc2104 . The hash algorithm is SHA1 and is described in https://tools.ietf.org/html/rfc3174 . The URI is http://www.w3.org/2000/09/xmldsig#hmac-sha1 . No known exploits exist when using SHA1 with a keyed hash, however, SHA1 was broken in 2017 so use of this algorithm is not recommended.
Security	SymmetricSignatureAlgorithm_HMAC-SHA2-256	A keyed hash used for message authentication which is defined in https://tools.ietf.org/html/rfc2104 . The hash algorithm is SHA2 with 256 bits and described in https://tools.ietf.org/html/rfc4634

Category	Title	Description
Security	SymmetricEncryptionAlgorithm_None	This algorithm does not apply.
Security	SymmetricEncryptionAlgorithm_AES128-CBC	The AES encryption algorithm which is defined in http://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.197.pdf. Multiple blocks encrypted using the CBC mode described in http://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-38a.pdf. The key size is 128 bits. The block size is 16 bytes. The URI is http://www.w3.org/2001/04/xmlenc#aes128-cbc.
Security	SymmetricEncryptionAlgorithm_AES256-CBC	The AES encryption algorithm which is defined in http://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.197.pdf. Multiple blocks encrypted using the CBC mode described in http://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-38a.pdf. The key size is 256 bits. The block size is 16 bytes. The URI is http://www.w3.org/2001/04/xmlenc#aes256-cbc.
Security	SymmetricEncryptionAlgorithm_AES128-CTR	The AES encryption algorithm which is defined in http://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.197.pdf. Multiple blocks encrypted using the CTR mode described in http://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-38a.pdf. The counter block format is defined in https://tools.ietf.org/html/rfc3686. The key size is 128 bits. The block size is 16 bytes. The input nonce length is 4 bytes. The URI is http://opcfoundation.org/UA/security/aes128-ctr.
Security	SymmetricEncryptionAlgorithm_AES256-CTR	The AES encryption algorithm which is defined in http://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.197.pdf. The key size is 256 bits. The block size is 16 bytes. Multiple blocks encrypted using the CTR mode described in http://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-38a.pdf. The counter block format is defined in https://tools.ietf.org/html/rfc3686. The key size is 128 bits. The block size is 16 bytes. The input nonce length is 4 bytes. The URI is http://opcfoundation.org/UA/security/aes256-ctr.
Security	AsymmetricSignatureAlgorithm_None	This algorithm does not apply.
Security	AsymmetricSignatureAlgorithm_RSA-PKCS15-SHA1	The RSA signature algorithm which is defined in https://tools.ietf.org/html/rfc3447. The RSASSA-PKCS1-v1_5 scheme is used. The hash algorithm is SHA1 and is described in https://tools.ietf.org/html/rfc3174. The URI is http://www.w3.org/2000/09/xmldsig#rsa-sha1. SHA1 was broken in 2017 so this algorithm should not be used.
Security	AsymmetricSignatureAlgorithm_RSA-PKCS15-SHA2-256	The RSA signature algorithm which is defined in https://tools.ietf.org/html/rfc3447. The RSASSA-PKCS1-v1_5 scheme is used. The hash algorithm is SHA2 with 256bits and is described in https://tools.ietf.org/html/rfc6234. The URI is http://www.w3.org/2001/04/xmldsig-more#rsa-sha256.

Category	Title	Description
Security	AsymmetricSignatureAlgorithm_RSA-PSS - SHA2-256	The RSA signature algorithm which is defined in https://tools.ietf.org/html/rfc3447. The RSASSA-PSS scheme is used. The hash algorithm is SHA2 with 256bits and is described in https://tools.ietf.org/html/rfc6234. The mask generation algorithm also uses SHA2 with 256 bits. The salt length is 32 bytes. The URI is http://opcfoundation.org/UA/security/rsa-pss-sha2-256.
Security	AsymmetricEncryptionAlgorithm_None	This algorithm does not apply.
Security	AsymmetricEncryptionAlgorithm_RSA-PKCS15	The RSA encryption algorithm which is defined in https://tools.ietf.org/html/rfc3447. The RSAES-PKCS1-v1_5 scheme is used. The URI is http://www.w3.org/2001/04/xmlenc#rsa-1_5. The RSAES-PKCS1-v1_5 scheme has known weaknesses and is not recommended.
Security	AsymmetricEncryptionAlgorithm_RSA-OAEP-SHA1	The RSA encryption algorithm which is defined in https://tools.ietf.org/html/rfc3447. The RSAES-OAEP scheme is used. The hash algorithm is SHA1 and is described in https://tools.ietf.org/html/rfc6234. The mask generation algorithm also uses SHA1. The URI is http://www.w3.org/2001/04/xmlenc#rsa-oaep. No known exploits exist when using SHA1 with RSAES-OAEP, however, SHA1 was broken in 2017 so use of this algorithm is not recommended.
Security	AsymmetricEncryptionAlgorithm_RSA-OAEP-SHA2-256	The RSA encryption algorithm which is defined in https://tools.ietf.org/html/rfc3447. The RSAES-OAEP scheme is used. The hash algorithm is SHA2 with 256 bits and is described in https://tools.ietf.org/html/rfc6234. The mask generation algorithm also uses SHA2 with 256 bits. The URI is http://opcfoundation.org/UA/security/rsa-oaep-sha2-256.
Security	KeyDerivationAlgorithm_None	This algorithm does not apply.
Security	KeyDerivationAlgorithm_P-SHA1	The P_SHA-1 pseudo-random function defined in https://tools.ietf.org/html/rfc4346. The URI is http://docs.oasis-open.org/ws-sx/ws-secureconversation/200512/dk/p_sha1. No known exploits exist when using SHA1 with P-SHA-1, however, SHA1 was broken in 2017 so use of this algorithm is not recommended.
Security	KeyDerivationAlgorithm_P-SHA2-256	The P_SHA256 pseudo-random function defined in https://tools.ietf.org/html/rfc5246. The URI is http://docs.oasis-open.org/ws-sx/ws-secureconversation/200512/dk/p_sha256.
Security	CertificateSignatureAlgorithm_None	This algorithm does not apply.

Category	Title	Description
Security	CertificateSignatureAlgorithm_RSA-PKCS15-SHA2-256	The RSA signature algorithm which is defined in https://tools.ietf.org/html/rfc3447. The RSASSA-PKCS1-v1_5 scheme is used. The hash algorithm is SHA2 with 256bits and is described in https://tools.ietf.org/html/rfc6234. The SHA2 algorithm with 384 or 512 bits may be used instead of SHA2 with 256 bits. The URI is http://www.w3.org/2001/04/xmldsig-more#rsa-sha256.
Security	CertificateSignatureAlgorithm_RSA-PKCS15-SHA1	The RSA signature algorithm which is defined in https://tools.ietf.org/html/rfc3447. The RSASSA-PKCS1-v1_5 scheme is used. The hash algorithm is SHA1 and is described in https://tools.ietf.org/html/rfc3174. The URI is http://www.w3.org/2000/09/xmldsig#rsa-sha1. SHA1 was broken in 2017 so this algorithm should not be used. The SHA2 algorithm with 244, 256, 384 or 512 bits may be used instead of SHA1. The SHA2 algorithm is described in https://tools.ietf.org/html/rfc6234.
Security	SecurityPolicy_None_Limits	DerivedSignatureKeyLength: 0
Security	Aes128-Sha256-RsaOaep_Limits	-> DerivedSignatureKeyLength: 256 bits -> MinAsymmetricKeyLength: 2048 bits -> MaxAsymmetricKeyLength: 4096 bits -> SecureChannelNonceLength: 32 bytes
Security	Basic256Sha256_Limits	-> DerivedSignatureKeyLength: 256 bits -> MinAsymmetricKeyLength: 2048 bits -> MaxAsymmetricKeyLength: 4096 bits -> SecureChannelNonceLength: 32 bytes
Security	Aes256-Sha256-RsaPss_Limits	-> DerivedSignatureKeyLength: 256 bits -> MinAsymmetricKeyLength: 2048 bits -> MaxAsymmetricKeyLength: 4096 bits -> SecureChannelNonceLength: 32 bytes
Security	Basic128Rsa15_Limits	-> DerivedSignatureKeyLength: 128 bits -> MinAsymmetricKeyLength: 1024 bits -> MaxAsymmetricKeyLength: 2048 bits -> SecureChannelNonceLength: 16 bytes
Security	Basic256_Limits	-> DerivedSignatureKeyLength: 192 bits -> MinAsymmetricKeyLength: 1024 bits -> MaxAsymmetricKeyLength: 2048 bits -> SecureChannelNonceLength: 32 bytes

Table 12 describes protocol and encoding related features that can be profiled. These features are defined in detail in IEC 62541-6. It is recommended that *Servers* and *Clients* support as many of these options as possible for greatest interoperability.

Table 12 – Protocol and Encoding

Category	Title	Description
<i>Server</i>	Protocol Reverse Connect <i>Server</i>	Support reverse connectivity by sending a ReverseHello message to a Client. The reverse connect procedure can be applied to several transports as specified in IEC 62541-6 and shall be supported for all of these that are available in a Server.
<i>Server</i>	Protocol Configuration	Allow administration of the Endpoints and the port number used by the Endpoints.
<i>Client</i>	Protocol Reverse Connect <i>Client</i>	Support reverse connectivity by accepting Reverse Hello messages from Servers and establish a Secure Channel if the URI of the Server is accepted by Client or Client user. The reverse connect procedure can be applied to several transports as specified in IEC 62541-6 and shall be supported for all of these transports that are supported by the Client.
Transport	Protocol UA TCP	Support the UA TCP transport protocol as defined in IEC 62541-6.
Transport	Protocol HTTPS	Support the HTTPS protocol as defined in IEC 62541-6.
Transport	Protocol Web Sockets	Support the WebSocket protocol (WSS) with at least one of the sub-protocols defined in IEC 62541-6.
Transport	UA Secure Conversation	Support UA Secure Conversation specified in IEC 62541-6.
Transport	UA Binary Encoding	Support UA Binary Encoding. Values of these data types are encoded in compact binary formats, contiguously and without tagging. I.e. the receiver is assumed to understand the structure it is decoding.
Transport	UA SOAP-XML Encoding	Support Soap V1.2 based Xml Encoding as defined in IEC 62541-6. The XML elements include all information necessary to convert it back into OPC UA structures of any language.
Transport	JSON Reversible Encoding	Support reversible JSON Encoding as defined in IEC 62541-6. The JSON object includes all information necessary to convert it back into OPC UA structures of any language.

5.4 Information Model and AddressSpace related features

Table 13 describes base features related items that can be profiled. For additional information about these items, please refer to IEC 62541-3, IEC 62541-5 and IEC 62541-6. *Servers* with a larger resource capacity would support most of this functionality, but smaller resource constraint *Server* may omit some of this functionality. Many *Clients* would utilize some of this functionality and more robust *Clients* would utilize most of this functionality.

Table 13 – Base Information

Category	Title	Description
Server	Base Info Core Structure	The <i>Server</i> supports the <i>Server Object</i> , <i>ServerCapabilities</i> and supports the OPC UA <i>AddressSpace</i> structure.
Server	Base Info <i>Server</i> Capabilities	The <i>Server</i> supports publishing of the <i>Server</i> limitation in the <i>ServerCapabilities</i> , including <i>MaxArrayLength</i> , <i>MaxStringLength</i> , <i>MaxNodePerRead</i> , <i>MaxNodesPerWrite</i> , <i>MaxNodesPerSubscription</i> and <i>MaxNodesPerBrowse</i> .
Server	Base Info Progress Events	The <i>Server</i> exposes if generation of Progress events for long running service calls such as <i>HistoryRead</i> or <i>Query</i> is supported. If it is listed as supported in <i>ServerCapabilities</i> , then the actual events are verified.
Server	Base Info Diagnostics	The <i>Server</i> supports the collection of diagnostic information. The <i>EnabledFlag</i> in the <i>ServerDiagnostics Object</i> can be set TRUE and in that case all static and dynamic <i>Objects</i> and <i>Variables</i> for diagnostic data as defined in IEC 62541-5 are supported.
Server	Base Info System Status	The <i>Server</i> supports generating <i>SystemStatusChangeEvent</i> indicating shutdown of the <i>Server</i> (<i>SourceNode=Server</i>).
Server	Base Info Estimated Return Time	<i>Server</i> supports the <i>EstimatedReturnTime Property</i> . It indicates the time at which the <i>Server</i> is expected to have a <i>ServerStatus.State</i> of <i>RUNNING</i> . <i>Clients</i> can use this information to govern the reconnect logic.
Server	Base Info System Status Underlying System	The <i>Server</i> supports generating <i>SystemStatusChangeEvent</i> indicating changes to an Underlying System (<i>SourceNode = Server</i>). This event can also be used to indicate that the OPC UA <i>Server</i> has underlying systems.
Server	Base Info Device Failure	The <i>Server</i> supports generating <i>DeviceFailureEvent</i> indicating changes to individual devices in an underlying system.
Server	Base Info <i>GetMonitoredItems Method</i>	The <i>Server</i> supports obtaining subscription information via <i>GetMonitoredItems Method</i> on the <i>Server</i> object.
Server	Base Info <i>ResendData Method</i>	Support the standard <i>Method</i> <i>ResendData</i> (defined in IEC 62541-5) to get the latest value of the monitored items of a <i>Subscription</i> .
Server	Base Info Type System	The <i>Server</i> exposes a Type System with <i>DataTypes</i> , <i>ReferenceTypes</i> , <i>ObjectTypes</i> and <i>VariableTypes</i> including all of the OPC UA (namespace 0) types that are used by the <i>Server</i> , as defined in IEC 62541-5. Items that are defined in Namespace 0 but are defined in other specification parts are tested as part of the other information models.
Server	Base Info Custom Type System	The <i>Server</i> supports custom types (i.e. types that are derived from well-known <i>ObjectTypes</i> , <i>VariableTypes</i> , <i>ReferenceTypes</i> or <i>DataTypes</i>). Supporting this conformance unit requires that the custom types with their full inheritance tree are exposed in the <i>AddressSpace</i> .
Server	Base Info Model Change	The <i>Server</i> supports <i>ModelChange Event</i> and <i>NodeVersion Property</i> for all <i>Nodes</i> that the server allows Model changes for.
Server	Base Info Placeholder Modelling Rules	The <i>Server</i> supports defining custom <i>Object</i> or <i>Variables</i> that include the use of <i>OptionalPlaceholder</i> or <i>MandatoryPlaceholder</i> modelling rules.
Server	Base Info SemanticChange	The <i>Server</i> supports <i>SemanticChangeEvent</i> for some <i>Properties</i> . This includes setting the <i>SemanticChange</i> Bit in the status when a semantic change occurs, such as a change in the engineering unit associated with a value.
Server	Base Info <i>EventQueueOverflow EventType</i>	The <i>Server</i> supports the <i>EventQueueOverflowEventType</i> as defined in IEC 62541-4.
Server	Base Info <i>OptionSet</i>	The <i>Server</i> supports the <i>VariableType OptionSet</i> .
Server	Base Info <i>ValueAsText</i>	The <i>Server</i> supports the <i>Property ValueAsText</i> for enumerated <i>DataTypes</i> .
Server	Base Info Engineering Units	The <i>Server</i> supports defining <i>Variables</i> that include the <i>Engineering Units Property</i> . This property makes use of the <i>EUInformation</i> data structure. This structure by default represents the UN/CEFACT "Codes for Units of Measurement". If a different EU representation is required then the <i>EUInformation.namespaceUri</i> will indicate the alternate namespace.

Category	Title	Description
Server	Base Info Selection List	The <i>Server</i> supports Variables of the SelectionListType VariableType.
Server	Base Info FileType Base	The <i>Server</i> supports the FileType <i>Object</i> (see IEC 62541-5). File writing may be restricted.
Server	Base Info FileType Write	The <i>Server</i> supports the FileType <i>Object</i> , including writing of files. Also included is the support of user access control on FileType <i>Object</i> .
Server	Base Info RequestServerStateChange Method	The <i>Server</i> supports the RequestServerStateChange <i>Method</i> .
Server	Base Info State Machine Instance	Support instances of the StateMachineType or a sub-type in the AddressSpace. Generate Events when significant state changes occur. At least one GeneratesEvent <i>Reference</i> exists to define the Event(s) triggered on state changes.
Server	Base Info Finite State Machine Instance	Support instances of the FiniteStateMachineType or a sub-type in the AddressSpace.
Server	Base Info Available States and Transitions	Support the Properties AvailableStates and AvailableTransitions defined for the FiniteStateMachineType.
Client	Base Info Client Basic	The <i>Client</i> uses the defined OPC UA AddressSpace. Access or provide access to <i>Server</i> information like the <i>Server's</i> state, BuildInfo, capabilities, Namespace Table and Type Model.
Client	Base Info Client Honour Operation Limits	The <i>Client</i> shall honour <i>Server</i> limits described in ServerCapabilites <i>Object</i> of <i>Server</i> .
Client	Base Info Event Processing	The <i>Client</i> is able to subscribe for and process base OPC UA Events.
Client	Base Info Client System Status	The <i>Client</i> makes use of SystemStatusChangeEventType to detect server shutdowns.
Client	Base Info Client Estimated Return Time	<i>Client</i> uses the EstimatedReturnTime <i>Property</i> to govern the reconnect logic.
Client	Base Info Client System Status Underlying System	The <i>Client</i> makes use of SystemStatusChangeEventType to detect changes to an Underlying System (SourceNode = <i>Server</i>).
Client	Base Info Client Device Failure	The <i>Client</i> makes use of DeviceFailureEventType to detect failed devices in underlying systems
Client	Base Info Client Progress Events	The <i>Client</i> makes use of ProgressEvents, including checking for their support.
Client	Base Info Client Diagnostics	The <i>Client</i> provides interactive or programmatic access to the <i>Server's</i> diagnostic information.
Client	Base Info Client Type Programming	The <i>Client</i> programmatically process instances of <i>Objects</i> or <i>Variables</i> by using their type definitions. This includes custom DataTypes, <i>ObjectTypes</i> and VariableTypes.
Client	Base Info Client Type Pre-Knowledge	The <i>Client</i> shall interoperate with <i>Servers</i> that do not expose OPC UA Types in AddressSpace.
Client	Base Info Client Remote Nodes	The <i>Client</i> can access Nodes that have an extended NodeID that reference a <i>Server</i> different then the originating <i>Server</i> . It is acceptable that the <i>Server</i> configuration information be pre-configured on the <i>Client</i> .
Client	Base Info Client Change Events	The <i>Client</i> processes ModelChangeEvents to detect changes in the <i>Server's</i> OPC UA AddressSpace and take appropriate action for a given change.
Client	Base Info Client GetMonitoredItems Method	The <i>Client</i> makes use of GetMonitoredItems <i>Method</i> to recover for communication interruptions and/or to recover subscription information.
Client	Base Data Client ResendData Method	The <i>Client</i> makes use of ResendData <i>Method</i> to fetch the last value of the data monitored items.
Client	Base Info Client Selection List	The <i>Client</i> uses and understands Variables of the SelectionListType VariableType.
Client	Base Info Client FileType Base	The <i>Client</i> can access a FileType <i>Object</i> to transfer a file from the <i>Server</i> to the <i>Client</i> . This includes large files.
Client	Base Info Client FileType Write	The <i>Client</i> can access a FileType <i>Object</i> to transfer a file from the <i>Client</i> to the <i>Server</i> . This includes large files.

Category	Title	Description
<i>Client</i>	Base Info <i>Client</i> RequestServerStateChange	The <i>Client</i> can invoke the RequestServerStateChange <i>Method</i> .
<i>Client</i>	Base Info <i>Client</i> State Machine Instance	Use instances of the StateMachineType or a sub-type. Monitor either the CurrentState component <i>Variable</i> of the instance or the Events triggered as effect of state changes. Use Methods when defined for the StateMachineType to affect the state.
<i>Client</i>	Base Info <i>Client</i> Finite State Machine Instance	Use instances of the FinitStateMachineType or a sub-type. Monitor either the CurrentState component <i>Variable</i> of the instance or the Events triggered as effect of state changes.
<i>Client</i>	Base Info Client Available States and Transitions	Use the Properties AvailableStates and AvailableTransitions when exposed by a <i>Server</i> .

Table 14 describes Address Space Model information related items that can be profiled. The details of these model items are defined in IEC 62541-3 and IEC 62541-5. This includes *Server Facets* that describe what a *Server* exposes and *Client Facets* that describe what a *Client* consumes.

Table 14 – Address Space Model

Category	Title	Description
<i>Server</i>	Address Space Base	Support the <i>NodeClasses</i> with their <i>Attributes</i> and <i>References</i> as defined in IEC 62541-3. This includes for instance: <i>Object</i> , <i>ObjectType</i> , <i>Variable</i> , <i>VariableType</i> , <i>References</i> and <i>DataTypes</i> .
<i>Server</i>	Address Space Atomicity	Support setting the NonatomicRead and NonatomicWrite flags in the AccessLevelEx <i>Attribute</i> for <i>Variable Nodes</i> to indicate whether Read or Write operations can be performed in atomic manner. If the flags are set to '1', atomicity cannot be assured.
<i>Server</i>	Address Space Full Array Only	Support setting the WriteFullArrayOnly flag in the AccessLevelEx <i>Attribute</i> for <i>Variable Nodes</i> of non-scalar data types to indicate whether write operations for an array can be performed with an IndexRange.
<i>Server</i>	Address Space Events	Support OPC UA <i>AddressSpace</i> elements for generating <i>Event</i> notifications. This includes at least one <i>Node</i> with an <i>EventNotifier Attribute</i> set to True (<i>Server Node</i>).
<i>Server</i>	Address Space Complex Data Dictionary	Support structured <i>DataTypes</i> with a Data Dictionary. Note that V1.04 of IEC 62541-3 specifies a simplified approach using the new <i>DataTypeDefinition Attribute</i> . The "Address Space <i>DataTypeDefinition Attribute</i> " Conformance Unit requires support of the <i>DataTypeDefinition Attribute</i> . Support of a <i>DataDictionary</i> will be deprecated in one of the next OPC UA versions.
<i>Server</i>	Address Space <i>DataTypeDefinition Attribute</i>	Support structured <i>DataTypes</i> and expose the meta data and encoding information with a <i>StructureDefinitionType</i> via the <i>DataTypeDefinition Attribute</i> .
<i>Server</i>	Address Space <i>Method</i>	Support <i>Method Nodes</i> .
<i>Server</i>	Address Space Notifier Hierarchy	Supports using the HasNotifier reference to build a hierarchy of <i>Object Nodes</i> that are notifiers with other notifier <i>Object Nodes</i> .
<i>Server</i>	Address Space Source Hierarchy	Supports hierarchies of event sources where each hierarchy roots in an <i>Object Node</i> that is a notifier. The HasEventSource <i>Reference</i> is used to relate the <i>Nodes</i> within a hierarchy. If <i>Conditions</i> are supported, the hierarchy shall include HasCondition <i>References</i> .
<i>Server</i>	Address Space WriteMask	Supports WriteMask indicating the write access availability for all attributes, including not supported attributes.
<i>Server</i>	Address Space UserWriteMask	Supports UserWriteMask indicating the write access availability for all attributes for the given user, including not supported attributes. Support includes at least two levels of users.
<i>Server</i>	Address Space UserWriteMask Multilevel	Supports UserWriteMask indicating the write access availability for all attributes for the given user, including not supported attributes. This includes supporting multiple levels of access control for all nodes in the system.
<i>Server</i>	Address Space User Access Level Full	Implements User Access Level security, this includes supporting multiple levels of access control for <i>Variable</i> nodes in the system. This includes an indication of read, write, Historical read and Historical write access to the <i>Value Attribute</i> .

Category	Title	Description
Server	Address Space User Access Level Base	Implements User Access Level Security for <i>Variable</i> nodes, this includes at least two users in the system. This includes an indication of read, write, historical read and Historical write access to the value attribute
Client	Address Space <i>Client</i> Base	Uses and understands the <i>NodeClasses</i> with their <i>Attributes</i> and behaviour as defined in IEC 62541-3. This includes for instance: <i>Object</i> , <i>ObjectType</i> , <i>Variable</i> , <i>VariableType</i> , <i>References</i> and <i>DataType</i> . This includes treating BrowseNames and String NodeIds as case sensitive.
Client	Address Space <i>Client</i> Atomicity	Access the NonatomicRead or NonatomicWrite flags in the AccessLevelEx <i>Attribute</i> of <i>Variable Nodes</i> to determine whether Read or Write operations can be performed in atomic manner. This information will typically be shown to a user for further action.
Client	Address Space <i>Client</i> Full Array Only	Access the WriteFullArrayOnly flag in the AccessLevelEx <i>Attribute</i> of <i>Variable Nodes</i> with non-scalar data types to determine whether writing to an array with an IndexRange is allowed.
Client	Address Space <i>Client</i> Complex Data Dictionary	Uses and understands arbitrary structured DataTypes via Data Dictionary. Note that V1.04 of IEC 62541-3 specifies a simplified approach using the new DataTypeDefinition <i>Attribute</i> . The "Address Space <i>Client</i> DataTypeDefinition <i>Attribute</i> " Conformance Unit requires support of the DataTypeDefinition <i>Attribute</i> .
Client	Address Space <i>Client</i> DataTypeDefinition <i>Attribute</i>	Uses and understands arbitrary structured DataTypes where the meta data and encoding information are exposed with the StructureDefinitionType via the DataTypeDefinition <i>Attribute</i> .
Client	Address Space <i>Client</i> Notifier Hierarchy	Uses hierarchy of <i>Object Nodes</i> that are notifiers to detect specific areas where the <i>Client</i> can subscribe for Events.
Client	Address Space <i>Client</i> Source Hierarchy	Detect and use the hierarchy of event sources exposed for specific <i>Object Nodes</i> that are event notifiers.

Table 15 describes Data Access information model related items that can be profiled. The details of this model are defined in IEC 62541-8. *Servers* may expose this information model and *Clients* may utilize this information model.

Table 15 – Data Access

Category	Title	Description
Server	Data Access DataItems	Provide <i>Variables</i> of DataItemType or one of its subtypes. Support the StatusCodes specified in IEC 62541-8. Support of optional Properties (e.g. "InstrumentRange") shall be verified during certification testing and will be shown in the <i>Certificate</i> .
Server	Data Access AnalogItems	Support AnalogItemType <i>Variables</i> with corresponding Properties. The support of optional properties will be listed.
Server	Data Access PercentDeadband	Support PercentDeadband filter when monitoring AnalogItemType <i>Variables</i> .
Server	Data Access Semantic Changes	Support semantic changes of AnalogItemType items (EURange <i>Property</i> and/or EngineeringUnits <i>Property</i>). Support semantic change StatusCode bits where appropriate.
Server	Data Access TwoState	Support TwoStateDiscreteType <i>Variables</i> with corresponding Properties.
Server	Data Access MultiState	Support MultiStateDiscreteType <i>Variables</i> with corresponding Properties.
Server	Data Access MultiStateValueDiscrete	Support MultiStateValueDiscreteType <i>Variables</i> with corresponding Properties.
Server	Data Access ArrayItemType	Provide <i>Variables</i> of ArrayItemType or one of its subtypes (YArrayItemType, XYArrayItemType, ImageArrayType, CubeArrayType and NDimensionArrayType). The supported subtypes will be listed. Support for this type includes supporting all of the mandatory properties including AxisInformation.
Server	Data Access Complex Number	Supports the Complex Number data type. This data type is available for any variable types that do not have other explicit restrictions.
Server	Data Access DoubleComplex Number	Supports the DoubleComplex Number data type. This data type is available for any variable types that do not have other explicit restrictions.

Category	Title	Description
<i>Client</i>	Data Access <i>Client</i> Basic	Understand the <i>.DataAccess Variable</i> Types. Make use of the standard Properties if applicable.
<i>Client</i>	Data Access <i>Client</i> AnalogItems	Understand <i>AnalogItemType Variables</i> with corresponding Properties.
<i>Client</i>	Data Access <i>Client</i> TwoState	Understand <i>TwoStateDiscreteType Variables</i> with corresponding Properties.
<i>Client</i>	Data Access <i>Client</i> MultiState	Understand <i>MultiStateDiscreteType Variables</i> with corresponding Properties.
<i>Client</i>	Data Access <i>Client</i> MultiStateValueDiscrete	Understand <i>MultiStateValueDiscreteType Variables</i> with corresponding Properties.
<i>Client</i>	Data Access <i>Client</i> Deadband	Use <i>PercentDeadband</i> to filter value changes of <i>AnalogItemType Variables</i> .
<i>Client</i>	Data Access <i>Client</i> SemanticChange	Recognize the semantic change bit in the <i>StatusCode</i> while monitoring items and take proper action. Typically, the <i>Client</i> has to re-read Properties that define type-specific semantic like the <i>EURange</i> and <i>EngineeringUnits</i> Properties.

Table 16 describes *Alarm* and *Conditions* information model related items that can be profiled. The details of this model are defined in IEC 62541-9. *Servers* that deal with *Alarm* and *Conditions* would expose this information model and *Clients* that process *Alarms* and *Conditions* would utilize this information model.

Table 16 – Alarms and Conditions

Category	Title	Description
<i>Server</i>	A & C Basic	Supports <i>Alarm & Condition</i> model <i>ConditionType</i> .
<i>Server</i>	A & C Enable	Supports Enable and Disable Methods.
<i>Server</i>	A & C Refresh	Supports <i>ConditionRefresh Method</i> and the concept of a refresh.
<i>Server</i>	A & C Refresh2	Supports <i>ConditionRefresh2 Method</i> and the concept of a monitored item-based refresh.
<i>Server</i>	A & C Instances	Support exposing of A&C <i>Condition</i> instances in the <i>AddressSpace</i> .
<i>Server</i>	A & C ConditionClasses	Supports multiple <i>Condition</i> classes for grouping and filtering of <i>Alarms</i> .
<i>Server</i>	A & C Condition Sub-Classes	Support assigning multiple <i>Condition</i> sub-classes for grouping and filtering of <i>Alarms</i> .
<i>Server</i>	A & C Acknowledge	Supports the Acknowledge concept, Acknowledge <i>Method</i> , and AcknowledgeableCondition Type.
<i>Server</i>	A & C Confirm	Supports the concept of Confirm and the Confirm <i>Method</i> .
<i>Server</i>	A & C Comment	Supports the concept of Comments and the AddComment <i>Method</i> .
<i>Server</i>	A & C Alarm	Supports the mandatory features of the AlarmCondition Type.
<i>Server</i>	A & C Alarm Metrics	Support the collection of alarm metrics data as defined in IEC 62541-9. This implies one of more instances of the AlarmMetricsType.
<i>Server</i>	A & C Branch	Support for branching of <i>Condition</i> Types and any subtypes, such as AcknowledgeableConditionType and AlarmConditionType etc.
<i>Server</i>	A & C Shelving	Support the concept of shelving and the TimedShelve, OneShotShelve and Unshelve Methods.
<i>Server</i>	A & C Suppression	Support the SuppressedState.
<i>Server</i>	A & C Suppression by Operator	Support the Suppress and UnSuppress <i>Methods</i> to allow an operator control over the SuppressedState.
<i>Server</i>	A & C Silencing	Support the concept of silencing and the Silence Method.
<i>Server</i>	A & C Out Of Service	Support the OutOfService state and the OutOfService Method.
<i>Server</i>	A & C On-Off Delay	Support the OnDelay and OffDelay Properties to eliminate nuisance <i>Alarms</i> .
<i>Server</i>	A & C Re-Alarming	Support the ReAlarmTime and ReAlarmRepeatCount Properties that define automatic re-annunciation of <i>Alarms</i> in certain conditions.

Category	Title	Description
Server	A & C First in Group <i>Alarm</i>	Support the "FirstInGroup" elements for an <i>Alarm</i> , indicating which <i>Alarm</i> of a group was the trigger.
Server	A & C Audible Sound	Support the AudibleSound <i>Property</i> . This <i>Property</i> contains the sound file that is to be played if an audible <i>Alarm</i> is to be generated.
Server	A & C Exclusive Level	Supports Exclusive Level <i>Alarm</i> type.
Server	A & C Exclusive Limit	Supports Exclusive Limit <i>Alarms</i> . A <i>Server</i> that supports this shall support at least one of the sub-types: Level, Deviation or RateOfChange.
Server	A & C Exclusive Deviation	Supports Exclusive Deviation <i>Alarm</i> type.
Server	A & C Exclusive RateOfChange	Supports Exclusive RateOfChange <i>Alarm</i> type.
Server	A & C Non-Exclusive Limit	Supports Non-Exclusive Limit <i>Alarms</i> . A <i>Server</i> that supports this shall support at least one of the sub-types: Level, Deviation or RateOfChange.
Server	A & C Non-Exclusive Level	Supports Non-Exclusive Level <i>Alarm</i> type.
Server	A & C Non-Exclusive Deviation	Supports Non-Exclusive Deviation <i>Alarm</i> type.
Server	A & C Non-Exclusive RateOfChange	Supports Non-Exclusive RateOfChange <i>Alarm</i> type.
Server	A & C Discrete	Supports Discrete <i>Alarm</i> types.
Server	A & C OffNormal	Supports OffNormalAlarmType.
Server	A & C SystemOffNormal	Supports SystemOffNormalAlarmType.
Server	A & C Trip	Supports Trip <i>Alarm</i> type.
Server	A & C Discrepancy	Supports Discrepancy <i>Alarm</i> type.
Server	A & C Dialog	Supports DialogConditionType including Respond <i>Method</i> .
Server	A & C CertificateExpiration	Supports CertificateExpirationAlarmType.
Server	A & E Wrapper Mapping	The <i>Server</i> uses the COM A&E mapping specified in the annex of IEC 62541-9 to map OPC-COM Events to A&C Events. This includes <i>Condition</i> Class mapping.
Client	A & C Basic <i>Client</i>	Uses the <i>Alarm</i> & <i>Condition</i> model ConditionType.
Client	A & C Enable <i>Client</i>	Uses Enable and Disable Methods.
Client	A & C Refresh <i>Client</i>	Uses ConditionRefresh <i>Method</i> and the concept of a refresh.
Client	A & C Refresh2 <i>Client</i>	Uses ConditionRefresh2 <i>Method</i> and the concept of a monitored item-based refresh.
Client	A & C Instances <i>Client</i>	Uses A&C <i>Condition</i> instances when they are exposed in the AddressSpace.
Client	A & C ConditionClasses <i>Client</i>	Uses <i>Condition</i> classes to group <i>Alarms</i> .
Client	A & C <i>Condition</i> Sub-Classes <i>Client</i>	Uses <i>Condition</i> sub-classes to group or filter <i>Alarms</i> .
Client	A & C Acknowledge <i>Client</i>	Understands the Acknowledge concept and the AcknowledgeableCondition Type, and uses the Acknowledge <i>Method</i> if requested.
Client	A & C Confirm <i>Client</i>	Understands the concept of confirming <i>Conditions</i> and uses the Confirm <i>Method</i> .
Client	A & C Comment <i>Client</i>	Understands the concept of Comments and uses the AddComment <i>Method</i> .
Client	A & C <i>Alarm</i> <i>Client</i>	Understands the concept of <i>Alarms</i> and uses the mandatory features of the AlarmCondition Type,
Client	A & C <i>Alarm</i> Metrics <i>Client</i>	Understand and use <i>Alarm</i> metrics data as defined in IEC 62541-9. This implies discovery of instances of the AlarmMetricsType that can exist anywhere in the HasNotifier hierarchy.
Client	A & C Branch <i>Client</i>	Can make use of and process <i>Condition</i> Branches, including all actions associated with previous <i>Condition</i> instances.
Client	A & C Shelving <i>Client</i>	Understand the shelving model and use the TimedShelve, OneShotShelve and Unshelve Methods.
Client	A & C Suppression <i>Client</i>	Understand the SuppressedState model.
Client	A & C Suppression by Operator <i>Client</i>	Use the Suppress and UnSuppress <i>Methods</i> to allow an operator control over the SuppressedState.
Client	A & C Silencing <i>Client</i>	Understand the SilencedState model and use the Silence <i>Method</i> .

Category	Title	Description
<i>Client</i>	A & C Out Of Service Client	Understand the OutOfServiceState model and use the OutOfService Method.
<i>Client</i>	A & C On-Off Delay <i>Client</i>	Uses the OnDelay and OffDelay Properties to eliminate nuisance Alarms.
<i>Client</i>	A & C Re-Alarming <i>Client</i>	Understand and use the ReAlarmTime and ReAlarmRepeatCount Properties. Configure the ReAlarmTime <i>Property</i> for automatic re-annunciation of an <i>Alarm</i> . Note that configuration is only possible for <i>Servers</i> that expose <i>Alarm</i> instances.
<i>Client</i>	A & C First in Group Alarm <i>Client</i>	Use the "FirstInGroup" elements for an <i>Alarm</i> to determine which <i>Alarm</i> of a group was the trigger.
<i>Client</i>	A & C Audible Sound <i>Client</i>	Use the AudibleSound <i>Property</i> and – if reported – play the sound file.
<i>Client</i>	A & C Exclusive Level <i>Client</i>	Uses Exclusive Level Alarms.
<i>Client</i>	A & C Exclusive Limit <i>Client</i>	Uses Exclusive Limit Alarms. Requires that at least one of the sub-types be used.
<i>Client</i>	A & C Exclusive Deviation <i>Client</i>	Uses Exclusive Deviation Alarms.
<i>Client</i>	A & C Exclusive RateOfChange <i>Client</i>	Uses Exclusive RateOfChange Alarms.
<i>Client</i>	A & C Non-Exclusive Level <i>Client</i>	Uses Non-Exclusive Level Alarms.
<i>Client</i>	A & C Non-Exclusive Limit <i>Client</i>	Uses Non-Exclusive Limit Alarms. Requires that at least one of the sub-types be used.
<i>Client</i>	A & C Non-Exclusive Deviation <i>Client</i>	Uses Non-Exclusive Deviation Alarms.
<i>Client</i>	A & C Non-Exclusive RateOfChange <i>Client</i>	Uses Non-Exclusive RateOfChange Alarms.
<i>Client</i>	A & C Discrete <i>Client</i>	Uses Discrete Alarm types.
<i>Client</i>	A & C OffNormal <i>Client</i>	Uses OffNormalAlarmType.
<i>Client</i>	A & C SystemOffNormal <i>Client</i>	Uses SystemOffNormalAlarmType.
<i>Client</i>	A & C Trip <i>Client</i>	Uses TripAlarmType.
<i>Client</i>	A & C Discrepancy <i>Client</i>	Uses Discrepancy Alarm type.
<i>Client</i>	A & C Dialog <i>Client</i>	Uses DialogConditionType including Respond <i>Method</i> .
<i>Client</i>	A & C CertificateExpiration <i>Client</i>	Uses CertificateExpirationAlarmType.

Table 17 describes Historical Data Access information model related items that can be profiled. The details of this model are defined in IEC 62541-11. *Servers* that support some level of historical data would expose this information model and *Clients* that utilize historical data would utilize this information model.

Table 17 – Historical Access

Category	Title	Description
Server	Historical Access Read Raw	General support for basic historical access, reading raw data using the ReadRawModifiedDetails structure. Where the time range is specified using a start time, stop time and number of values (a minimum of two of the three parameters shall be provided) and the ReadModified flag is set to False.
Server	Historical Access Data Max Nodes Read Continuation Point	Supports enough continuation points to cover the number of supported points indicated in the MaxNodesPerHistoryReadData Server OperationLimits Property for historical data access.
Server	Historical Access Time Instance	Supports reading historical data at a specified instance in time using the ReadAtTimeDetails structure.
Server	Historical Access Aggregates	Supports reading one or more Aggregates of historical values of Variables using the ReadProcessedDetails structure. At least one of the Aggregates described in IEC 62541-13 shall be supported.
Server	Historical Access Insert Value	Supports inserting historical values of Variables.
Server	Historical Access Delete Value	Supports deleting historical values of Variables.
Server	Historical Access Update Value	Supports updating historical values of Variables.
Server	Historical Access Replace Value	Supports replacing historical values of Variables.
Server	Historical Access Modified Values	Supports maintaining old values for historical data that have been updated and the retrieval of these values using the ReadRawModifiedDetails structure (ReadModified flag set to true).
Server	Historical Access Annotations	Supports the entry and retrieval of Annotations for historical data. The retrieval is accomplished using the standard historical read raw functionality (ReadRawModifiedDetails). The entry uses the standard historical update (UpdateStructureDataDetails) functionality.
Server	Historical Access ServerTimestamp	Supports providing a ServerTimestamp (as well as the default SourceTimestamp).
Server	Historical Access Structured Data Read Raw	Supports ReadRawModified historical access for structured data. Supporting the structure for an annotation is not considered supporting generic structured data.
Server	Historical Access Structured Data Time Instance	Supports historical access for structured data. Supporting ReadAtTimeDetails for structured data. Supporting the structure for an annotation is not considered supporting generic structured data.
Server	Historical Access Structured Data Insert	Supports historical access for structured data. Inserting Structured data. Supporting the structure for an annotation is not considered supporting generic structured data.
Server	Historical Access Structured Data Delete	Supports historical access for structured data. Delete of existing data. Supporting the structure for an annotation is not considered supporting generic structured data.
Server	Historical Access Structured Data Update	Supports historical access for structured data. Updates of existing data. Supporting the structure for an annotation is not considered supporting generic structured data.
Server	Historical Access Structured Data Replace	Supports replacing structured historical data. Supporting the structure for an annotation is not considered supporting generic structured data.
Server	Historical Access Structured Data Read Modified	Supports maintaining old values for historical structured data that have been updated and the retrieval of these values. Using the ReadRawModifiedDetails structure (ReadModified flag set to true) for structured data. Supporting the structure for an annotation is not considered supporting generic structured data.
Server	Historical Access Events	Supports the retrieval of historical Events using the ReadEventDetails structure. This includes support for simple filtering of Events. The Event fields that are stored are server specific, but at least the mandatory fields of BaseEventType are required.

Category	Title	Description
Server	Historical Access <i>Event</i> Max Events Read Continuation Point	Supports enough continuation points to cover the number of supported <i>Event</i> reads indicated in the <i>MaxNodesPerHistoryReadEvents</i> <i>Server</i> <i>OperationLimits</i> <i>Property</i> for Historical <i>Event</i> access.
Server	Historical Access Insert <i>Event</i>	Supports inserting historical Events.
Server	Historical Access Update <i>Event</i>	Supports updating historical Events.
Server	Historical Access Replace <i>Event</i>	Supports replacing historical Events.
Server	Historical Access Delete <i>Event</i>	Supports deleting of historical Events.
Client	Historical Access <i>Client</i> Browse	Uses the View <i>Service</i> Set to discover <i>Nodes</i> with historical data.
Client	Historical Access <i>Client</i> Read Raw	Uses the HistoryRead <i>Service</i> to read raw historical data using the ReadRawModifiedDetails Structure (ReadModified Flag set to False).
Client	Historical Access <i>Client</i> Read Modified	Uses the HistoryRead <i>Service</i> to read modified historical data using the ReadRawModifiedDetails Structure (ReadModified Flag set to True).
Client	Historical Access <i>Client</i> Read Aggregates	Uses the HistoryRead <i>Service</i> to read Aggregated historical data. This includes using at least one of the Aggregates defined in IEC 62541-13.
Client	Historical Access <i>Client</i> Structure Data Raw	Uses the HistoryRead <i>Service</i> to read raw historical data using the ReadRawModifiedDetails Structure (ReadModified Flag set to False) for structured data.
Client	Historical Access <i>Client</i> Structure Data Read Modified	Uses the HistoryRead <i>Service</i> to read modified structured historical data using the ReadRawModifiedDetails Structure (ReadModified Flag set to True).
Client	Historical Access <i>Client</i> Structure Data Insert	Uses the HistoryUpdate <i>Service</i> to insert historical data values for structured data.
Client	Historical Access <i>Client</i> Structure Data Delete	Uses the HistoryUpdate <i>Service</i> to delete historical data values for structured data.
Client	Historical Access <i>Client</i> Structure Data Update	Uses the HistoryUpdate <i>Service</i> to update historical data values for structured data.
Client	Historical Access <i>Client</i> Structure Data Replace	Uses the HistoryUpdate <i>Service</i> to replace historical data values for structured data.
Client	Historical Access <i>Client</i> Structure Data Time Instance	Reads historical data at a specified instance in time for structured data. Using the ReadAtTimeDetails structure.
Client	Historical Access <i>Client</i> Read Events	Uses the HistoryRead <i>Service</i> to read historical <i>Event</i> data using the ReadEventDetails Structure.
Client	Historical Access <i>Client</i> Event Inserts	Uses the HistoryUpdate <i>Service</i> to insert historical Events.
Client	Historical Access <i>Client</i> Event Updates	Uses the HistoryUpdate <i>Service</i> to update historical Events.
Client	Historical Access <i>Client</i> Event Replaces	Uses the HistoryUpdate <i>Service</i> to replace historical Events.
Client	Historical Access <i>Client</i> Event Deletes	Uses the HistoryUpdate <i>Service</i> to delete historical Events.
Client	Historical Access <i>Client</i> Data Insert	Uses the HistoryUpdate <i>Service</i> to insert historical data values.
Client	Historical Access <i>Client</i> Data Delete	Uses the HistoryUpdate <i>Service</i> to delete historical data values.
Client	Historical Access <i>Client</i> Data Update	Uses the HistoryUpdate <i>Service</i> to update historical data values.

Category	Title	Description
<i>Client</i>	Historical Access <i>Client</i> Data Replace	Uses the HistoryUpdate <i>Service</i> to replace historical data values.
<i>Client</i>	Historical Access <i>Client</i> Annotations	Enters and retrieves Annotations of historical data. The retrieval is accomplished using the standard historical read raw functionality (ReadRawModifiedDetails). The entry uses the standard Historical Update (UpdateStructureDataDetails) functionality.
<i>Client</i>	Historical Access <i>Client</i> Time Instance	Reads historical data at a specified instance in time using the ReadAtTimeDetails structure.
<i>Client</i>	Historical Access <i>Client</i> Server Timestamp	Uses the ServerTimestamp (as well as the default SourceTimestamp), if it is provided by the <i>Server</i> .

Table 18 describes Aggregate related items that can be profiled. *Servers* that support the Aggregates would expose this functionality and *Clients* that utilize Aggregates would implement some of this functionality.

IECNORM.COM : Click to view the full PDF of IEC 62541-7:2020 DRLV

Table 18 – Aggregates

Category	Title	Description
Server	Aggregate Master Configuration	Supports an AggregateConfigurationType <i>Object</i> as part of the HistoricalServerCapabilities (defined in IEC 62541-11).
Server	Aggregate Historical Configuration	Supports at least one AggregateConfigurationType <i>Object</i> . AggregateConfigurationType <i>Objects</i> occur as part of an HistoricalConfiguration <i>Object</i> , allowing <i>Variable</i> specific configurations.
Server	Aggregate – Interpolative	Supports the Interpolative Aggregate for Historical access.
Server	Aggregate – Average	Supports the Average Aggregate for Historical access.
Server	Aggregate – TimeAverage	Supports the TimeAverage Aggregate for Historical access.
Server	Aggregate – TimeAverage2	Supports the TimeAverage2 Aggregate for Historical access.
Server	Aggregate – Total	Supports the Total Aggregate for Historical access.
Server	Aggregate – Total2	Supports the Total2 Aggregate for Historical access.
Server	Aggregate – Minimum	Supports the Minimum Aggregate for Historical access.
Server	Aggregate – MinimumActualTime	Supports the MinimumActualTime Aggregate for Historical access.
Server	Aggregate – Minimum2	Supports the Minimum2 Aggregate for Historical access.
Server	Aggregate – MinimumActualTime2	Supports the MinimumActualTime2 Aggregate for Historical access.
Server	Aggregate – Maximum	Supports the Maximum Aggregate for Historical access.
Server	Aggregate – MaximumActualTime	Supports the MaximumActualTime Aggregate for Historical access.
Server	Aggregate – Maximum2	Supports the Maximum2 Aggregate for Historical access.
Server	Aggregate – MaximumActualTime2	Supports the MaximumActualTime2 Aggregate for Historical access.
Server	Aggregate – Range	Supports the Range Aggregate for Historical access.
Server	Aggregate – Range2	Supports the Range2 Aggregate for Historical access.
Server	Aggregate – Count	Supports the Count Aggregate for Historical access.
Server	Aggregate – DurationInStateZero	Supports the DurationInStateZero Aggregate for Historical access.
Server	Aggregate – DurationInStateNonZero	Supports the DurationInStateNonZero Aggregate for Historical access.
Server	Aggregate – NumberOfTransitions	Supports the NumberOfTransitions Aggregate for Historical access.
Server	Aggregate – Start	Supports the Start Aggregate for Historical access.
Server	Aggregate – StartBound	Supports the StartBound Aggregate for Historical access.
Server	Aggregate – End	Supports the End Aggregate for Historical access.
Server	Aggregate – EndBound	Supports the EndBound Aggregate for Historical access.
Server	Aggregate – Delta	Supports the Delta Aggregate for Historical access.
Server	Aggregate – DeltaBounds	Supports the DeltaBounds Aggregate for Historical access.
Server	Aggregate – DurationGood	Supports the DurationGood Aggregate for Historical access.
Server	Aggregate – DurationBad	Supports the DurationBad Aggregate for Historical access.
Server	Aggregate – PercentGood	Supports the PercentGood Aggregate for Historical access.
Server	Aggregate – PercentBad	Supports the PercentBad Aggregate for Historical access.
Server	Aggregate – WorstQuality	Supports the WorstQuality Aggregate for Historical access.
Server	Aggregate – WorstQuality2	Supports the WorstQuality2 Aggregate for Historical access.
Server	Aggregate – AnnotationCount	Supports the AnnotationCount Aggregate for Historical access.
Server	Aggregate – StandardDeviationSample	Supports the StandardDeviationSample Aggregate for Historical access.
Server	Aggregate – VarianceSample	Supports the VarianceSample Aggregate for Historical access.

Category	Title	Description
Server	Aggregate – StandardDeviationPopulation	Supports the StandardDeviationPopulation for Historical access.
Server	Aggregate – VariancePopulation	Supports the VariancePopulation for Historical access.
Server	Aggregate – Custom	The <i>Server</i> supports custom Aggregates for Historical access that do not have standard tests defined. These Aggregates are list as untested by this <i>ConformanceUnit</i> .
Server	Aggregate <i>Subscription</i> – Filter	Supports Aggregate subscription filters which requires at least one of the defined Aggregates is supported as defined in IEC 62541-13.
Server	Aggregate <i>Subscription</i> – Interpolative	Supports subscription filter for the Interpolative Aggregate.
Server	Aggregate <i>Subscription</i> – Average	Supports subscription filter for the Average Aggregate.
Server	Aggregate <i>Subscription</i> – TimeAverage	Supports subscription filter for the TimeAverage Aggregate.
Server	Aggregate <i>Subscription</i> – TimeAverage2	Supports subscription filter for the TimeAverage2 Aggregate.
Server	Aggregate <i>Subscription</i> – Total	Supports subscription filter for the Total Aggregate.
Server	Aggregate <i>Subscription</i> – Total2	Supports subscription filter for the Total2 Aggregate.
Server	Aggregate <i>Subscription</i> – Minimum	Supports subscription filter for the Minimum Aggregate.
Server	Aggregate <i>Subscription</i> – MinimumActualTime	Supports subscription filter for the MinimumActualTime Aggregate.
Server	Aggregate <i>Subscription</i> – Minimum2	Supports subscription filter for the Minimum2 Aggregate.
Server	Aggregate <i>Subscription</i> – MinimumActualTime2	Supports subscription filter for the MinimumActualTime2 Aggregate.
Server	Aggregate <i>Subscription</i> – Maximum	Supports subscription filter for the Maximum Aggregate.
Server	Aggregate <i>Subscription</i> – MaximumActualTime	Supports subscription filter for the MaximumActualTime Aggregate.
Server	Aggregate <i>Subscription</i> – Maximum2	Supports subscription filter for the Maximum2 Aggregate.
Server	Aggregate <i>Subscription</i> – MaximumActualTime2	Supports subscription filter for the MaximumActualTime2 Aggregate.
Server	Aggregate <i>Subscription</i> – Range	Supports subscription filter for the Range Aggregate.
Server	Aggregate <i>Subscription</i> – Range2	Supports subscription filter for the Range2 Aggregate.
Server	Aggregate <i>Subscription</i> – Count	Supports subscription filter for the Count Aggregate.
Server	Aggregate <i>Subscription</i> – DurationInStateZero	Supports subscription filter for the DurationInStateZero Aggregate.
Server	Aggregate <i>Subscription</i> – DurationInStateNonZero	Supports subscription filter for the DurationInStateNonZero Aggregate.
Server	Aggregate <i>Subscription</i> – NumberOfTransitions	Supports subscription filter for the NumberOfTransitions Aggregate.
Server	Aggregate <i>Subscription</i> – Start	Supports subscription filter for the Start Aggregate.
Server	Aggregate <i>Subscription</i> – StartBound	Supports subscription filter for the StartBound Aggregate.
Server	Aggregate <i>Subscription</i> – End	Supports subscription filter for the End Aggregate.
Server	Aggregate <i>Subscription</i> – EndBound	Supports subscription filter for the EndBound Aggregate.
Server	Aggregate <i>Subscription</i> – Delta	Supports subscription filter for the Delta Aggregate.
Server	Aggregate <i>Subscription</i> – DeltaBounds	Supports subscription filter for the DeltaBounds Aggregate.

Category	Title	Description
Server	Aggregate <i>Subscription</i> – DurationGood	Supports subscription filter for the DurationGood Aggregate.
Server	Aggregate <i>Subscription</i> – DurationBad	Supports subscription filter for the DurationBad Aggregate.
Server	Aggregate <i>Subscription</i> – PercentGood	Supports subscription filter for the PercentGood Aggregate.
Server	Aggregate <i>Subscription</i> – PercentBad	Supports subscription filter for the PercentBad Aggregate.
Server	Aggregate <i>Subscription</i> – WorstQuality	Supports subscription filter for the WorstQuality Aggregate.
Server	Aggregate <i>Subscription</i> – WorstQuality2	Supports subscription filter for the WorstQuality2 Aggregate.
Server	Aggregate <i>Subscription</i> – AnnotationCount	Supports subscription filter for the AnnotationCount Aggregate.
Server	Aggregate <i>Subscription</i> – StandardDeviationSample	Supports subscription filter for the StandardDeviationSample Aggregate.
Server	Aggregate <i>Subscription</i> – VarianceSample	Supports subscription filter for the VarianceSample Aggregate.
Server	Aggregate <i>Subscription</i> – StandardDeviationPopulation	Supports subscription filter for the StandardDeviationPopulation Aggregate.
Server	Aggregate <i>Subscription</i> – VariancePopulation	Supports subscription filter for the VariancePopulation Aggregate.
Server	Aggregate <i>Subscription</i> – Custom	The <i>Server</i> supports subscribing to custom Aggregates that do not have standard tests defined. These Aggregates are listed as untested by this <i>ConformanceUnit</i> .
Client	Aggregate – <i>Client</i> Usage	Uses Historical access to Aggregate which requires at least one of the defined Aggregates is supported as defined in IEC 62541-13.
Client	Aggregate – <i>Client</i> Interpolative	Uses Historical access to the Interpolative Aggregate.
Client	Aggregate – <i>Client</i> Average	Uses Historical access to the Average Aggregate.
Client	Aggregate – <i>Client</i> TimeAverage	Uses Historical access to the TimeAverage Aggregate.
Client	Aggregate – <i>Client</i> TimeAverage2	Uses Historical access to the TimeAverage2 Aggregate.
Client	Aggregate – <i>Client</i> Total	Uses Historical access to the Total Aggregate.
Client	Aggregate – <i>Client</i> Total2	Uses Historical access to the Total2 Aggregate.
Client	Aggregate – <i>Client</i> Minimum	Uses Historical access to the Minimum Aggregate.
Client	Aggregate – <i>Client</i> MinimumActualTime	Uses Historical access to the MinimumActualTime Aggregate.
Client	Aggregate – <i>Client</i> Minimum2	Uses Historical access to the Minimum2 Aggregate.
Client	Aggregate – <i>Client</i> MinimumActualTime2	Uses Historical access to the MinimumActualTime2 Aggregate.
Client	Aggregate – <i>Client</i> Maximum	Uses Historical access to the Maximum Aggregate.
Client	Aggregate – <i>Client</i> MaximumActualTime	Uses Historical access to the MaximumActualTime Aggregate.
Client	Aggregate – <i>Client</i> Maximum2	Uses Historical access to the Maximum2 Aggregate.
Client	Aggregate – <i>Client</i> MaximumActualTime2	Uses Historical access to the MaximumActualTime2 Aggregate.
Client	Aggregate – <i>Client</i> Range	Uses Historical access to the Range Aggregate.
Client	Aggregate – <i>Client</i> Range2	Uses Historical access to the Range2 Aggregate.
Client	Aggregate – <i>Client</i> Count	Uses Historical access to the Count Aggregate.
Client	Aggregate – <i>Client</i> DurationInStateZero	Uses Historical access to the DurationInStateZero Aggregate.
Client	Aggregate – <i>Client</i> DurationInStateNonZero	Uses Historical access to the DurationInStateNonZero Aggregate.

Category	Title	Description
<i>Client</i>	Aggregate – <i>Client</i> NumberOfTransitions	Uses Historical access to the NumberOfTransitions Aggregate.
<i>Client</i>	Aggregate – <i>Client</i> Start	Uses Historical access to the Start Aggregate.
<i>Client</i>	Aggregate – <i>Client</i> StartBound	Uses Historical access to the StartBound Aggregate.
<i>Client</i>	Aggregate – <i>Client</i> End	Uses Historical access to the End Aggregate.
<i>Client</i>	Aggregate – <i>Client</i> EndBound	Uses Historical access to the EndBound Aggregate.
<i>Client</i>	Aggregate – <i>Client</i> Delta	Uses Historical access to the Delta Aggregate.
<i>Client</i>	Aggregate – <i>Client</i> DeltaBounds	Uses Historical access to the DeltaBounds Aggregate.
<i>Client</i>	Aggregate – <i>Client</i> DurationGood	Uses Historical access to the DurationGood Aggregate.
<i>Client</i>	Aggregate – <i>Client</i> DurationBad	Uses Historical access to the DurationBad Aggregate.
<i>Client</i>	Aggregate – <i>Client</i> PercentGood	Uses Historical access to the PercentGood Aggregate.
<i>Client</i>	Aggregate – <i>Client</i> PercentBad	Uses Historical access to the PercentBad Aggregate.
<i>Client</i>	Aggregate – <i>Client</i> WorstQuality	Uses Historical access to the WorstQuality Aggregate.
<i>Client</i>	Aggregate – <i>Client</i> WorstQuality2	Uses Historical access to the WorstQuality2 Aggregate.
<i>Client</i>	Aggregate – <i>Client</i> AnnotationCount	Uses Historical access to the AnnotationCount Aggregate.
<i>Client</i>	Aggregate – <i>Client</i> StandardDeviationSample	Uses Historical access to the StandardDeviationSample Aggregate.
<i>Client</i>	Aggregate – <i>Client</i> VarianceSample	Uses Historical access to the VarianceSample Aggregate.
<i>Client</i>	Aggregate – <i>Client</i> StandardDeviationPopulation	Uses Historical access to the StandardDeviationPopulation Aggregate.
<i>Client</i>	Aggregate – <i>Client</i> VariancePopulation	Uses Historical access to the VariancePopulation Aggregate.
<i>Client</i>	Aggregate – <i>Client</i> Custom Aggregates	The <i>Client</i> can make use of all custom Aggregates in the list of Aggregates, via Historical access, exposed by the <i>Server</i> . This includes displaying or utilizing the data in some manner.
<i>Client</i>	Aggregate Subscription – <i>Client</i> Filter	Subscribes for data using Aggregate filters which requires at least one of the Aggregates defined in IEC 62541-13 is supported.
<i>Client</i>	Aggregate Subscription – <i>Client</i> Interpolative	Subscribes for data using the Interpolative Aggregate filter.
<i>Client</i>	Aggregate Subscription – <i>Client</i> Average	Subscribes for data using the Average Aggregate filter.
<i>Client</i>	Aggregate Subscription – <i>Client</i> TimeAverage	Subscribes for data using the TimeAverage Aggregate filter.
<i>Client</i>	Aggregate Subscription – <i>Client</i> TimeAverage2	Subscribes for data using the TimeAverage2 Aggregate filter.
<i>Client</i>	Aggregate Subscription – <i>Client</i> Total	Subscribes for data using the Total Aggregate filter.
<i>Client</i>	Aggregate Subscription – <i>Client</i> Total2	Subscribes for data using the Total2 Aggregate filter.
<i>Client</i>	Aggregate Subscription – <i>Client</i> Minimum	Subscribes for data using the Minimum Aggregate filter.
<i>Client</i>	Aggregate Subscription – <i>Client</i> MinimumActualTime	Subscribes for data using the MinimumActualTime Aggregate filter.
<i>Client</i>	Aggregate Subscription – <i>Client</i> Minimum2	Subscribes for data using the Minimum2 Aggregate filter.
<i>Client</i>	Aggregate Subscription – <i>Client</i> MinimumActualTime2	Subscribes for data using the MinimumActualTime2 Aggregate filter.
<i>Client</i>	Aggregate Subscription – <i>Client</i> Maximum	Subscribes for data using the Maximum Aggregate filter.
<i>Client</i>	Aggregate Subscription – <i>Client</i> MaximumActualTime	Subscribes for data using the MaximumActualTime Aggregate filter.

Category	Title	Description
<i>Client</i>	Aggregate Subscription – <i>Client</i> MaximumActualTime2	Subscribes for data using the MaximumActualTime2 Aggregate filter.
<i>Client</i>	Aggregate Subscription – <i>Client</i> Maximum2	Subscribes for data using the Maximum2 Aggregate filter.
<i>Client</i>	Aggregate Subscription – <i>Client</i> Range	Subscribes for data using the Range Aggregate filter.
<i>Client</i>	Aggregate Subscription – <i>Client</i> Range2	Subscribes for data using the Range2 Aggregate filter.
<i>Client</i>	Aggregate Subscription – <i>Client</i> Count	Subscribes for data using the Count Aggregate filter.
<i>Client</i>	Aggregate Subscription – <i>Client</i> DurationInStateZero	Subscribes for data using the DurationInStateZero Aggregate filter.
<i>Client</i>	Aggregate Subscription – <i>Client</i> DurationInStateNonZero	Subscribes for data using the DurationInStateNonZero Aggregate filter.
<i>Client</i>	Aggregate Subscription – <i>Client</i> NumberOfTransitions	Subscribes for data using the NumberOfTransitions Aggregate filter.
<i>Client</i>	Aggregate Subscription – <i>Client</i> Start	Subscribes for data using the Start Aggregate filter.
<i>Client</i>	Aggregate Subscription – <i>Client</i> StartBound	Subscribes for data using the StartBound Aggregate filter.
<i>Client</i>	Aggregate Subscription – <i>Client</i> End	Subscribes for data using the End Aggregate filter.
<i>Client</i>	Aggregate Subscription – <i>Client</i> EndBound	Subscribes for data using the EndBound Aggregate filter.
<i>Client</i>	Aggregate Subscription – <i>Client</i> Delta	Subscribes for data using the Delta Aggregate filter.
<i>Client</i>	Aggregate Subscription – <i>Client</i> DeltaBounds	Subscribes for data using the DeltaBounds Aggregate filter.
<i>Client</i>	Aggregate Subscription – <i>Client</i> DurationGood	Subscribes for data using the DurationGood Aggregate filter.
<i>Client</i>	Aggregate Subscription – <i>Client</i> DurationBad	Subscribes for data using the DurationBad Aggregate filter.
<i>Client</i>	Aggregate Subscription – <i>Client</i> PercentGood	Subscribes for data using the PercentGood Aggregate filter.
<i>Client</i>	Aggregate Subscription – <i>Client</i> PercentBad	Subscribes for data using the PercentBad Aggregate filter.
<i>Client</i>	Aggregate Subscription – <i>Client</i> WorstQuality	Subscribes for data using the WorstQuality Aggregate filter.
<i>Client</i>	Aggregate Subscription – <i>Client</i> WorstQuality2	Subscribes for data using the WorstQuality2 Aggregate filter.
<i>Client</i>	Aggregate Subscription – <i>Client</i> AnnotationCount	Subscribes for data using the AnnotationCount Aggregate filter.
<i>Client</i>	Aggregate Subscription – <i>Client</i> StandardDeviationSample	Subscribes for data using the StandardDeviationSample Aggregate filter.
<i>Client</i>	Aggregate Subscription – <i>Client</i> VarianceSample	Subscribes for data using the VarianceSample Aggregate filter.
<i>Client</i>	Aggregate Subscription – <i>Client</i> StandardDeviationPopulation	Subscribes for data using the StandardDeviationPopulation Aggregate filter.
<i>Client</i>	Aggregate Subscription – <i>Client</i> VariancePopulation	Subscribes for data using the VariancePopulation Aggregate filter.
<i>Client</i>	Aggregate Subscription – <i>Client</i> Custom Aggregates	The <i>Client</i> supports subscribing to all custom Aggregates in the list of Aggregates exposed by the <i>Server</i> . This includes displaying or utilizing the data in some manner.

Table 19 describes auditing related items that can be profiled. Most full function *Servers* would support these features, although some resource constrained *Servers* may not provide this functionality. *Clients* that are security aware or are used to support security logging would support these features.

Table 19 – Auditing

Category	Title	Description
<i>Server</i>	Auditing Base	Support AuditEvents. The list of supported AuditEvents shall be verified during certification testing and will be shown in the certification test result. Base AuditEvents are defined in IEC 62541-3 and in IEC 62541-5.
<i>Client</i>	Auditing <i>Client</i> Audit ID	<i>Client</i> supports generating AuditEvents ids and providing them to <i>Servers</i> .
<i>Client</i>	Auditing <i>Client</i> Subscribes	The <i>Client</i> supports subscribing for AuditEvents and storing / processing them in a secure manner.

Table 20 describes Redundancy related items that are profiled. *Servers* that support redundancy would support appropriate *ConformanceUnits* based on the type of redundancy they support. *Clients* that are capable of handling redundancy would support the appropriate *ConformanceUnits* based of the type of redundancy they support.

Table 20 – Redundancy

Category	Title	Description
<i>Server</i>	Redundancy <i>Server</i>	Supports <i>Server</i> based redundancy.
<i>Server</i>	Redundancy <i>Server</i> Transparent	Supports transparent <i>Server</i> redundancy.
<i>Client</i>	Redundancy <i>Client</i>	<i>Client</i> supports <i>Client</i> redundancy. <i>Clients</i> that support <i>Client</i> redundancy can failover to another <i>Client</i> (requires some out of band communication).
<i>Client</i>	Redundancy <i>Client</i> Switch	<i>Clients</i> supporting this <i>ConformanceUnit</i> monitor the redundancy status for non-transparent redundancy <i>Servers</i> and switch to the backup <i>Server</i> when they recognize a change in server status.

Table 21 describes items for a Global *Discovery Server* (GDS). *Servers* that act as a GDS would support these *ConformanceUnits*.

Table 21 – Global Discovery Server

Category	Title	Description
Global Directory Service	GDS Application Directory	Supports the Directory <i>Object</i> with all Methods like RegisterApplication and QueryServers.
Global Directory Service	GDS Query Applications	Supports the QueryApplications <i>Method</i> on the Directory <i>Object</i> specified in IEC 62541-12.
Global Directory Service	GDS LDS-ME Connectivity	The GDS can be configured to use specific LDS-ME installations for semi-automatic application registration for all <i>Servers</i> on a subnet.
Global Directory Service	GDS <i>Certificate</i> Manager Pull Model	This Conformance Unit requires support of the complete Information Model and <i>Services</i> for <i>Certificate</i> management including the Pull Model as specified in IEC 62541-12.
Global Directory Service	GDS <i>Certificate</i> Manager Push Model	This Conformance Unit requires use of the complete Information model and <i>Services</i> for the <i>Certificate</i> management Push Model as specified in IEC 62541-12.
Global Directory Service	GDS Key Credential Service Pull Model	This Conformance Unit requires support of the complete Information Model and <i>Services</i> for KeyCredential Pull Management as specified in IEC 62541-12.
Global Directory Service	GDS Key Credential Service Push Model	This Conformance Unit requires use of the complete Information model and <i>Services</i> for KeyCredential Push Management as specified in IEC 62541-12.
Global Directory Service	GDS Authorization Service Server	This Conformance Unit requires support of AuthorizationServiceType Objects as specified in IEC 62541-12. UA Clients use the RequestAccessToken Method on these Objects to request an Access Token from an Identity Provider.

5.5 Miscellaneous

Table 22 describes miscellaneous *ConformanceUnits*.

Table 22 – Miscellaneous

Category	Title	Description
<i>Server</i>	Documentation – Supported <i>Profiles</i>	The documentation includes a description of the profiles supported by the product. This description includes the level of Certification testing the product has passed.
<i>Server</i>	Documentation – Multiple Languages	The documentation is available in multiple languages. The results of this conformance unit include the list of supported languages.
<i>Server</i>	Documentation – Users Guide	The application includes documentation that describes the available functionality provided by the application. For <i>Servers</i> it includes a summary of all functionality provided by the <i>Server</i> .
<i>Server</i>	Documentation – On-line	The documentation provided by the application is available in electronic format as part of the application. The electronic documentation, could be a WEB page, installed document or CD/DVD, but in all case it can be accessed from the application or from a link installed with the application.
<i>Server</i>	Documentation – Installation	The application includes installation instructions that are sufficient to easily install the application. This includes descriptions of any and all possible configuration items. Instructions for loading or configuring security related items such as Application Instance <i>Certificates</i> .
<i>Server</i>	Documentation – Trouble Shooting Guide	The application includes documentation that describes typical problems a user may encounter and actions that the user could perform to resolve the problem. It could also describe tip, tricks or other actions that could help a user diagnose or fix a problem. It could also describe tools or other items that can be used in diagnosing or repairing problems. The actual Trouble Shooting Guide can be part of other documentation, but should be complete enough to provide useful information to a novice user.
<i>Client</i>	Documentation <i>Client</i> – Supported <i>Profiles</i>	The documentation includes a description of the profiles supported by the product. This description includes any software certificates that describes the level of Certification testing the product has passed.
<i>Client</i>	Documentation <i>Client</i> – Multiple Languages	The documentation is available in multiple languages. The results of this conformance unit include the list of supported languages.
<i>Client</i>	Documentation <i>Client</i> – Users Guide	The application includes documentation that describes the available functionality provided by the application. For client applications, this includes any operator restrictions or general functionality that the client application makes use of.
<i>Client</i>	Documentation <i>Client</i> – On-line	The documentation provided by the application is available in electronic format as part of the application. The electronic documentation could be a WEB page, installed document or CD/DVD, but in all cases it can be accessed from the application or from a link installed with the application.
<i>Client</i>	Documentation <i>Client</i> – Installation	The application includes installation instructions that are sufficient to easily install the application. This includes descriptions of any and all possible configuration items. Instructions for loading or configuring security related items such as Application Instance <i>Certificates</i> .
<i>Client</i>	Documentation <i>Client</i> – Trouble Shooting Guide	The application includes documentation that describes typical problems a user may encounter and actions that the user could perform to resolve the problem. It could also describe tips, tricks or other actions that could help a user diagnose or fix a problem. It could also describe tools or other items that can be used in diagnosing or repairing problems. The actual Trouble Shooting Guide can be part of other documentation, but should be complete enough to provide useful information to a novice user.
Security	Best Practice – Timeouts	The user is able to configure reasonable timeouts for Secure Channels, sessions and subscriptions to limit Denial of <i>Service</i> and resource consumption issues (see IEC 62541-2 for additional details).
Security	Best Practice – Strict <i>Message</i> Handling	The application assures that messages that are illegally or incorrectly formed are rejected with appropriate error code or appropriate actions as specified in IEC 62541-4 and IEC 62541-6.
Security	Best Practice – Random Numbers	All random numbers that are required for security use appropriate cryptographic library based random number generators.
Security	Best Practice – Administrative Access	The <i>Server</i> and <i>Client</i> allow for appropriate restriction of access to administrative personnel. This includes multiple levels of administrative access on platforms that support multiple administrative roles (such as Windows or Linux).

Category	Title	Description
Security	Best Practice – Alarm Handling	A <i>Server</i> should restrict critical alarm functionality to users that have the appropriate rights to perform these actions. This would include disabling or alarms, shelving of alarms and generation of dialog messages. It would also include other security related functionality such maintaining appropriate timeouts for shelving and dialogs and preventing an overload of dialog messages.
Security	Best Practice – Audit Events	Subscriptions for Audit Events are restricted to authorized personnel. A <i>Server</i> may also reject a <i>Subscription</i> for Audit Events that is not over a Secure Channel if one is available.
Security	Best Practice – Audit Events <i>Client</i>	Audit tracking system connects to a <i>Server</i> using a Secure Channel and under the appropriate administrative rights to allow access to Audit Events.

6 Profiles

6.1 Overview

This clause includes a listing of the categories that a *Profile* can be grouped into, a list of named *Profiles* and the detailed listing of each *Profile* including directly defined *ConformanceUnits* and any sub *Profiles* that are included in the *Profile*.

6.2 Profile list

Table 23 lists *Profiles*. The *Profile* table is ordered by *Profile* category and then alphabetically by the name of the *Profile*. The table includes a list of categories the *Profile* is associated with and a URI. The URI is used to uniquely identify a *Profile*. The URI shall be able to be used to access the information provided in this document with regard to the given *Profile* in an on-line display.

An application (*Client* or *Server*) shall implement all of the *ConformanceUnits* in a *Profile* in order to be compliant with the *Profile*. Some *Profiles* contain optional *ConformanceUnits*. An optional *ConformanceUnit* means that an application has the option to not support the *ConformanceUnit*. However, if supported, the application shall pass all tests associated with the *ConformanceUnit*. For example, some *ConformanceUnits* require specific information model items to be available. They are, therefore, listed as optional in order to allow for the information model items to be omitted. If a *Server* desires to be listed as supporting the optional *ConformanceUnit* then it shall include any required information model items in the configuration provided for certification testing. The test result that is generated by the certification testing lists all optional *ConformanceUnits* and whether they are supported or not by the tested UA application. Some *ConformanceUnits* also include lists of supported DataTypes or optional Subtypes; the lists are handled in the same manner as optional *ConformanceUnits*. All reporting requirements for optional *ConformanceUnits* also apply to these lists of supported DataTypes or Subtypes.

Table 23 – Profile list

Profile	Related Category	URI
Core 2017 Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/Core2017Facet
Sessionless Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/SessionLess
Reverse Connect Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/ReverseConnect
Base Server Behaviour Facet	Server	http://opcfoundation.org/UA-Profile/Server/Behaviour
Request State Change Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/RequestStateChange
Subnet Discovery Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/SubnetDiscovery
Global Certificate Management Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/GlobalCertificateManagement
Authorization Service Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/AuthorizationServiceConfiguration
KeyCredential Service Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/KeyCredentialManagement
Attribute WriteMask Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/AttributeWriteMask
File Access Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/FileAccess
Documentation Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/Documentation
Embedded DataChange Subscription Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/EmbeddedDataChangeSubscription
Standard DataChange Subscription 2017 Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/StandardDataChangeSubscription2017
Enhanced DataChange Subscription 2017 Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/EnhancedDataChangeSubscription2017
Durable Subscription Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/DurableSubscription
Data Access Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/DataAccess
ComplexType 2017 Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/ComplexTypes2017
Standard Event Subscription Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/StandardEventSubscription
Address Space Notifier Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/AddressSpaceNotifier
A & C Base Condition Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/ACBaseCondition
A & C Refresh2 Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/ACRefresh2
A & C Address Space Instance Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/ACAddressSpaceInstance
A & C Enable Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/ACEnable
A & C AlarmMetrics Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/ACAlarmMetrics
A & C Alarm Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/ACAlarm
A & C Acknowledgeable Alarm Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/ACAckAlarm
A & C Exclusive Alarming Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/ACExclusiveAlarming
A & C Non-Exclusive Alarming Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/ACNon-ExclusiveAlarming
A & C Previous Instances Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/ACPreviousInstances
A & C Dialog Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/ACDialog

Profile	Related Category	URI
A & C CertificateExpiration Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/ACCertificateExpiration
A & E Wrapper Facet	Server	http://opcfoundation.org/UA-Profile/Server/AEWrapper
Method Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/Methods
Auditing Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/Auditing
Node Management Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/NodeManagement
User Role Base Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/UserRoleBase
User Role Management Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/UserRoleManagement
State Machine Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/StateMachine
Client Redundancy Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/ClientRedundancy
Redundancy Transparent Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/TransparentRedundancy
Redundancy Visible Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/VisibleRedundancy
Historical Raw Data Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/HistoricalRawData
Historical Aggregate Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/AggregateHistorical
Historical Data AtTime Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/HistoricalDataAtTime
Historical Access Modified Data Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/HistoricalModifiedData
Historical Annotation Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/HistoricalAnnotation
Historical Data Insert Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/HistoricalDataInsert
Historical Data Update Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/HistoricalDataUpdate
Historical Data Replace Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/HistoricalDataReplace
Historical Data Delete Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/HistoricalDataDelete
Historical Access Structured Data Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/HistoricalStructuredData
Base Historical Event Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/BaseHistoricalEvent
Historical Event Update Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/HistoricalEventUpdate
Historical Event Replace Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/HistoricalEventReplace
Historical Event Insert Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/HistoricalEventInsert
Historical Event Delete Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/HistoricalEventDelete
Aggregate Subscription Server Facet	Server	http://opcfoundation.org/UA-Profile/Server/AggregateSubscription
Nano Embedded Device 2017 Server Profile	Server	http://opcfoundation.org/UA-Profile/Server/NanoEmbeddedDevice2017
Micro Embedded Device 2017 Server Profile	Server	http://opcfoundation.org/UA-Profile/Server/MicroEmbeddedDevice2017

Profile	Related Category	URI
Embedded 2017 UA Server Profile	Server	http://opcfoundation.org/UA-Profile/Server/EmbeddedUA2017
Standard 2017 UA Server Profile	Server	http://opcfoundation.org/UA-Profile/Server/StandardUA2017
Core 2017 <i>Client</i> Facet	<i>Client</i>	http://opcfoundation.org/UA-Profile/Client/Core2017
Sessionless <i>Client</i> Facet	<i>Client</i>	http://opcfoundation.org/UA-Profile/Client/SessionLess
Reverse Connect Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/ReverseConnect
Base Client Behaviour Facet	Client	http://opcfoundation.org/UA-Profile/Client/Behaviour
Discovery Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/Discovery
Subnet Discovery Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/SubnetDiscovery
Global Discovery Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/GlobalDiscovery
Global Certificate Management Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/GlobalCertificateManagement
KeyCredential Service Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/KeyCredentialManagement
Access Token Request Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/AccessTokenRequest
AddressSpace Lookup Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/AddressSpaceLookup
Request State Change Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/RequestStateChange
File Access Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/FileAccess
Entry Level Support 2015 Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/EntryLevelSupport2015
Multi-Server Client Connection Facet	Client	http://opcfoundation.org/UA-Profile/Client/MultiServer
Documentation – Client	Client	http://opcfoundation.org/UA-Profile/Client/Documentation
Attribute Read Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/AttributeRead
Attribute Write Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/AttributeWrite
DataChange Subscriber Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/DataChangeSubscriber
Durable Subscription Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/DurableSubscription
DataAccess Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/DataAccess
Event Subscriber Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/EventSubscriber
Base Event Processing Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/BaseEventProcessing
Notifier and Source Hierarchy Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/NotifierAndSourceHierarchy
A & C Base Condition Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/ACBaseCondition
A & C Refresh2 Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/ACRefresh2
A & C Address Space Instance Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/ACAddressSpaceInstance
A & C Enable Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/ACEnable
A & C AlarmMetrics Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/ACAlarmMetrics
A & C Alarm Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/ACAlarm
A & C Exclusive Alarming Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/ACExclusiveAlarming
A & C Non-Exclusive Alarming Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/ACNon-ExclusiveAlarming

Profile	Related Category	URI
A & C Previous Instances Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/ACPreviousInstances
A & C Dialog Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/ACDialog
A & C CertificateExpiration Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/ACCertificateExpiration
A & E Proxy Facet	Client	http://opcfoundation.org/UA-Profile/Client/AEProxy
Method Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/Method
Auditing Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/Auditing
Node Management Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/NodeManagement
Advanced Type Programming Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/TypeProgramming
User Role Management Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/UserRoleManagement
State Machine Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/StateMachine
Diagnostic Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/Diagnostic
Redundant Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/Redundancy
Redundancy Switch Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/RedundancySwitch
Historical Access Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/HistoricalAccess
Historical Data AtTime Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/HistoricalAccessAtTime
Historical Aggregate Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/HistoricalAccessAggregate
Historical Annotation Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/HistoricalAnnotation
Historical Access Modified Data Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/HistoricalAccessModifiedData
Historical Data Insert Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/HistoricalInsertData
Historical Data Update Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/HistoricalUpdateData
Historical Data Replace Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/HistoricalReplaceData
Historical Data Delete Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/HistoricalDeleteData
Historical Access Client Server Timestamp Facet	Client	http://opcfoundation.org/UA-Profile/Client/HistoricalServerTimeStamp
Historical Structured Data Access Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/HistoricalAccessStructuredData
Historical Structured Data AtTime Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/HistoricalAtTimeStructuredData
Historical Structured Data Modified Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/HistoricalModifiedStructuredData
Historical Structured Data Insert Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/HistoricalInsertStructuredData
Historical Structured Data Update Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/HistoricalUpdateStructuredData
Historical Structured Data Replace Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/HistoricalReplaceStructuredData
Historical Structured Data Delete Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/HistoricalDeleteStructuredData

Profile	Related Category	URI
Historical Events Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/HistoricalEvents
Historical Event Insert Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/HistoricalInsertEvents
Historical Event Update Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/HistoricalUpdateEvents
Historical Event Replace Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/HistoricalReplaceEvents
Historical Event Delete Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/HistoricalDeleteEvents
Aggregate Subscriber Client Facet	Client	http://opcfoundation.org/UA-Profile/Client/AggregateSubscriber
Standard UA Client Profile	Client	http://opcfoundation.org/UA-Profile/Client/Standard
Standard UA Client 2017 Profile	Client	http://opcfoundation.org/UA-Profile/Client/Standard2017
UA-TCP UA-SC UA-Binary	Transport	http://opcfoundation.org/UA-Profile/Transport/uatcp-uasc-uabinary
HTTPS UA-Binary	Transport	http://opcfoundation.org/UA-Profile/Transport/https-uabinary
HTTPS UA-XML	Transport	http://opcfoundation.org/UA-Profile/Transport/https-uasoapxml
HTTPS UA-JSON	Transport	http://opcfoundation.org/UA-Profile/Transport/https-uajson
WSS UA-SC UA-Binary	Transport	http://opcfoundation.org/UA-Profile/Transport/wss-uasc-uabinary
WSS UA-JSON	Transport	http://opcfoundation.org/UA-Profile/Transport/wss-uajson
Security User Access Control Full	Security, Server	http://opcfoundation.org/UA-Profile/Security/UserAccessFull
Security User Access Control Base	Security, Server	http://opcfoundation.org/UA-Profile/Security/UserAccessBase
Security Time Synchronization	Security	http://opcfoundation.org/UA-Profile/Security/TimeSync
Best Practice – Audit Events	Security, Server	http://opcfoundation.org/UA-Profile/Security/BestPracticeAuditEvents
Best Practice – Alarm Handling	Security, Server	http://opcfoundation.org/UA-Profile/Security/BestPracticeAlarmHandling
Best Practice – Random Numbers	Security	http://opcfoundation.org/UA-Profile/Security/BestPracticeRandomNumbers
Best Practice – Timeouts	Security	http://opcfoundation.org/UA-Profile/Security/BestPracticeTimeouts
Best Practice – Administrative Access	Security	http://opcfoundation.org/UA-Profile/Security/BestPracticeAdministrativeAccess
Best Practice – Strict Message Handling	Security, Server	http://opcfoundation.org/UA-Profile/Security/BestPracticeStrictMessage
Best Practice – Audit Events Client	Client, Security	http://opcfoundation.org/UA-Profile/Security/BestPracticeAuditEventsClient
TransportSecurity – TLS 1.2	Security	http://opcfoundation.org/UA-Profile/TransportSecurity/TLS-1-2
TransportSecurity – TLS 1.2 with PFS	Security	http://opcfoundation.org/UA-Profile/TransportSecurity/TLS-1-2-PFS
SecurityPolicy – None	Security	http://opcfoundation.org/UA/SecurityPolicy#None
SecurityPolicy [A] – Aes128-Sha256-RsaOaep	Security	http://opcfoundation.org/UA/SecurityPolicy#Aes128_Sha256_RsaOaep
SecurityPolicy [B] – Basic256Sha256	Security	http://opcfoundation.org/UA/SecurityPolicy#Basic256Sha256

Profile	Related Category	URI
SecurityPolicy – Aes256-Sha256-RsaPss	Security	http://opcfoundation.org/UA/SecurityPolicy#Aes256_Sha256_RsaPss
User Token – Anonymous Facet	Security	http://opcfoundation.org/UA-Profile/Security/UserToken/Anonymous
User Token – User Name Password Server Facet	Security, Server	http://opcfoundation.org/UA-Profile/Security/UserToken/Server/UserNamePassword
User Token – X509 Certificate Server Facet	Security, Server	http://opcfoundation.org/UA-Profile/Security/UserToken/Server/X509Certificate
User Token – Issued Token Server Facet	Security, Server	http://opcfoundation.org/UA-Profile/Security/UserToken/Server/IssuedToken
User Token – Issued Token Windows Server Facet	Security, Server	http://opcfoundation.org/UA-Profile/Security/UserToken/Server/IssuedTokenWindows
User Token – JWT Server Facet	Security	http://opcfoundation.org/UA-Profile/Security/UserToken/Server/JsonWebToken
User Token – User Name Password Client Facet	Client, Security	http://opcfoundation.org/UA-Profile/Security/UserToken/Client/UserNamePassword
User Token – X509 Certificate Client Facet	Client, Security	http://opcfoundation.org/UA-Profile/Security/UserToken/Client/X509Certificate
User Token – Issued Token Client Facet	Client, Security	http://opcfoundation.org/UA-Profile/Security/UserToken/Client/IssuedToken
User Token – Issued Token Windows Client Facet	Client, Security	http://opcfoundation.org/UA-Profile/Security/UserToken/Client/IssuedTokenWindows
User Token – JWT Client Facet	Security	http://opcfoundation.org/UA-Profile/Security/UserToken/Client/JsonWebToken
Global Discovery Server 2017 Profile	Global Directory Service, Server	http://opcfoundation.org/UA-Profile/Server/GlobalDiscovery2017
Global Discovery and Certificate Mgmt 2017 Server	Global Directory Service, Server	http://opcfoundation.org/UA-Profile/Server/GlobalDiscoveryAndCertificateManagement2017
Global Certificate Management Client 2017 Profile	Client, Global Directory Service	http://opcfoundation.org/UA-Profile/Client/GlobalCertificateManagement2017
Global Service Authorization Request Server Facet	Global Directory Service	http://opcfoundation.org/UA-Profile/Server/GlobalServiceAuthorization
Global Service KeyCredential Pull Facet	Global Directory Service	http://opcfoundation.org/UA-Profile/Server/GlobalServiceKeyCredentials
Global Service KeyCredential Push Facet	Global Directory Service	http://opcfoundation.org/UA-Profile/Client/GlobalServiceKeyCredentials

The contents of each of the listed *Profiles* will be described in a subclause of 6.6 and where the title is equal to the *Profile* name. Each table may contain references to additional *Profiles* and or *ConformanceUnits*. If a *Profile* is referenced it means that it is completely included. The *ConformanceUnits* are referenced using their name and conformance group. For the details of the *ConformanceUnit* the reader should examine the *ConformanceUnit* details in clause 5.

6.3 Conventions for Profile definitions

Profiles have the following naming conventions:

- *Profiles* intended for OPC UA *Servers* contain the term *Server* in their titles,
- *Profiles* intended for OPC UA *Clients* contain the term *Client* in their titles
- The term *Facet* in the title of a *Profile* indicates that this *Profile* is expected to be part of another larger *Profile* or concerns a specific aspect of OPC UA. *Profiles* with the term *Facet* in their title are expected to be combined with other *Profiles* to define the complete functionality of an OPC UA *Server* or *Client*.

6.4 Profile versioning

Versioning of *Profile* is accomplished with a naming convention. Whenever a *profile* is revised, the year of the new revision is added to the name. Example:

Version 1	Core <i>Server</i> Facet
Version 2	Core 2017 <i>Server</i> Facet

6.5 Applications

A vendor that is developing a UA application, whether it is a *Server* application or a *Client* application, shall review the list of available *Profiles*. From this list the vendor shall select the *Profiles* that include the functionality required by the application. Typically this will be multiple *Profiles*. Conformance to a single *Profile* may not yield a complete application. In most cases multiple *Profiles* are needed to yield a useful application. All *Servers* and *Clients* shall support at least a core *Profile* (Core *Server* Facet or Core *Client* Facet) and at least one Transport *Profile*

For example an HMI *Client* application may choose to support the "Core *Client* Facet", the "UA-TCP UA-SC UA-Binary" *Profile*, the "Data Access *Client* Facet", the "DataChange Subscriber *Client* Facet" and the "Attribute Write *Client* Facet". If the *Client* is to be *TestLab* tested then it would also support "Base *Client* Behaviour" *Profile*. This list of *Profiles* would allow the *Client* to communicate with an OPC UA *Server* using UA-TCP/UA Security/UA binary. It would be able to subscribe for data, write to data and would support the DA data model. It would also follow the best practice guideline for behaviour. Figure 2 illustrates the *Profile* hierarchy that this application may contain. This figure is only an illustration and the represented *Profiles* may change.

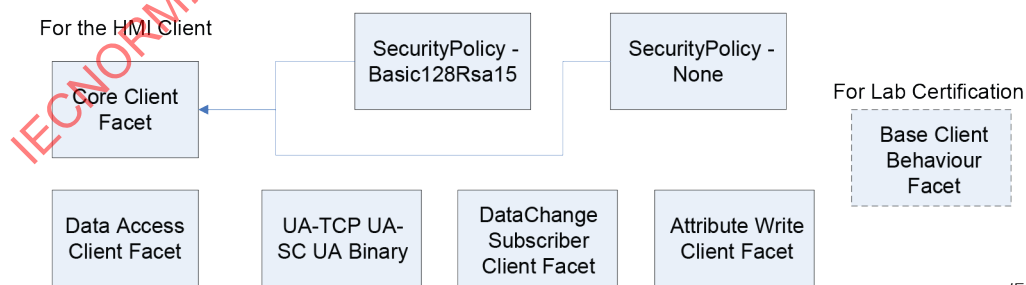


Figure 2 – HMI Client sample

All *Clients* should take into account the types of *Servers* and *Server Profiles* that they are targeted to support. Some *Servers* might not support *Subscriptions* and *Clients* should be able to fall back to Read *Services*.

A special case is a generic *Client* that is designed to communicate with a large number of *Servers* and therefore able to perform a broad range of functionality. "Standard UA *Client Profile*" has been defined for this kind of *Clients*.

Many *Clients*, however, will be specialized and do not need all of the functionality in the "Standard UA *Client Profile*" and thus would only support the limited set of functionality they require. A trend *Client*, for example, would only need functionality to subscribe to or read data.

Another example is an embedded device OPC UA *Server* application that may choose to support "Embedded UA *Server Profile*" and the "DataAccess *Server Facet Profile*". This device would be a resource constrained device that would support UA-TCP, UA-Security, UA Binary encoding, data subscriptions and the DA data model. It may not support the optional attribute write. Figure 3 illustrates the hierarchy that this application may contain. This figure is just an illustration and the represented *Profiles* may change.

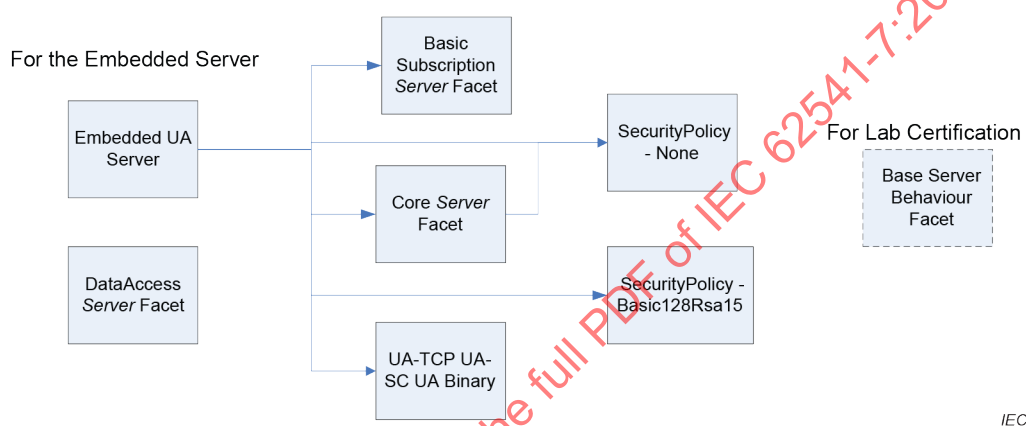


Figure 3 – Embedded Server sample

Another simple system *Server* application may choose to support: "Standard UA *Server Profile*" and the "DataAccess *Server Facet Profile*". If the *Server* is to be lab tested then it would also support "Base Server Behaviour" *Profile*. This device would be a mid-level OPC UA *Server* that would support all that the embedded *Server* in the previous example supported and it would add support for an enhance level of the subscription service and support for writes. Figure 4 illustrates the hierarchy that this application may contain. This figure is just an illustration and the represented *Profile* may change.

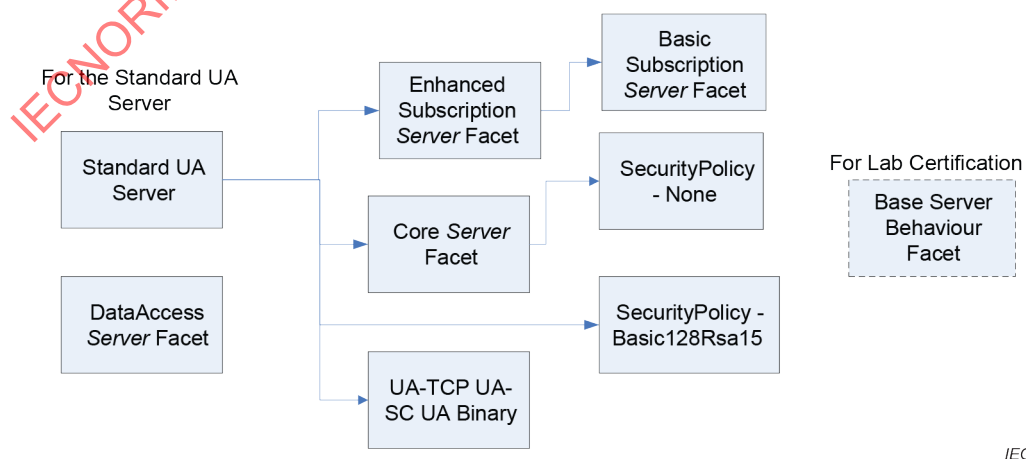


Figure 4 – Standard UA Server sample

If the example HMI *Client* were to connect to either of the example *Servers*, it may have to adjust its behavior based on the *Profile* reported by the respective *Servers*. If the HMI *Client* were communicating with the embedded device it would not be able to perform any write operations. It may also have to limit the number of subscriptions or sessions based on the performance limits of the *Server*. If the HMI *Client* is connected to the Standard *Server* it would be able to open additional windows, have higher limits on performance related items and it would be able to allow writes.

6.6 Profile tables

6.6.1 General

Subclauses 6.6.2 to 6.6.186 describe *Profiles* in a tabular format.

Each table contains three columns. The first column is a description of the conformance group that the *ConformanceUnit* is part of. This allows the reader to easily find the *ConformanceUnit*. This column may also state "*Profile*" in which case the listed item is not a *ConformanceUnit*, but an included *Profile*. The second column is a brief description of the *ConformanceUnit* or included *Profile*. The last column indicates if the *ConformanceUnit* is optional or required.

6.6.2 Core Server Facet

Core Server Facet has been deprecated as it is superseded with Core 2017 Server Facet in v1.04.

6.6.3 Core 2017 Server Facet

Table 24 describes the details of the Core 2017 *Server Facet*. This Facet defines the core functionality required for any UA *Server* implementation. The core functionality includes the ability to discover endpoints, establish secure communication channels, create Sessions, browse the *AddressSpace* and read and/or write to *Attributes* of *Nodes*. The key requirements are: support for a single *Session*, support for the *Server* and *Server Capabilities Object*, all mandatory *Attributes* for *Nodes* in the *AddressSpace*, and authentication with Username and Password. For broad applicability, it is recommended that *Servers* support multiple transport and security *Profiles*. This Facet supersedes the "Core Server Facet".

Table 24 – Core 2017 Server Facet

Group	Conformance Unit / Profile Title	Optional
<i>Profile</i>	SecurityPolicy – None	False
<i>Profile</i>	User Token – User Name Password <i>Server</i> Facet	False
Address Space Model	Address Space Atomicity	False
Address Space Model	Address Space Base	False
Address Space Model	Address Space Full Array Only	False
<i>Attribute Services</i>	<i>Attribute</i> Read	False
<i>Attribute Services</i>	<i>Attribute</i> Write Index	True
<i>Attribute Services</i>	<i>Attribute</i> Write Values	True
Base Information	Base Info Core Structure	False
Base Information	Base Info Estimated Return Time	True
Base Information	Base Info OptionSet	True
Base Information	Base Info Placeholder Modelling Rules	True
Base Information	Base Info Selection List	True
Base Information	Base Info <i>Server</i> Capabilities	True
Base Information	Base Info ValueAsText	True
<i>Discovery Services</i>	<i>Discovery</i> Find <i>Servers</i> Self	False
<i>Discovery Services</i>	<i>Discovery</i> Get Endpoints	False
Security	Security Administration	True
Security	Security Role <i>Server</i> Authorization	True
<i>Session Services</i>	<i>Session</i> Base	False
<i>Session Services</i>	<i>Session</i> General <i>Service</i> Behaviour	False
<i>Session Services</i>	<i>Session</i> Minimum 1	False
<i>View Services</i>	<i>View</i> Basic	False
<i>View Services</i>	<i>View</i> Minimum Continuation Point 01	False
<i>View Services</i>	<i>View</i> RegisterNodes	False
<i>View Services</i>	<i>View</i> TranslateBrowsePath	False

6.6.4 Sessionless Server Facet

Table 25 describes the details of the Sessionless *Server* Facet. Defines the use of Sessionless *Service* invocation in a *Server*.

Table 25 – Sessionless Server Facet

Group	Conformance Unit / Profile Title	Optional
<i>Discovery Services</i>	<i>Discovery</i> Get Endpoints SessionLess	False
<i>Session Services</i>	<i>Session</i> Sessionless Invocation	False

6.6.5 Reverse Connect Server Facet

Table 26 describes the details of the Reverse Connect *Server* Facet. This Facet defines support of reverse connectivity in a *Server*. Usually, a connection is opened by the *Client* before starting the UA-specific handshake. This will fail, however, when *Servers* are behind firewalls with no open ports to connect to. In the reverse connectivity scenario, the *Server* opens the connection and starts with a ReverseHello message requesting that the *Client* establish a Secure Channel using this connection.

Table 26 – Reverse Connect Server Facet

Group	Conformance Unit / Profile Title	Optional
Protocol and Encoding	Protocol Reverse Connect Server	False

6.6.6 Base Server Behaviour Facet

Table 27 describes the details of the Base Server Behaviour Facet. This Facet defines best practices for the configuration and management of Servers when they are deployed in a production environment. It provides the ability to enable or disable certain protocols and to configure the *Discovery Server* and specify where this Server shall be registered.

Table 27 – Base Server Behaviour Facet

Group	Conformance Unit / Profile Title	Optional
<i>Discovery Services</i>	<i>Discovery</i> Configuration	False
Protocol and Encoding	Protocol Configuration	False
Security	Security Administration	False
Security	Security Administration – XML Schema	False
Security	Security <i>Certificate</i> Administration	False

6.6.7 Request State Change Server Facet

Table 28 describes the details of the Request State Change Server Facet. This Facet specifies the support of the RequestServerStateChange *Method*.

Table 28 – Request State Change Server Facet

Group	Conformance Unit / Profile Title	Optional
Base Information	Base Info RequestServerStateChange <i>Method</i>	False

6.6.8 Subnet Discovery Server Facet

Table 29 describes the details of the Subnet *Discovery Server* Facet. Support of this Facet enables discovery of the Server on a subnet using mDNS. This functionality is only applicable when Servers do not register with an LDS.

Table 29 – Subnet Discovery Server Facet

Group	Conformance Unit / Profile Title	Optional
<i>Discovery Services</i>	<i>Discovery Server</i> Announcement using mDNS	False

6.6.9 Global Certificate Management Server Facet

Table 30 describes the details of the Global *Certificate* Management Server Facet. This Facet defines the capability to interact with a Global *Certificate* Management Server to obtain an initial or renewed *Certificate* and Trust Lists.

Table 30 – Global Certificate Management Server Facet

Group	Conformance Unit / Profile Title	Optional
Security	Push Model for Global <i>Certificate</i> and TrustList Management	False

6.6.10 Authorization Service Server Facet

Table 31 describes the details of the Authorization *Service Server* Facet. This Facet defines the support for configuring the necessary information to validate access tokens when presented by a Client during session establishment. Access Tokens are issued by Authorization *Services*.

Table 31 – Authorization Service Server Facet

Group	Conformance Unit / Profile Title	Optional
Security	Authorization Service Configuration Server	False

6.6.11 KeyCredential Service Server Facet

Table 32 describes the details of the KeyCredential *Service Server* Facet. This Facet defines the capability to interact with a KeyCredential *Service* to obtain KeyCredentials. For example KeyCredentials are needed to access an Authorization *Service* or a Broker. The KeyCredential *Service* is typically part of a system-wide tool, like a GDS that also manages Applications, Access Tokens, and *Certificates*.

Table 32 – KeyCredential Service Server Facet

Group	Conformance Unit / Profile Title	Optional
Security	Push Model for KeyCredential <i>Service</i>	False

6.6.12 Attribute WriteMask Server Facet

Table 33 describes the details of the Attribute WriteMask *Server* Facet. This Facet defines the capability to update characteristics of individual *Nodes* in the *AddressSpace* by allowing writing to *Node Attributes*. It requires support for authenticating user access as well as providing information related to access rights in the *AddressSpace* and actually restricting the access rights as described.

Table 33 – Attribute WriteMask Server Facet

Group	Conformance Unit / Profile Title	Optional
<i>Profile</i>	Security User Access Control Base	False
Address Space Model	Address Space UserWriteMask	False
Address Space Model	Address Space UserWriteMask Multilevel	True
Address Space Model	Address Space WriteMask	False

6.6.13 File Access Server Facet

Table 34 describes the details of the File Access *Server* Facet. This Facet specifies the support of exposing File information via the defined FileType. This includes reading of file as well as optionally writing of file data.

Table 34 – File Access Server Facet

Group	Conformance Unit / Profile Title	Optional
Base Information	Base Info FileType Base	False
Base Information	Base Info FileType Write	True

6.6.14 Documentation Server Facet

Table 35 describes the details of the Documentation *Server Facet*. This Facet defines a list of user documentation that a server application should provide.

Table 35 – Documentation Server Facet

Group	Conformance Unit / Profile Title	Optional
Miscellaneous	Documentation – Installation	False
Miscellaneous	Documentation – Multiple Languages	True
Miscellaneous	Documentation – On-line	True
Miscellaneous	Documentation – Supported Profiles	True
Miscellaneous	Documentation – Trouble Shooting Guide	True
Miscellaneous	Documentation – Users Guide	False

6.6.15 Embedded DataChange Subscription Server Facet

Table 36 describes the details of the Embedded DataChange *Subscription Server Facet*. This Facet specifies the minimum level of support for data change notifications within subscriptions. It includes limits which minimize memory and processing overhead required to implement the Facet. This Facet includes functionality to create, modify and delete Subscriptions and to add, modify and remove Monitored Items. As a minimum for each *Session*, *Servers* shall support one *Subscription* with up to two items. In addition, support for two parallel Publish requests is required. This Facet is geared for a platform such as the one provided by the Micro Embedded Device *Server Profile* in which memory is limited and needs to be managed.

Table 36 – Embedded DataChange Subscription Server Facet

Group	Conformance Unit / Profile Title	Optional
Monitored Item Services	Monitor Basic	False
Monitored Item Services	Monitor Items 2	False
Monitored Item Services	Monitor QueueSize_1	False
Monitored Item Services	Monitor Value Change	False
Subscription Services	Subscription Basic	False
Subscription Services	Subscription Minimum 1	False
Subscription Services	Subscription Publish Discard Policy	False
Subscription Services	Subscription Publish Min 02	False

6.6.16 Standard DataChange Subscription Server Facet

Standard DataChange Subscription Server Facet has been deprecated as it is superseded with Standard DataChange Subscription 2017 Server Facet in v1.04.

6.6.17 Standard DataChange Subscription 2017 Server Facet

Table 37 describes the details of the Standard DataChange *Subscription* 2017 *Server* Facet. This Facet specifies the standard support of subscribing to data changes and extends features and limits defined by the Embedded Data Change *Subscription* Facet. See *ConformanceUnits* for these limits. Note that the *Method Call Service* is only required for the *Methods* defined in this Facet. This Facet supersedes the "Standard DataChange *Subscription Server* Facet".

Table 37 – Standard DataChange Subscription 2017 Server Facet

Group	Conformance Unit / Profile Title	Optional
<i>Profile</i>	Embedded DataChange <i>Subscription</i> Server Facet	False
Base Information	Base Info GetMonitoredItems <i>Method</i>	False
Base Information	Base Info ResendData <i>Method</i>	False
<i>Method Services</i>	<i>Method</i> Call	False
Monitored Item <i>Services</i>	Monitor Items 10	False
Monitored Item <i>Services</i>	Monitor Items 100	False
Monitored Item <i>Services</i>	Monitor MinQueueSize_02	False
Monitored Item <i>Services</i>	Monitor Triggering	False
Monitored Item <i>Services</i>	Monitored Items Deadband Filter	False
<i>Subscription Services</i>	<i>Subscription</i> Minimum 02	False
<i>Subscription Services</i>	<i>Subscription</i> Publish Min 05	False

6.6.18 Enhanced DataChange Subscription Server Facet

Enhanced DataChange *Subscription* Server Facet has been deprecated as it is superseded with Enhanced DataChange Subscription 2017 Server Facet in v1.04.

6.6.19 Enhanced DataChange Subscription 2017 Server Facet

Table 38 describes the details of the Enhanced DataChange *Subscription* 2017 *Server* Facet. This Facet specifies an enhanced support of subscribing to data changes. It is part of the Standard UA *Server* 2017 *Profile*. This Facet increases the limits defined by the Standard Data Change *Subscription* 2017 *Server* Facet.

Table 38 – Enhanced DataChange Subscription 2017 Server Facet

Group	Conformance Unit / Profile Title	Optional
<i>Profile</i>	Standard DataChange <i>Subscription</i> 2017 <i>Server</i> Facet	False
Monitored Item <i>Services</i>	Monitor Items 500	False
Monitored Item <i>Services</i>	Monitor MinQueueSize_05	False
<i>Subscription Services</i>	<i>Subscription</i> Minimum 05	False
<i>Subscription Services</i>	<i>Subscription</i> Publish Min 10	False

6.6.20 Durable Subscription Server Facet

Table 39 describes the details of the Durable *Subscription* Server Facet. This Facet specifies support of durable storage of data and events even when *Clients* are disconnected. This Facet implies support of any of the DataChange or *Event Subscription* Facets.

Table 39 – Durable Subscription Server Facet

Group	Conformance Unit / Profile Title	Optional
<i>Subscription Services</i>	<i>Subscription Durable</i>	False
<i>Subscription Services</i>	<i>Subscription Durable StorageLevel High</i>	True
<i>Subscription Services</i>	<i>Subscription Durable StorageLevel Medium</i>	True
<i>Subscription Services</i>	<i>Subscription Durable StorageLevel Small</i>	True

6.6.21 Data Access Server Facet

Table 40 describes the details of the Data Access *Server* Facet. This Facet specifies the support for an Information Model used to provide industrial automation data. This model defines standard structures for analog and discrete data items and their quality of service. This Facet extends the Core *Server* Facet which includes support of the basic *AddressSpace* behaviour.

Table 40 – Data Access Server Facet

Group	Conformance Unit / Profile Title	Optional
Data Access	Data Access AnalogItems	True
Data Access	Data Access ArrayItemType	True
Data Access	Data Access Complex Number	True
Data Access	Data Access DataItems	False
Data Access	Data Access DoubleComplex Number	True
Data Access	Data Access MultiState	True
Data Access	Data Access MultiStateValueDiscrete	True
Data Access	Data Access PercentDeadband	True
Data Access	Data Access Semantic Changes	True
Data Access	Data Access TwoState	True

6.6.22 ComplexType Server Facet

ComplexType *Server* Facet has been deprecated as it is superseded with ComplexType 2017 *Server* Facet in v1.04.

6.6.23 ComplexType 2017 Server Facet

Table 41 describes the details of the ComplexType 2017 *Server* Facet. This Facet extends the Core *Server* Facet to include *Variables* with structured data, i.e. data that are composed of multiple elements such as a structure and where the individual elements are exposed as component variables. Support of this Facet requires the implementation of structured DataTypes and *Variables* that make use of these DataTypes. The Read, Write and Subscriptions service set shall support the encoding and decoding of these structured DataTypes. As an option the *Server* can also support alternate encodings, such as an XML encoding when the binary protocol is currently used and vice-versa.

Table 41 – ComplexType 2017 Server Facet

Group	Conformance Unit / Profile Title	Optional
Address Space Model	Address Space DataTypeDefinition Attribute	False
Attribute Services	Attribute Alternate Encoding	True
Attribute Services	Attribute Read Complex	False
Attribute Services	Attribute Write Complex	False
Monitored Item Services	Monitor Alternate Encoding	True
Monitored Item Services	Monitor Complex Value	True

6.6.24 Standard Event Subscription Server Facet

Table 42 describes the details of the Standard *Event Subscription Server Facet*. This Facet specifies the standard support for subscribing to events and is intended to supplement any of the FullFeatured *Profiles*. Support of this Facet requires the implementation of *Event Types* representing the Events that the *Server* can report and their specific fields. It also requires at least the *Server Object* to have the *EventNotifier Attribute* set. It includes the *Services* to Create, Modify and Delete Subscriptions and to Add, Modify and Remove Monitored Items for *Object Nodes* with an "*EventNotifier Attribute*". Creating a monitoring item may include a filter that includes SimpleAttribute FilterOperands and a select list of Operators. The operators include: Equals, IsNull, GreaterThan, LessThan, GreaterThanOrEqualTo, LessThanOrEqualTo, Like, Not, Between, InList, And, Or, Cast, BitwiseAnd, BitwiseOr and TypeOf. Support of more complex filters is optional. This Facet has been updated to include several optional Base Information *ConformanceUnits*. These *ConformanceUnits* are optional to allow for backward compatibility, in the future these optional *ConformanceUnits* will become required, and so it is highly recommended that all servers support them.

Table 42 – Standard Event Subscription Server Facet

Group	Conformance Unit / Profile Title	Optional
Address Space Model	Address Space Events	False
Base Information	Base Info Device Failure	True
Base Information	Base Info EventQueueOverflow EventType	True
Base Information	Base Info Progress Events	True
Base Information	Base Info SemanticChange	True
Base Information	Base Info System Status	True
Base Information	Base Info System Status Underlying System	True
Monitored Item Services	Monitor Basic	False
Monitored Item Services	Monitor Complex Event Filter	True
Monitored Item Services	Monitor Events	False
Monitored Item Services	Monitor Items 10	False
Monitored Item Services	Monitor QueueSize_ServerMax	False
Subscription Services	Subscription Basic	False
Subscription Services	Subscription Minimum 02	False
Subscription Services	Subscription Publish Discard Policy	False
Subscription Services	Subscription Publish Min 05	False

6.6.25 Address Space Notifier Server Facet

Table 43 describes the details of the Address Space Notifier Server Facet. This Facet requires the support of a hierarchy of *Object Nodes* that are notifiers and *Nodes* that are event sources. The hierarchy is commonly used as a way to organize a plant into areas that can be managed by different operators.

Table 43 – Address Space Notifier Server Facet

Group	Conformance Unit / Profile Title	Optional
Address Space Model	Address Space Notifier Hierarchy	False
Address Space Model	Address Space Source Hierarchy	False

6.6.26 A & C Base Condition Server Facet

Table 44 describes the details of the A & C Base Condition Server Facet. This Facet requires basic support for *Conditions*. Information about *Conditions* is provided through *Event* notifications and thus this Facet builds upon the Standard *Event Subscription Server Facet*. *Conditions* that are in an "interesting" state (as defined by the *Server*) can be refreshed using the *Refresh Method*, which requires support for the *Method Server Facet*. Optionally the server may also provide support for *Condition* classes

Table 44 – A & C Base Condition Server Facet

Group	Conformance Unit / Profile Title	Optional
Profile	Method Server Facet	False
Profile	Standard Event Subscription Server Facet	False
Alarms and Conditions	A & C Basic	False
Alarms and Conditions	A & C Condition Sub-Classes	True
Alarms and Conditions	A & C ConditionClasses	True
Alarms and Conditions	A & C Refresh	False

6.6.27 A & C Refresh2 Server Facet

Table 45 describes the details of the A & C Refresh2 Server Facet. This Facet enhances the A & C Base Condition Server Facet with support of the *ConditionRefresh2 Method*.

Table 45 – A & C Refresh2 Server Facet

Group	Conformance Unit / Profile Title	Optional
Profile	A & C Base Condition Server Facet	False
Alarms and Conditions	A & C Refresh2	False

6.6.28 A & C Address Space Instance Server Facet

Table 46 describes the details of the A & C Address Space Instance Server Facet. This Facet specifies the support required for a *Server* to expose *Alarms* and *Conditions* in its *AddressSpace*. This includes the A & C *AddressSpace* information model.

Table 46 – A & C Address Space Instance Server Facet

Group	Conformance Unit / <i>Profile</i> Title	Optional
<i>Alarms and Conditions</i>	A & C Instances	False

6.6.29 A & C Enable Server Facet

Table 47 describes the details of the A & C Enable *Server* Facet. This Facet requires the enabling and disabling of *Conditions*. This Facet builds upon the A&C Base *Condition Server* Facet. Enabling and disabling also requires that instances of these ConditionTypes exist in the *AddressSpace* since the enable *Method* can only be invoked on an instance of the *Condition*

Table 47 – A & C Enable Server Facet

Group	Conformance Unit / <i>Profile</i> Title	Optional
<i>Profile</i>	A & C Base <i>Condition Server</i> Facet	False
<i>Alarms and Conditions</i>	A & C Enable	False
<i>Alarms and Conditions</i>	A & C Instances	False

6.6.30 A & C AlarmMetrics Server Facet

Table 48 describes the details of the A & C AlarmMetrics *Server* Facet. This Facet requires support for AlarmMetrics. AlarmMetrics expose status and potential issues in the alarm system. A *Server* can provide these metrics at various levels (operator station, plant area, overall system etc.).

Table 48 – A & C AlarmMetrics Server Facet

Group	Conformance Unit / <i>Profile</i> Title	Optional
<i>Alarms and Conditions</i>	A & C <i>Alarm</i> Metrics	False

6.6.31 A & C Alarm Server Facet

Table 49 describes the details of the A & C *Alarm Server* Facet. This Facet requires support for *Alarms*. *Alarms* extend the ConditionType by adding an Active state which indicates when something in the system requires attention by an Operator. This Facet builds upon the A&C Base *Condition Server* Facet. This facet requires that discrete AlarmTypes be supported, it also allows for optional support of shelving, alarm comments and other discrete AlarmTypes such as Trip or Off-Normal.

Table 49 – A & C Alarm Server Facet

Group	Conformance Unit / Profile Title	Optional
<i>Profile</i>	A & C Base <i>Condition</i> Server Facet	False
<i>Alarms and Conditions</i>	A & C <i>Alarm</i>	False
<i>Alarms and Conditions</i>	A & C Audible Sound	True
<i>Alarms and Conditions</i>	A & C Comment	True
<i>Alarms and Conditions</i>	A & C Discrepancy	True
<i>Alarms and Conditions</i>	A & C Discrete	False
<i>Alarms and Conditions</i>	A & C First in Group <i>Alarm</i>	True
<i>Alarms and Conditions</i>	A & C OffNormal	True
<i>Alarms and Conditions</i>	A & C On-Off Delay	True
<i>Alarms and Conditions</i>	A & C Out Of Service	True
<i>Alarms and Conditions</i>	A & C Re-Alarming	True
<i>Alarms and Conditions</i>	A & C Shelving	True
<i>Alarms and Conditions</i>	A & C Silencing	True
<i>Alarms and Conditions</i>	A & C Suppression	True
<i>Alarms and Conditions</i>	A & C Suppression by Operator	True
<i>Alarms and Conditions</i>	A & C SystemOffNormal	True
<i>Alarms and Conditions</i>	A & C Trip	True

6.6.32 A & C Acknowledgeable Alarm Server Facet

Table 50 describes the details of the A & C Acknowledgeable *Alarm* Server Facet. This Facet requires support for Acknowledgement of active *Alarms*. This Facet builds upon the A & C *Alarm* Server Facet. Acknowledgement requires support of the Acknowledge *Method* and the Acknowledged state. Support of the Confirmed state and the Confirm *Method* is optional.

Table 50 – A & C Acknowledgeable Alarm Server Facet

Group	Conformance Unit / Profile Title	Optional
<i>Profile</i>	A & C <i>Alarm</i> Server Facet	False
<i>Alarms and Conditions</i>	A & C Acknowledge	False
<i>Alarms and Conditions</i>	A & C Confirm	True

6.6.33 A & C Exclusive Alarming Server Facet

Table 51 describes the details of the A & C Exclusive Alarming *Server* Facet. This Facet requires support for *Alarms* with multiple sub-states that identify different limit *Conditions*. This facet builds upon the A&C *Alarm* Server Facet. The term exclusive means only one sub-state can be active at a time. For example, a temperature exceeds the HighHigh limit the associated exclusive LevelAlarm will be in the HighHigh sub-state and not in the High sub-state. This Facet requires that a *Server* support at least one of the optional *Alarm* models: Limit, RateOfChange or Deviation.

Table 51 – A & C Exclusive Alarming Server Facet

Group	Conformance Unit / Profile Title	Optional
Profile	A & C Alarm Server Facet	False
Alarms and Conditions	A & C Exclusive Deviation	True
Alarms and Conditions	A & C Exclusive Level	True
Alarms and Conditions	A & C Exclusive Limit	False
Alarms and Conditions	A & C Exclusive RateOfChange	True

6.6.34 A & C Non-Exclusive Alarming Server Facet

Table 52 describes the details of the A & C Non-Exclusive Alarming Server Facet. This Facet requires support for *Alarms* with multiple sub-states that identify different limit *Conditions*. This Facet builds upon the A&C Alarm Server Facet. The term non-exclusive means more than one sub-state can be active at a time. For example, if a temperature exceeds the HighHigh limit the associated non-exclusive LevelAlarm will be in both the High and the HighHigh sub-state. This Facet requires that a server support at least one of the optional alarm models: Limit, RateOfChange or Deviation.

Table 52 – A & C Non-Exclusive Alarming Server Facet

Group	Conformance Unit / Profile Title	Optional
Profile	A & C Alarm Server Facet	False
Alarms and Conditions	A & C Non-Exclusive Deviation	True
Alarms and Conditions	A & C Non-Exclusive Level	True
Alarms and Conditions	A & C Non-Exclusive Limit	False
Alarms and Conditions	A & C Non-Exclusive RateOfChange	True

6.6.35 A & C Previous Instances Server Facet

Table 53 describes the details of the A & C Previous Instances Server Facet. This Facet requires support for *Conditions* with previous states that still require action on the part of the operator. This Facet builds upon the A&C Base Condition Server Facet. A common use case for this Facet is a safety critical system that requires that all *Alarms* be acknowledged even if it the original problem goes away and the *Alarm* returns to the inactive state. In these cases, the previous state with active *Alarm* is still reported by the *Server* until the Operator acknowledges it. When a *Condition* has previous states it will produce events with different Branch identifiers. When previous state no longer needs attention the branch will disappear.

Table 53 – A & C Previous Instances Server Facet

Group	Conformance Unit / Profile Title	Optional
Profile	A & C Base Condition Server Facet	False
Alarms and Conditions	A & C Branch	False

6.6.36 A & C Dialog Server Facet

Table 54 describes the details of the A & C Dialog Server Facet. This Facet requires support of Dialog *Conditions*. This Facet builds upon the A & C BaseCondition Server Facet Dialogs are ConditionTypes used to request user input. They are typically used when a *Server* has entered some state that requires intervention by a *Client*. For example, a *Server* monitoring a paper machine indicates that a roll of paper has been wound and is ready for inspection. The *Server* would activate a Dialog *Condition* indicating to the user that an inspection is required.

Once the inspection has taken place the user responds by informing the *Server* of an accepted or unaccepted inspection allowing the process to continue.

Table 54 – A & C Dialog Server Facet

Group	Conformance Unit / Profile Title	Optional
<i>Profile</i>	A & C Base <i>Condition</i> Server Facet	False
<i>Alarms and Conditions</i>	A & C Dialog	False

6.6.37 A & C CertificateExpiration Server Facet

Table 55 describes the details of the A & C CertificateExpiration *Server* Facet. This Facet requires support of the CertificateExpirationAlarmType. It is used to inform *Clients* when the *Server's* *Certificate* is within the defined expiration period.

Table 55 – A & C CertificateExpiration Server Facet

Group	Conformance Unit / Profile Title	Optional
<i>Profile</i>	A & C Base <i>Condition</i> Server Facet	False
<i>Alarms and Conditions</i>	A & C Acknowledge	False
<i>Alarms and Conditions</i>	A & C <i>Alarm</i>	False
<i>Alarms and Conditions</i>	A & C CertificateExpiration	False
<i>Alarms and Conditions</i>	A & C Comment	True
<i>Alarms and Conditions</i>	A & C Confirm	True
<i>Alarms and Conditions</i>	A & C Shelving	True

6.6.38 A & E Wrapper Facet

Table 56 describes the details of the A & E Wrapper Facet. This Facet specifies the requirements for a UA *Server* that wraps an OPC *Alarm & Event* (AE) *Server* (COM). This *Profile* identifies the sub-set of the UA *Alarm & Condition* model which is provided by the COM OPC AE specification. It is intended to provide guidance to developers who are creating servers that front end existing applications. It is important to note that some OPC A&E COM *Servers* may not support all of the functionality provided by an OPC UA A&C server, in these cases similar functionality maybe available via some non-OPC interface. For example if an A&E COM server does not support sending *Alarm* Acknowledgement messages to the system that it is obtaining alarm information from, this functionality may be available via some out of scope features in the underlying *Alarm* system. Another possibility is that the underlying system does not require acknowledgements or automatically acknowledges the alarm.

Table 56 – A & E Wrapper Facet

Group	Conformance Unit / <i>Profile</i> Title	Optional
Address Space Model	Address Space Events	False
Address Space Model	Address Space Notifier Hierarchy	False
Address Space Model	Address Space Source Hierarchy	False
<i>Alarms and Conditions</i>	A & C Acknowledge	False
<i>Alarms and Conditions</i>	A & C Alarm	False
<i>Alarms and Conditions</i>	A & C Basic	False
<i>Alarms and Conditions</i>	A & C ConditionClasses	False
<i>Alarms and Conditions</i>	A & C Refresh	False
<i>Alarms and Conditions</i>	A & E Wrapper Mapping	False
Monitored Item <i>Services</i>	Monitor Basic	False
Monitored Item <i>Services</i>	Monitor Complex <i>Event</i> Filter	False
Monitored Item <i>Services</i>	Monitor Events	False
Monitored Item <i>Services</i>	Monitor Items 2	False
Monitored Item <i>Services</i>	Monitor QueueSize_ServerMax	False
<i>Subscription Services</i>	<i>Subscription</i> Basic	False
<i>Subscription Services</i>	<i>Subscription</i> Minimum 1	False
<i>Subscription Services</i>	<i>Subscription</i> Publish Discard Policy	False
<i>Subscription Services</i>	<i>Subscription</i> Publish Min 02	False

6.6.39 Method Server Facet

Table 57 describes the details of the *Method Server* Facet. This Facet specifies the support of *Method* invocation via the Call service. Methods are "lightweight" functions which are similar to the methods of a class found in any object-oriented programming language. A *Method* can have its scope bounded by an owning *Object* or an owning *ObjectType*. Methods with an *ObjectType* as their scope are similar to static methods in a class.

Table 57 – Method Server Facet

Group	Conformance Unit / <i>Profile</i> Title	Optional
Address Space Model	Address Space <i>Method</i>	False
<i>Method Services</i>	<i>Method</i> Call	False

6.6.40 Auditing Server Facet

Table 58 describes the details of the Auditing *Server* Facet. This Facet requires the support of Auditing which includes the Standard *Event Subscription Server* Facet. Support of this Facet requires that Audit Events be produced when a client performs some action to change the state of the server, such as changing the *AddressSpace*, inserting or updating a value etc. The auditEntryId passed by the *Client* is a field contained in every Audit *Event* and allows actions to be traced across multiple systems. The Audit *Event* Types and their fields shall be exposed in the *Server's AddressSpace*

Table 58 – Auditing Server Facet

Group	Conformance Unit / Profile Title	Optional
Profile	Standard Event Subscription Server Facet	False
Auditing	Auditing Base	False

6.6.41 Node Management Server Facet

Table 59 describes the details of the *Node Management Server* Facet. This Facet requires the support of the *Services* that allow the *Client* to add, modify and delete *Nodes* in the *AddressSpace*. These *Services* provide an interface which can be used to configure *Servers*. This means all changes to the *AddressSpace* are expected to persist even after the *Client* has disconnected from the *Server*.

Table 59 – Node Management Server Facet

Group	Conformance Unit / Profile Title	Optional
Address Space Model	Address Space Base	False
Base Information	Base Info Model Change	False
Base Information	Base Info Type System	False
Node Management Services	Node Management Add Node	False
Node Management Services	Node Management Add Ref	False
Node Management Services	Node Management Delete Node	False
Node Management Services	Node Management Delete Ref	False

6.6.42 User Role Base Server Facet

Table 60 describes the details of the *User Role Base Server* Facet. This Facet defines support of the OPC UA Information Model to expose configured user roles and permissions.

Table 60 – User Role Base Server Facet

Group	Conformance Unit / Profile Title	Optional
Security	Security Role Server Base	False

6.6.43 User Role Management Server Facet

Table 61 describes the details of the *User Role Management Server* Facet. This Facet defines support of the OPC UA approach to manage user roles and permissions and to grant access to *Nodes* and *Services* based on the assigned roles and permissions.

Table 61 – User Role Management Server Facet

Group	Conformance Unit / Profile Title	Optional
Profile	User Role Base Server Facet	False
Security	Security Role Server DefaultRolePermissions	False
Security	Security Role Server IdentityManagement	False
Security	Security Role Server Management	False
Security	Security Role Server Restrict Applications	True
Security	Security Role Server Restrict Endpoints	True
Security	Security Role Server RolePermissions	True
Security	Security Role Well Known	False

6.6.44 State Machine Server Facet

Table 62 describes the details of the State Machine Server Facet. This Facet defines support of StateMachines based on the types in IEC 62541-5.

Table 62 – State Machine Server Facet

Group	Conformance Unit / Profile Title	Optional
Base Information	Base Info Available States and Transitions	True
Base Information	Base Info Finite State Machine Instance	True
Base Information	Base Info State Machine Instance	False

6.6.45 Client Redundancy Server Facet

Table 63 describes the details of the Client Redundancy Server Facet. This Facet defines the Server actions that are required for support of redundant Clients. Support of this Facet requires the implementation of the TransferSubscriptions Service which allows the transfer of Subscriptions from one Client's Session to another Client's Session.

Table 63 – Client Redundancy Server Facet

Group	Conformance Unit / Profile Title	Optional
Subscription Services	Subscription Transfer	False

6.6.46 Redundancy Transparent Server Facet

Table 64 describes the details of the Redundancy Transparent Server Facet. This Facet requires support for transparent redundancy. If Servers implement transparent redundancy then the failover from one Server to another is transparent to the Client such that the Client is unaware that a failover has occurred; the Client does not need to do anything at all to keep data flowing. This type of redundancy is usually a hardware solution.

Table 64 – Redundancy Transparent Server Facet

Group	Conformance Unit / Profile Title	Optional
Redundancy	Redundancy Server Transparent	False

6.6.47 Redundancy Visible Server Facet

Table 65 describes the details of the Redundancy Visible Server Facet. This Facet specifies the support for non-transparent redundancy. Failover for this type of redundancy requires the *Client* to monitor *Server* status and to switch to a backup *Server* if it detects a failure. The *Server* shall expose the methods of failover it supports (cold, warm or hot). The failover method tells the *Client* what it shall do when connecting to a *Server* and when a failure occurs. Cold redundancy requires a *Client* to reconnect to a backup *Server* after the initial *Server* has failed. Warm redundancy allows a *Client* to connect to multiple *Servers*, but only one *Server* will be providing values. In hot redundancy multiple *Servers* are able to provide data and a *Client* can connect to multiple *Servers* for the data.

Table 65 – Redundancy Visible Server Facet

Group	Conformance Unit / Profile Title	Optional
Redundancy	Redundancy Server	False

6.6.48 Historical Raw Data Server Facet

Table 66 describes the details of the Historical Raw Data Server Facet. This Facet defines the basic functionality when supporting historical data access for raw data.

Table 66 – Historical Raw Data Server Facet

Group	Conformance Unit / Profile Title	Optional
Attribute Services	Attribute Historical Read	False
Historical Access	Historical Access Data Max Nodes Read Continuation Point	False
Historical Access	Historical Access Read Raw	False
Historical Access	Historical Access ServerTimestamp	True

6.6.49 Historical Aggregate Server Facet

Table 67 describes the details of the Historical Aggregate Server Facet. This Facet indicates that the server supports aggregate processing to produce derived values from raw historical data.

Table 67 – Historical Aggregate Server Facet

Group	Conformance Unit / Profile Title	Optional
Aggregates	Aggregate – AnnotationCount	True
Aggregates	Aggregate – Average	True
Aggregates	Aggregate – Count	True
Aggregates	Aggregate – Custom	True
Aggregates	Aggregate – Delta	True
Aggregates	Aggregate – DeltaBounds	True
Aggregates	Aggregate – DurationBad	True
Aggregates	Aggregate – DurationGood	True
Aggregates	Aggregate – DurationInStateNonZero	True
Aggregates	Aggregate – DurationInStateZero	True
Aggregates	Aggregate – End	True
Aggregates	Aggregate – EndBound	True
Aggregates	Aggregate – Interpolative	True
Aggregates	Aggregate – Maximum	True
Aggregates	Aggregate – Maximum2	True
Aggregates	Aggregate – MaximumActualTime	True
Aggregates	Aggregate – MaximumActualTime2	True
Aggregates	Aggregate – Minimum	True
Aggregates	Aggregate – Minimum2	True
Aggregates	Aggregate – MinimumActualTime	True
Aggregates	Aggregate – MinimumActualTime2	True
Aggregates	Aggregate – NumberOfTransitions	True
Aggregates	Aggregate – PercentBad	True
Aggregates	Aggregate – PercentGood	True
Aggregates	Aggregate – Range	True
Aggregates	Aggregate – Range2	True
Aggregates	Aggregate – StandardDeviationPopulation	True
Aggregates	Aggregate – StandardDeviationSample	True
Aggregates	Aggregate – Start	True
Aggregates	Aggregate – StartBound	True
Aggregates	Aggregate – TimeAverage	True
Aggregates	Aggregate – TimeAverage2	True
Aggregates	Aggregate – Total	True
Aggregates	Aggregate – Total2	True
Aggregates	Aggregate – VariancePopulation	True
Aggregates	Aggregate – VarianceSample	True
Aggregates	Aggregate – WorstQuality	True
Aggregates	Aggregate – WorstQuality2	True
Aggregates	Aggregate Historical Configuration	True
Aggregates	Aggregate Master Configuration	False
Attribute Services	Attribute Historical Read	False
Historical Access	Historical Access Aggregates	False
Historical Access	Historical Access Data Max Nodes Read Continuation Point	False

6.6.50 Historical Data AtTime Server Facet

Table 68 describes the details of the Historical Data AtTime Server Facet. This Facet indicates that the historical Server supports reading data by specifying specific timestamps.

Table 68 – Historical Data AtTime Server Facet

Group	Conformance Unit / <i>Profile</i> Title	Optional
<i>Attribute Services</i>	<i>Attribute</i> Historical Read	False
Historical Access	Historical Access Data Max <i>Nodes</i> Read Continuation Point	False
Historical Access	Historical Access Time Instance	False

6.6.51 Historical Access Modified Data Server Facet

Table 69 describes the details of the Historical Access Modified Data *Server* Facet. This Facet defines support of reading modified historical values (values that were modified or inserted).

Table 69 – Historical Access Modified Data Server Facet

Group	Conformance Unit / <i>Profile</i> Title	Optional
<i>Attribute Services</i>	<i>Attribute</i> Historical Read	False
Historical Access	Historical Access Modified Values	False

6.6.52 Historical Annotation Server Facet

Table 70 describes the details of the Historical Annotation *Server* Facet. This Facet defines support for the storage and retrieval of annotations for historical data.

Table 70 – Historical Annotation Server Facet

Group	Conformance Unit / <i>Profile</i> Title	Optional
<i>Attribute Services</i>	<i>Attribute</i> Historical Read	False
<i>Attribute Services</i>	<i>Attribute</i> Historical Update	False
Historical Access	Historical Access Annotations	False

6.6.53 Historical Data Insert Server Facet

Table 71 describes the details of the Historical Data Insert *Server* Facet. This Facet includes Historical Data Insert functionality.

Table 71 – Historical Data Insert Server Facet

Group	Conformance Unit / <i>Profile</i> Title	Optional
<i>Attribute Services</i>	<i>Attribute</i> Historical Update	False
Historical Access	Historical Access Insert Value	False
Historical Access	Historical Access ServerTimestamp	True

6.6.54 Historical Data Update Server Facet

Table 72 describes the details of the Historical Data Update *Server* Facet. This Facet includes Historical Data Update functionality.

Table 72 – Historical Data Update Server Facet

Group	Conformance Unit / Profile Title	Optional
<i>Attribute Services</i>	<i>Attribute</i> Historical Update	False
Historical Access	Historical Access ServerTimestamp	True
Historical Access	Historical Access Update Value	False

6.6.55 Historical Data Replace Server Facet

Table 73 describes the details of the Historical Data Replace Server Facet. This Facet includes Historical Data Replace functionality.

Table 73 – Historical Data Replace Server Facet

Group	Conformance Unit / Profile Title	Optional
<i>Attribute Services</i>	<i>Attribute</i> Historical Update	False
Historical Access	Historical Access Replace Value	False
Historical Access	Historical Access ServerTimestamp	True

6.6.56 Historical Data Delete Server Facet

Table 74 describes the details of the Historical Data Delete Server Facet. This Facet includes Historical Data Delete functionality.

Table 74 – Historical Data Delete Server Facet

Group	Conformance Unit / Profile Title	Optional
<i>Attribute Services</i>	<i>Attribute</i> Historical Update	False
Historical Access	Historical Access Delete Value	False

6.6.57 Historical Access Structured Data Server Facet

Table 75 describes the details of the Historical Access Structured Data Server Facet. This Facet indicates that the Server supports storage and retrieval of structured values for all supported access types. If a listed access type is supported then the corresponding optional *ConformanceUnit* shall be supported.

Table 75 – Historical Access Structured Data Server Facet

Group	Conformance Unit / Profile Title	Optional
Historical Access	Historical Access Structured Data Delete	True
Historical Access	Historical Access Structured Data Insert	True
Historical Access	Historical Access Structured Data Read Modified	True
Historical Access	Historical Access Structured Data Read Raw	False
Historical Access	Historical Access Structured Data Replace	True
Historical Access	Historical Access Structured Data Time Instance	True
Historical Access	Historical Access Structured Data Update	True

6.6.58 Base Historical Event Server Facet

Table 76 describes the details of the Base Historical *Event* Server Facet. This Facet defines the server requirements to support basic Historical *Event* functionality, including simple filtering and general access.

Table 76 – Base Historical Event Server Facet

Group	Conformance Unit / Profile Title	Optional
<i>Attribute Services</i>	<i>Attribute</i> Historical Read	False
Historical Access	Historical Access <i>Event</i> Max Events Read Continuation Point	False
Historical Access	Historical Access Events	False

6.6.59 Historical Event Update Server Facet

Table 77 describes the details of the Historical *Event* Update Server Facet. This Facet includes Historical *Event* update access functionality.

Table 77 – Historical Event Update Server Facet

Group	Conformance Unit / Profile Title	Optional
<i>Attribute Services</i>	<i>Attribute</i> Historical Update	False
Historical Access	Historical Access Update <i>Event</i>	False

6.6.60 Historical Event Replace Server Facet

Table 78 describes the details of the Historical *Event* Replace Server Facet. This Facet includes Historical *Event* replace access functionality.

Table 78 – Historical Event Replace Server Facet

Group	Conformance Unit / Profile Title	Optional
<i>Attribute Services</i>	<i>Attribute</i> Historical Update	False
Historical Access	Historical Access Replace <i>Event</i>	False

6.6.61 Historical Event Insert Server Facet

Table 79 describes the details of the Historical *Event* Insert Server Facet. This Facet includes Historical *Event* insert access functionality.

Table 79 – Historical Event Insert Server Facet

Group	Conformance Unit / Profile Title	Optional
<i>Attribute Services</i>	<i>Attribute</i> Historical Update	False
Historical Access	Historical Access Insert <i>Event</i>	False

6.6.62 Historical Event Delete Server Facet

Table 80 describes the details of the Historical *Event* Delete Server Facet. This Facet includes Historical *Event* delete access functionality

Table 80 – Historical Event Delete Server Facet

Group	Conformance Unit / Profile Title	Optional
<i>Attribute Services</i>	<i>Attribute</i> Historical Update	False
Historical Access	Historical Access Delete <i>Event</i>	False

6.6.63 Aggregate Subscription Server Facet

Table 81 describes the details of the Aggregate *Subscription Server* Facet. This Facet defines the handling of the aggregate filter when subscribing for *Attribute* values.

Table 81 – Aggregate Subscription Server Facet

Group	Conformance Unit / Profile Title	Optional
<i>Profile</i>	Standard DataChange <i>Subscription Server</i> Facet	False
Aggregates	Aggregate <i>Subscription</i> – AnnotationCount	True
Aggregates	Aggregate <i>Subscription</i> – Average	True
Aggregates	Aggregate <i>Subscription</i> – Count	True
Aggregates	Aggregate <i>Subscription</i> – Custom	True
Aggregates	Aggregate <i>Subscription</i> – Delta	True
Aggregates	Aggregate <i>Subscription</i> – DeltaBounds	True
Aggregates	Aggregate <i>Subscription</i> – DurationBad	True
Aggregates	Aggregate <i>Subscription</i> – DurationGood	True
Aggregates	Aggregate <i>Subscription</i> – DurationInStateNonZero	True
Aggregates	Aggregate <i>Subscription</i> – DurationInStateZero	True
Aggregates	Aggregate <i>Subscription</i> – End	True
Aggregates	Aggregate <i>Subscription</i> – EndBound	True
Aggregates	Aggregate <i>Subscription</i> – Filter	False
Aggregates	Aggregate <i>Subscription</i> – Interpolative	True
Aggregates	Aggregate <i>Subscription</i> – Maximum	True
Aggregates	Aggregate <i>Subscription</i> – Maximum2	True
Aggregates	Aggregate <i>Subscription</i> – MaximumActualTime	True
Aggregates	Aggregate <i>Subscription</i> – MaximumActualTime2	True
Aggregates	Aggregate <i>Subscription</i> – Minimum	True
Aggregates	Aggregate <i>Subscription</i> – Minimum2	True
Aggregates	Aggregate <i>Subscription</i> – MinimumActualTime	True
Aggregates	Aggregate <i>Subscription</i> – MinimumActualTime2	True
Aggregates	Aggregate <i>Subscription</i> – NumberOfTransitions	True
Aggregates	Aggregate <i>Subscription</i> – PercentBad	True
Aggregates	Aggregate <i>Subscription</i> – PercentGood	True
Aggregates	Aggregate <i>Subscription</i> – Range	True
Aggregates	Aggregate <i>Subscription</i> – Range2	True
Aggregates	Aggregate <i>Subscription</i> – StandardDeviationPopulation	True
Aggregates	Aggregate <i>Subscription</i> – StandardDeviationSample	True
Aggregates	Aggregate <i>Subscription</i> – Start	True
Aggregates	Aggregate <i>Subscription</i> – StartBound	True
Aggregates	Aggregate <i>Subscription</i> – TimeAverage	True
Aggregates	Aggregate <i>Subscription</i> – TimeAverage2	True
Aggregates	Aggregate <i>Subscription</i> – Total	True
Aggregates	Aggregate <i>Subscription</i> – Total2	True
Aggregates	Aggregate <i>Subscription</i> – VariancePopulation	True
Aggregates	Aggregate <i>Subscription</i> – VarianceSample	True
Aggregates	Aggregate <i>Subscription</i> – WorstQuality	True
Aggregates	Aggregate <i>Subscription</i> – WorstQuality2	True
Monitored Item <i>Services</i>	Monitor Aggregate Filter	False

6.6.64 Nano Embedded Device Server Profile

Nano Embedded Device Server Profile has been deprecated as it is superseded with Nano Embedded Device 2017 Server Profile in v1.04.

6.6.65 Nano Embedded Device 2017 Server Profile

Table 82 describes the details of the Nano Embedded Device 2017 *Server Profile*. This *Profile* is a FullFeatured *Profile* intended for chip level devices with limited resources. This *Profile* is functionally equivalent to the Core *Server Facet* and defines the OPC UA TCP binary protocol as the required transport profile. The support of Diagnostic *Objects* and *Variables* is optional for this *Profile* despite it being defined as "mandatory" in IEC 62541-5. Support of Diagnostic *Objects* and *Variables* is mandatory in some higher-level *Profiles*. Exposing types in the *AddressSpace* is optional for this *Profile* except if custom types (i.e. types that are derived from well-known *ObjectTypes*, *VariableTypes*, *ReferenceType* or *DataTypes*) are used. Exposing all supported types in the *AddressSpace* is mandatory in some higher-level *Profiles*. This profile supersedes the "Nano Embedded Device *Server Profile*".

Table 82 – Nano Embedded Device 2017 Server Profile

Group	Conformance Unit / Profile Title	Optional
<i>Profile</i>	Core 2017 <i>Server Facet</i>	False
<i>Profile</i>	UA-TCP UA-SC UA-Binary	False
Base Information	Base Info Custom Type System	True
Base Information	Base Info Diagnostics	True

6.6.66 Micro Embedded Device Server Profile

Micro Embedded Device Server Profile has been deprecated as it is superseded with Micro Embedded Device 2017 Server Profile in v1.04.

6.6.67 Micro Embedded Device 2017 Server Profile

Table 83 describes the details of the Micro Embedded Device 2017 *Server Profile*. This *Profile* is a FullFeatured *Profile* intended for small devices with limited resources. This *Profile* builds upon the Nano Embedded Device *Server Profile*. The most important additions are: support for subscriptions via the Embedded Data Change *Subscription Server Facet* and support for at least two sessions. A complete Type System is not required; however, if the *Server* implements any non-UA types, then these types and their super-types shall be exposed. This profile supersedes the "Micro Embedded Device *Server Profile*".

Table 83 – Micro Embedded Device 2017 Server Profile

Group	Conformance Unit / Profile Title	Optional
<i>Profile</i>	Embedded DataChange <i>Subscription Server Facet</i>	False
<i>Profile</i>	Nano Embedded Device 2017 <i>Server Profile</i>	False
<i>Session Services</i>	<i>Session</i> Minimum 2 Parallel	False

6.6.68 Embedded UA Server Profile

Embedded UA Server Profile has been deprecated as it is superseded with Embedded 2017 UA Server Profile in v1.04.

6.6.69 Embedded 2017 UA Server Profile

Table 84 describes the details of the Embedded 2017 UA *Server Profile*. This *Profile* is a FullFeatured *Profile* that is intended for devices with more than 50 MB of memory and a more powerful processor. This *Profile* builds upon the Micro Embedded Device *Server Profile*. The most important additions are: support for security via the Security Policies and support for the Standard DataChange *Subscription Server* Facet. This *Profile* also requires that Servers expose all OPC-UA types that are used by the *Server* including their components and their super-types. This profile supersedes the "Embedded Device *Server Profile*".

Table 84 – Embedded 2017 UA Server Profile

Group	Conformance Unit / Profile Title	Optional
Profile	Micro Embedded Device 2017 <i>Server Profile</i>	False
Profile	Standard DataChange <i>Subscription</i> 2017 <i>Server</i> Facet	False
Base Information	Base Info Engineering Units	True
Base Information	Base Info Type System	False
Security	Security – No Application Authentication	True
Security	Security Default ApplicationInstance Certificate	False
Security	Security Policy Required	False

6.6.70 Standard UA Server Profile

Standard UA Server Profile has been deprecated as it is superseded with Standard 2017 UA Server Profile in v1.04.

6.6.71 Standard 2017 UA Server Profile

Table 85 describes the details of the Standard 2017 UA *Server Profile*. This *Profile* is a FullFeatured *Profile* that defines a minimum set of functionality required for PC based OPC UA servers. Compared to the embedded profiles, the *Profile* requires higher limits for Sessions, Subscriptions and Monitored Items. It also requires support of diagnostic information. This profile supersedes the "Standard UA *Server Profile*".

Table 85 – Standard 2017 UA Server Profile

Group	Conformance Unit / Profile Title	Optional
Profile	Embedded 2017 UA <i>Server Profile</i>	False
Profile	Enhanced DataChange <i>Subscription</i> 2017 <i>Server</i> Facet	False
Profile	User Token – X509 <i>Certificate</i> <i>Server</i> Facet	False
Attribute Services	Attribute Write StatusCode & Timestamp	True
Base Information	Base Info Diagnostics	False
Discovery Services	Discovery Register	False
Discovery Services	Discovery Register2	False
Session Services	Session Cancel	False
Session Services	Session Change User	True
Session Services	Session Minimum 50 Parallel	False
View Services	View Minimum Continuation Point 05	False

6.6.72 Core Client Facet

Core Client Facet has been deprecated as it is superseded with Core 2017 Client Facet in v1.04.

6.6.73 Core 2017 Client Facet

Table 86 describes the details of the Core 2017 *Client* Facet. This Facet defines the core functionality required for any *Client*. This Facet includes the core functions for Security and Session handling.

This Facet supersedes the Core *Client* Facet.

Table 86 – Core 2017 Client Facet

Group	Conformance Unit / Profile Title	Optional
Profile	SecurityPolicy – None	False
Profile	User Token – User Name Password <i>Client</i> Facet	False
Profile	User Token – X509 Certificate <i>Client</i> Facet	False
Base Information	Base Info <i>Client</i> Estimated Return Time	True
Base Information	Base Info <i>Client</i> Selection List	True
Security	Security Administration	False
Security	Security Policy Required	False
Session Services	Session <i>Client</i> Auto Reconnect	False
Session Services	Session <i>Client</i> Base	False
Session Services	Session <i>Client</i> Cancel	True
Session Services	Session <i>Client</i> Detect Shutdown	False
Session Services	Session <i>Client</i> General Service Behaviour	False
Session Services	Session <i>Client</i> Impersonate	True
Session Services	Session <i>Client</i> KeepAlive	False
Session Services	Session <i>Client</i> Renew NodeIds	True

6.6.74 Sessionless Client Facet

Table 87 describes the details of the Sessionless *Client* Facet. Defines the use of Sessionless Service invocation in a *Client*.

Table 87 – Sessionless Client Facet

Group	Conformance Unit / Profile Title	Optional
Discovery Services	Discovery <i>Client</i> Get Endpoints SessionLess	True
Session Services	Session <i>Client</i> SessionLess Service Calls	False

6.6.75 Reverse Connect Client Facet

Table 88 describes the details of the Reverse Connect *Client* Facet. This Facet defines support of reverse connectivity in a *Client*. Usually, a connection is opened by the *Client* before starting the UA-specific handshake. This will fail, however, when Servers are behind firewalls. In the reverse connectivity scenario, the *Client* accepts a connection request and a ReverseHello message from a *Server* and establishes a Secure Channel using this connection.

Table 88 – Reverse Connect Client Facet

Group	Conformance Unit / Profile Title	Optional
Protocol and Encoding	Protocol Reverse Connect <i>Client</i>	False

6.6.76 Base Client Behaviour Facet

Table 89 describes the details of the Base *Client* Behaviour Facet. This Facet indicates that the *Client* supports behaviour that *Clients* shall follow for best use by operators and administrators. They include allowing configuration of an endpoint for a server without using the discovery service set; Support for manual security setting configuration and behaviour with regard to security issues; support for Automatic reconnection to a disconnected server. These behaviours can only be tested in a test lab. They are best practice guidelines.

Table 89 – Base Client Behaviour Facet

Group	Conformance Unit / Profile Title	Optional
Base Information	Base Info <i>Client</i> Remote Nodes	True
Discovery Services	Discovery <i>Client</i> Configure Endpoint	False
Security	Security Administration	False
Security	Security Administration – XML Schema	False
Security	Security <i>Certificate</i> Administration	False
Session Services	Session <i>Client</i> Auto Reconnect	True
Subscription Services	Subscription <i>Client</i> Multiple	False
Subscription Services	Subscription <i>Client</i> Publish Configurable	False

6.6.77 Discovery Client Facet

Table 90 describes the details of the *Discovery Client* Facet. This Facet defines the ability to discover Servers and their Endpoints.

Table 90 – Discovery Client Facet

Group	Conformance Unit / Profile Title	Optional
Discovery Services	Discovery <i>Client</i> Configure Endpoint	False
Discovery Services	Discovery <i>Client</i> Find Servers Basic	False
Discovery Services	Discovery <i>Client</i> Find Servers Dynamic	False
Discovery Services	Discovery <i>Client</i> Find Servers with URI	True
Discovery Services	Discovery <i>Client</i> Get Endpoints Basic	False
Discovery Services	Discovery <i>Client</i> Get Endpoints Dynamic	False

6.6.78 Subnet Discovery Client Facet

Table 91 describes the details of the Subnet *Discovery Client* Facet. Support of this Facet enables discovery of the Server on a subnet.

Table 91 – Subnet Discovery Client Facet

Group	Conformance Unit / Profile Title	Optional
<i>Discovery Services</i>	<i>Discovery Client</i> Find Servers on Network	False
<i>Discovery Services</i>	<i>Discovery Client</i> Find Servers on Network using LDS-ME	True
<i>Discovery Services</i>	<i>Discovery Client</i> Find Servers on Network using mDNS	True

6.6.79 Global Discovery Client Facet

Table 92 describes the details of the Global *Discovery Client* Facet. Support of this Facet enables system-wide discovery of Servers using a Global *Discovery Server* (GDS).

Table 92 – Global Discovery Client Facet

Group	Conformance Unit / Profile Title	Optional
<i>Discovery Services</i>	<i>Discovery Client</i> Find Applications in GDS	True
<i>Discovery Services</i>	<i>Discovery Client</i> Find Servers in GDS	False

6.6.80 Global Certificate Management Client Facet

Table 93 describes the details of the Global *Certificate Management Client* Facet. This Facet defines the capability to interact with a Global *Certificate Management Server* to obtain an initial or renewed *Certificate* and Trust Lists.

Table 93 – Global Certificate Management Client Facet

Group	Conformance Unit / Profile Title	Optional
Security	Pull Model for Global Certificate and TrustList Management	False

6.6.81 KeyCredential Service Client Facet

Table 94 describes the details of the KeyCredential *Service Client* Facet. This Facet defines the capability to interact with a KeyCredential *Service* to obtain KeyCredentials. For example KeyCredentials are needed to access an Authorization *Service* or a Broker. The KeyCredential *Service* is typically part of a system-wide tool, like a GDS that also manages Applications, Access Tokens, and *Certificates*.

Table 94 – KeyCredential Service Client Facet

Group	Conformance Unit / Profile Title	Optional
Security	Pull Model for KeyCredential <i>Service</i>	False

6.6.82 Access Token Request Client Facet

Table 95 describes the details of the Access Token Request *Client* Facet. A *Client* Facet for using the RequestAccessToken *Method* on an Authorization *Server* (defined in IEC 62541-12) to request such a token.

Table 95 – Access Token Request Client Facet

Group	Conformance Unit / Profile Title	Optional
Security	Authorization <i>Service Client</i>	False

6.6.83 AddressSpace Lookup Client Facet

Table 96 describes the details of the *AddressSpace* Lookup *Client* Facet. This Facet defines the ability to navigate through the *AddressSpace* and includes basic *AddressSpace* concepts, view and browse functionality and simple attribute read functionality.

Table 96 – AddressSpace Lookup Client Facet

Group	Conformance Unit / Profile Title	Optional
Address Space Model	Address Space <i>Client</i> Base	False
<i>Attribute Services</i>	<i>Attribute Client</i> Read Base	False
<i>Attribute Services</i>	<i>Attribute Client</i> Remote Nodes <i>Attribute</i> Access	True
Base Information	Base Info <i>Client</i> Basic	False
Base Information	Base Info <i>Client</i> GetMonitoredItems <i>Method</i>	True
View <i>Services</i>	View <i>Client</i> Basic Browse	False
View <i>Services</i>	View <i>Client</i> Basic ResultSet Filtering	False
View <i>Services</i>	View <i>Client</i> RegisterNodes	True
View <i>Services</i>	View <i>Client</i> Remote Nodes Browse	True
View <i>Services</i>	View <i>Client</i> Remote Nodes Translate Browse	True
View <i>Services</i>	View <i>Client</i> TranslateBrowsePath	True

6.6.84 Request State Change Client Facet

Table 97 describes the details of the Request State Change *Client* Facet. This Facet specifies the ability to invoke the RequestServerStateChange *Method*.

Table 97 – Request State Change Client Facet

Group	Conformance Unit / Profile Title	Optional
Base Information	Base Info <i>Client</i> RequestServerStateChange	False

6.6.85 File Access Client Facet

Table 98 describes the details of the File Access *Client* Facet. This Facet defines the ability to use File transfer via the defined FileType. This includes reading and optionally writing.

Table 98 – File Access Client Facet

Group	Conformance Unit / Profile Title	Optional
Base Information	Base Info <i>Client</i> FileType Base	False
Base Information	Base Info <i>Client</i> FileType Write	True

6.6.86 Entry Level Support 2015 Client Facet

Table 99 describes the details of the Entry Level Support 2015 *Client* Facet. This Facet defines the ability to interoperate with low-end *Servers*, in particular *Servers* that support the Nano Embedded *Profile* but in general *Servers* with defined limits.

Table 99 – Entry Level Support 2015 Client Facet

Group	Conformance Unit / <i>Profile</i> Title	Optional
Base Information	Base Info <i>Client</i> Honour Operation Limits	False
Base Information	Base Info <i>Client</i> Type Pre-Knowledge	False
<i>Session Services</i>	<i>Session Client</i> Single <i>Session</i>	False
<i>Subscription Services</i>	<i>Subscription Client</i> Fallback	False

6.6.87 Multi-Server Client Connection Facet

Table 100 describes the details of the Multi-Server *Client* Connection Facet. This Facet defines the ability for simultaneous access to multiple *Servers*.

Table 100 – Multi-Server Client Connection Facet

Group	Conformance Unit / <i>Profile</i> Title	Optional
<i>Session Services</i>	<i>Session Client</i> Multiple Connections	False

6.6.88 Documentation – Client

Table 101 describes the details of the Documentation – *Client*. This Facet provides a list of user documentation that a *Client* application should provide.

Table 101 – Documentation – Client

Group	Conformance Unit / <i>Profile</i> Title	Optional
Miscellaneous	Documentation <i>Client</i> – Installation	False
Miscellaneous	Documentation <i>Client</i> – Multiple Languages	True
Miscellaneous	Documentation <i>Client</i> – On-line	True
Miscellaneous	Documentation <i>Client</i> – Supported <i>Profiles</i>	True
Miscellaneous	Documentation <i>Client</i> – Trouble Shooting Guide	True
Miscellaneous	Documentation <i>Client</i> – Users Guide	False

6.6.89 Attribute Read Client Facet

Table 102 describes the details of the *Attribute* Read *Client* Facet. This Facet defines the ability to read *Attribute* values of *Nodes*.

Table 102 – Attribute Read Client Facet

Group	Conformance Unit / Profile Title	Optional
Address Space Model	Address Space <i>Client</i> Atomicity	True
Address Space Model	Address Space <i>Client</i> Complex Data Dictionary	True
Address Space Model	Address Space <i>Client</i> DataTypeDefinition <i>Attribute</i>	True
Address Space Model	Address Space <i>Client</i> Full Array Only	True
<i>Attribute Services</i>	<i>Attribute Client</i> Read Base	False
<i>Attribute Services</i>	<i>Attribute Client</i> Read Complex	True
<i>Attribute Services</i>	<i>Attribute Client</i> Read with proper Encoding	True

6.6.90 Attribute Write Client Facet

Table 103 describes the details of the *Attribute Write Client* Facet. This Facet defines the ability to write *Attribute* values of *Nodes*.

Table 103 – Attribute Write Client Facet

Group	Conformance Unit / Profile Title	Optional
Address Space Model	Address Space <i>Client</i> Atomicity	True
Address Space Model	Address Space <i>Client</i> Complex Data Dictionary	True
Address Space Model	Address Space <i>Client</i> DataTypeDefinition <i>Attribute</i>	True
Address Space Model	Address Space <i>Client</i> Full Array Only	True
<i>Attribute Services</i>	<i>Attribute Client</i> Write Base	False
<i>Attribute Services</i>	<i>Attribute Client</i> Write Complex	True
<i>Attribute Services</i>	<i>Attribute Client</i> Write Quality & Timestamp	True

6.6.91 DataChange Subscriber Client Facet

Table 104 describes the details of the DataChange Subscriber *Client* Facet. This Facet defines the ability to monitor *Attribute* values for data change.

Table 104 – DataChange Subscriber Client Facet

Group	Conformance Unit / <i>Profile</i> Title	Optional
Address Space Model	Address Space <i>Client</i> Atomicity	True
Address Space Model	Address Space <i>Client</i> Complex Data Dictionary	True
Address Space Model	Address Space <i>Client</i> DataTypeDefinition <i>Attribute</i>	True
Address Space Model	Address Space <i>Client</i> Full Array Only	True
Base Information	Base Data <i>Client</i> ResendData <i>Method</i>	True
Base Information	Base Info <i>Client</i> GetMonitoredItems <i>Method</i>	True
Monitored Item <i>Services</i>	Monitor <i>Client</i> by Index	False
Monitored Item <i>Services</i>	Monitor <i>Client</i> Complex Value	True
Monitored Item <i>Services</i>	Monitor <i>Client</i> Deadband Filter	True
Monitored Item <i>Services</i>	Monitor <i>Client</i> Modify	True
Monitored Item <i>Services</i>	Monitor <i>Client</i> Trigger	True
Monitored Item <i>Services</i>	Monitor <i>Client</i> Value Change	False
<i>Subscription Services</i>	<i>Subscription Client</i> Basic	False
<i>Subscription Services</i>	<i>Subscription Client</i> Modify	True
<i>Subscription Services</i>	<i>Subscription Client</i> Multiple	True
<i>Subscription Services</i>	<i>Subscription Client</i> Republish	False

6.6.92 Durable Subscription Client Facet

Table 105 describes the details of the Durable *Subscription Client* Facet. This Facet specifies use of durable Subscriptions. It implies support of any of the DataChange or *Event* Subscriber Facets.

Table 105 – Durable Subscription Client Facet

Group	Conformance Unit / <i>Profile</i> Title	Optional
<i>Subscription Services</i>	<i>Subscription Client</i> Durable	False

6.6.93 DataAccess Client Facet

Table 106 describes the details of the DataAccess *Client* Facet. This Facet defines the ability to utilize the DataAccess Information Model, i.e., industrial automation data like analogue and discrete data items and their quality of service.

Table 106 – DataAccess Client Facet

Group	Conformance Unit / Profile Title	Optional
Address Space Model	Address Space <i>Client</i> Base	False
Address Space Model	Address Space <i>Client</i> Complex Data Dictionary	True
<i>Attribute Services</i>	<i>Attribute Client</i> Read Base	False
<i>Attribute Services</i>	<i>Attribute Client</i> Read Complex	True
<i>Attribute Services</i>	<i>Attribute Client</i> Read with proper Encoding	True
Data Access	Data Access <i>Client</i> AnalogItems	True
Data Access	Data Access <i>Client</i> Basic	False
Data Access	Data Access <i>Client</i> Deadband	True
Data Access	Data Access <i>Client</i> MultiState	True
Data Access	Data Access <i>Client</i> MultiStateValueDiscrete	True
Data Access	Data Access <i>Client</i> SemanticChange	True
Data Access	Data Access <i>Client</i> TwoState	True

6.6.94 Event Subscriber Client Facet

Table 107 describes the details of the *Event Subscriber Client* Facet. This Facet defines the ability to subscribe for *Event Notifications*. This includes basic *AddressSpace* concept and the browsing of it, adding events and event filters as monitored items and adding subscriptions.

Table 107 – Event Subscriber Client Facet

Group	Conformance Unit / Profile Title	Optional
Address Space Model	Address Space <i>Client</i> Base	False
Monitored Item <i>Services</i>	Monitor <i>Client</i> Complex <i>Event</i> Filter	True
Monitored Item <i>Services</i>	Monitor <i>Client</i> <i>Event</i> Filter	False
Monitored Item <i>Services</i>	Monitor <i>Client</i> Events	False
Monitored Item <i>Services</i>	Monitor <i>Client</i> Modify	True
Monitored Item <i>Services</i>	Monitor <i>Client</i> Trigger	True
<i>Subscription Services</i>	<i>Subscription Client</i> Basic	False
<i>Subscription Services</i>	<i>Subscription Client</i> Modify	True
<i>Subscription Services</i>	<i>Subscription Client</i> Multiple	True
<i>Subscription Services</i>	<i>Subscription Client</i> Republish	False
View <i>Services</i>	View <i>Client</i> Basic Browse	True
View <i>Services</i>	View <i>Client</i> TranslateBrowsePath	True

6.6.95 Base Event Processing Client Facet

Table 108 describes the details of the Base *Event Processing Client* Facet. This Facet defines the ability to subscribe for and process basic OPC UA Events. The *Client* has to support at least one of the Events in the Facet.

Table 108 – Base Event Processing Client Facet

Group	Conformance Unit / Profile Title	Optional
<i>Profile</i>	<i>Event Subscriber Client Facet</i>	False
Base Information	Base Info <i>Client</i> Change Events	True
Base Information	Base Info <i>Client</i> Device Failure	True
Base Information	Base Info <i>Client</i> Progress Events	True
Base Information	Base Info <i>Client</i> System Status	True
Base Information	Base Info <i>Client</i> System Status Underlying System	True
Base Information	Base Info <i>Event</i> Processing	False

6.6.96 Notifier and Source Hierarchy Client Facet

Table 109 describes the details of the Notifier and Source Hierarchy *Client* Facet. This Facet defines the ability to find and use a hierarchy of *Objects* that are event notifier and *Nodes* that are event sources in the *Server AddressSpace*.

Table 109 – Notifier and Source Hierarchy Client Facet

Group	Conformance Unit / Profile Title	Optional
<i>Profile</i>	<i>Event Subscriber Client Facet</i>	False
Address Space Model	Address Space <i>Client</i> Notifier Hierarchy	False
Address Space Model	Address Space <i>Client</i> Source Hierarchy	False
<i>Subscription Services</i>	<i>Subscription Client</i> Publish Configurable	False

6.6.97 A & C Base Condition Client Facet

Table 110 describes the details of the A & C Base *Condition Client* Facet. This Facet defines the ability to use the *Alarm* and *Condition* basic model. This includes the ability to subscribe for Events and to initiate a Refresh *Method*.

Table 110 – A & C Base Condition Client Facet

Group	Conformance Unit / Profile Title	Optional
<i>Profile</i>	<i>Event Subscriber Client Facet</i>	False
<i>Profile</i>	<i>Method Client Facet</i>	False
<i>Alarms and Conditions</i>	A & C Basic <i>Client</i>	False
<i>Alarms and Conditions</i>	A & C <i>Condition</i> Sub-Classes <i>Client</i>	True
<i>Alarms and Conditions</i>	A & C ConditionClasses <i>Client</i>	False
<i>Alarms and Conditions</i>	A & C Refresh <i>Client</i>	False

6.6.98 A & C Refresh2 Client Facet

Table 111 describes the details of the A & C Refresh2 *Client* Facet. This Facet enhances the A & C Base *Condition Server* Facet with the ability to initiate a ConditionRefresh2 *Method*.

Table 111 – A & C Refresh2 Client Facet

Group	Conformance Unit / Profile Title	Optional
<i>Profile</i>	A & C Base <i>Condition Client</i> Facet	False
<i>Alarms and Conditions</i>	A & C Refresh2 <i>Client</i>	False

6.6.99 A & C Address Space Instance Client Facet

Table 112 describes the details of the A & C Address Space Instance *Client* Facet. This Facet defines the ability to use *Condition* instances in the *AddressSpace*.

Table 112 – A & C Address Space Instance Client Facet

Group	Conformance Unit / Profile Title	Optional
<i>Alarms and Conditions</i>	A & C Instances <i>Client</i>	False

6.6.100 A & C Enable Client Facet

Table 113 describes the details of the A & C Enable *Client* Facet. This Facet defines the ability to enable and disable *Alarms*.

Table 113 – A & C Enable Client Facet

Group	Conformance Unit / Profile Title	Optional
<i>Profile</i>	A & C Base <i>Condition Client</i> Facet	False
<i>Alarms and Conditions</i>	A & C Enable <i>Client</i>	False

6.6.101 A & C AlarmMetrics Client Facet

Table 114 describes the details of the A & C AlarmMetrics *Client* Facet. This Facet defines the ability to use the AlarmMetrics model, i.e. understand and use the collected alarm metrics at any level in the HasNotifier hierarchy.

Table 114 – A & C AlarmMetrics Client Facet

Group	Conformance Unit / Profile Title	Optional
<i>Alarms and Conditions</i>	A & C <i>Alarm Metrics Client</i>	False

6.6.102 A & C Alarm Client Facet

Table 115 describes the details of the A & C *Alarm Client* Facet. This Facet defines the ability to use the alarming model (the AlarmType or any of the sub-types).

Table 115 – A & C Alarm Client Facet

Group	Conformance Unit / Profile Title	Optional
<i>Profile</i>	A & C Base <i>Condition Client</i> Facet	False
<i>Alarms and Conditions</i>	A & C Acknowledge <i>Client</i>	False
<i>Alarms and Conditions</i>	A & C Alarm <i>Client</i>	False
<i>Alarms and Conditions</i>	A & C Audible Sound <i>Client</i>	True
<i>Alarms and Conditions</i>	A & C Comment <i>Client</i>	True
<i>Alarms and Conditions</i>	A & C Confirm <i>Client</i>	True
<i>Alarms and Conditions</i>	A & C Discrepancy <i>Client</i>	True
<i>Alarms and Conditions</i>	A & C Discrete <i>Client</i>	False
<i>Alarms and Conditions</i>	A & C First in Group Alarm <i>Client</i>	True
<i>Alarms and Conditions</i>	A & C OffNormal <i>Client</i>	True
<i>Alarms and Conditions</i>	A & C On-Off Delay <i>Client</i>	True
<i>Alarms and Conditions</i>	A & C Out Of Service <i>Client</i>	True
<i>Alarms and Conditions</i>	A & C Re-Alarming <i>Client</i>	True
<i>Alarms and Conditions</i>	A & C Shelving <i>Client</i>	True
<i>Alarms and Conditions</i>	A & C Silencing <i>Client</i>	True
<i>Alarms and Conditions</i>	A & C Suppression by Operator <i>Client</i>	True
<i>Alarms and Conditions</i>	A & C Suppression <i>Client</i>	True
<i>Alarms and Conditions</i>	A & C SystemOffNormal <i>Client</i>	True
<i>Alarms and Conditions</i>	A & C Trip <i>Client</i>	True

6.6.103 A & C Exclusive Alarming Client Facet

Table 116 describes the details of the A & C Exclusive Alarming *Client* Facet. This Facet defines the ability to use the exclusive *Alarm* model. This includes understanding the various subtypes such as ExclusiveRateOfChangeAlarm, ExclusiveLevelAlarm and ExclusiveDeviationAlarm.

Table 116 – A & C Exclusive Alarming Client Facet

Group	Conformance Unit / Profile Title	Optional
<i>Profile</i>	A & C Alarm <i>Client</i> Facet	False
<i>Alarms and Conditions</i>	A & C Exclusive Deviation <i>Client</i>	True
<i>Alarms and Conditions</i>	A & C Exclusive Level <i>Client</i>	True
<i>Alarms and Conditions</i>	A & C Exclusive Limit <i>Client</i>	False
<i>Alarms and Conditions</i>	A & C Exclusive RateOfChange <i>Client</i>	True

6.6.104 A & C Non-Exclusive Alarming Client Facet

Table 117 describes the details of the A & C Non-Exclusive Alarming *Client* Facet. This Facet defines the ability to use the non-exclusive *Alarm* model. This includes understanding the various subtypes such as NonExclusiveRateOfChangeAlarm, NonExclusiveLevelAlarm and NonExclusiveDeviationAlarm.

Table 117 – A & C Non-Exclusive Alarming Client Facet

Group	Conformance Unit / Profile Title	Optional
<i>Profile</i>	A & C <i>Alarm Client</i> Facet	False
<i>Alarms and Conditions</i>	A & C Non-Exclusive Deviation <i>Client</i>	True
<i>Alarms and Conditions</i>	A & C Non-Exclusive Level <i>Client</i>	True
<i>Alarms and Conditions</i>	A & C Non-Exclusive Limit <i>Client</i>	False
<i>Alarms and Conditions</i>	A & C Non-Exclusive RateOfChange <i>Client</i>	True

6.6.105 A & C Previous Instances Client Facet

Table 118 describes the details of the A & C Previous Instances *Client* Facet. This Facet defines the ability to use previous instances of *Alarms*. This implies the ability to understand branchIds.

Table 118 – A & C Previous Instances Client Facet

Group	Conformance Unit / Profile Title	Optional
<i>Profile</i>	A & C Base <i>Condition Client</i> Facet	False
<i>Alarms and Conditions</i>	A & C Branch <i>Client</i>	False

6.6.106 A & C Dialog Client Facet

Table 119 describes the details of the A & C Dialog *Client* Facet. This Facet defines the ability to use the dialog model. This implies the support of *Method* invocation to respond to dialog messages.

Table 119 – A & C Dialog Client Facet

Group	Conformance Unit / Profile Title	Optional
<i>Profile</i>	A & C Base <i>Condition Client</i> Facet	False
<i>Alarms and Conditions</i>	A & C Dialog <i>Client</i>	False

6.6.107 A & C CertificateExpiration Client Facet

Table 120 describes the details of the A & C CertificateExpiration *Client* Facet. This Facet defines the ability to use the CertificateExpirationAlarmType.

Table 120 – A & C CertificateExpiration Client Facet

Group	Conformance Unit / Profile Title	Optional
<i>Profile</i>	A & C Base <i>Condition Client</i> Facet	False
<i>Alarms and Conditions</i>	A & C Acknowledge <i>Client</i>	True
<i>Alarms and Conditions</i>	A & C <i>Alarm Client</i>	False
<i>Alarms and Conditions</i>	A & C CertificateExpiration <i>Client</i>	False
<i>Alarms and Conditions</i>	A & C Comment <i>Client</i>	True
<i>Alarms and Conditions</i>	A & C Confirm <i>Client</i>	True
<i>Alarms and Conditions</i>	A & C Shelving <i>Client</i>	True

6.6.108 A & E Proxy Facet

Table 121 describes the details of the A & E Proxy Facet. This Facet describes the functionality used by a default A & E *Client* proxy. A *Client* exposes this Facet so that a *Server* may be able to better understand the commands that are being issued by the *Client*, since this Facet indicates that the *Client* is an A&E Com *Client*.

Table 121 – A & E Proxy Facet

Group	Conformance Unit / Profile Title	Optional
Address Space Model	Address Space <i>Client</i> Base	False
Alarms and Conditions	A & C Acknowledge <i>Client</i>	False
Alarms and Conditions	A & C Alarm <i>Client</i>	False
Alarms and Conditions	A & C Basic <i>Client</i>	False
Alarms and Conditions	A & C ConditionClasses <i>Client</i>	False
Alarms and Conditions	A & C Discrete <i>Client</i>	False
Alarms and Conditions	A & C Exclusive Deviation <i>Client</i>	False
Alarms and Conditions	A & C Exclusive Level <i>Client</i>	False
Alarms and Conditions	A & C Exclusive Limit <i>Client</i>	False
Alarms and Conditions	A & C Exclusive RateOfChange <i>Client</i>	False
Alarms and Conditions	A & C Instances <i>Client</i>	False
Alarms and Conditions	A & C Non-Exclusive Deviation <i>Client</i>	False
Alarms and Conditions	A & C Non-Exclusive Level <i>Client</i>	False
Alarms and Conditions	A & C Non-Exclusive Limit <i>Client</i>	False
Alarms and Conditions	A & C Non-Exclusive RateOfChange <i>Client</i>	False
Alarms and Conditions	A & C OffNormal <i>Client</i>	False
Alarms and Conditions	A & C Refresh <i>Client</i>	False
Alarms and Conditions	A & C SystemOffNormal <i>Client</i>	True
Alarms and Conditions	A & C Trip <i>Client</i>	False
Attribute Services	Attribute <i>Client</i> Read Base	False
Base Information	Base Info <i>Client</i> Basic	False
Base Information	Base Info <i>Client</i> Change Events	False
Discovery Services	Discovery <i>Client</i> Configure Endpoint	False
Discovery Services	Discovery <i>Client</i> Find Servers Basic	False
Discovery Services	Discovery <i>Client</i> Find Servers Dynamic	False
Discovery Services	Discovery <i>Client</i> Find Servers with URI	False
Discovery Services	Discovery <i>Client</i> Get Endpoints Basic	False
Discovery Services	Discovery <i>Client</i> Get Endpoints Dynamic	False
Method Services	Method <i>Client</i> Call	False
Monitored Item Services	Monitor <i>Client</i> Complex Event Filter	False
Monitored Item Services	Monitor <i>Client</i> Event Filter	False
Monitored Item Services	Monitor <i>Client</i> Events	False
Security	Security Administration	False
Security	Security Administration – XML Schema	False
Security	Security Certificate Administration	False
Session Services	Session <i>Client</i> Auto Reconnect	False
Subscription Services	Subscription <i>Client</i> Basic	False

Group	Conformance Unit / Profile Title	Optional
Subscription Services	Subscription Client Multiple	False
Subscription Services	Subscription Client Publish Configurable	False
Subscription Services	Subscription Client Republish	False
View Services	View Client Basic Browse	False
View Services	View Client Basic ResultSet Filtering	False
View Services	View Client TranslateBrowsePath	False

6.6.109 Method Client Facet

Table 122 describes the details of the *Method Client* Facet. This Facet defines the ability to call arbitrary Methods.

Table 122 – Method Client Facet

Group	Conformance Unit / Profile Title	Optional
Method Services	Method Client Call	False

6.6.110 Auditing Client Facet

Table 123 describes the details of the Auditing *Client* Facet. This Facet defines the ability to monitor Audit Events.

Table 123 – Auditing Client Facet

Group	Conformance Unit / Profile Title	Optional
Profile	Event Subscriber Client Facet	False
Auditing	Auditing Client Audit ID	False
Auditing	Auditing Client Subscribes	False

6.6.111 Node Management Client Facet

Table 124 describes the details of the *Node Management Client* Facet. This Facet defines the ability to configure the *AddressSpace* of an OPC UA Server through OPC UA *Node Management Service Set*.

Table 124 – Node Management Client Facet

Group	Conformance Unit / Profile Title	Optional
Address Space Model	Address Space Client Base	False
Node Management Services	Node Management Client	False

6.6.112 Advanced Type Programming Client Facet

Table 125 describes the details of the Advanced Type Programming *Client* Facet. This Facet defines the ability to use the type model and process the instance *AddressSpace* based on the type model. For example a client may contain generic displays that are based on a type, in that they contain a relative path from some main type. On call up this main type is matched to an instance and all of display items are resolved based on the provided type model.

Table 125 – Advanced Type Programming Client Facet

Group	Conformance Unit / Profile Title	Optional
Address Space Model	Address Space <i>Client</i> Base	False
Base Information	Base Info <i>Client</i> Basic	False
Base Information	Base Info <i>Client</i> Type Programming	False
View <i>Services</i>	View <i>Client</i> TranslateBrowsePath	False

6.6.113 User Role Management Client Facet

Table 126 describes the details of the User Role Management *Client Facet*. This *Facet* defines knowledge of the OPC UA *Information Model* for user roles and permissions and the use of the *Methods* to manage them.

Table 126 – User Role Management Client Facet

Group	Conformance Unit / Profile Title	Optional
Security	Security Role <i>Client</i> Base	False
Security	Security Role <i>Client</i> DefaultRolePermissions	False
Security	Security Role <i>Client</i> Management	False
Security	Security Role <i>Client</i> Restrict Applications	True
Security	Security Role <i>Client</i> Restrict Endpoints	True
Security	Security Role <i>Client</i> RolePermissions	False

6.6.114 State Machine Client Facet

Table 127 describes the details of the State Machine *Client Facet*. This *Facet* defines the ability to use state machines based on the *StateMachineType* or a sub-type.

Table 127 – State Machine Client Facet

Group	Conformance Unit / Profile Title	Optional
Base Information	Base Info Client Available States and Transitions	True
Base Information	Base Info <i>Client</i> Finite State Machine Instance	True
Base Information	Base Info <i>Client</i> State Machine Instance	False

6.6.115 Diagnostic Client Facet

Table 128 describes the details of the Diagnostic *Client Facet*. This *Facet* defines the ability to read and process diagnostic information that is part of the OPC UA information model.

Table 128 – Diagnostic Client Facet

Group	Conformance Unit / Profile Title	Optional
Address Space Model	Address Space <i>Client</i> Base	False
Base Information	Base Info <i>Client</i> Basic	False
Base Information	Base Info <i>Client</i> Diagnostics	False

6.6.116 Redundant Client Facet

Table 129 describes the details of the Redundant *Client* Facet. This Facet defines the ability to use the redundancy feature available for redundant *Clients*.

Table 129 – Redundant Client Facet

Group	Conformance Unit / <i>Profile</i> Title	Optional
Redundancy	Redundancy <i>Client</i>	False
<i>Subscription Services</i>	<i>Subscription Client</i> TransferSubscriptions	True

6.6.117 Redundancy Switch Client Facet

Table 130 describes the details of the Redundancy Switch *Client* Facet. A *Client* that supports this Facet supports monitoring the redundancy status for non-transparent redundant *Servers* and switching to the backup *Server* when they recognize a change.

Table 130 – Redundancy Switch Client Facet

Group	Conformance Unit / <i>Profile</i> Title	Optional
Redundancy	Redundancy <i>Client</i> Switch	False

6.6.118 Historical Access Client Facet

Table 131 describes the details of the Historical Access *Client* Facet. This Facet defines the ability to read, process, and update historical data.

Table 131 – Historical Access Client Facet

Group	Conformance Unit / <i>Profile</i> Title	Optional
<i>Attribute Services</i>	<i>Attribute Client</i> Historical Read	False
Historical Access	Historical Access <i>Client</i> Browse	False
Historical Access	Historical Access <i>Client</i> Read Raw	False

6.6.119 Historical Data AtTime Client Facet

Table 132 describes the details of the Historical Data AtTime *Client* Facet. This Facet defines the ability to access data at specific instances in time.

Table 132 – Historical Data AtTime Client Facet

Group	Conformance Unit / <i>Profile</i> Title	Optional
<i>Profile</i>	Historical Access <i>Client</i> Facet	False
Historical Access	Historical Access <i>Client</i> Time Instance	False

6.6.120 Historical Aggregate Client Facet

Table 133 describes the details of the Historical Aggregate *Client* Facet. This Facet defines the ability to read historical data by specifying the needed aggregate. This implies consideration of the list of aggregates supported by the *Server*.

Table 133 – Historical Aggregate Client Facet

Group	Conformance Unit / Profile Title	Optional
Aggregates	Aggregate – <i>Client</i> AnnotationCount	True
Aggregates	Aggregate – <i>Client</i> Average	True
Aggregates	Aggregate – <i>Client</i> Count	True
Aggregates	Aggregate – <i>Client</i> Custom Aggregates	True
Aggregates	Aggregate – <i>Client</i> Delta	True
Aggregates	Aggregate – <i>Client</i> DeltaBounds	True
Aggregates	Aggregate – <i>Client</i> DurationBad	True
Aggregates	Aggregate – <i>Client</i> DurationGood	True
Aggregates	Aggregate – <i>Client</i> DurationInStateNonZero	True
Aggregates	Aggregate – <i>Client</i> DurationInStateZero	True
Aggregates	Aggregate – <i>Client</i> End	True
Aggregates	Aggregate – <i>Client</i> EndBound	True
Aggregates	Aggregate – <i>Client</i> Interpolative	True
Aggregates	Aggregate – <i>Client</i> Maximum	True
Aggregates	Aggregate – <i>Client</i> Maximum2	True
Aggregates	Aggregate – <i>Client</i> MaximumActualTime	True
Aggregates	Aggregate – <i>Client</i> MaximumActualTime2	True
Aggregates	Aggregate – <i>Client</i> Minimum	True
Aggregates	Aggregate – <i>Client</i> Minimum2	True
Aggregates	Aggregate – <i>Client</i> MinimumActualTime	True
Aggregates	Aggregate – <i>Client</i> MinimumActualTime2	True
Aggregates	Aggregate – <i>Client</i> NumberOfTransitions	True
Aggregates	Aggregate – <i>Client</i> PercentBad	True
Aggregates	Aggregate – <i>Client</i> PercentGood	True
Aggregates	Aggregate – <i>Client</i> Range	True
Aggregates	Aggregate – <i>Client</i> Range2	True
Aggregates	Aggregate – <i>Client</i> StandardDeviationPopulation	True
Aggregates	Aggregate – <i>Client</i> StandardDeviationSample	True
Aggregates	Aggregate – <i>Client</i> Start	True
Aggregates	Aggregate – <i>Client</i> StartBound	True
Aggregates	Aggregate – <i>Client</i> TimeAverage	True
Aggregates	Aggregate – <i>Client</i> TimeAverage2	True
Aggregates	Aggregate – <i>Client</i> Total	True
Aggregates	Aggregate – <i>Client</i> Total2	True
Aggregates	Aggregate – <i>Client</i> Usage	False
Aggregates	Aggregate – <i>Client</i> VariancePopulation	True
Aggregates	Aggregate – <i>Client</i> VarianceSample	True
Aggregates	Aggregate – <i>Client</i> WorstQuality	True
Aggregates	Aggregate – <i>Client</i> WorstQuality2	True
Historical Access	Historical Access <i>Client</i> Read Aggregates	False

6.6.121 Historical Annotation Client Facet

Table 134 describes the details of the Historical Annotation *Client* Facet. This Facet defines the ability to retrieve and write annotations for historical data.

Table 134 – Historical Annotation Client Facet

Group	Conformance Unit / Profile Title	Optional
Profile	Historical Access <i>Client</i> Facet	False
Profile	Historical Data Update <i>Client</i> Facet	False
Historical Access	Historical Access <i>Client</i> Annotations	False

6.6.122 Historical Access Modified Data Client Facet

Table 135 describes the details of the Historical Access Modified Data *Client* Facet. This Facet defines the ability to access prior historical data (values that were modified or inserted).

Table 135 – Historical Access Modified Data Client Facet

Group	Conformance Unit / Profile Title	Optional
Profile	Historical Access <i>Client</i> Facet	False
Historical Access	Historical Access <i>Client</i> Read Modified	False

6.6.123 Historical Data Insert Client Facet

Table 136 describes the details of the Historical Data Insert *Client* Facet. This Facet defines the ability to insert historical data.

Table 136 – Historical Data Insert Client Facet

Group	Conformance Unit / Profile Title	Optional
Attribute Services	Attribute <i>Client</i> Historical Updates	False
Historical Access	Historical Access <i>Client</i> Data Insert	False

6.6.124 Historical Data Update Client Facet

Table 137 describes the details of the Historical Data Update *Client* Facet. This Facet defines the ability to update historical data.

Table 137 – Historical Data Update Client Facet

Group	Conformance Unit / Profile Title	Optional
Attribute Services	Attribute <i>Client</i> Historical Updates	False
Historical Access	Historical Access <i>Client</i> Data Update	False

6.6.125 Historical Data Replace Client Facet

Table 138 describes the details of the Historical Data Replace *Client* Facet. This Facet defines the ability to replace historical data.

Table 138 – Historical Data Replace Client Facet

Group	Conformance Unit / Profile Title	Optional
<i>Attribute Services</i>	<i>Attribute Client</i> Historical Updates	False
Historical Access	Historical Access <i>Client</i> Data Replace	False

6.6.126 Historical Data Delete Client Facet

Table 139 describes the details of the Historical Data Delete *Client* Facet. This Facet defines the ability to delete historical data.

Table 139 – Historical Data Delete Client Facet

Group	Conformance Unit / Profile Title	Optional
<i>Attribute Services</i>	<i>Attribute Client</i> Historical Updates	False
Historical Access	Historical Access <i>Client</i> Data Delete	False

6.6.127 Historical Access Client Server Timestamp Facet

Table 140 describes the details of the Historical Access *Client Server* Timestamp Facet. This Facet defines the ability to request and process *Server* timestamps, in addition to source timestamps.

Table 140 – Historical Access Client Server Timestamp Facet

Group	Conformance Unit / Profile Title	Optional
Historical Access	Historical Access <i>Client Server</i> Timestamp	False

6.6.128 Historical Structured Data Access Client Facet

Table 141 describes the details of the Historical Structured Data Access *Client* Facet. This Facet defines the ability to read structured values for historical nodes.

Table 141 – Historical Structured Data Access Client Facet

Group	Conformance Unit / Profile Title	Optional
<i>Profile</i>	Historical Access <i>Client</i> Facet	False
Historical Access	Historical Access <i>Client</i> Structure Data Raw	False

6.6.129 Historical Structured Data AtTime Client Facet

Table 142 describes the details of the Historical Structured Data AtTime *Client* Facet. This Facet defines the ability to read structured values for historical nodes at specific instances in time.

Table 142 – Historical Structured Data AtTime Client Facet

Group	Conformance Unit / Profile Title	Optional
<i>Profile</i>	Historical Data AtTime <i>Client</i> Facet	False
Historical Access	Historical Access <i>Client</i> Structure Data Time Instance	False

6.6.130 Historical Structured Data Modified Client Facet

Table 143 describes the details of the Historical Structured Data Modified *Client* Facet. This Facet defines the ability to read structured values for prior historical data (values that were modified or inserted).

Table 143 – Historical Structured Data Modified Client Facet

Group	Conformance Unit / Profile Title	Optional
Profile	Historical Access Modified Data <i>Client</i> Facet	False
Historical Access	Historical Access <i>Client</i> Structure Data Read Modified	False

6.6.131 Historical Structured Data Insert Client Facet

Table 144 describes the details of the Historical Structured Data Insert *Client* Facet. This Facet defines the ability to insert structured historical data.

Table 144 – Historical Structured Data Insert Client Facet

Group	Conformance Unit / Profile Title	Optional
Profile	Historical Data Insert <i>Client</i> Facet	False
Historical Access	Historical Access <i>Client</i> Structure Data Insert	False

6.6.132 Historical Structured Data Update Client Facet

Table 145 describes the details of the Historical Structured Data Update *Client* Facet. This Facet defines the ability to update structured historical data.

Table 145 – Historical Structured Data Update Client Facet

Group	Conformance Unit / Profile Title	Optional
Profile	Historical Data Update <i>Client</i> Facet	False
Historical Access	Historical Access <i>Client</i> Structure Data Update	False

6.6.133 Historical Structured Data Replace Client Facet

Table 146 describes the details of the Historical Structured Data Replace *Client* Facet. This Facet defines the ability to replace structured historical data.

Table 146 – Historical Structured Data Replace Client Facet

Group	Conformance Unit / Profile Title	Optional
Profile	Historical Data Replace <i>Client</i> Facet	False
Historical Access	Historical Access <i>Client</i> Structure Data Replace	False

6.6.134 Historical Structured Data Delete Client Facet

Table 147 describes the details of the Historical Structured Data Delete *Client* Facet. This Facet defines the ability to remove structured historical data.

Table 147 – Historical Structured Data Delete Client Facet

Group	Conformance Unit / <i>Profile</i> Title	Optional
<i>Profile</i>	Historical Data Delete <i>Client</i> Facet	False
Historical Access	Historical Access <i>Client</i> Structure Data Delete	False

6.6.135 Historical Events Client Facet

Table 148 describes the details of the Historical Events *Client* Facet. This Facet defines the ability to read Historical Events, including simple filtering.

Table 148 – Historical Events Client Facet

Group	Conformance Unit / <i>Profile</i> Title	Optional
<i>Attribute Services</i>	<i>Attribute Client</i> Historical Read	False
Historical Access	Historical Access <i>Client</i> Read Events	False

6.6.136 Historical Event Insert Client Facet

Table 149 describes the details of the Historical *Event* Insert *Client* Facet. This Facet defines the ability to insert historical events.

Table 149 – Historical Event Insert Client Facet

Group	Conformance Unit / <i>Profile</i> Title	Optional
<i>Attribute Services</i>	<i>Attribute Client</i> Historical Updates	False
Historical Access	Historical Access <i>Client</i> Event Inserts	False

6.6.137 Historical Event Update Client Facet

Table 150 describes the details of the Historical *Event* Update *Client* Facet. This Facet defines the ability to update historical events.

Table 150 – Historical Event Update Client Facet

Group	Conformance Unit / <i>Profile</i> Title	Optional
<i>Attribute Services</i>	<i>Attribute Client</i> Historical Updates	False
Historical Access	Historical Access <i>Client</i> Event Updates	False

6.6.138 Historical Event Replace Client Facet

Table 151 describes the details of the Historical *Event* Replace *Client* Facet. This Facet defines the ability to replace historical events.

Table 151 – Historical Event Replace Client Facet

Group	Conformance Unit / <i>Profile</i> Title	Optional
<i>Attribute Services</i>	<i>Attribute Client</i> Historical Updates	False
Historical Access	Historical Access <i>Client</i> Event Replaces	False

6.6.139 Historical Event Delete Client Facet

Table 152 describes the details of the Historical *Event Delete Client* Facet. This Facet defines the ability to delete Historical events.

Table 152 – Historical Event Delete Client Facet

Group	Conformance Unit / Profile Title	Optional
<i>Attribute Services</i>	<i>Attribute Client</i> Historical Updates	False
Historical Access	Historical Access <i>Client Event</i> Deletes	False

6.6.140 Aggregate Subscriber Client Facet

Table 153 describes the details of the Aggregate Subscriber *Client* Facet. This Facet defines the ability to use the aggregate filter when subscribing for *Attribute* values.

Table 153 – Aggregate Subscriber Client Facet

Group	Conformance Unit / Profile Title	Optional
Aggregates	Aggregate <i>Subscription – Client</i> AnnotationCount	True
Aggregates	Aggregate <i>Subscription – Client</i> Average	True
Aggregates	Aggregate <i>Subscription – Client</i> Count	True
Aggregates	Aggregate <i>Subscription – Client</i> Custom Aggregates	True
Aggregates	Aggregate <i>Subscription – Client</i> Delta	True
Aggregates	Aggregate <i>Subscription – Client</i> DeltaBounds	True
Aggregates	Aggregate <i>Subscription – Client</i> DurationBad	True
Aggregates	Aggregate <i>Subscription – Client</i> DurationGood	True
Aggregates	Aggregate <i>Subscription – Client</i> DurationInStateNonZero	True
Aggregates	Aggregate <i>Subscription – Client</i> DurationInStateZero	True
Aggregates	Aggregate <i>Subscription – Client</i> End	True
Aggregates	Aggregate <i>Subscription – Client</i> EndBound	True
Aggregates	Aggregate <i>Subscription – Client</i> Filter	False
Aggregates	Aggregate <i>Subscription – Client</i> Interpolative	True
Aggregates	Aggregate <i>Subscription – Client</i> Maximum	True
Aggregates	Aggregate <i>Subscription – Client</i> Maximum2	True
Aggregates	Aggregate <i>Subscription – Client</i> MaximumActualTime	True
Aggregates	Aggregate <i>Subscription – Client</i> MaximumActualTime2	True
Aggregates	Aggregate <i>Subscription – Client</i> Minimum	True
Aggregates	Aggregate <i>Subscription – Client</i> Minimum2	True
Aggregates	Aggregate <i>Subscription – Client</i> MinimumActualTime	True
Aggregates	Aggregate <i>Subscription – Client</i> MinimumActualTime2	True
Aggregates	Aggregate <i>Subscription – Client</i> NumberOfTransitions	True
Aggregates	Aggregate <i>Subscription – Client</i> PercentBad	True
Aggregates	Aggregate <i>Subscription – Client</i> PercentGood	True
Aggregates	Aggregate <i>Subscription – Client</i> Range	True
Aggregates	Aggregate <i>Subscription – Client</i> Range2	True
Aggregates	Aggregate <i>Subscription – Client</i> StandardDeviationPopulation	True

Group	Conformance Unit / Profile Title	Optional
Aggregates	Aggregate <i>Subscription</i> – <i>Client</i> StandardDeviationSample	True
Aggregates	Aggregate <i>Subscription</i> – <i>Client</i> Start	True
Aggregates	Aggregate <i>Subscription</i> – <i>Client</i> StartBound	True
Aggregates	Aggregate <i>Subscription</i> – <i>Client</i> TimeAverage	True
Aggregates	Aggregate <i>Subscription</i> – <i>Client</i> TimeAverage2	True
Aggregates	Aggregate <i>Subscription</i> – <i>Client</i> Total	True
Aggregates	Aggregate <i>Subscription</i> – <i>Client</i> Total2	True
Aggregates	Aggregate <i>Subscription</i> – <i>Client</i> VariancePopulation	True
Aggregates	Aggregate <i>Subscription</i> – <i>Client</i> VarianceSample	True
Aggregates	Aggregate <i>Subscription</i> – <i>Client</i> WorstQuality	True
Aggregates	Aggregate <i>Subscription</i> – <i>Client</i> WorstQuality2	True
Monitored Item <i>Services</i>	Monitor <i>Client</i> Aggregate Filter	False
Monitored Item <i>Services</i>	Monitor <i>Client</i> by Index	False
Monitored Item <i>Services</i>	Monitor <i>Client</i> Modify	True
Monitored Item <i>Services</i>	Monitor <i>Client</i> Value Change	False
<i>Subscription Services</i>	<i>Subscription Client</i> Basic	False
<i>Subscription Services</i>	<i>Subscription Client</i> Modify	True
<i>Subscription Services</i>	<i>Subscription Client</i> Multiple	True
<i>Subscription Services</i>	<i>Subscription Client</i> Republish	True

6.6.141 Standard UA Client Profile

Standard UA Client Profile has been deprecated as it is superseded with Standard UA Client 2017 Profile in v1.04.

6.6.142 Standard UA Client 2017 Profile

Table 154 describes the details of the Standard UA *Client* 2017 *Profile*. This *Profile* is a FullFeatured *Profile* that defines a minimum set of functionality required for generic OPC UA *Clients*. Such a *Client* shall be able to use local, subnet and global discovery. It shall be able to maintain a connection with a single *Session* (as required for nano embedded *Servers*). If *Subscriptions* are used, the *Client* shall respect the limits of *Servers* with limited resources. If a *Server* does not support *Subscriptions*, the *Client* shall provide read access as fallback. The *Client* shall provide connection establishment through the OPC UA TCP binary protocol with and without security.

This *Profile* supersedes the "Standard UA *Client Profile*".

Table 154 – Standard UA Client 2017 Profile

Group	Conformance Unit / Profile Title	Optional
Profile	AddressSpace Lookup Client Facet	False
Profile	Attribute Read Client Facet	False
Profile	Attribute Write Client Facet	False
Profile	Base Client Behaviour Facet	False
Profile	Core 2017 Client Facet	False
Profile	DataChange Subscriber Client Facet	False
Profile	Discovery Client Facet	False
Profile	Entry Level Support 2015 Client Facet	False
Profile	Global Certificate Management Client Facet	False
Profile	Global Discovery Client Facet	False
Profile	Method Client Facet	False
Profile	Subnet Discovery Client Facet	False
Profile	UA-TCP UA-SC UA-Binary	False
Profile	User Token – Anonymous Facet	False

6.6.143 UA-TCP UA-SC UA-Binary

Table 155 describes the details of the UA-TCP UA-SC UA-Binary. This transport Facet defines a combination of network protocol, security protocol and message encoding that is optimized for low resource consumption and high performance. It combines the simple TCP based network protocol UA-TCP 1.0 with the binary security protocol UA-SecureConversation 1.0 and the binary message encoding UA-Binary 1.0.

Table 155 – UA-TCP UA-SC UA-Binary

Group	Conformance Unit / Profile Title	Optional
Protocol and Encoding	Protocol UA TCP	False
Protocol and Encoding	UA Binary Encoding	False
Protocol and Encoding	UA Secure Conversation	False

6.6.144 HTTPS UA-Binary

Table 156 describes the details of the HTTPS UA-Binary. This transport Facet defines a combination of network protocol, security protocol and message encoding that balances compatibility with widely used HTTPS transport and a compact UA-Binary encoded message for added performance. It is expected that this transport will be used to support installations where firewalls only permit HTTPS or where a WEB browser is used as *Client*. This transport requires that one of the TransportSecurity Profiles for TLS be provided.

Table 156 – HTTPS UA-Binary

Group	Conformance Unit / Profile Title	Optional
Protocol and Encoding	Protocol HTTPS	False
Protocol and Encoding	UA Binary Encoding	False
Security	Security TLS General	False

6.6.145 HTTPS UA-XML

Table 157 describes the details of the HTTPS UA-XML. This transport Facet defines a combination of network protocol, security protocol and message encoding that uses HTTPS transport and a SOAP XML encoded message for use with standard SOAP V1.2 toolkits. This transport requires that one of the TransportSecurity *Profiles* for TLS be provided.

Table 157 – HTTPS UA-XML

Group	Conformance Unit / Profile Title	Optional
Protocol and Encoding	Protocol HTTPS	False
Protocol and Encoding	UA SOAP-XML Encoding	False
Security	Security TLS General	False

6.6.146 HTTPS UA-JSON

Table 158 describes the details of the HTTPS UA-JSON. This transport Facet defines a combination of network protocol, security protocol and message encoding that uses HTTPS transport and a UA-JSON encoded message. This transport requires that one of the TransportSecurity *Profiles* for TLS be provided.

Table 158 – HTTPS UA-JSON

Group	Conformance Unit / Profile Title	Optional
Protocol and Encoding	JSON Reversible Encoding	False
Protocol and Encoding	Protocol HTTPS	False
Security	Security TLS General	False

6.6.147 WSS UA-SC UA-Binary

Table 159 describes the details of the WSS UA-SC UA-Binary. This transport Facet defines a combination of network protocol, security protocol and message encoding that uses WSS transport as a tunnel for UA-SecureConversation and UA-Binary encoded messages. Although transport security is available in WSS via TLS, additional message security can be used to assure end-to-end security.

Table 159 – WSS UA-SC UA-Binary

Group	Conformance Unit / Profile Title	Optional
Protocol and Encoding	Protocol Web Sockets	False
Protocol and Encoding	UA Binary Encoding	False
Protocol and Encoding	UA Secure Conversation	False
Security	Security TLS General	False

6.6.148 WSS UA-JSON

Table 160 describes the details of the WSS UA-JSON. This transport Facet defines a combination of network protocol, security protocol and message encoding that uses WSS transport with UA-JSON encoded messages.

Table 160 – WSS UA-JSON

Group	Conformance Unit / Profile Title	Optional
Protocol and Encoding	JSON Reversible Encoding	False
Protocol and Encoding	Protocol Web Sockets	False
Security	Security TLS General	False

6.6.149 Security User Access Control Full

Table 161 describes the details of the Security User Access Control Full. A *Server* that supports this profile supports restricting multiple levels of access to all *Nodes* in the *AddressSpace* based on the validated user.

Table 161 – Security User Access Control Full

Group	Conformance Unit / Profile Title	Optional
Profile	Security User Access Control Base	False
Address Space Model	Address Space User Access Level Full	False

6.6.150 Security User Access Control Base

Table 162 describes the details of the Security User Access Control Base. A *Server* that supports this profile supports restricting some level of access to some *Nodes* in the *AddressSpace* based on the validated user.

Table 162 – Security User Access Control Base

Group	Conformance Unit / Profile Title	Optional
Address Space Model	Address Space User Access Level Base	False
Security	Security User IssuedToken Kerberos	True
Security	Security User IssuedToken Kerberos Windows	True
Security	Security User Name Password	False
Security	Security User X509	True

6.6.151 Security Time Synchronization

Table 163 describes the details of the Security Time Synchronization. This Facet indicates that the application supports the minimum required level of time synchronization to ensure secure communication. One of the optional time synchronization conformance units shall be supported.

Table 163 – Security Time Synchronization

Group	Conformance Unit / Profile Title	Optional
Security	Security Time Synch – Configuration	False
Security	Security Time Synch – NTP / OS Based support	True
Security	Security Time Synch – UA based support	True

6.6.152 Best Practice – Audit Events

Table 164 describes the details of the Best Practice – Audit Events. Subscriptions for Audit Events shall be restricted to authorized personnel.

Table 164 – Best Practice – Audit Events

Group	Conformance Unit / Profile Title	Optional
Miscellaneous	Best Practice – Audit Events	False

6.6.153 Best Practice – Alarm Handling

Table 165 describes the details of the Best Practice – *Alarm* Handling. A *Server* should restrict critical alarm handling functionality to users that have the appropriate rights to perform these actions.

Table 165 – Best Practice – Alarm Handling

Group	Conformance Unit / Profile Title	Optional
Miscellaneous	Best Practice – <i>Alarm</i> Handling	False

6.6.154 Best Practice – Random Numbers

Table 166 describes the details of the Best Practice – Random Numbers. All random numbers that are required for security should use appropriate cryptographic library based random number generators.

Table 166 – Best Practice – Random Numbers

Group	Conformance Unit / Profile Title	Optional
Miscellaneous	Best Practice – Random Numbers	False

6.6.155 Best Practice – Timeouts

Table 167 describes the details of the Best Practice – Timeouts. The administrator should be able to configure reasonable timeouts for Secure Channels, Sessions and Subscriptions. Setting these timeouts allows limiting Denial of *Service* attacks and overload issues.

Table 167 – Best Practice – Timeouts

Group	Conformance Unit / Profile Title	Optional
Miscellaneous	Best Practice – Timeouts	False

6.6.156 Best Practice – Administrative Access

Table 168 describes the details of the Best Practice – Administrative Access. The *Server* and *Client* allow restricting the use of certain *Services* and access to parts of the *AddressSpace* to administrative personnel. This includes multiple level of administrative access on platforms that support multiple administrative roles (such as Windows or Linux).

Table 168 – Best Practice – Administrative Access

Group	Conformance Unit / Profile Title	Optional
Miscellaneous	Best Practice – Administrative Access	False

6.6.157 Best Practice – Strict Message Handling

Table 169 describes the details of the Best Practice – Strict *Message* Handling. *Server* and *Client* reject messages that are incorrectly formed as specified in IEC 62541-4 and IEC 62541-6.

Table 169 – Best Practice – Strict Message Handling

Group	Conformance Unit / Profile Title	Optional
Miscellaneous	Best Practice – Strict <i>Message</i> Handling	False

6.6.158 Best Practice – Audit Events Client

Table 170 describes the details of the Best Practice – Audit Events *Client*. Audit Tracking system connect to a *Server* using a Secure Channel and under the appropriate authorization to allow access to Audit Events.

Table 170 – Best Practice – Audit Events Client

Group	Conformance Unit / Profile Title	Optional
Miscellaneous	Best Practice – Audit Events <i>Client</i>	False

6.6.159 TransportSecurity – TLS 1.2

Table 171 describes the details of the TransportSecurity – TLS 1.2. This Facet defines a transport security for configurations with high security needs. It makes use of TLS 1.2 and uses TLS_RSA_WITH_AES_256_CBC_SHA256. As computing power increases, security algorithms are expected to expire. NIST provides guidelines for expected expiration dates for individual algorithms. These guidelines provide recommended dates at which the algorithm should be replaced or upgraded to a more secure algorithm. They do not indicate a failure of the algorithm. NIST has no recommendations for this TransportSecurity. It is recommended that *Servers* and *Clients* support all security profiles and developers provide the recommended profile as a default. It is up to an administrator to configure the actual exposed TransportSecurity *Profiles*.

Table 171 – TransportSecurity – TLS 1.2

Group	Conformance Unit / Profile Title	Optional
Security	Security TLS_RSA with AES_256_CBC_SHA256	False

6.6.160 TransportSecurity – TLS 1.2 with PFS

Table 172 describes the details of the TransportSecurity – TLS 1.2 with PFS. This Facet defines a transport security for configurations with high security needs and perfect forward secrecy (PFS). It makes use of TLS 1.2 and uses TLS_DHE_RSA_WITH_AES_128_CBC_SHA256 or TLS_DHE_RSA_WITH_AES_256_CBC_SHA256.

As computing power increases, security algorithms are expected to expire. NIST provides guidelines for expected expiration dates for individual algorithms. These guidelines provide recommended dates at which the algorithm should be replaced or upgraded to a more secure algorithm. They do not indicate a failure of the algorithm. NIST has no recommendations for this TransportSecurity. It is recommended that *Servers* and *Clients* support all security profiles and developers provide the recommended profile as a default. It is up to an administrator to configure the actual exposed TransportSecurity *Profiles*.

Table 172 – TransportSecurity – TLS 1.2 with PFS

Group	Conformance Unit / Profile Title	Optional
Security	Security TLS_DHE_RSA with AES_nnn_CBC_SHA256	False

6.6.161 SecurityPolicy – None

Table 173 describes the details of the SecurityPolicy – None. This security Facet defines a security policy used for configurations with the lowest security needs. This security policy can affect the behaviour of the *CreateSession* and *ActivateSession* Services. It also results in a SecureChannel which has no channel security. By default this security policy should be disabled if any other security policies are available.

Table 173 – SecurityPolicy – None

Group	Conformance Unit / Profile Title	Optional
Security	AsymmetricEncryptionAlgorithm_None	False
Security	AsymmetricSignatureAlgorithm_None	False
Security	KeyDerivationAlgorithm_None	False
Security	Security None CreateSession ActivateSession	False
Security	Security None CreateSession ActivateSession 1.0	True
Security	SecurityPolicy_None Limits	False
Security	SymmetricEncryptionAlgorithm_None	False
Security	SymmetricSignatureAlgorithm_None	False

6.6.162 SecurityPolicy – Basic128Rsa15

SecurityPolicy – Basic128Rsa15 has been deprecated in v1.04 since the hash algorithm Sha-1 is not considered secure anymore.

6.6.163 SecurityPolicy – Basic256

SecurityPolicy – Basic128Rsa15 has been deprecated in v1.04 since the hash algorithm Sha-1 is not considered secure anymore.

6.6.164 SecurityPolicy [A] – Aes128-Sha256-RsaOaep

Table 174 describes the details of the SecurityPolicy [A] – Aes128-Sha256-RsaOaep. This security Facet defines a security policy for configurations with average security needs. It requires a PKI infrastructure. As computing power increases, security policies are expected to expire. NIST provides guidelines for expected expiration dates for individual algorithms. These guidelines provide recommended dates at which the algorithm should be replaced or upgraded to a more secure algorithm. They do not indicate a failure of the algorithm. This security policy has no published end dates as of this time. It is recommended that *Servers* and *Clients* support all security profiles and support configurability of actual exposed and default security policies.

Table 174 – SecurityPolicy [A] – Aes128-Sha256-RsaOaep

Group	Conformance Unit / Profile Title	Optional
Security	Aes128-Sha256-RsaOaep_Limits	False
Security	AsymmetricEncryptionAlgorithm_RSA-OAEP-SHA1	False
Security	AsymmetricSignatureAlgorithm_RSA-PKCS15-SHA2-256	False
Security	CertificateSignatureAlgorithm_RSA-PKCS15-SHA2-256	False
Security	KeyDerivationAlgorithm_P-SHA2-256	False
Security	Security <i>Certificate</i> Validation	False
Security	Security Encryption Required	False
Security	Security Signing Required	False
Security	SymmetricEncryptionAlgorithm_AES128-CBC	False
Security	SymmetricSignatureAlgorithm_HMAC-SHA2-256	False

6.6.165 SecurityPolicy [B] – Basic256Sha256

Table 175 describes the details of the SecurityPolicy [B] – Basic256Sha256. This security Facet defines a security policy for configurations with high security needs. It requires a PKI infrastructure.

As computing power increases, security policies are expected to expire. NIST provides guidelines for expected expiration dates for individual algorithms. These guidelines provided recommended dates at which the algorithm should be replaced or upgraded to a more secure algorithm. They do not indicate a failure of the algorithm. This security policy has no published end dates as of this time. It is recommended that *Servers* and *Clients* support all security profiles and developers provide the recommended profile as a default. It is up to an administrator to configure the actual exposed security policies.

Table 175 – SecurityPolicy [B] – Basic256Sha256

Group	Conformance Unit / Profile Title	Optional
Security	AsymmetricEncryptionAlgorithm_RSA-OAEP-SHA1	False
Security	AsymmetricSignatureAlgorithm_RSA-PKCS15-SHA2-256	False
Security	Basic256Sha256_Limits	False
Security	CertificateSignatureAlgorithm_RSA-PKCS15-SHA2-256	False
Security	KeyDerivationAlgorithm_P-SHA2-256	False
Security	Security <i>Certificate</i> Validation	False
Security	Security Encryption Required	False
Security	Security Signing Required	False
Security	SymmetricEncryptionAlgorithm_AES256-CBC	False
Security	SymmetricSignatureAlgorithm_HMAC-SHA2-256	False

6.6.166 SecurityPolicy – Aes256-Sha256-RsaPss

Table 176 describes the details of the SecurityPolicy – Aes256-Sha256-RsaPss. This security Facet defines a security policy for configurations with a need for high security. It requires a PKI infrastructure. As computing power increases, security policies are expected to expire. NIST provides guidelines for expected expiration dates for individual algorithms. These guidelines provide recommended dates at which the algorithm should be replaced or upgraded to a more secure algorithm. They do not indicate a failure of the algorithm. This security policy has no published end dates as of this time. It is recommended that *Servers*

and *Clients* support all security profiles and support configurability of actual exposed and default security policies.

Table 176 – SecurityPolicy – Aes256-Sha256-RsaPss

Group	Conformance Unit / Profile Title	Optional
Security	Aes256-Sha256-RsaPss_Limits	False
Security	AsymmetricEncryptionAlgorithm_RSA-OAEP-SHA2-256	False
Security	AsymmetricSignatureAlgorithm_RSA-PSS -SHA2-256	False
Security	CertificateSignatureAlgorithm_RSA-PKCS15-SHA2-256	False
Security	KeyDerivationAlgorithm_P-SHA2-256	False
Security	Security <i>Certificate</i> Validation	False
Security	Security Encryption Required	False
Security	Security Signing Required	False
Security	SymmetricEncryptionAlgorithm_AES256-CBC	False
Security	SymmetricSignatureAlgorithm_HMAC-SHA2-256	False

6.6.167 User Token – Anonymous Facet

Table 177 describes the details of the User Token – Anonymous Facet. This Facet indicates that anonymous User Tokens are supported.

Table 177 – User Token – Anonymous Facet

Group	Conformance Unit / Profile Title	Optional
Security	Security User Anonymous	False

6.6.168 User Token – User Name Password Server Facet

Table 178 describes the details of the User Token – User Name Password *Server* Facet. This Facet indicates that a user token that is comprised of a username and password is supported. This user token can affect the behaviour of the *ActivateSession Service*.

Table 178 – User Token – User Name Password Server Facet

Group	Conformance Unit / Profile Title	Optional
Security	Security Invalid user token	False
Security	Security User Name Password	False

6.6.169 User Token – X509 Certificate Server Facet

Table 179 describes the details of the User Token – X509 *Certificate Server* Facet. This Facet indicates that the use of an X509 certificates to identify users is supported.

Table 179 – User Token – X509 Certificate Server Facet

Group	Conformance Unit / Profile Title	Optional
Security	Security Invalid user token	False
Security	Security User X509	False

6.6.170 User Token – Issued Token Server Facet

Table 180 describes the details of the User Token – Issued Token *Server* Facet. This Facet indicates that a User Token that is comprised of an issued token is supported.

Table 180 – User Token – Issued Token Server Facet

Group	Conformance Unit / <i>Profile</i> Title	Optional
Security	Security Invalid user token	False
Security	Security User IssuedToken Kerberos	False

6.6.171 User Token – Issued Token Windows Server Facet

Table 181 describes the details of the User Token – Issued Token Windows *Server* Facet. This Facet further refines the User Token – Issued Token to indicate a windows implementation of Kerberos

Table 181 – User Token – Issued Token Windows Server Facet

Group	Conformance Unit / <i>Profile</i> Title	Optional
<i>Profile</i>	User Token – Issued Token <i>Server</i> Facet	False
Security	Security User IssuedToken Kerberos Windows	False

6.6.172 User Token – JWT Server Facet

Table 182 describes the details of the User Token – JWT *Server* Facet. This Facet defines support for JSON Web Tokens (JWT) to identify the user during *Session* setup. A JWT is the Access Token format which OPC UA requires when using OAuth2.

Table 182 – User Token – JWT Server Facet

Group	Conformance Unit / <i>Profile</i> Title	Optional
Security	Azure Identity Provider Authority <i>Profile</i>	True
Security	OAuth2 Authority <i>Profile</i>	True
Security	OPC UA Authority <i>Profile</i>	True
Security	Security Invalid user token	False
Security	Security User JWT IssuedToken	False
Security	Security User JWT Token Policy	False

6.6.173 User Token – User Name Password Client Facet

Table 183 describes the details of the User Token – User Name Password *Client* Facet. This Facet defines the ability to use a user token that is comprised of a username and password.

Table 183 – User Token – User Name Password Client Facet

Group	Conformance Unit / <i>Profile</i> Title	Optional
Security	Security User Name Password <i>Client</i>	False

6.6.174 User Token – X509 Certificate Client Facet

Table 184 describes the details of the User Token – X509 *Certificate Client* Facet. This Facet defines the ability to use an X509 certificates to identify users.

Table 184 – User Token – X509 Certificate Client Facet

Group	Conformance Unit / Profile Title	Optional
Security	Security User X509 <i>Client</i>	False

6.6.175 User Token – Issued Token Client Facet

Table 185 describes the details of the User Token – Issued Token *Client* Facet. This Facet defines the ability to use the User Token – Issued Token (Kerberos) to connect to a *Server*.

Table 185 – User Token – Issued Token Client Facet

Group	Conformance Unit / Profile Title	Optional
Security	Security User IssuedToken Kerberos <i>Client</i>	False

6.6.176 User Token – Issued Token Windows Client Facet

Table 186 describes the details of the User Token – Issued Token Windows *Client* Facet. This Facet defines the ability to use the User Token – Issued Token (Windows implementation of Kerberos) to connect to a *Server*

Table 186 – User Token – Issued Token Windows Client Facet

Group	Conformance Unit / Profile Title	Optional
Security	Security User IssuedToken Kerberos Windows <i>Client</i>	False

6.6.177 User Token – JWT Client Facet

Table 187 describes the details of the User Token – JWT *Client* Facet. This Facet defines the ability to use JSON Web Tokens (JWT) as user identification during *Session* setup. JWTs are used to request an access token from an external Authorization *Service*.

Table 187 – User Token – JWT Client Facet

Group	Conformance Unit / Profile Title	Optional
Security	Azure Identity Provider Authority <i>Profile</i>	True
Security	OAuth2 Authority <i>Profile</i>	True
Security	OPC UA Authority <i>Profile</i>	True
Security	Security User JWT IssuedToken <i>Client</i>	False
Security	Security User JWT Token Policy <i>Client</i>	False

6.6.178 Global Discovery Server Profile

Global Discovery Server Profile has been deprecated as it is superseded with Global Discovery Server 2017 Profile in v1.04.

6.6.179 Global Discovery Server 2017 Profile

Table 188 describes the details of the *Global Discovery Server 2017 Profile*. This *Profile* is a *FullFeatured Profile* that covers the necessary *Services* and Information Model of a *UA Server* that acts as a *GDS*.

This *Profile* supersedes the "*Global Discovery Server Profile*".

Table 188 – Global Discovery Server 2017 Profile

Group	Conformance Unit / Profile Title	Optional
Profile	Core 2017 Server Facet	False
Profile	Method Server Facet	False
Profile	Standard DataChange Subscription 2017 Server Facet	False
Profile	UA-TCP UA-SC UA-Binary	False
GDS	GDS Application Directory	False
GDS	GDS LDS-ME Connectivity	False
GDS	GDS Query Applications	False
Security	Security Default ApplicationInstance Certificate	False
Security	Security Policy Required	False
Session Services	Session Minimum 50 Parallel	False

6.6.180 Global Discovery and Certificate Management Server

Global Discovery and Certificate Management Server Profile has been deprecated as it is superseded with Global Discovery and Certificate Mgmt 2017 Server Profile in v1.04.

6.6.181 Global Discovery and Certificate Mgmt 2017 Server

Table 189 describes the details of the *Global Discovery and Certificate Mgmt 2017 Server*. This *Profile* is a *FullFeatured Profile* that covers the necessary *Services* and Information Model of a *UA Server* that acts as a *GDS* and a global *Certificate Manager*.

This *Profile* supersedes the "*Global Discovery and Certificate Management Server*".

Table 189 – Global Discovery and Certificate Mgmt 2017 Server

Group	Conformance Unit / Profile Title	Optional
Profile	Auditing Server Facet	False
Profile	File Access Server Facet	False
Profile	Global Discovery Server 2017 Profile	False
Profile	Standard Event Subscription Server Facet	False
GDS	GDS Certificate Manager Pull Model	False

6.6.182 Global Certificate Management Client Profile

Global Certificate Management Client Profile has been deprecated as it is superseded with Global Certificate Management Client 2017 Profile in v1.04.

6.6.183 Global Certificate Management Client 2017 Profile

Table 190 describes the details of the *Global Certificate Management Client 2017 Profile*. This *Profile* is a *FullFeatured Profile* that uses the Push Model for the management of *Certificates* and Trust Lists.

This *Profile* supersedes the "Global Certificate Management Client Profile".

Table 190 – Global Certificate Management Client 2017 Profile

Group	Conformance Unit / Profile Title	Optional
Profile	Core 2017 Client Facet	False
Profile	Discovery Client Facet	False
Profile	Entry Level Support 2015 Client Facet	False
Profile	File Access Client Facet	False
Profile	Method Client Facet	False
Profile	UA-TCP UA-SC UA-Binary	False
GDS	GDS Certificate Manager Push Model	False
Security	Security Default ApplicationInstance Certificate	False

6.6.184 Global Service Authorization Request Server Facet

Table 191 describes the details of the Global Service Authorization Request Server Facet. This Facet defines the capability of a *Server* (like a GDS) to provide access tokens to OPC UA *Clients* via an Authorization Service as defined in IEC 62541-12.

Table 191 – Global Service Authorization Request Server Facet

Group	Conformance Unit / Profile Title	Optional
GDS	GDS Authorization Service Server	False

6.6.185 Global Service KeyCredential Pull Facet

Table 192 describes the details of the Global Service KeyCredential Pull Facet. This Facet requires providing the *Information Model* for Pull Management as defined in IEC 62541-12. For example KeyCredentials are needed to access an Authorization Service or a Broker. OPC UA *Clients* use this Information Model to request and update KeyCredentials they need.

Table 192 – Global Service KeyCredential Pull Facet

Group	Conformance Unit / Profile Title	Optional
GDS	GDS Key Credential Service Pull Model	False

6.6.186 Global Service KeyCredential Push Facet

Table 193 describes the details of the Global Service KeyCredential Push Facet. This Facet requires the use of KeyCredential Push Management functions to set or update credentials in an OPC UA Server. For example KeyCredentials are needed to access an Authorization Service or a Broker. This OPC UA Server in turn has to provide the KeyCredentialConfigurationType *Objects* that represent required credentials.

Table 193 – Global Service KeyCredential Push Facet

Group	Conformance Unit / Profile Title	Optional
GDS	GDS Key Credential Service Push Model	False

Bibliography

Test Specifications

Compliance Part 8 UA Server: *OPC Test Lab Specification – Part 8 – UA Server*

Compliance Part 9 UA Client: *OPC Test Lab Specification – Part 9 – UA Client*

IECNORM.COM : Click to view the full PDF of IEC 62541-7:2020 RLV

SOMMAIRE

AVANT-PROPOS	136
1 Domaine d'application	139
2 Références normatives	139
3 Termes, définitions et termes abrégés	140
3.1 Termes et définitions	140
3.2 Termes abrégés	141
4 Vue d'ensemble	141
4.1 Généralités	141
4.2 ConformanceUnit	142
4.3 Profils	142
4.4 Catégories de profils	143
5 Unités de Conformité	143
5.1 Vue d'ensemble	143
5.2 Services	144
5.3 Caractéristiques relatives au transport et à la communication	155
5.4 Modèle d'information et caractéristiques relatives à l'AddressSpace	164
5.5 Divers	182
6 Profils	184
6.1 Vue d'ensemble	184
6.2 Liste des profils	184
6.3 Conventions applicables aux définitions des profils	191
6.4 Versionnage des Profils	191
6.5 Applications	191
6.6 Tableaux de Profils	193
6.6.1 Généralités	193
6.6.2 Facette Serveur principal	193
6.6.3 Facette Serveur 2017 principal	193
6.6.4 Facette Serveur Sans session	194
6.6.5 Facette Serveur Connexion inversée	195
6.6.6 Facette Serveur Comportement de base du serveur	195
6.6.7 Facette Serveur Demande modification d'état	195
6.6.8 Facette Serveur Découverte sous-réseau	195
6.6.9 Facette Serveur Gestion globale des certificats	196
6.6.10 Facette Serveur Service d'autorisation	196
6.6.11 Facette Serveur Service KeyCredential	196
6.6.12 Facette Serveur Attribut WriteMask	196
6.6.13 Facette Serveur Accès fichier	197
6.6.14 Facette Serveur Documentation	197
6.6.15 Facette Serveur Abonnement intégré aux DataChanges	197
6.6.16 Facette Serveur Abonnement normalisé aux DataChanges	198
6.6.17 Facette Serveur 2017 Abonnement normalisé aux DataChanges	198
6.6.18 Facette Serveur Abonnement amélioré aux DataChanges	198
6.6.19 Facette Serveur 2017 Abonnement amélioré aux DataChanges	199
6.6.20 Facette Serveur Abonnement durable	199
6.6.21 Facette Serveur Accès aux données	199
6.6.22 Facette Serveur ComplexType	200

6.6.23	Facette Serveur 2017 ComplexType	200
6.6.24	Facette Serveur Abonnement normalisé aux événements	200
6.6.25	Facette Serveur Notification de l'espace d'adressage	201
6.6.26	Facette Serveur A & C Condition de base	201
6.6.27	Facette Serveur A & C Refresh2	202
6.6.28	Facette Serveur A & C Instance de l'espace d'adressage	202
6.6.29	Facette Serveur A & C Activation	202
6.6.30	Facette Serveur A & C AlarmMetrics	203
6.6.31	Facette Serveur A & C Alarme	203
6.6.32	Facette Serveur A & C Alarme acquittable	203
6.6.33	Facette Serveur A & C Alarme exclusive	204
6.6.34	Facette Serveur A & C Alarme non exclusive	204
6.6.35	Facette Serveur A & C Instances précédentes	205
6.6.36	Facette Serveur A & C Dialogue	205
6.6.37	Facette Serveur A & C CertificateExpiration	205
6.6.38	Facette Conteneur A & E	206
6.6.39	Facette Serveur Méthode	206
6.6.40	Facette Serveur Audit	207
6.6.41	Facette Serveur Gestion des nœuds	207
6.6.42	Facette Serveur Rôle de base utilisateur	207
6.6.43	Facette Serveur Gestion des rôles d'utilisateurs	208
6.6.44	Facette Serveur Diagramme d'état	208
6.6.45	Facette Serveur Redondance client	208
6.6.46	Facette Serveur Redondance transparente	209
6.6.47	Facette Serveur Redondance visible	209
6.6.48	Facette Serveur Données brutes historiques	209
6.6.49	Facette Serveur Agrégat historique	209
6.6.50	Facette Serveur Données historiques à temps	210
6.6.51	Facette Serveur Accès à l'historique Données modifiées	211
6.6.52	Facette Serveur Annotation Historique	211
6.6.53	Facette Serveur Insertion Données historiques	211
6.6.54	Facette Serveur Mise à Jour Données historiques	211
6.6.55	Facette Serveur Remplacement données historiques	212
6.6.56	Facette Serveur Suppression données historiques	212
6.6.57	Facette Serveur Accès à l'historique Données structurées	212
6.6.58	Facette Serveur Événement Historique de base	213
6.6.59	Facette Serveur Mise à jour Événement historique	213
6.6.60	Facette Serveur Remplacement Événement historique	213
6.6.61	Facette Serveur Insertion Événement historique	213
6.6.62	Facette Serveur Suppression Événement historique	213
6.6.63	Facette Serveur Abonnement aux agrégats	214
6.6.64	Profil Serveur à dispositif nano-intégré	215
6.6.65	Profil Serveur 2017 Dispositif Nano-Intégré	215
6.6.66	Profil Serveur Dispositif Micro-Intégré	215
6.6.67	Profil Serveur 2017 Dispositif Micro-Intégré	215
6.6.68	Profil Serveur UA intégré	216
6.6.69	Profil Serveur UA 2017 intégré	216
6.6.70	Profil Serveur UA normalisé	216
6.6.71	Profil Serveur UA 2017 normalisé	216

6.6.72	Facette Client principal	217
6.6.73	Facette Client 2017 principal	217
6.6.74	Facette Client Sans Session	218
6.6.75	Facette <i>Client</i> Connexion inversée	218
6.6.76	Facette Client Comportement de base	218
6.6.77	Facette Client Découverte	219
6.6.78	Facette Client Découverte Sous-Réseau.....	219
6.6.79	Facette Client Découverte Globale	219
6.6.80	Facette Client Gestion Globale des Certificats	219
6.6.81	Facette Client Service KeyCredential.....	220
6.6.82	Facette Client Demande de jeton d'accès	220
6.6.83	Facette Client Consultation de l'AddressSpace	220
6.6.84	Facette Client Demande de Modification d'Etat.....	221
6.6.85	Facette Client Accès Fichier	221
6.6.86	Facette Client 2015 Prise en Charge Entrée de Gamme	221
6.6.87	Facette Client Connexion Multi-Serveur.....	222
6.6.88	Documentation – Client	222
6.6.89	Facette Client Attribut Lecture	222
6.6.90	Facette Client Attribut Ecriture.....	223
6.6.91	Facette Client Abonné aux DataChanges.....	223
6.6.92	Facette Client Abonnement durable	224
6.6.93	Facette Client DataAccess.....	224
6.6.94	Facette Client Abonné aux Événements.....	225
6.6.95	Facette Client Traitement des Événements de base	225
6.6.96	Facette Client Hiérarchie de Notification et de Source	226
6.6.97	Facette Client A & C Condition de base	226
6.6.98	Facette Client A & C Refresh2.....	226
6.6.99	Facette Client A & C Instance de l'Espace d'adressage	227
6.6.100	Facette Client A & C Activer	227
6.6.101	Facette Client A & C AlarmMetrics.....	227
6.6.102	Facette Client A & C Alarme	227
6.6.103	Facette Client A & C Alarme exclusive.....	228
6.6.104	Facette Client A & C Alarme non exclusive	228
6.6.105	Facette Client A & C Instances précédentes	229
6.6.106	Facette Client A & C Dialogue	229
6.6.107	Facette Client A & C CertificateExpiration.....	229
6.6.108	Facette Proxy A & E	230
6.6.109	Facette Client Méthode.....	231
6.6.110	Facette Client Audit	231
6.6.111	Facette Client Gestion des Nœuds	231
6.6.112	Facette Client Programmation de type avancé	231
6.6.113	Facette Client Gestion des Rôles d'Utilisateurs.....	232
6.6.114	Facette Client Diagramme d'état.....	232
6.6.115	Facette Client Diagnostic.....	232
6.6.116	Facette Client Redondant	233
6.6.117	Facette Client Commutateur de redondance	233
6.6.118	Facette Client Accès à l'historique	233
6.6.119	Facette Client Données Historiques AtTime	233
6.6.120	Facette Client Agrégat Historique	233

6.6.121	Facette Client Annotation Historique.....	235
6.6.122	Facette Client Accès à l'historique Données Modifiées	235
6.6.123	Facette Client Insertion Données Historiques	235
6.6.124	Facette Client Mise à Jour Données Historiques.....	235
6.6.125	Facette Client Remplacement données historiques.....	235
6.6.126	Facette Client Supprimer données structurées historiques.....	236
6.6.127	Facette Client Accès à l'historique Horodatage Serveur.....	236
6.6.128	Facette Client Accès données structurées historiques	236
6.6.129	Facette Client Données structurées historiques AtTime	236
6.6.130	Facette Client Données structurées historiques modifiées	237
6.6.131	Facette Client Insérer données structurées historiques.....	237
6.6.132	Facette Client Mettre à Jour données structurées historiques	237
6.6.133	Facette Client Remplacer données structurées historiques	237
6.6.134	Facette Client Supprimer données structurées historiques.....	238
6.6.135	Facette Client Événements Historiques.....	238
6.6.136	Facette Client Insertion Événements Historiques	238
6.6.137	Facette Client Mise à Jour Événements Historiques.....	238
6.6.138	Facette Client Remplacement Événements Historiques	238
6.6.139	Facette Client Suppression Événements Historiques	239
6.6.140	Facette Client Abonnement aux Agrégats	239
6.6.141	profil client UA normalisé;.....	240
6.6.142	Profil 2017 Client UA normalisé	240
6.6.143	Profil UA binaire UA-TCP UA-SC.....	241
6.6.144	Profil UA binaire HTTPS	241
6.6.145	Protocole de transport XML UA HTTPS	242
6.6.146	Protocole de transport UA JSON HTTPS	242
6.6.147	Profil UA binaire WSS UA-SC	242
6.6.148	Protocole de transport UA JSON WSS.....	242
6.6.149	Sécurité utilisateur Contrôle accès complet	243
6.6.150	Sécurité utilisateur Contrôle accès de base	243
6.6.151	Synchronisation Temporelle Sécurité.....	243
6.6.152	Meilleures pratiques – Événements d'audit	244
6.6.153	Meilleures pratiques – Gestion d'alarmes.....	244
6.6.154	Meilleures pratiques – Nombres aléatoires	244
6.6.155	Meilleures pratiques – Temporisations.....	244
6.6.156	Meilleures pratiques – Accès administratif	244
6.6.157	Meilleures pratiques – Gestion stricte des messages.....	245
6.6.158	Meilleures pratiques – Client Événements d'audit	245
6.6.159	Profil TransportSecurity – TLS 1.2	245
6.6.160	Profil TransportSecurity – TLS 1.2 avec PFS	245
6.6.161	SecurityPolicy – None.....	246
6.6.162	SecurityPolicy – Basic128Rsa15.....	246
6.6.163	SecurityPolicy – Basic256.....	246
6.6.164	SecurityPolicy [A] – Aes128-Sha256-RsaOaep	246
6.6.165	SecurityPolicy [B] – Basic256Sha256)	247
6.6.166	SecurityPolicy – Aes256-Sha256-RsaPss	248
6.6.167	Facette Jeton Utilisateur – Anonyme	248
6.6.168	Facette Serveur Jeton utilisateur – Nom d'utilisateur Mot de passe.....	248
6.6.169	Facette Serveur Jeton utilisateur – Certificat X509	249

6.6.170	Facette Serveur Jeton utilisateur – Jeton émis.....	249
6.6.171	Facette Serveur Windows Jeton utilisateur – Jeton émis.....	249
6.6.172	Facette Serveur Jeton utilisateur – JWT	249
6.6.173	Facette Client Jeton Utilisateur – Nom d'Utilisateur Mot de Passe	250
6.6.174	Facette Client Jeton Utilisateur – Certificat X509	250
6.6.175	Facette Client Jeton Utilisateur – Jeton Emis	250
6.6.176	Facette Client Windows Jeton Utilisateur – Jeton Emis	250
6.6.177	Facette Client Jeton Utilisateur – JWT	250
6.6.178	Profil Serveur Découverte Globale.....	251
6.6.179	Profil 2017 Serveur Découverte Globale	251
6.6.180	Serveur Découverte Globale et Gestion des Certificats.....	251
6.6.181	Serveur 2017 Découverte Globale et Gestion des Certificats	251
6.6.182	Profil Client Gestion Globale des Certificats	252
6.6.183	Profil 2017 Client Gestion Globale des Certificats.....	252
6.6.184	Facette Serveur Demande d'autorisation Service global	252
6.6.185	Facette Pull KeyCredential Service Global.....	252
6.6.186	Facette Push KeyCredential Service Global.....	253
	Bibliographie.....	254
	Figure 1 – Profil – ConformanceUnits – TestCases	142
	Figure 2 – Echantillon Client IHM.....	191
	Figure 3 – Échantillon Serveur intégré	192
	Figure 4 – Echantillon Serveur UA normalisé	193
	Tableau 1 – Catégories de profils	143
	Tableau 2 – Groupes de Conformité	144
	Tableau 3 – Services Découverte	145
	Tableau 4 – Services Session	147
	Tableau 5 – Services Gestion des Nœuds	148
	Tableau 6 – Services Vue	149
	Tableau 7 – Services Attribut	150
	Tableau 8 – Services Méthode.....	151
	Tableau 9 – Services Eléments surveillés	152
	Tableau 10 – Services Abonnement.....	154
	Tableau 11 – Sécurité.....	156
	Tableau 12 – Protocole et codage.....	163
	Tableau 13 – Informations de base	164
	Tableau 14 – Modèle d'espace d'adressage.....	167
	Tableau 15 – Accès aux données	169
	Tableau 16 – Alarmes et conditions	170
	Tableau 17 – Accès à l'historique	173
	Tableau 18 – Agrégats.....	176
	Tableau 19 – Audit.....	181
	Tableau 20 – Redondance	181
	Tableau 21 – Serveur "Découverte" Global	182

Tableau 22 – Divers	183
Tableau 23 – Liste des Profils	185
Tableau 24 – Facette Serveur 2017 principal	194
Tableau 25 – Facette Serveur Sans session	194
Tableau 26 – Facette Serveur Connexion inversée	195
Tableau 27 – Facette Serveur Comportement de base du serveur	195
Tableau 28 – Facette Serveur Demande modification d'état	195
Tableau 29 – Facette Serveur Découverte sous-Réseau	196
Tableau 30 – Facette Serveur Gestion globale des certificats	196
Tableau 31 – Facette Serveur Service d'autorisation	196
Tableau 32 – Facette Serveur Service KeyCredential	196
Tableau 33 – Facette Serveur Attribut WriteMask	197
Tableau 34 – Facette Serveur Accès fichier	197
Tableau 35 – Facette Serveur Documentation	197
Tableau 36 – Facette Serveur Abonnement intégré aux DataChanges	198
Tableau 37 – Facette Serveur 2017 Abonnement normalisé aux DataChanges	198
Tableau 38 – Facette Serveur 2017 Abonnement amélioré aux DataChanges	199
Tableau 39 – Facette Serveur Abonnement durable	199
Tableau 40 – Facette Serveur Accès aux données	200
Tableau 41 – Facette Serveur 2017 ComplexType	200
Tableau 42 – Facette Serveur Abonnement normalisé aux événements	201
Tableau 43 – Facette Serveur Notification de l'espace d'adressage	201
Tableau 44 – Facette Serveur A & C Condition de base	202
Tableau 45 – Facette Serveur A & C Refresh2	202
Tableau 46 – Facette Serveur A & C Instance Espace d'adressage	202
Tableau 47 – Facette Serveur A & C Activer	202
Tableau 48 – Facette Serveur A & C AlarmMetrics	203
Tableau 49 – Facette Serveur A & C Alarme	203
Tableau 50 – Facette Serveur A & C Alarme acceptable	204
Tableau 51 – Facette Serveur A & C Alarme exclusive	204
Tableau 52 – Facette Serveur A & C Alarme non exclusive	204
Tableau 53 – Facette Serveur A & C Instances précédentes	205
Tableau 54 – Facette Serveur A & C Dialogue	205
Tableau 55 – Facette Serveur A & C CertificateExpiration	205
Tableau 56 – Facette Conteneur A & E	206
Tableau 57 – Facette Serveur Méthode	207
Tableau 58 – Facette Serveur Audit	207
Tableau 59 – Facette Serveur Gestion des Nœuds	207
Tableau 60 – Facette Serveur Rôle de base Utilisateur	208
Tableau 61 – Facette Serveur Gestion des Rôles d'Utilisateurs	208
Tableau 62 – Facette Serveur Diagramme d'état	208
Tableau 63 – Facette Serveur Redondance Client	208
Tableau 64 – Facette Serveur Redondance transparente	209

Tableau 65 – Facette Serveur Redondance visible.....	209
Tableau 66 – Facette Serveur Données Brutes Historiques.....	209
Tableau 67 – Facette Serveur Agrégat Historique	210
Tableau 68 – Facette Serveur Données Historiques AtTime.....	211
Tableau 69 – Facette Serveur Accès à l'Historique Données Modifiées.....	211
Tableau 70 – Facette Serveur Annotation Historique	211
Tableau 71 – Facette Serveur Insertion Données Historiques	211
Tableau 72 – Facette Serveur Mise à Jour Données Historiques.....	212
Tableau 73 – Facette Serveur Remplacement données historiques.....	212
Tableau 74 – Facette Serveur Suppression données historiques	212
Tableau 75 – Facette Serveur Accès à l'historique données structurées	212
Tableau 76 – Facette Serveur Événements Historiques de base	213
Tableau 77 – Facette Serveur Mise à Jour Événements Historiques	213
Tableau 78 – Facette Serveur Remplacement Événements Historiques	213
Tableau 79 – Facette Serveur Insertion Événements Historiques	213
Tableau 80 – Facette Serveur Suppression Événements Historiques	214
Tableau 81 – Facette Serveur Abonnement aux Agrégats	214
Tableau 82 – Profil Serveur 2017 à dispositif nano-intégré	215
Tableau 83 – Profil Serveur 2017 à dispositif micro-intégré.....	216
Tableau 84 – Profil Serveur UA 2017 intégré	216
Tableau 85 – Profil Serveur UA 2017 normalisé.....	217
Tableau 86 – Facette Client 2017 principal	217
Tableau 87 – Facette Client Sans Session.....	218
Tableau 88 – Facette Client Connexion inversée	218
Tableau 89 – Facette Comportement Client de base	218
Tableau 90 – Facette Client Découverte	219
Tableau 91 – Facette Client Découverte Sous-Réseau	219
Tableau 92 – Facette Client Découverte Globale	219
Tableau 93 – Facette Client Gestion Globale des Certificats	220
Tableau 94 – Facette Client Service KeyCredential.....	220
Tableau 95 – Facette Client Demande de jeton d'accès	220
Tableau 96 – Facette Client Consultation de l'AddressSpace	221
Tableau 97 – Facette Client Demande de Modification d'Etat.....	221
Tableau 98 – Facette Client Accès Fichier	221
Tableau 99 – Facette Client 2015 Prise en Charge Entrée de Gamme	222
Tableau 100 – Facette Client Connexion Multi-Serveur.....	222
Tableau 101 – Documentation – Client.....	222
Tableau 102 – Facette Client Attribut Lecture	223
Tableau 103 – Facette Client Attribut Ecriture.....	223
Tableau 104 – Facette Client Abonné aux DataChanges.....	224
Tableau 105 – Facette Client Abonnement durable	224
Tableau 106 – Facette Client DataAccess.....	225
Tableau 107 – Facette Client Abonné aux Événements.....	225

Tableau 108 – Facette Client Traitement des Événements de base	226
Tableau 109 – Facette Client Hiérarchie de Notification et de Source	226
Tableau 110 – Facette Client A & C Condition de base	226
Tableau 111 – Facette Client A & C Refresh2	227
Tableau 112 – Facette Client A & C Instance Espace d'adressage	227
Tableau 113 – Facette Client A & C Activer	227
Tableau 114 – Facette Client A & C AlarmMetrics	227
Tableau 115 – Facette Client A & C Alarme	228
Tableau 116 – Facette Client A & C Alarme exclusive	228
Tableau 117 – Facette Client A & C Alarme non exclusive	229
Tableau 118 – Facette Client A & C Instances précédentes	229
Tableau 119 – Facette Client A & C Dialogue	229
Tableau 120 – Facette Client A & C CertificateExpiration	229
Tableau 121 – Facette Proxy A & E	230
Tableau 122 – Facette Client Méthode	231
Tableau 123 – Facette Client Audit	231
Tableau 124 – Facette Client Gestion des Nœuds	231
Tableau 125 – Facette Client Programmation de type avancé	232
Tableau 126 – Facette Client Gestion des Rôles d'Utilisateurs	232
Tableau 127 – Facette Client Diagramme d'état	232
Tableau 128 – Facette Client Diagnostic	232
Tableau 129 – Facette Client Redondant	233
Tableau 130 – Facette Client Commutateur de redondance	233
Tableau 131 – Facette Client Accès à l'historique	233
Tableau 132 – Facette Client Données Historiques AtTime	233
Tableau 133 – Facette Client Agrégat Historique	234
Tableau 134 – Facette Client Annotation Historique	235
Tableau 135 – Facette Client Accès à l'Histoire Données Modifiées	235
Tableau 136 – Facette Client Insertion Données Historiques	235
Tableau 137 – Facette Client Mise à Jour Données Historiques	235
Tableau 138 – Facette Client Remplacement données historiques	236
Tableau 139 – Facette Client Suppression données historiques	236
Tableau 140 – Facette Client Accès à l'historique Horodatage Serveur	236
Tableau 141 – Facette Client Accès données structurées historiques	236
Tableau 142 – Facette Client données structurées historiques AtTime	236
Tableau 143 – Facette Client données structurées historiques modifiées	237
Tableau 144 – Facette Client Insérer données structurées historiques	237
Tableau 145 – Facette Client Mettre à jour données structurées historiques	237
Tableau 146 – Facette Client Remplacer données structurées historiques	237
Tableau 147 – Facette Client Supprimer données structurées historiques	238
Tableau 148 – Facette Client Événements Historiques	238

Tableau 149 – Facette Client Insertion Evénements Historiques	238
Tableau 150 – Facette Client Mise à Jour Evénements Historiques	238
Tableau 151 – Facette Client Remplacement Evénements Historiques	239
Tableau 152 – Facette Client Suppression Evénements Historiques	239
Tableau 153 – Facette Client Abonné aux Agrégats.....	239
Tableau 154 – Profil 2017 Client UA normalisé	241
Tableau 155 – Profil UA binaire UA-TCP UA-SC	241
Tableau 156 – Profil USA binaire HTTPS	241
Tableau 157 – Protocole de transport XML UA HTTPS	242
Tableau 158 – Protocole de transport UA JSON HTTPS	242
Tableau 159 – Profil UA binaire WSS UA-SC	242
Tableau 160 – Protocole de transport UA JSON WSS	243
Tableau 161 – Sécurité utilisateur Contrôle accès complet	243
Tableau 162 – Sécurité utilisateur Contrôle accès de base	243
Tableau 163 – Synchronisation Temporelle Sécurité	243
Tableau 164 – Meilleures pratiques – Evénements d'audit	244
Tableau 165 – Meilleures pratiques – Gestion d'alarmes	244
Tableau 166 – Meilleures pratiques – Nombres aléatoires	244
Tableau 167 – Meilleures pratiques – Temporisations	244
Tableau 168 – Meilleures pratiques – Accès administratif	245
Tableau 169 – Meilleures pratiques – Gestion stricte des messages	245
Tableau 170 – Meilleures pratiques – Client Evénements d'audit	245
Tableau 171 – Profil TransportSecurity – TLS 1.2	245
Tableau 172 – Profil TransportSecurity – TLS 1.2 avec PFS	246
Tableau 173 – SecurityPolicy – None	246
Tableau 174 – SecurityPolicy [A] – Aes128-Sha256-RsaOaep	247
Tableau 175 – SecurityPolicy [B] – Basic256Sha256)	247
Tableau 176 – SecurityPolicy – Aes256-Sha256-RsaPss	248
Tableau 177 – Facette Jeton Utilisateur – Anonyme.....	248
Tableau 178 – Facette Serveur Jeton Utilisateur – Nom d'utilisateur Mot de passe	248
Tableau 179 – Facette Serveur Jeton Utilisateur – Certificat X509	249
Tableau 180 – Facette Serveur Jeton Utilisateur – Jeton Emis	249
Tableau 181 – Facette Serveur Windows Jeton Utilisateur – Jeton Emis	249
Tableau 182 – Facette Serveur Jeton Utilisateur – JWT	249
Tableau 183 – Facette Client Jeton Utilisateur – Nom d'Utilisateur Mot de Passe	250
Tableau 184 – Facette Client Jeton Utilisateur – Certificat X509	250
Tableau 185 – Facette Client Jeton Utilisateur – Jeton Emis	250
Tableau 186 – Facette ClientWindows Jeton Utilisateur – Jeton Emis	250
Tableau 187 – Facette Client Jeton Utilisateur – JWT	251
Tableau 188 – Profil 2017 Serveur Découverte Globale	251
Tableau 189 – Serveur 2017 Découverte Globale et Gestion des Certificats	252
Tableau 190 – Profil 2017 Client Gestion Globale des Certificats	252
Tableau 191 – Facette Serveur Demande d'autorisation Service global	252

Tableau 192 – Facette Pull KeyCredential Service Global.....	253
Tableau 193 – Facette Push KeyCredential Service Global.....	253

IECNORM.COM : Click to view the full PDF of IEC 62541-7:2020 RLV

COMMISSION ÉLECTROTECHNIQUE INTERNATIONALE

ARCHITECTURE UNIFIÉE OPC –

Partie 7: Profils

AVANT-PROPOS

- 1) La Commission Electrotechnique Internationale (IEC) est une organisation mondiale de normalisation composée de l'ensemble des comités électrotechniques nationaux (Comités nationaux de l'IEC). L'IEC a pour objet de favoriser la coopération internationale pour toutes les questions de normalisation dans les domaines de l'électricité et de l'électronique. A cet effet, l'IEC – entre autres activités – publie des Normes internationales, des Spécifications techniques, des Rapports techniques, des Spécifications accessibles au public (PAS) et des Guides (ci-après dénommés "Publication(s) de l'IEC"). Leur élaboration est confiée à des comités d'études, aux travaux desquels tout Comité national intéressé par le sujet traité peut participer. Les organisations internationales, gouvernementales et non gouvernementales, en liaison avec l'IEC, participent également aux travaux. L'IEC collabore étroitement avec l'Organisation Internationale de Normalisation (ISO), selon des conditions fixées par accord entre les deux organisations.
- 2) Les décisions ou accords officiels de l'IEC concernant les questions techniques représentent, dans la mesure du possible, un accord international sur les sujets étudiés, étant donné que les Comités nationaux de l'IEC intéressés sont représentés dans chaque comité d'études.
- 3) Les Publications de l'IEC se présentent sous la forme de recommandations internationales et sont agréées comme telles par les Comités nationaux de l'IEC. Tous les efforts raisonnables sont entrepris afin que l'IEC s'assure de l'exactitude du contenu technique de ses publications; l'IEC ne peut pas être tenue responsable de l'éventuelle mauvaise utilisation ou interprétation qui en est faite par un quelconque utilisateur final.
- 4) Dans le but d'encourager l'uniformité internationale, les Comités nationaux de l'IEC s'engagent, dans toute la mesure possible, à appliquer de façon transparente les Publications de l'IEC dans leurs publications nationales et régionales. Toutes divergences entre toutes Publications de l'IEC et toutes publications nationales ou régionales correspondantes doivent être indiquées en termes clairs dans ces dernières.
- 5) L'IEC elle-même ne fournit aucune attestation de conformité. Des organismes de certification indépendants fournissent des services d'évaluation de conformité et, dans certains secteurs, accèdent aux marques de conformité de l'IEC. L'IEC n'est responsable d'aucun des services effectués par les organismes de certification indépendants.
- 6) Tous les utilisateurs doivent s'assurer qu'ils sont en possession de la dernière édition de cette publication.
- 7) Aucune responsabilité ne doit être imputée à l'IEC, à ses administrateurs, employés, auxiliaires ou mandataires, y compris ses experts particuliers et les membres de ses comités d'études et des Comités nationaux de l'IEC, pour tout préjudice causé en cas de dommages corporels et matériels, ou de tout autre dommage de quelque nature que ce soit, directe ou indirecte, ou pour supporter les coûts (y compris les frais de justice) et les dépenses découlant de la publication ou de l'utilisation de cette Publication de l'IEC ou de toute autre Publication de l'IEC, ou au crédit qui lui est accordé.
- 8) L'attention est attirée sur les références normatives citées dans cette publication. L'utilisation de publications référencées est obligatoire pour une application correcte de la présente publication.
- 9) L'attention est attirée sur le fait que certains des éléments de la présente Publication de l'IEC peuvent faire l'objet de droits de brevet. L'IEC ne saurait être tenue pour responsable de ne pas avoir identifié de tels droits de brevets et de ne pas avoir signalé leur existence.

La Norme internationale IEC 62541-7 a été établie par le sous-comité 65E: Les dispositifs et leur intégration dans les systèmes de l'entreprise, du comité d'études 65 de l'IEC: Mesure, commande et automation dans les processus industriels.

Cette troisième édition annule et remplace la deuxième édition parue en 2015. Cette édition constitue une révision technique.

Cette édition inclut les modifications techniques majeures suivantes par rapport à l'édition précédente:

a) ajout de nouveaux profils fonctionnels:

- profils pour la découverte globale et la gestion globale des certificats;
- profils pour la gestion globale de KeyCredential et pour la gestion globale des jetons d'accès;

- facette pour les abonnements durables;
 - profil client UA normalisé;
 - profils pour l'administration des rôles et permissions des utilisateurs.
- b) ajout de nouveaux profils de transport:
- HTTPS avec codage JSON;
 - Secure WebSockets (WSS) avec codage binaire ou JSON;
 - connectivité inversée.
- c) ajout de nouveaux profils de sécurité:
- TransportSecurity – TLS 1.2 avec PFS (confidentialité persistante);
 - SecurityPolicy [A] – Aes128-Sha256-RsaOaep (en remplacement de Base128Rsa15);
 - SecurityPolicy – Aes256-Sha256-RsaPss ajoute la confidentialité persistante pour UA TCP);
 - jeton d'utilisateur JWT (Jason Web Token).
- d) spécification des profils de sécurité déconseillés (en raison des algorithmes cassés):
- SecurityPolicy – Basic128Rsa15 (algorithme Sha1 cassé);
 - SecurityPolicy – Basic256 (algorithme Sha1 cassé);
 - TransportSecurity – TLS 1.0 (algorithme RC4 cassé);
 - TransportSecurity – TLS 1.1 (algorithme RC4 cassé).
- e) transport déconseillé (absence de prise en charge sur la plupart des plateformes):
- SOAP/HTTP avec WS-SecureConversation (tous les codages).

Le texte de cette Norme internationale est issu des documents suivants:

FDIS	Rapport de vote
65E/707/FDIS	65E/725/RVD

Le rapport de vote indiqué dans le tableau ci-dessus donne toute information sur le vote ayant abouti à l'approbation de cette Norme internationale.

Ce document a été rédigé selon les Directives ISO/IEC, Partie 2.

Dans l'ensemble du présent document et dans les autres parties de la série IEC 62541, certaines conventions de document sont utilisées:

Le format *italique* est utilisé pour mettre en évidence un terme défini ou une définition qui apparaît à l'article "Termes et définitions" dans l'une des parties de la série IEC 62541.

Le format *italique* est également utilisé pour mettre en évidence le nom d'un paramètre d'entrée ou de sortie de service, ou le nom d'une structure ou d'un élément de structure habituellement défini dans les tableaux.

Par ailleurs, les *termes* et les *noms en italique* sont, à quelques exceptions près, écrits en camel-case (pratique qui consiste à joindre, sans espace, les éléments des mots ou expressions composés, la première lettre de chaque élément étant en majuscule). Par exemple, le terme défini est *AddressSpace* et non Espace d'adressage. Cela permet de mieux comprendre qu'il existe une définition unique pour *AddressSpace*, et non deux définitions distinctes pour Espace et pour Adressage.

Une liste de toutes les parties de la série IEC 62541, publiées sous le titre général *Architecture unifiée OPC*, peut être consultée sur le site web de l'IEC.

Le comité a décidé que le contenu de ce document ne sera pas modifié avant la date de stabilité indiquée sur le site web de l'IEC sous "<http://webstore.iec.ch>" dans les données relatives au document recherché. A cette date, le document sera

- reconduit,
- supprimé,
- remplacé par une édition révisée, ou
- amendé.

IMPORTANT – Le logo "*colour inside*" qui se trouve sur la page de couverture de cette publication indique qu'elle contient des couleurs qui sont considérées comme utiles à une bonne compréhension de son contenu. Les utilisateurs devraient, par conséquent, imprimer cette publication en utilisant une imprimante couleur.

IECNORM.COM : Click to view the full PDF of IEC 62541-7:2020

ARCHITECTURE UNIFIÉE OPC –

Partie 7: Profils

1 Domaine d'application

La présente partie de l'IEC 62541 définit les *Profils* de l'architecture unifiée OPC (OPC UA). Les *Profils* du présent document permettent de classer les caractéristiques en fonction des essais de produits OPC UA et de la nature des essais (via un outil ou en laboratoire). Cela inclut les essais effectués à l'aide de l'outil d'essai de conformité CTT OPC UA développé par la Fondation OPC (outil d'essai autonome), ainsi que les essais réalisés par des laboratoires de certification indépendants de cette même fondation. Le présent document peut également faire référence aux outils d'essai ou au laboratoire d'essai d'un autre organisme. Dans le cas présent, l'élément important est le concept qui oppose les essais fondés sur un outil automatisé et les essais en laboratoire. Le domaine d'application de la présente norme inclut la définition d'une fonctionnalité qui ne peut être soumise à l'essai qu'en laboratoire, ainsi que la définition du regroupement des fonctionnalités à utiliser durant les essais de produits OPC UA effectués en laboratoire ou à l'aide d'outils automatisés. La définition des *TestCases* réels ne relève pas du domaine d'application du présent document, mais les catégories de *TestCases* générales relèvent du domaine d'application du présent document.

La plupart des applications OPC UA sont conformes à plusieurs *Profils*, mais pas à l'ensemble des *Profils*.

2 Références normatives

Les documents ci-après sont des références normatives indispensables à l'application du présent document. Pour les références datées, seule l'édition citée s'applique. Pour les références non datées, la dernière édition du document de référence s'applique (y compris les éventuels amendements).

IEC TR 62541-1, *OPC Unified Architecture – Part 1: Overview and Concepts* (disponible en anglais seulement)

IEC TR 62541-2, *OPC Unified Architecture – Part 2: Security Model* (disponible en anglais seulement)

IEC 62541-3, *Architecture unifiée OPC – Partie 3: Modèle d'espace d'adressage*

IEC 62541-4, *Architecture unifiée OPC – Partie 4: Services*

IEC 62541-5, *Architecture unifiée OPC – Partie 5: Modèle d'information*

IEC 62541-6, *Architecture unifiée OPC – Partie 6: Mappings*

IEC 62541-8, *Architecture unifiée OPC – Partie 8: Accès aux données*

IEC 62541-9, *Architecture unifiée OPC – Partie 9: Alarmes et conditions*

IEC 62541-11, *Architecture unifiée OPC – Partie 11: Accès à l'historique*

IEC 62541-12, *Architecture unifiée OPC – Partie 12: Services globaux et de découverte*

IEC 62541-13, *Architecture unifiée OPC – Partie 13: Agrégats*

Compliance Part 8 UA Server: OPC Test Lab Specification: Part 8 – UA Server
<http://www.opcfoundation.org/Test/Part8/>

Compliance Part 9 UA Client: OPC Test Lab Specification: Part 9 – UA Client
<http://www.opcfoundation.org/Test/Part9/>

3 Termes, définitions et termes abrégés

3.1 Termes et définitions

Pour les besoins du présent document, les termes et définitions donnés dans l'IEC TR 62541-1, l'IEC TR 62541-2, l'IEC 62541-3, l'IEC 62541-4, l'IEC 62541-6 et l'IEC 62541-8 ainsi que les suivants s'appliquent.

L'ISO et l'IEC tiennent à jour des bases de données terminologiques destinées à être utilisées en normalisation, consultables aux adresses suivantes:

- IEC Electropedia: disponible à l'adresse <http://www.electropedia.org/>
- ISO Online browsing platform: disponible à l'adresse <http://www.iso.org/obp>

NOTE La Figure 1 fournit une vue d'ensemble des termes définis dans le présent document et de leurs interactions.

3.1.1

application

programme logiciel qui exécute ou met en œuvre certains aspects d'OPC UA

Note 1 à l'article: L'application peut fonctionner sur tout type de machine et exécuter toute fonction, de même qu'elle peut être logicielle ou matérielle. La seule exigence est qu'elle mette en œuvre OPC UA.

3.1.2

ConformanceUnit

ensemble spécifique de caractéristiques OPC UA qui peuvent être soumises à l'essai comme une entité individuelle

Note 1 à l'article: Une ConformanceUnit peut représenter un groupe de services, des portions de services ou des modèles d'information.

3.1.3

ConformanceGroup

groupe de ConformanceUnits auquel un nom est attribué

Note 1 à l'article: Ce regroupement est destiné uniquement à faciliter l'organisation des ConformanceUnits. Les ConformanceGroups types incluent les groupes propres à chaque jeu de services OPC UA et à chacune des normes du Modèle d'information.

3.1.4

Facette

profil dédié à une caractéristique spécifique qu'un Serveur ou Client peut exiger

Note 1 à l'article: Les Facettes sont généralement combinées pour former des Profils d'un niveau plus élevé. L'utilisation du terme Facette dans l'intitulé d'un Profil indique que le Profil donné n'est pas un Profil autonome.

3.1.5

Profil FullFeatured

Profil qui définit toutes les caractéristiques nécessaires à la construction d'une Application OPC UA fonctionnelle

Note 1 à l'article: Un Profil complet en particulier définit les exigences de transport et de sécurité.

3.1.6

ProfileCategory

catégorie qui organise les Profils en classes d'applications, tels que Serveur ou Client

Note 1 à l'article: Ces catégories permettent de déterminer le type d'Application pour lequel un Profil donné serait utilisé. Pour plus d'informations, voir 4.4.

3.1.7

TestCase

description technique d'un ensemble d'étapes nécessaires pour soumettre à l'essai une fonction ou un modèle d'information particulier

Note 1 à l'article: Les TestCases fournissent les informations suffisantes pour permettre à un développeur de les mettre en œuvre dans un code. Les TestCases fournissent également un récapitulatif complet des résultats d'exécution du code mis en œuvre attendus, ainsi que les éventuelles conditions préalables qu'il importe d'établir avant de pouvoir exécuter le TestCase.

3.1.8

TestLab

installation conçue pour fournir des services d'essai

Note 1 à l'article: Ces services incluent, entre autres, le personnel qui effectue directement les essais, les essais automatisés et un processus reproductible formel. La Fondation OPC a élaboré une norme complète décrivant les TestLabs OPC UA ainsi que les essais à organiser (voir Compliance Part 8 UA Server, Compliance Part 9 UA Client).

3.2 Termes abrégés

DA data access (accès aux données)

HA historical access (accès à l'historique)

IHM interface homme-machine

NIST National Institute of Standard and Technology (Institut national des normes et de la technologie)

PKI Public Key Infrastructure (Infrastructure à clés publiques)

RSA Rivest-Shamir-Adleman

UA Unified Architecture (Architecture unifiée)

4 Vue d'ensemble

4.1 Généralités

La série IEC 62541, qui est la norme d'architecture unifiée OPC, décrit un certain nombre de *Services* et différents modèles d'information. Ces *Services* et modèles d'information peuvent être désignés comme les caractéristiques d'un *Serveur* ou d'un *Client*. Il est nécessaire que les *Serveurs* et *Clients* puissent décrire les caractéristiques qu'ils prennent en charge et celles qu'ils souhaitent voir certifiées. Le présent document procède au regroupement de ces caractéristiques. Les caractéristiques individuelles sont regroupées en *ConformanceUnits* qui sont regroupées en *Profils*. La Figure 1 représente les interactions entre les *Profils*, les *ConformanceUnits* et les *TestCases*. Les flèches larges désignent les composants utilisés pour composer le parent. Par exemple, un *Profil* est construit à partir de *Profils* et de *ConformanceUnits*. La figure représente également une caractéristique de l'outil d'essai de conformité CTT (Compliance Test Tool) OPC UA, dans le sens où elle permet de vérifier par essai si un *Profil* invoqué satisfait à l'ensemble des *ConformanceUnits*. L'outil permet également de vérifier par essai toutes les autres *ConformanceUnits* et de consigner dans un rapport les *Profils* qui satisfont aux essais de conformité. Chaque *TestCase* est défini dans un document distinct (voir Compliance Part 8 UA Server et Compliance Part 9 UA Client). Les *TestCases* sont à leur retour associés aux *ConformanceUnits* adéquates définies dans le présent document. Cette relation est également reflétée par l'outil d'essai de conformité OPC UA.

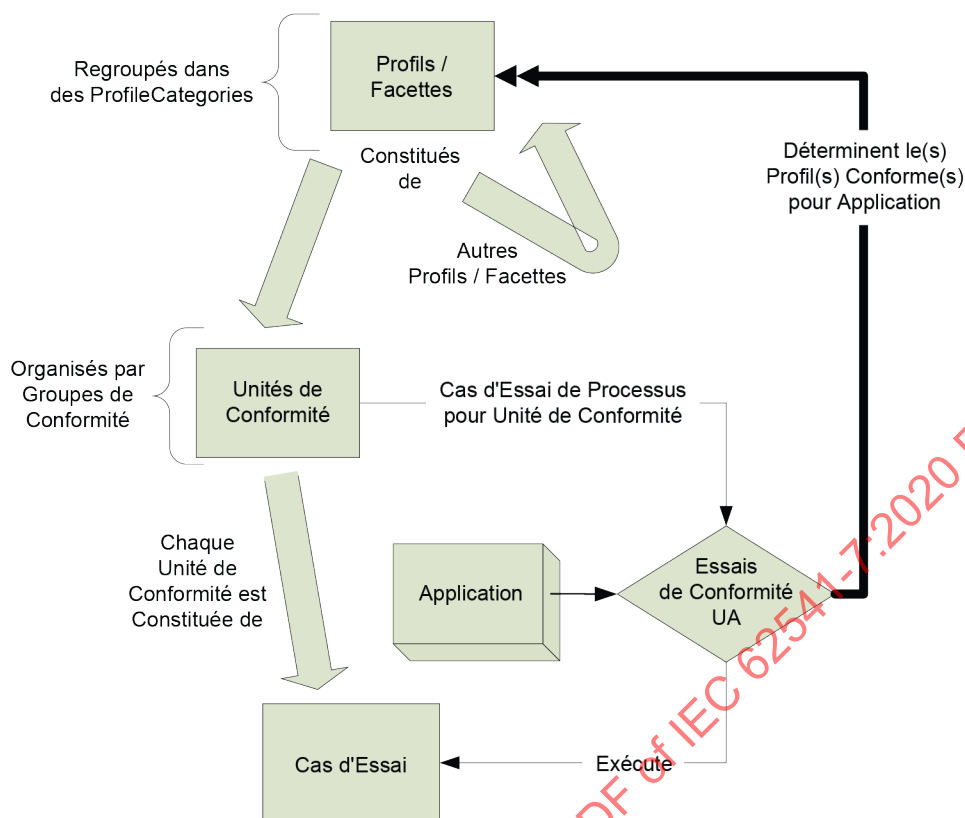


Figure 1 – Profil – ConformanceUnits – TestCases

4.2 ConformanceUnit

Chaque *ConformanceUnit* représente un ensemble spécifique de caractéristiques (un groupe de services, des portions de services ou des modèles d'information, par exemple) qui peuvent être soumises à l'essai en tant qu'entité individuelle. Les *ConformanceUnits* sont les blocs fonctionnels d'un *Profil*. Chaque *ConformanceUnit* peut également être utilisée comme catégorie d'essai. Pour chaque *ConformanceUnit*, il existe un certain nombre de TestCases qui permettent de soumettre à l'essai la fonctionnalité décrite par la *ConformanceUnit*. La description d'une *ConformanceUnit* est destinée à fournir des informations suffisantes pour représenter la fonctionnalité exigée. Dans de nombreux cas, il peut néanmoins être nécessaire que le lecteur examine également la partie applicable de l'OPC UA afin d'acquérir une compréhension complète de la *ConformanceUnit*. Des informations supplémentaires concernant les essais de la *ConformanceUnit* sont fournies dans les normes d'essai Compliance Part 8 UA Server ou Compliance Part 9 UA Client.

Les mêmes caractéristiques n'apparaissent pas dans plusieurs *ConformanceUnits*.

4.3 Profils

Un *Profil* est une agrégation nommée de *ConformanceUnits* et d'autres *Profils*. Pour prendre en charge un *Profil*, une application doit prendre en charge les *ConformanceUnits* et tous les *Profils* agrégés. La définition des *Profils* est une activité continue, dans le sens où il est prévu d'ajouter de nouveaux *Profils* ultérieurement.

Une application OPC UA prend généralement en charge plusieurs *Profils*.

Ces *Profils* peuvent comporter la même *ConformanceUnit*.

Lorsqu'un *Profil* est soumis à l'essai, ce sont les *ConformanceUnits* individuelles qui comportent le *Profil* qui sont soumises à l'essai.

Les *Profils* sont nommés selon les conventions de nommage (voir 6.3).

4.4 Catégories de profils

Les *Profils* sont regroupés en catégories afin d'aider les fournisseurs et utilisateurs finaux à comprendre le domaine d'application d'un *Profil*. Un *Profil* peut être attribué à une ou plusieurs catégories.

Le Tableau 1 contient la liste des *ProfileCategories* actuellement définies.

Tableau 1 – Catégories de profils

Catégorie	Description
Client	Les <i>Profils</i> de cette catégorie spécifient les fonctions d'un <i>Client</i> OPC UA.
Service d'annuaire global	Les <i>Profils</i> de cette catégorie spécifient les fonctions pour la découverte globale et la gestion des certificats.
Sécurité	Les <i>Profils</i> de cette catégorie spécifient les fonctions relatives à la Sécurité. Les politiques de sécurité font partie de cette catégorie. L'URI de ces politiques de sécurité doit faire partie intégrante d'une Description de point d'extrémité transmise par le service GetEndpoint. Les <i>Profils</i> de cette catégorie s'appliquent aux <i>Clients</i> et aux <i>Serveurs</i> .
Serveur	Les <i>Profils</i> de cette catégorie spécifient des fonctions pour un <i>Serveur</i> OPC UA. L'URI de ces <i>Profils</i> peut être présentée dans les fonctions du <i>Serveur</i> .
Transport	Les <i>Profils</i> de cette catégorie spécifient des mappings de protocoles spécifiques. L'URI de ce type de <i>Profils</i> doit faire partie intégrante d'une Description de point d'extrémité. Ces <i>Profils</i> s'appliquent aux <i>Clients</i> et aux <i>Serveurs</i> .

5 Unités de Conformité

5.1 Vue d'ensemble

Une *ConformanceUnit* représente une entité qui peut être soumise individuellement à l'essai. Pour faciliter la compréhension, la longue liste des *ConformanceUnits* a été regroupée en *ConformanceGroups*. Ces groupes reflètent les *Jeux de services* définis dans l'IEC 62541-4 et les modèles d'information OPC UA. Le Tableau 2 fournit la liste des *ConformanceGroups*. Ces groupes et les *ConformanceUnits* qu'ils décrivent sont décrits à l'Article 5, à partir du 5.2. Les *ConformanceGroups* n'ont aucune influence sur les essais. Ils sont uniquement utilisés pour des raisons d'organisation, autrement dit pour simplifier la lisibilité du présent document.

Tableau 2 – Groupes de Conformité

Groupe	Description
Modèle d'espace d'adressage	Définit les <i>ConformanceUnits</i> pour différentes caractéristiques de l' <i>AddressSpace</i> OPC UA.
Agrégats	Ensemble des <i>ConformanceUnits</i> relatives aux Agrégats, y compris les <i>ConformanceUnits</i> individuelles pour chaque Agrégat pris en charge, selon la description de l'IEC 62541-13.
Alarmes et conditions	Ensemble des <i>ConformanceUnits</i> associées au modèle d'information OPC UA pour les <i>Conditions</i> , <i>Conditions</i> acceptables, confirmations et <i>Alarmes</i> spécifiées dans l'IEC 62541-9.
Services Attribut	Inclut les <i>ConformanceUnits</i> permettant de lire ou d'écrire les valeurs d' <i>Attribut</i> actuelles ou historiques.
Audits	La sécurité au niveau utilisateur inclut la prise en charge des journaux d'audit de sécurité, avec une traçabilité entre les listes de contrôle <i>Client</i> et <i>Serveur</i> .
Informations de base	Tous les éléments d'information définis dans l'IEC 62541-5.
Accès aux données	<i>ConformanceUnits</i> spécifiques aux <i>Clients</i> et aux <i>Serveurs</i> qui traitent de la représentation et de l'utilisation des données d'automatisation, selon les spécifications de l'IEC 62541-8.
Services Découverte	<i>ConformanceUnits</i> axées sur la <i>Découverte</i> des Points d'extrémité du <i>Serveur</i> .
GDS	<i>ConformanceUnits</i> d'un GDS. Comprend des unités pour la découverte globale et la gestion globale des certificats.
Accès à l'historique	Accès aux données archivées des valeurs d' <i>Attribut de Nœud</i> ou Événements.
Services Méthode	Les Méthodes représentent les appels de fonction des <i>Objets</i> . Les Méthodes sont invoquées et renvoyées uniquement après exécution (réussite ou échec).
Divers	Ce groupe contient des <i>ConformanceUnits</i> qui couvrent des sujets divers, tels que les comportements recommandés, la documentation, etc. Généralement, ces <i>ConformanceUnits</i> ne correspondent à aucun des autres groupes.
Services Eléments surveillés	Les <i>Clients</i> définissent des <i>MonitoredItems</i> pour s'abonner à des données et à des Événements. Chaque <i>MonitoredItem</i> identifie l'élément à surveiller et l' <i>Abonnement</i> à utiliser pour envoyer des <i>Notifications</i> .
Services Gestion des Nœuds	Groupe les <i>ConformanceUnits</i> pour tous les <i>Services</i> afin d'ajouter et de supprimer des <i>Nœuds</i> et des <i>Références</i> de l' <i>AddressSpace</i> OPC UA.
Protocole et Codage	Couvre toutes les combinaisons de transport et de codage spécifiées dans l'IEC 62541-6.
Redondance	La conception d'OPC UA permet aux fournisseurs de créer des <i>Clients</i> et des <i>Serveurs</i> redondants de manière cohérente. La Redondance peut être utilisée pour une disponibilité élevée, la tolérance aux anomalies et l'équilibrage des charges.
Sécurité	<i>ConformanceUnits</i> relatives à la sécurité qui peuvent être intégrées dans des profils (cela couvre tous les aspects relatifs à la sécurité).
Services Session	Une <i>Session</i> (OPC UA) est une connexion de couche d'application.
Services Abonnement	Les abonnements permettent de reporter les <i>Notifications</i> au <i>Client</i> .
Services Vue	Les <i>Clients</i> utilisent le <i>Jeu de services</i> Vue pour naviguer dans l' <i>AddressSpace</i> OPC UA ou dans une vue (un sous-ensemble) de l' <i>AddressSpace</i> OPC UA.

5.2 Services

Les Tableaux 3 à 10 décrivent les *ConformanceUnits* pour les *Services* spécifiés dans l'IEC 62541-4. Les tableaux sont associés aux *Jeux de services*.

Une *ConformanceUnit* simple peut référencer plusieurs *Services* (*CreateSession*, *ActivateSession* et *CloseSession*, par exemple), mais elle peut également faire référence aux aspects individuels des *Services* (l'utilisation d'*ActivateSession* pour masquer un nouvel utilisateur, par exemple).

Chaque tableau fournit une liste des *Catégories de profils* auxquels appartient une *ConformanceUnit*, ainsi que le titre et la description de la *ConformanceUnit*. Dans certains cas, une *ConformanceUnit* est dérivée d'une autre *ConformanceUnit*. Cette unité parente sera donc spécifiée dans la description de chacune des unités dérivées. Dans ce cas, les unités dérivées héritent de tous les essais de leurs parents, avec un ou plusieurs TestCases supplémentaires. Ces TestCases peuvent seulement restreindre les TestCases existants. C'est le cas par exemple d'un TestCase visant à soumettre à l'essai le nombre de connexions, où le TestCase du parent exige au moins une connexion et où la *ConformanceUnit* obtenue exige un *TestCase* pour au moins cinq connexions.

Le *Jeu de services "Découverte"* comprend plusieurs *ConformanceUnits* (voir Tableau 3). Tous les *Serveurs* fournissent certains aspects de cette fonctionnalité; se reporter aux *Profils* catégorisés comme *Profils de serveur* pour plus d'informations. Les *Clients* peuvent prendre en charge certains aspects de cette fonctionnalité; se reporter aux *Profils* catégorisés comme *Profils de client* pour plus d'informations.

Tableau 3 – Services Découverte

Catégorie	Intitulé	Description
<i>Serveur</i>	<i>Découverte</i> "Obtenir des Points d'extrémité"	Prend en charge le <i>Service</i> GetEndpoints afin d'obtenir tous les points d'extrémité du <i>Serveur</i> . Cela inclut le filtrage fondé sur les <i>Profils</i> .
<i>Serveur</i>	<i>Découverte</i> "Obtenir les points d'extrémité SessionLess"	Prend en charge au moins un point d'extrémité pour émettre les <i>Services</i> SessionLess. Prend en charge l'obtention de ces points d'extrémité en acceptant l'URI de Transport comme filtre pour le <i>Service</i> GetEndpoints avec la chaîne de requête "SL" annexée à l'URI de Transport. Par exemple, "http://opcfoundation.org/UA-Profile/Transport/https-uajson?SL"
<i>Serveur</i>	<i>Découverte</i> "Trouver des Serveurs individuellement"	Prend en charge le <i>Service</i> FindServers uniquement pour un usage individuel.
<i>Serveur</i>	<i>Découverte</i> "Register"	Appelle le <i>Service</i> RegisterServer pour s'enregistrer (<i>Serveur</i> OPC UA) auprès d'un <i>Service "Découverte"</i> externe via un canal sécurisé par un SecurityMode autre que NONE.
<i>Serveur</i>	<i>Découverte</i> "Register2"	Appelle le <i>Service</i> RegisterServer2 pour s'enregistrer auprès d'un <i>Service "Découverte"</i> externe via un Canal sécurisé par un SecurityMode autre que "None". Cela comprend le passage d'une liste de brefs identificateurs de capacité. Les identificateurs et leur utilisation sont spécifiés dans l'IEC 62541-12.
<i>Serveur</i>	Notification d'un <i>Serveur</i> " <i>Découverte</i> " au moyen du mDNS	Fournit une fonction mDNS pour notifier un <i>Serveur</i> avec ses capacités. Les identificateurs de capacité et l'utilisation des journaux mDNS aux fins de la <i>Découverte</i> OPC UA sont spécifiés dans l'IEC 62541-12. Noter que cette fonctionnalité est exigée uniquement pour les <i>Serveurs</i> qui ne s'enregistrent pas avec un LDS. Les identificateurs de capacité et leur utilisation dans les journaux mDNS sont spécifiés dans l'IEC 62541-12.
<i>Serveur</i>	<i>Découverte</i> "Configuration"	Permet la configuration de l'URL du <i>Serveur "Découverte"</i> , où le <i>Serveur</i> s'enregistre lui-même. Permet la désactivation complète de l'enregistrement avec un <i>Serveur "Découverte"</i> .
<i>Client</i>	<i>Découverte</i> <i>Client</i> "Trouver les Serveurs de base"	Utilise le <i>Service</i> FindServers pour obtenir tous les <i>Serveurs</i> installés sur une plateforme donnée.
<i>Client</i>	<i>Découverte</i> <i>Client</i> "Trouver les Serveurs ayant un URI"	Utilise le <i>Service</i> FindServers pour obtenir les URL des URI des <i>Serveurs</i> spécifiques.

Catégorie	Intitulé	Description
<i>Client</i>	<i>Découverte</i> <i>Client</i> "Trouver les <i>Serveurs</i> dynamiques"	Détecte les nouveaux <i>Serveurs</i> après un appel initial du <i>Service</i> FindServers.
<i>Client</i>	<i>Découverte</i> <i>Client</i> "Trouver les <i>Serveurs</i> sur le Réseau"	Prend en charge l'une des options permettant de localiser les <i>Serveurs</i> dans le réseau.
<i>Client</i>	<i>Découverte</i> <i>Client</i> "Trouver les <i>Serveurs</i> sur le Réseau au moyen d'un LDS-ME"	Utilise le <i>Service</i> FindServersOnNetwork pour obtenir les URL des URI des <i>Serveurs</i> spécifiques. Noter que ce <i>Service</i> est disponible au moyen du <i>Serveur</i> " <i>Découverte</i> " Local avec l'extension de multidiffusion (LDS-ME).
<i>Client</i>	<i>Découverte</i> <i>Client</i> "Trouver les <i>Serveurs</i> sur le Réseau au moyen du mDNS"	Utilise la <i>Découverte</i> de <i>Service</i> au moyen du mDNS pour localiser les <i>Serveurs</i> d'un même réseau de multidiffusion. Le contenu des journaux mDNS pour la <i>Découverte</i> OPC UA est décrit dans l'IEC 62541-12. Noter que cette fonctionnalité est exigée pour les <i>Clients</i> uniquement lorsqu'il n'y a pas de <i>Serveur</i> " <i>Découverte</i> " Local comportant l'extension de multidiffusion (LDS-ME). Les identificateurs de capacité et leur utilisation dans les journaux mDNS sont spécifiés dans l'IEC 62541-12.
<i>Client</i>	<i>Découverte</i> <i>Client</i> "Trouver les <i>Serveurs</i> dans le GDS"	Utilise la <i>Méthode</i> QueryServers sur l' <i>Objet</i> du répertoire du GDS pour localiser les <i>Serveurs</i> qui remplissent les critères du filtre spécifié dans la requête. Cette <i>Méthode</i> est spécifiée dans l'IEC 62541-12.
<i>Client</i>	<i>Découverte</i> <i>Client</i> "Trouver les Applications dans le GDS"	Utilise la <i>Méthode</i> QueryApplications sur l' <i>Objet</i> du Répertoire du GDS pour localiser les <i>Applications</i> qui remplissent les critères du filtre spécifié dans la requête. Cette <i>Méthode</i> est spécifiée dans l'IEC 62541-12.
<i>Client</i>	<i>Découverte</i> <i>Client</i> "Obtenir les Points d'extrémité de base"	Utilise le <i>Service</i> GetEndpoints pour obtenir tous les Points d'extrémité d'un URI de <i>Serveur</i> donné.
<i>Client</i>	<i>Découverte</i> <i>Client</i> "Obtenir les Points d'extrémité SessionLess"	Utilise le <i>Service</i> GetEndpoints avec un filtre pour obtenir les Points d'extrémité qui peuvent être utilisés pour l'invocation du <i>Service</i> SessionLess. Le filtre est l'URI de Transport étendue par la chaîne de requête "SL". Par exemple, http://opcfoundation.org/UA-Profile/Transport/https-uajson?SL .
<i>Client</i>	<i>Découverte</i> <i>Client</i> "Obtenir les Points d'extrémité dynamiques"	Détecte les modifications apportées sur les Points d'extrémité après un appel initial du <i>Service</i> GetEndpoints.
<i>Client</i>	<i>Découverte</i> <i>Client</i> "Configurer les Points d'extrémité"	Permet la spécification d'un Point d'extrémité sans passer par le <i>Jeu de services</i> " <i>Découverte</i> ".

Le *Jeu de services* "*Session*" comprend plusieurs *ConformanceUnits* (voir Tableau 4). Les services CreateSession, ActivateSession et CloseSession sont pris en charge comme une unité individuelle. Tous les *Serveurs* et *Clients* fournissent cette fonctionnalité.

Tableau 4 – Services Session

Catégorie	Intitulé	Description
Serveur	Session "Comportement Service général"	Met en œuvre le comportement de <i>Service</i> de base. Ceci inclut notamment: – la vérification du jeton d'authentification; – la transmission de la requestHandle dans les réponses; – la transmission des informations de diagnostic disponibles demandée, avec le paramètre 'returnDiagnostics'; – le respect du message timeoutHint.
Serveur	Session "Base"	Prend en charge le <i>Jeu de services "Session"</i> (CreateSession, ActivateSession, CloseSession) à l'exception de l'utilisation d'ActivateSession pour changer l'utilisateur de la Session. Ceci inclut le traitement correct de tous les paramètres fournis. Pour les services CreateSession et ActivateSession, si le SecurityMode = None, alors: 1) le <i>Certificat</i> d'application et le Nonce sont facultatifs; 2) les signatures sont nulles/vides. Ces détails sont décrits dans l'IEC 62541-4.
Serveur	Session "Modification d'utilisateur"	Prend en charge l'utilisation d'ActivateSession pour modifier l'utilisateur de la Session.
Serveur	Session "Annuler"	Prend en charge le <i>Service Annuler</i> pour annuler les requêtes en suspens.
Serveur	Session "Minimum 1"	Prend en charge au moins 1 session (total).
Serveur	Session "Minimum 2 en parallèle"	Prend en charge au moins 2 Sessions en parallèle (total pour tous les Clients).
Serveur	Session "Minimum 50 en parallèle"	Prend en charge au minimum 50 Sessions en parallèle (total pour tous les Clients).
Serveur	Session "Invocation Sans Session"	Définit la prise en charge du Service "SessionlessInvoke" défini dans l'IEC 62541-4 pour traiter l'un des Services (comme Lecture/Ecriture, Parcourir ou Appel) qui sont conçus pour l'Invocation Sans Session.
Client	Session Client "Comportement de Service général"	Met en œuvre le comportement de <i>Service</i> de base. Ceci inclut notamment: – l'intégration du jeton d'authentification correct de la Session; – la création d'une requestHandle si nécessaire; – la demande d'informations de diagnostic avec le paramètre 'returnDiagnostics'; – l'évaluation du serviceResult et des résultats d'exploitation.
Client	Session Client "Base"	Utilise le <i>Jeu de services "Session"</i> (CreateSession, ActivateSession, CloseSession) à l'exception de l'utilisation d'ActivateSession pour changer l'utilisateur de la Session. Ceci inclut le traitement correct de tous les paramètres fournis. Pour les services CreateSession et ActivateSession, si le SecurityMode = None, alors: 1) le <i>Certificat</i> d'application et le Nonce sont facultatifs; 2) les signatures sont nulles/vides.
Client	Session Client "Connexions multiples"	Prend en charge un nombre illimité de connexions (côté client) avec plusieurs Serveurs. S'il y a une limite pour le nombre de connexions, elle est du côté serveur. Peut comporter une limite de mémoire, mais aucune limite de capacité logicielle.
Client	Session Client "Renouveler les Nodelds"	Cette <i>ConformanceUnit</i> s'applique aux Clients qui autorisent les Nodelds conservés. Vérifie que le tableau d'espace de noms n'a pas été modifié pour les Nodelds que le Client a conservés et va réutiliser au-delà de la durée de vie de la Session. Si des modifications sont intervenues, le Client doit recalculer les Indices d'espace de noms des Nodelds respectifs.
Client	Session Client "Masquer"	Utilise ActivateSession pour changer l'utilisateur de la Session (masquage).
Client	Session Client "KeepAlive"	Formule des demandes périodiques de maintien de la Session active.

Catégorie	Intitulé	Description
<i>Client</i>	<i>Session Client</i> "Détecter un arrêt"	Lit ou surveille la Variable <i>ServerStatus/State</i> afin de reconnaître un arrêt potentiel du <i>Serveur</i> et de "nettoyer" les ressources.
<i>Client</i>	<i>Session Client</i> "Annuler"	Utilise le <i>Service</i> Annuler pour annuler les requêtes en suspens.
<i>Client</i>	<i>Session Client</i> "Reconnexion automatique"	Reconnexion automatique du <i>Client</i> , y compris: – <i>ActivateSession</i> avec nouveau <i>SecureChannel</i> si le <i>SecureChannel</i> existant n'est plus valide, mais que la <i>Session</i> est toujours valide; – la création d'une nouvelle <i>Session</i> uniquement si la <i>Session</i> n'est plus valide
<i>Client</i>	<i>Session Client</i> "Session unique"	Le <i>Client</i> doit interagir avec des <i>Serveurs</i> qui prennent en charge une seule <i>Session</i> .
<i>Client</i>	<i>Session Client</i> "Appels de Services <i>SessionLess</i> "	Définit l'utilisation du <i>Service</i> "SessionlessInvoke" défini dans l'IEC 62541-4 pour traiter l'un des <i>Services</i> (comme <i>Lecture/Ecriture</i> ou <i>Parcourir</i>) qui sont admis pour l'invocation sans session. L'IEC 62541-6 spécifie quels transports peuvent être utilisés et comment.

Le *Jeu de services Gestion des Nœuds* comprend plusieurs *ConformanceUnits* (voir Tableau 5). Les *Serveurs* peuvent fournir certains aspects de cette fonctionnalité; se reporter aux *Profils* catégorisés comme *Profils de serveur* pour plus d'informations. Les *Clients* peuvent prendre en charge certains aspects de cette fonctionnalité; se reporter aux *Profils* catégorisés comme *Profils de client* pour plus d'informations.

Tableau 5 – Services Gestion des Nœuds

Catégorie	Intitulé	Description
<i>Serveur</i>	Gestion des <i>Nœuds</i> "Ajouter des <i>Nœuds</i> "	Prend en charge le <i>Service</i> <i>AddNodes</i> pour ajouter un ou plusieurs <i>Nœuds</i> dans l' <i>AddressSpace</i> OPC UA.
<i>Serveur</i>	Gestion des <i>Nœuds</i> "Supprimer des <i>Nœuds</i> "	Prend en charge le <i>Service</i> <i>DeleteNodes</i> pour supprimer un ou plusieurs <i>Nœuds</i> de l' <i>AddressSpace</i> OPC UA.
<i>Serveur</i>	Gestion des <i>Nœuds</i> "Ajouter Référence"	Prend en charge le <i>Service</i> <i>AddReferences</i> pour ajouter une ou plusieurs <i>Références</i> à un ou plusieurs <i>Nœuds</i> dans l' <i>AddressSpace</i> OPC UA.
<i>Serveur</i>	Gestion des <i>Nœuds</i> "Supprimer des <i>Références</i> "	Prend en charge le <i>Service</i> <i>DeleteReferences</i> pour supprimer une ou plusieurs <i>Références</i> d'un <i>Nœud</i> dans l' <i>AddressSpace</i> OPC UA.
<i>Client</i>	Gestion des <i>Nœuds</i> <i>Client</i>	Utilise les <i>Services</i> Gestion des <i>Nœuds</i> pour ajouter ou supprimer des <i>Nœuds</i> et pour ajouter ou supprimer des <i>Références</i> dans l' <i>AddressSpace</i> OPC UA du <i>Serveur</i> .

Le *Jeu de services Vue* comprend plusieurs *ConformanceUnits* (voir Tableau 6). Tous les *Serveurs* prennent en charge certains aspects de ce groupe de conformité. Les *Clients* peuvent prendre en charge certains aspects de cette fonctionnalité; se reporter aux *Profils* catégorisés comme *Profils de client* pour plus d'informations.

Tableau 6 – Services Vue

Catégorie	Intitulé	Description
<i>Serveur</i>	Vue de base	Prend en charge le <i>Jeu de services</i> Vue (Browse, BrowseNext).
<i>Serveur</i>	Vue "TranslateBrowsePath"	Prend en charge le <i>Service</i> TranslateBrowsePathsToNodeIds
<i>Serveur</i>	Vue "RegisterNodes"	Prend en charge les <i>Services</i> RegisterNodes et UnregisterNodes comme méthode d'optimisation de l'accès aux <i>Nœuds</i> à utilisation répétitive dans l' <i>AddressSpace</i> OPC UA du <i>Serveur</i> .
<i>Serveur</i>	Vue "Point de continuation minimum 01"	Prend en charge au minimum 1 point de continuation par <i>Session</i> .
<i>Serveur</i>	Vue "Point de continuation minimum 05"	Prend en charge au minimum 5 points de continuation par <i>Session</i> . Ce nombre doit être pris en charge pour au moins la moitié des sessions minimales exigées.
<i>Client</i>	Vue <i>Client</i> "Parcourir" de base	Utilise les <i>Services</i> Parcourir et BrowseNext pour naviguer dans l' <i>AddressSpace</i> OPC UA du <i>Serveur</i> . Utilise le <i>referenceTypeId</i> et le <i>nodeClassMask</i> pour spécifier les <i>Références</i> nécessaires.
<i>Client</i>	Vue <i>Client</i> "Parcourir les <i>Nœuds</i> distants"	Le <i>Client</i> peut naviguer jusqu'aux nœuds ayant un <i>NodeId</i> étendu qui référence un <i>Serveur</i> différent du <i>Serveur</i> d'origine. Cela comprend la connexion automatique au <i>Serveur</i> distant. Il est acceptable que les informations de configuration du <i>Serveur</i> soient préconfigurées sur le <i>Client</i> et/ou que l'utilisateur soit invité à se connecter.
<i>Client</i>	Vue <i>Client</i> "Filtrage de base du ResultSet"	Utilise le paramètre <i>ResultMask</i> pour optimiser l'ensemble de résultats que le <i>Serveur</i> a à renvoyer.
<i>Client</i>	Vue <i>Client</i> "TranslateBrowsePath"	Utilise le <i>Service</i> TranslateBrowsePathsToNodeIds pour identifier les <i>NodeIds</i> pour lesquels un <i>Nœud</i> de départ et un <i>BrowsePath</i> sont connus. Effectue des opérations d'ensemble plutôt que des appels multiples lorsque cela est possible.
<i>Client</i>	Vue <i>Client</i> "Parcourir la Traduction des <i>Nœuds</i> distants"	Le <i>Client</i> peut traduire les chemins de navigation comprenant des nœuds avec un <i>NodeId</i> qui référence un <i>Serveur</i> différent du <i>Serveur</i> d'origine et les renvoyer dans le cadre du <i>Service</i> TranslateBrowsePathsToNodeIds. Il est acceptable que les informations de configuration du <i>Serveur</i> soient préconfigurées sur le <i>Client</i> .
<i>Client</i>	Vue <i>Client</i> "RegisterNodes"	Utilise le <i>Service</i> RegisterNodes pour optimiser l'accès aux <i>Nœuds</i> à utilisation répétitive. Utilise le <i>Service</i> UnregisterNodes lorsque les <i>Nœuds</i> ne sont plus utilisés.

Le *Jeu de services Attribut* comprend plusieurs *ConformanceUnits* (voir Tableau 7). La majorité du *Jeu de services Attribut* constitue une fonctionnalité principale de l'OPC UA et est de ce fait pris en charge par la plupart des *Serveurs*. La plupart des *Clients* prennent également en charge certains aspects du *Jeu de services Attribut*.

Tableau 7 – Services Attribut

Catégorie	Intitulé	Description
<i>Serveur</i>	<i>Attribut Lecture</i>	Prend en charge le <i>Service Lecture</i> afin de déchiffrer un ou plusieurs <i>Attributs</i> d'un ou plusieurs <i>Nœuds</i> . Ceci inclut la prise en charge du paramètre <i>IndexRange</i> afin de déchiffrer un seul élément ou un ensemble d'éléments lorsque la valeur d' <i>Attribut</i> constitue une matrice.
<i>Serveur</i>	<i>Attribut Lecture Complexe</i>	Prend en charge la lecture et le codage de Valeurs avec des <i>DataTypes</i> structurés.
<i>Serveur</i>	<i>Attribut Ecriture des Valeurs</i>	Prend en charge l'écriture de valeurs à un ou plusieurs <i>Attributs</i> d'un ou plusieurs <i>Nœuds</i> .
<i>Serveur</i>	<i>Attribut Ecriture Complexe</i>	Prend en charge l'écriture et le codage de Valeurs avec <i>DataTypes</i> structurés.
<i>Serveur</i>	<i>Attribut Ecriture de StatusCode & Horodatage</i>	Prend en charge l'écriture du <i>StatusCode</i> et des Horodatages associés à la Valeur.
<i>Serveur</i>	<i>Attribut Ecriture d'Indices</i>	Prend en charge le paramètre <i>IndexRange</i> pour écrire un seul élément ou un ensemble d'éléments lorsque la valeur d' <i>Attribut</i> constitue une matrice et que les mises à jour partielles sont admises pour cette matrice.
<i>Serveur</i>	<i>Attribut Codage alternatif</i>	Prend en charge le codage alternatif des données lors de la lecture des <i>Attributs</i> de la Valeur. Par défaut, chaque <i>Serveur</i> doit prendre en charge le codage des données du <i>Profil</i> de Pile réellement utilisé (c'est-à-dire binaire avec le Codage UA binaire et XML avec le Codage XML). Cette <i>ConformanceUnit</i> (lorsqu'elle est prise en charge) spécifie que l'autre codage des données est également pris en charge.
<i>Serveur</i>	<i>Attribut Lecture d'Histoire</i>	Prend en charge le <i>Service HistoryRead</i> . Les détails des aspects utilisés par ce service sont décrits par des <i>ConformanceUnits</i> supplémentaires, mais il importe qu'au moins l'une des unités suivantes soit prise en charge: <i>ReadRaw</i> , <i>ReadProcessed</i> , <i>ReadModified</i> , <i>ReadAtTime</i> ou <i>ReadEvents</i> .
<i>Serveur</i>	<i>Attribut Mise à jour de l'Histoire</i>	Prend en charge le <i>Service HistoryUpdate</i> . Les détails des caractéristiques prises en charge de ce service sont décrits par des <i>ConformanceUnits</i> supplémentaires, mais il importe qu'au moins l'une des unités suivantes soit prise en charge: <i>InsertData</i> , <i>InsertEvents</i> , <i>ReplaceData</i> , <i>ReplaceEvents</i> , <i>UpdateData</i> , <i>UpdateEvents</i> , <i>DeleteData</i> , <i>DeleteEvents</i> ou <i>DeleteAtTime</i> .
<i>Client</i>	<i>Attribut Client Lecture de base</i>	Utilise le <i>Service Lecture</i> pour lire un ou plusieurs <i>Attributs</i> de un ou plusieurs <i>Nœuds</i> . Ceci inclut l'utilisation d'un <i>IndexRange</i> pour choisir un seul élément ou un ensemble d'éléments lorsque la valeur d' <i>Attribut</i> constitue une matrice. Les <i>Clients</i> doivent effectuer des opérations d'ensemble lorsque cela est possible, afin de réduire le nombre d'invocations de <i>Service</i> .
<i>Client</i>	<i>Attribut Client Accès à l'Attribut des Nœuds distants</i>	Le <i>Client</i> peut extraire les attributs des nœuds ayant un <i>Identificateur</i> de <i>NodeID</i> qui référence un <i>Serveur</i> différent du <i>Serveur</i> d'origine. Cela exige une connexion au <i>Serveur</i> distant pour pouvoir y accéder (pas nécessairement affichée comme une connexion). Il est acceptable que les informations de configuration du <i>Serveur</i> soient préconfigurées sur le <i>Client</i> .
<i>Client</i>	<i>Attribut Client Lecture avec Codage approprié</i>	Cette <i>ConformanceUnit</i> fait référence à la capacité d'un <i>Client</i> à découvrir les codages disponibles et à en choisir un spécifique lorsqu'il appelle le <i>Service Lecture</i> .
<i>Client</i>	<i>Attribut Client Lecture Complexe</i>	Lecture et décodage de Valeurs avec des <i>DataTypes</i> structurés.
<i>Client</i>	<i>Attribut Client Ecriture de base</i>	Utilise le <i>Service Ecriture</i> pour écrire des valeurs à un ou plusieurs <i>Attributs</i> d'un ou plusieurs <i>Nœuds</i> . Ceci inclut l'utilisation d'un <i>IndexRange</i> pour choisir un seul élément ou un ensemble d'éléments lorsque la valeur d' <i>Attribut</i> constitue une matrice. Les <i>Clients</i> doivent effectuer des opérations d'ensemble lorsque cela est possible, afin de réduire le nombre d'invocations de <i>Service</i> .
<i>Client</i>	<i>Attribut Client Ecriture Complexe</i>	Ecriture et codage de Valeurs avec des <i>DataTypes</i> structurés.
<i>Client</i>	<i>Attribut Client Ecriture Qualité & Horodatage</i>	Utilise le <i>Service Ecriture</i> pour écrire également le <i>StatusCode</i> et/ou les Horodatages associés à une Valeur.

Catégorie	Intitulé	Description
<i>Client</i>	<i>Attribut Client</i> Lecture d'Historique	Le <i>Client</i> utilise le service HistoryRead. Les détails des aspects utilisés par ce service sont fournis par des <i>ConformanceUnits</i> supplémentaires, mais au moins une ou plusieurs des unités suivantes sont utilisées: ReadRaw, ReadAtTime, ReadProcessed, ReadModified ou ReadEvents.
<i>Client</i>	<i>Attribut Client</i> Mises à jour de l'Historique	Le <i>Client</i> utilise le service HistoryUpdate. Les détails de cette utilisation sont décrits par des <i>ConformanceUnits</i> supplémentaires, mais il importe qu'au moins l'une des unités suivantes soit fournie: InsertData, InsertEvents, ReplaceData, ReplaceEvents, UpdateData, UpdateEvents, DeleteData ou DeleteEvents ou DeleteAtTime.

Le *Jeu de services Méthode* est constitué de *ConformanceUnits* (voir Tableau 8). Les *ConformanceUnits* principales assurent la prise en charge de la fonctionnalité d'appel. Les *Serveurs* peuvent fournir certains aspects de cette fonctionnalité; se reporter aux *Profils* catégorisés comme *Profils de serveur* pour plus d'informations. Les *Clients* peuvent prendre en charge certains aspects de cette fonctionnalité; se reporter aux *Profils* catégorisés comme *Profils de client* pour plus d'informations.

Tableau 8 – Services Méthode

Catégorie	Intitulé	Description
<i>Serveur</i>	<i>Méthode</i> Appel	Prend en charge le <i>Service Appel</i> pour appeler (invoquer) une <i>Méthode</i> qui inclut la prise en charge des paramètres de <i>Méthode</i> .
<i>Client</i>	<i>Méthode Client</i> Appel	Utilise le <i>Service Appel</i> pour appeler une ou plusieurs <i>Méthodes</i> .

Le *Jeu de services MonitoredItems* comprend plusieurs *ConformanceUnits* (voir Tableau 9). Les *Serveurs* peuvent fournir certains aspects de cette fonctionnalité; se reporter aux *Profils* catégorisés comme *Profils de serveur* pour plus d'informations. Les *Clients* peuvent prendre en charge certains aspects de cette fonctionnalité; se reporter aux *Profils* catégorisés comme *Profils de client* pour plus d'informations.

Tableau 9 – Services Eléments surveillés

Catégorie	Intitulé	Description
<i>Serveur</i>	Contrôle de base	Prend en charge les <i>Services MonitoredItem</i> suivants: CreateMonitoredItems, ModifyMonitoredItems, DeleteMonitoredItems, SetMonitoringMode.
<i>Serveur</i>	Contrôle Modification de valeur	Prend en charge la création de <i>MonitoredItems</i> pour les modifications de valeurs d' <i>Attributs</i> . Ceci inclut la prise en charge d' <i>IndexRange</i> afin de choisir un seul élément ou un ensemble d'éléments lorsque la valeur <i>Attribut</i> constitue une matrice.
<i>Serveur</i>	Contrôle de Valeur Complexe	Prend en charge la surveillance et le codage de Valeurs avec des <i>DataTypes</i> structurés.
<i>Serveur</i>	Filtre de Bande Morte d'Eléments Surveillés	Prend en charge un Filtre de bande morte absolu comme <i>DataChangeFilter</i> pour les types de données numériques.
<i>Serveur</i>	Contrôle de Filtre d'Agrégat	Prend en charge les filtres d'Agrégat pour les <i>MonitoredItems</i> . Le résultat de ces <i>ConformanceUnits</i> inclut une liste d'Agrégats qui sont pris en charge en tant que partie du <i>Certificat de Profil</i> .
<i>Serveur</i>	Contrôle de Codage alternatif	Prend en charge le codage alternatif lors de la surveillance des <i>Attributs</i> de la valeur. Par défaut, chaque <i>Serveur</i> doit prendre en charge le codage du <i>Profil</i> de Pile réellement utilisé (c'est-à-dire binaire avec le Codage UA binaire et XML avec le Codage XML). Cette <i>ConformanceUnit</i> (lorsqu'elle est prise en charge) spécifie que l'autre codage des données est également pris en charge.
<i>Serveur</i>	Contrôle de 2 éléments	Prend en charge au moins 2 <i>MonitoredItems</i> par <i>Abonnement</i> , où la taille de chaque <i>MonitoredItem</i> est au moins égale à la taille du Double.
<i>Serveur</i>	Contrôle de 10 éléments	Prend en charge au moins 10 <i>MonitoredItems</i> par <i>Abonnement</i> , où la taille de chaque <i>MonitoredItem</i> est au moins égale à la taille du Double.
<i>Serveur</i>	Contrôle de 100 éléments	Prend en charge au moins 100 <i>MonitoredItems</i> par <i>Abonnement</i> . Ce nombre doit être pris en charge pour au moins la moitié des Abonnements exigés, pour la moitié des Sessions exigées.
<i>Serveur</i>	Contrôle de 500 éléments	Prend en charge au moins 500 <i>MonitoredItems</i> par <i>Abonnement</i> . Ce nombre doit être pris en charge pour au moins la moitié des Abonnements exigés, pour la moitié des Sessions exigées.
<i>Serveur</i>	Contrôle de QueueSize_1	Cette <i>ConformanceUnit</i> n'exige pas la mise en file d'attente lorsque plusieurs modifications de valeurs surviennent pendant une période d'édition. Autrement dit, la dernière modification est envoyée dans la <i>Notification</i> .
<i>Serveur</i>	Contrôle de MinQueueSize_02	Prend en charge au moins 2 entrées de file d'attente pour les <i>MonitoredItems</i> . Les <i>Serveurs</i> adaptent souvent la taille de la file d'attente au nombre réel de <i>MonitoredItems</i> . Toutefois, il est prévu que les <i>Serveurs</i> prennent en charge cette taille de file d'attente minimale pour au moins un tiers des <i>MonitoredItems</i> pris en charge.
<i>Serveur</i>	Contrôle de MinQueueSize_05	Prend en charge au moins 5 entrées de file d'attente pour les <i>MonitoredItems</i> . Les <i>Serveurs</i> adaptent souvent la taille de la file d'attente au nombre réel de <i>MonitoredItems</i> . Toutefois, il est prévu que les <i>Serveurs</i> prennent en charge cette taille de file d'attente minimale pour au moins un tiers des <i>MonitoredItems</i> pris en charge.
<i>Serveur</i>	Contrôle de QueueSize_ServerMax	Cette <i>ConformanceUnit</i> s'applique aux Evénements. Lorsque le <i>Client</i> effectue une demande de queuesize=MAXUInt32, le <i>Serveur</i> doit retourner la taille maximale de la file d'attente qu'il peut prendre en charge pour les notifications d'événement en tant que revisedQueueSize.
<i>Serveur</i>	Contrôle de Déclenchement	Prend en charge le <i>Service SetTriggering</i> pour créer et/ou supprimer les liens actifs d'un élément déclencheur.

Catégorie	Intitulé	Description
<i>Serveur</i>	Contrôle d'Événements	Prend en charge la création des <i>MonitoredItems</i> pour un "Attribut <i>EventNotifier</i> " à des fins de <i>Notification d'Événements</i> . L'abonnement inclut la prise en charge d'un filtre qui comporte des opérandes <i>SimpleAttribute</i> et une liste sélective d'Opérateurs. La liste des Opérateurs inclut: <i>Equals</i> , <i>IsNull</i> , <i>GreaterThan</i> , <i>LessThan</i> , <i>GreaterThanEqual</i> , <i>LessThanEqual</i> , <i>Like</i> , <i>Not</i> , <i>Between</i> , <i>InList</i> , <i>And</i> , <i>Or</i> , <i>Cast</i> , <i>BitwiseAnd</i> , <i>BitwiseOr</i> .
<i>Serveur</i>	Contrôle de Filtre d'Événements complexes	Prend en charge l'opérateur de filtre "TypeOf" pour les <i>Événements</i> complexes.
<i>Client</i>	Contrôle <i>Client</i> Modification de valeur	Utilise le <i>Jeu de services MonitoredItem</i> pour enregistrer les éléments propres aux modifications de la valeur d'Attribut. Utiliser <i>CreateMonitoredItems</i> pour enregistrer la séquence <i>Nœud/Attribut</i> . Définir l'intervalle d'échantillonnage, le Filtre de Bande Morte et le mode de mise en file d'attente appropriés. Utilise le mode désactivation/activation de préférence au mode suppression et recréation d'un <i>MonitoredItem</i> . Exécuter des opérations d'ensemble de préférence plutôt que des requêtes de service individuel pour réduire les volumes de communication.
<i>Client</i>	Contrôle <i>Client</i> de Valeur Complexe	Surveille et décode les Valeurs avec des <i>DataTypes</i> structurés.
<i>Client</i>	Contrôle <i>Client</i> de Filtre de Bande Morte	Utilise les filtres de Bande Morte Absolus pour les abonnements.
<i>Client</i>	Contrôle <i>Client</i> par Indice	Utiliser <i>IndexRange</i> pour choisir un seul élément ou un ensemble d'éléments lorsque la valeur d'Attribut constitue une matrice.
<i>Client</i>	Contrôle <i>Client</i> de Filtre d'Agrégat	Utilise des filtres d'Agrégat pour les Abonnements.
<i>Client</i>	Contrôle <i>Client</i> d'Événements	Utilise le <i>Jeu de services MonitoredItem</i> pour créer les <i>MonitoredItems</i> destinés aux notifications d'événements.
<i>Client</i>	Contrôle <i>Client</i> de Filtre d'Événements	Utilise le filtre d'Événement pour appeler le Service <i>CreateMonitoredItems</i> afin de filtrer les Événements souhaités et de choisir les colonnes à prévoir pour chaque <i>Notification d'Événement</i> .
<i>Client</i>	Contrôle <i>Client</i> de Filtre d'Événements complexes	Utilise l'opérateur de filtre "TypeOf" pour les <i>Événements</i> complexes.
<i>Client</i>	Contrôle <i>Client</i> de Modification	Utilise le Service <i>ModifyMonitoredItems</i> pour modifier les paramètres de configuration. Utilise le Service <i>SetMonitoringMode</i> pour désactiver/activer l'échantillonnage et/ou l'édition.
<i>Client</i>	Contrôle <i>Client</i> de Déclenchement	Utilise le Modèle de déclenchement si certains éléments doivent être consignés uniquement en cas de déclenchement d'autres éléments spécifiques. Utilise le mode de surveillance approprié pour ces éléments. Utilise le Service <i>SetTriggering</i> pour relier ces éléments à l'élément déclencheur.

Le *Jeu de services Subscription* comprend plusieurs *ConformanceUnits* (voir Tableau 10). Les *Serveurs* peuvent fournir certains aspects de cette fonctionnalité; se reporter aux *Profils* catégorisés comme *Profils de serveur* pour plus d'informations. Les *Clients* peuvent prendre en charge certains aspects de cette fonctionnalité; se reporter aux *Profils* catégorisés comme *Profils de client* pour plus d'informations.

Tableau 10 – Services Abonnement

Catégorie	Intitulé	Description
Serveur	Abonnement de base	Prend en charge les <i>Services Abonnement</i> suivants: CreateSubscription, ModifySubscription, DeleteSubscriptions, Editer, Rééditer et SetPublishingMode
Serveur	Abonnement Minimum 1	Prend en charge au moins 1 <i>Abonnement</i> par <i>Session</i> . Ce nombre doit être pris en charge pour l'ensemble des sessions minimales exigées.
Serveur	Abonnement Minimum 02	Prend en charge au moins 2 <i>Abonnements</i> par <i>Session</i> . Ce nombre doit être pris en charge pour au moins la moitié des sessions minimales exigées.
Serveur	Abonnement Minimum 05	Prend en charge au moins 5 <i>Abonnements</i> par <i>Session</i> . Ce nombre doit être pris en charge pour au moins la moitié des sessions minimales exigées.
Serveur	Abonnement Editer Min 02	Prend en charge au moins 2 demandes de <i>Service "Editer"</i> par <i>Session</i> . Ce nombre doit être pris en charge pour l'ensemble des sessions minimales exigées. La prise en charge d'une file d'attente pour la retransmission du NotificationMessage n'est pas exigée; si elle n'est pas disponible, le <i>Service "Rééditer"</i> renvoie Bad_MessageNotAvailable.
Serveur	Abonnement Editer Min 05	Prend en charge au moins 5 demandes de <i>Service "Editer"</i> par <i>Session</i> . Ce nombre doit être pris en charge pour au moins la moitié des sessions minimales exigées. Prend en charge au minimum le nombre de requêtes d'édition par session comme taille de la file de retransmission du NotificationMessage à des fins de réédition.
Serveur	Abonnement Editer Min 10	Prend en charge au moins 10 demandes de <i>Service "Editer"</i> par <i>Session</i> . Ce nombre doit être pris en charge pour au moins la moitié des sessions minimales exigées. Prend en charge au minimum le nombre de requêtes d'édition par session comme taille de la file de retransmission du NotificationMessage à des fins de réédition.
Serveur	Abonnement Politique d'élimination d'édition	Respecte la politique spécifiée d'élimination des demandes de <i>Service "Editer"</i> . Si le nombre maximal des demandes de <i>Service "Editer"</i> est dans une file d'attente et si une nouvelle demande de <i>Service "Editer"</i> se présente, la demande la "plus ancienne" doit être éliminée en renvoyant l'erreur appropriée.
Serveur	Abonnement Transfert	Prend en charge le <i>Service TransferSubscriptions</i> pour un transfert d' <i>Abonnement</i> d'une <i>Session</i> à l'autre.
Serveur	Abonnement Durable	Prend en charge la configuration des Abonnements en mode durable. Ce mode exige que les données et les événements collectés soient stockés et distribués même si un <i>Client</i> est déconnecté depuis longtemps ou si le <i>Serveur</i> a été redémarré. Prend en charge l'une des <i>ConformanceUnits "Abonnement Durable StorageLevel nnn"</i> .
Serveur	Abonnement Durable StorageLevel Faible	Prend en charge au moins 20 éléments surveillés avec une file d'attente de 10 000 pour chaque élément et lorsque la taille de chaque <i>MonitoredItem</i> est au moins égale à la taille du Double. Une capacité de stockage 200 000 valeurs est exigée pour le DataType Double.
Serveur	Abonnement Durable StorageLevel Moyen	Prend en charge au moins 100 éléments surveillés avec une file d'attente de 50 000 pour chaque élément et lorsque la taille de chaque <i>MonitoredItem</i> est au moins égale à la taille du Double. Une capacité de stockage de 5 millions de valeurs est exigée pour le DataType Double.
Serveur	Abonnement Durable StorageLevel Elevé	Prend en charge au moins 2 000 éléments surveillés avec une file d'attente de 200 000 pour chaque élément et lorsque la taille de chaque <i>MonitoredItem</i> est au moins égale à la taille du Double. Une capacité de stockage de 400 millions de valeurs est exigée pour le DataType Double.

Catégorie	Intitulé	Description
<i>Client</i>	<i>Abonnement Client de base</i>	Utilise le <i>Jeu de services Subscription</i> et <i>MonitoredItems</i> comme moyen efficace de détecter les modifications de valeurs d' <i>Attributs</i> et/ou de recevoir les occurrences d' <i>Evénements</i> . Définir les intervalles d'édition appropriés et maintenir actives les notifications et la durée de vie totale d' <i>Abonnement</i> . Fournir un nombre suffisant de demandes "Editor" au <i>Serveur</i> de manière à pouvoir transmettre les <i>Notifications</i> à chaque expiration d'un temporisateur d'édition. Acquitter des <i>Notifications</i> reçues avec les requêtes d'édition ultérieures.
<i>Client</i>	<i>Abonnement Client Repli</i>	Le <i>Client</i> doit interagir avec les <i>Serveurs</i> qui ne prennent pas en charge les Abonnements, ou qui ont atteint leur nombre limite d' <i>Abonnements</i> , pour la Surveillance au moyen du <i>Service Lecture</i> .
<i>Client</i>	<i>Abonnement Réédition Client</i>	Evalue le nombre de séquences des <i>Notifications</i> afin de détecter les <i>Notifications</i> perdues. Utilise "Rééditer" pour demander les <i>Notifications</i> manquantes.
<i>Client</i>	<i>Abonnement Modification Client</i>	Permet la modification de la configuration d' <i>Abonnement</i> en utilisant le <i>Service ModifySubscription</i> .
<i>Client</i>	<i>Abonnement Client TransferSubscriptions</i>	Le <i>Client</i> prend en charge les <i>Abonnements</i> de transfert d'autres <i>Clients</i> . Cette <i>ConformanceUnit</i> est utilisée dans le cadre des <i>Clients</i> redondants.
<i>Client</i>	<i>Abonnement Client Multiple</i>	Utilise les différents abonnements pour réduire la charge utile des <i>Notifications</i> individuelles.
<i>Client</i>	<i>Abonnement Edition configurable Client</i>	Transmet plusieurs demandes de <i>Service "Editor"</i> afin de s'assurer que le <i>Serveur</i> est toujours capable de transmettre des <i>Notifications</i> . Le nombre de demandes parallèles de <i>Service "Editor"</i> par <i>Session</i> doit être configurable.
<i>Client</i>	<i>Abonnement Durable Client</i>	Utilise les abonnements durables.

5.3 Caractéristiques relatives au transport et à la communication

Le Tableau 11 décrit les *ConformanceUnits* relatives à la sécurité. Toutes ces *ConformanceUnits* s'appliquent de manière égale aux *Clients* et aux *Serveurs*, lorsqu'un *Client* utilise l'unité relative à la sécurité et un *Serveur* prend en charge l'utilisation de cette dernière. Ces éléments sont définis en détail dans l'IEC 62541-6. Il est recommandé que le *Serveur* et le *Client* prennent en charge le plus grand nombre possible de ces options afin d'obtenir des niveaux d'interopérabilité plus élevés. Il revient à l'administrateur de déterminer quelle *ConformanceUnit* est présentée dans une application *Serveur* ou *Client* déployée donnée.

Tableau 11 – Sécurité

Catégorie	Intitulé	Description
Sécurité	Sécurité Nom d'Utilisateur Mot de Passe	Le Serveur prend en charge les combinaisons Nom d'utilisateur/Mot de passe. Le jeton est chiffré si la politique de sécurité de la Politique relative aux jetons d'utilisateurs ou la politique de sécurité du point d'extrémité l'exige. Un jeton non chiffré exige soit le chiffrement du message, soit un dispositif hors de l'OPC UA permettant de sécuriser le jeton d'identité afin qu'il ne puisse pas être récupéré par l'écoute des communications (sniffing). Un transport sécurisé comme un VPN serait une option.
Sécurité	Sécurité utilisateur X509	Le <i>Serveur</i> prend en charge une paire de clés publique/privée pour l'identité de l'utilisateur. Il importe que l'utilisation de cette caractéristique puisse être activée ou désactivée par un administrateur.
Sécurité	Sécurité utilisateur IssuedToken Kerberos	Le <i>Serveur</i> prend en charge un jeton de <i>Serveur</i> Kerberos pour l'identité de l'Utilisateur. Il importe que l'utilisation de cette caractéristique puisse être activée ou désactivée par un administrateur. L'utilisation de ce jeton est définie dans la Documentation de jeton Kerberos. Le jeton est chiffré si la politique de sécurité de la Politique relative aux jetons d'utilisateurs ou la politique de sécurité du point d'extrémité l'exige. Un jeton non chiffré exige soit le chiffrement du message, soit un dispositif hors de l'OPC UA permettant de sécuriser le jeton d'identité afin qu'il ne puisse pas être récupéré par l'écoute des communications (sniffing). Un transport sécurisé comme un VPN serait une option.
Sécurité	Sécurité utilisateur IssuedToken Kerberos Windows	Le <i>Serveur</i> prend en charge la mise en œuvre Windows des jetons Kerberos. Cette <i>ConformanceUnit</i> ne s'applique que si la "Sécurité utilisateur IssuedToken Kerberos" est prise en charge. Le jeton est chiffré si la politique de sécurité de la Politique relative aux jetons d'utilisateurs ou la politique de sécurité du point d'extrémité l'exige. Un jeton non chiffré exige soit le chiffrement du message, soit un dispositif hors de l'OPC UA permettant de sécuriser le jeton d'identité afin qu'il ne puisse pas être récupéré par l'écoute des communications (sniffing). Un transport sécurisé comme un VPN serait une option.
Sécurité	Sécurité utilisateur IssuedToken JWT	Le <i>Serveur</i> prend en charge un jeton JWT (JSON Web Token) pour l'identité de l'Utilisateur. L'IEC 62541-6 décrit OAuth2 et les jetons JWT plus en détail. Il importe que l'utilisation de cette caractéristique puisse être activée ou désactivée par un administrateur. Le jeton est chiffré si la politique de sécurité de la Politique relative aux jetons d'utilisateurs ou la politique de sécurité du point d'extrémité l'exige. Un jeton non chiffré exige soit le chiffrement du message, soit un dispositif hors de l'OPC UA permettant de sécuriser le jeton d'identité afin qu'il ne puisse pas être récupéré par l'écoute des communications (sniffing). Un transport sécurisé comme un VPN serait une option.
Sécurité	Sécurité utilisateur Anonyme	Le <i>Serveur</i> prend en charge l'accès Anonyme. Il importe que l'utilisation de cette caractéristique puisse être activée ou désactivée par un administrateur. L'accès Anonyme doit être désactivé par défaut.
Sécurité	Sécurité utilisateur Nom d'utilisateur/Mot de passe <i>Client</i>	Un <i>Client</i> utilise une combinaison Nom d'utilisateur/Mot de passe. Le jeton est chiffré si la politique de sécurité de la Politique relative aux jetons d'utilisateurs ou la politique de sécurité du point d'extrémité l'exige. Un jeton non chiffré exige soit le chiffrement du message, soit un dispositif hors de l'OPC UA permettant de sécuriser le jeton d'identité afin qu'il ne puisse pas être récupéré par l'écoute des communications (sniffing). Un transport sécurisé comme un VPN serait une option.
Sécurité	Sécurité utilisateur X509 <i>Client</i>	Un <i>Client</i> utilise une paire de clés publique/privée pour l'identité de l'utilisateur. Cela comprend tous les problèmes de validation et de confiance associés à un certificat.
Sécurité	Sécurité utilisateur IssuedToken Kerberos <i>Client</i>	Un <i>Client</i> utilise un jeton de <i>Serveur</i> Kerberos. L'utilisation de ce jeton est définie dans la documentation Kerberos. Le jeton est chiffré si la politique de sécurité de la Politique relative aux jetons d'utilisateurs ou la politique de sécurité du point d'extrémité l'exige. Un jeton non chiffré exige soit le chiffrement du message, soit un dispositif hors de l'OPC UA permettant de sécuriser le jeton d'identité afin qu'il ne puisse pas être récupéré par l'écoute des communications (sniffing). Un transport sécurisé comme un VPN serait une option.

Catégorie	Intitulé	Description
Sécurité	Sécurité utilisateur IssuedToken Kerberos Windows Client	Un <i>Client</i> utilise la mise en œuvre Windows des jetons Kerberos. Cette <i>ConformanceUnit</i> ne s'applique que si la "Sécurité utilisateur <i>Client</i> IssuedToken Kerberos" est prise en charge. Le jeton est chiffré si la politique de sécurité de la Politique relative aux jetons d'utilisateurs ou la politique de sécurité du point d'extrémité l'exige. Un jeton non chiffré exige soit le chiffrement du message, soit un dispositif hors de l'OPC UA permettant de sécuriser le jeton d'identité afin qu'il ne puisse pas être récupéré par l'écoute des communications (sniffing). Un transport sécurisé comme un VPN serait une option.
Sécurité	Sécurité utilisateur IssuedToken JWT <i>Client</i>	Un <i>Client</i> utilise un jeton JWT (JSON Web Token) pour l'identité de l'utilisateur. L'IEC 62541-6 décrit OAuth2 et les jetons JWT plus en détail. Le jeton est chiffré si la politique de sécurité de la Politique relative aux jetons d'utilisateurs ou la politique de sécurité du point d'extrémité l'exige. Un jeton non chiffré exige soit le chiffrement du message, soit un dispositif hors de l'OPC UA permettant de sécuriser le jeton d'identité afin qu'il ne puisse pas être récupéré par l'écoute des communications (sniffing). Un transport sécurisé comme un VPN serait une option.
Sécurité	Sécurité Jeton d'utilisateur non valide	Les <i>Serveurs</i> doivent prendre des mesures appropriées pour se protéger des attaques contre les jetons d'identité d'utilisateur. Par hypothèse, ces attaques correspondent à des tentatives de connexion répétées au moyen de jetons d'identité d'utilisateur non valides. Voir <i>Service ActivateSession</i> dans l'IEC 62541-4.
Sécurité	Sécurité utilisateur Politique relative aux jetons JWT	Le <i>Serveur</i> prend en compte un ou plusieurs Points d'extrémité avec une <i>UserTokenPolicy</i> comprenant une <i>IssuerEndpointUrl</i> pour les jetons JWT définis dans l'IEC 62541-6. Pour les jetons JWT, l' <i>IssuerEndpointUrl</i> est un objet JSON incluant tous les paramètres qui définissent l' <i>AuthorizationService</i> . Selon la Politique relative aux jetons JWT, le <i>Serveur</i> doit prendre en charge au moins l'une des Unités de Conformité des Profils d'Autorité suivantes. Les URI définies dans la <i>ConformanceUnit</i> doivent être présentées dans le champ <i>authorityProfileURI</i> de la Politique relative aux jetons JWT.
Sécurité	Sécurité utilisateur Politique relative aux jetons JWT <i>Client</i>	Le <i>Client</i> comprend et utilise la définition du Service d'Autorisation incluse dans l' <i>UserTokenPolicy</i> pour les jetons JWT renvoyée avec <i>GetEndpoints</i> . Elle doit prendre en charge au moins l'une des Unités de Conformité de Profil d'Autorité suivantes. Les URI définies dans la <i>ConformanceUnit</i> figurent dans le champ <i>authorityProfileURI</i> de la Politique relative aux jetons JWT présentée dans les Points d'extrémité des <i>Serveurs</i> .
Sécurité	Profil d'autorité OAuth2	Cette unité indique la prise en charge de OAuth2 plutôt que HTTPS pour demander les jetons d'accès. L'URI pour les interactions avec cette autorité est "http://opcfoundation.org/UA/Authorization#OAuth2"
Sécurité	Profil d'autorité OPC UA	Cette unité indique la prise en charge des <i>Méthodes</i> OPC UA définies dans l'IEC 62541-12 pour demander les jetons d'accès. L'URI pour les interactions avec cette autorité est "http://opcfoundation.org/UA/Authorization#OPCUA"
Sécurité	Profil d'Autorité Fournisseur d'Identité Azure	Cette unité indique la prise en charge du Fournisseur d'Identité Azure pour demander les jetons d'accès. L'URI pour les interactions avec cette autorité est "http://opcfoundation.org/UA/Authorization#Azure"
Sécurité	Sécurité Validation de <i>Certificat</i>	Un <i>Certificat</i> est validé comme spécifié dans l'IEC 62541-4. Ceci inclut, entre autres, l'examen de la structure et de la signature. Permet la suppression de certaines erreurs de validation par directive administrative.
Sécurité	Sécurité Certificat ApplicationInstance par défaut	Une application, une fois installée, a par défaut un <i>ApplicationInstanceCertificate</i> valide. L' <i>ApplicationInstanceCertificate</i> par défaut doit soit être créé dans le cadre de l'installation, soit les instructions d'installation doivent décrire de façon explicite le processus pour créer et appliquer un <i>ApplicationInstanceCertificate</i> par défaut à l'application.

Catégorie	Intitulé	Description
Sécurité	Sécurité – Pas d'Authentification d'Application	Le <i>Serveur</i> prend en charge la capacité de n'être configuré pour aucune authentification d'application, seulement l'authentification de l'utilisateur et le mode de cryptage/signature normal: – le <i>Serveur</i> est configuré pour accepter tous les certificats; – les <i>Certificats</i> sont utilisés uniquement pour la sécurité des messages (signature et cryptage); – le niveau Utilisateurs est utilisé pour l'authentification.
Sécurité	Sécurité Politique exigée	Prend en charge au moins la Politique de sécurité [A] et la Politique de sécurité [B]. La prise en charge de plusieurs Politiques de sécurité (même obsolètes) est recommandée. Cela garantit une meilleure interopérabilité tout en permettant à l'utilisateur de choisir le niveau de sécurité exigé. Les Politiques de sécurité obsolètes ne doivent pas être activées/utilisables sans intervention de l'administrateur.
Sécurité	Security None CreateSession ActivateSession	Lorsque SecurityPolicy = None, les Services CreateSession et ActivateSession permettent une signature NULLE/vide et n'exigent ni <i>Certificat</i> d'application, ni Nonce.
Sécurité	Security None CreateSession ActivateSession 1.0	Le <i>Client</i> peut se connecter aux <i>Serveurs</i> qui exigent de transmettre un certificat lors de l'établissement de la Session. Dans ce cas, le <i>Client</i> effectue une première tentative sans <i>Certificat</i> et, en cas d'échec, présente un <i>Certificat</i> .
Sécurité	Sécurité TLS Générale	Cette <i>ConformanceUnit</i> indique qu'au moins un des <i>Profils</i> de sécurité transport pour TLS (sécurité de niveau de transport) est pris en charge par cette application. Elle est utilisée dans les <i>Profils</i> de transport TLS, mais le choix du profil de sécurité transport est facultatif. Le profil de sécurité réellement utilisé est par défaut le plus sécurisé.
Sécurité	Sécurité TLS_RSA avec AES_256_CBC_SHA256	La connexion est établie à l'aide de TLS_RSA_WITH_AES_256_CBC_SHA256. Ses caractéristiques sont MinAsymmetricKeyLength – 2048, MaxAsymmetricKeyLength – 4096, AsymmetricSignatureAlgorithm – RSA_SHA256. (TLS 1.2)
Sécurité	Sécurité TLS_DHE_RSA avec AES_nnn_CBC_SHA256	La connexion est établie à l'aide de TLS_DHE_RSA avec AES_128_CBC_SHA256 ou TLS_DHE_RSA avec AES_256_CBC_SHA256. Ses caractéristiques sont MinAsymmetricKeyLength – 2048, MaxAsymmetricKeyLength – 4096, CertificateSignatureAlgorithm – RSA_SHA256. (TLS 1.2). Les <i>Clients</i> et les <i>Serveurs</i> doivent prendre en charge les deux algorithmes.
Sécurité	Sécurité Cryptage exigé	Un cryptage utilisant les algorithmes fournis dans la suite d'algorithmes de sécurité est exigé.
Sécurité	Sécurité Signature exigée	Une signature utilisant les algorithmes fournis dans la suite d'algorithmes de sécurité est exigée.
Sécurité	Synchronisation Temporelle Sécurité – Configuration	L'application prend en charge la configuration d'un décalage d'horloge acceptable.
Sécurité	Synchronisation temporelle Sécurité – Prise en charge NTP/OS	L'application prend en charge la synchronisation temporelle, soit par la mise en œuvre du Protocole d'Heure Réseau (NTP – Network Time Protocol), soit par les caractéristiques d'un système d'exploitation normalisé.
Sécurité	Synchronisation temporelle Sécurité – Prise en charge UA	Une application utilise les réponses de l'horodatage d'en-tête fournies par une source connue configurée, telle qu'un <i>Serveur "Découverte"</i> , afin de synchroniser le temps sur l'application et afin que cette synchronisation temporelle se produise régulièrement. L'utilisation de cette TimeSyncing peut être configurée.

Catégorie	Intitulé	Description
Sécurité	Sécurité Administration	Permet la configuration des éléments suivants relatifs à la sécurité (si applicables). * choisit la(les) politique(s) autorisée(s) d'identification de l'utilisateur (par exemple, Nom/Mot de passe de l'Utilisateur ou X509) * active/désactive la politique de sécurité "None" ou autres politiques de sécurité * active/désactive les points d'extrémité avec MessageSecurityMode SIGN ou SIGNANDENCRYPT. * définit les autorités de certification admises * définit la méthode de réaction aux Certificats inconnus * permet d'accepter tous les Certificats valides
Sécurité	Sécurité Administration – Schéma XML	Prend en charge le schéma XML défini par OPC UA pour l'importation et l'exportation des informations de configuration de sécurité. Ce schéma est spécifié dans l'IEC 62541-6.
Sécurité	Sécurité <i>Certificat</i> d'Administration	Autoriser l'administrateur du site à pouvoir attribuer un ApplicationInstanceCertificate spécifique au site, et s'il le souhaite, configurer une Autorité de Certification (CA) spécifique au site.
Sécurité	Sécurité Rôle de base <i>Serveur</i>	Prend en charge le Modèle d'information sur les Autorisations d'utilisateurs défini dans l'UA, Parties 3 et 5 (comme les rôles) et les <i>Attributs</i> RolePermissions et UserRolePermissions.
Sécurité	Sécurité Rôle bien connu	Prend en charge les rôles connus "ConfigureAdmin" et "SecurityAdmin" avec les permissions suggérées définies dans l'IEC 62541-3.
Sécurité	Sécurité Rôle <i>Serveur</i> IdentityManagement	Permet aux utilisateurs autorisés d'ajouter et/ou de supprimer des Identités dans les Rôles selon les <i>Méthodes</i> adéquates.
Sécurité	Sécurité Rôle <i>Serveur</i> Gestion	Permet aux utilisateurs autorisés de créer de nouveaux Rôles et/ou de supprimer des Rôles selon les <i>Méthodes</i> adéquates.
Sécurité	Sécurité Rôle <i>Serveur</i> Restreindre Applications	Prend en charge l'ajout d'applications à un Rôle selon les <i>Méthodes</i> adéquates, afin que seules ces applications puissent utiliser ce Rôle
Sécurité	Sécurité Rôle <i>Serveur</i> Restreindre les Points d'extrémité	Prend en charge l'ajout de Points d'extrémité à un Rôle selon les <i>Méthodes</i> adéquates. Avec cette restriction, un Rôle est appliqué seulement lorsqu'un <i>Client</i> se connecte au moyen de l'un de ces Points d'extrémité.
Sécurité	Sécurité Rôle <i>Serveur</i> DefaultRolePermissions	Permet aux utilisateurs autorisés de régler la <i>Propriété</i> DefaultRolePermissions pour certains NameSpaces. Les DefaultRolePermissions sont appliquées si aucune RolePermission n'est associée à un <i>Nœud</i> .
Sécurité	Sécurité Rôle <i>Serveur</i> RolePermissions	Permet aux utilisateurs autorisés de régler l' <i>Attribut</i> RolePermissions sur <i>Nœuds</i> .
Sécurité	Sécurité – Autorisation de Rôles <i>Serveur</i>	Restreindre l'accès selon les Rôles et permissions configurés.
Sécurité	Sécurité Rôle <i>Client</i> de base	Comprend et utilise le Modèle d'information sur l'Autorisation de l'utilisateur défini dans l'IEC 62541-5 et l' <i>Attribut</i> RolePermissions.
Sécurité	Sécurité Rôle <i>Client</i> Gestion	Prend en charge la création de nouveaux Rôles et l'ajout d'identités, ainsi que la suppression de Rôles ou d'Identités selon les <i>Méthodes</i> adéquates.
Sécurité	Sécurité Rôle <i>Client</i> Restreindre Applications	Utilise les <i>Méthodes</i> adéquates pour ajouter des applications à un Rôle, afin que seules ces applications puissent utiliser ce Rôle.
Sécurité	Sécurité Rôle <i>Client</i> Restreindre Points d'extrémité	Utilise les <i>Méthodes</i> adéquates pour ajouter des Points d'extrémité à un Rôle. Avec cette restriction, un Rôle est appliqué seulement lorsqu'un <i>Client</i> se connecte au moyen de l'un de ces Points d'extrémité.
Sécurité	Sécurité Rôle <i>Client</i> DefaultRolePermissions	Capacité de régler la <i>Propriété</i> DefaultRolePermissions pour certains NameSpaces. Les DefaultRolePermissions sont appliquées si aucune RolePermission n'est associée à un <i>Nœud</i> .
Sécurité	Sécurité Rôle <i>Client</i> RolePermissions	Prend en charge le réglage de l' <i>Attribut</i> RolePermissions sur des <i>Nœuds</i> .
Sécurité	Modèle Pull pour la Gestion Globale des <i>Certificats</i> et des TrustLists	Utilise les services de gestion de <i>Certificats</i> de l'IEC 62541-12 pour que le modèle Pull gère les <i>Certificats</i> d'instance d'application et les Listes de confiance, y compris des Listes de révocation.

Catégorie	Intitulé	Description
Sécurité	Modèle Push pour la Gestion Globale des <i>Certificats</i> et des TrustLists	Prend en charge les services de gestion de <i>Certificats</i> de l'IEC 62541-12 pour que le modèle Push gère les <i>Certificats</i> d'instance d'application et les Listes de confiance, y compris des Listes de révocation.
Sécurité	Modèle Pull pour le Service KeyCredential	Utilise les <i>Méthodes</i> sur une instance du KeyCredentialServiceType (modèle Pull) pour obtenir les KeyCredentials spécifiés dans l'IEC 62541-12.
Sécurité	Modèle Push pour le Service KeyCredential	Prend en charge le Modèle <i>Push</i> des Services KeyCredential de l'IEC 62541-12 pour obtenir les KeyCredentials. Cela comprend la prise en charge d'une ou plusieurs instances du KeyCredentialConfigurationType et des <i>Méthodes</i> pour mettre à jour ou supprimer les authentifiants.
Sécurité	Configuration du Service d'autorisation Serveur	Prend en charge les Types d'objets définis dans l'IEC 62541-12 afin de permettre la configuration des informations nécessaires pour accepter les Jetons d'accès lorsqu'ils sont présentés par le Client pendant l'établissement de la session. Les Jetons d'accès sont émis par les Services d'autorisation.
Sécurité	Service d'Autorisation Client	Utilise la méthode RequestAccessToken définie dans l'IEC 62541-12.
Sécurité	SymmetricSignatureAlgorithm_None	Cet algorithme ne s'applique pas.
Sécurité	SymmetricSignatureAlgorithm_HMAC-SHA1	Hachage avec clé défini dans https://tools.ietf.org/html/rfc2104 . L'algorithme de hachage est SHA1 et il est décrit dans https://tools.ietf.org/html/rfc3174 . L'URI est http://www.w3.org/2000/09/xmldsig#hmac-sha1 Il n'existe aucune exploitation connue utilisant SHA1 avec un hachage à clé. Toutefois, SHA1 a été cassé en 2017 et l'utilisation de cet algorithme n'est donc pas recommandée.
Sécurité	SymmetricSignatureAlgorithm_HMAC-SHA2-256	Hachage avec clé utilisé pour l'authentification des messages, défini dans https://tools.ietf.org/html/rfc2104 . L'algorithme de hachage est SHA2 avec 256 bits et il est décrit dans https://tools.ietf.org/html/rfc4634 .
Sécurité	SymmetricEncryptionAlgorithm_None	Cet algorithme ne s'applique pas.
Sécurité	SymmetricEncryptionAlgorithm_AES128-CBC	L'algorithme de chiffrement AES qui est défini dans http://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.197.pdf . Plusieurs blocs chiffrés au moyen du mode CBC décrit dans http://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-38a.pdf . La taille de la clé est de 128 bits. La taille du bloc est de 16 octets. L'URI est http://www.w3.org/2001/04/xmlenc#aes128-cbc
Sécurité	SymmetricEncryptionAlgorithm_AES256-CBC	L'algorithme de chiffrement AES qui est défini dans http://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.197.pdf . Plusieurs blocs chiffrés au moyen du mode CBC décrit dans http://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-38a.pdf . La taille de la clé est de 256 bits. La taille du bloc est de 16 octets. L'URI est http://www.w3.org/2001/04/xmlenc#aes256-cbc
Sécurité	SymmetricEncryptionAlgorithm_AES128-CTR	L'algorithme de chiffrement AES qui est défini dans http://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.197.pdf . Plusieurs blocs chiffrés au moyen du mode CTR décrit dans http://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-38a.pdf . Le format du bloc compteur est défini dans https://tools.ietf.org/html/rfc3686 . La taille de la clé est de 128 bits. La taille du bloc est de 16 octets. La longueur du nonce d'entrée est de 4 octets. L'URI est http://opcfoundation.org/UA/security/aes128-ctr

Catégorie	Intitulé	Description
Sécurité	SymmetricEncryptionAlgorithm_AES256-CTR	L'algorithme de chiffrement AES qui est défini dans http://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.197.pdf. La taille de la clé est de 256 bits. La taille du bloc est de 16 octets. Plusieurs blocs chiffrés au moyen du mode CTR décrit dans http://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-38a.pdf. Le format du bloc compteur est défini dans https://tools.ietf.org/html/rfc3686. La taille de la clé est de 128 bits. La taille du bloc est de 16 octets. La longueur du nonce d'entrée est de 4 octets. L'URI est http://opcfoundation.org/UA/security/aes256-ctr.
Sécurité	AsymmetricSignatureAlgorithm_None	Cet algorithme ne s'applique pas.
Sécurité	AsymmetricSignatureAlgorithm_RSA- PKCS15-SHA1	L'algorithme de signature RSA qui est défini dans https://tools.ietf.org/html/rfc3447. Le schéma RSASSA-PKCS1-v1_5 est utilisé. L'algorithme de hachage est SHA1 et il est décrit dans https://tools.ietf.org/html/rfc3174. L'URI est http://www.w3.org/2000/09/xmldsig#rsa-sha1. SHA1 a été cassé en 2017 et il convient donc de ne pas utiliser cet algorithme.
Sécurité	AsymmetricSignatureAlgorithm_RSA-PKCS15-SHA2-256	L'algorithme de signature RSA qui est défini dans https://tools.ietf.org/html/rfc3447. Le schéma RSASSA-PKCS1-v1_5 est utilisé. L'algorithme de hachage est SHA2 avec 256 bits et il est décrit dans https://tools.ietf.org/html/rfc6234. L'URI est http://www.w3.org/2001/04/xmldsig-more#rsa-sha256.
Sécurité	AsymmetricSignatureAlgorithm_RSA-PSS -SHA2-256	L'algorithme de signature RSA qui est défini dans https://tools.ietf.org/html/rfc3447. Le schéma RSASSA-PSS est utilisé. L'algorithme de hachage est SHA2 avec 256 bits et il est décrit dans https://tools.ietf.org/html/rfc6234. L'algorithme de génération du masque utilise aussi SHA2 avec 256 bits. La longueur du grain de sel est 32 octets. L'URI est http://opcfoundation.org/UA/security/rsa-pss-sha2-256.
Sécurité	AsymmetricEncryptionAlgorithm_None	Cet algorithme ne s'applique pas.
Sécurité	AsymmetricEncryptionAlgorithm_RSA-PKCS15	L'algorithme de chiffrement RSA qui est défini dans https://tools.ietf.org/html/rfc3447. Le schéma RSAES-PKCS1-v1_5 est utilisé. L'URI est http://www.w3.org/2001/04/xmlenc#rsa-1_5. Le schéma RSAES-PKCS1-v1_5 a des faiblesses connues et il n'est pas recommandé.
Sécurité	AsymmetricEncryptionAlgorithm_RSA-OAEP-SHA1	L'algorithme de chiffrement RSA qui est défini dans https://tools.ietf.org/html/rfc3447. Le schéma RSAES-OAEP est utilisé. L'algorithme de hachage est SHA1 et il est décrit dans https://tools.ietf.org/html/rfc6234. L'algorithme de génération du masque utilise également SHA1. L'URI est http://www.w3.org/2001/04/xmlenc#rsa-oaep. Il n'existe aucune exploitation connue utilisant SHA1 avec RSAES-OAEP. Toutefois, SHA1 a été cassé en 2017 et l'utilisation de cet algorithme n'est donc pas recommandée.

Catégorie	Intitulé	Description
Sécurité	AsymmetricEncryptionAlgorithm_RSA-OAEP-SHA2-256	L'algorithme de chiffrement RSA qui est défini dans https://tools.ietf.org/html/rfc3447 . Le schéma RSAES-OAEP est utilisé. L'algorithme de hachage est SHA2 avec 256 bits et il est décrit dans https://tools.ietf.org/html/rfc6234 . L'algorithme de génération du masque utilise aussi SHA2 avec 256 bits. L'URI est http://opcfoundation.org/UA/security/rsa-oaep-sha2-256 .
Sécurité	KeyDerivationAlgorithm_None	Cet algorithme ne s'applique pas.
Sécurité	KeyDerivationAlgorithm_P-SHA1	La fonction pseudoaléatoire P_SHA-1 est définie dans https://tools.ietf.org/html/rfc4346 . L'URI est http://docs.oasis-open.org/ws-sx/ws-secureconversation/200512/dk/p_sha1 . Il n'existe aucune exploitation connue utilisant SHA1 avec P-SHA-1. Toutefois, SHA1 a été cassé en 2017 et l'utilisation de cet algorithme n'est donc pas recommandée.
Sécurité	KeyDerivationAlgorithm_P-SHA2-256	La fonction pseudoaléatoire P_SHA256 est définie dans https://tools.ietf.org/html/rfc5246 . L'URI est http://docs.oasis-open.org/ws-sx/ws-secureconversation/200512/dk/p_sha256 .
Sécurité	CertificateSignatureAlgorithm_None	Cet algorithme ne s'applique pas.
Sécurité	CertificateSignatureAlgorithm_RSA-PKCS15-SHA2-256	L'algorithme de signature RSA qui est défini dans https://tools.ietf.org/html/rfc3447 . Le schéma RSASSA-PKCS1-v1_5 est utilisé. L'algorithme de hachage est SHA2 avec 256 bits et il est décrit dans https://tools.ietf.org/html/rfc6234 . L'algorithme SHA2 avec 384 ou 512 bits peut être utilisé à la place de SHA2 avec 256 bits. L'URI est http://www.w3.org/2001/04/xmldsig-more#rsa-sha256 .
Sécurité	CertificateSignatureAlgorithm_RSA-PKCS15-SHA1	L'algorithme de signature RSA qui est défini dans https://tools.ietf.org/html/rfc3447 . Le schéma RSASSA-PKCS1-v1_5 est utilisé. L'algorithme de hachage est SHA1 et il est décrit dans https://tools.ietf.org/html/rfc3174 . L'URI est http://www.w3.org/2000/09/xmldsig#rsa-sha1 . SHA1 a été cassé en 2017 et il convient donc de ne pas utiliser cet algorithme. L'algorithme SHA2 avec 244, 256, 384 ou 512 bits peut être utilisé à la place de SHA1. L'algorithme SHA2 est décrit dans https://tools.ietf.org/html/rfc6234 .
Sécurité	SecurityPolicy_None_Limits	DerivedSignatureKeyLength: 0
Sécurité	Aes128-Sha256-RsaOaep_Limits	-> DerivedSignatureKeyLength: 256 bits -> MinAsymmetricKeyLength: 2048 bits -> MaxAsymmetricKeyLength: 4096 bits -> SecureChannelNonceLength: 32 octets
Sécurité	Basic256Sha256_Limits	-> DerivedSignatureKeyLength: 256 bits -> MinAsymmetricKeyLength: 2048 bits -> MaxAsymmetricKeyLength: 4096 bits -> SecureChannelNonceLength: 32 octets

Catégorie	Intitulé	Description
Sécurité	Aes256-Sha256-RsaPss_Limits	-> DerivedSignatureKeyLength: 256 bits -> MinAsymmetricKeyLength: 2048 bits -> MaxAsymmetricKeyLength: 4096 bits -> SecureChannelNonceLength: 32 octets
Sécurité	Basic128Rsa15_Limits	-> DerivedSignatureKeyLength: 128 bits -> MinAsymmetricKeyLength: 1024 bits -> MaxAsymmetricKeyLength: 2048 bits -> SecureChannelNonceLength: 16 octets
Sécurité	Basic256_Limits	-> DerivedSignatureKeyLength: 192 bits -> MinAsymmetricKeyLength: 1024 bits -> MaxAsymmetricKeyLength: 2048 bits -> SecureChannelNonceLength: 32 octets

Le Tableau 12 décrit les caractéristiques liées au protocole et au chiffrement qui peuvent être profilées. Ces caractéristiques sont définies en détail dans l'IEC 62541-6. Il est recommandé que les *Serveurs* et les *Clients* prennent en charge le plus grand nombre possible de ces options pour une plus grande interopérabilité.

Tableau 12 – Protocole et codage

Catégorie	Intitulé	Description
<i>Serveur</i>	Protocole de Connexion inversée <i>Serveur</i>	Prend en charge la connectivité inversée en envoyant un message ReverseHello à un Client. La procédure de connexion inversée peut être appliquée à plusieurs transports, spécifiés dans l'IEC 62541-6, et doit être prise en charge pour tous ces transports disponibles dans un Serveur.
<i>Serveur</i>	Protocole Configuration	Permet l'administration des Points d'extrémité et du nombre d'accès utilisé par ces derniers.
<i>Client</i>	Protocole de Connexion inversée <i>Client</i>	Prend en charge la connectivité inversée en acceptant les messages ReverseHello des Serveurs et établit un Canal sécurisé si l'URI du Serveur est acceptée par le Client ou par l'utilisateur du Client. La procédure de connexion inversée peut être appliquée à plusieurs transports, spécifiés dans l'IEC 62541-6, et doit être prise en charge pour tous ces transports pris en charge par le Client.
Transport	Protocole UA TCP	Prend en charge le protocole de transport UA TCP défini dans l'IEC 62541-6.
Transport	Protocole HTTPS	Prend en charge le protocole HTTPS défini dans l'IEC 62541-6.
Transport	Protocole Web Sockets	Prend en charge le protocole WSS (WebSocket) avec au moins l'un des sous-protocoles définis dans l'IEC 62541-6.
Transport	Conversation sécurisée UA	Prend en charge la conversation sécurisée UA spécifiée dans l'IEC 62541-6.
Transport	Codage UA binaire	Prend en charge le codage UA binaire. Les valeurs de ces types de données sont codées dans des formats binaires compacts, de manière contiguë et sans balises. Autrement dit, il est admis par hypothèse que le récepteur comprend la structure qu'il décode.
Transport	Codage UA SOAP-XML	Prend en charge le codage XML SOAP V1.2 défini dans l'IEC 62541-6. Les éléments XML incluent toutes les informations nécessaires pour les reconvertir en structures OPC UA dans tous les langages.
Transport	Codage réversible JSON	Prend en charge le codage JSON réversible défini dans l'IEC 62541-6. L'élément JSON inclut toutes les informations nécessaires pour le reconvertir en structures OPC UA dans tous les langages.

5.4 Modèle d'information et caractéristiques relatives à l'AddressSpace

Le Tableau 13 décrit les éléments relatifs aux caractéristiques de base qui peuvent être intégrés dans les profils. Pour plus d'informations sur ces éléments, se reporter à l'IEC 62541-3, l'IEC 62541-5 et l'IEC 62541-6. Les *Serveurs* avec une plus grande capacité de ressources prennent en charge la plus grande part de cette fonctionnalité, un *Serveur* avec une capacité de ressources moindre ne peut prendre en charge qu'une partie de cette même fonctionnalité. De nombreux *Clients* utilisent une partie de cette fonctionnalité et des *Clients* plus robustes utilisent en revanche la plus grande partie de cette fonctionnalité.

Tableau 13 – Informations de base

Catégorie	Intitulé	Description
<i>Serveur</i>	Informations de base – Structure principale	Le <i>Serveur</i> prend en charge l' <i>Objet Serveur</i> , les <i>ServerCapabilities</i> et la structure de l' <i>AddressSpace</i> OPC UA.
<i>Serveur</i>	Informations de base – Capacités <i>Serveur</i>	Le <i>Serveur</i> prend en charge l'édition de la limite du <i>Serveur</i> dans les <i>ServerCapabilities</i> , y compris <i>MaxArrayLength</i> , <i>MaxStringLength</i> , <i>MaxNodePerRead</i> , <i>MaxNodesPerWrite</i> , <i>MaxNodesPerSubscription</i> et <i>MaxNodesPerBrowse</i> .
<i>Serveur</i>	Informations de base – Événements de progression	Le <i>Serveur</i> présente si la génération d'événements de progression est prise en charge pour des appels de services de longue durée, tels que <i>HistoryRead</i> ou <i>Query</i> . Si elle est indiquée comme prise en charge dans les <i>ServerCapabilities</i> , alors les événements réels sont vérifiés.
<i>Serveur</i>	Informations de base – Diagnostics	Le <i>Serveur</i> prend en charge la collecte d'informations de diagnostic. Le fanion <i>EnabledFlag</i> de l' <i>Objet ServerDiagnostics</i> peut être réglé sur <i>TRUE</i> . Dans ce cas, tous les <i>Objets</i> et <i>Variables</i> statiques et dynamiques des données de diagnostic définies dans l'IEC 62541-5 sont pris en charge.
<i>Serveur</i>	Informations de base – Statut Système	Le <i>Serveur</i> prend en charge la génération de <i>SystemStatusChangeEvent</i> indiquant l'arrêt du <i>Serveur</i> (<i>SourceNode=Serveur</i>).
<i>Serveur</i>	Informations de base – Temps de retour estimé	Le <i>Serveur</i> prend en charge la <i>Propriété</i> <i>EstimatedReturnTime</i> . Elle indique le moment où il est prévu que le <i>ServerStatus.State</i> du <i>Serveur</i> soit <i>RUNNING_0</i> . Les <i>Clients</i> peuvent utiliser cette information pour régir la logique de reconnexion.
<i>Serveur</i>	Informations de base – Statut Système – Système sous-jacent	Le <i>Serveur</i> prend en charge la génération de <i>SystemStatusChangeEvent</i> indiquant les modifications d'un Système sous-jacent (<i>SourceNode = Serveur</i>). Cet événement peut aussi être utilisé pour indiquer que le <i>Serveur</i> OPC UA a des systèmes sous-jacents.
<i>Serveur</i>	Informations de base – Défaillance d'appareil	Le <i>Serveur</i> prend en charge la génération des <i>DeviceFailureEvent</i> indiquant les modifications apportées à chaque appareil dans un système sous-jacent.
<i>Serveur</i>	Informations de base – Méthode <i>GetMonitoredItems</i>	Le <i>Serveur</i> prend en charge l'obtention d'informations relatives aux abonnements via la <i>Méthode</i> <i>GetMonitoredItems</i> sur l'objet <i>Serveur</i> .
<i>Serveur</i>	Informations de base – Méthode <i>ResendData</i>	Prend en charge la <i>Méthode</i> normalisée <i>ResendData</i> (définie dans l'IEC 62541-5) pour obtenir la dernière valeur des éléments surveillés d'un <i>Abonnement</i> .
<i>Serveur</i>	Informations de base – Système type	Le <i>Serveur</i> présente un Système Type avec des <i>DataTypes</i> , des <i>ReferenceTypes</i> , des <i>ObjectTypes</i> et des <i>VariableTypes</i> , y compris tous les types OPC UA (namespace 0) que le <i>Serveur</i> utilise, selon la définition de l'IEC 62541-5. Les éléments définis dans Namespace 0, mais définis dans d'autres parties de la spécification sont soumis à l'essai dans le cadre des autres modèles d'information.
<i>Serveur</i>	Informations de base – Système type personnalisé	Le <i>Serveur</i> prend en charge les types personnalisés (c'est-à-dire les types dérivés des <i>ObjectTypes</i> , <i>VariableTypes</i> , <i>ReferenceTypes</i> ou <i>DataTypes</i> connus). La prise en charge de cette unité de conformité exige que les types personnalisés et leur arbre d'héritage complet soient présentés dans l' <i>AddressSpace</i> .

Catégorie	Intitulé	Description
<i>Serveur</i>	Informations de base – Modification de modèle	Le <i>Serveur</i> prend en charge l' <i>Événement</i> ModelChange et la <i>Propriété</i> NodeVersion pour tous les <i>Nœuds</i> pour lesquels le serveur permet des modifications du Modèle.
<i>Serveur</i>	Informations de base – Règles de modélisation – Paramètre fictif	Le <i>Serveur</i> prend en charge la définition d' <i>Objet</i> personnalisé ou de <i>Variables</i> qui comprennent l'utilisation des règles de modélisation OptionalPlaceholder ou MandatoryPlaceholder.
<i>Serveur</i>	Informations de base – SemanticChange	Le <i>Serveur</i> prend en charge SemanticChangeEvent pour certaines <i>Propriétés</i> . Cela comprend le réglage du bit SemanticChange dans le statut lorsqu'un changement sémantique se produit, comme une modification de l'unité technique associée à une valeur.
<i>Serveur</i>	Informations de base – EventType EventQueueOverflow	Le <i>Serveur</i> prend en charge l'EventQueueOverflowEventType défini dans l'IEC 62541-4.
<i>Serveur</i>	Informations de base – OptionSet	Le <i>Serveur</i> prend en charge le <i>VariableType</i> OptionSet.
<i>Serveur</i>	Informations de base – ValueAsText	Le <i>Serveur</i> prend en charge la <i>Propriété</i> ValueAsText pour les DataTypes répertoriés.
<i>Serveur</i>	Informations de base – Unités techniques	Le <i>Serveur</i> prend en charge la définition des <i>Variables</i> comprenant la <i>Propriété</i> Unités techniques. Cette propriété utilise la structure de données EUInformation. Cette structure par défaut représente les "Codes for Units of Measurement" de l'UN/CEFACT. Si une autre représentation EU est exigée, l'EUInformation.namespaceUri indique l'espace de noms alternatif.
<i>Serveur</i>	Informations de base – Liste de sélection	Le <i>Serveur</i> prend en charge les Variables du VariableType SelectListType.
<i>Serveur</i>	Informations de base – FileType Base	Le <i>Serveur</i> prend en charge l' <i>Objet</i> FileType (voir l'IEC 62541-5). L'écriture de fichier peut être limitée.
<i>Serveur</i>	Informations de base – FileType Ecriture	Le <i>Serveur</i> prend en charge l' <i>Objet</i> FileType, y compris l'écriture des fichiers. La prise en charge du contrôle d'accès des utilisateurs dans l' <i>Objet</i> FileType est également incluse.
<i>Serveur</i>	Informations de base – Méthode RequestServerStateChange	Le <i>Serveur</i> prend en charge la <i>Méthode</i> RequestServerStateChange.
<i>Serveur</i>	Informations de base – Instance de diagramme d'états	Prend en charge les instances de StateMachineType ou un sous-type dans l'AddressSpace. Génère des Événements lorsque des changements d'état significatifs ont lieu. Il existe au moins une <i>Référence</i> GeneratesEvent pour définir un ou plusieurs <i>Événement(s)</i> déclenché(s) aux changements d'état.
<i>Serveur</i>	Informations de base – Instance de diagramme d'états finis	Prend en charge les instances de FiniteStateMachineType ou un sous-type dans l'AddressSpace.
<i>Serveur</i>	Informations de base – Etats et transitions disponibles	Prend en charge les <i>Propriétés</i> AvailableStates et AvailableTransitions pour le FiniteStateMachineType.
<i>Client</i>	Informations de base <i>Client</i> – Base	Le <i>Client</i> utilise l'AddressSpace OPC UA défini. Accède ou permet l'accès aux informations du <i>Serveur</i> telles que l'état du <i>Serveur</i> , BuildInfo, les capacités, le tableau d'espace de noms et le modèle de type.
<i>Client</i>	Informations de base <i>Client</i> – Respecter les limites de fonctionnement	Le <i>Client</i> doit respecter les limites du <i>Serveur</i> décrites dans l' <i>Objet</i> ServerCapabilities du <i>Serveur</i> .
<i>Client</i>	Informations de base <i>Événements</i> – Traitement	Le <i>Client</i> est capable de s'abonner aux Événements OPC UA de base et de les traiter.
<i>Client</i>	Informations de base <i>Client</i> – Statut de système	Le <i>Client</i> utilise SystemStatusChangeEventType pour détecter un arrêt du <i>Serveur</i> .
<i>Client</i>	Informations de base <i>Client</i> – Temps de retour estimé	Le <i>Client</i> utilise la <i>Propriété</i> EstimatedReturnTime pour régir la logique de reconnexion.
<i>Client</i>	Informations de base <i>Client</i> – Statut système – Système sous-jacent	Le <i>Client</i> utilise le SystemStatusChangeEventType pour détecter les modifications d'un Système Sous-jacent (SourceNode = <i>Serveur</i>).

Catégorie	Intitulé	Description
<i>Client</i>	Informations de base <i>Client</i> – Défaillance d'appareil	Le <i>Client</i> utilise le <i>DeviceFailureEventType</i> pour détecter les appareils défaillants dans des systèmes sous-jacents.
<i>Client</i>	Informations de base <i>Client</i> – Événements de progression	Le <i>Client</i> utilise <i>ProgressEvents</i> , y compris la vérification de leur prise en charge.
<i>Client</i>	Informations de base <i>Client</i> – Diagnostics	Le <i>Client</i> permet un accès interactif ou par programmation aux informations de diagnostic du <i>Serveur</i> .
<i>Client</i>	Informations de base <i>Client</i> – Programmation type	Le <i>Client</i> traite par programmation les instances d' <i>Objets</i> ou de Variables en se servant de leurs définitions de type. Ceci inclut les <i>DataTypes</i> , les <i>ObjectTypes</i> et les <i>VariableTypes</i> .
<i>Client</i>	Informations de base <i>Client</i> – Connaissances préalables types	Le <i>Client</i> doit interagir avec les <i>Serveurs</i> qui ne présentent pas les types d'OPC UA dans l' <i>AddressSpace</i> .
<i>Client</i>	Informations de base <i>Client</i> – Nœuds distants	Le <i>Client</i> peut accéder aux Nœuds ayant un <i>NodeId</i> étendu qui référence un <i>Serveur</i> différent du <i>Serveur</i> d'origine. Il est acceptable que les informations de configuration du <i>Serveur</i> soient préconfigurées sur le <i>Client</i> .
<i>Client</i>	Informations de base <i>Client</i> – Événements de modification	Le <i>Client</i> traite les <i>ModelChangeEvents</i> afin de détecter les modifications de l' <i>AddressSpace</i> OPC UA du <i>Serveur</i> et de prendre les mesures appropriées pour une modification donnée.
<i>Client</i>	Informations de base <i>Client</i> – Méthode <i>GetMonitoredItems</i>	Le <i>Client</i> utilise la Méthode <i>GetMonitoredItems</i> pour reprendre après des interruptions de communication et/ou pour récupérer des informations relatives aux abonnements.
<i>Client</i>	Données de base <i>Client</i> – Méthode <i>ResendData</i>	Le <i>Client</i> utilise la Méthode <i>ResendData</i> pour rechercher la dernière valeur des éléments surveillés par les données.
<i>Client</i>	Informations de base <i>Client</i> – Liste de sélection	Le <i>Client</i> utilise et comprend les Variables du <i>VariableType</i> <i>SelectionListType</i> .
<i>Client</i>	Informations de base <i>Client</i> – <i>FileType</i> Base	Le <i>Client</i> peut accéder à un <i>Objet</i> <i>FileType</i> pour transférer un fichier du <i>Serveur</i> au <i>Client</i> . Cela comprend des fichiers de grande taille.
<i>Client</i>	Informations de base <i>Client</i> – <i>FileType</i> Ecriture	Le <i>Client</i> peut accéder à un <i>Objet</i> <i>FileType</i> pour transférer un fichier du <i>Client</i> au <i>Serveur</i> . Cela comprend des fichiers de grande taille.
<i>Client</i>	Informations de base <i>Client</i> – <i>RequestServerStateChange</i>	Le <i>Client</i> peut invoquer la Méthode <i>RequestServerStateChange</i> .
<i>Client</i>	Informations de base <i>Client</i> – Instance de diagramme d'états	Utilise les instances de <i>StateMachineType</i> ou un sous-type. Surveille la <i>Variable</i> du composant <i>CurrentState</i> de l'instance ou les Événements déclenchés sous l'effet des changements d'état. Utilise les Méthodes, si elles sont définies, pour que le <i>StateMachineType</i> affecte l'état.
<i>Client</i>	Informations de base <i>Client</i> – Instance de diagramme d'états finis	Utilise les instances de <i>FinitStateMachineType</i> ou d'un sous-type. Surveille la <i>Variable</i> du composant <i>CurrentState</i> de l'instance ou les Événements déclenchés sous l'effet des changements d'état.
<i>Client</i>	Informations de base <i>Client</i> – Etats et transitions disponibles	Utilise les Propriétés <i>AvailableStates</i> et <i>AvailableTransitions</i> lorsqu'elles sont présentées par un <i>Serveur</i> .

Le Tableau 14 décrit les éléments relatifs aux informations concernant le modèle d'espace d'adressage qui peuvent être intégrés dans les profils. Les détails de ces éléments de modèle sont définis dans l'IEC 62541-3 et l'IEC 62541-5. Cela inclut les *Facettes serveur* qui décrivent les éléments présentés par un *Serveur* et les *Facettes client* qui décrivent les éléments consommés par un *Client*.

Tableau 14 – Modèle d'espace d'adressage

Catégorie	Intitulé	Description
<i>Serveur</i>	Espace d'adressage – Base	Prend en charge les <i>NodeClasses</i> avec leurs <i>Attributs</i> et leurs <i>Références</i> , définis dans l'IEC 62541-3. Ceci inclut, par exemple, les attributs suivants: <i>Objet</i> , <i>ObjectType</i> , <i>Variable</i> , <i>VariableType</i> , <i>Références</i> et <i>DataType</i> .
<i>Serveur</i>	Espace d'adressage – Atomicité	Prend en charge le réglage des fanions NonatomicRead et NonatomicWrite dans l' <i>Attribut</i> AccessLevelEx pour que les <i>Nœuds de Variables</i> indiquent si les opérations de lecture ou d'écriture peuvent être réalisées de manière atomique. Si les fanions sont réglés sur '1', l'atomicité ne peut pas être assurée.
<i>Serveur</i>	Espace d'adressage – Matrice complète seulement	Prend en charge le réglage du fanion WriteFullArrayOnly dans l' <i>Attribut</i> AccessLevelEx pour que les <i>Nœuds de Variables</i> des types de données non scalaires indiquent si les opérations d'écriture peuvent être réalisées avec une plage d'indices pour une matrice.
<i>Serveur</i>	Espace d'adressage – Événements	Prend en charge les éléments de l' <i>AddressSpace</i> OPC UA afin de générer les notifications d' <i>Événements</i> . Ceci inclut au moins un <i>Nœud</i> avec un <i>Attribut</i> EventNotifier mis à True (<i>Nœud Serveur</i>).
<i>Serveur</i>	Espace d'adressage – Dictionnaire de données complexes	Prend en charge les <i>DataTypes</i> structurés avec un Dictionnaire de Données. Noter que la V1.04 de l'IEC 62541-3 spécifie une approche simplifiée utilisant le nouvel <i>Attribut</i> DataTypeDefinition. L'unité de conformité "Address Space DataTypeDefinition Attribute" exige la prise en charge de l' <i>Attribut</i> DataTypeDefinition. La prise en charge d'un DataDictionary est déconseillée dans l'une des versions ultérieures de l'OPC UA.
<i>Serveur</i>	Espace d'adressage – Attribut DataTypeDefinition	Prend en charge les <i>DataTypes</i> structurés et présente les métadonnées et les informations sur le codage avec un StructureDefinitionType au moyen de l' <i>Attribut</i> DataTypeDefinition.
<i>Serveur</i>	Espace d'adressage – Méthode	Prend en charge les <i>Nœuds</i> de <i>Méthode</i> .
<i>Serveur</i>	Espace d'adressage – Hiérarchie de Notification	Prend en charge l'utilisation de la référence HasNotifier afin de construire une hiérarchie des <i>Nœuds d'Objet</i> qui sont des notifications avec d'autres <i>Nœuds d'Objet</i> de notification.
<i>Serveur</i>	Espace d'adressage – Hiérarchie de Source	Prend en charge des hiérarchies de sources d'événements dont chacune prend racine dans un <i>Nœud d'Objet</i> qui est une notification. La <i>Référence</i> HasEventSource est utilisée pour relier les <i>Nœuds</i> dans une hiérarchie. Si des <i>Conditions</i> sont prises en charge, la hiérarchie doit comprendre les <i>Références</i> HasCondition.
<i>Serveur</i>	Espace d'adressage – WriteMask	Prend en charge le WriteMask qui indique la disponibilité de l'accès en écriture pour tous les attributs, y compris les attributs non pris en charge.
<i>Serveur</i>	Espace d'adressage – UserWriteMask	Prend en charge le UserWriteMask qui indique la disponibilité de l'accès en écriture pour tous les attributs d'un utilisateur donné, y compris les attributs non pris en charge. La prise en charge comprend au moins deux niveaux d'utilisateurs.
<i>Serveur</i>	Espace d'adressage – UserWriteMask Multiniveaux	Prend en charge le UserWriteMask qui indique la disponibilité de l'accès en écriture pour tous les attributs d'un utilisateur donné, y compris les attributs non pris en charge. La prise en charge comprend plusieurs niveaux de contrôle de l'accès pour tous les nœuds du système.
<i>Serveur</i>	Espace d'adressage – Niveau d'accès utilisateur complet	Met en œuvre la sécurité du niveau d'accès de l'utilisateur; cela comprend la prise en charge de plusieurs niveaux de contrôle de l'accès pour les nœuds de <i>Variables</i> dans le système. Cela comprend une indication d'accès à l' <i>Attribut</i> Valeur en lecture, écriture, lecture Historique et écriture Historique.
<i>Serveur</i>	Espace d'adressage – Niveau d'accès utilisateur de base	Met en œuvre la sécurité du niveau d'accès de l'utilisateur pour les nœuds de <i>Variables</i> ; cela comprend au moins deux utilisateurs dans le système. Cela comprend une indication d'accès à l' <i>Attribut</i> Valeur en lecture, écriture, lecture Historique et écriture Historique.
<i>Client</i>	Espace d'adressage <i>Client</i> – Base	Utilise et comprend les <i>NodeClasses</i> avec leurs <i>Attributs</i> et leur comportement, définis dans l'IEC 62541-3. Ceci inclut, par exemple, les attributs suivants: <i>Objet</i> , <i>ObjectType</i> , <i>Variable</i> , <i>VariableType</i> , <i>Références</i> et <i>DataType</i> . Ceci inclut de traiter les BrowseNames et NodeIds de Chaîne comme sensibles à la casse.

Catégorie	Intitulé	Description
<i>Client</i>	Espace d'adressage <i>Client</i> – Atomicité	Accède aux fanions NonatomicRead ou NonatomicWrite dans l' <i>Attribut</i> AccessLevelEx des <i>Nœuds de Variables</i> pour définir si les opérations de lecture ou d'écriture peuvent être réalisées de manière atomique. Ces informations s'affichent normalement à l'attention de l'utilisateur aux fins d'actions ultérieures.
<i>Client</i>	Espace d'adressage <i>Client</i> – Matrice complète seulement	Accède au fanion WriteFullArrayOnly dans l' <i>Attribut</i> AccessLevelEx des <i>Nœuds de Variables</i> ayant des types de données non scalaires pour déterminer si les opérations d'écriture avec une IndexRange sont autorisées dans une matrice.
<i>Client</i>	Espace d'adressage <i>Client</i> – Dictionnaire de données complexes	Utilise et comprend les DataTypes structurés arbitraires au moyen du Dictionnaire de Données. Noter que la V1.04 de l'IEC 62541-3 spécifie une approche simplifiée utilisant le nouvel Attribut DataTypeDefinition. L'unité de conformité "Address Space Client DataTypeDefinition Attribute" exige la prise en charge de l'Attribut DataTypeDefinition.
<i>Client</i>	Espace d'adressage <i>Client</i> – Attribut DataTypeDefinition	Utilise et comprend les DataTypes structurés arbitraires où les métadonnées et les informations sur le codage sont présentées avec un StructureDefinitionType au moyen de l' <i>Attribut</i> DataTypeDefinition.
<i>Client</i>	Espace d'adressage <i>Client</i> – Hiérarchie de Notification	Utilise la hiérarchie des <i>Nœuds d'Objet</i> qui sont des notifications afin de détecter des zones spécifiques où le <i>Client</i> peut s'abonner à des Événements.
<i>Client</i>	Espace d'adressage <i>Client</i> – Hiérarchie de Source	Détecte et utilise la hiérarchie des sources d'événements présentée pour des <i>Nœuds d'Objet</i> spécifiques qui sont des notifications d'événements.

Le Tableau 15 décrit les éléments relatifs au modèle d'information d'accès aux données qui peuvent être intégrés dans les profils. Les détails de ce modèle sont définis dans l'IEC 62541-8. Les *Serveurs* peuvent présenter ce modèle d'information et les *Clients* peuvent utiliser ce modèle d'information.

Tableau 15 – Accès aux données

Catégorie	Intitulé	Description
<i>Serveur</i>	Accès aux données – DataItems	Fournit des <i>Variables</i> du DataItem type ou un de ses sous-types. Prend en charge les StatusCodes spécifiés dans l'IEC 62541-8. La prise en charge des propriétés facultatives ("InstrumentRange", par exemple) doit être vérifiée lors des essais de certification et est indiquée sur le <i>Certificat</i> .
<i>Serveur</i>	Accès aux données – AnalogItems	Prend en charge les <i>Variables</i> AnalogItem type avec les Propriétés correspondantes. La prise en charge des propriétés facultatives est indiquée.
<i>Serveur</i>	Accès aux données – PercentDeadband	Prend en charge le filtre PercentDeadband lors de la surveillance des <i>Variables</i> AnalogItem type.
<i>Serveur</i>	Accès aux données – SemanticChanges	Prend en charge les changements sémantiques des éléments AnalogItem type (<i>Propriété</i> EURange et/ou <i>Propriété</i> EngineeringUnits). Prend en charge les bits de StatusCode des changements sémantiques, le cas échéant.
<i>Serveur</i>	Accès aux données – TwoState	Prend en charge les <i>Variables</i> TwoStateDiscreteType avec les Propriétés correspondantes.
<i>Serveur</i>	Accès aux données – MultiState	Prend en charge les <i>Variables</i> MultiStateDiscreteType avec les Propriétés correspondantes.
<i>Serveur</i>	Accès aux données – MultiStateValueDiscrete	Prend en charge les <i>Variables</i> MultiStateValueDiscreteType avec les Propriétés correspondantes.
<i>Serveur</i>	Accès aux données – ArrayItemType	Fournit des <i>Variables</i> de l'ArrayItem type ou l'un de ses sous-types (YArrayItem type, XYArrayItem type, ImageArrayType, CubeArrayType et NDimensionArrayType). Les sous-types pris en charge sont répertoriés. La prise en charge de ce type comprend la prise en charge de toutes les propriétés obligatoires, y compris AxisInformation.
<i>Serveur</i>	Accès aux données – Nombre Complexe	Prend en charge le type de données de Nombre Complexe. Ce type de données est disponible pour tout type de variable n'ayant pas d'autre restriction explicite.
<i>Serveur</i>	Accès aux données – Nombre DoubleComplex	Prend en charge le type de données de Nombre DoubleComplex. Ce type de données est disponible pour tout type de variable n'ayant pas d'autre restriction explicite.
<i>Client</i>	Accès aux données <i>Client</i> – Base	Comprend les types de <i>Variables</i> DataAccess. Utilise les Propriétés normalisées, le cas échéant.
<i>Client</i>	Accès aux données <i>Client</i> – AnalogItems	Comprend les <i>Variables</i> AnalogItem type avec les Propriétés correspondantes.
<i>Client</i>	Accès aux données <i>Client</i> – TwoState	Comprend les <i>Variables</i> TwoStateDiscreteType avec les Propriétés correspondantes.
<i>Client</i>	Accès aux données <i>Client</i> – MultiState	Comprend les <i>Variables</i> MultiStateDiscreteType avec les Propriétés correspondantes.
<i>Client</i>	Accès aux données <i>Client</i> – MultiStateValueDiscrete	Comprend les <i>Variables</i> MultiStateValueDiscrete avec les Propriétés correspondantes.
<i>Client</i>	Accès aux données <i>Client</i> – Bande Morte	Utilise PercentDeadband pour filtrer les modifications de valeurs des <i>Variables</i> AnalogItem type.
<i>Client</i>	Accès aux données <i>Client</i> – SemanticChange	Reconnaît le bit de changement sémantique dans le StatusCode tout en surveillant les éléments et en prenant les mesures appropriées. Généralement, le <i>Client</i> doit relire les Propriétés qui définissent la sémantique spécifique au type, telles que les propriétés EURange et EngineeringUnits.

Le Tableau 16 décrit les éléments relatifs au modèle d'information *Alarme* et *Conditions* qui peuvent être intégrés dans les profils. Les détails de ce modèle sont définis dans l'IEC 62541-9. Les *Serveurs* qui traitent des *Alarmes* et *Conditions* présentent ce modèle d'information et les *Clients* qui traitent des *Alarmes* et *Conditions* utilisent ce même modèle.

Tableau 16 – Alarmes et conditions

Catégorie	Intitulé	Description
Serveur	A & C – Base	Prend en charge le ConditionType du modèle d'Alarme & Condition.
Serveur	A & C – Activer	Prend en charge les Méthodes Activer et Désactiver.
Serveur	A & C – Rafraîchissement	Prend en charge la Méthode ConditionRefresh et le concept de rafraîchissement
Serveur	A & C – Refresh2	Prend en charge la Méthode ConditionRefresh2 et le concept de rafraîchissement fondé sur un élément surveillé.
Serveur	A & C – Instances	Prend en charge la présentation des instances de Conditions A&C dans l'AddressSpace.
Serveur	A & C – ConditionClasses	Prend en charge les différentes classes de Conditions pour le regroupement et le filtrage des Alarmes.
Serveur	A & C – Sous Classes de Conditions	Prend en charge l'assignation des différentes sous-classes de Conditions pour le regroupement et le filtrage des Alarmes.
Serveur	A & C – Acquiescement	Prend en charge le concept d'Acquiescement, la Méthode d'Acquiescement et le Type AcknowledgeableCondition.
Serveur	A & C – Confirmer	Prend en charge le concept de confirmation et la Méthode Confirmer.
Serveur	A & C – Commentaire	Prend en charge le concept de Commentaires et la Méthode AddComments.
Serveur	A & C – Alarme	Prend en charge les caractéristiques obligatoires du Type AlarmCondition.
Serveur	A & C – Paramètres d'Alarme	Prend en charge la collecte des données des paramètres d'alarme définies dans l'IEC 62541-9. Cela implique une instance parmi d'autres de l'AlarmMetricsType.
Serveur	A & C – Branche	Prend en charge l'embranchement des Types de Conditions et tous les sous-types, par exemple AcknowledgeableConditionType et AlarmConditionType, etc.
Serveur	A & C – Suspension	Prend en charge le concept de suspension et les méthodes TimedShelve, OneShotShelve et Unshelve.
Serveur	A & C – Suppression	Prend en charge le SuppressedState.
Serveur	A & C – Suppression par l'opérateur	Prend en charge les Méthodes Supprimer et UnSuppress pour permettre à un opérateur d'avoir un contrôle sur le SuppressedState.
Serveur	A & C – Silence	Prend en charge le concept de silence et la Méthode Silence.
Serveur	A & C – Hors Service	Prend en charge l'état OutOfService et la Méthode OutOfService.
Serveur	A & C – Activation et désactivation différées	Prend en charge les Propriétés OnDelay et OffDelay pour éliminer les Alarmes perturbatrices.
Serveur	A & C – Répétition des alarmes	Prend en charge les Propriétés ReAlarmTime et ReAlarmRepeatCount qui définissent la répétition automatique des Alarmes dans certaines conditions.
Serveur	A & C – Alarme "First in Group"	Prend en charge les éléments "FirstInGroup" d'une Alarme, qui indiquent quelle Alarme au sein d'un groupe a déclenché les autres.
Serveur	A & C – Son Audible	Prend en charge la Propriété AudibleSound. Cette Propriété contient le fichier son à lire si une Alarme sonore est générée.
Serveur	A & C – Niveau exclusif	Prend en charge le type d'Alarme de Niveau Exclusif.
Serveur	A & C – Limite exclusive	Prend en charge les Alarmes de Limite exclusive. Il importe qu'un Serveur qui prend en charge ce type d'alarme prenne en charge au moins l'un des sous-types: Niveau, Ecart ou RateofChange.
Serveur	A & C – Ecart exclusif	Prend en charge le type d'Alarme d'Ecart exclusif.
Serveur	A & C – RateOfChange exclusif	Prend en charge le type d'Alarme de RateOfChange exclusif.
Serveur	A & C – Limite non exclusive	Prend en charge les Alarmes de Limite Non Exclusive. Il importe qu'un Serveur qui prend en charge ce type d'alarme prenne en charge au moins l'un des sous-types: Niveau, Ecart ou RateofChange.
Serveur	A & C – Niveau non exclusif	Prend en charge le type d'Alarme de Niveau Non Exclusif.
Serveur	A & C – Ecart non exclusif	Prend en charge le type d'Alarme d'Ecart Non Exclusif.

Catégorie	Intitulé	Description
Serveur	A & C – RateOfChange non exclusif	Prend en charge le type d' <i>Alarme</i> de RateOfChange Non Exclusif.
Serveur	A & C – Discrètes	Prend en charge les types d' <i>Alarmes</i> discrètes.
Serveur	A & C – OffNormal	Prend en charge l'OffNormalAlarmType.
Serveur	A & C – OffNormal	Prend en charge SystemOffNormalAlarmType.
Serveur	A & C – Déclenchement	Prend en charge le type d' <i>Alarme</i> de Déclenchement.
Serveur	A & C – Anomalie	Prend en charge le type d' <i>Alarme</i> d'Anomalie.
Serveur	A & C – Dialogue	Prend en charge le DialogConditionType, y compris la <i>Méthode</i> Répondre.
Serveur	A & C – CertificateExpiration	Prend en charge le CertificateExpirationAlarmType.
Serveur	A & E – Conteneur de Mapping	Le <i>Serveur</i> utilise le mapping COM A&E spécifié dans l'annexe de l'IEC 62541-9 pour mettre en correspondance les Événements OPC-COM et les Événements A&C. Ceci inclut le mapping de la Classe <i>Condition</i> .
Client	A & C <i>Client</i> – Base	Utilise le ConditionType du modèle d' <i>Alarme & Condition</i> .
Client	A & C <i>Client</i> – Activer	Utilise les Méthodes Activer et Désactiver.
Client	A & C <i>Client</i> – Rafraîchissement	Utilise la <i>Méthode</i> ConditionRefresh et le concept de rafraîchissement.
Client	A & C <i>Client</i> – Refresh2	Utilise la <i>Méthode</i> ConditionRefresh2 et le concept de rafraîchissement fondé sur un élément surveillé.
Client	A & C <i>Client</i> – Instances	Utilise les instances de <i>Conditions</i> A&C lorsqu'elles sont présentées dans l'AddressSpace.
Client	A & C <i>Client</i> – ConditionClasses	Utilise les classes de <i>Conditions</i> pour grouper les <i>Alarmes</i> .
Client	A & C <i>Client</i> – Sous Classes de Conditions	Utilise les sous-classes de <i>Conditions</i> pour grouper ou pour filtrer les <i>Alarmes</i> .
Client	A & C <i>Client</i> – Acquiescement	Comprend le concept d'Acquiescement et le Type AcknowledgeableCondition, et utilise la <i>Méthode</i> d'Acquiescement si elle est demandée.
Client	A & C <i>Client</i> – Confirmer	Comprend le concept de confirmation des <i>Conditions</i> et utilise la <i>Méthode</i> Confirmer.
Client	A & C <i>Client</i> – Commentaire	Comprend le concept de Commentaires et la <i>Méthode</i> AddComments.
Client	A & C <i>Client</i> – Alarme	Comprend le concept d' <i>Alarmes</i> et utilise les caractéristiques obligatoires du Type AlarmCondition.
Client	A & C <i>Client</i> – Paramètres d'Alarme	Comprend et utilise les données des Paramètres d' <i>Alarme</i> définis dans l'IEC 62541-9. Cela implique la découverte d'instances de l'AlarmMetricsType, qui peuvent exister partout dans la hiérarchie HasNotifier.
Client	A & C <i>Client</i> – Branche	Peut utiliser et traiter les Branches de <i>Conditions</i> , y compris toutes les actions associées aux instances de <i>Conditions</i> précédentes.
Client	A & C <i>Client</i> – Suspension	Utilise le modèle de suspension, y compris les Méthodes TimedShelve, OneShotShelve et Unshelve.
Client	A & C <i>Client</i> – Suppression	Comprend le modèle de SuppressedState.
Client	A & C <i>Client</i> – Suppression par l'opérateur	Utilise les <i>Méthodes</i> Supprimer et UnSuppress pour permettre à un opérateur d'avoir un contrôle sur le SuppressedState.
Client	A & C <i>Client</i> – Silence	Comprend le modèle SilencedState et utilise la <i>Méthode</i> Silence.
Client	A & C <i>Client</i> – Hors Service	Comprend le modèle OutOfServiceState et utilise la <i>Méthode</i> OutOfService.
Client	A & C <i>Client</i> – Activation et désactivation différées	Utilise les Propriétés OnDelay et OffDelay pour éliminer les <i>Alarmes</i> perturbatrices.

Catégorie	Intitulé	Description
<i>Client</i>	A & C <i>Client</i> – Répétition des alarmes	Comprend et utilise les Propriétés ReAlarmTime et ReAlarmRepeatCount. Configure la <i>Propriété</i> ReAlarmTime pour la répétition automatique d'une <i>Alarme</i> . Noter que la configuration est possible seulement pour les <i>Serveurs</i> qui présentent des instances d' <i>Alarme</i> .
<i>Client</i>	A & C <i>Client</i> – <i>Alarme</i> "First in Group"	Utilise les éléments "FirstInGroup" d'une <i>Alarme</i> pour déterminer quelle <i>Alarme</i> au sein d'un groupe a déclenché les autres.
<i>Client</i>	A & C <i>Client</i> – Son Audible	Utilise la <i>Propriété</i> AudibleSound et lit le fichier son (le cas échéant).
<i>Client</i>	A & C <i>Client</i> – Niveau exclusif	Utilise les <i>Alarmes</i> de Niveau Exclusif.
<i>Client</i>	A & C <i>Client</i> – Limite exclusive	Utilise les <i>Alarmes</i> de Limite Exclusive. Exige l'utilisation d'au moins un des sous-types.
<i>Client</i>	A & C <i>Client</i> – Ecart exclusif	Utilise les <i>Alarmes</i> d'Ecart Exclusif.
<i>Client</i>	A & C <i>Client</i> – RateOfChange exclusif	Utilise les <i>Alarmes</i> de RateOfChange exclusif.
<i>Client</i>	A & C <i>Client</i> – Niveau non exclusif	Utilise les <i>Alarmes</i> de Niveau Non Exclusif
<i>Client</i>	A & C <i>Client</i> – Limite non exclusive	Utilise les <i>Alarmes</i> de Limite Non Exclusive. Exige l'utilisation d'au moins un des sous-types.
<i>Client</i>	A & C <i>Client</i> – Ecart non exclusif	Utilise les <i>Alarmes</i> d'Ecart Non Exclusif.
<i>Client</i>	A & C <i>Client</i> – RateOfChange non exclusif	Utilise les <i>Alarmes</i> de RateOfChange non exclusif.
<i>Client</i>	A & C <i>Client</i> – Discrètes	Utilise les types d' <i>Alarmes</i> discrètes.
<i>Client</i>	A & C <i>Client</i> – OffNormal	Utilise l'OffNormalAlarmType.
<i>Client</i>	A & C <i>Client</i> – SystemOffNormal	Utilise le SystemOffNormalAlarmType.
<i>Client</i>	A & C <i>Client</i> – Déclenchement	Utilise le TripAlarmType.
<i>Client</i>	A & C <i>Client</i> – Anomalie	Utilise le type d' <i>Alarme</i> d'Anomalie.
<i>Client</i>	A & C <i>Client</i> – Dialogue	Utilise DialogConditionType, y compris la <i>Méthode</i> Répondre.
<i>Client</i>	A & C <i>Client</i> – CertificateExpiration	Utilise le CertificateExpirationAlarmType.

Le Tableau 17 décrit les éléments relatifs au modèle d'information d'accès aux données historiques qui peuvent être intégrés dans les profils. Les détails de ce modèle sont définis dans l'IEC 62541-11. Les *Serveurs* qui prennent en charge un niveau de données historiques présentent ce modèle d'information et les *Clients* qui utilisent les données historiques utilisent ce même modèle.

Tableau 17 – Accès à l'historique

Catégorie	Intitulé	Description
Serveur	Accès à l'historique – Lecture de Données Brutes	Fournit une prise en charge générale de l'Accès à l'historique de base, avec la lecture de données brutes en utilisant la structure ReadRawModifiedDetails. Si l'intervalle de temps est spécifié avec un temps de départ et d'arrêt ainsi qu'un nombre de valeurs (il importe de fournir au moins deux des trois paramètres), le fanion ReadModified est mis à FALSE.
Serveur	Accès à l'historique – Point de Continuation de Lecture des Données avec un Maximum de Nœuds	Prend en charge suffisamment de points de continuation pour couvrir le nombre de points pris en charge indiqué dans la <i>Propriété</i> OperationLimits du <i>Serveur</i> MaxNodesPerHistoryReadData pour accéder aux données historiques.
Serveur	Accès à l'historique – Instance Temporelle	Prend en charge la lecture des données historiques à une instance spécifiée dans le temps à l'aide de la structure ReadAtTimeDetails.
Serveur	Accès à l'historique – Agrégats	Prend en charge la lecture d'un ou plusieurs Agrégats de valeurs historiques de <i>Variables</i> à l'aide de la structure ReadProcessedDetails. Il importe de prendre en charge au moins l'un des Agrégats décrits dans l'IEC 62541-13.
Serveur	Accès à l'historique – Insertion de Valeurs	Prend en charge l'insertion de valeurs historiques de <i>Variables</i> .
Serveur	Accès à l'historique – Suppression de Valeurs	Prend en charge la suppression de valeurs historiques de <i>Variables</i> .
Serveur	Accès à l'historique – Mise à Jour de Valeurs	Prend en charge la mise à jour de valeurs historiques de <i>Variables</i> .
Serveur	Accès à l'historique – Remplacement de Valeurs	Prend en charge le remplacement de valeurs historiques de <i>Variables</i> .
Serveur	Accès à l'historique – Valeurs Modifiées	Prend en charge le maintien d'anciennes valeurs pour les données historiques qui ont été mises à jour et la récupération de ces valeurs à l'aide de la structure ReadRawModifiedDetails (fanion ReadModified mis à TRUE).
Serveur	Accès à l'historique – Annotations	Prend en charge la saisie et la récupération des Annotations pour les données historiques. La récupération est effectuée à l'aide de la fonctionnalité de lecture de données historiques brutes normalisée (ReadRawModifiedDetails). La saisie utilise la fonctionnalité de mise à jour historique (UpdateStructureDataDetails) normalisée.
Serveur	Accès à l'historique – ServerTimestamp	Prend en charge l'apport d'un ServerTimestamp (ainsi que le SourceTimestamp par défaut).
Serveur	Accès à l'historique – Lecture des données structurées brutes	Prend en charge l'Accès à l'historique ReadRawModified pour les données structurées. La prise en charge de la structure pour une annotation n'est pas jugée comme une prise en charge des données structurées génériques.
Serveur	Accès à l'historique – Instance temporelle de données structurées	Prend en charge l'Accès à l'historique pour les données structurées. Prend en charge ReadAtTimeDetails pour les données structurées. La prise en charge de la structure pour une annotation n'est pas jugée comme une prise en charge des données structurées génériques.
Serveur	Accès à l'historique – Insérer des données structurées	Prend en charge l'Accès à l'historique pour les données structurées. Insère des données structurées. La prise en charge de la structure pour une annotation n'est pas jugée comme une prise en charge des données structurées génériques.
Serveur	Accès à l'historique – Supprimer des données structurées	Prend en charge l'Accès à l'historique pour les données structurées. Supprime les données existantes. La prise en charge de la structure pour une annotation n'est pas jugée comme une prise en charge des données structurées génériques.
Serveur	Accès à l'historique – Mise à Jour de données structurées	Prend en charge l'Accès à l'historique pour les données structurées. Met à jour les données existantes. La prise en charge de la structure pour une annotation n'est pas jugée comme une prise en charge des données structurées génériques.
Serveur	Accès à l'historique – Remplacer des données structurées	Prend en charge le remplacement de données historiques structurées. La prise en charge de la structure pour une annotation n'est pas jugée comme une prise en charge des données structurées génériques.

Catégorie	Intitulé	Description
Serveur	Accès à l'historique – Lecture de données structurées modifiées	Prend en charge le maintien d'anciennes valeurs pour les données historiques structurées qui ont été mises à jour et la récupération de ces valeurs. Utilise la structure ReadRawModifiedDetails (fanion ReadModified mis à TRUE) pour les données structurées. La prise en charge de la structure pour une annotation n'est pas jugée comme une prise en charge des données structurées génériques.
Serveur	Accès à l'historique – Evénements	Prend en charge la récupération d'Evénements historiques à l'aide de la structure ReadEventDetails. Cela comprend la prise en charge du filtrage simple d'Evénements. Les champs d'Evénements stockés sont spécifiques au serveur, mais au moins les champs obligatoires de baseEventType sont exigés.
Serveur	Accès à l'historique – Evénements – Point de Continuation de Lecture d'Evénements Max	Prend en charge suffisamment de points de continuation pour couvrir le nombre de lectures d'Evénements prises en charge indiqué dans la Propriété OperationLimits du Serveur MaxNodesPerHistoryReadEvents pour l'Accès à l'historique d'Evénements.
Serveur	Accès à l'historique – Insertion d'Evénements	Prend en charge l'insertion d'Evénements historiques.
Serveur	Accès à l'historique – Mise à Jour d'Evénements	Prend en charge la mise à jour d'Evénements historiques.
Serveur	Accès à l'historique – Remplacement d'Evénements	Prend en charge le remplacement d'Evénements historiques.
Serveur	Accès à l'historique – Suppression d'Evénements	Prend en charge la suppression d'Evénements historiques.
Client	Accès à l'historique Client – Parcourir	Utilise le <i>Jeu de services</i> Vue pour découvrir des <i>Nœuds</i> avec des données historiques.
Client	Accès à l'historique Client – Lecture de Données Brutes	Utilise le <i>Service</i> HistoryRead pour lire des données historiques brutes à l'aide de la Structure ReadRawModifiedDetails (fanion ReadModified mis à FALSE).
Client	Accès à l'historique Client – Lecture de Données Modifiées	Utilise le <i>Service</i> HistoryRead pour lire des données historiques modifiées à l'aide de la Structure ReadRawModifiedDetails (fanion ReadModified mis à TRUE).
Client	Accès à l'historique Client – Lecture d'Agrégats	Utilise le <i>Service</i> HistoryRead pour lire des données historiques Agrégées. Cela comprend l'utilisation d'au moins un des Agrégats définis dans l'IEC 62541-13.
Client	Accès à l'historique Client – Données brutes structurées	Utilise le <i>Service</i> HistoryRead pour lire des données historiques brutes à l'aide de la Structure ReadRawModifiedDetails (fanion ReadModified mis à FALSE) pour les données structurées.
Client	Accès à l'historique Client – Lecture de données structurées modifiées	Utilise le <i>Service</i> HistoryRead pour lire des données historiques structurées modifiées à l'aide de la Structure ReadRawModifiedDetails (fanion ReadModified mis à TRUE).
Client	Accès à l'historique Client – Insérer des données structurées	Utilise le <i>Service</i> HistoryUpdate pour insérer des valeurs de données historiques pour les données structurées.
Client	Accès à l'historique Client – Supprimer des données structurées	Utilise le <i>Service</i> HistoryUpdate pour supprimer des valeurs de données historiques pour les données structurées.
Client	Accès à l'historique Client – Mettre à Jour des données structurées	Utilise le <i>Service</i> HistoryUpdate pour mettre à jour des valeurs de données historiques pour les données structurées.
Client	Accès à l'historique Client – Remplacer des données structurées	Utilise le <i>Service</i> HistoryUpdate pour remplacer des valeurs de données historiques pour les données structurées.
Client	Accès à l'historique Client – Instance temporelle de données structurées	Lit les données historiques à une instance spécifiée dans le temps pour les données structurées. Utilise la structure ReadAtTimeDetails.

Catégorie	Intitulé	Description
<i>Client</i>	Accès à l'historique <i>Client</i> – Lecture d'Événements	Utilise le <i>Service</i> HistoryRead pour lire les données historiques d'Événements à l'aide de la Structure ReadEventDetails.
<i>Client</i>	Accès à l'historique <i>Client</i> – Insertion d'Événements	Utilise le <i>Service</i> HistoryUpdate pour insérer des Événements historiques.
<i>Client</i>	Accès à l'historique <i>Client</i> – Mise à Jour d'Événements	Utilise le <i>Service</i> HistoryUpdate pour mettre à jour des Événements historiques.
<i>Client</i>	Accès à l'historique <i>Client</i> – Remplacement d'Événements	Utilise le <i>Service</i> HistoryUpdate pour remplacer des Événements historiques.
<i>Client</i>	Accès à l'historique <i>Client</i> – Suppression d'Événements	Utilise le <i>Service</i> HistoryUpdate pour supprimer des Événements historiques.
<i>Client</i>	Accès à l'historique <i>Client</i> – Insertion de Données	Utilise le <i>Service</i> HistoryUpdate pour insérer des valeurs de données historiques.
<i>Client</i>	Accès à l'historique <i>Client</i> – Suppression de Données	Utilise le <i>Service</i> HistoryUpdate pour supprimer des valeurs de données historiques.
<i>Client</i>	Accès à l'historique <i>Client</i> – Mise à Jour de Données	Utilise le <i>Service</i> HistoryUpdate pour mettre à jour des valeurs de données historiques.
<i>Client</i>	Accès à l'historique <i>Client</i> – Remplacement de Données	Utilise le <i>Service</i> HistoryUpdate pour remplacer des valeurs de données historiques.
<i>Client</i>	Accès à l'historique <i>Client</i> – Annotations	Saisit et récupère des Annotations de données historiques. La récupération est effectuée à l'aide de la fonctionnalité de lecture de données historiques brutes normalisée (ReadRawModifiedDetails). La saisie utilise la fonctionnalité de mise à jour historique (UpdateStructureDataDetails) normalisée.
<i>Client</i>	Accès à l'historique <i>Client</i> – Instance Temporelle	Lit les données historiques à une instance spécifiée dans le temps à l'aide de la structure ReadAtTimeDetails.
<i>Client</i>	Accès à l'historique <i>Client</i> – Horodatage <i>Serveur</i>	Utilise le ServerTimestamp (ainsi que le SourceTimestamp par défaut), s'il est fourni par le <i>Serveur</i> .

Le Tableau 18 décrit les éléments relatifs aux Agrégats qui peuvent être intégrés dans les profils. Les *Serveurs* qui prennent en charge les Agrégats présentent cette fonctionnalité et les *Clients* qui utilisent les Agrégats mettent en œuvre une partie de la fonctionnalité.

Tableau 18 – Agrégats

Catégorie	Intitulé	Description
Serveur	Agrégat – Configuration maître	Prend en charge un <i>Objet</i> d'AggregateConfigurationType parmi les HistoricalServerCapabilities (définies dans l'IEC 62541-11).
Serveur	Agrégat – Configuration historique	Prend en charge au moins un <i>Objet</i> d'AggregateConfigurationType. Les <i>Objets</i> d'AggregateConfigurationType surviennent dans un <i>Objet</i> d'HistoricalConfiguration, permettant les configurations spécifiques aux <i>Variables</i> .
Serveur	Agrégat – Interpolative	Prend en charge l'Agrégat Interpolative pour l'Accès à l'historique.
Serveur	Agrégat – Average	Prend en charge l'Agrégat Average pour l'Accès à l'historique.
Serveur	Agrégat – TimeAverage	Prend en charge l'Agrégat TimeAverage pour l'Accès à l'historique.
Serveur	Agrégat – TimeAverage2	Prend en charge l'Agrégat TimeAverage2 pour l'Accès à l'historique.
Serveur	Agrégat – Total	Prend en charge l'Agrégat Total pour l'Accès à l'historique.
Serveur	Agrégat – Total2	Prend en charge l'Agrégat Total2 pour l'Accès à l'historique.
Serveur	Agrégat – Minimum	Prend en charge l'Agrégat Minimum pour l'Accès à l'historique.
Serveur	Agrégat – MinimumActualTime	Prend en charge l'Agrégat MinimumActualTime pour l'Accès à l'historique.
Serveur	Agrégat – Minimum2	Prend en charge l'Agrégat Minimum2 pour l'Accès à l'historique.
Serveur	Agrégat – MinimumActualTime2	Prend en charge l'Agrégat MinimumActualTime2 pour l'Accès à l'historique.
Serveur	Agrégat – Maximum	Prend en charge l'Agrégat Maximum pour l'Accès à l'historique.
Serveur	Agrégat – MaximumActualTime	Prend en charge l'Agrégat MaximumActualTime pour l'Accès à l'historique.
Serveur	Agrégat – Maximum2	Prend en charge l'Agrégat Maximum2 pour l'Accès à l'historique.
Serveur	Agrégat – MaximumActualTime2	Prend en charge l'Agrégat MaximumActualTime2 pour l'Accès à l'historique.
Serveur	Agrégat – Range	Prend en charge l'Agrégat Range pour l'Accès à l'historique.
Serveur	Agrégat – Range2	Prend en charge l'Agrégat Range2 pour l'Accès à l'historique.
Serveur	Agrégat – Count	Prend en charge l'Agrégat Count pour l'Accès à l'historique.
Serveur	Agrégat – DurationInStateZero	Prend en charge l'Agrégat DurationInStateZero pour l'Accès à l'historique.
Serveur	Agrégat – DurationInStateNonZero	Prend en charge l'Agrégat DurationInStateNonZero pour l'Accès à l'historique.
Serveur	Agrégat – NumberOfTransitions	Prend en charge l'Agrégat NumberOfTransitions pour l'Accès à l'historique.
Serveur	Agrégat – Start	Prend en charge l'Agrégat Start pour l'Accès à l'historique.
Serveur	Agrégat – StartBound	Prend en charge l'Agrégat StartBound pour l'Accès à l'historique.
Serveur	Agrégat – End	Prend en charge l'Agrégat End pour l'Accès à l'historique.
Serveur	Agrégat – EndBound	Prend en charge l'Agrégat EndBound pour l'Accès à l'historique.
Serveur	Agrégat – Delta	Prend en charge l'Agrégat Delta pour l'Accès à l'historique.
Serveur	Agrégat – DeltaBounds	Prend en charge l'Agrégat DeltaBounds pour l'Accès à l'historique.
Serveur	Agrégat – DurationGood	Prend en charge l'Agrégat DurationGood pour l'Accès à l'historique.
Serveur	Agrégat – DurationBad	Prend en charge l'Agrégat DurationBad pour l'Accès à l'historique.
Serveur	Agrégat – PercentGood	Prend en charge l'Agrégat PercentGood pour l'Accès à l'historique.
Serveur	Agrégat – PercentBad	Prend en charge l'Agrégat PercentBad pour l'Accès à l'historique.
Serveur	Agrégat – WorstQuality	Prend en charge l'Agrégat WorstQuality pour l'Accès à l'historique.

Catégorie	Intitulé	Description
Serveur	Agrégat – WorstQuality2	Prend en charge l'Agrégat WorstQuality2 pour l'Accès à l'historique.
Serveur	Agrégat – AnnotationCount	Prend en charge l'Agrégat AnnotationCount pour l'Accès à l'historique.
Serveur	Agrégat – StandardDeviationSample	Prend en charge l'Agrégat StandardDeviationSample pour l'Accès à l'historique.
Serveur	Agrégat – VarianceSample	Prend en charge l'Agrégat VarianceSample pour l'Accès à l'historique.
Serveur	Agrégat – StandardDeviationPopulation	Prend en charge l'Agrégat StandardDeviationPopulation pour l'Accès à l'historique.
Serveur	Agrégat – VariancePopulation	Prend en charge l'Agrégat VariancePopulation pour l'Accès à l'historique.
Serveur	Agrégat – Custom	Le <i>Serveur</i> prend en charge les Agrégats personnalisés pour l'Accès à l'historique pour lesquels aucun essai normalisé n'est défini. Ces Agrégats sont indiqués comme non soumis à l'essai par cette <i>ConformanceUnit</i> .
Serveur	Abonnement aux Agrégats – Filter	Prend en charge les filtres pour l'abonnement "Agrégat", qui nécessite qu'au moins un des Agrégats définis soit pris en charge, comme défini dans l'IEC 62541-13.
Serveur	Abonnement aux Agrégats – Interpolative	Prend en charge le filtre d'abonnement pour l'Agrégat "Interpolative".
Serveur	Abonnement aux Agrégats – Average	Prend en charge le filtre d'abonnement pour l'Agrégat "Average".
Serveur	Abonnement aux Agrégats – TimeAverage	Prend en charge le filtre d'abonnement pour l'Agrégat "TimeAverage".
Serveur	Abonnement aux Agrégats – TimeAverage2	Prend en charge le filtre d'abonnement pour l'Agrégat "TimeAverage2".
Serveur	Abonnement aux Agrégats – Total	Prend en charge le filtre d'abonnement pour l'Agrégat "Total".
Serveur	Abonnement Agrégat – Total2	Prend en charge le filtre d'abonnement pour l'Agrégat "Total2".
Serveur	Abonnement aux Agrégats – Minimum	Prend en charge le filtre d'abonnement pour l'Agrégat "Minimum".
Serveur	Abonnement aux Agrégats – MinimumActualTime	Prend en charge le filtre d'abonnement pour l'Agrégat "MinimumActualTime".
Serveur	Abonnement Agrégat – Minimum2	Prend en charge le filtre d'abonnement pour l'Agrégat "Minimum2".
Serveur	Abonnement Agrégat – MinimumActualTime2	Prend en charge le filtre d'abonnement pour l'Agrégat "MinimumActualTime2".
Serveur	Abonnement aux Agrégats – Maximum	Prend en charge le filtre d'abonnement pour l'Agrégat "Maximum".
Serveur	Abonnement aux Agrégats – MaximumActualTime	Prend en charge le filtre d'abonnement pour l'Agrégat "MaximumActualTime".
Serveur	Abonnement Agrégat – Maximum2	Prend en charge le filtre d'abonnement pour l'Agrégat "Maximum2".
Serveur	Abonnement Agrégat – MaximumActualTime2	Prend en charge le filtre d'abonnement pour l'Agrégat "MaximumActualTime2".
Serveur	Abonnement aux Agrégats – Range	Prend en charge le filtre d'abonnement pour l'Agrégat "Range".
Serveur	Abonnement Agrégat – Range2	Prend en charge le filtre d'abonnement pour l'Agrégat "Range2".
Serveur	Abonnement aux Agrégats – Count	Prend en charge le filtre d'abonnement pour l'Agrégat "Count".
Serveur	Abonnement aux Agrégats – DurationInStateZero	Prend en charge le filtre d'abonnement pour l'Agrégat "DurationInStateZero".
Serveur	Abonnement aux Agrégats – DurationInStateNonZero	Prend en charge le filtre d'abonnement pour l'Agrégat "DurationInStateNonZero".