DEUTSCHE NORM

August 2021

DIN EN IEC 62541-12

ICS 25.040.40; 35.200; 35.240.50

OPC Unified Architecture – Teil 12: Erkundung und globale Dienste (IEC 62541-12:2020); Englische Fassung EN IEC 62541-12:2020

OPC unified architecture –
Part 12: Discovery and global services
(IEC 62541-12:2020);
English version EN IEC 62541-12:2020

Architecture unifiée OPC –
Partie 12: Services globaux et de découverte
(IEC 62541-12:2020);
Version anglaise EN IEC 62541-12:2020

Gesamtumfang 114 Seiten

DKE Deutsche Kommission Elektrotechnik Elektronik Informationstechnik in DIN und VDE

Documents
Document Sharing Hub

Full document details: DIN EN IEC 62541-12:2021-08

Anwendungsbeginn

Anwendungsbeginn für die von CENELEC am 2020-07-21 angenommene Europäische Norm als DIN-Norm ist 2021-08-01.

Nationales Vorwort

Vorausgegangener Norm-Entwurf: E DIN EN IEC 62541-12:2019-09.

Für dieses Dokument ist das nationale Arbeitsgremium DKE/K 931 "Systemaspekte der Automatisierung" der DKE Deutsche Kommission Elektrotechnik Elektronik Informationstechnik in DIN und VDE (www.dke.de) zuständig.

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

Die enthaltene IEC-Publikation wurde vom IEC/SC 65E "Devices and integration in enterprise systems" erarbeitet und unterliegt dem Copyright der IEC.

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

In diesem Dokument und in den anderen Teilen der IEC 62541-Reihe werden bestimmte Dokumentenkonventionen verwendet:

Kursivschrift wird verwendet, um einen definierten Begriff oder eine Definition zu kennzeichnen, der/die in dem Abschnitt "Begriffe und Abkürzungen" in einem der Teile der IEC 62541-Serie erscheint.

Kursivschrift wird auch verwendet, um den Namen eines Service-Eingangs- oder -Ausgangsparameters oder den Namen einer Struktur oder eines Elements einer Struktur zu kennzeichnen, die normalerweise in Tabellen definiert sind.

Die *kursiv* gedruckten Begriffe und Namen werden mit wenigen Ausnahmen in Camel-Case-Schreibweise geschrieben (die Schreibweise von zusammengesetzten Wörtern oder Phrasen, in denen die Elemente ohne Leerzeichen verbunden sind, wobei der Anfangsbuchstabe jedes Elements innerhalb der Verbindung groß geschrieben wird). Zum Beispiel lautet der definierte Begriff "AddressSpace" statt "Address Space". Dies macht es einfacher zu verstehen, dass es eine einzige Definition für AddressSpace gibt und nicht separate Definitionen für Address und Space.

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

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

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

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

Documents
Document Sharing Hub

Full document details: DIN EN IEC 62541-12:2021-08

Da sich die Benutzer der vorliegenden Norm der englischen Sprache als Fachsprache bedienen, wird die Englische Fassung der EN IEC 62541-12 veröffentlicht. Zu deren Abschnitt 1, der den Anwendungsbereich festlegt und Abschnitt 3 (außer Unterabschnitte 3.2 und 3.3), der die Begriffe festlegt, wurde eine Übersetzung angefertigt und als informativer Nationaler Anhang NA der vorliegenden Norm hinzugefügt. Für die meisten der verwendeten Begriffe existieren keine gebräuchlichen deutschen Benennungen, da sich die deutschen Anwender in der Regel ebenfalls der englischen Benennungen bedienen. Diese Norm steht nicht in unmittelbarem Zusammenhang mit Rechtsvorschriften und ist nicht als Sicherheitsnorm anzusehen.

Documents
Document Sharing Hub
Full document details: DIN EN IEC 62541-12:2021-08

Nationaler Anhang NA
(informativ)

Auszugsweise Übersetzung

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

1 Anwendungsbereich

Dieser Teil von IEC 62541 legt fest, wie die *Clients* und *Server* der OPC Unified Architecture (OPC UA) in verschiedenen Szenarien mit *DiscoveryServers* interagieren. Er legt die Anforderungen für den *LocalDiscoveryServer, LocalDiscoveryServer-ME and GlobalDiscoveryServer* fest. Er legt auch die Informationsmodelle für die Verwaltung von *Zertifikaten, KeyCredential*-Verwaltung und *Autorisierungsdienste* fest.

3 Begriffe, Abkürzungen und Vereinbarungen

3.1 Begriffe

Für die Anwendung dieses Dokuments gelten die Begriffe nach IEC TR 62541-1, IEC TR 62541-2, IEC 62541-3, IEC 62541-4, IEC 62541-6, IEC 62541-9 und die folgenden Begriffe.

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

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

3.1.1
Zertifikatgruppe
(en: Certificate Group)
Kontext zur Beschreibung der *VertrauensListe* (en: *Trust List*) und des/der *Zertifikate(s)* in Zusammenhang mit einer *Anwendung*

3.1.2
Zertifikatanforderung
(en: Certificate Request)
PKCS #10-codierte Struktur zum Anfordern eines neuen *Zertifikats* von einer *Zertifizierungsstelle* (en: *Certificate Authority*)

3.1.3
SchlüsselInformation
(en: KeyCredential)
ein eindeutiger und ein geheimer Bezeichner für den Zugriff auf einen *Server*, einen *Autorisierungsdienst* (en: *Authorization Service*) oder einen Broker

Anmerkung 1 zum Begriff: Benutzername und Passwort sind Beispiele für Erkennungsmerkmale (en: credentials).

3.1.4
SchlüsselInformationsDienst
(en: KeyCredentialService)
Softwareanwendung, die *KeyCredentials* für den Zugriff auf einen *Server*, einen *Autorisierungsdienst* oder einen *Broker* bereitstellt

Documents
Document Sharing Hub
Full document details: DIN EN IEC 62541-12:2021-08

3.1.5
DirectoryDienst
(en: DirectoryService)
Softwareanwendung oder eine Reihe von Anwendungen, die Informationen über Ressourcen wie Computer oder Dienste speichert und organisiert

3.1.6
DiscoveryServer

Anwendung, in der eine Liste von OPC-UA-*Servern* verwaltet wird, die in dem Netzwerk vorhanden sind und Mechanismen zum Abrufen der Liste durch Clients bereitstellt

3.1.7
DiscoveryUrl

URL für einen Netzwerk-*Endpunkt* (en: *Endpoint*), die die Informationen zum Verbinden mit einem *Client* oder *Server* bereitstellt

3.1.8
GlobalDiscoveryServer (GDS)

DiscoveryServer, auf dem eine Liste von OPC-UA-*Anwendungen* verwaltet wird, die in einem Verwaltungsbereich verfügbar sind

Anmerkung 1 zum Begriff: Ein GDS kann auch Zertifikatverwaltungsdienste bereitstellen.

3.1.9
IPAddresse
(en: IPAdress)
eindeutige Nummer, die einer Netzwerkschnittstelle zugewiesen wird und die es ermöglicht, Internet Protocol (IP)-Anforderungen an diese Schnittstelle weiterzuleiten

Anmerkung 1 zum Begriff: Eine *IPAddress* eines Hosts kann sich nach einer bestimmten Zeit ändern.

3.1.10
LocalDiscoveryServer
LDS

DiscoveryServer, auf dem eine Liste aller *Server* verwaltet wird, die auf ihm registriert sind

Anmerkung 1 zum Begriff: *Server* registrieren sich üblicherweise mit dem LDS auf demselben Host.

3.1.11
LocalDiscoveryServer-ME
LDS-ME

LocalDiscoveryServer, der die MulticastExtension beinhaltet

3.1.12
MulticastErweiterung
(en: MulticastExtension)
Erweiterung eines *LocalDiscoveryServer*, mit der die Unterstützung für das mDNS-Protokoll hinzugefügt wird

3.1.13
MulticastUnternetz
(en: MulticastSubnet)
Netzwerk, mit dem ermöglicht wird, Multicast-Pakete an alle mit dem Netzwerk verbundene Knoten zu senden

Anmerkung 1 zum Begriff: Ein *MulticastSubnet* ist nicht zwingend das Gleiche wie ein TCP/IP-Subnet.

Documents
Document Sharing Hub
Full document details: DIN EN IEC 62541-12:2021-08

3.1.14
NetzwerkDienst
(en: NetworkService)
gesicherte Ressource in einem Netzwerk, die Funktionen bereitstellt, die von *Clients* und/oder *Servern* verwendet werden

Anmerkung 1 zum Begriff: Ein *AutorisierungsDienst* (en: *Authorization Service (AS)*) ist ein Beispiel für einen *NetzwerkDienst*.

3.1.15
ServerFähigkeitsIdentifikator
(en: ServerCapabilityIdentifier)
kurzer Bezeichner, der eine Reihe von ermittelbaren Fähigkeiten, die von einem Server unterstützt werden, eindeutig identifiziert

Anmerkung 1 zum Begriff: Die Liste der derzeit definierten *ServerCapabilityIdentifiers* befindet sich in Anhang D.

Documents
Document Sharing Hub

Full document details: DIN EN IEC 62541-12:2021-08

EUROPEAN STANDARD

NORME EUROPÉENNE

EUROPÄISCHE NORM

EN IEC 62541-12

August 2020

ICS 25.040.40

English Version

OPC unified architecture - Part 12: Discovery and global services (IEC 62541-12:2020)

Architecture unifiée OPC - Partie 12: Services globaux et de découverte
(IEC 62541-12:2020)

OPC Unified Architecture - Teil 12: Erkundung und globale Dienste
(IEC 62541-12:2020)

This European Standard was approved by CENELEC on 2020-07-21. CENELEC members are bound to comply with the CEN/CENELEC Internal Regulations which stipulate the conditions for giving this European Standard the status of a national standard without any alteration.

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

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

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

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

CEN-CENELEC Management Centre: Rue de la Science 23, B-1040 Brussels

Documents
Document Sharing Hub

Full document details: DIN EN IEC 62541-12:2021-08

European foreword

The text of document 65E/711/FDIS, future edition 1 of IEC 62541-12, prepared by SC 65E "Devices and integration in enterprise systems" of IEC/TC 65 "Industrial-process measurement, control and automation" was submitted to the IEC-CENELEC parallel vote and approved by CENELEC as EN IEC 62541-12:2020.

The following dates are fixed:

- latest date by which the document has to be implemented at national level by publication of an identical national standard or by endorsement (dop) 2021-04-21
- latest date by which the national standards conflicting with the document have to be withdrawn (dow) 2023-07-21

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

Endorsement notice

The text of the International Standard IEC 62541-12:2020 was approved by CENELEC as a European Standard without any modification.

Documents
Document Sharing Hub

Full document details: DIN EN IEC 62541-12:2021-08

Annex ZA
(normative)

Normative references to international publications with their corresponding European publications

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

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

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

Publication	Year	Title	EN/HD	Year
IEC/TR 62541-1	–	OPC unified architecture – Part 1: Overview and concepts	CLC/TR 62541-1	–
IEC/TR 62541-2	–	OPC unified architecture – Part 2: Security model	CLC/TR 62541-2	–
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 62541-7	–	OPC unified architecture – Part 7: Profiles	–	–
IEC 62541-9	–	OPC Unified Architecture – Part 9: Alarms and Conditions	–	–
IEC 62541-14	–	OPC Unified Architecture – Part 14: PubSub	–	–
X.500: ISO/IEC 9594-1	2017	Information technology – Open Systems Interconnection – The Directory – Part 1: Overview of concepts, models and services	–	
IETF RFC 1035	–	Domain Names – Implementation and Specification	–	–
IETF RFC 2986	–	PKCS #10: Certification Request Syntax Specification Version 1.7	–	–
IETF RFC 3927	–	Dynamic Configuration of IPv4 Link-Local Addresses	–	–
IETF RFC 5958	–	Asymmetric Key Packages	–	–
IETF RFC 6762	–	mDNS: Multicast DNS	–	–
IETF RFC 6763	–	DNS-SD: DNS Based Service Discovery	–	–
IETF RFC 7030	–	Enrollment over Secure Transport	–	–
PKCS #12	–	Personal Information Exchange Syntax	–	–
DI	–	OPC Unified Architecture for Devices (DI)	–	–

AVA Documents
Document Sharing Hub

Full document details: DIN EN IEC 62541-12:2021-08

ADI	–	OPC Unified Architecture for Analyzer Devices (ADI)	–	–
PLCopen	–	OPC Unified Architecture / PLCopen Information Model	–	–
FDI	–	OPC Unified Architecture for FDI	–	–
ISA-95	–	ISA-95 Common Object Model	–	–

Documents
Document Sharing Hub

Full document details: DIN EN IEC 62541-12:2021-08

CONTENTS

European foreword 8
Annex ZA (normative) Normative references to international publications with their corresponding European publications 9
1 Scope 18
2 Normative references 18
3 Terms, definitions, abbreviated terms and conventions 19
3.1 Terms and definitions 19
3.2 Abbreviated terms and symbols 21
3.3 Conventions for namespaces 21
4 The discovery process 22
4.1 Overview 22
4.2 Registration and announcement of Applications 22
4.2.1 Overview 22
4.2.2 Hosts with a LocalDiscoveryServer 22
4.2.3 Hosts without a LocalDiscoveryServer 23
4.3 The discovery process for Clients to find Servers 24
4.3.1 Overview 24
4.3.2 Security 24
4.3.3 Simple Discovery with a DiscoveryUrl 24
4.3.4 Local Discovery 25
4.3.5 MulticastSubnet Discovery 25
4.3.6 Global Discovery 26
4.3.7 Combined Discovery Process for Clients 27
5 Local Discovery Server 28
5.1 Overview 28
5.2 Security considerations for Multicast DNS 28
6 Global Discovery Server 28
6.1 Overview 28
6.2 Network architectures 29
6.2.1 Overview 29
6.2.2 Single MulticastSubnet 30
6.2.3 Multiple MulticastSubnet 31
6.2.4 No MulticastSubnet 32
6.2.5 Domain Names and MulticastSubnets 32
6.3 Information Model 33
6.3.1 Overview 33
6.3.2 Directory 33
6.3.3 DirectoryType 33

AVA Documents
Document Sharing Hub

Full document details: DIN EN IEC 62541-12:2021-08

6.3.4 FindApplications 34
6.3.5 ApplicationRecordDataType 35
6.3.6 RegisterApplication 36
6.3.7 UpdateApplication 37
6.3.8 UnregisterApplication 38
6.3.9 GetApplication 39
6.3.10 QueryApplications 39
6.3.11 QueryServers (deprecated) 41
6.3.12 ApplicationRegistrationChangedAuditEventType 43
7 Certificate management overview 43
7.1 Overview 43
7.2 Pull Management 44
7.3 Push management 45
7.4 Provisioning 46
7.5 Common Information Model 46
7.5.1 Overview 46
7.5.2 TrustListType 46
7.5.3 OpenWithMasks 47
7.5.4 CloseAndUpdate 48
7.5.5 AddCertificate 49
7.5.6 RemoveCertificate 50
7.5.7 TrustListDataType 51
7.5.8 TrustListMasks 51
7.5.9 TrustListOutOfDateAlarmType 52
7.5.10 CertificateGroupType 52
7.5.11 CertificateType 53
7.5.12 ApplicationCertificateType 53
7.5.13 HttpsCertificateType 54
7.5.14 UserCredentialCertificateType 54
7.5.15 RsaMinApplicationCertificateType 54
7.5.16 RsaSha256ApplicationCertificateType 55
7.5.17 CertificateGroupFolderType 55
7.5.18 TrustListUpdatedAuditEventType 56
7.6 Information Model for Pull Certificate Management 56
7.6.1 Overview 56
7.6.2 CertificateDirectoryType 57
7.6.3 StartSigningRequest 58
7.6.4 StartNewKeyPairRequest 60
7.6.5 FinishRequest 62

Documents
Document Sharing Hub
Full document details: DIN EN IEC 62541-12:2021-08

7.6.6 GetCertificateGroups 63
7.6.7 GetTrustList 64
7.6.8 GetCertificateStatus 65
7.6.9 CertificateRequestedAuditEventType 66
7.6.10 CertificateDeliveredAuditEventType 67
7.7 Information Model for Push Certificate Management 67
7.7.1 Overview 67
7.7.2 ServerConfiguration 68
7.7.3 ServerConfigurationType 68
7.7.4 UpdateCertificate 70
7.7.5 ApplyChanges 71
7.7.6 CreateSigningRequest 72
7.7.7 GetRejectedList 73
7.7.8 CertificateUpdatedAuditEventType 74
8 KeyCredential management 74
8.1 Overview 74
8.2 Pull management 75
8.3 Push management 76
8.4 Information Model for pull management 76
8.4.1 Overview 76
8.4.2 KeyCredentialManagement 77
8.4.3 KeyCredentialServiceType 77
8.4.4 StartRequest 78
8.4.5 FinishRequest 80
8.4.6 Revoke 81
8.4.7 KeyCredentialAuditEventType 82
8.4.8 KeyCredentialRequestedAuditEventType 82
8.4.9 KeyCredentialDeliveredAuditEventType 83
8.4.10 KeyCredentialRevokedAuditEventType 83
8.5 Information Model for push management 83
8.5.1 General 83
8.5.2 KeyCredentialConfiguration 84
8.5.3 KeyCredentialConfigurationType 84
8.5.4 UpdateCredential 85
8.5.5 DeleteCredential 86
8.5.6 KeyCredentialUpdatedAuditEventType 87
8.5.7 KeyCredentialDeletedAuditEventType 87
9 Authorization Services 88
9.1 Overview 88

Documents
Document Sharing Hub
Full document details: DIN EN IEC 62541-12:2021-08

9.2 Implicit ... 88
9.3 Explicit ... 89
9.4 Chained ... 90
9.5 Information Model for Requesting Access Tokens ... 91
9.5.1 Overview ... 91
9.5.2 AuthorizationServices ... 92
9.5.3 AuthorizationServiceType ... 92
9.5.4 RequestAccessToken ... 93
9.5.5 GetServiceDescription ... 94
9.5.6 AccessTokenIssuedAuditEventType ... 95
9.6 Information Model for configuring Servers ... 95
9.6.1 Overview ... 95
9.6.2 AuthorizationServices ... 96
9.6.3 AuthorizationServiceConfigurationType ... 96
Annex A (informative) Deployment and configuration ... 98
A.1 Firewalls and discovery ... 98
A.2 Resolving references to remote Servers ... 100
Annex B (normative) Constants ... 102
Annex C (normative) OPC UA Mapping to mDNS ... 103
C.1 DNS Server (SRV) record syntax ... 103
C.2 DNS Text (TXT) record syntax ... 103
C.3 DiscoveryUrl mapping ... 104
Annex D (normative) Server Capability Identifiers ... 105
Annex E (normative) DirectoryServices ... 106
E.1 Global Discovery via other directory services ... 106
E.2 UDDI ... 106
E.3 LDAP ... 107
Annex F (normative) Local Discovery Server ... 109
F.1 Certificate store directory layout ... 109
F.2 Installation directories on Windows ... 110
Annex G (normative) Application installation process ... 111
G.1 Provisioning with Pull Management ... 111
G.2 Provisioning with Push Management ... 111
G.3 Setting permissions ... 112
Annex H (informative) Comparison with RFC 7030 ... 113
H.1 Overview ... 113
H.2 Obtaining CA Certificates ... 113
H.3 Initial enrolment ... 113
H.4 Client Certificate reissuance ... 114

Documents
Document Sharing Hub
Full document details: DIN EN IEC 62541-12:2021-08

H.5 Server key generation 114
H.6 Certificate Signing Request (CSR) attributes request 114

Figure 1 – The Registration process with an LDS 23
Figure 2 – The simple Discovery process 24
Figure 3 – The Local Discovery process 25
Figure 4 – The MulticastSubnet Discovery process 26
Figure 5 – The Global Discovery process 27
Figure 6 – The Discovery Process for Clients 27
Figure 7 – The relationship between GDS and other components 29
Figure 8 – The Single MulticastSubnet architecture 30
Figure 9 – The Multiple MulticastSubnet architecture 31
Figure 10 – The No MulticastSubnet architecture 32
Figure 11 – The Address Space for the GDS 33
Figure 12 – The Pull Certificate management model 44
Figure 13 – The Push Certificate management model 45
Figure 14 – The Certificate Management AddressSpace for the GlobalDiscoveryServer 57
Figure 15 – The AddressSpace for the Server that supports Push Management 68
Figure 16 – The Pull Model for KeyCredential management 75
Figure 17 – The Push Model for KeyCredential management 76
Figure 18 – The Address Space used for Pull KeyCredential management 77
Figure 19 – The AddressSpace used for Push KeyCredential management 84
Figure 20 – Roles and Authorization Services 88
Figure 21 – Implicit authorization 89
Figure 22 – Explicit authorization 90
Figure 23 – Chained authorization 91
Figure 24 – The Model for Requesting Access Tokens from Authorization Services 92
Figure 25 – The Model for configuring Servers to use Authorization Services 96
Figure A.1 – Discovering Servers outside a firewall 98
Figure A.2 – Discovering Servers behind a firewall 99
Figure A.3 – Using a Discovery Server with a firewall 100
Figure A.4 – Following References to Remote Servers 101
Figure E.1 – The UDDI or LDAP Discovery process 106
Figure E.2 – UDDI registry structure 107
Figure E.3 – Sample LDAP hierarchy 108

Table 1 – GDS NamespaceMetadataType Object definition 21
Table 2 – Directory Object definition 33
Table 3 – DirectoryType definition 34

AVA Documents
Document Sharing Hub

Full document details: DIN EN IEC 62541-12:2021-08

Table 4 – FindApplications Method AddressSpace definition 35
Table 5 – ApplicationRecordDataType definition 36
Table 6 – RegisterApplication Method AddressSpace definition 37
Table 7 – UpdateApplication Method AddressSpace definition 38
Table 8 – UnregisterApplication Method AddressSpace definition 39
Table 9 – GetApplication Method AddressSpace definition 39
Table 10 – QueryApplications Method AddressSpace definition 41
Table 11 – QueryServers Method AddressSpace definition 43
Table 12 – ApplicationRegistrationChangedAuditEventType definition 43
Table 13 – TrustListType definition 47
Table 14 – OpenWithMasks Method AddressSpace definition 48
Table 15 – CloseAndUpdate Method AddressSpace definition 49
Table 16 – AddCertificate Method AddressSpace definition 50
Table 17 – RemoveCertificate Method AddressSpace definition 51
Table 18 – TrustListDataType definition 51
Table 19 – TrustListMasks values 51
Table 20 – TrustListOutOfDateAlarmType definition 52
Table 21 – CertificateGroupType definition 52
Table 22 – CertificateType definition 53
Table 23 – ApplicationCertificateType definition 54
Table 24 – HttpsCertificateType definition 54
Table 25 – UserCredentialCertificateType definition 54
Table 26 – RsaMinApplicationCertificateType definition 55
Table 27 – RsaSha256ApplicationCertificateType definition 55
Table 28 – CertificateGroupFolderType definition 55
Table 29 – TrustListUpdatedAuditEventType definition 56
Table 30 – CertificateDirectoryType ObjectType definition 57
Table 31 – StartSigningRequest Method AddressSpace definition 59
Table 32 – StartNewKeyPairRequest Method AddressSpace definition 62
Table 33 – FinishRequest Method AddressSpace definition 63
Table 34 – GetCertificateGroups Method AddressSpace definition 64
Table 35 – GetTrustList Method AddressSpace definition 65
Table 36 – GetCertificateStatus Method AddressSpace definition 66
Table 37 – CertificateRequestedAuditEventType definition 66
Table 38 – CertificateDeliveredAuditEventType definition 67
Table 39 – ServerConfiguration Object definition 68
Table 40 – ServerConfigurationType definition 69
Table 41 – UpdateCertificate Method AddressSpace Definition 71
Table 42 – ApplyChanges Method AddressSpace Definition 72

Documents
Document Sharing Hub

Full document details: DIN EN IEC 62541-12:2021-08

Table 43 – CreateSigningRequest Method AddressSpace definition 73
Table 44 – GetRejectedList Method AddressSpace definition 74
Table 45 – CertificateUpdatedAuditEventType definition 74
Table 46 – KeyCredentialManagement Object definition 77
Table 47 – KeyCredentialServiceType definition 78
Table 48 – StartRequest Method AddressSpace definition 79
Table 49 – FinishRequest Method AddressSpace definition 81
Table 50 – Revoke Method AddressSpace definition 81
Table 51 – KeyCredentialAuditEventType definition 82
Table 52 – KeyCredentialRequestedAuditEventType definition 82
Table 53 – KeyCredentialDeliveredAuditEventType definition 83
Table 54 – KeyCredentialRevokedAuditEventType definition 83
Table 55 – KeyCredentialConfiguration Object definition 84
Table 56 – KeyCredentialConfigurationType definition 84
Table 57 – UpdateCredential Method AddressSpace definition 86
Table 58 – DeleteCredential Method AddressSpace definition 87
Table 59 – KeyCredentialUpdatedAuditEventType definition 87
Table 60 – KeyCredentialUpdatedAuditEventType definition 87
Table 61 – AuthorizationServices Object definition 92
Table 62 – AuthorizationServiceType definition 92
Table 63 – RequestAccessToken Method AddressSpace definition 94
Table 64 – GetServiceDescription Method AddressSpace definition 95
Table 65 – AccessTokenIssuedAuditEventType definition 95
Table 66 – AuthorizationServices Object definition 96
Table 67 – AuthorizationServiceConfigurationType definition 96
Table C.1 – Allowed mDNS service names 103
Table C.2 – DNS TXT record string format 104
Table C.3 – DiscoveryUrl to DNS SRV and TXT Record Mapping 104
Table D.1 – Examples of *ServerCapabilityIdentifiers* 105
Table E.1 – UDDI tModels 107
Table E.2 – LDAP object class schema 108
Table F.1 – Application Certificate store directory layout 109
Table H.1 – Verifying that a Server is allowed to provide Certificates 113
Table H.2 – Verifying that a Client is allowed to request Certificates 113

AVA Documents
Document Sharing Hub

Full document details: DIN EN IEC 62541-12:2021-08

OPC UNIFIED ARCHITECTURE –

Part 12: Discovery and global services

1 Scope

This part of IEC 62541 specifies how OPC Unified Architecture (OPC UA) *Clients* and *Servers* interact with *DiscoveryServers* when used in different scenarios. It specifies the requirements for the *LocalDiscoveryServer, LocalDiscoveryServer-ME and GlobalDiscoveryServer.* It also defines information models for *Certificate* management*, KeyCredential m*anagement and *Authorization Services*.

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 62541-7, *OPC Unified Architecture – Part 7: Profiles*

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

IEC 62541-14, *OPC Unified Architecture – Part 14: PubSub*

X.500: ISO/IEC 9594-1:2017, *Information technology – Open Systems Interconnection – The Directory – Part 1: Overview of concepts, models and services*

IETF RFC 1035, *DNS-Name: Domain Names – Implementation and Specification*
http://www.ietf.org/rfc/rfc1035.txt

IETF RFC 2986, *PKCS #10: Certification Request Syntax Specification*
http://www.ietf.org/rfc/rfc2986.txt

IETF RFC 3927, *Auto-IP: Dynamic Configuration of IPv4 Link-Local Addresses*
http://www.ietf.org/rfc/rfc3927.txt

IETF RFC 5958, *Asymmetric Key Packages*
http://www.ietf.org/rfc/rfc5958.txt

IETF RFC 6762, *mDNS: Multicast DNS*
http://www.ietf.org/rfc/rfc6762.txt

IETF RFC 6763, *DNS-SD: DNS Based Service Discovery*
http://www.ietf.org/rfc/rfc6763.txt

Documents
Document Sharing Hub

Full document details: DIN EN IEC 62541-12:2021-08

IETF RFC 7030: *Enrollment over Secure Transport*
http://www.ietf.org/rfc/rfc7030.txt

PKCS #12: *Personal Information Exchange Syntax*
http://www.emc.com/emc-plus/rsa-labs/pkcs/files/h11301-wp-pkcs-12v1-1-personal-information-exchange-syntax.pdf

DI: *OPC Unified Architecture for Devices (DI)*
https://opcfoundation.org/developer-tools/specifications-unified-architecture/opc-unified-architecture-for-devices-di/

ADI: *OPC Unified Architecture for Analyzer Devices (ADI)*
https://opcfoundation.org/developer-tools/specifications-unified-architecture/opc-unified-architecture-for-analyzer-devices-adi/

PLCopen: *OPC Unified Architecture / PLCopen Information Model*
https://opcfoundation.org/developer-tools/specifications-unified-architecture/opc-unified-architecture-plcopen-information-model/

FDI: *OPC Unified Architecture for FDI*
https://opcfoundation.org/developer-tools/specifications-unified-architecture/opc-unified-architecture-for-fdi/

ISA-95: *ISA-95 Common Object Model*
https://opcfoundation.org/developer-tools/specifications-unified-architecture/isa-95-common-object-model/

3 Terms, definitions, abbreviated terms and conventions

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 62541-9 and the following apply.

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

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

3.1.1
Certificate Group
context used to describe the *Trust List* and *Certificate(s)* associated with an *Application*

3.1.2
Certificate Request
PKCS #10 encoded structure used to request a new *Certificate* from a *Certificate Authority*

3.1.3
KeyCredential
unique identifier and a secret used to access a *Server*, an *Authorization Service* or a *Broker*

Note 1 to entry: A user name and password is an example of a credential.

3.1.4
KeyCredentialService
software application that provides *KeyCredentials* needed to access a *Server*, an *Authorization Service* or a *Broker*

Documents
Document Sharing Hub
Full document details: DIN EN IEC 62541-12:2021-08

3.1.5
DirectoryService
software application, or a set of applications, that stores and organizes information about resources such as computers or services

3.1.6
DiscoveryServer
Application that maintains a list of OPC UA *Servers* that are available on the network and provides mechanisms for Clients to obtain this list

3.1.7
DiscoveryUrl
URL for a network *Endpoint* that provides the information required to connect to a *Client* or *Server*

3.1.8
GlobalDiscoveryServer
GDS
DiscoveryServer that maintains a list of OPC UA *Applications* available in an administrative domain

Note 1 to entry: A GDS can also provide certificate management services.

3.1.9
IPAddress
unique number assigned to a network interface that allows Internet Protocol (IP) requests to be routed to that interface

Note 1 to entry: An *IPAddress* for a host can change over time.

3.1.10
LocalDiscoveryServer
LDS
DiscoveryServer that maintains a list of all *Servers* that have registered with it

Note 1 to entry: *Servers* normally register with the LDS on the same host.

3.1.11
LocalDiscoveryServer-ME
LDS-ME
LocalDiscoveryServer that includes the MulticastExtension

3.1.12
MulticastExtension
extension to a *LocalDiscoveryServer* that adds support for the *mDNS* protocol

3.1.13
MulticastSubnet
network that allows multicast packets to be sent to all nodes connected to the network

Note 1 to entry: A *MulticastSubnet* is not necessarily the same as a TCP/IP subnet.

3.1.14
Network Service
secured resource on a network that provides functionality used by *Clients* and/or *Servers*

Note 1 to entry: An Authorization Service (AS) is an example of a Network Service.

3.1.15
ServerCapabilityIdentifier
short identifier which uniquely identifies a set of discoverable capabilities supported by a *Server*

Note 1 to entry: The list of the currently defined *ServerCapabilityIdentifiers* is in Annex D.

Documents
Document Sharing Hub

Full document details: DIN EN IEC 62541-12:2021-08

3.2 Abbreviated terms and symbols

API	application programming interface
CA	certificate authority
CRL	certificate revocation list
CSR	certificate signing request
DER	distinguished encoding rules
DNS	Domain Name System
EST	Enrolment over Secure Transport
GDS	Global Discovery Server
IANA	Internet Assigned Numbers Authority
LDAP	Lightweight Directory Access Protocol
LDS	Local Discovery Server
LDS-ME	Local Discovery Server with Multicast Extension
mDNS	Multicast Domain Name System
NAT	network address translation
PEM	Privacy-Enhanced Mail
PFX	Personal Information Exchange
PKCS	Public-Key Cryptography Standards
SHA1	Secure Hash Algorithm
SSL	Secure Sockets Layer
TLS	Transport Layer Security
UA	Unified Architecture
UDDI	Universal Description, Discovery And Integration

3.3 Conventions for namespaces

This document uses multiple namespaces to define *Nodes*. The following abbreviated terms are used in the definitions for these *Nodes*:

CORE	http://opcfoundation.org/UA/
GDS	http://opcfoundation.org/UA/GDS/

The default namespace for each *Node* is defined at the top of the table. All of the *BrowseNames* in the table use the default namespace unless the *BrowseName* is preceded by one of the above abbreviations.

The *NamespaceMetadataType Object* for the GDS namespace is defined in Table 1.

Table 1 – GDS NamespaceMetadataType Object definition

Attribute	**Value**		
BrowseName	http://opcfoundation.org/UA/GDS/		
Namespace	GDS (see 3.3)		
TypeDefinition	NamespaceMetadataType defined in IEC 62541-5.		
References	**NodeClass**	**BrowseName**	**Value**
HasProperty	Variable	NamespaceUri	http://opcfoundation.org/UA/GDS/

Documents
Document Sharing Hub
Full document details: DIN EN IEC 62541-12:2021-08

Attribute	Value		
HasProperty	Variable	NamespaceVersion	1.04
HasProperty	Variable	NamespacePublicationDate	2016-12-31

4 The discovery process

4.1 Overview

The discovery process allows applications to find other applications on the network and then discover how to connect to them. Note that this discussion builds on the discovery related concepts defined in IEC 62541-4. Discoverable applications are generally *Servers*, however, some *Clients* will support reverse connections as described in IEC 62541-6 and want *Servers* to be able to discover them.

Clients and *Servers* can be on the same host, on different hosts in the same subnet, or even on completely different locations in an administrative domain. The following clauses describe the different configurations and how discovery can be accomplished.

The mechanisms for *Clients* to discover *Servers* are specified in 4.3.

The mechanisms for *Servers* to make themselves discoverable are specified in 4.2.

The *Discovery Services* are specified in IEC 62541-4. They are implemented by individual *Servers* and by dedicated *DiscoveryServers*. The following dedicated *DiscoveryServers* provide a way for applications to discover registered OPC UA applications in different situations:

- a *LocalDiscoveryServer* (LDS) maintains discovery information for all applications that have registered with it, usually all applications available on the host that it runs on;
- a *LocalDiscoveryServer* with the *MulticastExtension* (LDS-ME) maintains discovery information for all applications that have been announced on the local *MulticastSubnet*;
- a *GlobalDiscoveryServer* (GDS) maintains discovery information for applications available in an administrative domain.

LDS and LDS-ME are specified in Clause 5. The GDS is specified in Clause 6.

Annex A provides guidelines on the use of firewalls in discovery. Annex B defines the Global Discovery NodeSet. Annex F defines a directory layout for applications to store their *Certificates* on a file system. Annex G provides instructions for the application installation procedure. Annex H describes how Certificate management described in this document compares to RFC7030.

4.2 Registration and announcement of Applications

4.2.1 Overview

This subclause describes how an application registers itself so it can be discovered. Most *Applications* will want other applications to discover them. *Applications* that do not wish to be discovered openly should not register with a *DiscoveryServer*. In this case, such *Applications* should only publish a *DiscoveryUrl* via some out-of-band mechanism to be discovered by specific *Applications*.

4.2.2 Hosts with a LocalDiscoveryServer

Applications register themselves with the LDS on the same host if they wish to be discovered. The registration ensures that the applications is visible for local discovery (see 4.3.4) and *MulticastSubnet* discovery if the LDS is an LDS-ME (see 4.3.5).

Documents
Document Sharing Hub
Full document details: DIN EN IEC 62541-12:2021-08

The OPC UA Standard (IEC 62541-4) defines a *RegisterServer2 Service*, which provides additional registration information. All *Applications* and the *LocalDiscoveryServer* shall support the *RegisterServer2 Service* and, for backwards compatibility, the older *RegisterServer Service*. If an *Application* encounters an older LDS that returns a *Bad_ServiceUnsupported* error when calling *RegisterServer2 Service*, it shall try again with *RegisterServer Service*.

The *RegisterServer2 Service* allows the *Application* to specify zero or more *ServerCapability Identifiers*. *ServerCapabilityIdentifiers* are short, string identifiers of well-known OPC UA features. *Applications* can use these identifiers as a filter during discovery.

The set of known *ServerCapabilityIdentifiers* is specified in Annex D and is limited to features that are considered to be important enough to report before an application makes a connection. For example, support for the GDS information model or the Alarms information model are *Server* capabilities that have a *ServerCapabilityIdentifier* defined.

Before an application registers with the LDS, it should call the *GetEndpoints Service* and choose the most secure endpoint supported by the LDS and then call *RegisterServer2* or *RegisterServer*.

Registration with LDS or LDS-ME is illustrated in Figure 1.

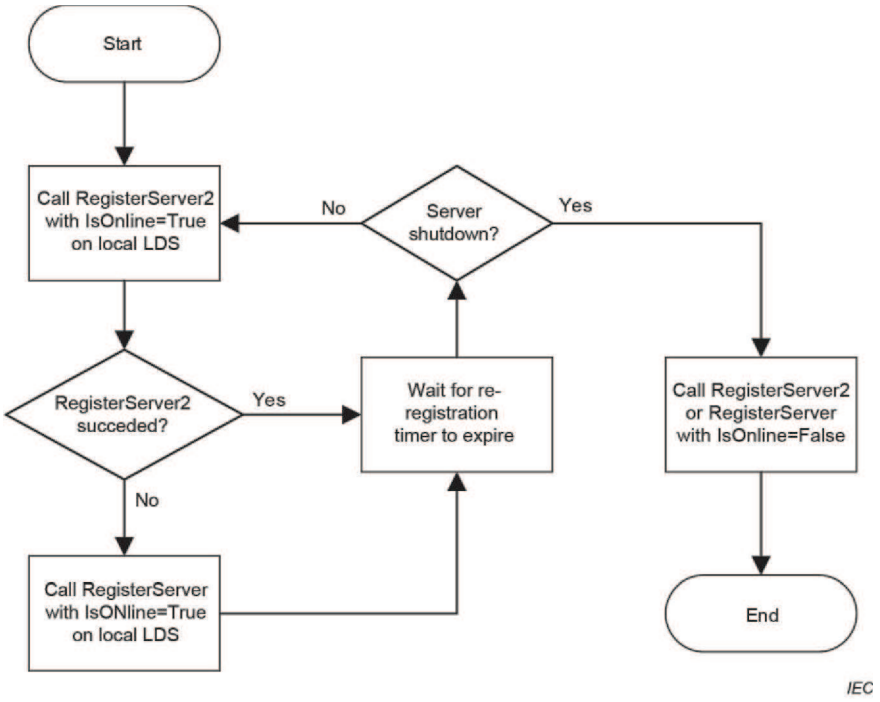

Figure 1 – The Registration process with an LDS

See IEC 62541-4 for more information on the re-registration timer and the *IsOnline* flag.

4.2.3 Hosts without a LocalDiscoveryServer

Dedicated systems (usually embedded systems) with exactly one *Server* installed may not have a separate LDS. Such *Servers* shall become their own LDS or LDS-ME by implementing *FindServers* and *GetEndpoints Services* at the well-known address for an LDS. They should also announce themselves on the *MulticastSubnet* with a basic *MulticastExtension*. This requires a small subset of an mDNS Responder (see *mDNS* and Annex C) that announces the *Server* and responds to mDNS probes. The *Server* may not provide the caching and address resolution implemented by a full mDNS Responder.

Documents
Document Sharing Hub
Full document details: DIN EN IEC 62541-12:2021-08

4.3 The discovery process for Clients to find Servers

4.3.1 Overview

The discovery process allows *Clients* to find *Servers* on the network and then discover how to connect to them. Once a *Client* has this information, it can save it and use it to connect directly to the *Server* again without going through the discovery process. *Clients* that cannot connect with the saved connection information should assume the *Server* configuration has changed and therefore repeat the discovery process.

A *Client* has several choices for finding *Servers*:

- out-of-band discovery (i.e. entry into a GUI) of a *DiscoveryUrl* for a *Server*;
- calling *FindServers* on the LDS installed on the *Client* host;
- calling *FindServers* on a remote LDS, where the *HostName* for the remote host is manually entered*;*
- calling *FindServersOnNetwork* (see IEC 62541-4) on the LDS-ME installed on *Client* host;
- supporting the LDS-ME functionality locally in the Client;
- searching for *Servers* known to a *GlobalDiscoveryServer*.

The *DiscoveryUrl* provides all of the information a *Client* needs to connect to a *DiscoveryEndpoint* (see 4.3.3).

4.3.2 Security

Clients should be aware of rogue *DiscoveryServers* that might direct them to rogue *Servers*. *Clients* can use the SSL/TLS server certificate (if available) to verify that the *DiscoveryServer* is a server that they trust and/or ensure that they trust any *Server* provided by the *DiscoveryServer*. See IEC TR 62541-2 for a detailed discussion of these issues.

4.3.3 Simple Discovery with a DiscoveryUrl

Every *Server* has one or more *DiscoveryUrls* that allow access to its *Endpoints*. Once a *Client* obtains (e.g. via manual entry into a form) the *DiscoveryUrl* for the *Server,* it reads the *EndpointDescriptions* using the *GetEndpoints Service* defined in IEC 62541-4.

The discovery process for this scenario is illustrated in Figure 2.

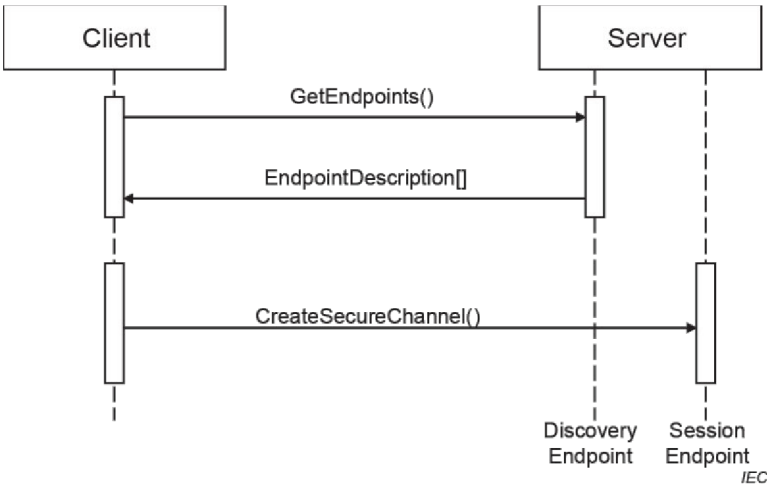

Figure 2 – The simple Discovery process

Documents
Document Sharing Hub
Full document details: DIN EN IEC 62541-12:2021-08

4.3.4 Local Discovery

In many cases, *Clients* do not know which *Servers* exist but possibly know which hosts can have *Servers* on them. In this situation, the *Client* will look for the *LocalDiscoveryServer* on a host by constructing a *DiscoveryUrl* using the well-known addresses defined in IEC 62541-6.

If a *Client* finds a *LocalDiscoveryServer* then it will call the *FindServers Service* on the LDS to obtain a list of *Servers* and their *DiscoveryUrls*. The *Client* would then call the *GetEndpoints* service for one of the *Servers* returned. The discovery process for this scenario is illustrated in Figure 3.

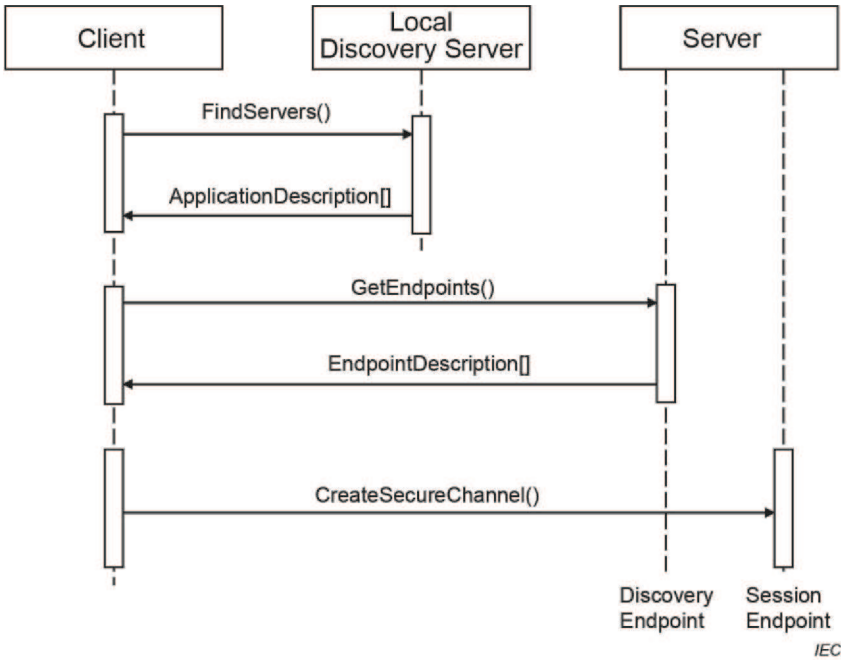

Figure 3 – The Local Discovery process

4.3.5 MulticastSubnet Discovery

In some situations *Clients* will not know which hosts have *Servers*. In these situations the *Client* will look for a *LocalDiscoveryServer* with the *MulticastExtension* on its local host and requests a list of *DiscoveryUrls* for *Servers* and *DiscoveryServers* available on the *MulticastSubnet*.

The discovery process for this scenario is illustrated in Figure 4.

Documents
Document Sharing Hub
Full document details: DIN EN IEC 62541-12:2021-08

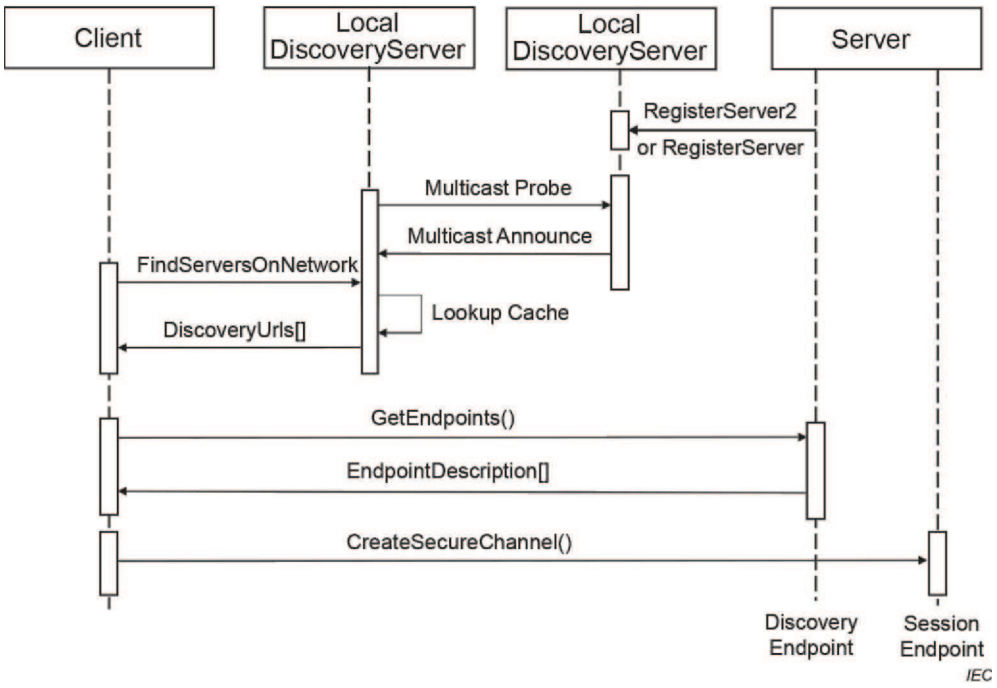

Figure 4 – The MulticastSubnet Discovery process

In this scenario, the *Server* uses the *RegisterServer2 Service* to tell a *LocalDiscoveryServer* to announce the *Server* on the *MulticastSubnet*. The *Client* will receive the *DiscoveryUrl* and *ServerCapabilityIdentifiers* for the Server when it calls *FindServersOnNetwork* and then connects directly to the *Server*. When a *Client* calls *FindServers* it only receives the *Servers* running on the same host as the LDS.

Clients running on embedded systems may not have a LDS-ME available on the system. These *Clients* can support an mDNS Responder that understands how OPC UA concepts are mapped to mDNS messages and maintains the same table of *Servers* as maintained by the LDS-ME. This mapping is described in Annex C.

4.3.6 Global Discovery

A GDS is an OPC UA *Server* that allows *Clients* to search for *Servers* in the administrative domain. It may also provide Certificate Services (see Clause 7). It provides *Methods* that allow applications to search for other applications (See Clause 6). To access the GDS, the *Client* will create a *Session* with the *GDS* and use the *Call* service to invoke the *QueryApplications Method* (see 6.3.11). The *QueryServers Method* is similar to the *FindServers* service except that it provides more advanced search and filter criteria. The discovery process is illustrated in Figure 5.

Documents
Document Sharing Hub
Full document details: DIN EN IEC 62541-12:2021-08

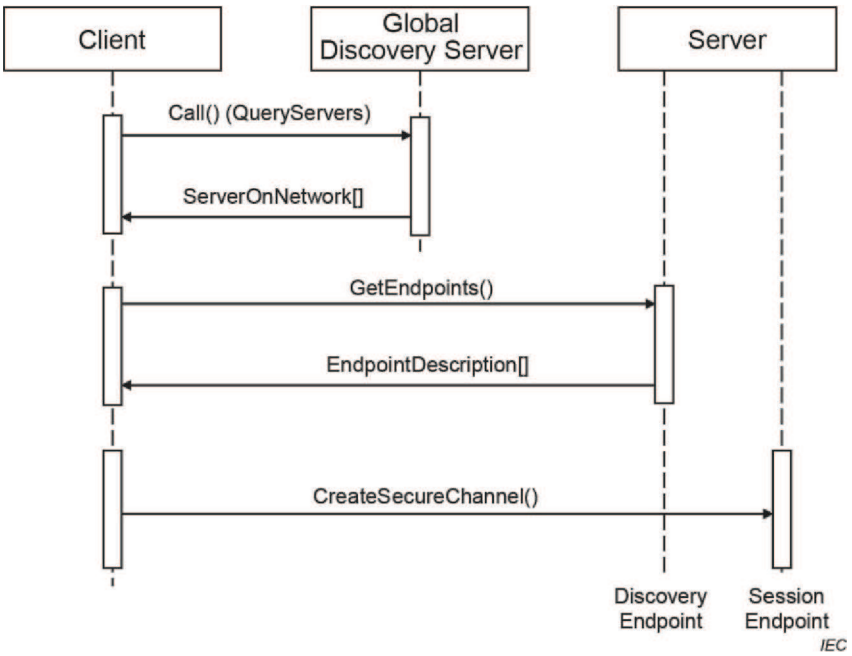

Figure 5 – The Global Discovery process

The GDS may be coupled with any of the previous network architectures. For each *MulticastSubnet,* one or more LDSs may be registered with a GDS.

The *Client* can also be configured with the URL of the GDS using an out of band mechanism.

The complete discovery process is shown in Figure 6.

4.3.7 Combined Discovery Process for Clients

The use cases in the preceding clauses imply a number of choices that have to be made by *Clients* when a *Client* needs to connect to a *Server*. These choices are combined together in Figure 6.

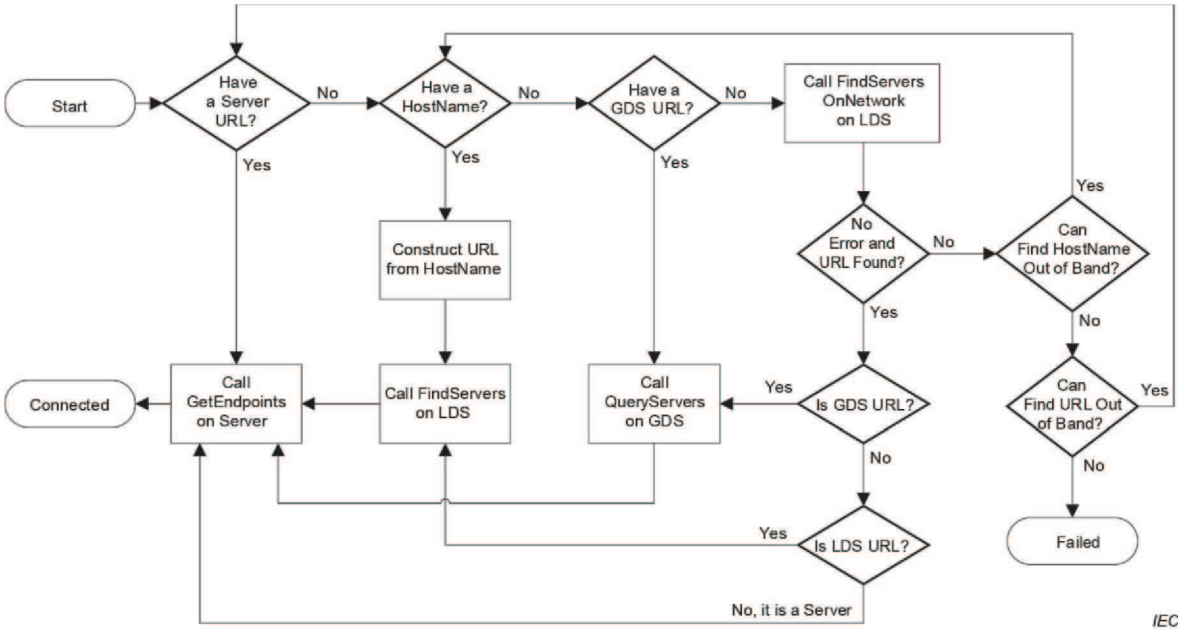

Figure 6 – The Discovery Process for Clients

Documents
Document Sharing Hub
Full document details: DIN EN IEC 62541-12:2021-08

FindServersOnNetwork can be called on the local LDS, however, it can also be called on a remote LDS which is part of a different *MulticastSubnet*.

An out-of-band mechanism is a way to find a URL or a *HostName* that is not described by this document. For example, a user could manually enter a URL or use system specific APIs to browse the network neighbourhood.

A *Client* that goes through the discovery process can save the URL that was discovered. If the *Client* restarts later, it can use that URL and bypass the discovery process. If reconnection fails, the *Client* will have to go through the process again.

5 Local Discovery Server

5.1 Overview

Each host that could have multiple discoverable applications installed should have a standalone *LocalDiscoveryServer* installed. The *LocalDiscoveryServer* shall expose one or more *Endpoints* which support the *FindServers* and *GetEndpoints* services defined in IEC 62541-4 for all applications on the host. In addition, the *LocalDiscoveryServer* shall provide at least one *Endpoint* which implements the *RegisterServer* service for these applications.

In systems (usually embedded systems) with exactly one *Server* installed this *Server* may also be the LDS (see 4.2.3).

An LDS-ME will announce all applications that it knows about on the local *MulticastSubnet*. In order to support this, a *LocalDiscoveryServer* supports the *RegisterServer2 Service* defined in IEC 62541-4. For backward compatibility, a *LocalDiscoveryServer* also supports the *RegisterServer Service* which is defined in IEC 62541-4.

Each host with OPC UA Applications (Clients and Servers) installed should have a *LocalDiscoveryServer* with a *MulticastExtension*.

The *MulticastExtension* incorporates the functionality of the mDNS Responder described in the Multicast DNS (mDNS) specification (see *mDNS*). In addition, the *LocalDiscoveryServer* that supports the *MulticastExtension* supports the *FindServersOnNetwork Service* described in IEC 62541-4.

5.2 Security considerations for Multicast DNS

The Multicast DNS (mDNS) specification is used for various commercial and consumer applications. This provides a benefit in that implementations exist; however, system administrators could choose to disable Multicast DNS operations. For this reason, *Applications* shall not rely on Multicast DNS capabilties.

Multicast DNS operations are insecure because of their nature; therefore, they should be disabled in environments where an attacker could cause problems by impersonating another host. This risk is minimized if OPC UA security is enabled and all *Applications* use *Certificate TrustLists* to control access.

6 Global Discovery Server

6.1 Overview

The *LocalDiscoveryServer* is useful for networks where the host names can be discovered. However, this is typically not the case in large systems with multiple servers on multiple subnets. For this reason, there is a need for an enterprise wide *DiscoveryServer* called a *GlobalDiscoveryServer*. The *GlobalDiscoveryServer* (GDS) is an OPC UA *Server* that allows *Clients* to search for *Servers* in the administrative domain. It provides methods that allow applications to register themselves and to search for other applications.

AVA Documents
Document Sharing Hub

Full document details: DIN EN IEC 62541-12:2021-08

The essential element of a *GlobalDiscoveryServer* (GDS) is that it can provide the *Certificate* management services defined in Clause 7. These services can simplify *Certificate* management even in medium to small systems; therefore, a GDS can be deployed in smaller systems. Different implementations are expected. Some of them will likely provide a front-end to an existing *DirectoryService* such as LDAP (see Annex E). By standardizing on an OPC UA based interface, OPC UA *Clients* do not need to have knowledge of different *DirectoryServices.*

If an administrator registers a *LocalDiscoveryServer* with the GDS, then the GDS shall periodically update its database by calling *FindServersOnNetwork* or *FindServers* on the LDS. Figure 7 shows the relationship between a GDS and the LDS-ME or LDS.

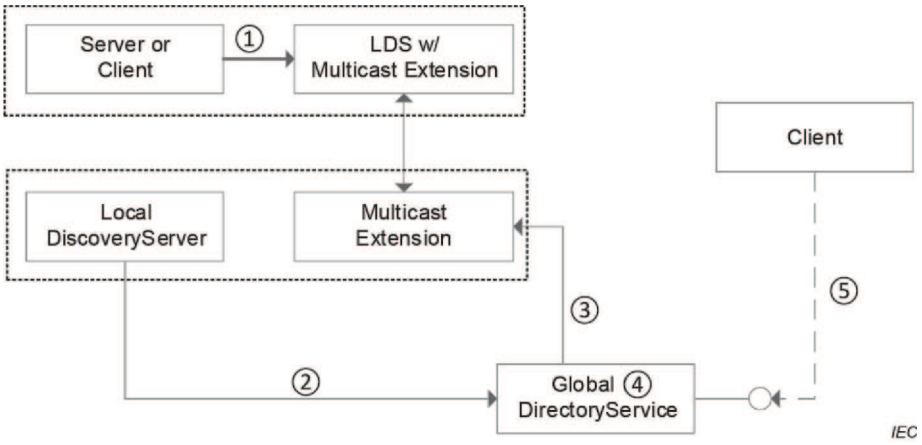

Figure 7 – The relationship between GDS and other components

The steps shown in Figure 7 are:

1) The Server calls *RegisterServer2* on the LDS running on the same machine.
2) The administrator registers LDS-ME installations with the GDS.
3) The GDS calls *FindServersOnNetwork* on the LDS-ME to find all applications on the same *MulticastSubnet*.
4) The GDS creates a record for each application returned by the LDS-ME. These records shall be approved before they are made available to *Clients* of the GDS. This approval can be obtained from an *Administrator or* the GDS can connect to the *Server* and verifies that it has a trusted *Certfiicate.*
5) The *Client* calls *QueryServers Method* on the GDS to discover applications.

The *Information Model* used for registration and discovery is shown in 6.2. Any *Client* shall be able to call the *QueryServers Method* to find applications known to GDS. The complete definitions for each of the types used are described in 7.5.

6.2 Network architectures

6.2.1 Overview

The discovery mechanisms defined in this document are expected to be used in many different network architectures. The following three architectures are illustrated:

- Single *MulticastSubnet*;
- Multiple *MulticastSubnets*;
- No *MulticastSubnet* (or multiple *MulticastSubnets* with exactly one host each).

A *MulticastSubnet* is a network segment where all hosts on the segment can receive multicast packets from the other hosts on the segment. A physical LAN segment is typically a *MulticastSubnet* unless the administrator

Documents
Document Sharing Hub
Full document details: DIN EN IEC 62541-12:2021-08

has specifically disabled multicast communication. In some cases, multiple physical LAN segments can be connected as a Single *MulticastSubnet.*

6.2.2 Single MulticastSubnet

The Single *MulticastSubnet* Architecture is shown in Figure 8.

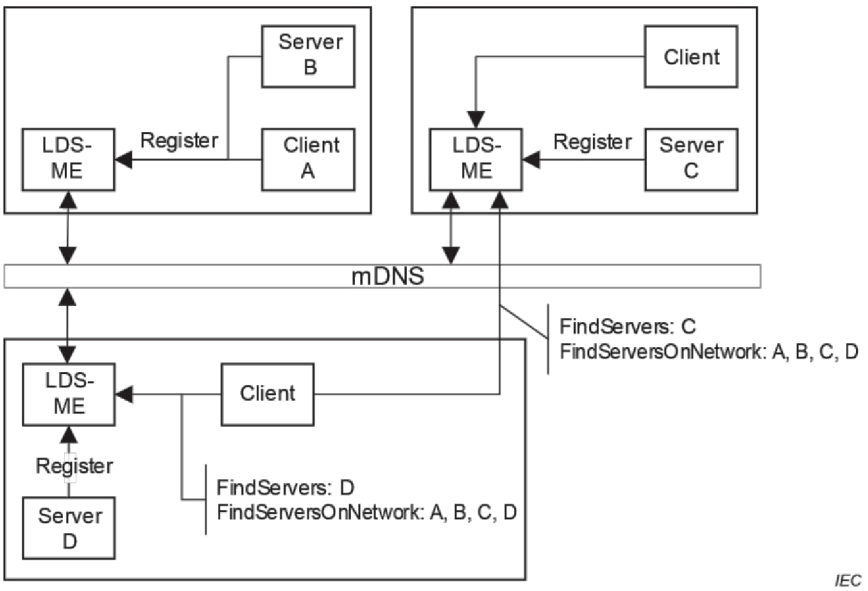

Figure 8 – The Single MulticastSubnet architecture

In this architecture, every host has an LDS-ME and uses mDNS to maintain a cache of the applications on the *MulticastSubnet*. A *Client* can call *FindServersOnNetwork* on any LDS-ME and receive the same set of applications. When a *Client* calls *FindServers*, it only receives the applications running on the same host as the LDS.

Documents
Document Sharing Hub
Full document details: DIN EN IEC 62541-12:2021-08

6.2.3 Multiple MulticastSubnet

The Multiple *MulticastSubnet* Architecture is shown in Figure 9.

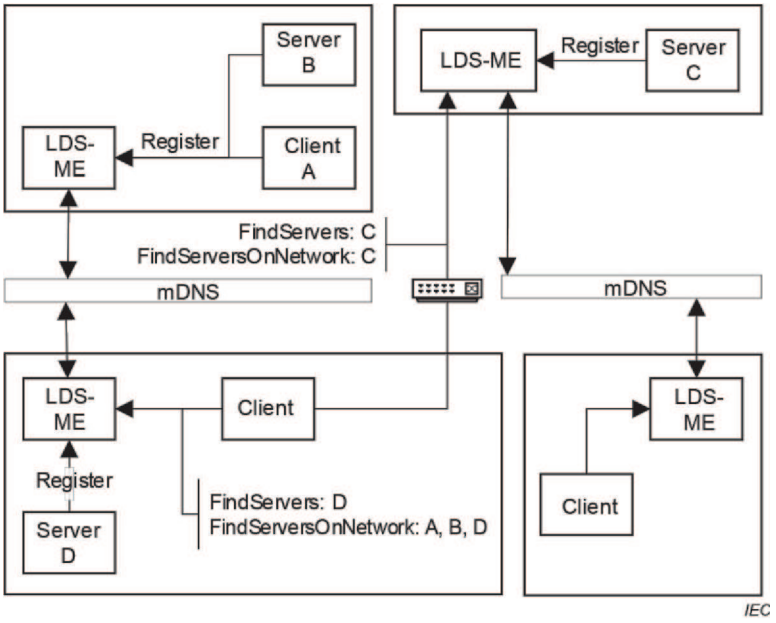

Figure 9 – The Multiple MulticastSubnet architecture

This architecture is the same as the previous architecture, except in this architecture the mDNS messages do not pass through routers connecting the *MulticastSubnets*. This means that a *Client* calling *FindServersOnNetwork* will only receive a list of applications running on the *MulticastSubnets* that the LDS-ME is connected to.

A *Client* that wants to connect to a remote *MulticastSubnet* shall use out of band discovery (i.e. manual entry) of a *HostName* or *DiscoveryUrl*. Once a *Client* finds an LDS-ME on a remote *MulticastSubnet*, it can use *FindServersOnNetwork* to discover all applications on that *MulticastSubnet.*

Documents
Document Sharing Hub

6.2.4 No MulticastSubnet

The No *MulticastSubnet* Architecture is shown in Figure 10.

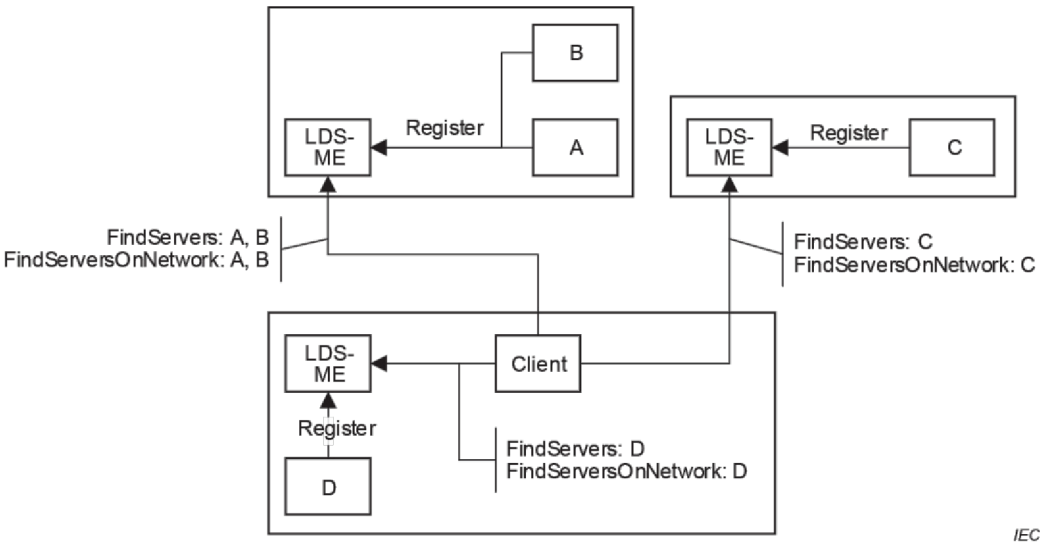

Figure 10 – The No MulticastSubnet architecture

In this architecture, the mDNS is not used at all because the Administrator has disabled multicast at a network level or by turning off multicast capabilities of each LDS-ME.

A *Client* that wants to discover a applications needs to use an out of band mechanism to find the *HostName* and call *FindServers* on the LDS of that host. *FindServersOnNetwork* can also work, but it will never return more than what *FindServers* returns.

6.2.5 Domain Names and MulticastSubnets

The mDNS specification requires that a fully qualified domain name be annouced on the network. If a *Server* is not configured with a fully qualified domain name, then mDNS requires that the "local" top level domain be appended to the domain names. The "local" top level domain indicates that the domain can only be consided to be unique within the subnet where the domain name was used. This means *Clients* need to be aware that URLs received from any LDS-ME other than the one on the *Client's* machine could contain "local" domains that are not reachable or will connect to a different machine with the same domain name that happens to be on the same subnet as the *Client*. It is recommended that *Clients* ignore all URLs with the "local" top level domain unless they are returned from the LDS-ME running on the same machine.

System administrators can eliminate this problem by configuring a normal DNS with the fully qualilfied domain names for all machines that need to be accessed by *Clients* outside the *MulticastSubnet*.

Servers configured with fully qualified domain names should specify the fully qualified domain name in its *ApplicationInstance Certificate*. *Servers* shall not specify domains with the "local" top level domain in their *Certificate*. *Clients* using a URL returned from an LDS-ME shall ignore the "local" top level domain when checking the domain against the *Server Certificate*.

Note that domain name validation is a necessary but not sufficient check against rogue *Servers* or man-in-the-middle attacks when *Server Certificates* do not contain fully qualified domain names. The *Certificate* trust relationship established by administrators is the primary mechanism used to protect against these risks.

Documents
Document Sharing Hub
Full document details: DIN EN IEC 62541-12:2021-08

6.3 Information Model

6.3.1 Overview

The *GlobalDiscoveryServer Information Model* used for *discovery* is shown in Figure 11. Most of the interactions between the *GlobalDiscoveryServer* and *Application* administrator or the *Client* will be via *Methods* defined on the *Directory* folder.

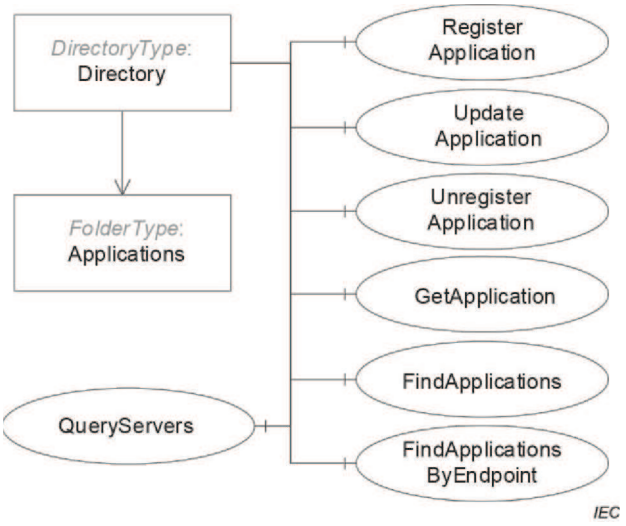

Figure 11 – The Address Space for the GDS

6.3.2 Directory

This *Object* is the root of the *GlobalDiscoveryServer AddressSpace*, and it is the target of an *Organizes* reference from the *Objects* folder defined in IEC 62541-5. It organizes the information that can be accessed into subfolders. The implementation of a GDS can customize and organize the folders in any manner it desires. For example, folders can exist for information models, or for optional services or for various locations in an administrative domain. It is defined in Table 2.

Table 2 – Directory Object definition

Attribute	Value				
BrowseName	Directory				
Namespace	GDS (see 3.3)				
TypeDefinition	DirectoryType defined in 6.3.3.				
References	**NodeClass**	**BrowseName**	**DataType**	**TypeDefinition**	**Modelling Rule**

6.3.3 DirectoryType

DirectoryType is the *ObjectType* for the root of the *GlobalDiscoveryServer AddressSpace*. It organizes the information that can be accessed into subfolders. It also provides methods that allow applications to register or find applications. It is defined in Table 3.

Documents
Document Sharing Hub
Full document details: DIN EN IEC 62541-12:2021-08

Table 3 – DirectoryType definition

Attribute	Value				
BrowseName	DirectoryType				
Namespace	GDS (see 3.3)				
IsAbstract	False				
References	**NodeClass**	**BrowseName**	**DataType**	**TypeDefinition**	**Modelling Rule**
Subtype of the *FolderType* defined in IEC 62541-5.					
Organizes	Object	Applications	-	FolderType	Mandatory
HasComponent	Method	FindApplications	Defined in 6.3.4		Mandatory
HasComponent	Method	RegisterApplication	Defined in 6.3.6		Mandatory
HasComponent	Method	UpdateApplication	Defined in 6.3.7		Mandatory
HasComponent	Method	UnregisterApplication	Defined in 6.3.8		Mandatory
HasComponent	Method	GetApplication	Defined in 6.3.9		Mandatory
HasComponent	Method	QueryApplications	Defined in 6.3.10		Mandatory
HasComponent	Method	QueryServers	Defined in 6.3.11		Mandatory

The *Applications* folder may contain *Objects* representing the *Applications* known to the GDS. These *Objects* may be organized into subfolders as determined by the GDS. Some characteristics for organizing applications are the networks, the physical location, or the supported profiles. The *QueryServers Method* can be used to search for OPC UA *Applications* based on various criteria.

A GDS is not required to expose its *Applications* as browsable *Objects* in its *AddressSpace*, however, each *Application* shall have a unique *NodeId* which can be passed to *Methods* used to administer the GDS.

The *FindApplications Method* returns the *Applications* associated with an *ApplicationUri*. It can be called by any *Client* application.

The *RegisterApplication Method* is used to add a new *Application* to the GDS. It requires administrative privileges.

The *UpdateApplication Method* is used to update an existing *Application* in the GDS. It requires administrative privileges.

The *UnregisterApplication Method* is used to remove an *Application* from the GDS. It requires administrative privileges.

The *QueryApplications Method* is used to find *Client* or *Server* applications that meet the criteria provided. This *Method* replaces the *QueryServers Method.*

The *QueryServers Method* is used to find *Servers* that meet the criteria specified. It can be called by any *Client* application. This *Method* has been replaced by the *QueryApplications Method*.

6.3.4 FindApplications

FindApplications is used to find the *ApplicationId* for an OPC UA *Application* known to the GDS. In normal situations, the list of records returned will not have more than one entry; however, system configuration errors can create situations where the GDS has multiple entries for a single *ApplicationUri*. If this happens, a human will likely have to look at records to determine which record is the true match for the *ApplicationUri*.

Documents
Document Sharing Hub
Full document details: DIN EN IEC 62541-12:2021-08

If the returned array is null or zero length then the GDS does not have an entry for the *ApplicationUri*.

Signature

```
FindApplications(
       [in]  String applicationUri
       [out] ApplicationRecordDataType[] applications
       );
```

Argument	Description
applicationUri	The *ApplicationUri* that identifies the *Application* of interest.
applications	A list of application records that match the *ApplicationUri.* The ApplicationRecordDataType is defined in 6.3.5.

Method Result Codes (defined in Call Service)

Result Code	Description
Bad_UserAccessDenied	The current user does not have the rights required.

Table 4 specifies the *AddressSpace* representation for the *FindApplications Method*.

Table 4 – FindApplications Method AddressSpace definition

Attribute	Value				
BrowseName	FindApplications				
References	**NodeClass**	**BrowseName**	**DataType**	**TypeDefinition**	**ModellingRule**
HasProperty	Variable	InputArguments	Argument[]	PropertyType	Mandatory
HasProperty	Variable	OutputArguments	Argument[]	PropertyType	Mandatory

6.3.5 ApplicationRecordDataType

This type defines a DataType which represents a record in the GDS (see Table 5).

Documents
Document Sharing Hub
Full document details: DIN EN IEC 62541-12:2021-08