

TECHNICAL REPORT



OPC unified architecture – Part 1: Overview and concepts



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 It also gives information on projects, replaced and withdrawn publications.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.

TECHNICAL REPORT



OPC unified architecture – Part 1: Overview and concepts

INTERNATIONAL
ELECTROTECHNICAL
COMMISSION

ICS 25.040.40; 35.100.01

ISBN 978-2-8322-9093-4

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

CONTENTS

FOREWORD	4
1 Scope	7
2 Normative references	7
3 Terms, definitions, and abbreviated terms	8
3.1 Terms and definitions	8
3.2 Abbreviated terms	12
4 Structure of the OPC UA series	13
4.1 Specification organization	13
4.2 Core specification parts	14
4.3 Access Type specification parts	14
4.4 Utility specification parts	15
5 Overview	15
5.1 UA scope	15
5.2 General	15
5.3 Design goals	16
5.4 Integrated models and services	18
5.4.1 Security model	18
5.4.2 Integrated AddressSpace model	19
5.4.3 Integrated object model	20
5.4.4 Integrated services	20
5.5 Sessions	20
5.6 Redundancy	21
6 Systems concepts	21
6.1 Client Server Overview	21
6.2 OPC UA Clients	21
6.3 OPC UA Servers	22
6.3.1 General	22
6.3.2 Real objects	23
6.3.3 Server application	23
6.3.4 OPC UA AddressSpace	23
6.3.5 Publisher/subscriber Subscription entities	24
6.3.6 OPC UA Service Interface	24
6.3.7 Server to Server interactions	25
6.4 Redundancy	26
6.5 Publish-Subscribe	26
6.6 Synergy of models	27
7 Service Sets	28
7.1 General	28
7.2 Discovery Service Set	28
7.3 SecureChannel Service Set	28
7.4 Session Service Set	29
7.5 NodeManagement Service Set	29
7.6 View Service Set	29
7.7 Query Service Set	29
7.8 Attribute Service Set	30
7.9 Method Service Set	30

7.10	MonitoredItem Service Set	30
7.11	Subscription Service Set	31

~~Bibliography~~

Figure 1	– OPC UA specification organization	13
Figure 2	– OPC UA target applications	17
Figure 3	– OPC UA System architecture	21
Figure 4	– OPC UA Client architecture	22
Figure 5	– OPC UA Server architecture	23
Figure 6	– Peer-to-peer interactions between Servers	25
Figure 7	– Chained Server example	26
Figure 8	– Integrated Client Server and PubSub models	27
Figure 9	– SecureChannel and Session Services	29

INTERNATIONAL ELECTROTECHNICAL COMMISSION

OPC UNIFIED ARCHITECTURE –

Part 1: Overview and concepts

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.

The main task of IEC technical committees is to prepare International Standards. However, a technical committee may propose the publication of a technical report when it has collected data of a different kind from that which is normally published as an International Standard, for example "state of the art".

IEC TR 62541-1, which is a Technical Report, 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 of IEC TR 62541-1, published in 2016. This edition constitutes a technical revision.

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

- a) added Subclauses 6.5 and 6.6 and other text throughout to include PubSub introduction;
- b) added new transports and encodings to existing overview sections;
- c) removed WS-SecureConversation example since this mapping has been deprecated;
- d) improved the definition of Certificate.

The text of this Technical Report is based on the following documents:

Enquiry draft	Report on voting
65E/678/DTR	65E/702/RVDTR

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

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

Throughout this document and the referenced other Parts of the 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 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 also often 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 publication will remain unchanged until the stability date indicated on the IEC web site under "<http://webstore.iec.ch>" in the data related to the specific publication. At this date, the publication will be

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

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 1: Overview and concepts

1 Scope

This part of IEC 62541 presents the concepts and overview of the OPC Unified Architecture (OPC UA). Reading this document is helpful to understand the remaining parts of this multi-part document set. Each of the other parts of IEC 62541 is briefly explained along with a suggested reading order.

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 62541-2, *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 62541-8, *OPC unified architecture – Part 8: Data access*

IEC 62541-9, *OPC unified architecture – Part 9: Alarms and Conditions*

IEC 62541-10, *OPC unified architecture – Part 10: Programs*

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*

IEC 62541-14, *OPC unified architecture – Part 14: PubSub*

ITU X.509, *Information technology – Open Systems Interconnection – The Directory: Public-key and attribute certificate frameworks*
<https://www.itu.int/rec/T-REC-X.509>

3 Terms, definitions, and abbreviated terms

3.1 Terms and definitions

For the purposes of this document, the following terms ~~and definitions~~ 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

AddressSpace

collection of information that ~~an OPC UA~~ a *Server* makes visible to its *Clients*

Note 1 to entry: See IEC 62541-3 for a description of the contents and structure of the *Server AddressSpace*.

3.1.2

Aggregate

function that calculates derived values from *Raw data*

Note 1 to entry: Raw data may be from a historian or buffered real time data. Common *Aggregates* include averages over a given time range, minimum over a time range and maximum over a time range.

3.1.3

Alarm

type of *Event* associated with a state condition that typically requires acknowledgement

Note 1 to entry: See IEC 62541-9 for a description of *Alarms*.

3.1.4

Attribute

primitive characteristic of a *Node*

Note 1 to entry: All *Attributes* are defined by OPC UA, and may not be defined by *Clients* or *Servers*. *Attributes* are the only elements in the *AddressSpace* permitted to have data values.

3.1.5

Broker

intermediary program module that routes *NetworkMessages* from *Publishers* to *Subscribers*

Note 1 to entry: Brokers are building blocks of *Message Oriented Middleware*.

3.1.6

Certificate

digitally signed data structure that ~~describes capabilities~~ contains a public key and the identity of a *Client* or *Server*

3.1.7

Client

software application that sends *Messages* to OPC UA *Servers* conforming to the *Services* specified in this set of specifications

3.1.8

Condition

generic term that is an extension to an *Event*

Note 1 to entry: A *Condition* represents the conditions of a system or one of its components and always exists in some state.

3.1.9**Communication Stack**

layered set of software modules between the application and the hardware that provides various functions to encode, encrypt and format a *Message* for sending, and to decode, decrypt and unpack a *Message* that was received

~~**3.1.9**~~~~**Complex Data**~~

~~data that is composed of elements of more than one primitive data type, such as a structure~~

3.1.10**DataSet**

list of named data values

Note 1 to entry: A *DataSet* typically consists of *Event* fields or *Variable* values.

3.1.11**DataSetMessage**

payload of a *NetworkMessage* created from a *DataSet*

Note 1 to entry: The *DataSetMessage* is an immutable payload of the *NetworkMessage* handed off to the *Message Oriented Middleware* (transport layer) for delivery by the *Publisher*. The *Subscriber* receives the *DataSetMessage* as the payload of a *NetworkMessage* from the *Publisher* with additional headers that may be supplied by the *Message Oriented Middleware* along the way.

3.1.12**Discovery**

process by which ~~OPC UA~~ *Client* obtains information about ~~OPC UA~~ *Servers*, including endpoint and security information

3.1.13**Event**

generic term used to describe an occurrence of some significance within a system or system component

3.1.14**EventNotifier**

special *Attribute* of a *Node* that signifies that a *Client* may subscribe to that particular *Node* to receive *Notifications* of *Event* occurrences

3.1.15**Information Model**

organizational framework that defines, characterizes and relates information resources of a given system or set of systems

Note 1 to entry: The core ~~address space~~ *AddressSpace* model supports the representation of *Information Models* in the *AddressSpace*. See IEC 62541-5 for a description of the base OPC UA Information Model.

3.1.16**Message**

data unit conveyed between *Client* and *Server* that represents a specific *Service* request or response

3.1.17**Message Oriented Middleware**

infrastructure supporting sending and receiving *NetworkMessages* between distributed systems

Note 1 to entry: An OPC UA *Application* may support different types of *Message Oriented Middleware* infrastructures and protocols like AMQP, MQTT, or UDP with IP multicast. Other types like DDS or XMPP can also be integrated into the OPC UA *PubSub* model.

3.1.18

Method

callable software function that is a component of an *Object*

3.1.19

MonitoredItem

Client-defined entity in the *Server* used to monitor *Attributes* or *EventNotifiers* for new values or *Event* occurrences and that generates *Notifications* for them

3.1.20

NetworkMessage

DataSetMessages and header to facilitate delivery, routing, security and filtering

Note 1 to entry: The *Publisher* hands off the *NetworkMessage* to the *Message Oriented Middleware* (transport layer) to deliver *DataSetMessages* to the *Subscribers*.

Note 2 to entry: The term message is used with various connotations in the messaging world. The *Publisher* might like to think of the message as an immutable payload handed off to the *Message Oriented Middleware* for delivery. The *Subscriber* often thinks of the message as not only that immutable payload from the sender, but also various annotations supplied by the *Message Oriented Middleware* along the way. To avoid confusion, the term *DataSetMessage* is used to mean the message as supplied by the *Publisher* for a *DataSet* and the term *NetworkMessage* is used to mean the *DataSetMessage* plus sections for annotation at the head and tail of the *DataSetMessage*.

3.1.21

Node

fundamental component of an *AddressSpace*

3.1.22

NodeClass

class of a *Node* in an *AddressSpace*

Note 1 to entry: *NodeClasses* define the metadata for the components of the OPC UA object model. They also define constructs, such as *Views*, that are used to organize the *AddressSpace*.

3.1.23

Notification

generic term for data that announces the detection of an *Event* or of a changed *Attribute* value; *Notifications* are sent in *NotificationMessages*.

~~Note 1 to entry: Notifications are sent in NotificationMessages.~~

3.1.24

NotificationMessage

Message published from a *Subscription* that contains one or more *Notifications*

3.1.25

Object

Object Instance

Node that represents a physical or abstract element of a system

Note 1 to entry: *Objects* are modelled using the OPC UA Object Model. Systems, subsystems and devices are examples of *Objects*. An *Object* may be defined as an instance of an *ObjectType*.

~~3.1.22~~

~~Object Instance~~

~~synonym for Object~~

~~Note 1 to entry: Not all Objects are defined by ObjectTypes.~~

3.1.26

ObjectType

Node that represents the type definition for an *Object*

3.1.27**OPC UA Application**

Client, which calls OPC UA *Services*, or a *Server*, which performs those *Services*, or an OPC UA *Publisher* or an OPC UA *Subscriber*

3.1.28**Publisher**

entity sending *NetworkMessages* to a *Message Oriented Middleware*

Note 1 to entry: A *Publisher* can be a native OPC UA *Application* or an application that only has knowledge about the *Message Oriented Middleware* and the rules for encoding the *NetworkMessages* and *DataSetMessages*.

3.1.29**PubSub**

OPC UA variant of the publish subscribe messaging pattern

3.1.30**Profile**

specific set of capabilities to which a *Server* may claim conformance

Note 1 to entry: Each *Server* may claim conformance to more than one *Profile*.

Note 2 to entry: The set of capabilities are defined in IEC 62541-7.

3.1.31**Program**

executable *Object* that, when invoked, immediately returns a response to indicate that execution has started, and then returns intermediate and final results through *Subscriptions* identified by the *Client* during invocation

3.1.32**Reference**

explicit relationship (a named pointer) from one *Node* to another

Note 1 to entry: The *Node* that contains the *Reference* is the source *Node*, and the referenced *Node* is the target *Node*. All *References* are defined by *ReferenceTypes*.

3.1.33**ReferenceType**

Node that represents the type definition of a *Reference*

Note 1 to entry: The *ReferenceType* specifies the semantics of a *Reference*. The name of a *ReferenceType* identifies how source *Nodes* are related to target *Nodes* and generally reflects an operation between the two, such as "A contains B".

3.1.28**RootNode**

~~beginning or top Node of a hierarchy~~

~~Note 1 to entry: The RootNode of the OPC UA AddressSpace is defined in IEC 62541-5.~~

3.1.34**Server**

software application that implements and exposes the *Services* specified in ~~the IEC 62541 series of standards~~ this set of specifications

3.1.35**Service**

Client-callable operation in ~~an OPC UA~~ a *Server*

Note 1 to entry: *Services* are defined in IEC 62541-4. A *Service* is similar to a method call in a programming language or an operation in a Web services WSDL contract.

3.1.36

Service Set

group of related *Services*

3.1.37

Session

logical long-running connection between a *Client* and a *Server*

Note 1 to entry: A *Session* maintains state information between *Service* calls from the *Client* to the *Server*.

3.1.38

Subscriber

entity receiving *DataSetMessages* from a *Message Oriented Middleware*

Note 1 to entry: A *Subscriber* can be a native OPC UA *Application* or an application that has just knowledge about the *Message Oriented Middleware* and the rules for decoding the *NetworkMessages* and *DataSetMessages*. A *Subscription* in the OPC UA *Client Server* model has a different meaning than the *Subscriber* in the *PubSub* model.

3.1.39

Subscription

Client-defined endpoint in the *Server*, used to return *Notifications* to the *Client*

Note 1 to entry: "Subscription" is a generic term that describes a set of *Nodes* selected by the *Client* (1) that the *Server* periodically monitors for the existence of some condition, and (2) for which the *Server* sends *Notifications* to the *Client* when the condition is detected.

3.1.40

Underlying System

hardware or software platforms that exist as an independent entity

Note 1 to entry: *UA Applications* are dependent on an entity's existence in order to perform UA services. However, the entity is not dependent on *UA Applications*.

Note 2 to entry: : Hardware and software platforms include physical hardware, firmware, operating system, networking, non-UA applications, as well as other *UA Applications*. A Distributed Control System, PLC/Device, and UA Server are examples of an *Underlying System*.

3.1.41

Variable

Node that contains a value

3.1.42

View

specific subset of the *AddressSpace* that is of interest to the *Client*

3.2 Abbreviated terms

A&E	Alarms and Events
AMQP	Advanced Message Queuing Protocol
API	Application Programming Interface
COM	Component Object Model
DA	Data Access
DCS	Distributed Control System
DX	Data Exchange
DDS	Data Distribution Service
HDA	Historical Data Access
HMI	Human-Machine Interface
JSON	JavaScript Object Notation

LDAP	Lightweight Directory Access Protocol
MES	Manufacturing Execution System
MQTT	Message Queue Telemetry Transport
OPC	OPC Foundation (a non-profit industry association) formerly an acronym for “OLE for Process Control”. Acronym no longer used anymore .
PLC	Programmable Logic Controller
SCADA	Supervisory Control And Data Acquisition
SOAP	Simple Object Access Protocol
UA	Unified Architecture
UDDI	Universal Description, Discovery and Integration
UML	Unified Modelling Language
WSDL	Web Services Definition Language
XML	Extensible Markup Language
XMPP	Extensible Messaging and Presence Protocol

4 Structure of the OPC UA series

4.1 Specification organization

~~OPC UA~~ This specification is organized as a multi-part specification, as illustrated in Figure 1.

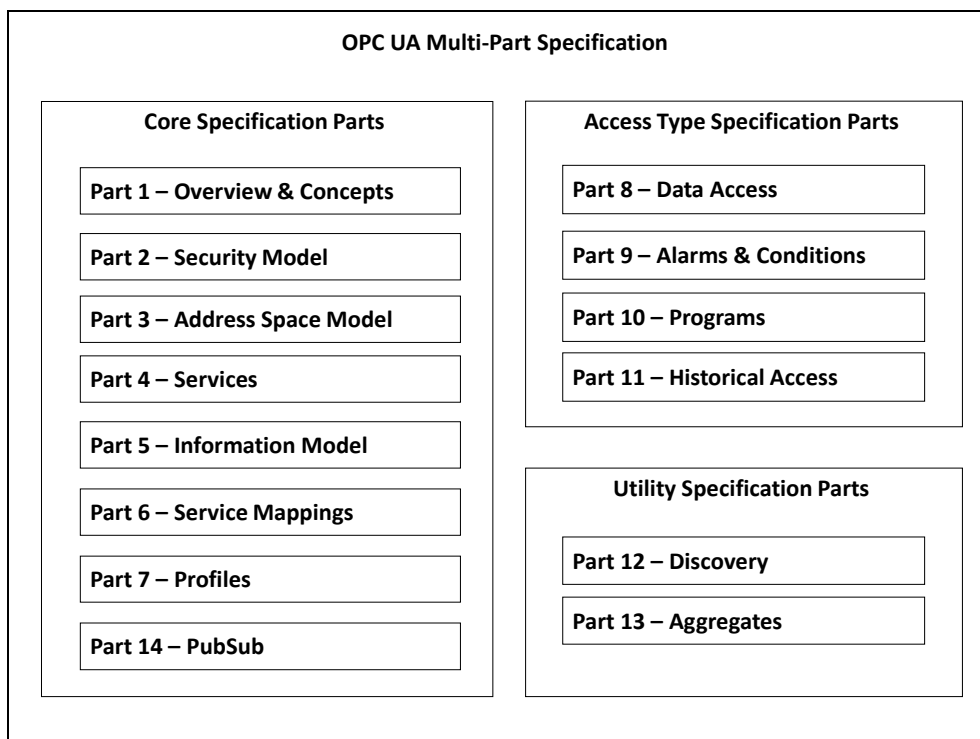


Figure 1 – OPC UA specification organization

IEC 62541-1 through IEC 62541-7 and IEC 62541-14 specify the core capabilities of OPC UA. These core capabilities define the structure of the OPC *AddressSpace* and the *Services* that operate on it. IEC 62541-14¹ defines an OPC UA publish subscribe pattern in addition to the *Client Server* pattern defined by the *Services* in IEC 62541-4. IEC 62541-8 through

¹ Under preparation. Stage at the time of publication: IEC 62541-14/FDIS:2019.

IEC 62541-11 apply these core capabilities to specific types of access previously addressed by separate OPC COM specifications, such as Data Access (DA), Alarms and Events (A&E) and Historical Data Access (HDA). IEC 62541-12 describes *Discovery* mechanisms for OPC UA and IEC 62541-13 describes ways of aggregating data.

~~Readers are encouraged to read IEC 62541-1 through IEC 62541-5 of the core specifications before reading IEC 62541-6 to IEC 62541-13. Some parts can be skipped depending on the needs of the reader.~~ Readers are encouraged to read IEC 62541-1 through IEC 62541-5 of the core specifications before reading IEC 62541-8 through IEC 62541-14. For example, a reader interested in UA Data Access should read IEC 62541-1 through IEC 62541-5 and IEC 62541-8. References in IEC 62541-8 may direct the reader to other parts of this specification.

4.2 Core specification parts

Part 1 – Overview and Concepts

Part 1 (this document) presents the concepts and overview of OPC UA.

Part 2 – Security Model

IEC TR 62541-2 describes the model for securing interactions between OPC UA ~~Clients and OPC UA Servers~~ *Applications*.

Part 3 – Address Space Model

IEC 62541-3 describes the contents and structure of the *Server's AddressSpace*.

Part 4 – Services

IEC 62541-4 specifies the *Services* provided by ~~OPC UA~~ Servers.

Part 5 – Information Model

IEC 62541-5 specifies the types and their relationships defined for ~~OPC UA~~ Servers.

Part 6 – Mappings

IEC 62541-6 specifies the mappings to transport protocols and data encodings supported by OPC UA.

Part 7 – Profiles

IEC 62541-7 specifies the *Profiles* that are available for OPC ~~Clients and Servers~~ *UA Applications*. These *Profiles* provide ~~groups~~ groupings of ~~Services or~~ functionality that can be used for conformance level certification. ~~Servers and Clients~~ *OPC UA Applications* will be tested against the *Profiles*.

4.3 Access Type specification parts

Part 8 – Data Access

IEC 62541-8 specifies the use of OPC UA for data access.

Part 9 – Alarms and Conditions

IEC 62541-9 specifies use of OPC UA support for access to *Alarms* and *Conditions*. The base system includes support for simple *Events*; this specification extends that support to include support for *Alarms* and *Conditions*.

Part 10 – Programs

IEC 62541-10 specifies OPC UA support for access to *Programs*.

Part 11 – Historical Access

IEC 62541-11 specifies use of OPC UA for historical access. This access includes both historical data and historical *Events*.

4.4 Utility specification parts

Part 12 – Discovery

IEC 62541-12 specifies how *Discovery Servers* operate in different scenarios and describes how UA *Clients* and *Servers* should interact with them. It also defines how UA related information should be accessed using common directory service protocols such as ~~UDDI and~~ LDAP.

Part 13 – Aggregates

IEC 62541-13 specifies how to compute and return aggregates like minimum, maximum, average, etc. *Aggregates* can be used with current and historical data.

Part 14 – PubSub

IEC 62541-14 specifies the OPC Unified Architecture (OPC UA) *PubSub* communication model. The *PubSub* communication model defines an OPC UA publish subscribe pattern in addition to the *Client Server* pattern defined by the *Services* in IEC 62541-4.

5 Overview

5.1 UA scope

OPC UA is applicable to ~~manufacturing software~~ components in ~~application areas~~ all industrial domains, such as ~~Field Devices~~ industrial sensors and actuators, control systems, Manufacturing Execution Systems and Enterprise Resource Planning Systems, including the Industrial Internet of Things (IIoT), Machine To Machine (M2M) as well as Industrie 4.0 and China 2025. These systems are intended to exchange information and to use command and control for industrial processes. OPC UA defines a common infrastructure model to facilitate this information exchange. OPC UA specifies the following:

- the information model to represent structure, behaviour and semantics;
- the message model to interact between applications;
- the communication model to transfer the data between end-points;
- the conformance model to guarantee interoperability between systems.

5.2 General

OPC UA is a platform-independent standard through which various kinds of systems and devices can communicate by sending *request and response Messages* between *Clients* and *Servers* *or NetworkMessages* between *Publishers* and *Subscribers* over various types of networks. It supports robust, secure communication that assures the identity of ~~Clients and Servers~~ *OPC UA Applications* and resists attacks.

In the *Client Server* model, OPC UA defines sets of *Services* that *Servers* may provide, and individual *Servers* specify to *Clients* what *Service* sets they support. Information is conveyed using OPC UA-defined and vendor-defined data types, and *Servers* define object models that *Clients* can dynamically discover. *Servers* can provide access to both current and historical data, as well as *Alarms* and *Events* to notify *Clients* of important changes. OPC UA can be mapped onto a variety of communication protocols and data can be encoded in various ways to trade off portability and efficiency.

In addition to the *Client Server* model, OPC UA defines a mechanism for *Publishers* to transfer the information to *Subscribers* using the *PubSub* model.

5.3 Design goals

OPC UA provides a consistent, integrated *AddressSpace* and service model. This allows a single ~~OPC UA~~ *Server* to integrate data, *Alarms* and *Events*, and history into its *AddressSpace*, and to provide access to them using an integrated set of *Services*. These *Services* also include an integrated security model.

OPC UA also allows *Servers* to provide *Clients* with type definitions for the *Objects* accessed from the *AddressSpace*. This allows information models to be used to describe the contents of the *AddressSpace*. OPC UA allows data to be exposed in many different formats, including binary structures and XML or JSON documents. The format of the data may be defined by OPC, other standard organizations or vendors. Through the *AddressSpace*, *Clients* can query the *Server* for the metadata that describes the format for the data. In many cases, *Clients* with no pre-programmed knowledge of the data formats will be able to determine the formats at runtime and properly utilize the data.

OPC UA adds support for many relationships between *Nodes* instead of being limited to just a single hierarchy. In this way, ~~an OPC UA~~ *a Server* may present data in a variety of hierarchies tailored to the way a set of *Clients* would typically like to view the data. This flexibility, combined with support for type definitions, makes OPC UA applicable to a wide array of problem domains. As illustrated in Figure 2, OPC UA is not targeted at just the SCADA, PLC and DCS interface, but also as a way to provide greater interoperability between higher level functions.

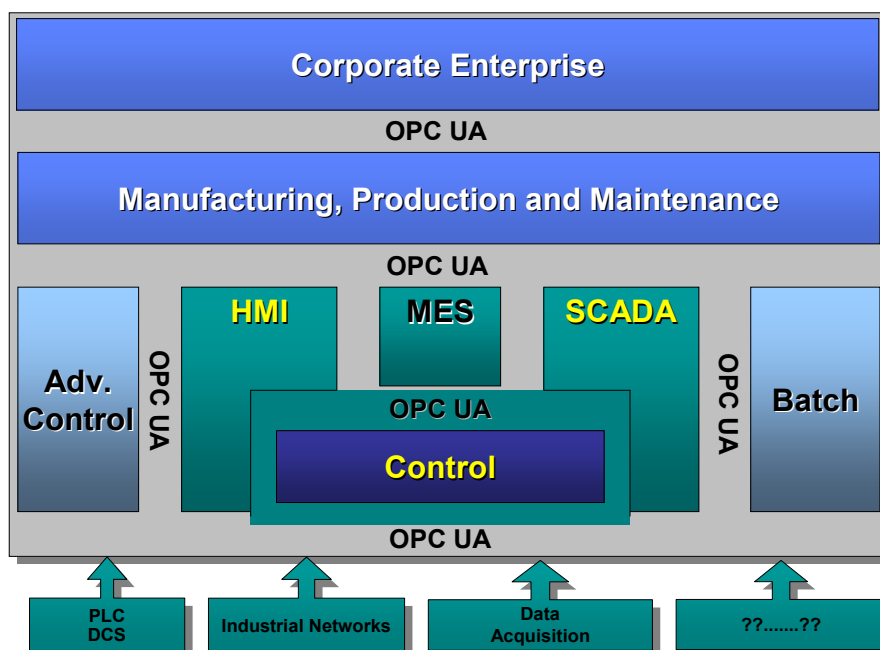


Figure 2 – OPC UA target applications

OPC UA is designed to provide robustness of published data. A major feature of all OPC servers is the ability to publish data and *Event Notifications*. OPC UA provides mechanisms for *Clients* to quickly detect and recover from communication failures associated with these transfers without having to wait for long timeouts provided by the underlying protocols.

OPC UA is designed to support a wide range of *Servers*, from plant floor PLCs to enterprise *Servers*. These *Servers* are characterized by a broad scope of size, performance, execution platforms, and functional capabilities. Therefore, OPC UA defines a comprehensive set of capabilities, and *Servers* may implement a subset of these capabilities. To promote interoperability, OPC UA defines subsets, referred to as *Profiles*, to which *Servers* may claim conformance. *Clients* can then discover the *Profiles* of a *Server*, and tailor their interactions with that *Server* based on the *Profiles*. *Profiles* are defined in IEC 62541-7.

The OPC UA specifications are layered to isolate the core design from the underlying computing technology and network transport. This allows OPC UA to be mapped to future technologies as necessary, without negating the basic design. Mappings and data encodings are described in IEC 62541-6. ~~Two~~ ~~Three~~ data encodings are defined:

- XML/text,
- UA Binary,
- JSON.

In addition, ~~three transport~~ ~~several~~ protocols are defined:

- OPC UA TCP,
- ~~SOAP/HTTP,~~
- HTTPS,
- WebSockets.

~~Clients and Servers~~ *OPC UA Applications* that support multiple transports and encodings will allow the end users to make decisions about trade-offs between performance and ~~XML-Web~~

~~service~~ compatibility at the time of deployment, rather than having these trade-offs determined by the OPC vendor at the time of product definition.

OPC UA is designed as the migration path for OPC clients and servers that are based on Microsoft COM technology. Care has been taken in the design of OPC UA so that existing data exposed by OPC COM servers (DA, HDA and A&E) can easily be mapped and exposed via OPC UA. Vendors may choose ~~to migrate~~ migrating their products natively to OPC UA or use external wrappers to convert from OPC COM to OPC UA and vice-versa. Each of the previous OPC specifications defined its own address space model and its own set of *Services*. OPC UA unifies the previous models into a single integrated address space with a single set of *Services*.

OPC UA *PubSub* opens new application fields for OPC UA. The following are some example uses for *PubSub*:

- Configurable peer to peer communication between controllers and between controllers and HMIs. The peers are not directly connected and do not even need to know about the existence of each other. The data exchange often requires a fixed time-window; it may be point-to-point or a multi-point connection.
- Asynchronous workflows. For example, an order processing application can place an order on a message queue or an enterprise service bus. From there it can be processed by one or more workers.
- Logging to multiple systems. For example, sensors or actuators can write logs to a monitoring system, an HMI, an archive application for later querying, and so on.
- *Servers* representing services or devices can stream data to applications hosted in the cloud; for example, backend servers, big data analytics for system optimization and predictive maintenance.

PubSub is not bound to a particular messaging system. Rather, it can be mapped to various different systems as illustrated with two examples:

- *PubSub* with UDP may be well-suited in production environments for frequent transmissions of small amounts of data. It also allows data exchange in one-to-one and one-to-many configurations.
- The use of established messaging protocols (e.g. the ISO/IEC AMQP 1.0 protocol) with JSON data encoding supports the cloud integration path and readily allows handling of the information in modern stream and batch analytics systems.

5.4 Integrated models and services

5.4.1 Security model

5.4.1.1 General

OPC UA security is concerned with the authentication of *Clients* and *Servers*, the authentication of users, the integrity and confidentiality of their communications, and the verifiability of claims of functionality. It does not specify the circumstances under which various security mechanisms are required. That specification is crucial, but it is made by the designers of the system at a given site and may be specified by other standards.

Rather, OPC UA provides a security model, described in IEC TR 62541-2, in which security measures can be selected and configured to meet the security needs of a given installation. This model includes security mechanisms and parameters. In some cases, the mechanism for exchanging security parameters is defined, but the way that applications use these parameters is not. This framework also defines a minimum set of security *Profiles* that all ~~OPC UA-Servers~~ *Applications* support, even though they may not be used in all installations. Security *Profiles* are defined in IEC 62541-7.