

INTERNATIONAL STANDARD

NORME INTERNATIONALE



**OPC Unified Architecture –
Part 13: Aggregates**

ITU STANDARD PREVIEW
(standards.iteh.ai)

**Architecture unifiée OPC –
Partie 13: Agrégats**

<https://standards.iteh.ai/catalog/standards/sist/6f18ca63-f801-433c-8cf5-e358f4ace2dd/iec-62541-13-2020>
[IEC 62541-13:2020](#)



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.

Droits de reproduction réservés. Sauf indication contraire, aucune partie de cette publication ne peut être reproduite ni utilisée sous quelque forme que ce soit et par aucun procédé, électronique ou mécanique, y compris la photocopie et les microfilms, sans l'accord écrit de l'IEC ou du Comité national de l'IEC du pays du demandeur. Si vous avez des questions sur le copyright de l'IEC ou si vous désirez obtenir des droits supplémentaires sur cette publication, utilisez les coordonnées ci-après ou contactez le Comité national de l'IEC de votre pays de résidence.

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 It also gives information on projects, replaced and withdrawn publications.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

A propos de l'IEC

La Commission Electrotechnique Internationale (IEC) est la première organisation mondiale qui élabore et publie des Normes internationales pour tout ce qui a trait à l'électricité, à l'électronique et aux technologies apparentées.

A propos des publications IEC

Le contenu technique des publications IEC est constamment revu. Veuillez vous assurer que vous possédez l'édition la plus récente, un corrigendum ou amendement peut avoir été publié.

Recherche de publications IEC -

webstore.iec.ch/advsearchform

La recherche avancée permet de trouver des publications IEC en utilisant différents critères (numéro de référence, texte, comité d'études,...). Elle donne aussi des informations sur les projets et les publications remplacées ou retirées.

IEC Just Published - webstore.iec.ch/justpublished

Restez informé sur les nouvelles publications IEC. Just Published détaille les nouvelles publications parues. Disponible en ligne et une fois par mois par email.

Service Clients - webstore.iec.ch/csc

Si vous désirez nous donner des commentaires sur cette publication ou si vous avez des questions contactez-nous: sales@iec.ch.

Electropedia - www.electropedia.org

Le premier dictionnaire d'électrotechnologie en ligne au monde, avec plus de 22 000 articles terminologiques en anglais et en français, ainsi que les termes équivalents dans 16 langues additionnelles. Egalement appelé Vocabulaire Electrotechnique International (IEV) en ligne.

Glossaire IEC - std.iec.ch/glossary

67 000 entrées terminologiques électrotechniques, en anglais et en français, extraites des articles Termes et Définitions des publications IEC parues depuis 2002. Plus certaines entrées antérieures extraites des publications des CE 37, 77, 86 et CISPR de l'IEC.



IEC 62541-13

Edition 2.0 2020-06

INTERNATIONAL STANDARD

NORME INTERNATIONALE



OPC Unified Architecture –
Part 13: Aggregates

ITeH STANDARD PREVIEW
(standards.iteh.ai)

Architecture unifiée OPC –
Partie 13: Agrégats

[IEC 62541-13:2020](#)

<https://standards.iteh.ai/catalog/standards/sist/6f18ca63-f801-433c-8cf5-e358f4ace2dd/iec-62541-13-2020>

INTERNATIONAL
ELECTROTECHNICAL
COMMISSION

COMMISSION
ELECTROTECHNIQUE
INTERNATIONALE

ICS 25.040.40; 35.100.05

ISBN 978-2-8322-8296-0

Warning! Make sure that you obtained this publication from an authorized distributor.
Attention! Veuillez vous assurer que vous avez obtenu cette publication via un distributeur agréé.

CONTENTS

FOREWORD	7
1 Scope	9
2 Normative references	9
3 Terms, definitions, and abbreviated terms	9
3.1 Terms and definitions	9
3.2 Abbreviated terms	12
4 Aggregate information model	13
4.1 General	13
4.2 Aggregate Objects	13
4.2.1 General	13
4.2.2 AggregateFunction Object	14
4.3 MonitoredItem AggregateFilter	16
4.3.1 MonitoredItem AggregateFilter Defaults	16
4.3.2 MonitoredItem Aggregates and Bounding Values	16
4.4 Exposing Supported Functions and Capabilities	16
5 Aggregate specific usage of Services	17
5.1 General	17
5.2 Aggregate data handling	18
5.2.1 Overview	18
5.2.2 ReadProcessedDetails structure overview	18
5.2.3 AggregateFilter structure overview	18
5.3 Aggregates StatusCodes	19
5.3.1 Overview	19
5.3.2 Operation level result codes	19
5.3.3 Aggregate Information Bits	20
5.4 Aggregate details	21
5.4.1 General	21
5.4.2 Common characteristics	21
5.4.3 Specific aggregated data handling	24
Annex A (informative) Aggregate Specific examples – Historical Access	67
A.1 Historical Aggregate specific characteristics	67
A.1.1 Example Aggregate data – Historian 1	67
A.1.2 Example Aggregate data – Historian 2	68
A.1.3 Example Aggregate data – Historian 3	69
A.1.4 Example Aggregate data – Historian 4	70
A.2 Interpolative	71
A.2.1 Description	71
A.2.2 Interpolative data	71
A.3 Average	73
A.3.1 Description	73
A.3.2 Average data	73
A.4 TimeAverage	74
A.4.1 Description	74
A.4.2 TimeAverage data	75
A.5 TimeAverage2	76
A.5.1 Description	76

A.5.2	TimeAverage2 data	76
A.6	Total	78
A.6.1	Description	78
A.6.2	Total data	78
A.7	Total2	80
A.7.1	Description	80
A.7.2	Total2 data	80
A.8	Minimum	81
A.8.1	Description	81
A.8.2	Minimum data	82
A.9	Maximum	82
A.9.1	Description	82
A.9.2	Maximum data	83
A.10	MininumActualTime	83
A.10.1	Description	83
A.10.2	MinimumActualTime data	84
A.11	MaximumActualTime	84
A.11.1	Description	84
A.11.2	MaximumActualTime data	85
A.12	Range	85
A.12.1	Description	85
A.12.2	Range data	86
A.13	Minimum2	86
A.13.1	Description	86
A.13.2	Minimum2 data	87
A.14	Maximum2	87
A.14.1	Description	87
A.14.2	Maximum2 data	88
A.15	MinimumActualTime2	88
A.15.1	Description	88
A.15.2	MinimumActualTime2 data	89
A.16	MaximumActualTime2	89
A.16.1	Description	89
A.16.2	MaximumActualTime2 data	90
A.17	Range2	90
A.17.1	Description	90
A.17.2	Range2 data	91
A.18	AnnotationCount	91
A.18.1	Description	91
A.18.2	AnnotationCount data	91
A.19	Count	92
A.19.1	Description	92
A.19.2	Count data	92
A.20	DurationInStateZero	93
A.20.1	Description	93
A.20.2	DurationInStateZero data	93
A.21	DurationInStateNonZero	93
A.21.1	Description	93
A.21.2	DurationInStateNonZero data	93

A.22	NumberOfTransitions	94
A.22.1	Description	94
A.22.2	NumberOfTransitions data	94
A.23	Start	95
A.23.1	Description	95
A.23.2	Start data	95
A.24	End	95
A.24.1	Description	95
A.24.2	End data	96
A.25	StartBound	96
A.25.1	Description	96
A.25.2	StartBound data	97
A.26	EndBound	97
A.26.1	Description	97
A.26.2	EndBound data	98
A.27	Delta	98
A.27.1	Description	98
A.27.2	Delta data	99
A.28	DeltaBounds	99
A.28.1	Description	99
A.28.2	DeltaBounds data	100
A.29	DurationGood	100
A.29.1	Description	100
A.29.2	DurationGood data	101
A.30	DurationBad	102
A.30.1	Description	102
A.30.2	DurationBad data	102
A.31	PercentGood	103
A.31.1	Description	103
A.31.2	PercentGood data	103
A.32	PercentBad	104
A.32.1	Description	104
A.32.2	PercentBad data	104
A.33	WorstQuality	105
A.33.1	Description	105
A.33.2	WorstQuality data	105
A.34	WorstQuality2	106
A.34.1	Description	106
A.34.2	WorstQuality2 data	106
A.35	StandardDeviationSample	107
A.35.1	Description	107
A.35.2	StandardDeviationSample data	107
A.36	VarianceSample	107
A.36.1	Description	107
A.36.2	VarianceSample data	108
A.37	StandardDeviationPopulation	108
A.37.1	Description	108
A.37.2	StandardDeviationPopulation data	108
A.38	VariancePopulation	109

ITh STANDARD PREVIEW

(standards.iteh.ai)

IEC 62541-13:2020

<https://standards.iteh.ai/catalog/standards/sist/6118ca63-f801-433c-8c15-e35894acc2dd/iec-62541-13-2020>

A.38.1	Description	109
A.38.2	VariancePopulation data	109
Figure 1 – Representation of Aggregate Configuration information in the AddressSpace..... 17		
Figure 2 – Variable with Stepped = False and Simple Bounding Values 25		
Figure 3 – Variable with Stepped = True and Interpolated Bounding Values 26		
Figure A.1 – Historian 1 68		
Figure A.2 – Historian 2 69		
Figure A.3 – Historian 3 70		
Table 1 – Interpolation examples 10		
Table 2 – AggregateConfigurationType Definition 13		
Table 3 – Aggregate Functions Definition..... 14		
Table 4 – AggregateFunctionType Definition..... 14		
Table 5 – Standard AggregateType Nodes..... 15		
Table 6 – ReadProcessedDetails 18		
Table 7 – AggregateFilter structure 19		
Table 8 – Bad operation level result codes..... 19		
Table 9 – Uncertain operation level result codes..... 20		
Table 10 – Data location (standards.itech.ai) 20		
Table 11 – Additional information..... 20		
Table 12 – History Aggregate interval information..... 22		
Table 13 – Standard History Aggregate Data Type information..... 23		
Table 14 – Aggregate table description..... 27		
Table 15 – Interpolative Aggregate summary 30		
Table 16 – Average Aggregate summary 31		
Table 17 – TimeAverage Aggregate summary..... 32		
Table 18 – TimeAverage2 Aggregate summary..... 33		
Table 19 – Total Aggregate summary..... 34		
Table 20 – Total2 Aggregate summary..... 35		
Table 21 – Minimum Aggregate summary 36		
Table 22 – Maximum Aggregate summary..... 37		
Table 23 – MinimumActualTime Aggregate summary 38		
Table 24 – MaximumActualTime Aggregate summary 39		
Table 25 – Range Aggregate summary 40		
Table 26 – Minimum2 Aggregate summary..... 41		
Table 27 – Maximum2 Aggregate summary..... 42		
Table 28 – MinimumActualTime2 Aggregate summary 43		
Table 29 – MaximumActualTime2 Aggregate summary 44		
Table 30 – Range2 Aggregate summary 45		
Table 31 – AnnotationCount Aggregate summary..... 46		
Table 32 – Count Aggregate summary 47		
Table 33 – DurationInStateZero Aggregate summary 48		

Table 34 – DurationInStateNonZero Aggregate Summary	49
Table 35 – NumberOfTransitions Aggregate summary	50
Table 36 – Start Aggregate summary	51
Table 37 – End Aggregate summary	52
Table 38 – Delta Aggregate summary	53
Table 39 – StartBound Aggregate summary	54
Table 40 – EndBound Aggregate summary	55
Table 41 – DeltaBounds Aggregate summary.....	56
Table 42 – DurationGood Aggregate summary	57
Table 43 – DurationBad Aggregate summary	58
Table 44 – PercentGood Aggregate summary	59
Table 45 – PercentBad Aggregate summary	60
Table 46 – WorstQuality Aggregate summary	61
Table 47 – WorstQuality2 Aggregate summary.....	62
Table 48 – StandardDeviationSample Aggregate summary	63
Table 49 – VarianceSample Aggregate summary	64
Table 50 – StandardDeviationPopulation Aggregate summary	65
Table 51 – VariancePopulation Aggregate summary	66

iteh STANDARD PREVIEW
(standards.iteh.ai)

[IEC 62541-13:2020](https://standards.iteh.ai/catalog/standards/sist/6f18ca63-f801-433c-8cf5-e358f4aee2dd/iec-62541-13-2020)

<https://standards.iteh.ai/catalog/standards/sist/6f18ca63-f801-433c-8cf5-e358f4aee2dd/iec-62541-13-2020>

INTERNATIONAL ELECTROTECHNICAL COMMISSION

OPC UNIFIED ARCHITECTURE –

Part 13: Aggregates

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.
<http://standards.iteh.ai/catalog/standards/sist/62541-13-2020>
- 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.

IEC 62541-13 has been prepared by subcommittee 65E: Devices and integration in enterprise systems, of IEC technical committee 65: Industrial-process measurement, control and automation.

This second edition cancels and replaces the first edition of IEC 62541-13, published in 2015. This edition constitutes a technical revision.

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

- a) no technical changes but numerous clarifications. Also some corrections to the examples.

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

FDIS	Report on voting
65E/697/FDIS	65E/712/RVD

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

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

Throughout this document and the 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 13: Aggregates

1 Scope

This part of IEC 62541 is part of the overall OPC Unified Architecture specification series and defines the information model associated with Aggregates.

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

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

3 Terms, definitions, and abbreviated terms

3.1 Terms and definitions

For the purposes of this document, the terms and definitions given in IEC TR 62541-1, IEC 62541-3, IEC 62541-4, and IEC 62541-11 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

ProcessingInterval

timespan for which derived values are produced based on a specified *Aggregate*

Note 1 to entry: The total time domain specified for *ReadProcessed* is divided by the *ProcessingInterval*. For example, performing a 10-minute Average over the time range 12:00 to 12:30 would result in a set of three intervals of *ProcessingInterval* length, with each interval having a start time of 12:00, 12:10 and 12:20 respectively. The rules used to determine the interval *Bounds* are discussed in 5.4.2.2.

3.1.2

interpolated data

data that is calculated from data samples

Note 1 to entry: Data samples may be historical data or buffered real time data. An *interpolated* value is calculated from the data points on either side of the requested timestamp.

3.1.3

EffectiveEndTime

time immediately before *endTime*

Note 1 to entry: All *Aggregate* calculations include the *startTime* but exclude the *endTime*. However, it is sometimes necessary to return an *Interpolated* End Bound as the value for an *Interval* with a timestamp that is in the *interval*. *Servers* are expected to use the time immediately before *endTime* where the time resolution of the *Server* determines the exact value (do not confuse this with hardware or operating system time resolution). For example, if the *endTime* is 12:01:00, the time resolution is 1 second, then the *EffectiveEndTime* is 12:00:59. See 5.4.2.4.

If time is flowing backwards, *Servers* are expected to use the time immediately after *endTime* where the time resolution of the *Server* determines the exact value.

3.1.4

extrapolated data

data constructed from a discrete data set but is outside of the discrete data set

Note 1 to entry: It is similar to the process of interpolation, which constructs new points between known points, but its result is subject to greater uncertainty. *Extrapolated* data is used in cases where the requested time period falls farther into the future than the data available in the underlying system. See example in Table 1.

3.1.5

SlopedInterpolation

simple linear interpolation

Note 1 to entry: Compare to curve fitting using linear polynomials. See example in Table 1.

3.1.6

SteppedInterpolation

interpolation holding the last data point constant or interpolating the value based on a horizontal line fit

Note 1 to entry: Consider the following Table 1 of raw and *Interpolated/Extrapolated* values:

Table 1 – Interpolation examples

Timestamp	Raw Value	Sloped Interpolation	Stepped Interpolation
12:00:00	10		
12:00:05		15	10
12:00:08		18	10
12:00:10	20		
12:00:15		25	20
12:00:20	30		
		SlopedExtrapolation	SteppedExtrapolation
12:00:25		35	30
12:00:27		37	30

3.1.7

bounding values

values at the *startTime* and *endTime* needed for *Aggregates* to compute the result

Note 1 to entry: If *Raw data* does not exist at the *startTime* and *endTime* a value shall be estimated. There are two ways to determine *Bounding Values* for an interval. One way (called *Interpolated Bounding Values*) uses the first non-Bad data points found before and after the timestamp to estimate the bound. The other (called *Simple Bounding Values*) uses the data points immediately before and after the boundary timestamps to estimate the bound even if these points are Bad. Subclauses 3.1.8 and 3.1.9 describe the two different approaches in more detail.

In all cases the *TreatUncertainAsBad* (see 4.2.1.2) flag is used to determine whether Uncertain values are Bad or non-Bad.

If a Raw value was not found and a non-Bad bounding value exists the *Aggregate Bits* (see 5.3.3) are set to 'Interpolated'.

When calculating *bounding values*, the value portion of *Raw data* that has Bad status is set to null. This means the value portion is not used in any calculation and a null is returned if the raw value is returned. The status portion is determined by the rules specified by the bound or *Aggregate*.

The *Interpolated Bounding Values* approach (see 3.1.8) is the same as what is used in Classic OPC Historical Data Access (HDA) and is important for applications such as advanced process control where having useful values at all times is important. The *Simple Bounding Values* approach (see 3.1.9) is new in this standard and is important for applications which shall produce regulatory reports and cannot use estimated values in place of Bad data.

3.1.8

interpolated bounding values

bounding values determined by a calculation using the nearest Good value

Note 1 to entry: *Interpolated Bounding Values* using *SlopedInterpolation* are calculated as follows:

- if a non-Bad Raw value exists at the timestamp then it is the bounding value;
- find the first non-Bad Raw value before the timestamp;
- find the first non-Bad Raw value after the timestamp;
- draw a line between before value and after value;
- use point where the line crosses the timestamp as an estimate of the bounding value.

The calculation can be expressed with the following formula:

$$V_{\text{bound}} = (T_{\text{bound}} - T_{\text{before}}) \times (V_{\text{after}} - V_{\text{before}}) / (T_{\text{after}} - T_{\text{before}}) + V_{\text{before}}$$

where V_x is a value at 'x' and T_x is the timestamp associated with V_x .

If no non-Bad values exist before the timestamp the *StatusCode* is *Bad_NoData*. The *StatusCode* is *Uncertain_DataSubNormal* if any Bad values exist between the before value and after value. If either the before value or the after value are Uncertain the *StatusCode* is *Uncertain_DataSubNormal*. If the after value does not exist the before value shall be extrapolated using *SlopedExtrapolation* or *SteppedExtrapolation*.

The period of time that is searched to discover the Good values before and after the timestamp is *Server* dependent, but if a Good value is not found within some reasonable time range then the *Server* will assume it does not exist. The *Server* as a minimum should search a time range which is at least the size of the *ProcessingInterval*.

Interpolated Bounding Values using *SlopedExtrapolation* are calculated as follows:

- find the first non-Bad Raw value before timestamp;
- find the second non-Bad Raw value before timestamp;
- draw a line between these two values;
- extend the line to where it crosses the timestamp;
- use the point where the line crosses the timestamp as an estimate of the bounding value.

The formula is the same as the one used for *SlopedInterpolation*.

The *StatusCode* is always *Uncertain_DataSubNormal*. If only one non-Bad raw value can be found before the timestamp then *SteppedExtrapolation* is used to estimate the bounding value.

Interpolated Bounding Values using *SteppedInterpolation* are calculated as follows:

- if a non-Bad Raw value exists at the timestamp then it is the bounding value;
- find the first non-Bad Raw value before timestamp;
- use the value as an estimate of the bounding value.

The *StatusCode* is *Uncertain_DataSubNormal* if any Bad values exist between the before value and the timestamp. If no non-Bad Raw data exists before the timestamp then the *StatusCode* is *Bad_NoData*. If the value before the timestamp is *Uncertain* the *StatusCode* is *Uncertain_DataSubNormal*. The value after the timestamp is not needed when using *SteppedInterpolation*; however, if the timestamp is after the end of the data then the bounding value is treated as extrapolated and the *StatusCode* is *Uncertain_DataSubNormal*.

SteppedExtrapolation is a term that describes *SteppedInterpolation* when a timestamp is after the last value in the history collection.

3.1.9

simple bounding values

bounding values determined by a calculation using the nearest value

Note 1 to entry: *Simple Bounding Values* using *SlopedInterpolation* are calculated as follows:

- if any Raw value exists at the timestamp then it is the bounding value;
- find the first Raw value before timestamp;
- find the first Raw value after timestamp;
- if the value after the timestamp is Bad then the before value is the bounding value;
- draw a line between before value and after value;
- use point where the line crosses the timestamp as an estimate of the bounding value.

The formula is the same as the one used for *SlopedInterpolation* in Clause 3.1.5.

If a Raw value at the timestamp is Bad the *StatusCode* is *Bad_NoData*. If the value before the timestamp is Bad the *StatusCode* is *Bad_NoData*. If the value before the timestamp is *Uncertain* the *StatusCode* is *Uncertain_DataSubNormal*. If the value after the timestamp is Bad or *Uncertain* the *StatusCode* is *Uncertain_DataSubNormal*.

Simple Bounding Values using *SteppedInterpolation* are calculated as follows:

- if any Raw value exists at the timestamp then it is the bounding value;
- find the first Raw value before timestamp;
- if the value before timestamp is non-Bad then it is the bounding value.

If a Raw value at the timestamp is Bad the *StatusCode* is *Bad_NoData*. If the value before the timestamp is Bad the *StatusCode* is *Bad_NoData*. If the value before the timestamp is *Uncertain* the *StatusCode* is *Uncertain_DataSubNormal*.

If either bounding time of an interval is beyond the last data point then the *Server* may use extrapolation or return an error. If extrapolation is used by the server the type [*SteppedExtrapolation* or *SloppedExtrapolation*] of extrapolation is server specific.

In some Historians, the last Raw value does not necessarily indicate the end of the data. Based on the Historian's knowledge of the data collection mechanism, i.e. frequency of data updates and latency, the Historian may extend the last value to a time known by the Historian to be covered. When calculating *Simple Bounding Values* the Historian will act as if there is another Raw value at this timestamp.

In the same way, if the earliest time of an interval starts before the first data point in history and the latest time is after the first data point in history, then the interval will be treated as if the interval extends from the first data point in history to the latest time of the interval and the *StatusCode* of the interval will have the Partial bit set (see 5.3.3.2).

The period of time that is searched to discover the values before and after the timestamp is *Server* dependent, but if a value is not found within some reasonable time range then the *Server* will assume it does not exist. The *Server* as a minimum should search a time range which is at least the size of the *ProcessingInterval*.

3.2 Abbreviated terms

DA	Data Access
HA	Historical Access (access to historical data or events)
HDA	Historical Data Access
UA	Unified Architecture

4 Aggregate information model

4.1 General

IEC 62541-3 and IEC 62541-5 standards define the representation of *Aggregate* historical or buffered real time data in the OPC Unified Architecture. This includes the definition of *Aggregates* used in processed data retrieval and in historical retrieval. This definition includes both standard *Reference* types and *Object* types.

4.2 Aggregate Objects

4.2.1 General

4.2.1.1 Overview

OPC UA *Servers* can support several different functionalities and capabilities. The following standard *Objects* are used to expose these capabilities in a common fashion, and there are several standard defined concepts that can be extended by vendors.

4.2.1.2 AggregateConfigurationType

The *AggregateConfigurationType* defines the general characteristics of a *Node* that defines the *Aggregate* configuration of any *Variable* or *Property*. *AggregateConfiguration Object* represents the browse entry point for information on how the *Server* treats *Aggregate* specific functionality such as handling Uncertain data. It is formally defined in Table 2.

Table 2 – AggregateConfigurationType Definition
(standards.iteh.ai)

Attribute	Value				
BrowseName	AggregateConfigurationType IEC 62541-13:2020				
IsAbstract	False https://standards.iteh.ai/catalog/standards/sist/6ff8ca63-f801-433c-8cf5-e358f4ace2dd/iec-62541-13-2020				
References	NodeClass	BrowseName	DataType	TypeDefinition	ModellingRule
Subtype of the <i>BaseObjectType</i> defined in IEC 62541-5					
HasProperty	Variable	TreatUncertainAsBad	Boolean	PropertyType	Mandatory
HasProperty	Variable	PercentDataBad	Byte	PropertyType	Mandatory
HasProperty	Variable	PercentDataGood	Byte	PropertyType	Mandatory
HasProperty	Variable	UseSlopedExtrapolation	Boolean	PropertyType	Mandatory

The *TreatUncertainAsBad Variable* indicates how the *Server* treats data returned with a *StatusCode* severity Uncertain with respect to *Aggregate* calculations. A value of True indicates the *Server* considers the severity equivalent to Bad, a value of False indicates the *Server* considers the severity equivalent to Good, unless the *Aggregate* definition says otherwise. The default value is True. Note that the value is still treated as Uncertain when the *StatusCode* for the result is calculated.

The *PercentDataBad Variable* indicates the minimum percentage of Bad data in a given interval required for the *StatusCode* for the given interval for processed data request to be set to Bad. (Uncertain is treated as defined above.) Refer to 5.4.3 for details on using this *Variable* when assigning *StatusCodes*. For details on which *Aggregates* use the *PercentDataBad Variable*, see the definition of each *Aggregate*. The default value is 100.

The *PercentDataGood Variable* indicates the minimum percentage of Good data in a given interval required for the *StatusCode* for the given interval for the processed data requests to be set to Good. Refer to 5.4.3 for details on using this *Variable* when assigning *StatusCodes*. For details on which *Aggregates* use the *PercentDataGood Variable*, see the definition of each *Aggregate*. The default value is 100.