Table of Contents

Public

Public

Reviewer OPC UA Server

Reviewer Enterprise Edition integrates an OPC UA server, certified by the OPC Foundation, providing read-only access to the Reviewer data.

The OPC UA Server supports:

- Security
- User Authentication
- Data Access
- Event Access
- Raw Historic Data
- Methods

It does not support Aggregated Data.

Security

The following security policies are supported:

- None
- Basic256Sha256
- Aes128_Sha256_RsaOaep
- Aes256_Sha256_RsaPss

The following security modes are supported:

- Sign
- SignAndEncrypt
- Sign plus SignAndEncrypt

Public

User Authentication

Data Reviewer OPC UA Server accepts three authentication methods to connect to the server,

1. via a valid Data Reviewer user
   Clients must log in using the Data Reviewer username and password to accept the connection.
2. via an anonymous authentication (login not required)
   Trust a client software certificate and link it to a valid Data Reviewer user. See Client Certificates.
3. via user certificate authentication
   Trust a user certificate and link it to a valid Data Reviewer user. See User Certificates Tab.

Once connected, the client will be limited to the data and actions defined for that user by:

- Reviewer view management
- Run scheduled reports privilege

Public

Data Access

All Reviewer data is read directly (polling) or through subscription, and includes:

- Instrument configuration data
- Channel values, messages, and batches
- Event logs

All timestamps are UTC.

Event Access

OPC UA events are generated when the Reviewer,

- Event log is updated
- Security revision is updated
- Configuration revision is updated
- Address space is updated

For more information, see OPC UA Event Generation.

Raw Historic Data

Raw historic data is available between the specified UTC start and end times for:

- Channel samples
- Channel values
- Messages
- Batches
- Events

All timestamps are UTC.

Methods

The methods Run, RunWithDates, and RunWithEndDate on the Report node are used to activate the Scheduled Reports within the Address space.

Server URL

The URL is *opc.tcp://< host name or IP address>:<port number>*. By default, the port number is 48030. However, the port number can be configured through the Reviewer's OPC UA settings page. The default URL to connect locally is *opc.tcp://localhost:48030*.

Public

Reviewer licencing

OPC UA server is only available to the Enterprise Edition users with OPC UA license, that is, the EOS DIR License.

OPC UA compliance

The server has been certified by the OPC Foundation to be Embedded 2017 UA Server Profile compliant.

Supported configurations

OPC UA server supports the following configurations:

Feature	Supported Value / Limitation
Session	Min 3 concurrent client sessions
Address space	120 instruments with 12 groups max, and 100 channels per group

System limits

The following are the enforced limitations on the server:

Feature	Supported Value / Limitation
History data	Max history values per read = 9999
Subscription	Min Publishing Interval = 50 milliseconds Publishing Interval Resolution = 50 milliseconds Max Subscription Count = 500 Max Subscriptions Per Session = 100 Max Notifications Per Publish = 25000 Max Message Queue Size = 100

Public

Configuration

The **OPC UA settings** page is accessed from the **Administration** menu.

Configuration Tab

- **Enable** – Check the **Enable** option to start / stop the server.
- **Create compliance test tool tree** – When checked (NOT recommended), this creates additional nodes in the address space that are required by the OPC Foundation for OPC UA compliance testing.
- **Network** – **Port** - The port number used in the server URL (*opc.tcp://< host name or IP address>:<port number>*).
- **Security Policies** – Defines the algorithms and key lengths used for signing, and encryption to establish a secure connection.
    - **None**
      http://opcfoundation.org/UA/SecurityPolicy#None
    - **Basic256Sha256**
      http://opcfoundation.org/UA/SecurityPolicy#Basic256Sha256
    - **Aes128_Sha256_RsaOaep**
      http://opcfoundation.org/UA/SecurityPolicy#Aes128_Sha256_RsaOaep
    - **Aes256_Sha256_RsaPss**
      http://opcfoundation.org/UA/SecurityPolicy#Aes256_Sha256_RsaPss
- **Message Security Mode** – Defines whether signing and encryption is used.
    - **None** - No security is applied.
    - **SignAndEncrypt** - All messages are signed and encrypted.
    - **Sign** - All messages are signed but not encrypted.
    - **Sign plus SignAndEncrypt** - Both messages that are only signed, and messages that are signed and encrypted.

Public

Certificates Tab

This page allows you to manage the server and client certificates.

You can link OPC UA client software certificates to a valid Data Reviewer user. See **Client Certificates** below.

Server Certificate

There is only one server certificate. It can either be a self-signed certificate automatically created by Reviewer, or a certificate authority (CA) signed certificate.

On first start up, Reviewer will automatically create a self-signed certificate. You can click the **New** button to automatically generate another self-signed certificate.

A CA certificate must be manually installed.

To install a CA certificate manually,

1. Copy the CA certificate private key in PFK format to the folder *C:\ProgramData\Eurotherm\Reviewer\OpcUaServer\pki\own\private*.

Note:

- The CA public key file is not used directly by the Reviewer OPC UA server. This must be installed on the client PC.
- The open-source tool *openssl* can be used to convert files between der and pfx format.

OPC UA Client(s)

OPC UA client(s) can use an anonymous authentication method to connect to a Data Reviewer OPC UA Server.

On first connection attempt, the client certificates will appear in the list as not trusted, and the connection will be unsuccessful. To allow connection, click the **Trust** button. The **Reject** button will untrust the certificate and prevent the connection.

Once trusted, to connect to the Data Reviewer OPC UA server via an anonymous connection, click the **Link to user** button. In the **Users** dialog box, you can search for a user and link it to the selected client certificate. This same user account will be used for all anonymous connections in the future.

Public

User Certificates Tab

OPC UA client(s) can use a user certificate authentication method to connect to a Data Reviewer OPC UA Server.

On first connection attempt, the user certificates will appear in the list as not trusted, and the connection will be unsuccessful. To allow the connection, click the **Trust** button. Clicking the **Reject** button will untrust the certificate and prevent the connection.

Once trusted, to connect to the Data Reviewer OPC UA server via a user certificate, click the **Link to user** button. In the **Users** dialog box, you can search for a user and link it to the selected user certificate. The same user account will be used for all user certificate connections in the future.

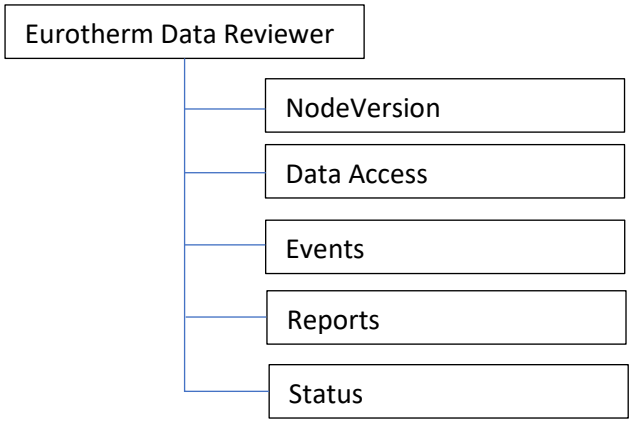
Notes:

- To allow user certificates, both the OPC UA client certificate and the user certificate must be Trusted.
- When you use user certificates, there is one certificate per user using the same OPC UA client and the connection will be trusted using different users on the Data Reviewer.

Public

Address space

This node displays all Reviewer data.



NodeVersion

A string containing an integer is incremented each time the nodes in the address space are created or deleted. For example, if a new instrument had been added, or the Views had been reconfigured.

When the address space is changed, an OPC UA event is also generated.

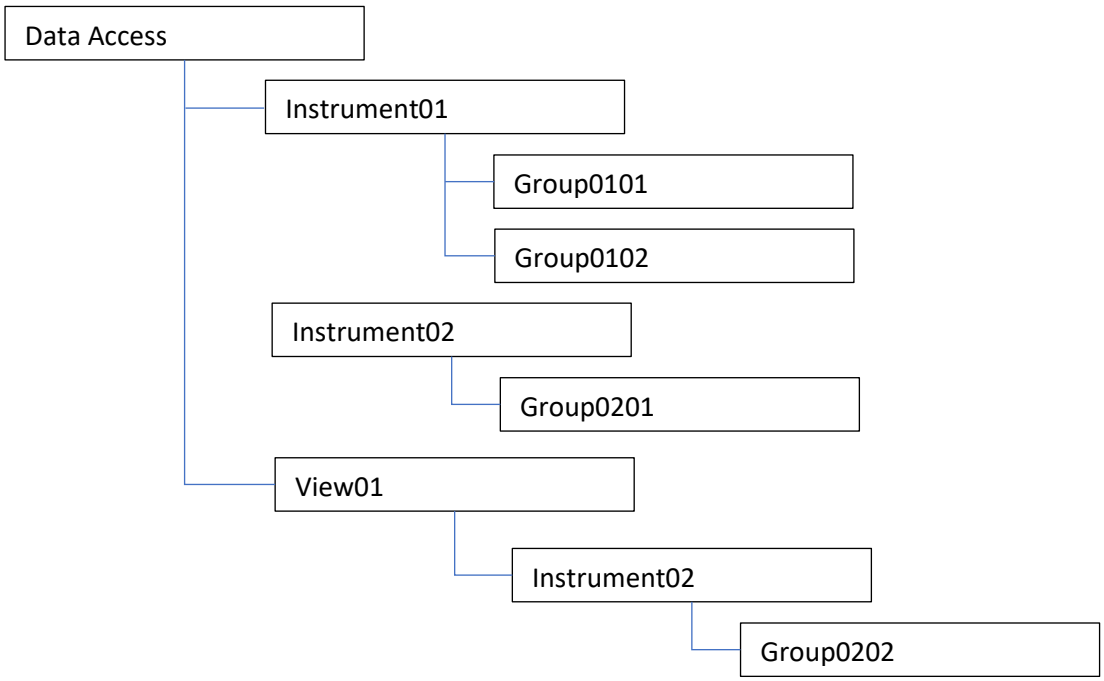
Public

Data Access

This node contains the instrument data.

The top level of the tree shows the Instruments, Groups and Views. If a group is unassigned to a View, it is shown directly below the Data Access node. Otherwise, it is shown below the View.

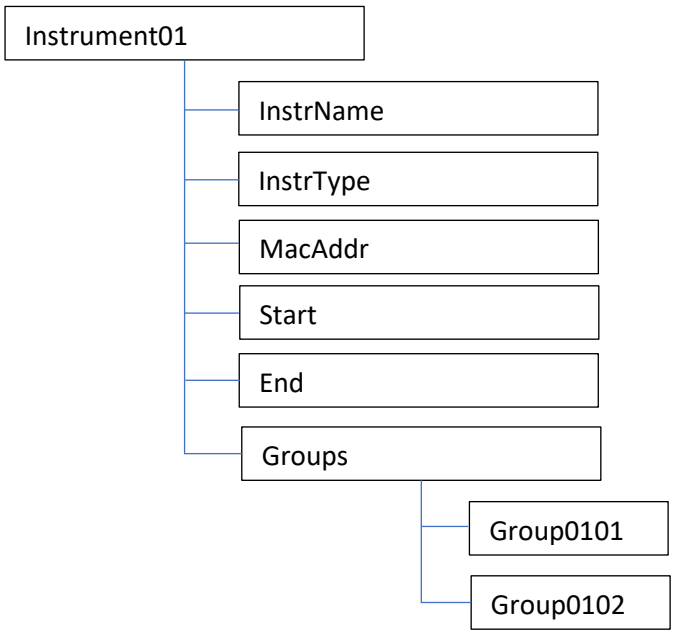
For example, in the tree below Instrument01 has two groups, neither of which is assigned to a View, whereas Instrument02 also has two groups one of which is assigned to View01.



Public

Instrument Node

The ReviewerInstrument object and subtree contains the instrument and group configuration. The example below shows the group node Instrument01and its subtree.

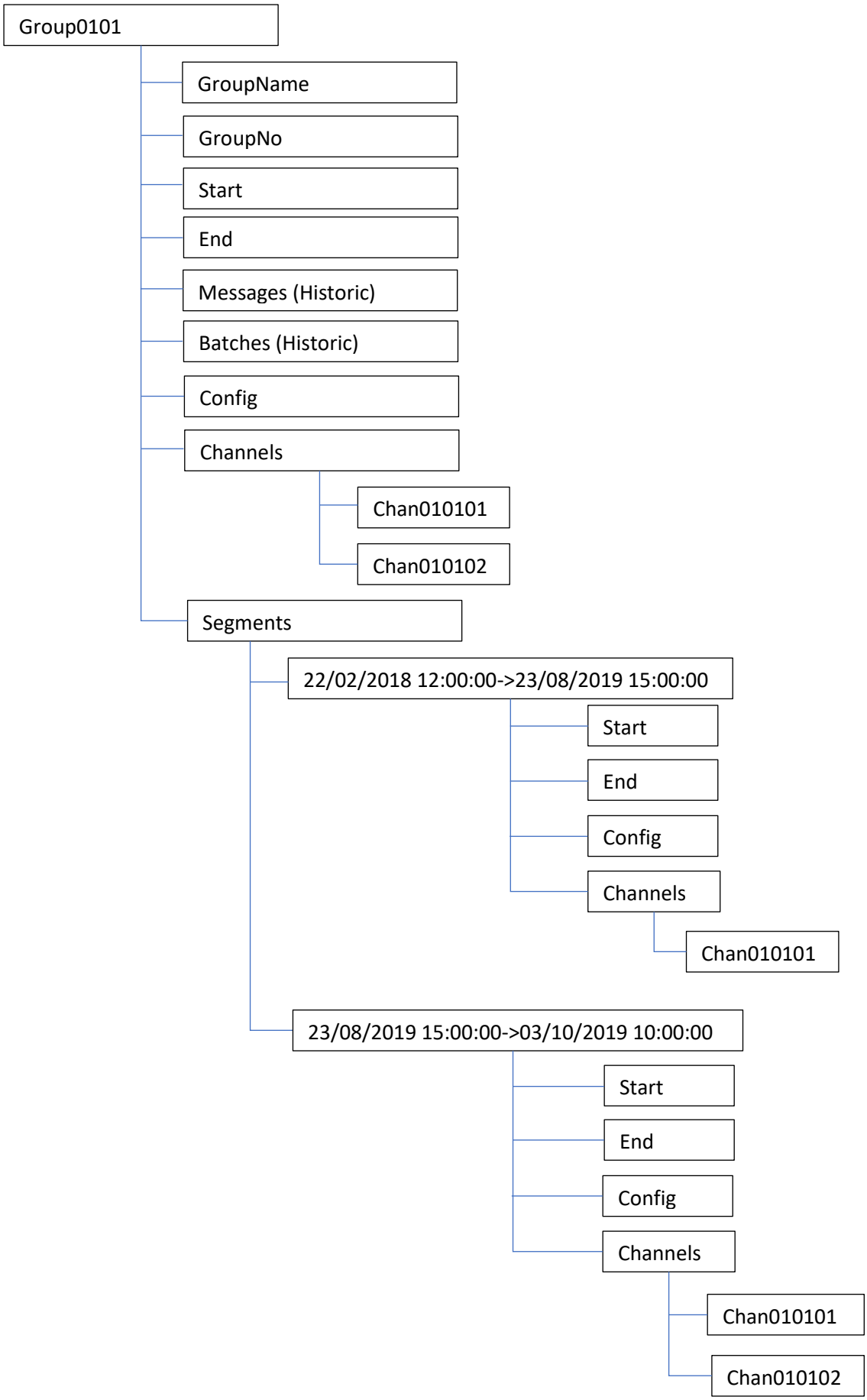


InstrName, InstrType and MacAddr identify the group. Start and End return the dates of the earliest and latest data for the instrument. Groups contains the ReviewerGroup objects associated with the instrument.

Group Node

The ReviewerGroup object and subtree contains the group and channel configuration. The example below shows group node Group0101 and its subtree.

Public

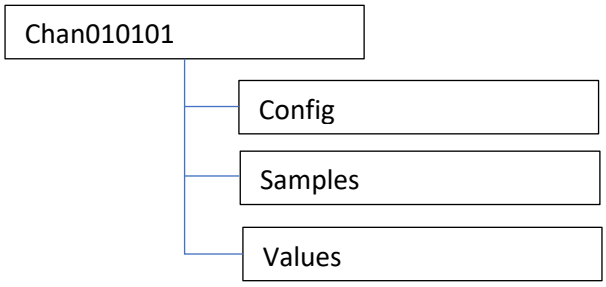

Group0101
GroupName
GroupNo
Start
End
Messages (Historic)
Batches (Historic)
Config
Channels
Chan010101
Chan010102
Segments
22/02/2018 12:00:00->23/08/2019 15:00:00
Start
End
Config
Channels
Chan010101
23/08/2019 15:00:00->03/10/2019 10:00:00
Start
End
Config
Channels
Chan010101
Chan010102

Public

- GroupName and GroupNo identify the group.
- Start and End return the dates of the earliest and latest data for the group.
- Messages return a ReviewerMessage structure the last Message for the Group, but also can return historic messages between specified start and end dates.
- Batches return a ReviewerBatch structure the last Batch, but also can return historic batches between specified start and end dates.
- Config returns a ReviewerSegmentConfig structure containing the group's current configuration.
- Channels contains the currently configured ReviewerChannel objects.
- Segments contains the groups historical configurations. Groups may be reconfigured in many ways. Typically, channels have been added or removed. Each group configuration, including the current configuration is known as a segment and is represented by a node named according to its start and end date.
- Each Segment node contains the Config in the ReviewerSegmentConfig structure and the Channels. The Config and Channels in the latest segment is a copy of those in the group node.

Channel Node

The ReviewerChannel object contains the configuration and values for the channel. The example below shows channel node Chan010101 and its subtree.



- Config is a ReviewerChannelConfig structure containing the channel configuration.
- Samples returns a ReviewerChannelSample structure containing the last value and all associated data. For example, the value status and the color of the instrument used to display the value. It can also return a history between specified start and end dates.
- Values returns a 64bit floating point value containing the channel's last value. It can also return a history between specified start and end dates.

Public

Events

The Events node tree lists all events from the Reviewer Event Log. The Events node generates an OPC UA event each time the log is updated. The sub nodes under Events node filter on different Event types.

- Account creation
- Administration
- All
- Annotation
- Email validation
- Login
- Logout
- Notification
- Other
- Password change
- Password reset
- Point properties
- Print config change
- Reporting
- Role management
- Server management
- Token
- User signature

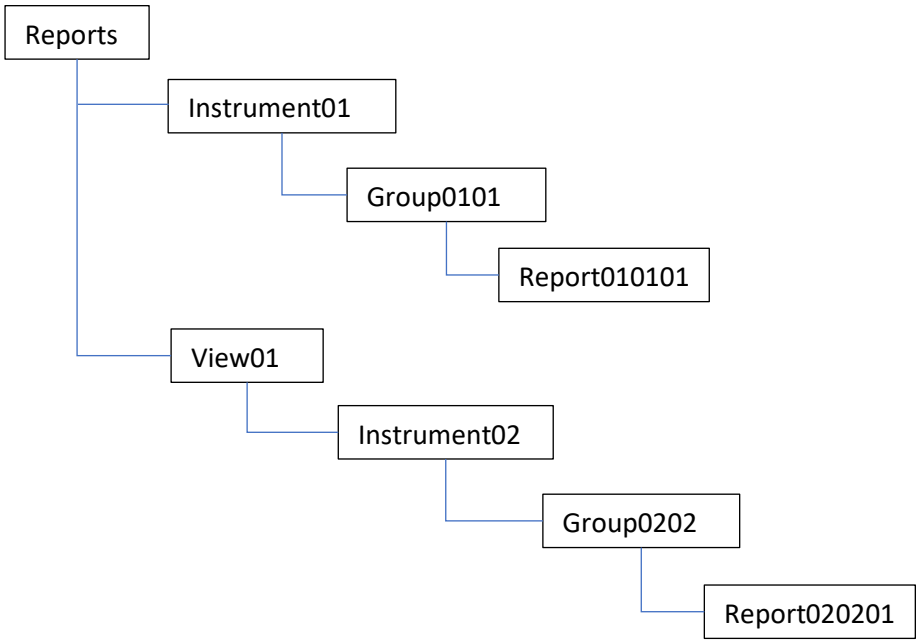
The Events property returns a ReviewerEvent structure containing the last event, or a history between specified start and end times.

Public

Reports

The Reports uses the same Instrument / Group / View tree structure as the Data Access to show groups with scheduled reports.

The example below shows two groups Group0101 and Group0202 with scheduled reports.



Calling the Run, RunWithDates, RunWithEndDate methods in the Report nodes will run the report. These will run the defined reports either at the current time, for a specified start and end time, or for a specified end time.

Status

This node contains system status information such as:

- Elasticsearch Version
- Reviewer Version
- Revision
    - ConfigurationRevision
    - SecurityRevision
- Windows Version
- Windows build

The ConfigurationRevision and SecurityRevision both generate OPC UA Events when they change.

Public

OPC UA Event Generation

OPC UA events are triggered when the,

- Address space changes (NodeVersion incremented) – for example, when a new group is imported.
- Reviewer Event Log is updated – for example, when a user log one.
- Reviewer configuration revision is incremented – for example, when the Global Settings are updated.
- Reviewer security revision is incremented – for example, when a new user is added.

Action	Object	Event Type	Message
Address space changes	Server	GeneralModelChangeEventType	The address space has changed
Reviewer Event Log updated	Events	ReviewerEventLogEvent	New event log
Reviewer configuration revision is updated	Status	ReviewerRevisionEvent	Revision number was changed
Reviewer security revision is updated	Status	ReviewerRevisionEvent	Revision number was changed

Some actions generate more than one event, for example,

- Editing the Global Settings generates
    - a Revision change event (the Configuration Revision is incremented)
    - an Event Log event
- Editing View Management generates
    - a Revision change event (the Security Revision is incremented)
    - an Event Log event
    - an Address Space Change event

Public

Reviewer Structures

Segment Configuration

This includes values returned by the Config variable in the ReviewerGroup and ReviewerSegment objects.

ReviewerSegmentConfig

- SWversion: Software version of instrument
- IntervalA: Primary nominal recording interval, in milliseconds
- IntervalB: Secondary nominal recording interval, in milliseconds
- TrendUnits: Trend units
- Language: Language
- Country: Country
- TzId: Time zone Id
- TzOffset: Timezone Offset
- TzDstStart: Daylight Saving Time start rule
- TzDstEnd: Daylight Saving Time end rule
- TzDstUse: 1 = Use DST, 0 = don't
- GridType: 0 = None, 1 = Linear, 2 = Log, 3 = From Point
- GridDivsOrDecades: Number of major divisions in Linear grid, or number of decades in Logarithmic grid
- GridMinorDivs: Number of minor divisions in Linear grid
- GridScaleType: Point Scale type: 0 = None, 1 = Linear, 2 = Log
- GridSpanLo: Point Low span limit
- GridSpanHi: Point High span limit
- GridZoneLo: Point Low zone limit
- GridZoneHi: Point High zone limit
- GridSpanLoB: Point Low span B limit
- GridSpanHiB: Point High span B limit
- GridZoneLoB: Point Low zone B limit
- GridZoneHiB: Point High zone B limit
- OtherMacAddr: The MAC address for the other member of a redundant pair. All zeroes if simplex.
- OtherInstAddr: The instrument address (see History Header record) for the other member of a redundant pair. All zeroes if simplex.
- GridScaleTypeText: As above
- GridTypeText: As above

Public

Message

This includes values returned by the Messages variable in the ReviewerGroup object.

ReviewerMessage

- Time: Timestamp of message
- Message: The message
- Category: Unknown = 0,
  Batch = 1,
  AlarmAbsoluteLow = 2,
  AlarmAbsoluteHigh = 3,
  AlarmDeviationIn = 4,
  AlarmDeviationOut = 5,
  AlarmRateOfChangeRise = 6,
  AlarmRateOfChangeFall = 7,
  AlarmDeviationLow = 8,
  AlarmDeviationHigh = 9,
  AlarmDigital = 10,
  AnnotationAnnotate = 11,
  AnnotationApproved = 12,
  AnnotationReviewed = 13,
  AnnotationReleased = 14,
  System = 15,
  Power = 16,
  General = 17,
  Login = 18,
  Signing = 19,
  Audit = 20,
  Report = 21,
  Ams2750 = 22,
  Alarm = 23,
  AnnotationFail = 24,

- Sequence: Numeric sequence of message in UHH file
- CatList: Array of UHH message categories
  System = 0x01,
  Alarm = 0x02,
  Power = 0x04,
  General = 0x08,
  Batch = 0x10,
  Login = 0x20,
  Signing = 0x40,

Public

Audit = 0x80,
Report = 0x100,
Ams2750 = 0x200

- AlarmType: Type of alarm (if message linked to alarm)
  0 = ABS_LOW, 1 = ABS_HIGH,2 = DEV_IN, 3 = DEV_OUT,4 = ROC_RISE, 5 = ROC_FALL
- AlarmTypeText: As above
- AlarmActive: True or false
- AlarmAck: True or false
- AnnotationType: None = 0, Annotate = 1, Approved = 2, Reviewed = 3, Released = 4, Fail = 5
- AnnotationTypeText: as above

Public

Batch

This includes values returned by the Batches variable in the ReviewerGroup object.

ReviewerBatch

- StartTime: The time that the batch was triggered
- StartActionedBy: Details of the user that started the batch
- Fields: Array of ReviewerBatchField structures
- StopTime: The time that the batch was stopped
- StopActionedBy: Details of the user that stopped the batch
- Phases: Array of ReviewerBatchPhase structures

ReviewerBatchField

- m_Descriptor: Title of Field (e.g., BatchId)
- Data: Data (e.g., the Id of the batch)

ReviewerBatchPhase

- Time: The time that the new batch phase started
- PhaseNo: Phase number
- PhaseName: The name of the new phase
- ActionedBy: Details of the user that started the phase

Channel configuration

Returned by the Config variable in the ReviewerChannel object.

ReviewerChannelConfig

- ChanNo: Channel number. Note that numbering within each of the 4-point types (real, derived, etc.) is independent.
- ChanType: 0 = ANA_IN, 1 = DIG_IN, 2 = NETWORK_ANA_IN (unused), 3 = NETWORK_DIG_IN (unused), 4 = ANA_OUT (unused), 5 = DIG_OUT (unused)
- AB_Pars: Array of ReviewerChannelABBlock (index 0 = A, index 1 = B)
- PvRep: PV representation:
  Uint8:4 Number of bytes
  Uint8:2 0 = float32, 1 = float64 (2 & 3 reserved)
  Uint8:2 0 = normal, 1 = min-max (2 & 3 reserved)
- Phofsit: PV offset which has been added before recording

Public

- PvFmt: PV display format: 0 = NUMERIC, 1 = DIGITAL, 2 = ELAPSED TIME, 3 = SCIENTIFIC, 4 = TIME, 5 = DATE
- PvDecPl: Number of decimal places to be displayed
- Descriptor: Channel descriptor
- Units: PV units
- Active: Text for digital when active. "" for analogue.
- Inactive: Text for digital when inactive. "" for analogue.
- NumAlms: Number of alarms
- AlmTypes: Uint8:4 Type of alarm1, Uint8:4 Type of alarm 2, etc.
  0 = ABS_LOW, 1 = ABS_HIGH,2 = DEV_IN, 3 = DEV_OUT,4 = ROC_RISE, 5 = ROC_FALL
- ScaleDivsMinorA: Number of minor divisions in a Linear grid for AB_Pars 'A'
- ScaleDivsMinorB: Number of minor divisions in a Linear grid for AB_Pars 'B'
- Block: Name of LIN function block (or a null string if not a LIN function block)
- Field: Name of LIN function block subfield (or a null string if not a LIN function block subfield)
- Subfield: Name of LIN function block subfield (or a null string if not a LIN function block subfield)
- PtType: 0 = Real Channel, 1 = Derived Channel, 2 = Totaliser, 3 = Counter
- ChanTypeText: As above
- PvFmtText: As above
- PvRepText: As above

ReviewerChannelABBlock

- Colour: Pen colour (see colours below)
- Spanlo: Low span limit
- SpanHi: High span limit
- ZoneLo: Low zone limit
- ZoneHi: High zone limit
- ScaleType: 0 = OFF, 1 = AUTO (Linear), 2 = (reserved), 3 = LOG, 4 = LOG_LINEAR
- ScaleDivsMajor: Number of major divisions in a Linear grid
- ScaleTypeText: As above

Public

Colour enumerations

Enumeration	Name
0	Red
1	Blue
2	Green
3	Honey
4	Violet
5	Russet
6	DarkBlue
7	Jade
8	Magenta
9	DuskyRose
10	Yellow
11	PowderBlue
12	DarkRed
13	Avocado
14	Indigo
15	DarkBrown
16	Aegean
17	Cyan
18	Aubergine
19	DarkOrange
20	PaleYellow
21	Hyacinth
22	DarkGreen

Public

23	SugarPink
24	Bluebell
25	Orange
26	Pink
27	Buttersilk
28	Terracotta
29	BlueBabe
30	Lime
31	BlueJive
32	Cucumber
33	EuroGreen
34	Wheatgerm
35	SeaBlue
36	Ginger
37	AquaPool
38	PaleRed
39	PaleBlue
40	Lilac
41	SkyBlue
42	WildMoss
43	Turquoise
44	PaleGreen
45	Coffee
46	Wicker
47	Black

Public

48	DkDkGray
49	DarkGray
50	Gray
51	LtLtDkGray
52	LtDkGray
53	LightGray
54	LtLtGray
55	White

Channel sample

This includes values returned by the Samples variable in the ReviewerChannel object.

ReviewerChannelSample

- ColorB: 0 = Use A Colour, 1 = Use B Colour
- PVData: Channel value
- Status:
  0 = GOOD_PV,
  1 = CHANNEL_OFF,
  2 = OVER_RANGE,
  3 = UNDER_RANGE,
  4 = HW_ERROR,
  5 = RANGING,
  6 = OVERFLOW_PV,
  7 = BAD_PV,
  8 = HW_CAPS_EXCEEDED,
  9 = NO_DATA,
  10 = TOO_MANY_LIN_TABLES
- Time: Timestamp
- StatusText: As above
- SpanZoneScaleB: 0= Use A Zone, 1= Use B zone

Public

Channel sample Min Max recording

Not currently supported.

ReviewerChannelSampleMinMax

- BoolData
- PVDataMax
- PVDataMin
- StatusMax
- StatusMin
- Time

ReviewerEvent

Returned by the Events variable in the ReviewerEvent object.

ReviewerEvent

- **UserId**: User generating the event
- **LogType**: Type of event (e.g., Login)
- **Result**: Result
- **Description**: Description of event
- **ReasonDif**: Change that occurred
- **IpFrom**: IP address

Public