OPC UA Specifications for Sensoft Vision. Version 2.2

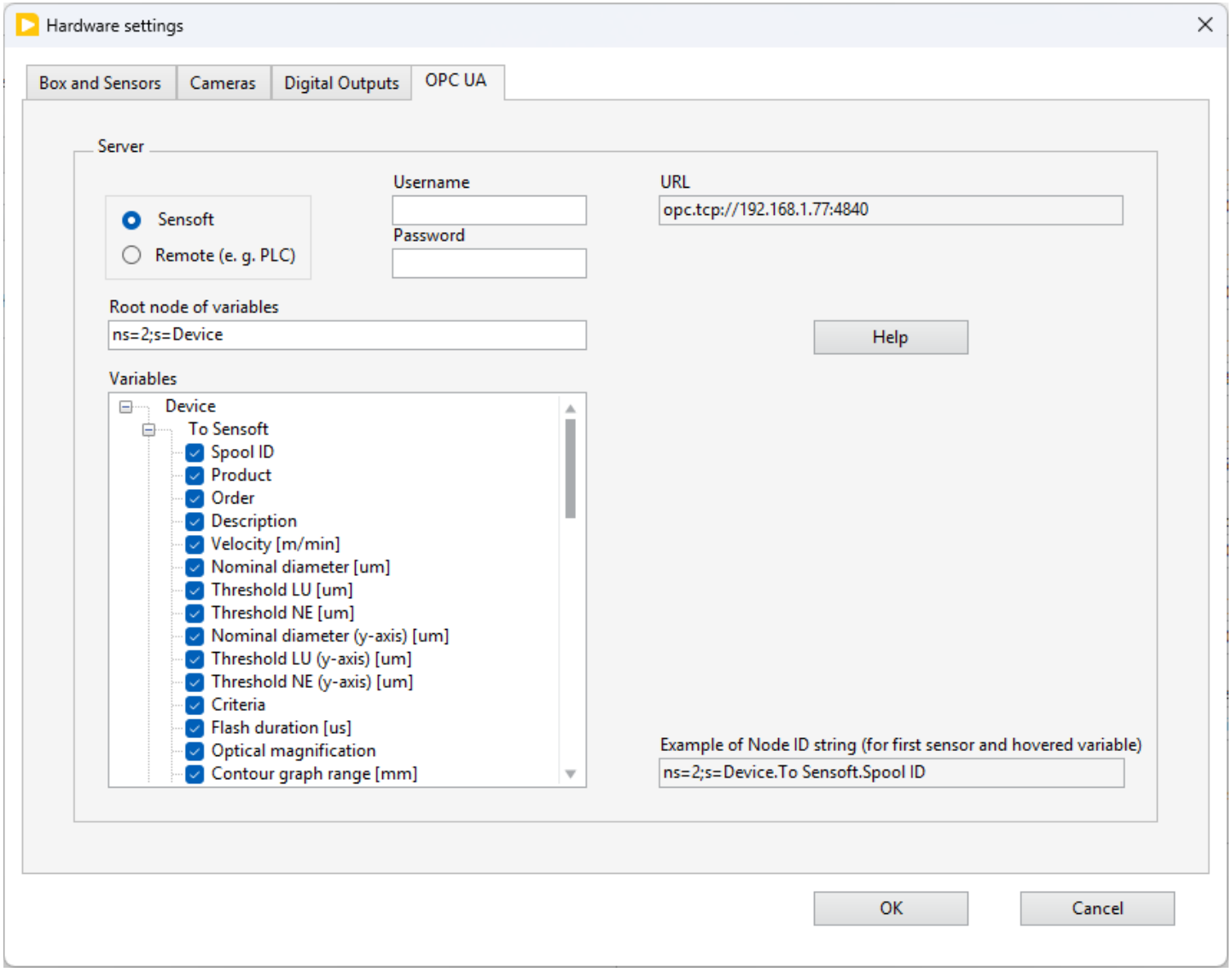


Figure 1: ***OPC UA*** *page of the* ***Hardware configuration*** *dialog window.*

A change log for this document is given at the end.

Node ID structure

Some variables can be written by the remote device (typically a PLC). They are all in one OPC folder node, different from the OPC folder node containing the variables only read by the remote device.

The node ID on the OPC UA server of the variables written by the remote device is:

<Root_node>.To Sensoft.<Var_name>

where:

<Root_node> is the field **Root node of variables** in Figure 1

<Var_name> is the variable name as listed in Chapter **To Sensoft**.

Correspondingly, the variables only read by the remote device have node ID:

<Root_node>.From Sensoft.<Var_name>

where now the variable name is listed in Chapter **From Sensoft**.

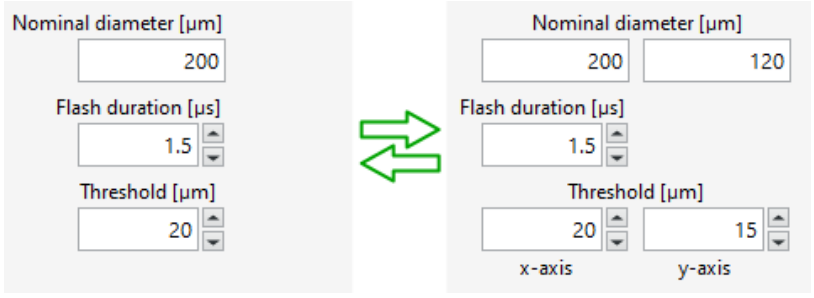
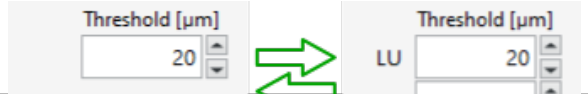
The resulting variable tree can be seen in the field **Variables** in Figure 1. The variables are the rows that have a checkbox. The field **Example of Node ID string (for first sensor and hovered variable)** in Figure 1 displays a resulting node ID, based on the values set in the fields.

Sensoft writes to all OPC variables. It writes to **From Sensoft** variables after measurement events (e.g. when a fault was detected). It writes to **To Sensoft** variables to maintain them consistent with its internal state (see peculiarities at the end of Chapter **To Sensoft**), so that the PLC can also read them.

Variables

To Sensoft

This chapter lists all OPC UA variables that can be written by the remote device and are read by Sensoft.

Variable name	Data type	Description
Spool ID	String (up to 255 characters)	Corresponds to the field **Spool ID** on page **Main** of Sensoft Vision. This field specifies where Sensoft Vision saves the measurement data and supports subfolders. E.g. if **Spool ID** is `Warmup\Product xy\Spool 123456` the measurement data will be saved to **Data folder**`\Warmup\Product xy\Spool 123456` . Note: Changes to **Spool ID** are ignored while a measurement is running (i.e. while the variable From Sensoft.**Measuring** is True), since it affects data saving. It is therefore good practice to check that variable before writing to **Spool ID**. Invalid value handling: The command is ignored while **Measuring** is True
Product	String (up to 255 characters)	The product name or number for the current (if **Measuring** is True) and next spools. The corresponding field in Sensoft Vision is hidden by default. **Product** is written to the daily report file, so that it can be used for data mining in Sensoft Multiline
Order	String (up to 255 characters)	The order number for the current (if **Measuring** is True) and next spools. The corresponding field in Sensoft Vision is hidden by default. **Order** is written to the daily report file, so that it can be used for data mining in Sensoft Multiline
Description	String (up to 255 characters)	A description for the current (if **Measuring** is True) and next spools. The corresponding field in Sensoft Vision is hidden by default. **Description** is written to the daily report file, so that it can be used for data mining in Sensoft Multiline
Velocity [m/min]	Double	Corresponds to the field **Velocity [m/min]** on page **Main**. Note: Sensoft Vision can take the velocity information from different sources (the internal encoder, manual entry in the field, and via OPC UA). The source is specified via OPC UA by the variable **Settings.Velocity source**. If this variable is equal to "0" (= encoder) Sensoft ignores **Velocity [m/min]**, if it is equal to "1" (= manual entry) it reads the variable **Velocity [m/min]** (when the variable **Velocity [m/min]** changes). Invalid value handling: NaN is ignored. The command is ignored unless **Settings.Velocity source** = 1
Nominal diameter [um]	Double	Corresponds to the field **Nominal diameter [µm]** on page **Main**. To clear the field: Set value 0. Note: If **Settings.Thr x equal Thr y** = True sets also **Nominal diameter (y-axis) [um]**. Invalid value handling: Negative values and NaN are ignored  *Figure 2: Fields in Sensoft Vision with **Settings.Thr x equal Thr y** = True (left) and **Settings.Thr x equal Thr y** = False (right).*
Threshold LU [um]	Double	Corresponds to the field **Threshold [µm]** (respectively **Threshold LU [µm]**) on page 

		Main. If (and only if) Sensoft Vision is in Advanced Criteria mode (with the **Criteria** field visible and the threshold fields hidden) **Threshold LU [um]** is zero. Setting a valid value to **Threshold LU [um]** automatically switches to Simple Criteria mode and makes the threshold field(s) visible. Invalid value handling: Negative values, 0 and NaN are ignored *Figure 3: Fields in Sensoft Vision with **Settings.Thr LU equal Thr NE** = True (left) and **Settings.Thr LU equal Thr NE** = False (right). Can be combined with **Settings.Thr x equal Thr y** = False (Figure 2) to show all four threshold fields.*
Threshold NE [um]	Double	Corresponds to the field **Threshold NE [µm]** on page **Main**. **Settings.Thr LU equal Thr NE** is automatically set to False if **Threshold NE [µm]** is different from **Threshold LU [µm]** (see Figure 3). Note: Visibility is not toggled (neither on nor off) if the value you set to **Threshold NE [µm]** is equal to **Threshold LU [µm]**. Attention: Reading this variable may return an outdated value. Invalid value handling: 0 and NaN are ignored. Negative values are converted to positive by taking the absolute value
Nominal diameter (y-axis) [um]	Double	Corresponds to the field **Nominal diameter (y-axis) [µm]** on page **Main**. Available only for sensors with 2 axes. **Settings.Thr x equal Thr y** is automatically set to False if **Nominal diameter (y-axis) [um]** is different from **Nominal diameter [um]** (see Figure 2). To clear the field: Set value 0. Invalid value handling: Negative values and NaN are ignored. Any value is ignored unless sensor has exactly 2 axes
Threshold LU (y-axis) [um]	Double	Corresponds to the field **Threshold [µm] LU (y-axis)** on page **Main**. Available only for sensors with 2 axes. **Settings.Thr x equal Thr y** is automatically set to False if **Threshold LU (y-axis) [µm]** is different from the value of the **Threshold LU [µm]** field (see Figure 2). If equal, visibility is not toggled (neither on nor off). Attention: Reading this variable may return an outdated value. Invalid value handling: Negative values, 0 and NaN are ignored. Any value is ignored unless sensor has exactly 2 axes
Threshold NE (y-axis) [um]	Double	Corresponds to the field **Threshold [µm] NE (y-axis)** on page **Main**. Available only for sensors with 2 axes. **Settings.Thr x equal Thr y** (resp. **Settings.Thr LU equal Thr NE**) is automatically set to False if **Threshold NE (y-axis) [µm]** is different from the value of the **Threshold NE [µm]** field (resp. **Threshold LU (y-axis) [µm])**, see Figure 2 and 3. If equal, visibility is not toggled (neither on nor off). Attention: Reading this variable may return an outdated value. Invalid value handling: 0 and NaN are ignored. Negative values are converted to positive by taking the absolute value. Any value is ignored unless sensor has exactly 2 axes
Criteria	Array (up to 6 elements) of Strings	Corresponds to the field **Criteria** on page **Main**. Faults are defined in Sensoft Vision either by **Criteria** or by **Threshold * [µm]**, and correspondingly either is visible the **Criteria** field or the **Threshold * [µm]** fields. When you write to the variable **Criteria**, the **Criteria** field is automatically made visible, the threshold fields are hidden, and the **Threshold * [µm]** variables are set to zero.

		To determine if the **Criteria** field is visible, test if **Threshold LU [um]** is zero. Invalid value handling: An empty array is ignored. An array of more than 6 elements in truncated, as Sensoft Vision allows no more than 6 criteria
Flash duration [us]	Double	Corresponds to the field **Flash duration [µs]** on page **Main**. Invalid value handling: Negative values, 0 and NaN are ignored
Optical magnification	UInt32	Corresponds the element index of the field **Optical magnification** on page **Main**. The corresponding magnifications values depend on the installed optics, but 0 = "" (no value set), 1 = lowest magnification, ... Invalid value handling: Values larger than the number of available elements are ignored
Contour graph range [mm]	Double	Changes the range of the horizontal axis of the graph on page **Main**. The given value is the full range of the x-axis, which is centered on x = 0. Invalid value handling: Negative values, 0 and NaN are ignored. Values are coerced to be between **Slit width [mm]** (typically 0.25 mm) and 1000 mm
Start	Boolean	Setting it to True corresponds to pressing the **Start** button on page **Main**. The variable is latched back to False by Sensoft. Invalid value handling: False is ignored. The command is not executed if Sensoft is in **Focus adjustment** mode or if it is already measuring. The variable is nevertheless latched back. Note: If there are really many faults (i.e. when the fault threshold was accidentally set below the noise level) it can take Sensoft an arbitrary long time after the **Stop** (or **Stop and discard data**) command to process the queued photos. It is therefore good practice to check that From Sensoft.**Measuring** is False before the **Start** command and checking that it changed to True afterwards
Stop	Boolean	Setting it to True corresponds to pressing the **Stop** button on page **Main**. The variable is latched back to False by Sensoft. Invalid value handling: False is ignored. The command is not executed if Sensoft is in **Focus adjustment** mode or if it is not measuring. The variable is nevertheless latched back.
Stop and discard data	Boolean	Setting it to True corresponds to pressing the **Stop** button on page **Main** and discarding all the measured data, i.e. not saving it to disk. The variable is latched back to False by Sensoft. Invalid value handling: False is ignored. The command is not executed if Sensoft is in **Focus adjustment** mode or if it is not measuring. The variable is nevertheless latched back.
Take and save photo now	Boolean	Setting it to True takes a photo and stores it with the name "<Date>T<Time> Section without defects.jpg" in the same directory as the other photos. The variable is latched back to False by Sensoft. Invalid value handling: False is ignored
Switch to tab	UInt32	Switches to that page. The values are 0 = Main, 1 = Results, 2 = Statistics, 3 = Settings Invalid value handling: Values larger than the number of available elements are ignored

Settings.Mean data interval [m]	Double	Corresponds to the field **Mean data interval [m]** on page **Settings**. Setting the value 0 prevents taking the mean data. Invalid value handling: Negative values and NaN are ignored
Settings.Filament type	UInt32	Corresponds to the field **Filament type** on page **Settings**. The values are 0 = Enameled wire, 1 = Monofilament, 2 = Multifilament Invalid value handling: Values larger than the number of available elements are ignored
Settings.Data folder	String (up to 255 characters)	Corresponds to the field **Data folder** on page **Settings**. Note: It is good practice to set **Data folder** just once, as parent folder of all measurements, and use **Spool ID** to specifying the subfolders. This permits to load all measurements. Changes to **Settings.Data folder** are ignored while a measurement is running (i.e. while the variable From Sensoft.**Measuring** is True), since it affects data saving. Invalid value handling: The command is ignored if **Measuring** is True
Settings.Velocity source	UInt32	Determines from where Sensoft Vision takes the velocity information. The values are 0 = Encoder (internal tachometer), 1 = Manual entry (by the field in the GUI or by OPC UA). Invalid value handling: Values larger than the number of available elements are ignored
Settings.Thr LU equal Thr NE	Boolean	Hide or show separate threshold fields for lumps and neck-downs. Corresponds to clicking with the right mouse button on the threshold field and select **Use different LU and NE thresholds** from the context menu. True corresponds to the situation shown in the left side of Figure 3, False to the right side. **Settings.Thr LU equal Thr NE** is automatically set to False when the LU and NE thresholds are different. Setting **Settings.Thr LU equal Thr NE** = True adapts **Threshold NE [um]** and **Threshold NE (y-axis) [um]**
Settings.Thr x equal Thr y	Boolean	Hide or show separate threshold and diameter fields for x-axis and y-axis. Available only for sensors with 2 axes. Corresponds to clicking with the right mouse button on the diameter or threshold field and select **Use different thresholds for x and y** from the context menu. True corresponds to the situation shown in the left side of Figure 2, False to the right side. **Settings.Thr x equal Thr y** is automatically set to False when the y-axis diameter or thresholds are different from those of the x-axis. Setting **Settings.Thr x equal Thr y** = True adapts **Nominal diameter (y-axis) [um]**, **Threshold LU (y-axis) [um]** and **Threshold NE (y-axis) [um]**. Invalid value handling: Ignored unless sensor has exactly 2 axes.

Sensoft is subscribed to the variables listed in **To Sensoft**, i.e. receives the variable when its value on the server changes. The publishing interval is 50 ms, which means that changes happening faster are not received. Sensoft treats changes to the values of the OPC UA variables like manual changes to the corresponding fields in Sensoft. This means in particular that manual changes are not disallowed. The variables in **To Sensoft** are updated by Sensoft when the value of the corresponding field changes, both if the change originated from a manual change and from an OPC change. The exceptions are latching Booleans and **Velocity [m/min]**. The **To Sensoft** variables are consistent with the local variables in Sensoft, with the following peculiarities:

- **Start**, **Stop**, **Stop and discard data** and **Take and save photo now** do not correspond to a local variable. They are latching Booleans, i.e. written by Sensoft just to latch them back to False after an OPC change. To read if a measurement is going on use **From Sensoft.Measuring**
- **Velocity [m/min]** is never written to OPC UA by Sensoft, in order not to interfere with a possibly fast-changing variable. To read the velocity use **From Sensoft.Velocity [m/min]**

- **Threshold NE [um]**, **Threshold LU (y-axis) [um]** and **Threshold NE (y-axis) [um]** should not be trusted when read from the remote client. Writing works. The reason is that for **Settings.Thr LU equal Thr NE** = True or **Settings.Thr x equal Thr y** = True these variables are dependent from each other and **Threshold LU [um]**. This dependence is honored for the local variables of Sensoft, but not for the OPC UA variables, because it could interfere with your OPC UA write commands to these variables.

When Sensoft starts in OPC UA mode or changes to OPC UA mode, its updates all **To Sensoft** variables to match their corresponding fields. It updates also the **From Sensoft** variables **Status**, **Measuring** and **Line name**, while the others are only updated while measuring.

From Sensoft

This chapter lists all OPC UA variables that can be read by the remote device and are written by Sensoft.

Variable name	Data type	Description
Last fault.Nr	Int32	The number of the last fault, i.e. also the number of faults since the beginning of the measurement. Supports Historical access, see note 1. One cannot rely on Last fault.Nr being consecutive, see note 2
Last fault.Time	DateTime	Time of the last fault. Supports Historical access, see note 1
Last fault.Position [m]	Double	Position of the last fault. Supports Historical access, see note 1
Last fault.Size [um]	Double	Size of the last fault. Supports Historical access, see note 1
Last fault.Type	UInt32	Type of the last fault (0 = Lump, 1 = Neck-down). Supports Historical access, see note 1
Last fault.Velocity [m/min]	Double	Speed at the moment of the last fault. Supports Historical access, see note 1
Last fault.Length [mm]	Double	Length of the fault. Supports Historical access, see note 1
Last fault.Has photo	Boolean	True if a photo of the defect was taken. Supports Historical access, see note 1
Last fault.Has graph	Boolean	True if the profile graph of the defect was recorded. Supports Historical access, see note 1

Last fault.Diameter [um]	Double	Reserved for future use. Supports Historical access, see note 1
Measurement start time	DateTime	Start time of the latest measurement
Measuring	Boolean	True while measuring
File path	String (up to 255 characters)	Path of the TDMS file, relative to **To Sensoft.Settings.Data folder** . The TDMS file contains all measured data and information about the spool. The photos are in the same folder as **File path** and their file name can be created with **Last fault.Time** and **Last fault.Nr** using the format string "%<%Y-%m-%dT%H_%M_%S>T Fault %d.jpg"
Line name	String (up to 15 characters)	The name of the line (i.e. the sensor) as specified in the field **Line name** on page **Box and Sensors** of dialog **Hardware settings...**
Position [m]	Double	Current position of the wire, i.e. wire length since the start of the measurement. Updated only while measuring, every 150 ms
Velocity [m/min]	Double	Current velocity of the wire. Updated only while measuring, every 150 ms
Signal [%]	Double	Current signal level, in percent of the expected signal. A sensor with clean optical windows has signal > 90%, below 80% it requires cleaning. Requires non-zero **Nominal diameter [um]**. Updated only while measuring, every 150 ms
Status	UInt32	Status of the Sensoft Vision software. The values are 0: Not running, 1: Running and ok, ≥ 2: Running with problems. The bits of the variable **Status** codify which problems the software has. For more details, please refer to sensoptic.ch/sensoft/vision/how-to/status
Mean data	Array [up to 32] of Double	Last Mean data values. Mean data is periodic data about the spool, written to disc every **Settings.Mean data interval [m]** and displayed in Sensoft on page **Statistics** in **Mean graph**. This variable is an array containing the following elements: Position [m], Relative diameter [um], Relative ovality [um], Max LU [um], Max NE [um], NaN, Uncompensated $Ø_x$ [um], Uncompensated $Ø_y$ [um] (, Uncompensated $Ø_z$ [um]). More elements could be added in the future, but the total number of elements will stay below or equal 32. NaN is used for missing elements, e.g. for Relative ovality [um] with uniaxial sensors. For biaxial sensors Relative ovality [um] is signed, so that the the x and y components of Relative diameter [um] can be calculated ($Ø_x$ = Ø + ovality, $Ø_y$ = Ø - ovality). The neck-down value Max NE [um] is negative. Uncompensated $Ø_z$ [um] is omitted except for three-axial sensors. The uncompensated diameters are intended for calibration tests on all axes. The signal levels of all axes can be calculated by Signals [%] = 100*($Ø_u$ - o)/($Ø_n$ - o), where $Ø_u$ is the vector of the uncompensated diameters, $Ø_n$ are the nominal diameters, o = (DCX_COEFF[1], DCY_COEFF[1], DCZ_COEFF[1]) are offsets where the values DCi_COEFF[1] are the first components (of unit µm) of the vectors DCi_COEFF from the calibration file (i = X, Y, Z). The smallest element of Signals [%] is also available in the variable **Signal [%]**.

		Updated only while measuring, normally every 3 seconds. Updates are skipped if there is no data (i.e. if **Settings.Mean data interval [m]** takes longer than 3 s) and advanced if there is much data (more than 100 mean data intervals in 3 s) or the measurement ends.
Alerts.Nr of alarms	Int32	Total number of alarms since the start of the measurement. Spool is a PASS if this variable is zero and a FAIL if greater than zero. While measuring, updated at least every 3 seconds.
Alerts.Nr of warnings	Int32	Total number of warnings since the start of the measurement. While measuring, updated at least every 3 seconds.
Alerts.Alerts	Array[Nr. of criteria] of Int32	Number of warnings/alarms for each criterion. While measuring, updated at least every 3 seconds.
Time on [s]	Double	Time passed since starting on the hardware. In the first 10 minutes measurements can be slightly less precise due to the sensor warming up. **Time on [s]** can also be used as heartbeat. Updated normally every 150 ms, but the update can stall up to 5 s in exceptional cases (e.g. when stopping a measurement with really many faults).

Clients typically read **From Sensoft** variables by subscribing to them, so that the OPC UA server sends them a notification when the value of variable changes. Note that the variables **Last fault.Type**, **Last fault.Has photo**, **Last fault.Has graph** and **Last fault.Length [mm]** often have the same value as for the fault before. To get a notification each time a variable is written to, i.e. even if the value stays the same, at subscription to the variable use DataChangeTrigger = StatusValueTimestamp instead of the default value StatusValue.

Note 1: For accessing data of previous faults, the OPC UA server provides Historical Access to all **Last fault.*** variables. If Sensoft is the OPC UA server, 10'000 old points of each variable are available. A remote server is free on how many samples to give access, if any. Sensoft writes the data of each fault (see note 2) exactly once to the OPC UA server, therefore reading the historical data of **Last fault.*** from the measurement start time (as stored in the variable **Measurement start time**) to now without limiting the number of resulting points, should return the data from the first to the last fault (an array normally of length **Last fault.Nr**, see note 2). The OPC UA Source timestamp of the **Last fault.*** variables is the time of the fault, i.e. equal to **Last fault.Time**.

Note 2: If the system is overloaded by faults, i.e. if there over 100 faults per second for a second or more, a batch of at least 50 faults is not written to OPC UA (and neither to the TDMS file). This means that **Last fault.Nr** can jump by 50 or more units.

Advanced settings

To access the advanced settings for OPC UA in Sensoft, click with the right mouse button in the OPC UA page of Hardware settings (Figure 2) and choose **Show advanced settings** from the context menu.

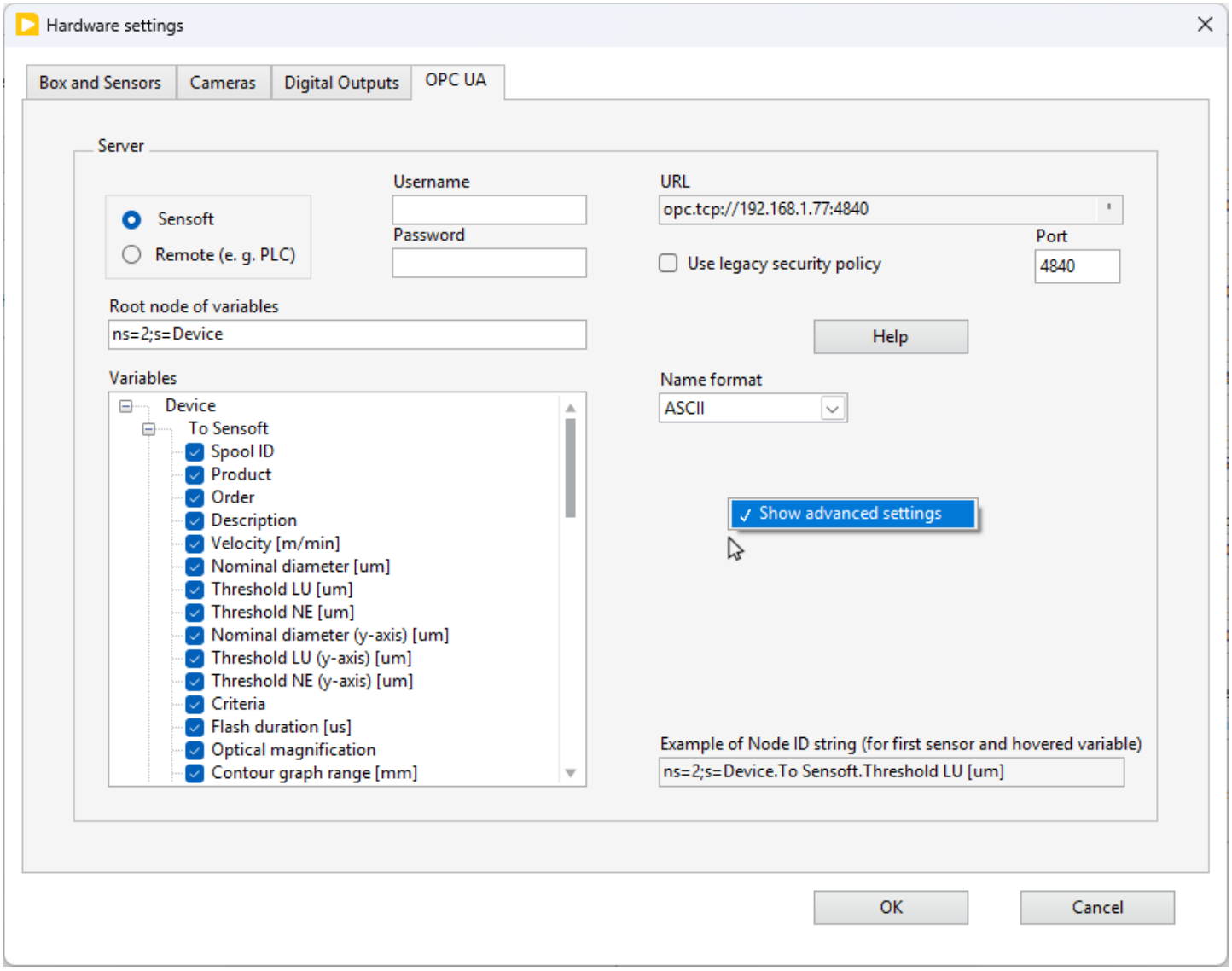


*Figure 2: **OPC UA** page of the **Hardware configuration** dialog window with Advanced settings shown. To show them, click with the right mouse button and select **Show advanced settings** from the context menu.*

Server and client

Sensoft can act either as OPC UA server or as OPC UA client. The user can choose it in the Hardware settings dialog (Figure 2).

The server has to provide the node structure for at least the activated variables. By default, the server is Sensoft. It provides the node structure for all variables.

Port

If Sensoft is OPC UA server, the advanced setting **Port** can be used to give a non-standard port to the OPC UA server. This allows running multiple OPC UA servers on same computer, e.g. or another instance of Sensoft. The setting Port is not used and remains hidden when Sensoft is OPC UA client, there the port is set in field URL.

IEC 61131-3 names

By default Sensoft uses the OPC UA variable names listed in the tables above. They consist of ASCII characters valid in OPC UA. However, some PLCs only support OPC UA variable names made up of IEC 61131-3 characters (i.e. letters, digits, and underscores). Therefore Sensoft can optionally use IEC 61131-3 variable names, listed in the following table. Use the field **Name format** to select IEC 61131-3 (Figure 2). As an example, in that case **To Sensoft.Spool ID** will be called **To_Sensoft.Spool_ID**.

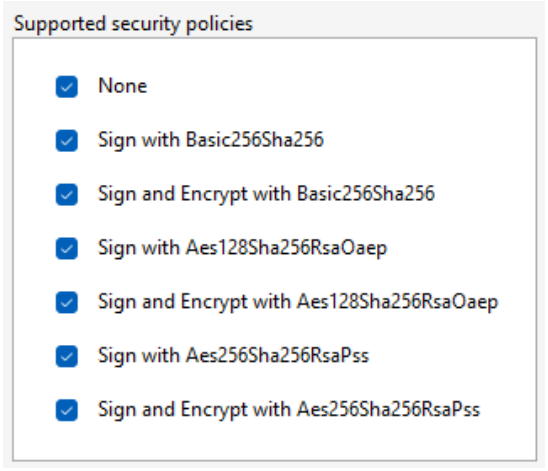
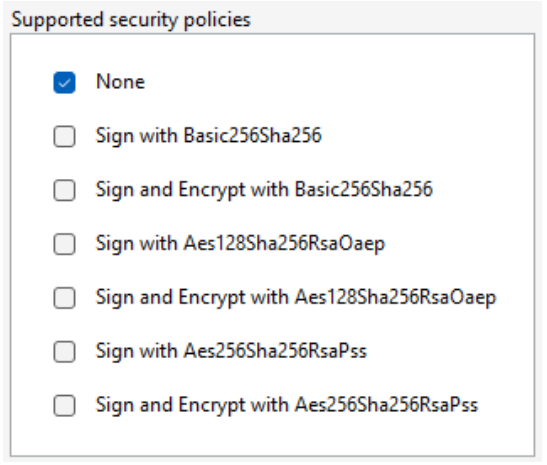
To Sensoft	To_Sensoft	From Sensoft	From_Sensoft
Variable name (ASCII)	**Variable name (IEC 61131-3)**	**Variable name (ASCII)**	**Variable name (IEC 61131-3)**
Spool ID	Spool_ID	Last fault.Nr	Last_fault.Nr
Product	Product	Last fault.Time	Last_fault.Time
Order	Order	Last fault.Position [m]	Last_fault.Position_m
Description	Description	Last fault.Size [um]	Last_fault.Size_um
Velocity [m/min]	Velocity_mpm	Last fault.Type	Last_fault.Type
Nominal diameter [um]	Nominal_diameter_um	Last fault.Velocity [m/min]	Last_fault.Velocity_mpm
Threshold LU [um]	Threshold_LU_um	Last fault.Length [mm]	Last_fault.Length_mm
Threshold NE [um]	Threshold_NE_um	Last fault.Has photo	Last_fault.Has_photo
Nominal diameter (y-axis) [um]	Nominal_diameter_y_axis_um	Last fault.Has graph	Last_fault.Has_graph
Threshold LU (y-axis) [um]	Threshold_LU_y_axis_um	Last fault.Diameter [um]	Last_fault.Diameter_um
Threshold NE (y-axis) [um]	Threshold_NE_y_axis_um	Measurement start time	Measurement_start_time
Criteria	Criteria	Measuring	Measuring
Flash duration [us]	Flash_duration_us	File path	File_path
Optical magnification	Optical_magnification	Line name	Line_name
Contour graph range [mm]	Contour_graph_range_mm	Position [m]	Position_m
Start	Start	Velocity [m/min]	Velocity_mpm
Stop	Stop	Signal [%]	Signal_percent
Stop and discard data	Stop_and_discard_data	Status	Status
Take and save photo now	Take_and_save_photo_now	Mean data	Mean_data
Switch to tab	Switch_to_tab	Alerts.Nr of alarms	Alerts.Nr_of_alarms
Settings.Mean data interval [m]	Settings.Mean_data_interval_m	Alerts.Nr of warnings	Alerts.Nr_of_warnings
Settings.Filament type	Settings.Filament_type	Alerts.Alerts	Alerts.Alerts
Settings.Data folder	Settings.Data_folder	Time on [s]	Time_on_s

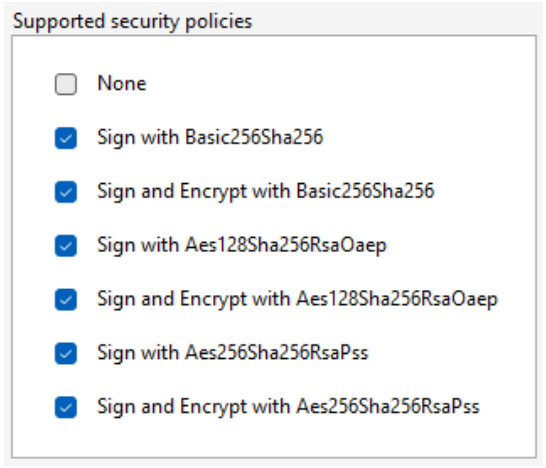
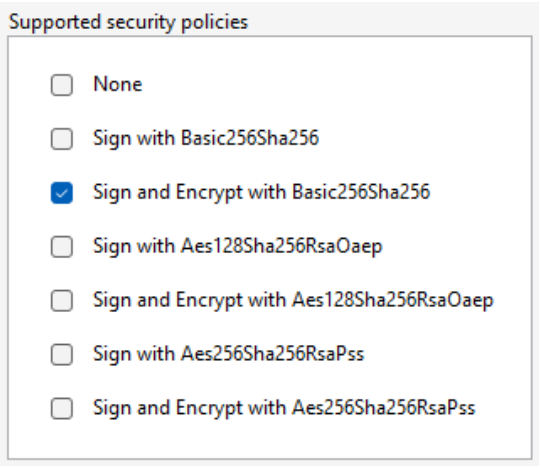
Settings.Velocity source	Settings.Velocity_source
Settings.Thr LU equal Thr NE	Settings.Thr_LU_equal_Thr_NE
Settings.Thr x equal Thr y	Settings.Thr_x_equal_Thr_y

Security

Supported security policies

Sensoft supports the following security policies:

	Sensoft as Server	Sensoft as Client
Without **Username** or **Password**		

With **Username** and **Password**	 **Important**: **Username** and **Password** are not verified, since NI does not support this.	

Legacy security policies (Basic128Rsa15 and Basic256) can be used instead of the current ones, by setting a checkmark near the advanced setting **Use legacy security policy** (Figure 2).

If you need to limit access to the OPC UA server, we recommend to use the Windows firewall to limit the access to port 4840, which is used by the OPC UA server (unless another port is set in field **Port**).

Server certificate files

As of now, no certificate files are used by Sensoft neither as client nor as server. We can implement it if our partners need it.

Changes

V 2.2 (Sensoft Vision 2.2 and later):

- Variables - From Sensoft, variable Status: Changed meaning of bits 1 to 31

V 2.1 (Sensoft Vision 2.1 and later):

- Added Chapter Advanced settings and Figure 2
- Chapter Security: Updated Security policies

V 2.0 (Sensoft Vision 2.0):

- Variables - IEC 61131-3 names: Added option to use variable names with only IEC 61131-3
- Variables - From Sensoft: Added new value to variable From Sensoft.Status to indicate that the system is in focusing mode
- Variables - From Sensoft: Added new components Uncompensated Øs to variable From Sensoft.Mean data
- Variables - From Sensoft: Added variable From Sensoft.Time on [s]

V 1.7 (Sensoft Vision 1.7):

- Variables - To Sensoft.Criteria: Specified that Criteria can have at most 6 elements
- Variables - From Sensoft: Added variable From Sensoft.File path
- Variables - From Sensoft: Added Note 2, i.e. that one cannot rely on Last fault.Nr on being consecutive
- Variables - From Sensoft.File path: Corrected a bug that File path was absolute instead of relative if Data folder is empty
- Security - Supported security policies: Added a note that Sensoft as OPC UA server does not use Username and Password

V 1.6 (Sensoft Vision 1.6):

- Variables - To Sensoft: Added variables Product, Order and Description, which behave as in Sensoft Multiline

V 1.5 (Sensoft Vision 1.5):

- Added support for Advanced Criteria like in Sensoft Multiline: Added variables To Sensoft.Criteria, From Sensoft.Alerts.* . Updated description of To Sensoft.Threshold LU [um]
- Variables - From Sensoft: Added variable Mean data

The version of this document now corresponds to the version of Sensoft Vision

V 1.1 (Sensoft Vision 1.4):

- The variables To Sensoft are now consistent with the local variables in Sensoft, i.e. can also be read, with the restrictions specified at the end of Chapter Variables - To Sensoft
- Variables - To Sensoft: Added variables Nominal diameter (y-axis) [um], Threshold LU (y-axis) [um] and Threshold NE (y-axis) [um]
- Variables - To Sensoft: Reformulated description of Nominal diameter [um], Threshold LU [um] (without semantic change) and Threshold NE [um] (warned that read value may be outdated)
- Variables - From Sensoft: Added variable Signal [%]
- Variables - From Sensoft: Added paragraph after table on subscription to an OPC UA variable

V 1.0 (Sensoft Vision 1.3):

- Initial release