New Features

ibaDatCoordinator v2.3.0

Authors: L. De Mol, R. Kolesnik, G. Schöppner, T. Seitz

Date: 28th Sept 2020

Table of contents

1 Creating offline events

With ibaDatCoordinator v2.3.0 it is possible to create offline-events and thus store data (e.g. computed values) as events in ibaHD-Event-Stores. Writing data to events is supported via the so-called *Offline Event Task*. Such a task can be added to any Job in ibaDatCoordinator.

1.1 Dependencies

Since a lot of different products are involved in order to create offline events, there are various dependencies. For using the complete functionality of the *offline-events*, it is mandatory to install the correct version of the other iba software tools.

ibaHD-Server v2.4.0 (or higher)

In order to write events with ibaDatCoordinator, the ibaHD-Server on the other side needs to support this. The necessary changes for this were implemented in ibaHD-Server v2.4.0. Writing to older versions of ibaHD-Server is not supported.

Note that during the installation of ibaHD-Server v2.4.0 existing ibaHD-Event-Stores are going to be converted automatically into a new structure and existing configurations are going to be adapted. Therefore, a down-grade to an older version ibaHD-Server will cause a loss of data in this case.

ibaPDA v7.2.0 (or higher)

With the new offline event task, ibaPDA is no longer the only tool writing data to ibaHD-Event-Stores. If an event store is used by ibaPDA and ibaDatCoordinator at the same time, a corresponding conflict management is necessary. These changes have been implemented in ibaPDA v7.2.0.

If ibaDatCoordinator is used to write data to an event store which is also used by an older version of ibaPDA this will break your existing configurations and result in a loss of data.

ibaAnalyzer v7.1.7 (or higher)

As usual, the calculations needed for the single tasks in ibaDatCoordinator are done by ibaAnalyzer. For a more convenient configuration and data selection in the context of offline events, some changes were implemented in ibaAnalyzer v.7.1.7.

It is possible to use the offline event task with an older version of ibaAnalyzer, however, some functions will not be available.

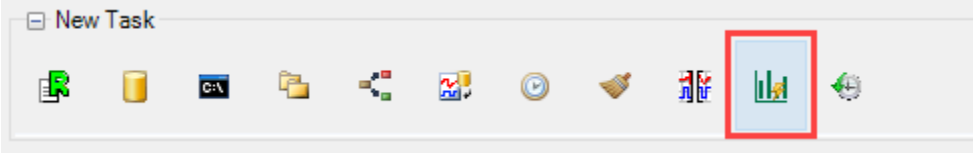
Summary

Table 1 summarizes the minimum software version of involved iba products to get full support of the *offline-events* in ibaDatCoordinator:

Software component	Minimum Software Version
ibaHD-Manager	v2.4.0 or higher
ibaPDA Client	v7.2.0 or higher
ibaDatCoordinator	v2.3.0 or higher
ibaAnalyzer	v7.1.7 or higher

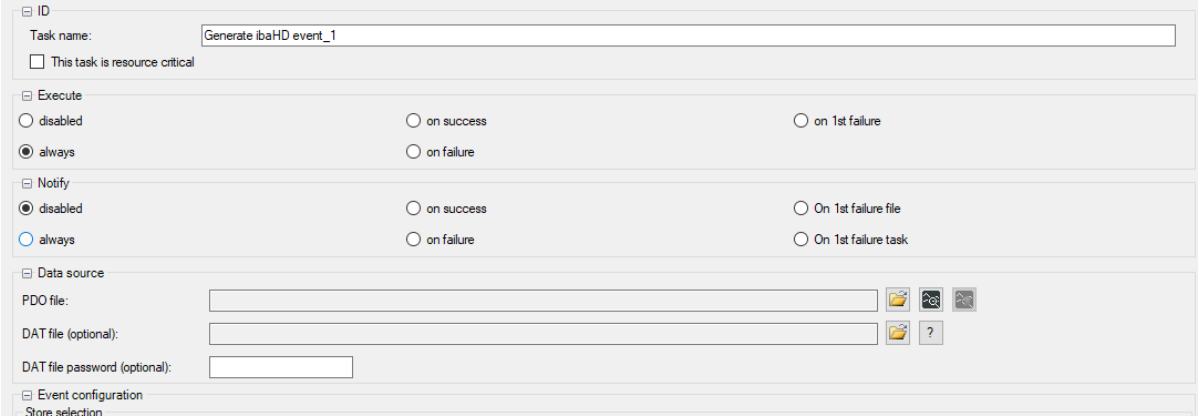
1.2 The Offline Event Task

It is possible to connect an offline event task to any ibaDatCoordinator job.



This is possible for all different job types.

The upper part of the task configuration is the same as for all other ibaDatCoordinator tasks, i.e. job name, execution, and notifications can be configured.

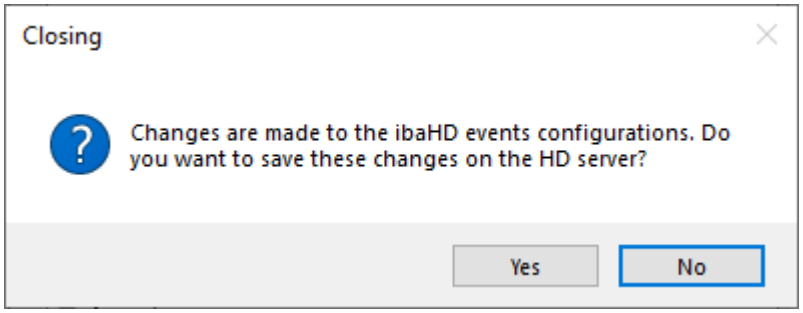


Note that for the next steps it is mandatory to select an appropriate pdo-file where the necessary calculations are done and are available in form of expressions. This can be compared to the extract task, with the difference that every expression and logical in the pdo can be used.

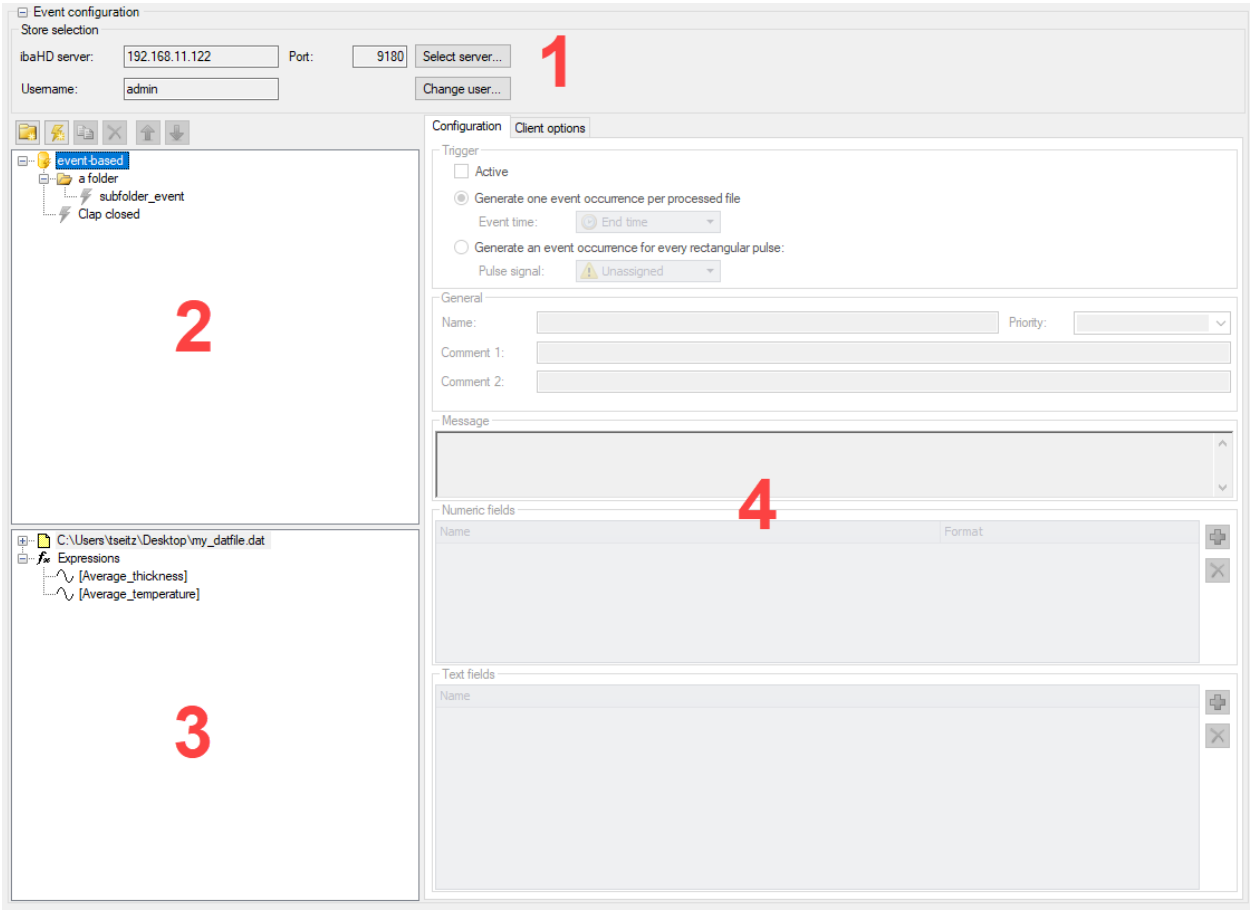
An optional dat-file can be specified, if also raw signal data shall be used for the offline event task. For the final job-processing, these data are then available from the job, i.e. by going through a list of dat-files or HD queries.

1.3 Overview of event configuration

The event configuration requires several steps which are done in different parts of the dialog. Note that changes done here are not saved automatically to the ibaHD-Server. Only if ibaDatCoordinator is closed or another task or job is selected, you are asked to confirm the changes and save them to ibaHD-Server.



The configuration dialog



Part 1

Configure the connection to the ibaHD-Event-Store. This is the same as in any other dialog where a connection to an ibaHD-Server is required.

Part 2

The destination event is selected here. Further, events can be managed, deleted, and newly created. This dialog is similar as in ibaPDA and will be discussed in chapter 1.4.

Part3

The signal tree provided by ibaAnalyzer. All expressions from the pdo are available here. If a dat-file has been selected, also raw signals and infofields are available

Part 4

The actual event configuration is done here. Again, this is similar to ibaPDA and is described in chapter 1.5.

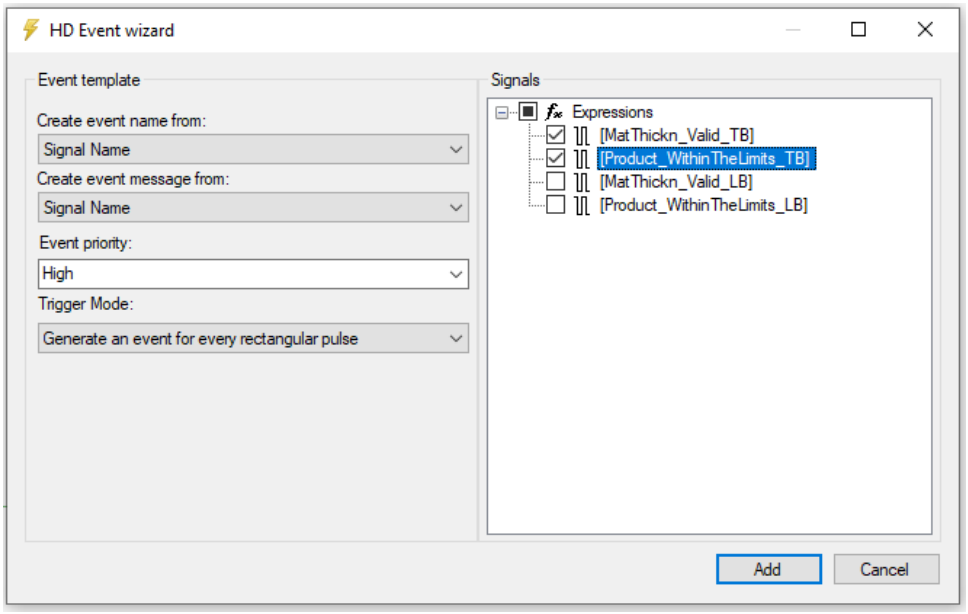
1.4 Managing, editing, and creating events

By using the available buttons

new folders and events can be created, existing events can be copied or deleted, and events can be moved up and down in the list. It is also possible to move events via drag and drop.

The add new event button will create a new empty event that needs to be configured manually, the copy event button will duplicate all currently selected events.

An alternative method of creating events is using the *HD Event wizard*.



This wizard can be opened by dragging signals or expressions from the signal tree (the lower tree) onto the ibaHD event tree (the upper tree). In the event wizard, it is possible to automatically generate and configure events. The event wizard will generate an event for each of the selected digital signals and use this signal as pulse signal. The name, message, priority and trigger mode can be configured and will be applied to every created event.

The ibaHD event tree shows all available events on the connected ibaHD server. It is possible to write data to any of these events from a single offline events task. A visual distinction is made between events that this task actively writes by the color of the flash icon (a yellow flash means this task writes this event, a grey flash means the event is available on the ibaHD server but is currently not written by this task)

Note that if an event shall be used that is currently written by ibaPDA, it is required that this ibaPDA is updated to v7.2.0 or higher. Otherwise this will break existing configurations. See also the remarks about conflict management in the next chapter.

1.5 Adding data to the event

A selected event from ibaHD-event tree is directly configurable in the dialog. Note that writing data must be allowed for the logged in HD user if the user management is activated on ibaHD-Server.

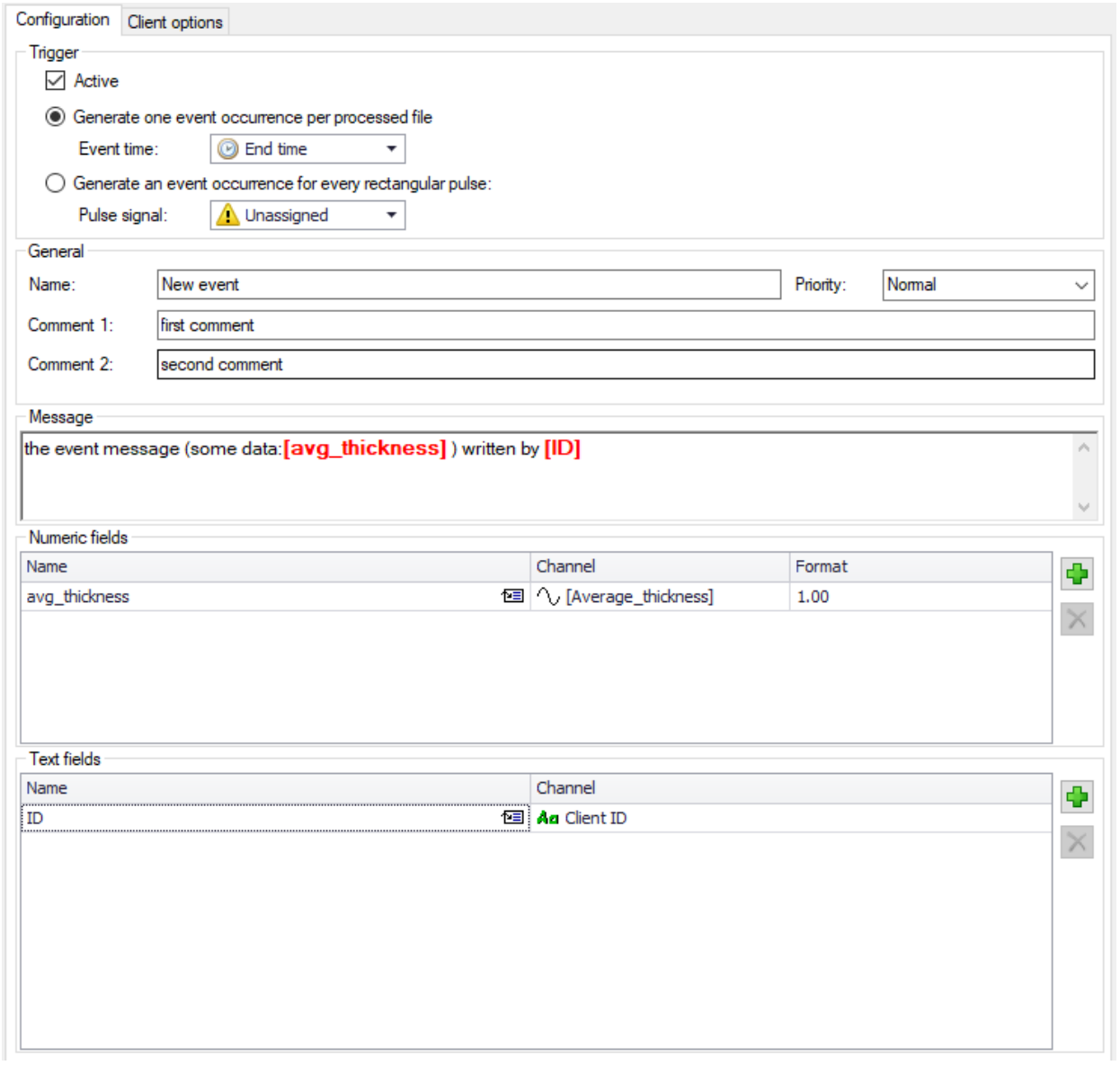
To start writing to an existing event, the ‘Active’ checkbox in the Trigger section needs to be marked. This will also change the color of the flash to yellow. Making a change to the event configuration will also mark this event as active. Newly created events are automatically activated.

Conflict management

It is possible to change the configuration of existing events. These changes will be stored on the ibaHD-server and will apply to all other clients currently using this event. If an ibaPDA client is currently writing to this event, the changes only take effect in the ibaPDA client if the data acquisition is restarted. If a change is made to the event name, numeric or text fields, the ibaPDA client will reject the configuration when simply restarting the acquisition but requires to apply this update in the data storage dialog. This is done as these are changes that will mostly require changes to the local field configuration.

It is highly recommended not to change the configuration of existing events that are being written to by other clients. This could lead to misconfigured events in other clients and also data-loss in some cases.

In the **Configuration** tab of the event, all parameters for an *Offline Event* can to be configured:



Generate one event occurrence per processed file
ibaDatCoordinator creates only one event per dat file or HD data extract. Non-constant numeric values will be automatically averaged over the complete period.

Event time
Determines the timestamp of the created event. There are two predefined fields: “Start time” and “End time” representing the start and end time of the dat file/HD extract. It is possible to select any expression or info field defined in the PDO file from the dropdown or to write an expression manually. These expressions must represent the number of seconds (as floating-point value) starting from the beginning of the processed dat file/HD extract.
Note that all signals are evaluated at the selected time. We therefore recommend to use constant values whenever a specific value needs to be written.

Generate an event occurrence for every rising edge (v2.3.1 or later)
This function will be available in v2.3.1 and may differ from the functionality described here.

ibaDatCoordinator searches in the defined digital signal of the dat-file or HD data extract for rising edges and creates an event for each found pulse. Non-constant numeric values will be automatically averaged over the time-ranges where the pulse signal is true. The time stamp of the event can be configured.

Using this option, it is possible to calculate the KPI for section which are bounded by a rectangular pulse signal. All other raw values outside the TRUE areas of the trigger signal will be ignored.

Trigger signal (v2.3.1 or later)
The signal where the ibaDatCoordinator searches for the rising edges in option ‘Generate an event occurrence for every rising edge’.

Name
The name of the event. This name is displayed in the event tree and also in the event column of the HD event table in ibaPDA.

Priority
Definition of the priority of the new event; the priority can be: High, Normal, Low

Comment 1
The first command field of the event.

Comment 2
The second command field of the event.

Message
The message text, combined with the numeric and/or text fields by adding placeholder with square brackets and the field name.

Numeric Fields
Dynamic signal values which can be added to the event message. The name of the numeric field can be added into the message with the button on the right. The signal or logical expression used for the calculation can be selected via dropdown menu or with drag and drop from the signal tree of the dat file. If the selected signal or logical expression is not constant, the value is the average value of the signal in the time frame of either the dat file, the HD time-range, or the individual periods of the rectangular pulse signal. The Format field defines the number of digits before and after the decimal point for the value which will be added to the message text.

Text Fields
Dynamic text signal values or file names which can be added to the event message. The name of the text field can be added into the message with the button on the right. The signal or logical expression used for the calculation can be selected via dropdown menu or with drag and drop from the signal tree of the dat file.

Additionally, there are two predefined fields available: ‘Processed filename’ and ‘Client ID’. Processed filename will result in the filepath of the processed dat file. Client ID will give the connection details of this application.

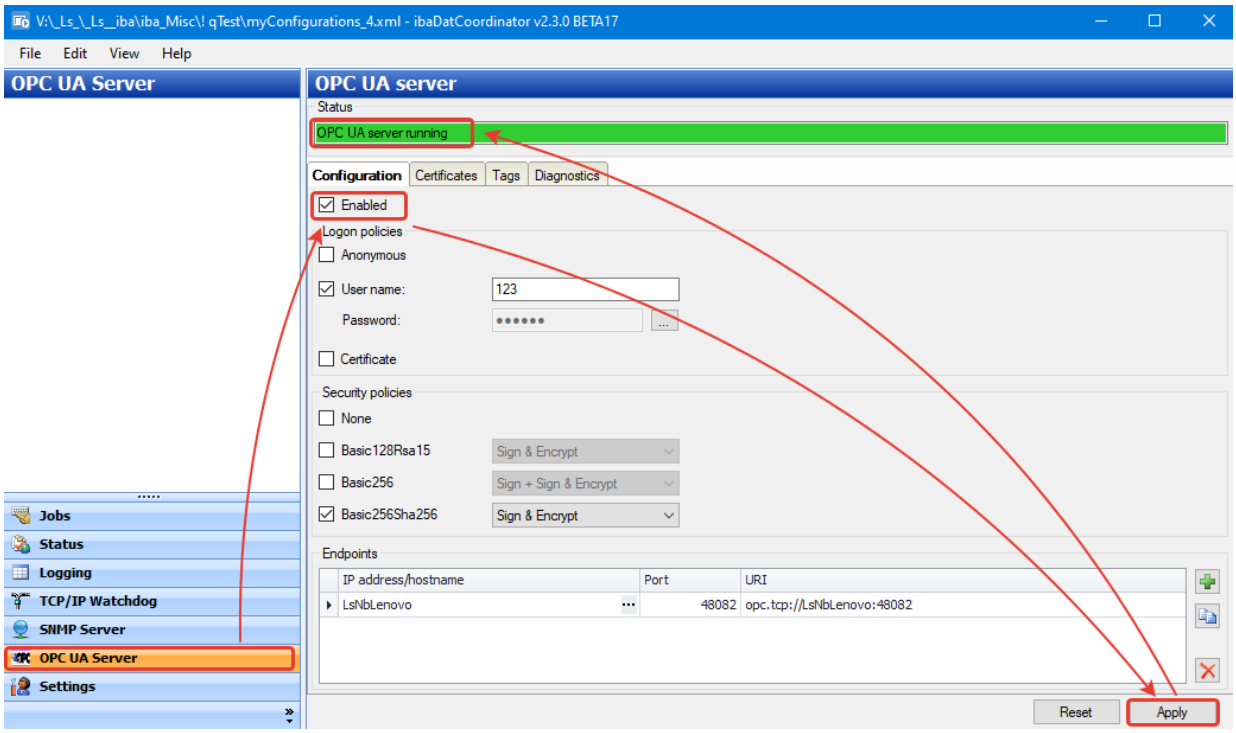
Important note
All used expressions need to be time-based. Length-based signals can be converted to time-based signals using the function*: ConvertBase ([lenghtbased_expression], 1, 0)*

2 OPC UA Server for application info

Monitoring of job/task status and general ibaDatCoordinator status is now available via OPC UA protocol. This feature was designed to work as similar as possible to the SNMP Server for monitoring which is already available.

2.1 Setting up the server

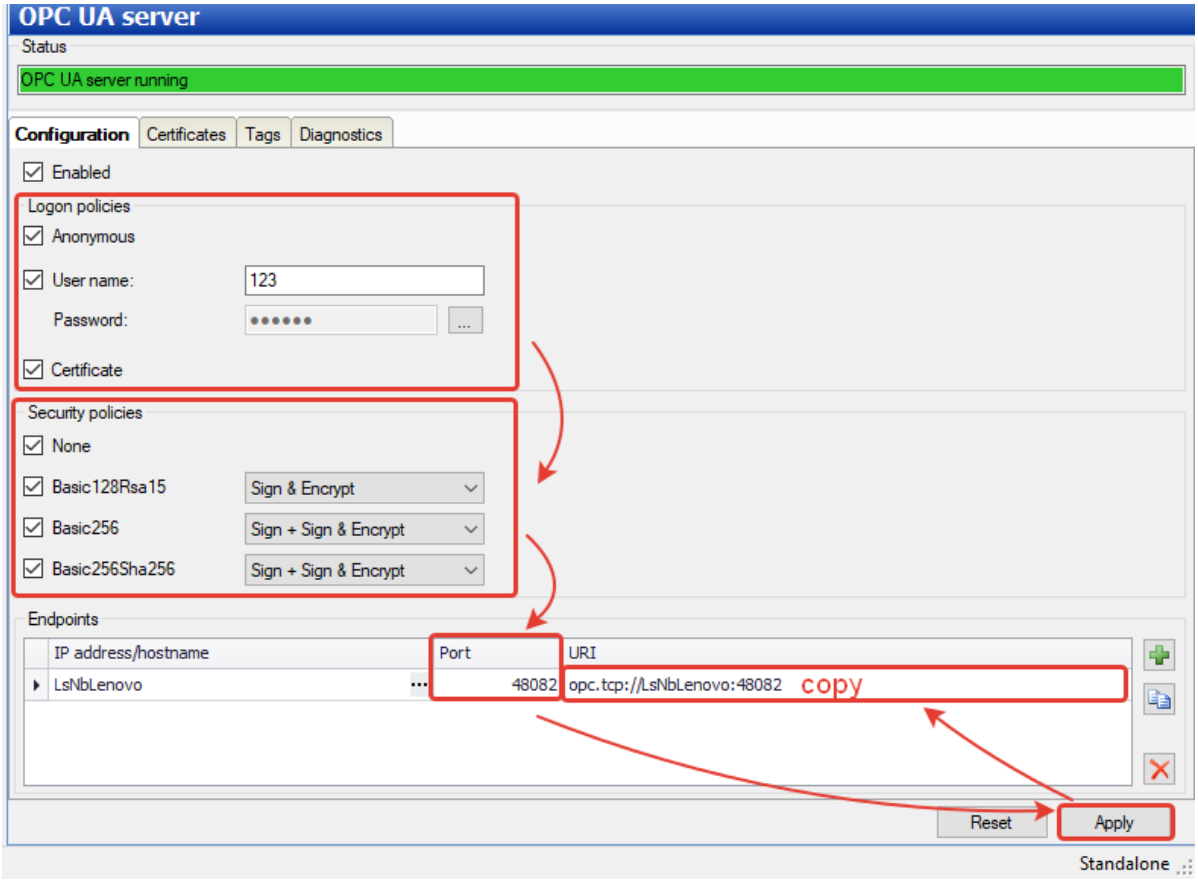
First of all, go to the **OPC UA Server** pane, check the box **Enabled** and click **Apply**. Normally status box should become green with message “OPC UA server running”.



2.1.1 Setting up anonymous mode with no security

For testing purposes, especially if you are not familiar with OPC UA you can enable all logon policies and all security policies. This will give you the maximum compatibility with other OPC UA applications (though at the cost of lowering the security).

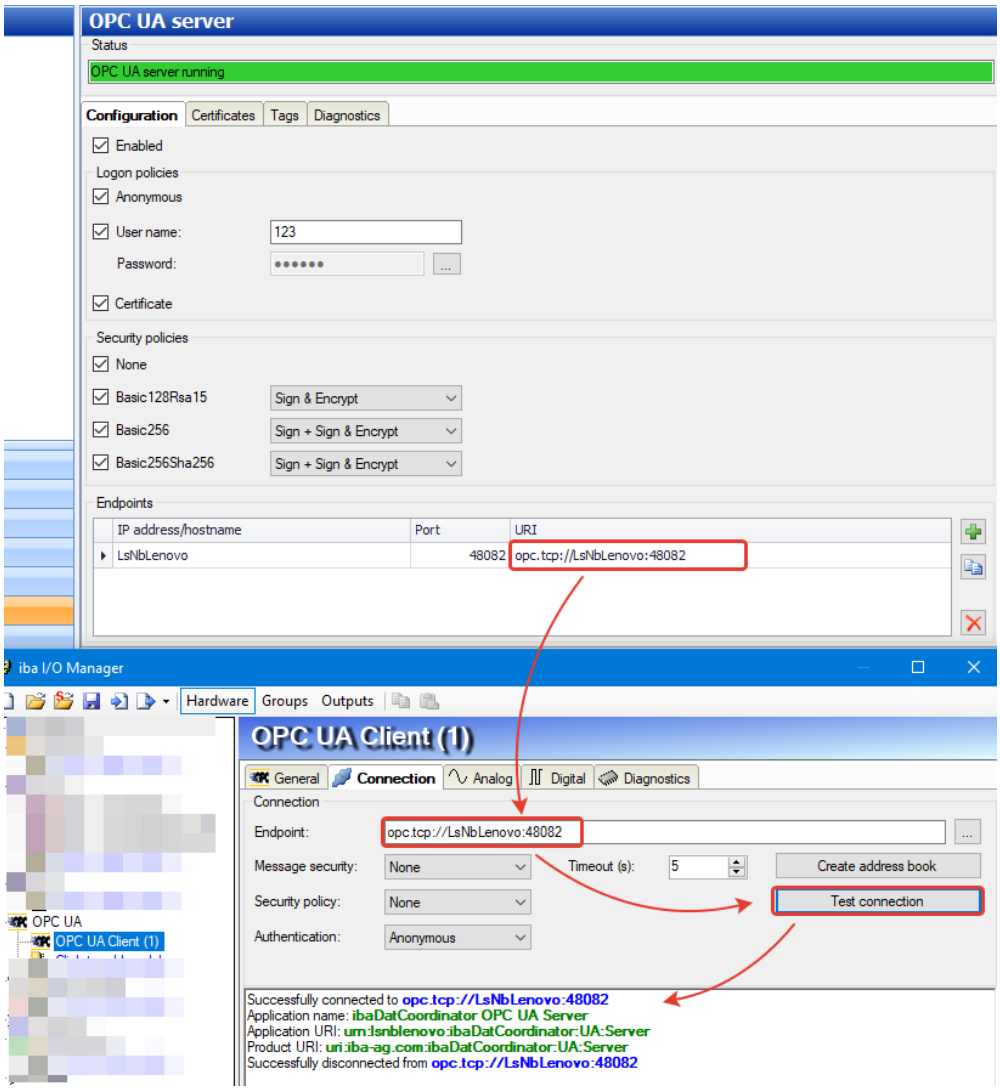
The hostname of your PC should be detected automatically. Also we recommend to check if the port number is not occupied by other apps and is not blocked by firewall.



Try to connect to **ibaDatCoordinator OPC UA Server** using some kind of OPC UA client, e.g.:

- ibaPDA OPC UA Client
- UaExpert (https://www.unified-automation.com/downloads/opc-ua-clients.html)
- Or any other application you're got used to.

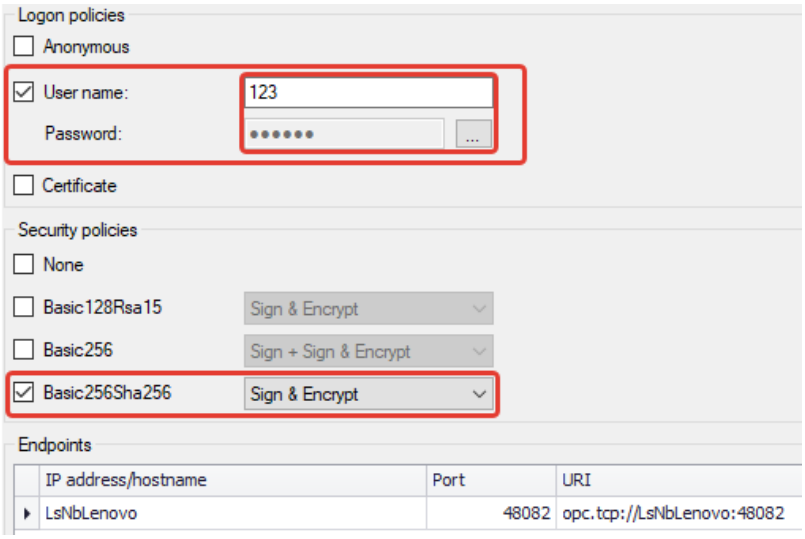
You will need to copy the URI to use it in your client. For example, here is the connection test using ibaPDA OPC UA Client.



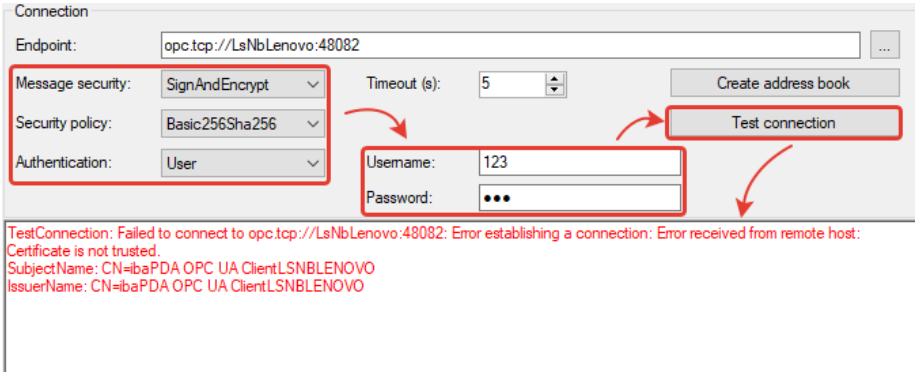
Anonymous logon with **Security None** is the easiest way to configure the connection, but has the lowest security, so, is **not recommended** for normal work.

2.1.2 Setting up secure connection using certificates

After you tested the connection in **anonymous** mode, it's recommended to set the higher security, for example:

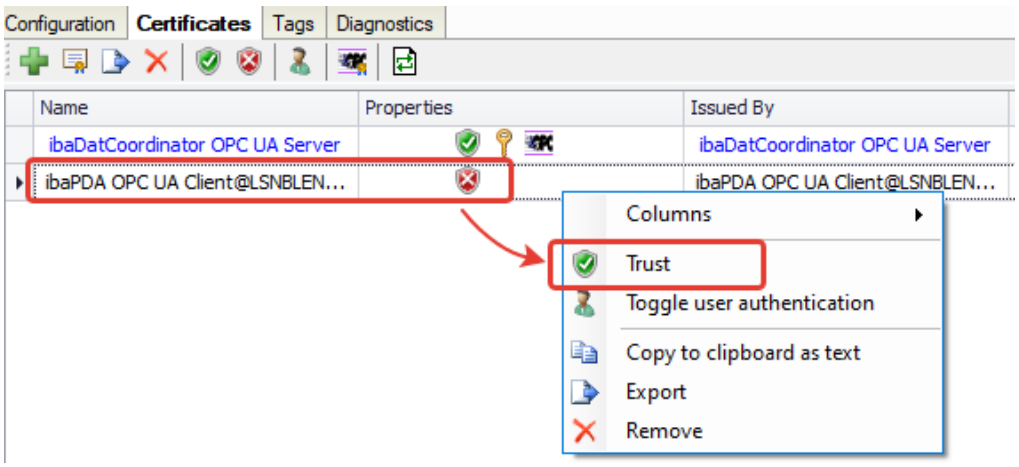


Change connection settings in your client accordingly, and try to connect again:



Connection should **fail** with message similar to “Certificate is not trusted”. This is normal, because we have not set a certificate trust yet.

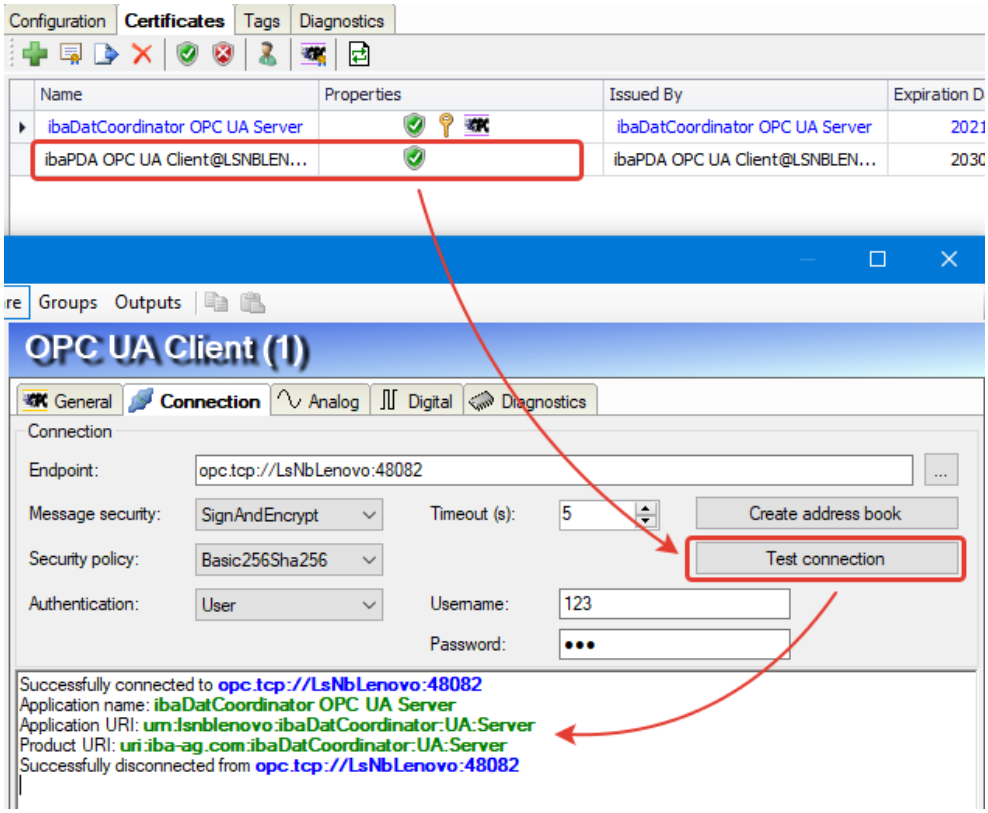
Go to **Certificates** tab. Here you’ll see the **ibaDatCoordinator**’s certificate (is autogenerated, on first launch) and certificates of all clients that attempted to connect **ibaDatCoordinator**.



Find your client’s certificate and, and use the context menu or the toolbar to set that you **trust** it.

Certificates of all new incoming connections are **automatically rejected**; so, you should set trust manually for each new client.

Try connecting again; you should succeed now.



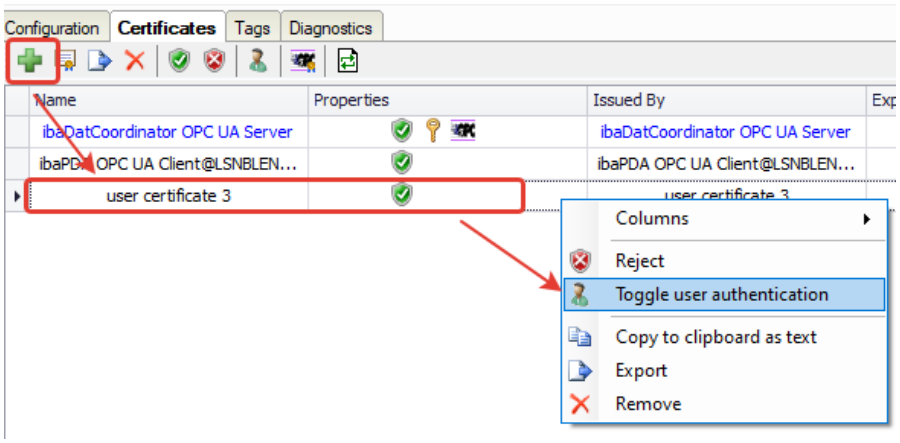
2.1.3 Setting up a user certificate

You can use a **certificate authentication** instead of username/password.
To configure this check corresponding checkbox:

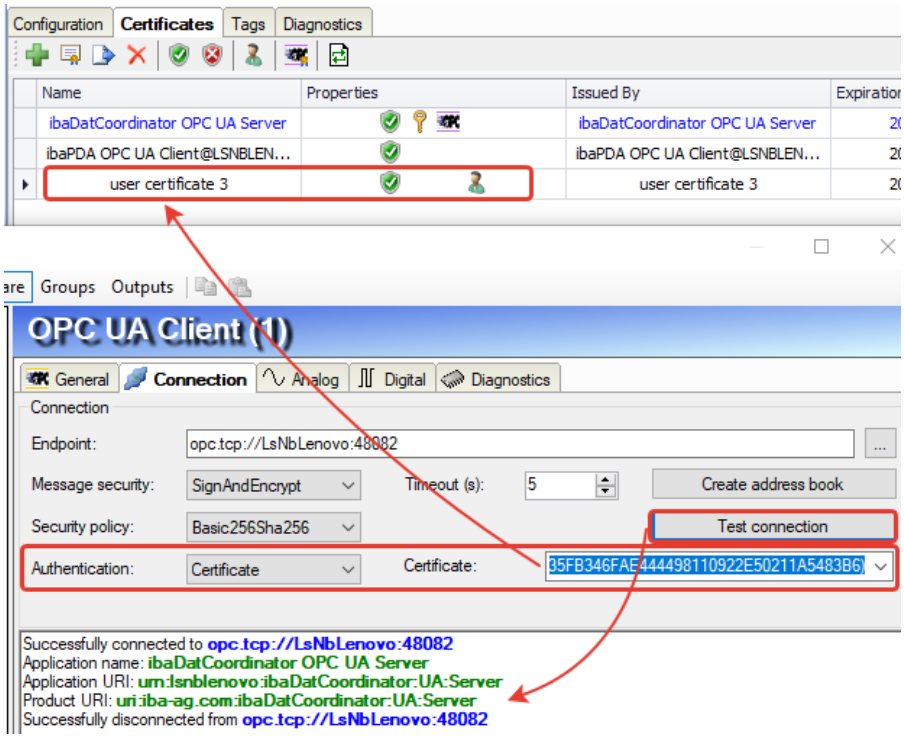
Both (or even all three) ways of authentication can be active simultaneously; though for better security it's recommended that only one of them is active at a time.

Generate a user certificate using tools of your OPC UA client. Try to connect with it. You should fail, because this certificate is not yet trusted by ibaDatCoordinator.

Export the certificate from you client to *.der, *.cer or *.pfx file. Import the certificate in ibaDatCoordinator using the toolbar button and then enable user authentication for it:



Click Apply to apply changes and test your connection again. Now you should succeed:



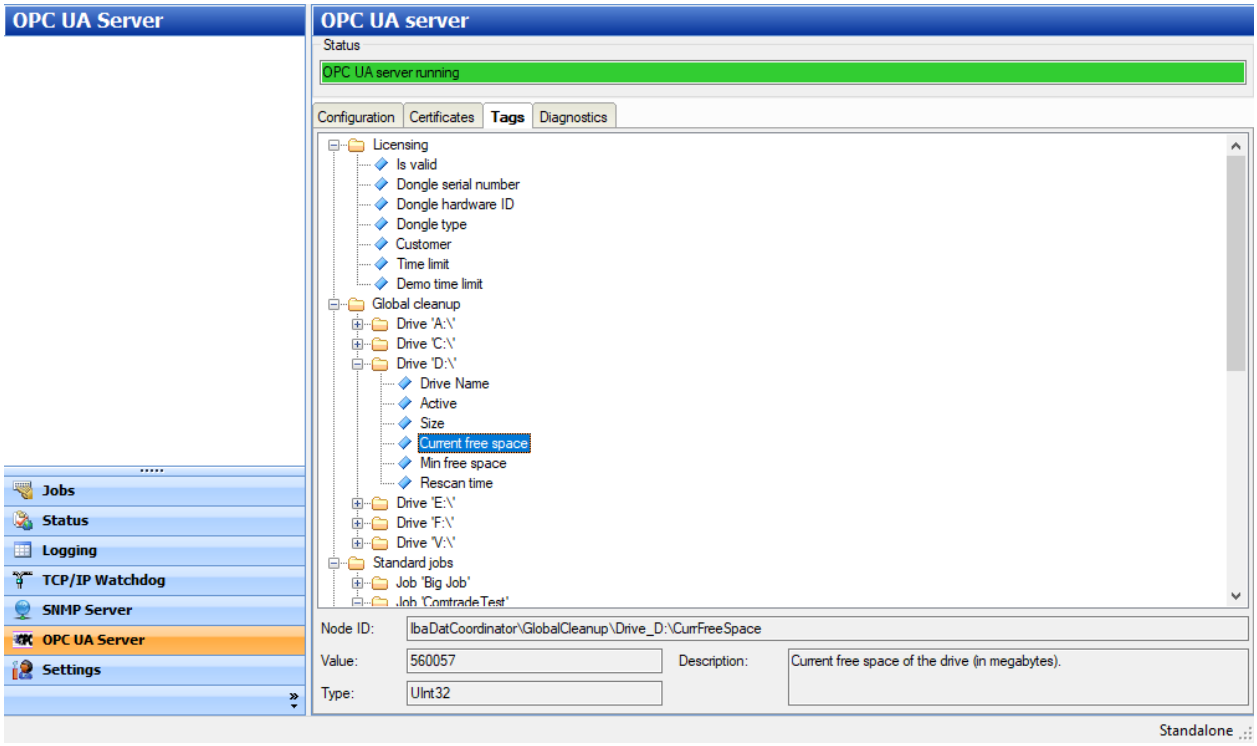
2.2 Available tags

To see what tags are available you can go to the Tags tab. All tags are read-only. They are divided into the following groups:

- License information
- Global cleanup settings for all local drives
- And information about all jobs and tasks, grouped by job type (Standard, Scheduled, etc.)

The most important information here is the **Node ID** of each tag – you should use this ID in your client to read or monitor the tag's value.

Also, you can see each tag's data type, current value and description:



The list of available tags is identical to the one in SNMP Server.