

1 Contents

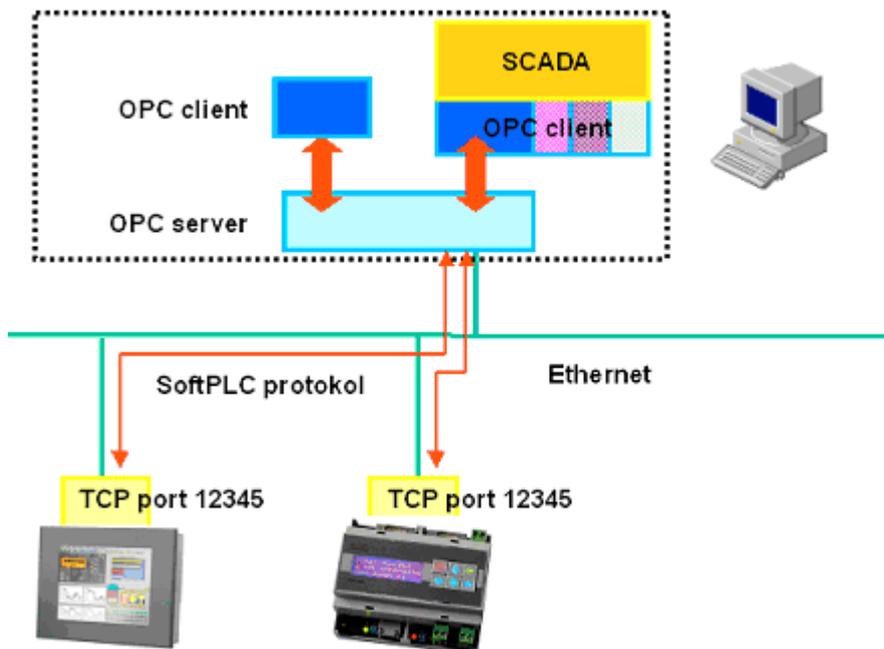
| | | |
|-----|-------------------------------------------------------|----|
| 1 | Contents | 2 |
| 2 | What is a OPC server good for..... | 3 |
| 3 | Installation and setup of the SoftPLC OPC Server..... | 3 |
| 3.1 | Proceedings | 3 |
| 3.2 | Communication test | 8 |
| 3.3 | OPC server and client at different machines | 10 |

2 What is a OPC server good for

OPC is a technology for process data exchange. It was created in cooperation of many world hardware and software producers. The OPC standard is based on Microsoft's OLE/COM/DCOM technology.

All SoftPLC process stations communicate using the Native SoftPLC Protocol. To connect them to any common SCADA software, an OPC server can be used, which represents an interface between the SoftPLC protocol and the unified OPC platform. The SoftPLC OPC Server is a program which connects to one or more SoftPLC runtime(s) at the „bottom“, while providing process variables at the OPC server interface on the „upper“ end.

Topology of the OPC communication



To read and write values, many client programs exist. A client connects to an OPC server and provides a platform to read/write values and diagnose OPC communication. Some of them are freeware, others have limited functionality in the demo mode. Proven programs are e.g. OPC Quick Client (Kepware), and Matrikon OPC Explorer.

An OPC client program is often used for setup and commissioning; it makes possible to check if the OPC server works correctly and provides valid data. často používáme při nastavování a uvádění do provozu – umožní zkontrolovat, zda OPC server pracuje správně a poskytuje platná data. This should be checked prior to the configuration of the client SCADA part.

3 Installation and setup of the SoftPLC OPC Server

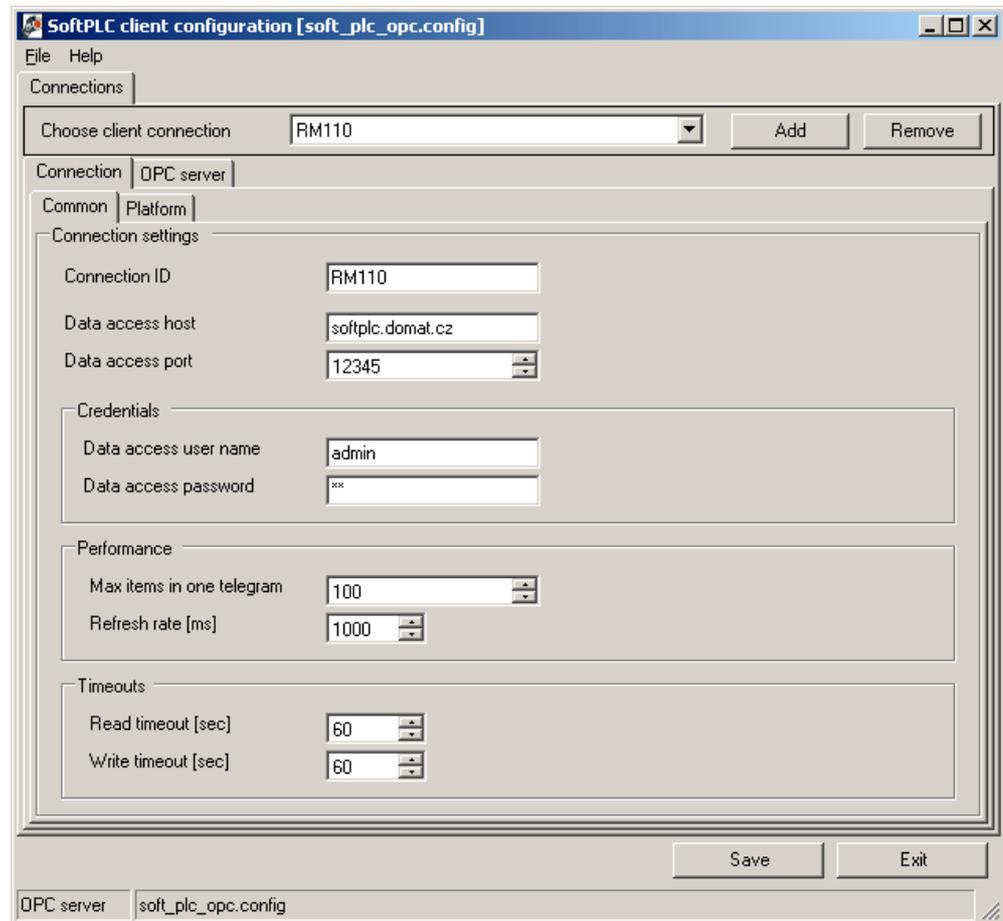
3.1 Proceedings

The machine which the OPC server will run at must have OPC support installed: **OPC Core Components Redistributable**. It can be downloaded from the Microsoft support web (use bold text as search parameter). If those components are not installed the OPC will not work - and this is one of the main reasons for it.

1. Install **OPC Core Components Redistributable**.
2. Install SoftPLC OPC Server. It is a typical „click-through“ installer.

In *Programs – RcWare – SoftPLC* run *OPC Server Configuration*. A configuration program opens with the file named **soft_plc_opc.config**. Do not change this file name.

Configuration program window



3. First, configure the connection(s) to the runtime(s), or process stations. Each runtime connected to the OPC server must have its connection defined in the *Connection* tab and properties of the OPC server set. The connection is defined at two tabs: *Common* and *Platform*, while OPC server properties are set up in the *OPC server* tab.

Select between connections with the drop-down box in the upper part (*Choose client connection*).

4. *Common*: general connection parameters are set here:

Connection ID: name of the connection, usually location or function of the runtime, such as **BA**, **Boilers**, **RMS1** etc. This string is the OPC prefix at the same time, which differs eventual variables of the same name in different runtimes.

RcWare Vision

When importing to RcWare Vision this Connection ID must be entered as Prefix in the import window (Import for OPC / SoftPLC). It is the Connection ID which makes difference between the runtimes.

Data access host: IP address or computer name where the runtime is running. The OPC server must not necessarily be installed at the same machine as the runtime (in case of multiple runtimes it would not even have sense). This address determines where the OPC server aims to establish connection over the SoftPLC Native protocol and connects to the runtime.

Data access port: TCP port on which the runtime listens to incoming connections. Default value is 12345, if not set otherwise in the runtime setup dialogue (or MiniPLC Platform Config dialogue).

Data access user name: The clients (touchscreen, web panel, and OPC server) use credentials to login to the runtime. This is where the user name is entered; default value is *admin*.

Data access password: password for client login, default value is *rw*.

Max items in one telegram: this value usually remains unchanged.

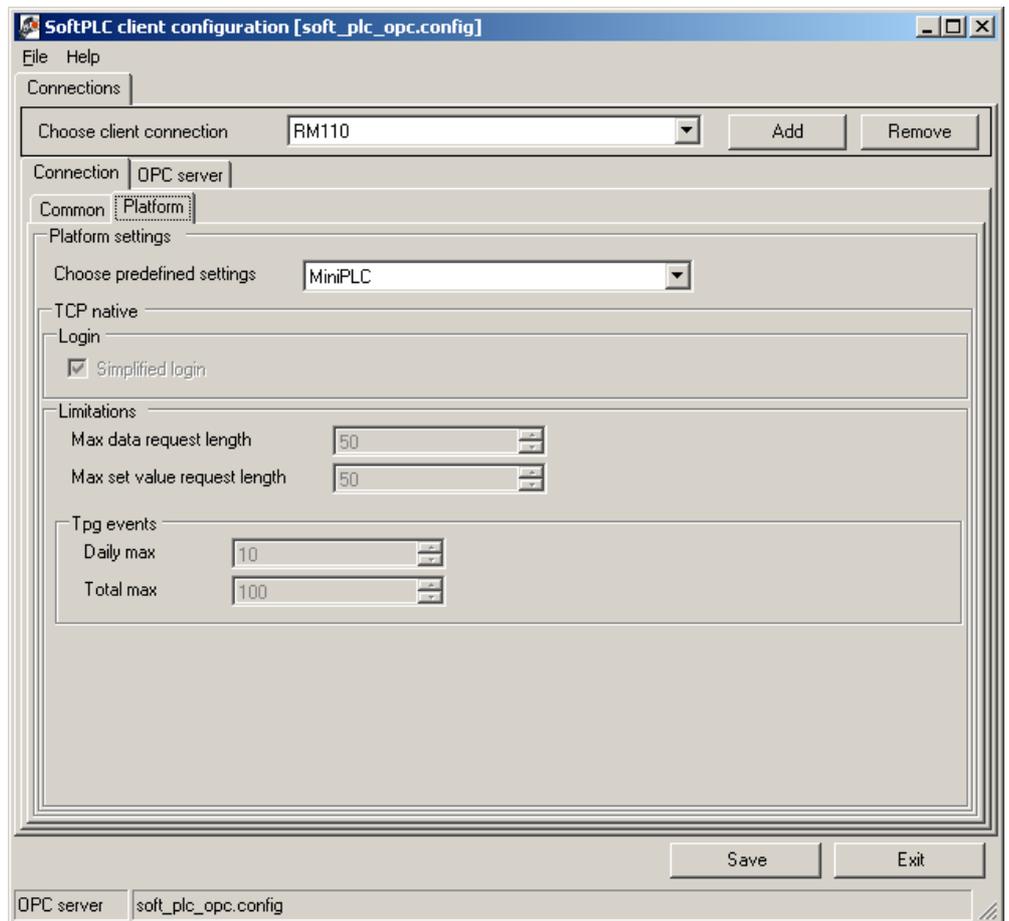
Refresh rate: this value usually remains unchanged.

Read timeout: this value usually remains unchanged.

Write timeout: this value usually remains unchanged.

Last four parameters are to be changed only in case of exceptional network parameters: if the connection is unstable or slow, e.g. GPRS, or if the data stream between the OPC server and runtimes has to be optimized.

5. *Platform:* specify if the device is based on Windows XP Embedded, or MiniPLC, which has a simplified login procedure.

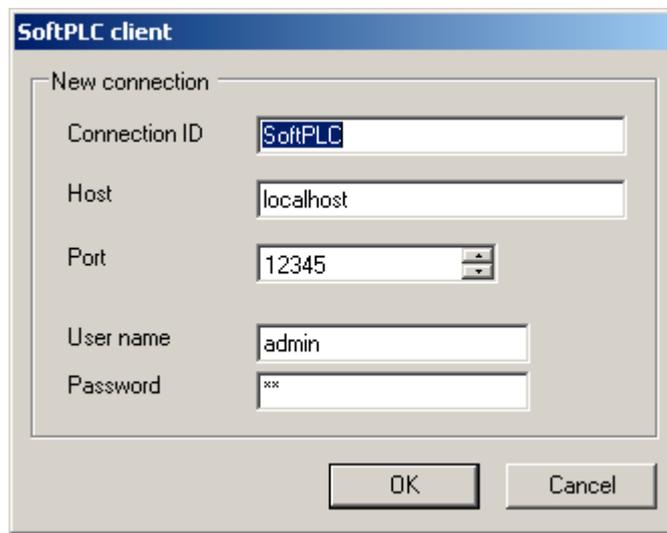


Choose predefined settings: *Full* – touch screen (IPCT.1 etc.) or PC, *MiniPLC* – web controller IPLC200 or IPLC300, *Custom* – not used now.

Click **Save** to save the settings.

Another connection:

New connection with runtime



Click *Add* to define another connection in case there are more runtimes linked to the OPC server. A dialogue opens to set up the basic connection properties which can be defined here. Those are the same parameters as in the *Connection – Common* tab:

Connection ID: name of the connection, usually location or function of the runtime, such as **BA, Boilers, RMS1** etc. This string is the OPC prefix at the same time, which differs eventual variables of the same name in different runtimes.

RcWare Vision

When importing to RcWare Vision this Connection ID must be entered as Prefix in the import window (Import for OPC / SoftPLC). It is the Connection ID which makes difference between the runtimes.

Data access host: IP address or computer name where the runtime is running. The OPC server must not necessarily be installed at the same machine as the runtime (in case of multiple runtimes it would not even have sense). This address determines where the OPC server aims to establish connection over the SoftPLC Native protocol and connects to the runtime.

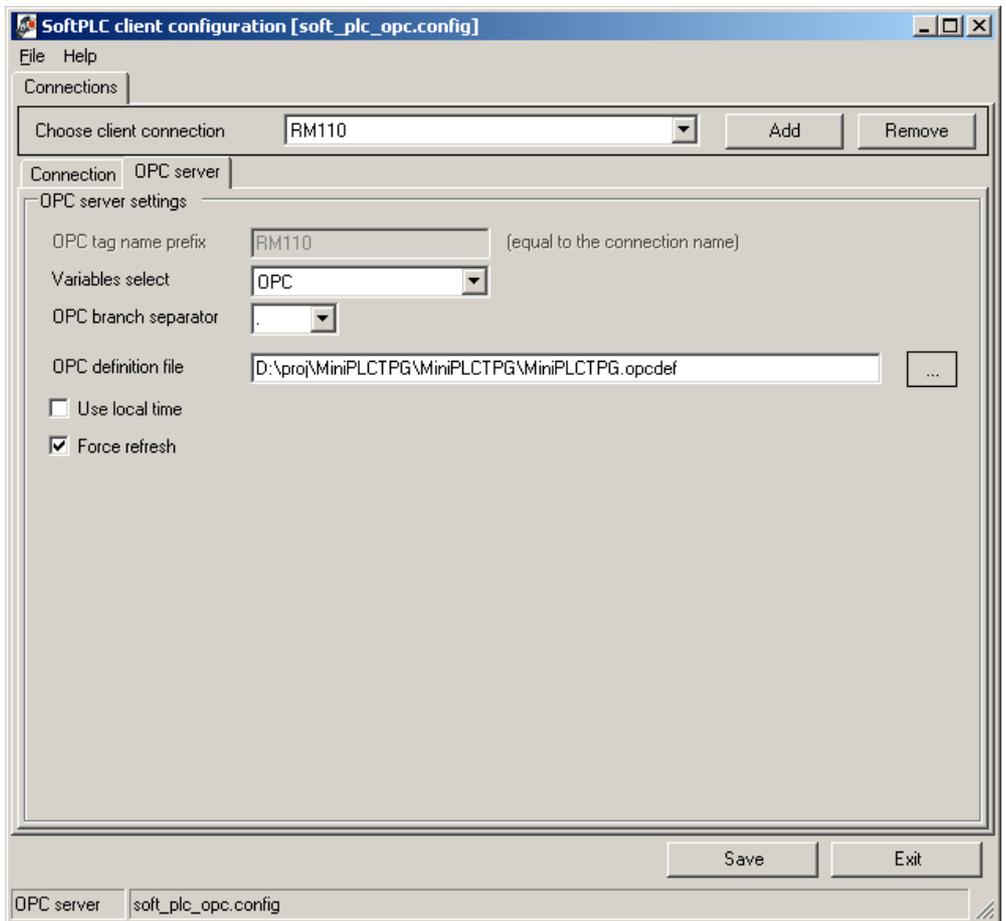
Data access port: TCP port on which the runtime listens to incoming connections. Default value is 12345, if not set otherwise in the runtime setup dialogue (or MiniPLC Platform Config dialogue).

Data access user name: The clients (touchscreen, web panel, and OPC server) use credentials to login to the runtime. This is where the user name is entered; default value is *admin*.

Data access password: password for client login, default value is *rw*.

For each SoftPLC connection the OPC connection properties must be defined, too, in the *OPC server* tab.

OPC server properties definition



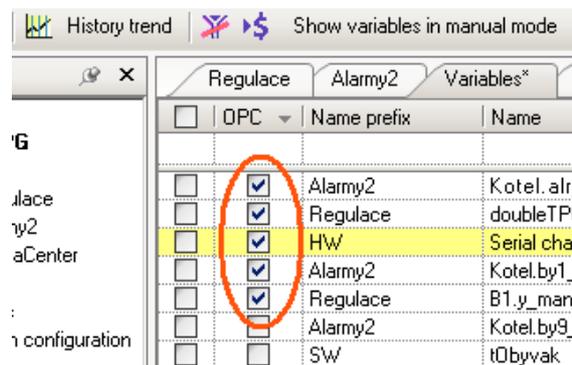
Variables select: set of runtime variables which are visible in the OPC server.

All: all the variables in runtime, means higher network traffic, good for tests and small plants – even variables „invisible“ in the IDE are communicated, e.g. unused inputs of logical function blocks.

OPC: only those variables which are defined in the **.opcdef** file – this file is created in SoftPLC IDE in the *Variables – OPC / Native SoftPLC – Export OPC definition file* menu. This exports only variables checked in the OPC column:

„OPC“ is the recommended option.

Marking of variables to export to .opcdef in IDE



When exporting into **.opcdef** and selecting „OPC“ variables, only those will be communicated in the OPC server.

Used: only variables used in the runtime will be visible, this option is useful in case there is no **.opcdef** file. The amount of variables is comparable to that transmitted with the *All* option.

OPC branch separator: In the OPC structure there are branches separated by a dedicated character – a separator. It is usually a dot („.“), however, some OPCclients require another specific character to work properly (e.g. „_“). Here it is possible to customize this character and thus adapt the OPC server to some non-standard clients –

see documentation to the specific OPC client. Usually, it is not necessary to change this item.

OPC definition file: when *Variables select* is *OPC*, enter path and file name of the **.opcdef** file (see above). This file is usually copied from the SoftPLC project directory, where it is created, to the OPC server working directory or to another defined location (project directory, data directory etc.) of the computer where the OPC server is running.

Use local time: for variable timestamps local PC time is used, instead of runtime machine timestamps.

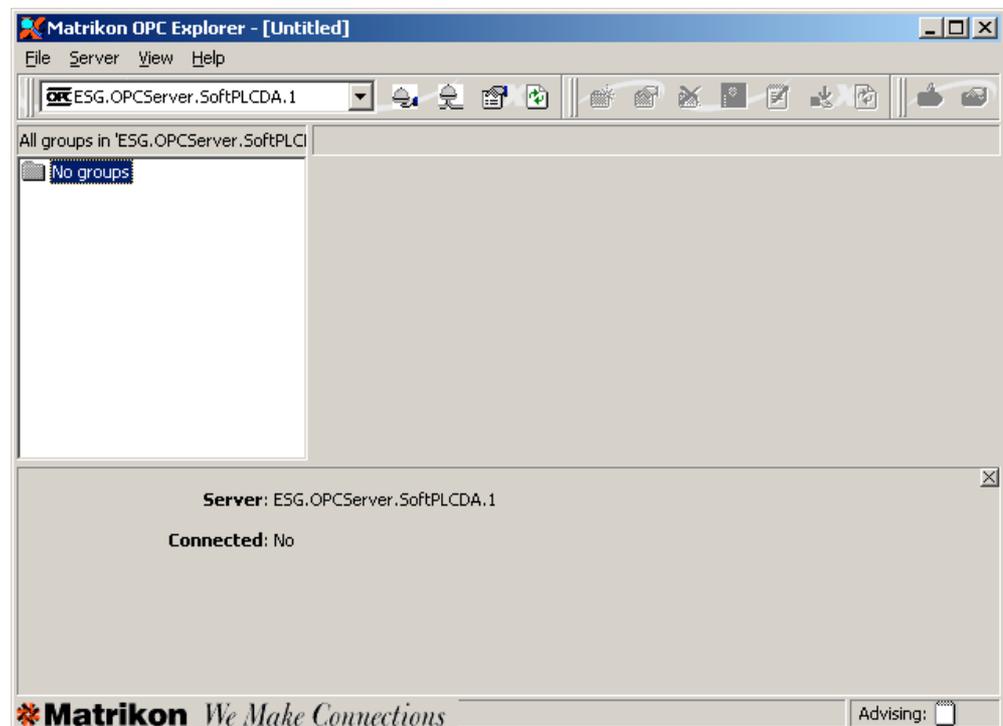
Force refresh: it is recommended to check this option. If Force refresh, even those variables are updated (and marked by actual timestamps) which do not change their value; this increases the bandwidth but keep the timestamp up-to-date, which prevents the variables from being considered as non-communicating due to their old timestamps.

Save the settings clicking the *Save* button.

3.2 Communication test

The OPC server is registered in the operating system at the installation time and its start is invoked by a communication request from the client. It is not necessary to start the server manually. Starting the client and exploring the server will do. Let's have a look at it using a typical client - Matrikon OPC Explorer:

Matrikon OPC Explorer –
main program window



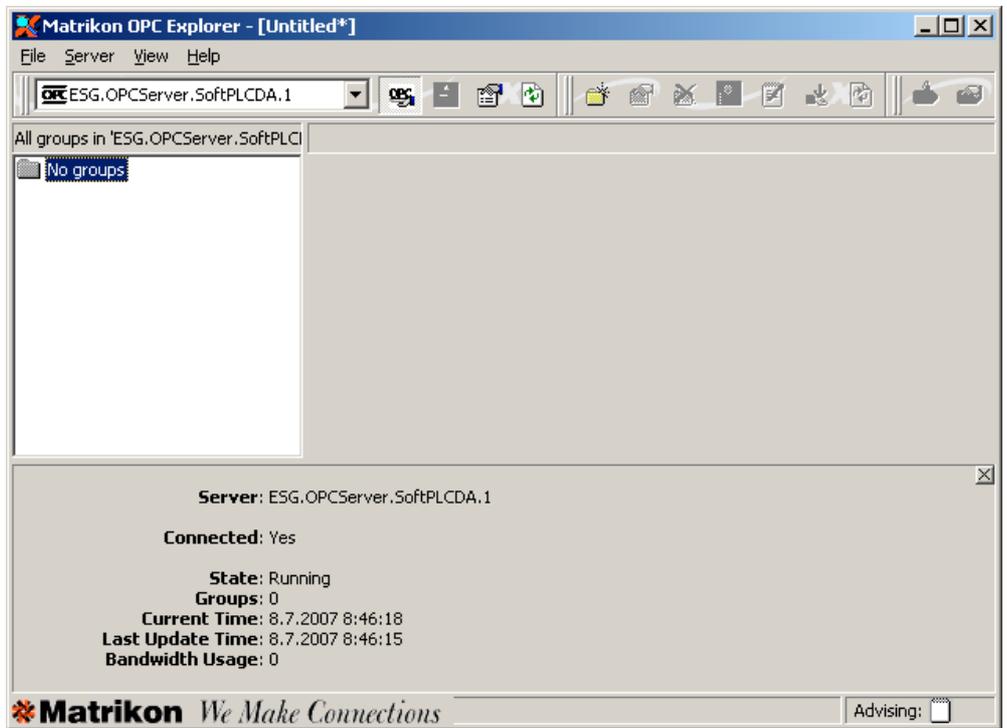
After program start, in the upper left corner a list of available OPC servers shows up. The SoftPLC OPC server can be found as *ESG.OPCServer.SoftPLCDA.1*, choose it in the menu and connect to it by clicking the *Connect to server* button (next to the servers list).

A successful connection is indicated in the lower part of the window (Connected: Yes), the server status can be seen at the same time (State: Running). The OPC server also establishes TCP connections to the runtimes, which is not apparent now.

Possible bug causes:

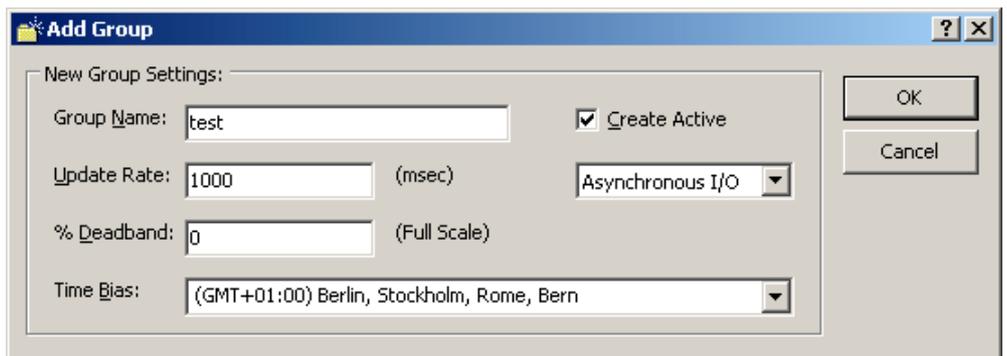
- **OPC Core Components Redistributable** not installed – install it
- SoftPLC OPC Server not installed or unregistered – reinstall the SoftPLC OPC Server

Matrikon OPC Explorer – connected



Now, a group of variables must be created and some variables to observe must be inserted.

Click on *Create new group* (yellow folder) and call the group e.g. test.



Click OK and then press *Insert* to open the OPC Tag studio – a dialog for inserting variables. In the *Available items...* window, a variable tree should appear, where one or more variables (tags) can be selected and inserted into the tag list in the context menu *Add to tag list / Add all items to tag list*.

If the tree does not open, check

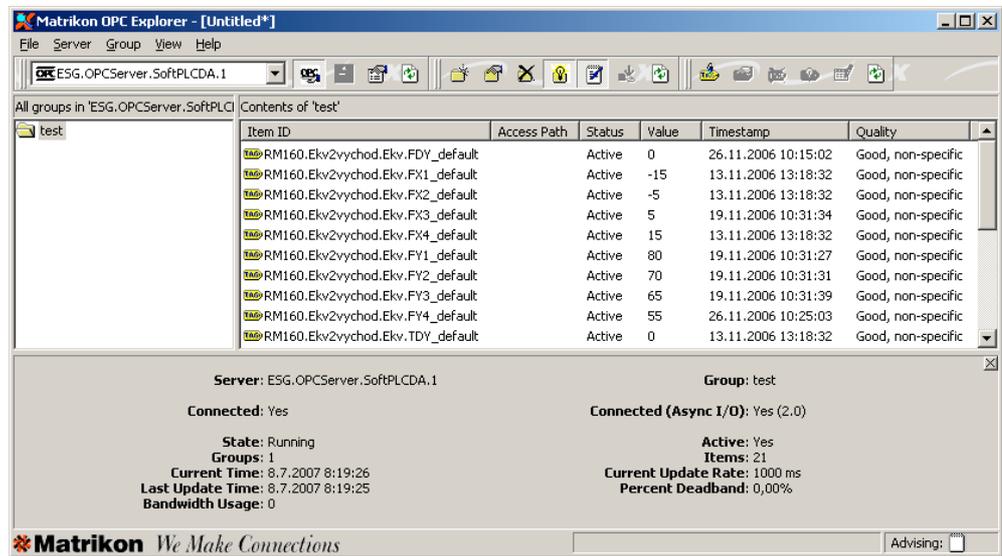
- runtime connection settings, or try to connect to the runtime with another SoftPLC client, such as the IDE, and check name, password, and TCP port settings.
- if the **OPC Core Components Redistributable** are installed.

After all requested variables have been inserted (for testing purposes, it is advisable to select variables which change their values periodically, such as temperature inputs, current time, etc.) close Tag studio and return back to Explorer. The variables should display their current values and the Quality attribute should be Good.

If it is not the case, check if:

- the runtime machines are operational and available (ping) from the machine where the OPC server is running
- firewalls are off or configured properly
- correct names, passwords, ports and IP addresses are set in the runtime definition.

A connected client



The screenshot shows the Matrikon OPC Explorer interface. The title bar reads "Matrikon OPC Explorer - [Untitled*]". The menu bar includes "File", "Server", "Group", "View", and "Help". The address bar shows "ESG.OPCServer.SoftPLCDA.1". The main window is divided into two panes. The left pane, titled "All groups in 'ESG.OPCServer.SoftPLCDA.1'", shows a tree view with a folder named "test". The right pane, titled "Contents of 'test'", displays a table with the following columns: Item ID, Access Path, Status, Value, Timestamp, and Quality. The table contains 10 rows of data, all with a status of "Active" and a quality of "Good, non-specific". The bottom section of the window provides connection details for the server "ESG.OPCServer.SoftPLCDA.1" and the group "test".

| Item ID | Access Path | Status | Value | Timestamp | Quality |
|----------------------------------|-------------|--------|-------|---------------------|--------------------|
| RM160.Ekv2vychod.Ekv.FDY_default | | Active | 0 | 26.11.2006 10:15:02 | Good, non-specific |
| RM160.Ekv2vychod.Ekv.FX1_default | | Active | -15 | 13.11.2006 13:18:32 | Good, non-specific |
| RM160.Ekv2vychod.Ekv.FX2_default | | Active | -5 | 13.11.2006 13:18:32 | Good, non-specific |
| RM160.Ekv2vychod.Ekv.FX3_default | | Active | 5 | 19.11.2006 10:31:34 | Good, non-specific |
| RM160.Ekv2vychod.Ekv.FX4_default | | Active | 15 | 13.11.2006 13:18:32 | Good, non-specific |
| RM160.Ekv2vychod.Ekv.FY1_default | | Active | 80 | 19.11.2006 10:31:27 | Good, non-specific |
| RM160.Ekv2vychod.Ekv.FY2_default | | Active | 70 | 19.11.2006 10:31:31 | Good, non-specific |
| RM160.Ekv2vychod.Ekv.FY3_default | | Active | 65 | 19.11.2006 10:31:39 | Good, non-specific |
| RM160.Ekv2vychod.Ekv.FY4_default | | Active | 55 | 26.11.2006 10:25:03 | Good, non-specific |
| RM160.Ekv2vychod.Ekv.TDY_default | | Active | 0 | 13.11.2006 13:18:32 | Good, non-specific |

Server: ESG.OPCServer.SoftPLCDA.1
Group: test
Connected: Yes
State: Running
Groups: 1
Current Time: 8.7.2007 8:19:26
Last Update Time: 8.7.2007 8:19:25
Bandwidth Usage: 0
Connected (Async I/O): Yes (2.0)
Active: Yes
Items: 21
Current Update Rate: 1000 ms
Percent Deadband: 0,00%

The Timestamp column displays time of the last variable change, which means that for e.g. parameters, which do not change frequently, old timestamps can be found. This is in case of the *Force Refresh* attribute in OPC server properties unchecked; what is important are the Status (Active) and Quality (Good) columns.

Note that the Connection ID string has become part of the OPC tag name.

Now the OPC server is properly installed and configured and ready to accept client connections (e.g. SCADA).

3.3 OPC server and client at different machines

So far, we have been supposing that both the OPC client and the OPC server (not the runtime) are running on one PC – see the communication topology above. Should this not be the case, things get a bit complicated: the DCOM (Distributed Component Object Model) properties must be set correctly for both machines, so that the client can request services running on another computer in the network.

The easiest way is to use the same user and password for both computers, which is not always possible.

The DCOM properties setting is not the subject of this manual. Read <http://www.opcfoundation.org/forum/> , especially <http://www.opcfoundation.org/forum/viewforum.php?f=10> .