

# **OPC ARRAY EXTRACTOR**

for providing OPC arrays data  
to SuiteLink/DDE Clients

**User Manual**  
**Draft Ver 1.x Rev 1.1**  
**DR 540 10**

KLINKMANN AUTOMATION  
P.O. Box 38  
FIN-00371 Helsinki Finland  
tel. int. + 358 9 5404940  
fax int. + 358 9 5413541  
[www.klinkmann.com](http://www.klinkmann.com)

# Table Of Contents

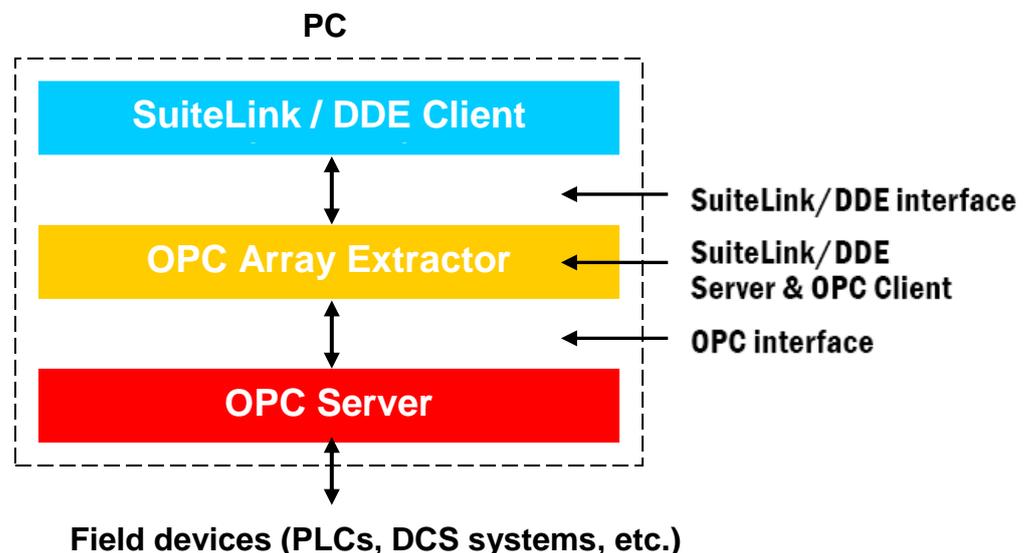
Overview .....	1
Communication Protocols .....	2
Accessing Remote Items via the OPCARREXTR .....	4
Installing the OPC ARRAY EXTRACTOR .....	4
Installing the product .....	4
Licensing by using HASP HL key .....	7
Software license key.....	7
Transferring the software license to other computer .....	8
Installing the I/O Server Infrastructure .....	10
Configuring the OPC Array Extractor .....	11
Server Settings Command.....	11
OPC Server Settings Command.....	12
Saving of the OPC Array Extractor Configuration File .....	14
Configuration File Location .....	14
Topic Definition Command .....	14
Item Names.....	16
"Array in string" type items.....	17
Monitoring and Controlling Communication with OPC Servers.....	19
Using OPC Array Extractor with Suite Link and DDE Clients .....	20
Using the OPC Array Extractor with InTouch .....	20
Defining the Access Names.....	20
Defining the Tag names.....	22
Monitoring the Status of Communication with InTouch.....	24
Notes on Using Microsoft Excel .....	24
Reading Values into Excel Spreadsheets.....	25
Troubleshooting .....	25
WIN.INI entries .....	25
Troubleshooting menu .....	26

# OPC ARRAY EXTRACTOR

## Overview

The **OPC Array Extractor** (hereafter referred also as “OPCARREXTR” or “OPCArrExtr” or “Server”) is a Microsoft Windows 32-bit application program that acts as a communication gateway and provides other Windows application programs with data from arrays and single items provided by OPC Servers. The main purpose of OPC Array Extractor application is to provide values of single elements of OPC arrays and single OPC items for **DDE**, **FastDDE** or **SuiteLink** clients. The OPC Array Extractor application is primarily intended for use with Wonderware FactorySuite 2000 and FactorySuite A<sup>2</sup>, but it may be used with any Microsoft Windows program capable of acting as a DDE, FastDDE or SuiteLink client.

In general, the system layout is like following:



The OPC Array Extractor application consists of two main parts:

- **OPC Client** part, for connection to appropriate OPC Server and obtaining arrays and single items data via OPC interface;
- **SuiteLink/DDE Server** part, providing data for SuiteLink/DDE Clients, e.g. Wonderware InTouch.

The OPC Array Extractor OPC Client part connects to appropriate OPC Server when some DDE/SuiteLink client (e.g. Wonderware InTouch) connects to some of configured OPC Array Extractor topics, and disconnects from OPC Server when all OPC Array Extractor DDE/Suitelink clients are disconnected. The reading of data from OPC Server is activated after activating any data item at DDE/Suitelink side and deactivated when all data items at DDE/Suitelink side are deactivated and all pending OPC transactions are completed.

When OPC Array Extractor OPC Client part receives new I/O item activation request from SuiteLink/DDE Server part, then at first this I/O item is validated - by checking if corresponding OPC array name or single item name is a valid OPC Server item name and OPC Server can receive data from field device. If item name is valid then OPC Server will start (or continue) the data reading (polling) from field device and OPC Array Extractor will provide DDE/SuiteLink client with values extracted from data received from OPC Server.

For array elements the correspondence between SuiteLink/DDE “single item” name (activated for OPC Array Extractor SuiteLink/DDE Server part) and OPC array name (used in OPC Server) is very simple - the “single item” name is equal with OPC array name plus one character prefix is added to specify the data type (discrete, integer, real or string) plus suffix with array element address (index) is added at the end. For example, if OPC integer array name is **Proc1.INT20** then “single item” name for this OPC array 5<sup>th</sup> element can be **iProc1.INT20[4]** (suffix syntax and whether index starts from 0 or 1 is configurable) .

For single items the SuiteLink/DDE item name is equal with OPC item name plus one character prefix is added to specify the data type (discrete, integer, real or string). For example, if OPC discrete item name is **M1\_8083.RMGAUGE.IECStatus** then SuiteLink/DDE item name is **dm1\_8083.RMGAUGE.IECStatus**.

OPC Array Extractor supports also special “**array in string**” type items, where array (or part of array) data is put into single string item thus allowing significantly reduce the amount of items needed to receive whole array data. In this case special rules are used for item naming and to store the array data inside the string.

## Communication Protocols

**Dynamic Data Exchange** (DDE) is a communication protocol developed by Microsoft to allow applications in the Windows environment to send/receive data and instructions to/from each other. It implements a client-server relationship between two concurrently running applications. The server application provides the data and accepts requests from any other application interested in its data. Requesting applications are called clients. Some applications such as Wonderware InTouch and Microsoft Excel can simultaneously be both a client and a server.

**FastDDE** provides a means of packing many proprietary Wonderware DDE messages into a single Microsoft DDE message. This packing improves efficiency and performance by reducing the total number of DDE transactions required between a client and a server. Although Wonderware's FastDDE has extended the usefulness of DDE for the industry, this extension is being pushed to its performance constraints in distributed environments. The OPC Array Extractor supports the FastDDE Version 3 - an extension to Wonderware's proprietary FastDDE Version 2. This extension supports the transfer of Value Time Quality (VTQ) information. The original DDE and FastDDE Version 2 formats are still supported, providing full backward compatibility with older DDE clients.

**NetDDE** extends the standard Windows DDE functionality to include communication over local area networks and through serial ports. Network extensions are available to allow DDE links between applications running on different computers connected via networks or

modems. For example, NetDDE supports DDE between applications running on IBM compatible computers connected via LAN or modem and DDE-aware applications running on non-PC based platforms under operating environments such as VMS and UNIX.

**SuiteLink** uses a TCP/IP based protocol and is designed by Wonderware specifically to meet industrial needs such as data integrity, high-throughput, and easier diagnostics. This protocol standard is supported on Windows NT and Windows NT-technology-based operating systems including Windows 2000 Professional, Windows 2000 Server, Windows 2000 Advanced Server, Windows 2003 Server, and Windows XP Pro. SuiteLink is not a replacement for DDE, FastDDE, or NetDDE. The protocol used between a client and a server depends on your network connections and configurations. SuiteLink was designed to be the industrial data network distribution standard and provides the following features:

- Value Time Quality (VTQ) places a time stamp and quality indicator on all data values delivered to VTQ-aware clients.
- Extensive diagnostics of the data throughput, server loading, computer resource consumption, and network transport are made accessible through the operating system performance monitor. This feature is critical for the scheme and maintenance of distributed industrial networks.
- Consistent high data volumes can be maintained between applications regardless if the applications are on a single node or distributed over a large node count.
- The network transport protocol is TCP/IP using Microsoft's standard WinSock interface.

**OPC** (OLE for Process Control) is an open interface standard to provide data from a data source and communicate the data to any client application in a common standard way. The OPC is based on Microsoft OLE, COM and DCOM technologies and enables simple and standardized data interchange between the industrial or office sector and the production sector. From general point of view many aspects of OPC are similar to DDE, but main difference is in the implementation by using Microsoft's COM (Component Object Model) technology. It enables fast exchange with process automation data and OPC open interface allows access to data from OPC Server in same standard way from OPC client applications supplied by different developers.

For more information on the basics of OPC, please refer to the **OPC Specification**. The OPC Data Access Custom Interface Specification is maintained by **OPC Foundation**, the current specification is 2.05a dated June 2002 (3.00 dated March 2003).

**Wonderware DAServers** are Wonderware's next-generation I/O servers built on Archestra™ technology. They are designed to provide simultaneous connectivity between plant floor devices and modern DDE, SuiteLink and/or OPC based client applications. DAServers support the OPC Data Access 2.05 specification and offer additional features beyond the standard, including powerful diagnostics and remote configuration capabilities.

The Suite Link, FastDDE (Version 3) and DDE support for OPC Array Extractor is implemented by **Wonderware I/O Server Toolkit** ver. 7,2,1,6.

The OPC support for OPC Array Extractor OPC Client part conforms to OPC Data Access Custom Interface Specification 2.05a.

## Accessing Remote Items via the OPCARREXTR

The communication protocol addresses an element of data in a conversation that uses a three-part naming convention that includes the **application name**, **topic name** and **item name**. The following briefly describes each portion of this naming convention:

### **application name**

The name of the Windows program (server) that will be accessing the data element. In the case of data coming from appropriate OPC Server via this Server (OPC Array Extractor), the application portion of the address is **OPCARREXTR**.

### **topic name**

Meaningful name of communication entity, that is referred also as "Logical device". Commonly the name considers one OPC Group used to access the corresponding OPC Server items for this topic. For example, **S7PLC**.

### **item name**

A specific data element within the specified topic. When using the OPC Array Extractor, items are elements of corresponding OPC arrays or single OPC items. The term "point" is used interchangeably with the term "item" in this User Manual. For more information on item names, see the **Item Names** section later in this manual.

## Installing the OPC ARRAY EXTRACTOR

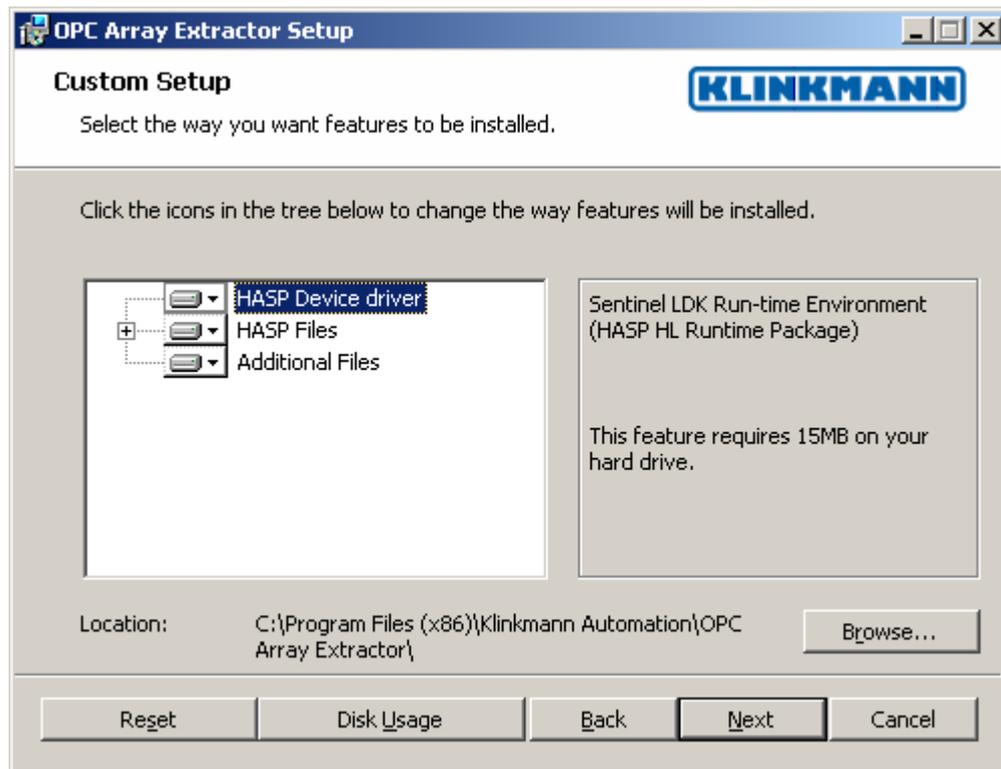
### Installing the product

The OPC Array Extractor installation package is supplied as a Microsoft Installer file DR54010\_xxx.msi, where xxx is the current (latest) version of the OPC Array Extractor.

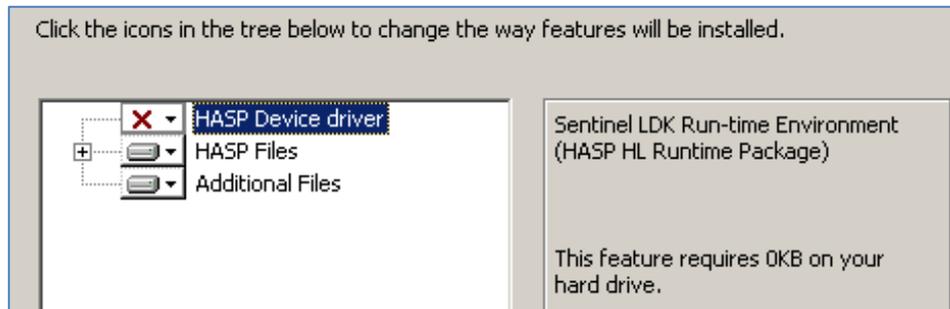
To install the OPC Array Extractor, run the DR54010\_10\_xxx.msi and proceed as directed by the OPC Array Extractor Setup Wizard. The installation is simple and straightforward, only it is important to select the correct protection (**HASP key** or **software license**) in "Custom Setup" dialog.

The HASP key or software license key is needed for full time running of OPC Array Extractor. The **HASP key** is an USB key (dongle) to be installed into PC USB port and needs the SafeNet Sentinel LDK Run-time Environment (HASP HL Runtime Package) to be installed and running – see details in "Licensing by using HASP HL key" section below. The **software license key** is a 16-character alphanumeric "computer-dependent" string, provided after purchasing the OPC Array Extractor (for more information, see "Software license key" section below. Without HASP key installed or software license key entered, the OPC Array Extractor will run one hour in demo mode. After purchasing the OPC Array Extractor, the appropriate HASP key or software license key is provided and no re-installation of OPC Array Extractor is needed.

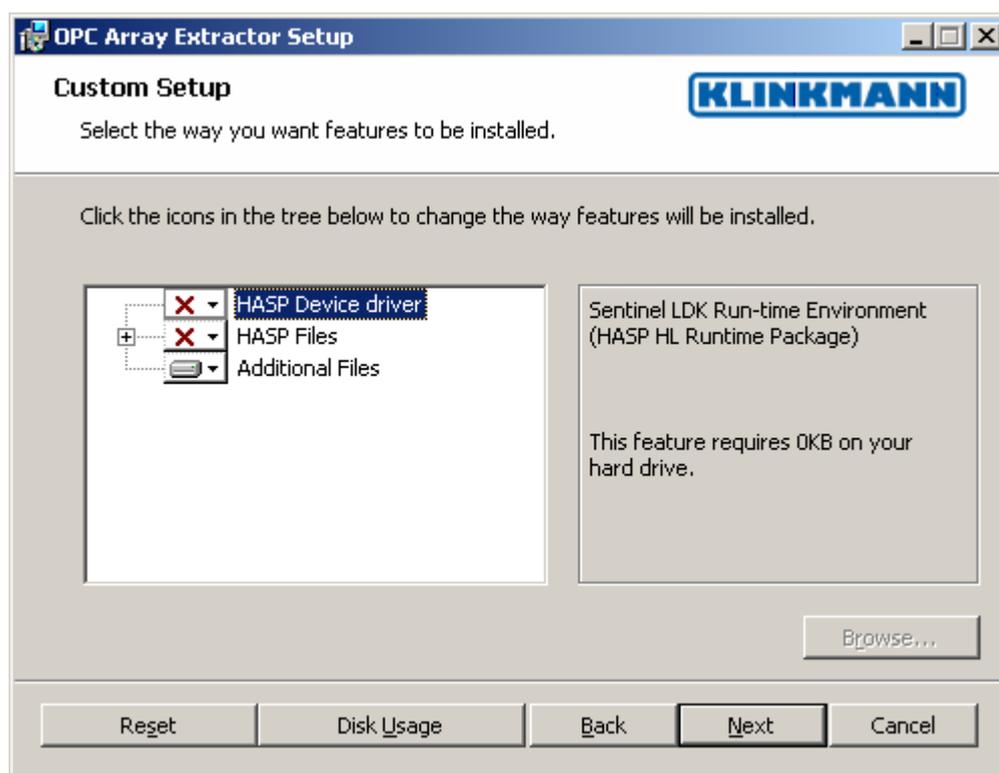
In case "HASP Device driver" and "HASP files" **are selected** then HASP USB key will be supported and both HASP-key and software license will be available (files needed for HASP USB key will be installed):



**Note:** In case the SafeNet Sentinel LDK Run-time Environment (HASP HL Runtime Package) is already installed on your computer (separately or by some other software) then it can be disabled:



In case “HASP Device driver” and “HASP files” **are not not selected** then HASP USB key will not be supported and only the **software license** will be available (files needed for HASP USB key will not be installed):



When installation is finished, the subdirectory specified as a folder where to install the OPC Array Extractor files will contain the following files:

- |                                   |   |
|-----------------------------------|---|
| <b>OPCArrExtr.exe</b>             | The OPC Array Extractor Program. This is a Microsoft Windows 32-bit application program.  |
| <b>OPCArrExtr.chm</b>             | The OPC Array Extractor Help file.  |
| <b>OPCArrExtr.cfg</b>             | An example configuration file.  |
| <b>WAS.ZIP</b>                    | Necessary files and description for using the OPCArrExtr "Array in string" type items in Wonderware Application Server environment. This .zip file is copied to in \additional files sub-folder only in case it is selected at "Custom Setup" dialog. |
| <b>hasp_windows_44<br/>42.dll</b> | Dynamic Link Library installed only if "HASP Files" is selected during the installation in "Custom Setup" dialog.   |
| <b>haspdinst.exe</b>              | Sentinel LDK Run-time Environment Installer (HASP HL Runtime Package), copied to OPC Array Extractor folder only if "HASP Device driver" is selected during the installation in "Custom Setup" dialog.  |
| <b>license.rtf</b>                | Klinkmann Automation software license file.   |

In case the "HASP Device driver" is selected during the installation in "Custom Setup" dialog, the Sentinel LDK Run-time Environment (HASP HL Runtime Package) is installed during the OPC Array Extractor installation (and will be uninstalled during the OPC Array Extractor uninstallation). The presence of Sentinel LDK Run-time Environment can be

checked after the OPC Array Extractor installation by looking-up in Control Panel / Administrative Tools Services – the Service “Sentinel Local License Manager” must be started.

**Notes:**

1. The OPC Array Extractor is developed with Wonderware I/O Server Toolkit (ver 7,2,1,6) and needs the **Wonderware FS2000 Common Components** to be installed on computer where the OMRONETH Server is running. The Wonderware FS2000 Common Components are installed automatically when any of Wonderware product (e.g. InTouch or some Wonderware I/O server) is installed.
2. If OPC Array Extractor will run on PC where Wonderware FS2000 Common Components are not installed then a special **I/O Server Infrastructure installation package** can be obtained from Klinkmann Automation (see **Installing the I/O Server Infrastructure** section below). This I/O Server Infrastructure installation package contains the minimum set of software needed to run the OPC Array Extractor and these infrastructure files must be install prior to executing the OPC Array Extractor. The **I/O Server Infrastructure** does not support using OPC Array Extractor as a Windows Service and remote access to OPC Array Extractor (DDE/SuiteLink Client is not located on same computer as OPC Array Extractor).

To **uninstall** the OPC Array Extractor, start Control Panel, select “Uninstall a program” (“Add/Remove Programs” on XP/2003) and select the “OPC Array Extractor” from the list of available software products. Click on “Uninstall” (“Add/Remove...” on XP/2003) and proceed as directed by the Uninstall Wizard.

## Licensing by using HASP HL key

The following should be done to enable the licensing by HASP HL key:

- The “HASP Device driver” is selected during the OPC Array Extractor installation in “Custom Setup” dialog – that causes correspondingly haspdinst.exe and hasp\_windows\_4442.dll files are copied to OPC Array Extractor folder and Sentinel LDK Run-time Environment (HASP HL Runtime Package) is installed and started, enabling the OPC Array Extractor can detect the HASP HL USB dongle;
- insert the received HASP HL key into USB port, and **wait** until “Installing device driver software” message disappears and “Device driver software installed successfully” message appears;
- start OPC Array Extractor and check - if “Software key or HASP HL key not found!” message does not appear then it means everything is done correctly and OPC Array Extractor runs in full mode with licensing by HASP HL key enabled.

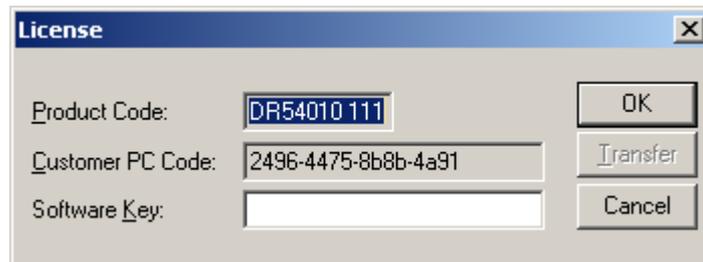
## Software license key

OPC Array Extractor supports the “computer dependent” **software licensing**. The following steps are required to enable it:

- 1) Start OPC Array Extractor, click on "Help" menu item (also short-cut Alt+H can be used) and pop-up menu with "Help" menu commands will appear:



Select "License" and "License" dialog will appear:



2) Here the "Customer PC Code" is "computer-dependent" string generated by OPC Array Extractor and it is unique for this computer. Write it down or Copy/Paste to e-mail when ordering the OPC Array Extractor.

3) After purchasing the OPC Array Extractor, you will get the software license key - 16-character alphanumeric string. Open the "License" dialog again and Copy/Paste it to "Software Key" field:



4) Click OK and restart OPC Array Extractor. OPC Array Extractor software license now is enabled.

Note – the "Software Key" string is saved to MS Windows registry to enable it is automatically detected at OPC Array Extractor next start-up.

### **Transferring the software license to other computer**

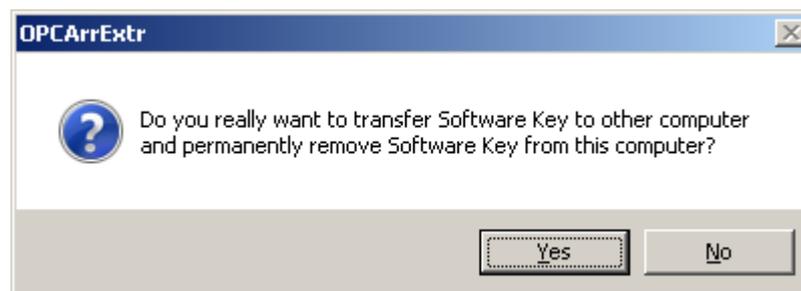
The transfer of Software License Key might be needed in very rare situations when it is necessary to move Klinkmann software to other computer (or operation system change is planned for same computer). Such transfer PERMANENTLY removes the Software License Key, so be very careful when deciding to use this option.

The following steps are required to transfer the Software License Key:

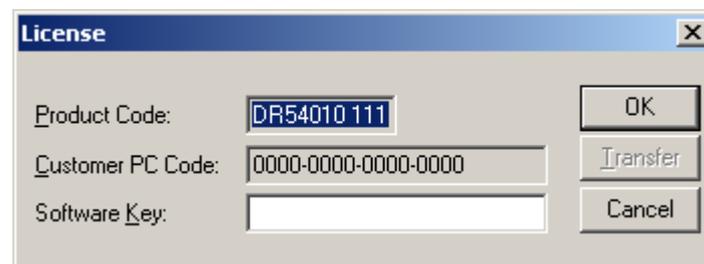
- 1) Select Help/License from main menu and click the “Transfer” button on “License” dialog:



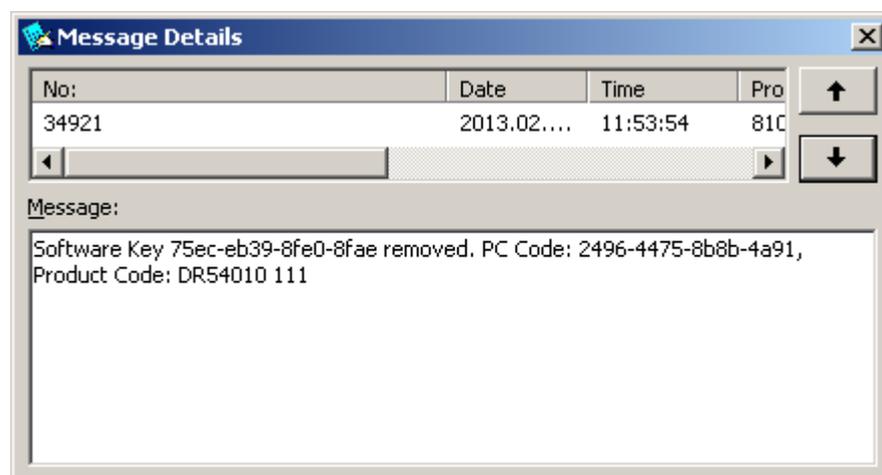
- 2) Confirm the transfer of Software License Key by clicking on Yes button:



The “License” dialog now will contain the empty “Customer PC Code” and “Software Key” fields:



- 3) Take the screenshot from ArchestrA SMC Logger window containing the “Software License Key removal message”, like below:



or take the string with “Software License Key removal message” directly from Archestra SMC Logger, like following:

**Software Key 75ec-eb39-8fe0-8fae removed. PC Code: 2496-4475-8b8b-4a91, Product Code: DR54010 111**

- 4) Provide the obtained “Software License Key removal message” screenshot or string together with new “Customer PC Code” when applying for new Software License Key without purchasing the new license (in situations when it is necessary to move Klinkmann software to other computer or operation system change is planned).

### **Note!**

Without providing the “Software License Key removal message” screenshot or string, the new Software License Key will not be assigned.

## **Installing the I/O Server Infrastructure**

The I/O Server Infrastructure installation package is supplied as a self-extracting archive (IOServerInfrastructure.exe) and can be downloaded from Klinkmann’s web site.

To **install** the I/O Server Infrastructure from the self-extracting archive, run the IOServerInfrastructure.exe and proceed as directed by the I/O Server Infrastructure Setup program.

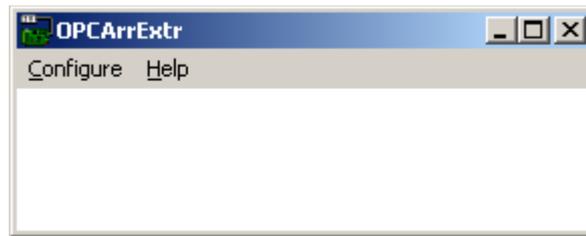
To **uninstall** the I/O Server Infrastructure, start Control Panel, select “Add/Remove Programs” and select the “IO Server Infrastructure” from the list of available software products. Click on “Add/Remove...” and proceed as directed by the UnInstallShield program.

**Note:** *The I/O Server Infrastructure installation will be rejected if Wonderware FS2000 Common Components are already installed on same computer.*

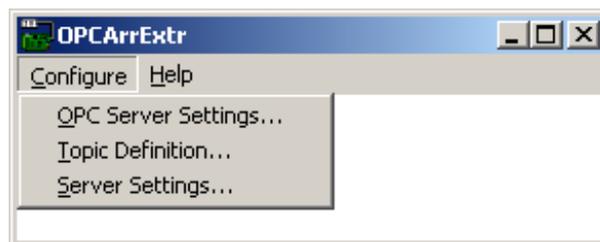
## Configuring the OPC Array Extractor

After the OPC Array Extractor is initially installed, a little of configuration is required. Configuring the OPC Array Extractor automatically creates **OPCARREXTR.CFG** file that holds all of the OPC Server(s) settings and topic(s) definitions entered. This file will be placed automatically in the same directory in which the OPC Array Extractor is located unless the path where the configuration file will be placed is specified via the */Configure/Server Settings...* command.

To perform the required configurations, start up the OPC Array Extractor program. If the OPC Array Extractor starts up as an icon, double-click on the icon to open the server's window. The following will appear:



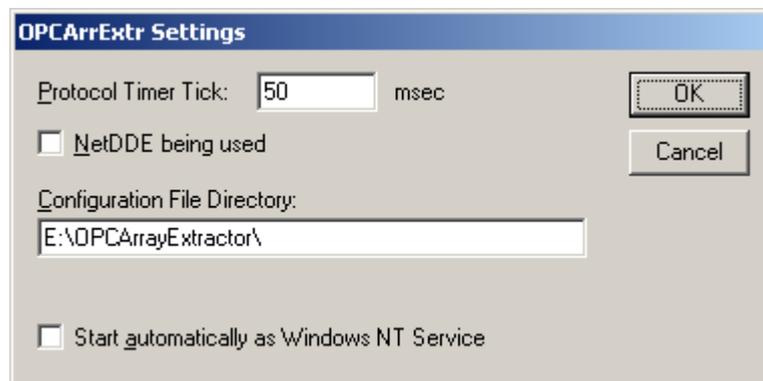
To access the commands used for the various configurations, open the */Configure* menu:



## Server Settings Command

A number of parameters that control the internal operation of OPC Array Extractor can be set. In most cases, the default settings for these parameters provide good performance and do not require changing. However, they can be changed to fine-tune the OPC Array Extractor for a specific environment.

To change the OPC Array Extractor's internal parameters, invoke the *Configure/Server Settings...* command. The "OPCArrExtr Settings" dialog box will appear:



The following describes each field in this dialog box:

### Protocol Timer Tick

This field is used to change the frequency at which the OPC Array Extractor checks for work to do - at this frequency OPC Array Extractor checks if there are new data obtained from corresponding OPC Server(s) and delivers new data to SuiteLink or DDE Client(s). If computer is very busy or some other MS Windows application is taking over the computer then OPC Array Extractor is activated rarely than setting in the **Protocol Timer Tick**.

**Note:** *The default value is 50 milliseconds. The minimum value is 10 milliseconds.*

### NetDDE being used

Select this option if you are networking using NetDDE.

### Configuration File Directory

The first field is used to specify the path (disk drive and directory) in which OPC Array Extractor will save its current configuration file. OPC Array Extractor will use this path to load the configuration file the next time it is started.

**Note:** *Only the "path" may be modified with this field. The configuration file is always named **OPCARREXTR.CFG**.*

**Note:** *There is no limit to the number of configuration files created, although each must be in a separate directory. When using the OPC Array Extractor with **InTouch**, it is good practice to place the configuration file in the application directory.*

### Start automatically as Windows Service

Enabling this option will cause the OPC Array Extractor to start as a Service.

Windows NT-technology-based operating systems offers the capability of running applications even when a user is not logged on to the system. This is valuable when systems must operate in an unattended mode. Enabling this option and rebooting the system will cause the OPC Array Extractor to run as a Service. However, to view configuration information or to reconfigure the OPC Array Extractor, the user must log on to the system. Any OPC Array Extractor related problems that may arise such as missing adapter cards, licensing failures or device drivers not loading will not be visible to the user until a log on is performed. Disabling this option and rebooting the system will cause the OPC Array Extractor to run as a Windows application program once again.

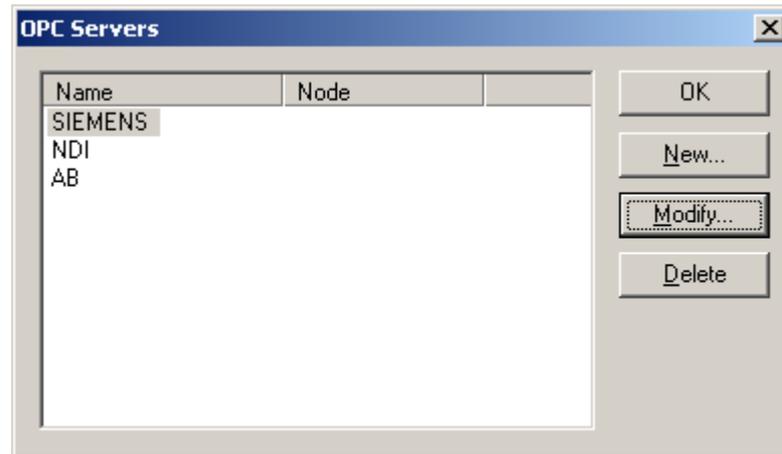
#### **Note:**

*The Service Startup configuration can be changed by MS Windows **Control Panel/Services** configuration dialogs. The **Allow Service to Interact with Desktop** checkbox in "Service" dialog box must be checked (the "Service" dialog box can be invoked by pressing the "Startup" button on "Services" dialog box when Service **OPCARREXTR\_IOServer** is selected). If **Allow Service to Interact with Desktop** is not selected then OPC Array Extractor full functionality is not ensured (e.g. the OPC Array Extractor configuration can not be changed, no message boxes will be displayed, etc.).*

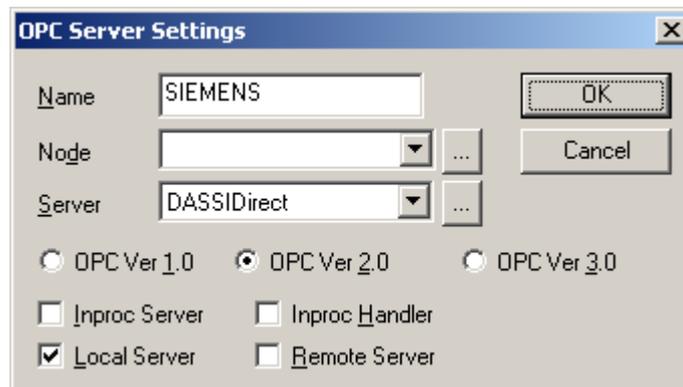
Once all entries have been made, click on **OK**.

### OPC Server Settings Command

This command is used to select and configure the OPC Array Extractor OPC Client part - the settings for OPC Server, providing OPC data, and to which the OPC Array Extractor OPC Client part will connect and communicate. Invoke the `/OPC Server Settings...` command. The "OPC Servers" dialog box like following will appear:



Click on **New** to create a new OPC Server definition or select the already defined OPC Server and click on **Modify** to edit the OPC Server settings - the "OPC Server Settings" dialog box will appear:



The following can be entered in this dialog box:

### Name

Enter the name of OPC Server. Any name can be entered here, this name will be used in "Topic definition" dialog as a reference to appropriate OPC Server.

### Node

Select the Node (computer name in local network) where the OPC Server resides.

### Server

Select (browse) the OPC Server from the list of available OPC Servers installed and registered on the selected Node.

### OPC standard version

Select the OPC standard (1.0 or 2.0; support for 3.0 is not yet implemented) supported by OPC Server. The default setting is OPC Ver2.0.

### OPC connection type

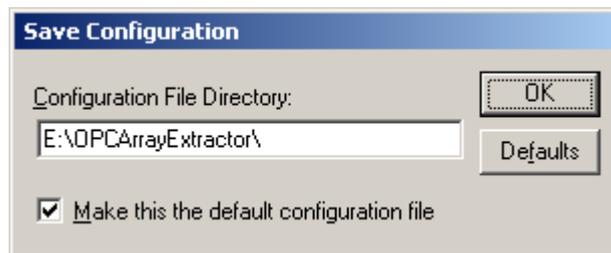
Select the OPC connection type(s) supported by OPC Server. The default and recommended setting is "Local Server".

Click **OK** to accept the new OPC Server settings and close the "OPC Server Settings" dialog box or click **Cancel** to reject the entered settings. The "OPC Server" dialog box will reappear.

Click **OK** to close the "OPC Server" dialog box.

## Saving of the OPC Array Extractor Configuration File

If the configuration file does not currently exist, or a new configuration path has been specified, the OPC Array Extractor will display the "Save Configuration" dialog box:



This dialog box displays the path where the OPC Array Extractor is going to save the current configuration file. The path may be changed if necessary. Also, the path can optionally be recorded in the **WIN.INI** file by selecting the "**Make this the default configuration file**" option. Doing so will allow the OPC Array Extractor to find the configuration file automatically each time it is started.

## Configuration File Location

When the OPC Array Extractor starts up, it first attempts to locate its configuration file by, first checking the **WIN.INI** file for a path that was previously specified. If the path is not present in the **WIN.INI** file, the Server will assume that the current working directory is to be used.

To start the OPC Array Extractor using a configuration file located in a different directory, click the Start button on the Taskbar, then choose Run and enter the following in the Open combo box:

**OPCARREXTR /d:directorynamepath**

For example, the following command allows to start the OPC Array Extractor using a configuration file located in directory e:\MyPrograms:

**OPCARREXTR /d:e:\MyPrograms**

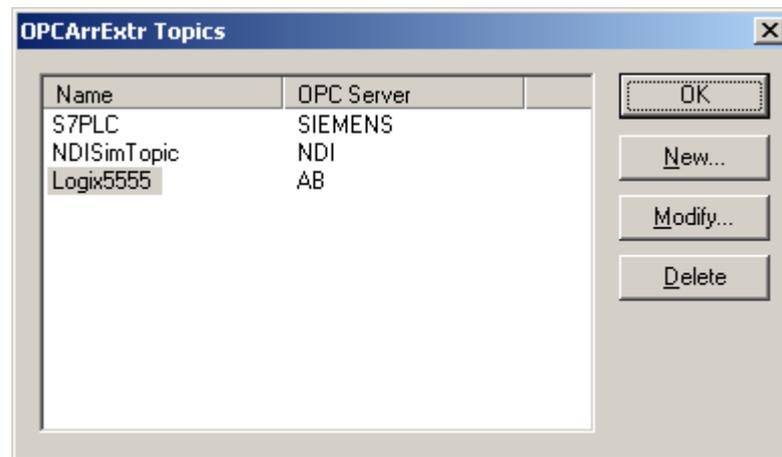
**Note:** There is no limit to the number of configuration files that may be created, although each must be in a separate directory.

## Topic Definition Command

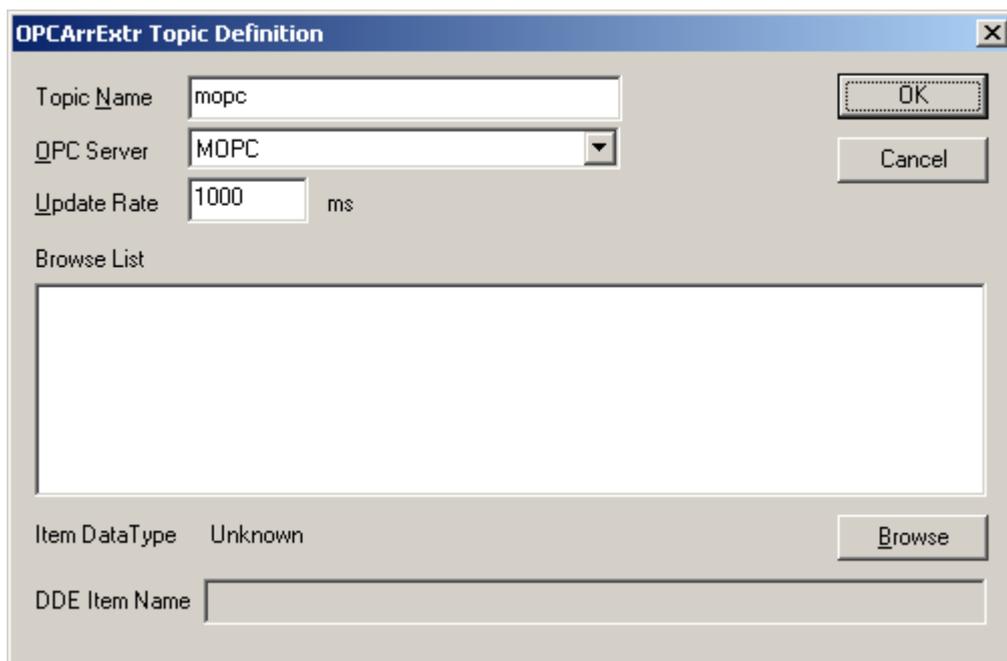
The user provides each OPC Server Group used to access the corresponding OPC Server items for this topic with an arbitrary name that is used as the Topic Name for all references to the OPC Server Group.

The following steps are taken to define the topics connected to OPC Array Extractor:

1. Invoke the Configure/Topic Definition... command. The "OPCArrExtr Topics" dialog box will appear:



2. To modify an existing topic, select the topic name and click on **Modify**. To define a new topic, click on **New**. The "OPCArrExtr Topic Definition" dialog box will appear:



3. Enter the **Topic Name**. If using **InTouch** the same Topic Name is to be entered in the "Add Access Name" dialog box described in the *Using the OPC Array Extractor with InTouch* section.
4. Click on the **OPC Server** button to associate a topic with the OPC Server (only OPC Server names entered in "OPC Server Settings" dialog box are available here). Additional topics may be associated with the same OPC Server later.
5. Set the **Update Rate** field to specify the 'update rate' for OPCArrExtr internally created OPC Group used to access the corresponding OPC Server items for this Topic. This is a time how often OPC Server updates its cache and sends notifications

to OPC Client (OPC Array Extractor) in purpose to keep the data fresh (the OPC Server should never send data to OPC Client at a rate faster than the client requests). Default value **0** means the data are updated as fast as possible, some non-zero value can be sometimes useful to decrease the load of OPC Server.

- Click on **Browse** to get the list of available OPC items. By clicking on OPC item name in "Browse List" area, the OPC data type of item will be displayed in "Item Data Type" field and the proper SuiteLink/DDE item name will be suggested in "DDE Item Name" field (from here the item name can be taken by Copy/Paste) - with one character **prefix** added to specify the correct data type and **[x]** added to specify this is an array (see *Item Names* section below):

The screenshot shows the 'OPCArrExtr Topic Definition' dialog box. It contains the following fields and controls:

- Topic Name:** mopc
- OPC Server:** MOPC
- Update Rate:** 1000 ms
- Browse List:** A list box containing several OPC item names. The selected item is 'MyM1.RES.RealArray'.
- Item DataType:** VT\_R4 | VT\_ARRAY
- DDE Item Name:** rMyM1.RES.RealArray[x]
- Buttons:** OK, Cancel, and Browse.

Once all entries have been made, click on **OK**.

Select **OK** in "Topic definition" dialog box when configuration for all Topics has been performed.

## Item Names

The OPC Array Extractor provides single elements of OPC arrays for DDE, FastDDE or SuiteLink clients, not supporting array data types; as well also single OPC items are supported. Therefore the items/points supported by OPC Array Extractor completely depend on available names of OPC items provided by OPC Server.

For **array elements** the correspondence between SuiteLink/DDE "single item" name (activated for OPC Array Extractor SuiteLink/DDE Server part) and OPC array name (used in OPC Server) is very simple - the "single item" name is equal with OPC array name plus one character prefix is added to specify the data type (discrete, integer, real or string) plus suffix with array element address (index) is added at the end.

For **single items** the SuiteLink/DDE item name is equal with OPC item name plus one character prefix is added to specify the data type (discrete, integer, real or string).

Both array element items and single items are Read/Write. For array element items at first whole array is read from OPC Server and then whole array including new (inserted) array element value is written back to OPC Server.

OPC Array Extractor supports also special “**array in string**” type items, where array (or part of array) data is put into single string item; in this case special rules are used to store the data, see “*Array in string*” type items section below. The “array in string” items are Read Only.

Each OPC Array Extractor item name must start with a prefix:

**d** or **D** - discrete value

**b** or **B** - byte (8-bit) value

**i** or **I** - short integer (16-bit) value

**l** or **L** - long integer (32-bit) value

**r** or **R** - real (floating point) value

**m** or **M** - message (string) value

As default, the OPC Array Extractor array element item name suffix is in following format:

**[n]**

where **n** is the array element address (index). The format of this suffix and the start index (default is 0) can be configured by using special WIN.INI file [OPCArrExtr] section entries, see *Troubleshooting/WIN.INI entries* section later in this manual.

For “**array in string**” type items after **[n]** (or after other item name suffix) additionally the length of array (number of array elements) must be added, see “*Array in string*” type items section below.

#### Examples:

1. If OPC integer array name is **Proc1.INT20** then “single item” name for this OPC array 5<sup>th</sup> element is **iProc1.INT20[4]** – in case default settings (suffix format **[n]** and start index **0**) are used.

2. In case the different from default suffix format (**n**) and start index **1** are configured in WIN.INI file [OPCArrExtr] section, then for OPC integer array **Proc1.INT20** the “single item” name for this OPC array 5<sup>th</sup> element is **iProc1.INT20(5)**.

## ”Array in string” type items

The “array in string” items allow receive the array (or part of array) data inside single string item, thus allowing significantly reduce the amount of items needed to receive whole array data. In this case special rules are used for item naming and to store the array data inside the string.

“Array in string” items are Read-Only string (message) type items. The item name prefix indicates the OPC array data type (not data type of item itself, like for array elements or single items).

The maximum length of “array in string” item string is 256 bytes. Some DDE or SuiteLink Clients does not support so long strings, so that must be carefully checked when using “array in string” items – for example, Wonderware InTouch supports only 131 bytes long strings.

For “array in string” type items after [n] (or after other item name suffix) additionally the length of array (number of array elements) must be added and the number inside [n] is the start address in the array (element address in the array).

The data inside the “array in string” item is the HexASCII representation of the array binary data in big endian format, the maximum possible length of arrays what can be get within one “array in string” type item is the following:

- 128 elements for discrete values array, item name prefix "d" (one discrete value needs 2 bytes);
- 128 elements for bytes array, item name prefix "b" (one byte value needs 2 bytes in the string);
- 64 elements for short integers (16-bit) array, item name prefix "i" (one short integer value needs 4 bytes in the string);

For example: short integer (16-bit) OPC array consisting from 2 elements, where the 1st element value is **3** and 2nd element value is **43776** (AB00 in hex); this array consists of 4 following bytes (binary, in hex format): **00 03 AB 00**

The corresponding “array in string” item can have item name like **iARRAYNAME[0]2** and will contain the following 8 bytes:

**0003AB00** (in ASCII format)

or same in binary hex format:

**30 30 30 33 41 42 30 30**

- 32 elements for long integers and floats (32-bit) arrays, item name prefixes "L" and "r" (one long integer or float value needs 8 bytes in the string);

- for OPC string arrays (arrays of text values, OPC Canonical Data Type VT\_ARRAY | VT\_BSTR, or 2008 in hex) the maximum number of elements (strings) in “array in string” item depends on lengths of OPC array single elements (strings); the following system is used: each single element (string) is put into “array in string” item as it is plus additional separator byte (FF in hex) is added to indicate the end of single element; in case “array in string” item all 256 bytes are in use and OPC array last single element (string) does not fit in then last single element (string) will be truncated.

Examples of “array in string” item names (“array start index” setting is equal to 0):

**iProc1.INT20[0]64**

The string containing 256 bytes with 64 short (16-bit) integers starting from 1st element of array

**iProc1.INT20[64]64**

The string containing 256 bytes with 64 short (16-bit) integers starting from from 65th element of array

**LProc2.INT30[0]32**

The string containing 256 bytes with 32 long (32-bit) integers) starting from 1st element of array

**LProc2.INT30[32]16**

The string containing 128 bytes with 16 long (32-bit) integers) starting from from 33rd element of array

**rProc7.REAL40[0]32**

The string containing 256 bytes with 32 floating point values starting from 1st element of array

**rProc7.REAL40[64]32**

The string containing 256 bytes with 32 floating point values starting from 65th element of array

**bProc7.B003[640]30**

The string containing 60 bytes with 30 data bytes starting from bytes array 641st byte

**mProc2.Str[1]2**

For OPC strings array consisting of 5 elements: "START", "OPEN", "CLOSE", "IN TRANSIT" and "STOP", the following 10 bytes long "array in string" will be delivered to DDE, FastDDE or SuiteLink client: "OPEN0xFFCLOSE"

## Monitoring and Controlling Communication with OPC Servers

For each topic, there are the following additional items offered by OPC Array Extractor to monitor and control the communication with corresponding OPC Server.

### STATUS

For each topic, there is a built-in discrete item that indicates the state of communication between OPC Array Extractor and corresponding OPC Server (as well it indicates OPC Server and field device communication state in case in "Topic Definition" dialog box the "Status Item" is specified). The discrete item (**STATUS**) is set to **0** when communication fails and set to **1** when communication is successful.

From **InTouch** the state of communication may be read by defining an I/O Discrete tagname and associating it with the topic configured for the corresponding OPC Server and using **STATUS** as the item name.

From **Excel**, the status of the communication may be read by entering the following formula in a cell:

**=OPCARREXTR|topic!STATUS**

### ITEMCOUNT

The **ITEMCOUNT** item is an Integer type Read Only item used to access the number of active items in the corresponding topic, including the ITEMCOUNT itself. The range of valid values is from 1 to 2147483647.

### ERRORCOUNT

The **ERRORCOUNT** item is an Integer type Read/Write Only (unique for each topic) used to indicate/reset the number of OPC items with errors (received bad quality values occurred since topic initialization or last reset). The **ERRORCOUNT** value can be reset by writing 0 to this item. The range of valid values is from 0 to 2147483647.

### ERRORITEMS

The **ERRORITEMS** item is an Integer type Read Only (unique for each topic) used to indicate the number of invalid items with incorrect syntax or which refer to invalid OPC array names.

### WATCHDOG

The **WATCHDOG** item is an Integer type Read Only (unique for each topic) used to provide the value of computer system time current seconds. This value is updated at OPC Array Extractor each Protocol Timer Tick (see *Server Settings Command* section above). The value range is from 0 to 59.

### OPCRESTART

The discrete Write Only item **OPCRESTART** is used to force closing and then – starting of OPC Server, specified in WIN.INI file. The specific OPC Server can be closed by setting the item to 1, setting to 0 have no effect.

## Using OPC Array Extractor with Suite Link and DDE Clients

The OPC Array Extractor is accessible from Suite Link clients (e.g. Wonderware InTouch) and DDE clients (e.g. MS Excel).

## Using the OPC Array Extractor with InTouch

To access OPC Array Extractor data from **InTouch**, the Access Names and Tag names should be defined in **WindowMaker**.

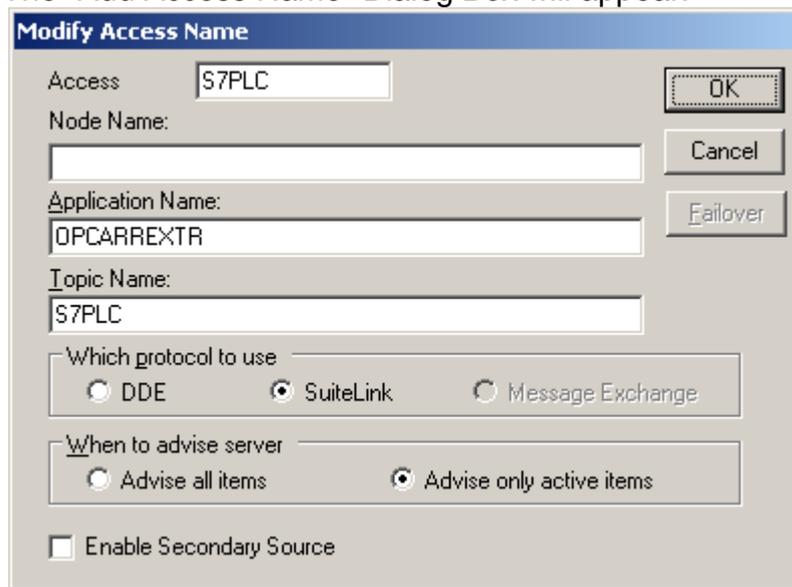
### Defining the Access Names

InTouch uses **Access Names** to reference real-time I/O data. Each Access Name equates to an I/O address, which can contain a **Node**, **Application**, and **Topic**. In a distributed application, I/O references can be set up as global addresses to a network I/O Server or local addresses to a local I/O Server.

To define the Access Names in WindowMaker node invoke the */Special/Access Names...* command. The "Access Names" dialog box will appear:



Click on **A**dd.... The "Add Access Name" Dialog Box will appear:



**Note:** If **A**dd is selected, this dialog box will be blank when it initially appears. Data has been entered here to illustrate the entries that are made.

The following fields are required entries when entering an Access Name Definition:

### Access Name

In the Access Name box type the name you want InTouch to use to this Access Name. (For simplicity, use the same name that you will use for the **Topic Name** here.)

### Node Name

If the data resides in a network I/O Server, in the Node Name box, type the remote node's name.

### Application Name

In the Application Name box, type the actual program name for the I/O Server program from which the data values will be acquired. In case the values (OPC data) are coming from appropriate OPC Server, the OPCARREXTR is used. Do not enter the .exe extension portion of the program name.

### Topic Name

Enter the name defined for the topic in the OPC Array Extractor to identify the topic the OPC Array Extractor will be accessing.

The Topic Name is an application-specific sub-group of data elements. In the case of data coming from OPC Array Extractor program, the topic name is the exact same name configured for the topic in the OPC Array Extractor.

**Note:** This will usually be the same as the "Access Name", although, if desired, they may be different. However, it must be the same name used when the topics were configured in **Configuring the OPC Array Extractor** section.

### Which protocol to use

Select the protocol (DDE or Suite Link) that you are using.

### When to advise server

Select **Advise all items** if you want the OPC Array Extractor program to activate all data whether or not it is in visible windows, alarmed, logged, trended or used in a script. Selecting this option will impact performance, therefore its use is not recommended.

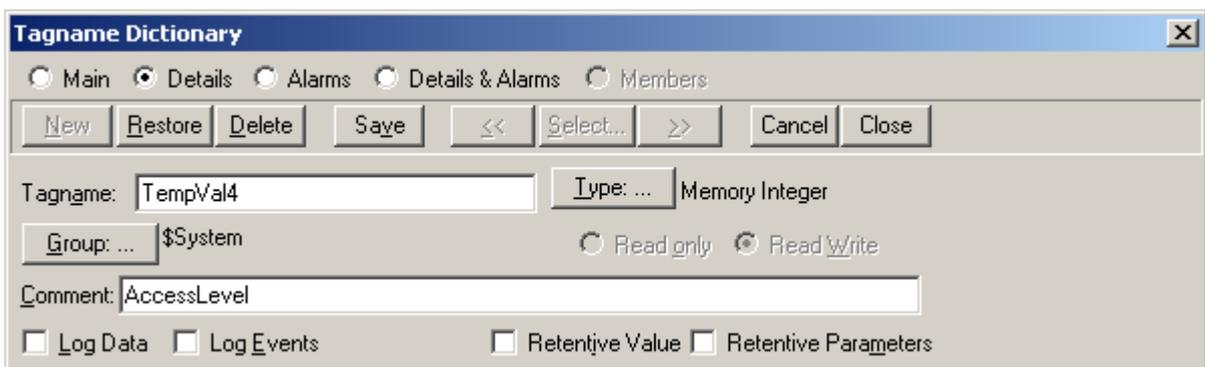
Select **Advise only active items** if you want the OPC Array Extractor program to activate only points in visible windows and points that are alarmed, logged, trended or used in any script.

Click **OK** to accept the new Access Name and close the "Add Access Name" dialog box. The "Access Names" dialog box will reappear displaying the new Access Name selected in the list.

Click **Close** to close the "Access Names" dialog box.

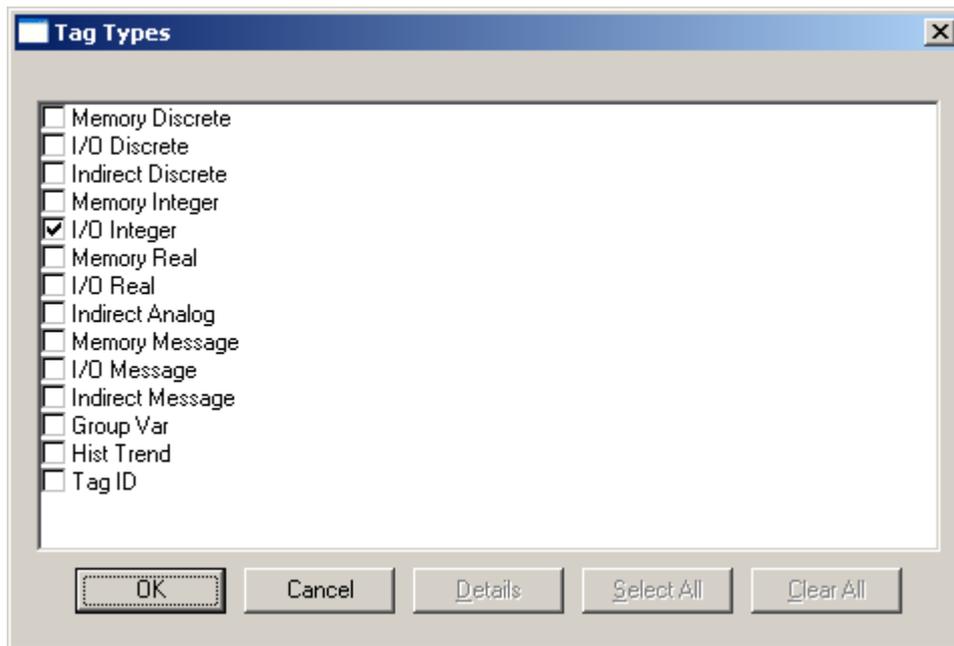
## Defining the Tag names

To define the Tag names associated with the new "Access Name", invoke the */Special/Tagname Dictionary...* command (in **WindowMaker**). The "Tagname Dictionary" dialog box will appear:



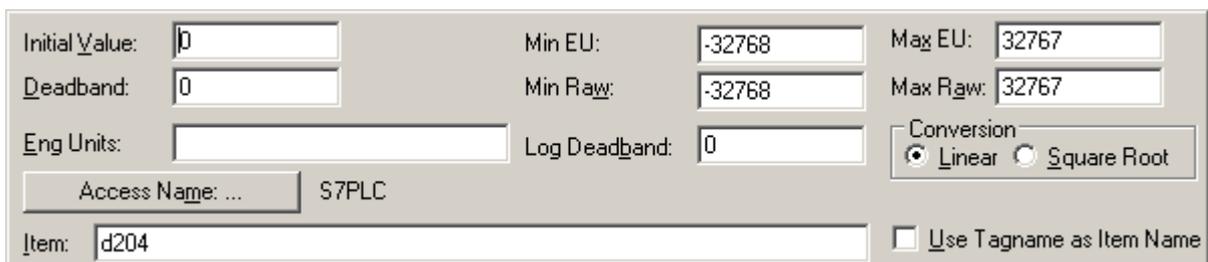
Click on **New** and enter the **Tag Name**. (The tag name defined here is the name **InTouch** will use. The OPC Array Extractor does not see this name.)

Select the tag type by clicking on the **Type:...** button. The "Tag Types" dialog box will appear:

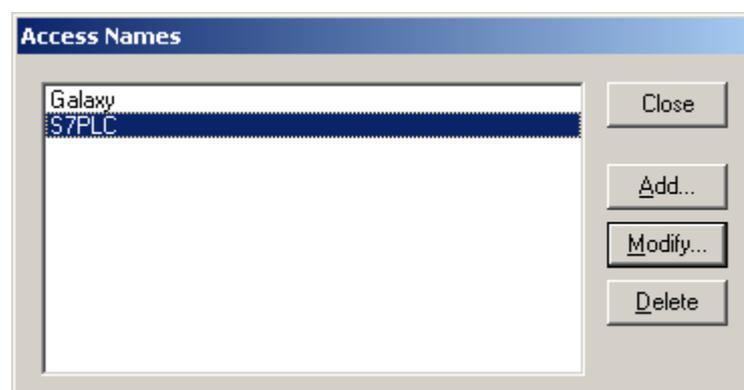


To access the elements of OPC arrays, the type must be **I/O Discrete**, **I/O Integer**, **I/O Real** or **I/O Message**. Select the Tag type.

The "Details" dialog box for the tag name will appear:



Select the Access Name for OPC Array Extractor by clicking on the **Access Name:...** button. The "Access Names" dialog box will appear:



Select the appropriate Access Name and click on **Close**. (If the Access Name has not been defined as previously described, click on **Add** and define the Access Name now.) The "Details" dialog box will appear displaying the selected Access Name:

Initial Value:	<input type="text" value="0"/>	Min EU:	<input type="text" value="-32768"/>	Max EU:	<input type="text" value="32767"/>
Deadband:	<input type="text" value="0"/>	Min Raw:	<input type="text" value="-32768"/>	Max Raw:	<input type="text" value="32767"/>
Eng Units:	<input type="text"/>	Log Deadband:	<input type="text" value="0"/>	Conversion <input checked="" type="radio"/> Linear <input type="radio"/> Square Root	
Access Name: ...	S7PLC				
Item:	<input type="text"/>				<input type="checkbox"/> Use Tagname as Item Name

Enter an OPC Array Extractor item name to be associated with this tagname in the **Item:** field in the "Details" box:

Initial Value:	<input type="text" value="0"/>	Min EU:	<input type="text" value="-32768"/>	Max EU:	<input type="text" value="32767"/>
Deadband:	<input type="text" value="0"/>	Min Raw:	<input type="text" value="-32768"/>	Max Raw:	<input type="text" value="32767"/>
Eng Units:	<input type="text"/>	Log Deadband:	<input type="text" value="0"/>	Conversion <input checked="" type="radio"/> Linear <input type="radio"/> Square Root	
Access Name: ...	S7PLC				
Item:	<input type="text" value="IDB20,WORD0,10[4]"/>				<input type="checkbox"/> Use Tagname as Item Name

(Refer to the **Item Names** section for complete details.)

Once all entries have been made, click on the **Save** button (in the top dialog box) to accept the new tagname. To define additional tagnames click on the **New** button. To return to the **WindowMaker** main screen, select **Close**.

## Monitoring the Status of Communication with InTouch

**InTouch** supports built-in topic names called **DDEStatus** and **IOStatus**, which are used to monitor the status of communication between the OPC Array Extractor and InTouch. For more information on the built-in topic names DDEStatus and IOStatus, see your online "InTouch User's Guide".

The status of communication between the OPC Array Extractor and InTouch can be read into **Excel** by entering the following DDE reference formula in a cell on a spreadsheet (in following examples **S7PLC** is the Topic Name configured for OPC Array Extractor):

**=view|DDEStatus!S7PLC** or **=view|IOStatus!S7PLC**

## Notes on Using Microsoft Excel

Data from OPC Array Extractor topics may be accessed from Excel spreadsheets. To do so, enter a formula like the following into a cell on the spreadsheet.

**=OPCARREXTR|topic!item**

Sometimes, Excel requires the topic and/or item to be surrounded by apostrophes.

In the formula, **topic** must be replaced with one of the valid topic names defined during the Server configuration process. Replace **item** with one of the valid item names described in the **Item Names** section.

## Reading Values into Excel Spreadsheets

Values may be read directly into Excel spreadsheets by entering a DDE formatted formula into a cell, as shown in the following examples:

```
=OPCARREXTR|S7PLC!'iDB20,WORD0,10[4]'  
=OPCARREXTR|Logix5555!'dalarms[28]'  
=OPCARREXTR|NDISimTopic!'rTag33[2]'
```

**Note:** Refer to the Microsoft Excel manual for complete details on entering Remote Reference formulas for cells.

## Troubleshooting

### WIN.INI entries

The first time you run the OPC Array Extractor configuration, most of the items in the following list will automatically appear in the WIN.INI file, located in the MS Windows system directory (e.g. C:\WINNT). It is an ASCII file and can be altered manually if you wish with any text editor, e.g., MS Windows Notepad (*do not use a program that formats text, such as MS Word or Write unless the file is saved as a DOS text*). The following is a typical entry for the OPC Array Extractor:

```
[OPCArrExtr]  
WinIconic=0  
WinFullScreen=0  
WinTop=112  
WinLeft=0  
WinWidth=200  
WinHeight=168  
ShowEvents=0  
ShowOPCCalls=0  
ShowOPCData=0  
ShowData=0  
ShowErrors=1  
Verbose=0
```

There are following additional WIN.INI entries available for OPC Array Extractor:

### StartIndex

The **StartIndex** entry can be used to specify the OPC array “start index” used in OPC Array Extractor item names. The default value is **0** – the OPC array 1<sup>st</sup> element will correspond to Array Extractor item name with suffix [0].

For example, to specify the “start index” equal to 1, the following entry can be used:

```
StartIndex=1
```

With above setting the OPC array 1<sup>st</sup> element will correspond to Array Extractor item name with suffix [1].

## Suffix

The **Suffix** entry can be used to specify the OPC Array Extractor item name suffix format. The default suffix format is **[n]**, where **n** is the array element address (index).

For example, to specify the item name suffix (**n**) to be used in OPC Array Extractor item names, the following entry can be used:

### Suffix=(n)

With above setting OPC Array Extractor item names must be ended with suffix in format (n), for example - iProc1.INT20(5).

## OPCRestart

The **OPCRestart** entry can be used to specify OPC Server process name (e.g., **OPCRestart=MOPCS31.exe**) to be closed when value of the DDE item **OPCRESTART** is set to 1. The entry is case sensitive.

## Troubleshooting menu

The following debugging choices are appended to the OPC Array Extractor's System Menu (the menu that appears when you click on the OPC Array Extractor icon in the upper left hand corner of the OPC Array Extractor window):

- ShowEvents** - if checked then OPC Array Extractor specific events are logged (not yet implemented).
- Show OPC Calls** - if checked then activity of OPC interface function calls is displayed.
- Show OPC Data** - if checked then OPC data received (OPC callbacks) is displayed.
- Show Array Data** - if checked then received array data is logged (use with care and for short time as produces very great amount of information logged).
- ShowErrors** - if checked then all information about errors is displayed.
- Verbose** - if checked then additional debugging information is displayed. This option is useful for getting additional information about error situations.
- Dump** - displays all information about OPC Array Extractor current internal data structures (topics, items, etc.). (not yet implemented)
- DumpScreen** - if checked then information about active OPC Server(s), topics and active arrays is displayed on OPC Array Extractor main window.
- ClearScreen** - if checked then OPC Array Extractor main window is cleaned.

All debugging information (except **DumpScreen** and **ClearScreen** is displayed via the Wonderware SMC (System Management Console) Log Viewer, which must be active for these commands to work.

Warning: if you check **Show OPC Calls**, **Show OPC Data** and/or **Show Array Data**, debug output grows very fast.

KLINKMANN AUTOMATION  
OPC Array Extractor  
Revision History

Mar 2007	Rev 0.90	First Release
May 2007	Rev 0.91	Single OPC items added
May 2008	Rev 0.92	“Array in string” type items added
Jun 2008	Rev 0.93	For “Array in string” type item names the “length of string item In bytes replaced with “length of array”.
Nov 2010	Rev 0.94	Software license key added, list of additional items is updated.
Dec 2012	Rev 1.0	OPC writes added.
Feb 2013	Rev 1.1	“Installing the OPC Array Extractor” section modified, by adding software licensing and new HASP HL Key support.