

# KEPServerEX Client Connectivity Guide

For OSI PI



KTSM-00012  
v. 1.03

Copyright © 2001, Kepware Technologies

## **KEPWARE END USER LICENSE AGREEMENT AND LIMITED WARRANTY**

The software accompanying this license agreement (the Software) is the property of Kepware Technologies, and is protected by United States and International Copyright laws and International treaty provisions. No ownership rights are granted by this Agreement or possession of the Software. Therefore, you must treat the Licensed Software like any other copyrighted material (e.g., a book or musical recording), except that you may make a single copy for backup or archival purposes. Your rights and obligations in its use are described as follows:

1. You may use and display this software on a single computer.
2. You may make one copy of the software for archival purposes or you may copy the software onto your hard disk and hold the original for archival purposes.
3. You may not modify or attempt to reverse engineer the software, or make any attempt to change or even examine the source code of the software.
4. You may transfer the software to another computer using the utilities provided. However, the software must be used on only a single computer at one time.
5. You may not give or distribute copies of the software or written materials associated with the software to others.
6. You may not sub-license, sell, or lease the software to any person or business.

### **Return Policy**

The original licensee of the software can return it within sixty (60) days of purchase. Please call us for a Return Material Authorization Number.

### **Limited Warranty**

Kepware does not warrant that the Software will be error free, that it will satisfy your planned applications or that all defects in the Software can be corrected. If Kepware provides information or assistance regarding the use of the Software or otherwise, Kepware is not assuming the role of engineering consultant. Kepware disclaims responsibility for any errors or omissions arising in connection with engineering in which its Software or such information or assistance is used.

The foregoing is the sole and exclusive warranty offered by Kepware.

Kepware disclaims all other warranties, express or implied, including but not limited to the implied warranties of merchantability and fitness for a particular purpose, with regard to the licensed software and all accompanying materials.

In no event shall Kepware be liable for incidental or consequential damages, including lost profit, lost savings, lost opportunities, or other incidental or consequential damages arising out of the use or inability to use the licensed software, even if Kepware has been advised of the possibility of such damages.

Kepware's entire liability shall be, at Kepware's option, either (a) return of the price paid for the Software (or component), or (b) repair or replacement of the Software (or component) that does not meet Kepware's Limited Warranty and which is returned to Kepware within the warranty period. This shall be the sole and exclusive obligation of Kepware and your sole and exclusive remedy with respect to any such failure. The Limited Warranty is void if failure of the Software (or component) has resulted from accident, abuse or misapplication.

### **Support**

Kepware provides *unlimited e-mail* support for all Software whether a demo or registered license. Kepware will provide a total of *two hours* free phone support for all registered Software after paying the applicable license fees. Kepware will provide *unlimited phone* support so long as you have paid Kepware any applicable maintenance or support fees and subject to the terms of those agreements. All corrections and maintenance releases will be made available through Kepware's Internet site. All major product releases of the Software are subject to upgrade fees. At no time will on-site support be provided without advance payment to Kepware for a minimum of two days on-site engineering support services, plus all expenses.

### **Trademarks**

Microsoft® and Microsoft Excel® are registered trademarks, Windows™ is a trademark of Microsoft Corporation.

32 Bit KEPServerEX Connectivity Guide

#### **Kepware Technologies**

P.O. Box 579

Portland, Maine 04112

**Sales:** (207) 775-1660 x208

**Technical Support:** (207) 775-1660 x211

**Fax:** (207) 775-1799

**E-mail:** sales@kepware.com or  
technical.support@kepware.com

**www.Kepware.com**

# Table of Contents

<b>INTRODUCTION TO KEPSERVEREX.....</b>	<b>1</b>
<b>ACCESSING KEPSERVEREX.....</b>	<b>1</b>
USING KEPSERVEREX DRIVERS.....	2
<b>OSI PI AS AN OPC CLIENT.....</b>	<b>3</b>
CONNECT TO THE SERVER FROM OSI PI.....	3
<i>Start the OSI Pi Process Services.....</i>	3
<i>Create PI Tags in the PI Pointbuilder.....</i>	3
STARTING THE OPC INTERFACE.....	5
CREATE A DISPLAY WITH PI PROCESSBOOK.....	7
<i>Open the PI ProcessBook.....</i>	7
<i>Create a New ProcessBook Project.....</i>	7
<i>Create a Display in the ProcessBook.....</i>	8
<i>Create a Trend Object in the Display.....</i>	9
Defining the Trend Object Properties.....	9
<i>Viewing Tag Data in the Trend.....</i>	13
SAVE AND CLOSE THE PROJECT.....	14
<b>KEPWARE'S OPC QUICKCLIENT AS AN OPC CLIENT.....</b>	<b>15</b>
CONNECT TO KEPSERVEREX FROM KEPWARE'S OPC QUICK CLIENT.....	15
<i>Auto Launch and Generate a Quick Client Project.....</i>	15
<i>Manually Create a Quick Client Project.....</i>	16
Browse for KEPServerEX.....	16
Add a Tag/Item Group.....	17
Set Group Properties.....	17
Add Tags/Items.....	17
<i>Check OPC Connection.....</i>	18



# Introduction to KEPServerEX

KEPServerEX is a 32 bit windows application that provides a means of bringing data and information from a wide range of industrial devices and systems into client applications on your windows PC. KEPServerEX falls under the category of a "Server" application. It is very common to hear the term "client/server application" in use across many software disciplines and business segments. In the industrial market, it has usually come to mean the sharing of manufacturing or production data between a variety of applications ranging from human machine interface software and data historians, to large MES and ERP applications.

Regardless of the business segment served, client/server applications have one thing in common: a standardized method of sharing data. In the industrial segment, many client/server technologies have been developed over the last ten years. Initially, some of these technologies were proprietary. In many cases these proprietary client/server architectures were in wide use but remained unavailable to third party applications. Early in the development of windows, Microsoft provided a generic client/server technology called DDE or Dynamic Data Exchange. DDE did provide a basic architecture that would allow many windows applications from a wide range of vendors to share data, but there was one problem. DDE was not designed for the industrial market. It lacked much of the speed and robustness desired in an industrial setting. However, this did not stop DDE from becoming dominant client/server architecture, largely due to its availability in most windows applications. In time, variations on Microsoft's DDE were developed by some of the leading vendors in the market. These variations addressed some of the speed and reliability issues of DDE but many people in the industrial segment agreed that a better system needed to be developed.

With the advent of 32 bit Operating Systems, and the use of Ethernet to provide communications between devices, there was a need for quicker and cleaner data transfer between software applications. This is where OPC saw its birth into the industry.

OPC (OLE for Process and Control) servers provide a standardized method of allowing multiple industrial applications to share data in a quick and robust manner. The OPC server provided in this package has been designed to meet the demanding requirements found in the industrial environment.

This OPC server has been designed as a two-part program. The primary component provides all of the OPC and DDE connectivity as well as the user interface functions. The second part is comprised of plug-in communications drivers. This two-part design allows you to add multiple communications options to your SCADA application while utilizing a single OPC server product thus reducing your learning curve as your project grows.

OPC technology reflects the move from closed proprietary solutions to open architectures that provide more cost-effective solutions based on established standards.

## Accessing KEPServerEX

A Windows based client application must be used to view data from the KEPServerEX application. In this section we will cover the basics of connecting a number of common OPC clients to KEPServerEX. While we cannot possibly cover every client application that exists, we believe that after reviewing this document you should be able to deal with most client applications.

The intention of this section is to show connectivity to KEPServerEX. It is assumed that you have already either configured your KEPServerEX application by selecting the appropriate driver and settings or you have run the Simulator demo (Simdemo.opf) which is included with KEPServerEX. For simplicity, the Simdemo project will be used for all examples contained in this section.

Before beginning any of the examples, start the KEPServerEX application by selecting it from your Start Menu or from its desktop icon. Once the server is loaded, use the File|Open command to load the “Simdemo” project. The KEPServerEX application is always active once you have opened an existing project or configured at least one channel and device in a new project. After you have selected a project, in this case the Simdemo project, KEPServerEX will automatically load this project when an OPC client application invokes KEPServerEX’s OPC server component.

Users have always had the ability to create what we refer to as “user defined tags” in their KEPServerEX application. Prior to OPC, defined tags gave a DDE application designer the ability to create a label for device data. Assume register 1000 contained the value of parts made, without defined tags a DDE application would have directly accessed register 1000. Using defined tags a label can be created like “PartsMade”. Now the DDE application could access the data via this new label, removing the machine level knowledge from the client application and keeping it at the server level where it belongs. This label, while useful for DDE is a necessity for OPC clients. For OPC clients, defined tags take on a greater role. Like the DDE example, defined tags allow you to create labels for your device data and keep the configuration of those tags in the server. OPC clients have a major advantage over DDE clients. OPC clients can browse the defined tags you create in your KEPServerEX application, which allows you to simply point and click on a tag to add it to your OPC client project.

*OPC Tag Browsing allows you to see a list of the defined tags you have created in your KEPServerEX application, directly within your OPC client application.*

For more information on defined tags see the “Designing a Project” section of the KEPServerEX help file, which can be accessed from the Help/Contents menu selection of the KEPServerEX application.

## Using KEPServerEX Drivers

Part of the innovative design of Kepware’s OPC/DDE Server Technology is the separation of the Hardware Protocol Driver from the Server Technology. This separation allows the user to use one or more drivers in the server at the same time. Each driver has its own help file that provides information on devices supported, communications parameters, cabling, addressing, and error messages.

The driver help file should contain all of the information you will need to connect your device to the PC so that the server can communicate with it. If you do not connect to the device, be sure to check the error messages and look up their meaning in the help file.

# OSI PI as an OPC Client

## Connect to the Server from OSI PI

The following steps will show you how to create an OPC connection to the Server from OSI PI. OSI PI works a little differently than other clients in that it uses several programs running in the background to handle connectivity. We will show you how to start the processes and how to get and display data from the server. All of our examples will use our “Simdemo.opf” project that is provided with every software installation. These examples show how to connect to the server using PI3. For information on connecting with PI2 see you OSI PI documentation.

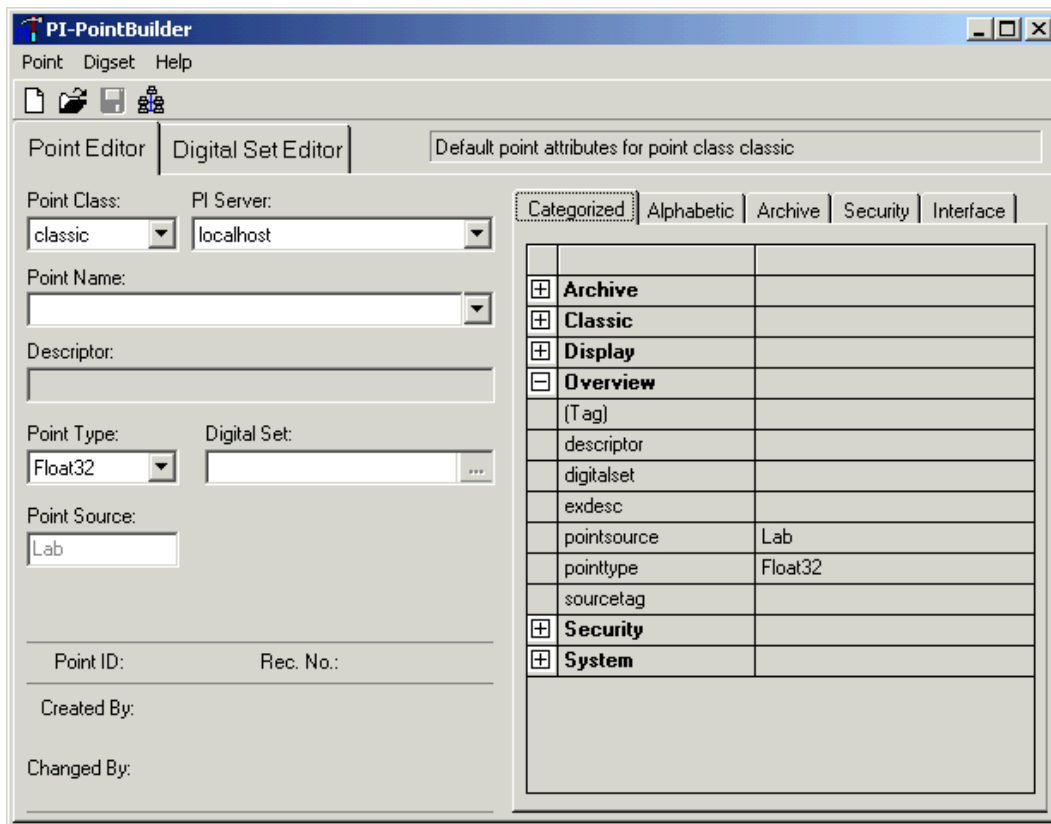
## Start the OSI Pi Process Services

First you will need to start the PI Process Services. This can take a few minutes so it is best if you start them at boot-up. You will want to create a shortcut in your Startup folder that points the “PISRVSTART.BAT” file locate in “C:\PI\ADM”. This file must run from the ADM folder.

*All of the PI software was installed using the default installation folders. If you did not use the defaults then you may have to use different path references.*

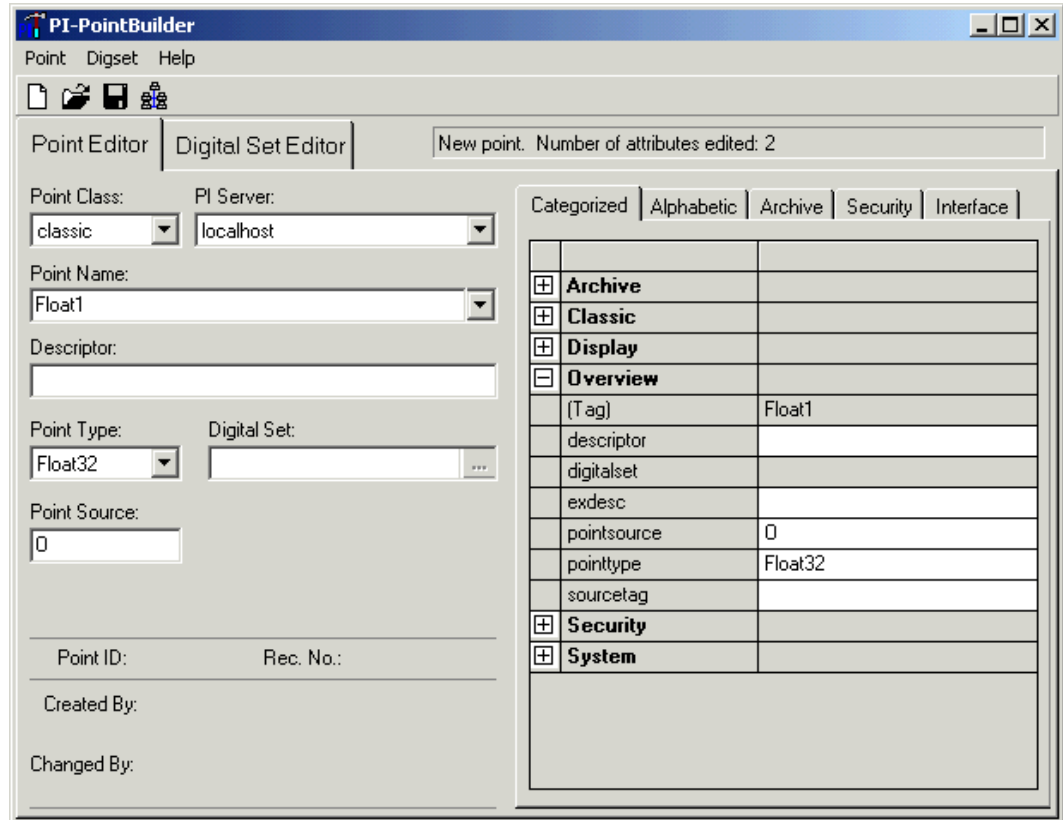
## Create PI Tags in the PI Pointbuilder

Now you need to create a point for use in your ProcessBook Project. To do this you need to open the PI Pointbuilder by selecting it from the PI System Start Folder.



By default, the Pointbuilder will open to add a new tag. First you will want to enter a Point Name (we named our tag “Float1”). Once you enter a Point Name, the Point Source field becomes available to edit. You will want to enter a capital “O” for OPC in the Point Source field. All tags

defined in the PI database and used by the OPC interface must share a common Point Source. Leave the Point Type at its default, since we are creating a Float.

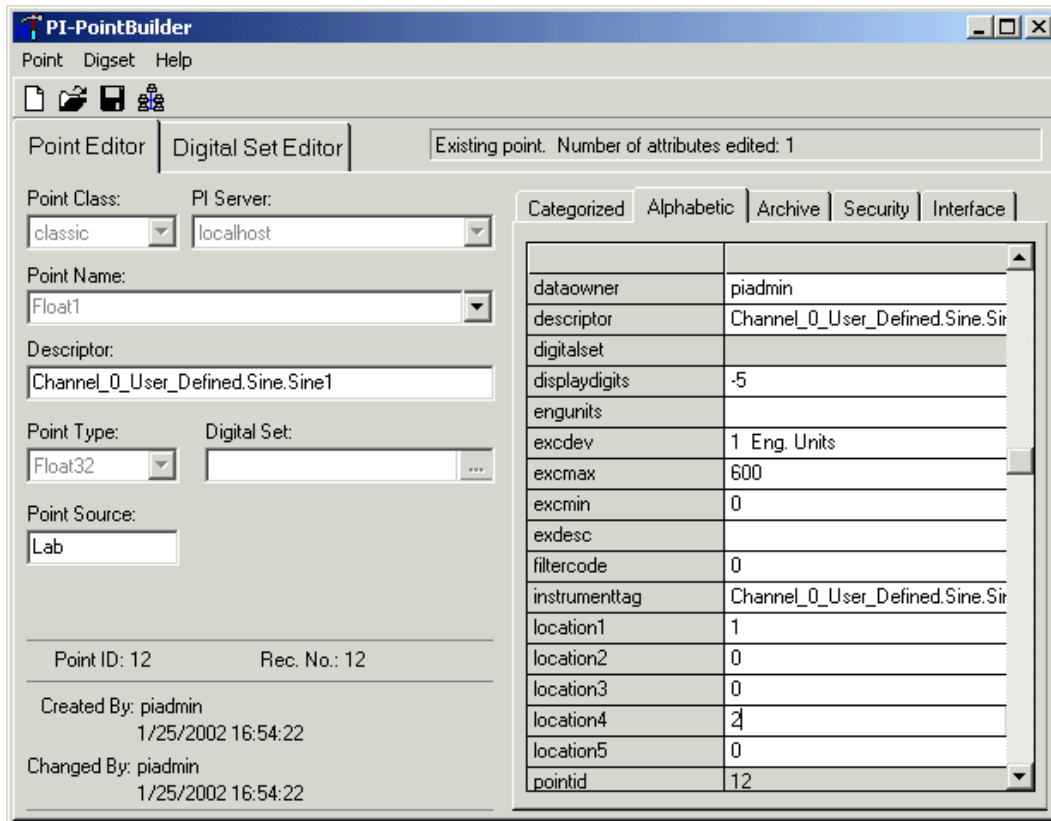


Next you are going to select the Alphabetic tab. This is where you will define the item in the server that you wish to use in PI. First, you will need to enter the full Item ID of the server item you want to use in the Instrumenttag field. We are using “Channel\_0\_User\_Defined.Sine.Sine1”. Next, copy the contents of the Instrumenttag to the Descriptor field, this provides text to display when using the tag if needed.

Now you are going to write a “1” to the Location1 field. This field specifies the interface instance number. In the **opcint.bat** file, there is a parameter for interface ID, /ID=#, where # is any number. The Location1 field for each tag must match that number, or the tag will be ignored. We will discuss the “.bat” file later.

The last thing you are going to do is write a “2” to Location4. This is the scan class number. Scan classes are defined in the interface startup file; each scan class defines an update period. This location code defines which scan class period is used to update the PI point. We are assigning this point to the second scan class period





Once you have entered all of information you can save the PI Point to the database.

## Starting the OPC Interface

If you have never used PI before, you will find a generic “.bat” file named “**opcint.bat**” in the “C:\Program Files\PIPC\Interfaces\OPCint” folder. You need to modify this file to allow the server to connect. The current file is configured as follows:

```

rem
rem Startup file for the OPC interface to PI
rem The ^ marks are continuation characters, they allow
rem you to have a command be split between multiple lines.
rem There must not be ANYTHING after the ^ on each line.
rem This is only a sample of the options available, the user
rem manual has the list and descriptions for them all.
rem
rem /ps=O           The pointsource -- this should match the pointsource for
rem                your tags
rem /ec=10          The event counter number for IORATES
rem /er=00:00:03   The requested update rate for event triggered tags
rem /id=1           The identifier string used in the pipc.log file for messages
rem                from this interface. It must match Location1 on the tags.
rem /SERVER=OPC.OSI.1 The OPC server name; format hostname::servername or
rem                just servename if it is local
rem /host=mabel:5450 The PI server name and port
rem /MA=Y           Should we try to add tags in large batches rather than
rem                singly ?
rem /ts=a           Where do we get timestamps ? (Y/N/A/U)
rem /stopstat      Write a status to PI tags when the OPC server goes away ?
rem /f=00:00:01   Scan classes. The first one is for Read On Change tags...

```

```

opcint ^
/ps=O ^
/ec=10 ^
/er=00:00:03 ^
/id=1 ^
/SERVER=OPC.OSI.1 ^
/host=localhost:5450 ^
/MA=Y ^
/ts=Y ^
/OPCstopstat ^
/f=00:00:01 ^
/f=00:00:01 ^
/f=00:00:01 ^
/f=00:00:02

```

#### TextBox 1: Generic OPCint.bat

Now you will modify the “.bat” file. First you are going to change the “/Server” parameter to read “KEPware.KEPServerEX.V4”. This is the program id of the KEPServerEX. Next, we are going to change the second “/f” scan class to read “00:00:00.1”. This directs the OPC Interface to request data from the server every 100 msec.

The last parameter you will need to know is how to change the “/host” parameter. This tells the OPC Interface where to look for the server. If you are running the server on the local PC then you will leave it set to “localhost”. However, if you are running the server on another PC then you should change it to the name of the PC running the server. You will also need to make sure that you have properly configured DCOM. You can find details on how to do that in the DCOM documents provided on the Kepware Technologies installation CD, at our Web site or in the OPCinc.doc (word) document provide by PI with the installation of the OPC interface. When you are done, you must save the edited batch file.

```

opcint ^
/ps=O ^
/ec=10 ^
/er=00:00:03 ^
/id=1 ^
/SERVER=KEPware.KEPServerEX.V4 ^
/host=localhost:5450 ^
/MA=Y ^
/ts=Y ^
/OPCstopstat ^
/f=00:00:01 ^
/f=00:00:00.1 ^
/f=00:00:01 ^
/f=00:00:02

```

#### TextBox 2: Edited OPCint.bat

The last thing you are going to do is run the batch file to start the OPC interface. This can take a few minutes if you are loading a large number of tags.

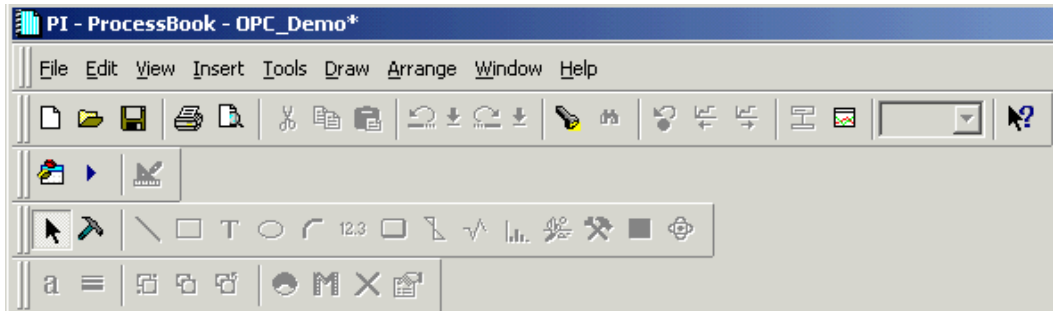
**Note:** See the *OPC Interface for OSI PI manual provide with you OSI PI software for more details on the parameters of the OPCint.bat file.*

# Create a Display With PI Processbook

We will now create a PI Process Book in which to monitor the server tag that we created.

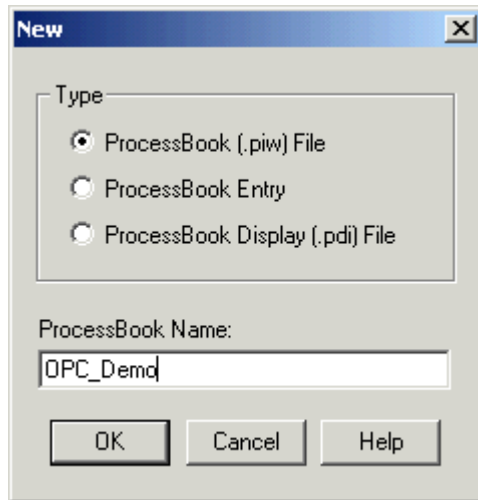
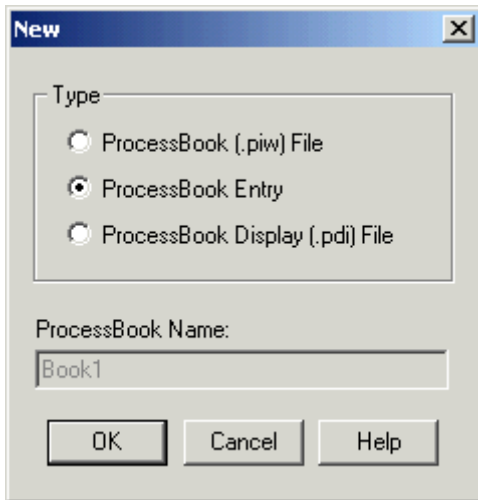
## Open the PI ProcessBook

From the PI Systems start folder, click on “PI ProcessBook” to start it. By default, it will open with the PI demo processbook file showing. You can close this and then click on **File|New** in the main menu to create a new Processbook file.



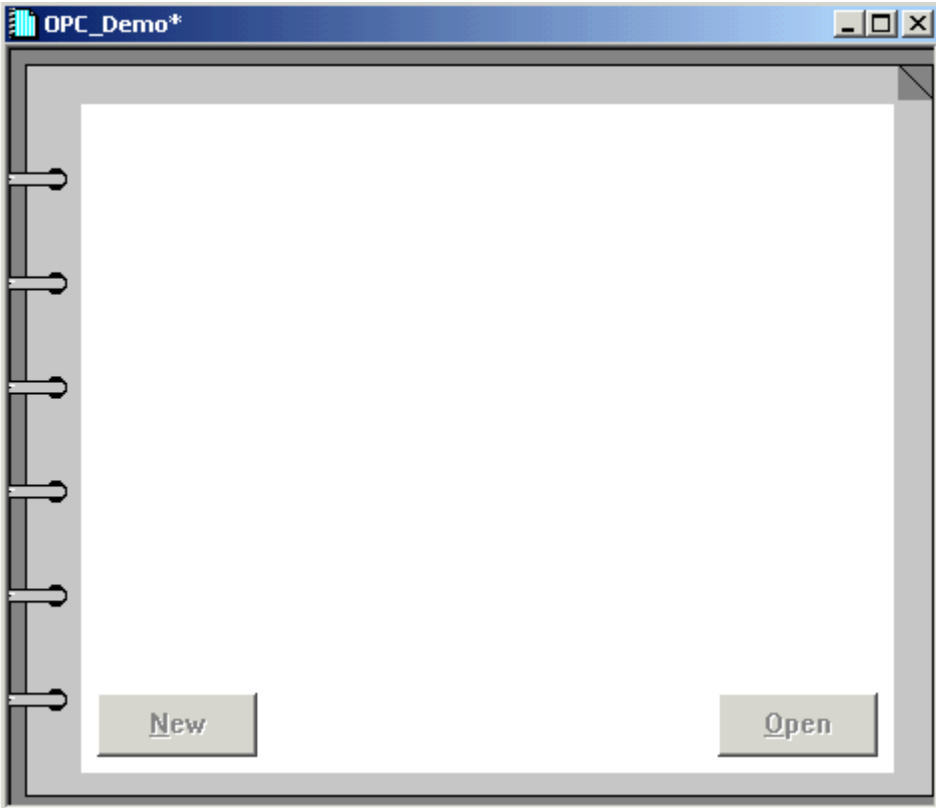
## Create a New ProcessBook Project

In the New dialog box you will want to select the radio button for “ProcessBook (.piw) file” and then give your new processbook file a name. We chose “OPC\_Demo” for our file name.

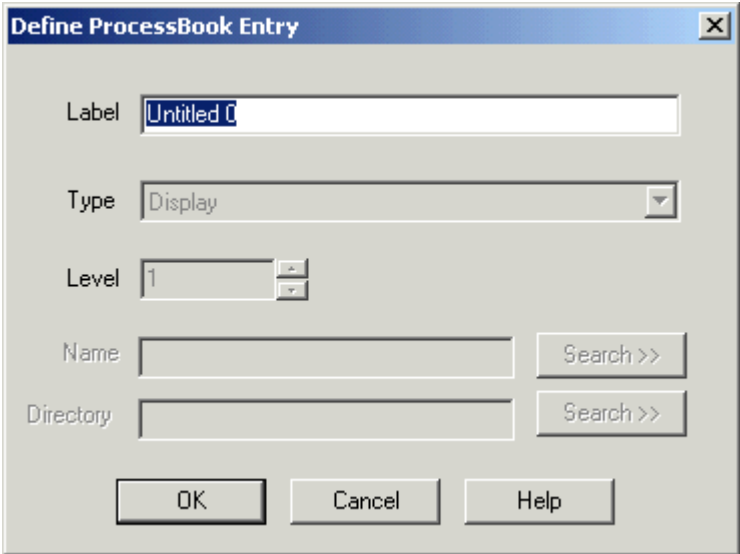


# Create a Display in the ProcessBook


When you click Ok in the dialog box you will get your new Process book. Next with your new process book selected, you are going to select Insert|Display from the main menu to add a display to the Process book.

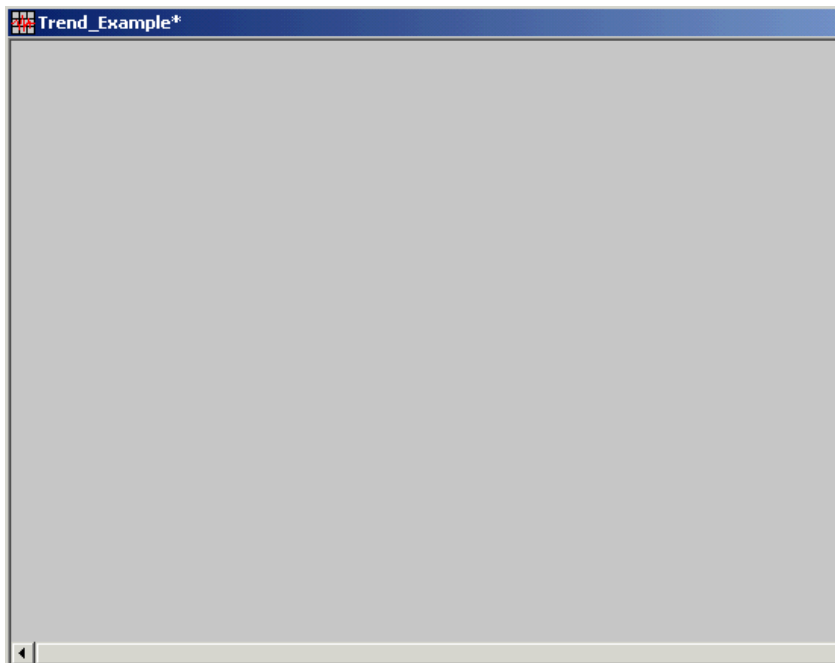


In the Define ProcessBook Entry dialog box you will need to enter a name for your display. We named our display "Trend\_Example". Click OK when you are done.



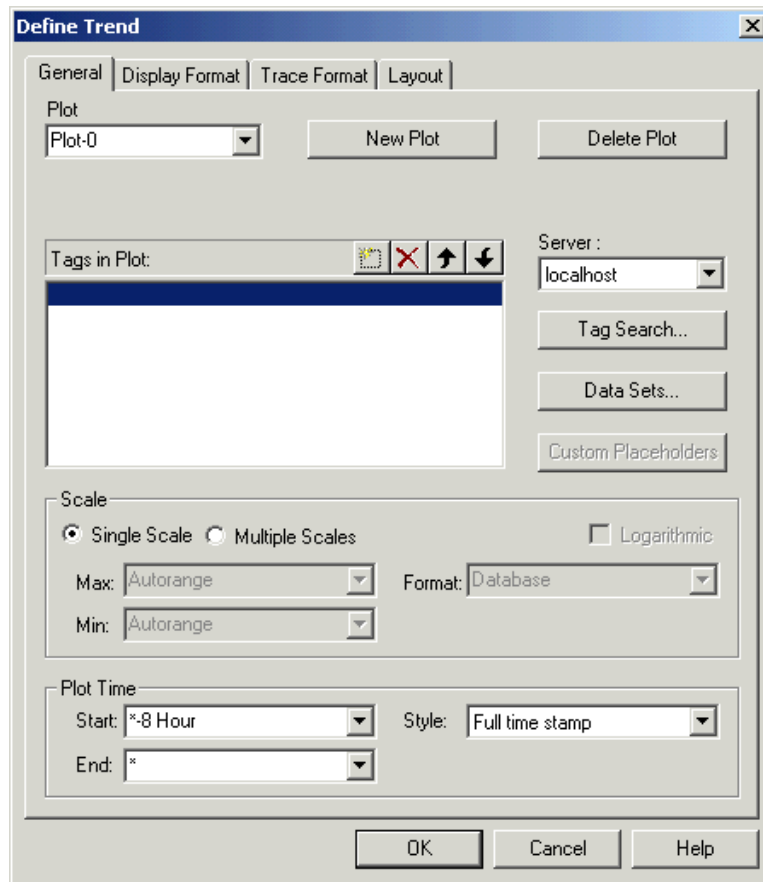
## Create a Trend Object in the Display

The new display should be open. As you may have guessed by the name of the display, we are going to put a trend object in our display. Click the Trend icon  from the button menu. Then with the mouse, draw a rectangle in the display that is the size you want for your trend.

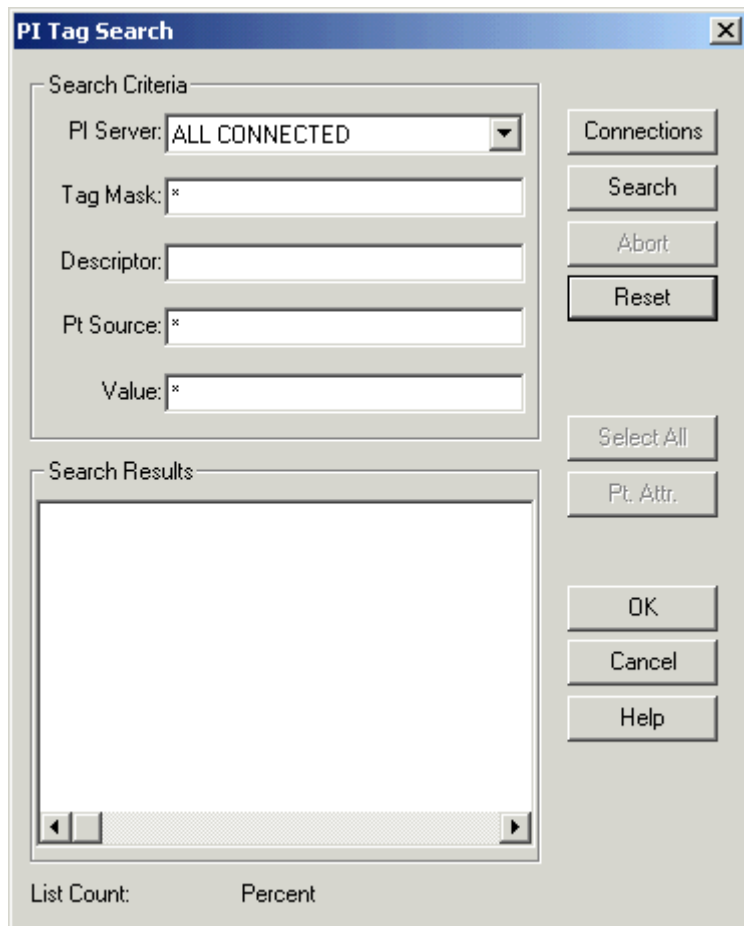


### *Defining the Trend Object Properties*

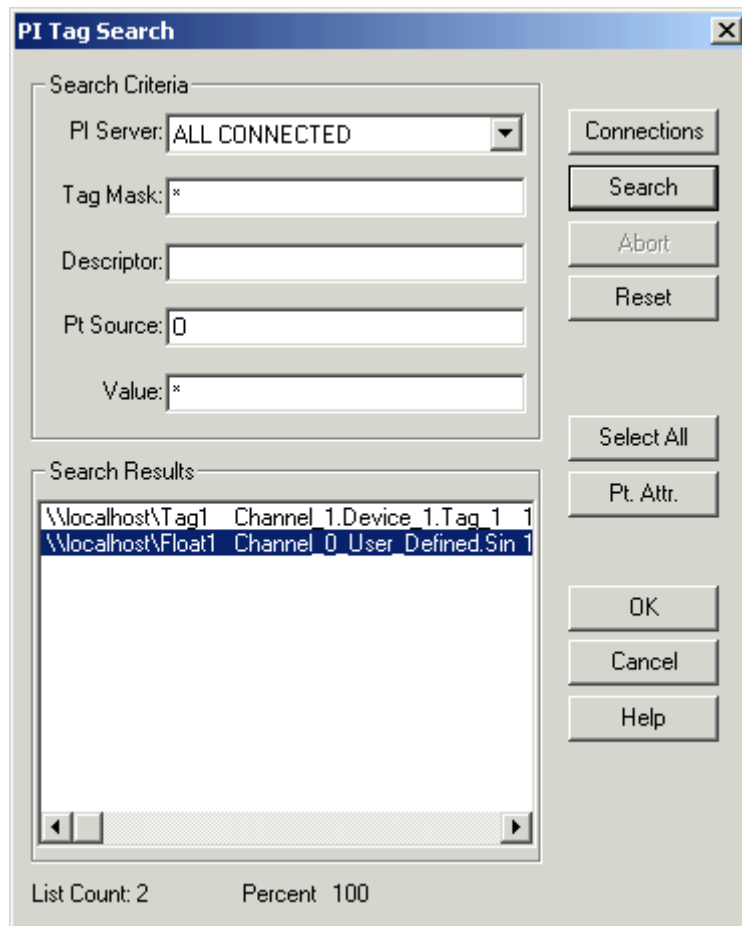
When you release the mouse button, the Define Trend dialog will automatically open. The first thing you need to do is tell the trend what data you are displaying in it. Click on the Tag Search... button to get the tag(s) that you will display.



You should now see the PI Tag Search Dialog box. If you click Search you will get all the tags that have been created in the PI Points database. Since you only want the tags we created, you will enter an "O" in the Pt Source field then click on Search.

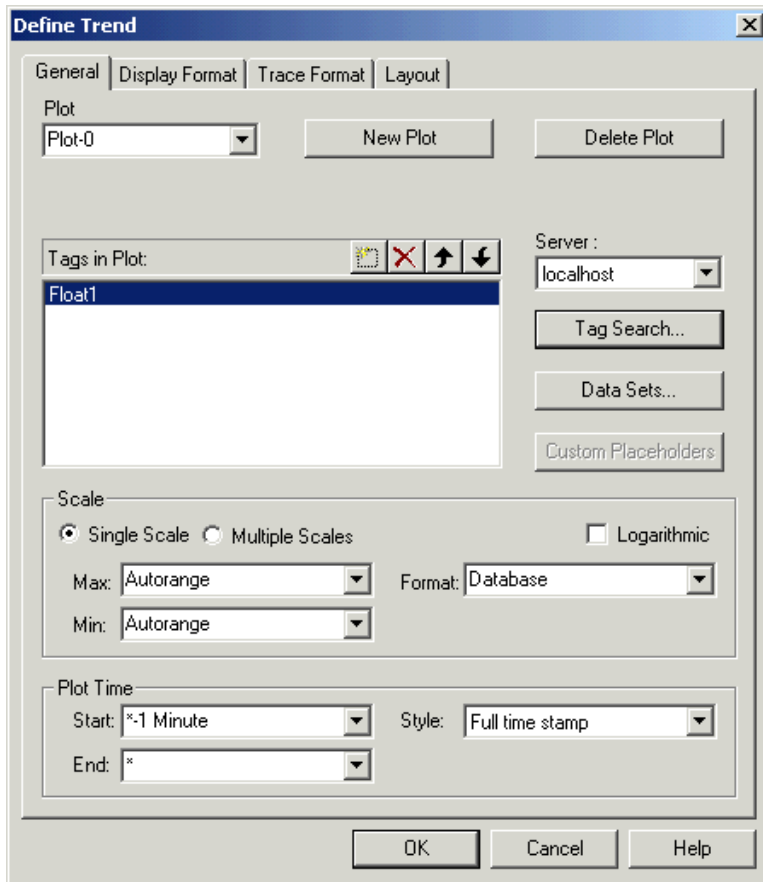


Now you should see the Float\_1 tag that we created, and any other tags you may have entered. You will want to highlight it by clicking on it with the mouse and then clicking OK to add it to the Trend properties.



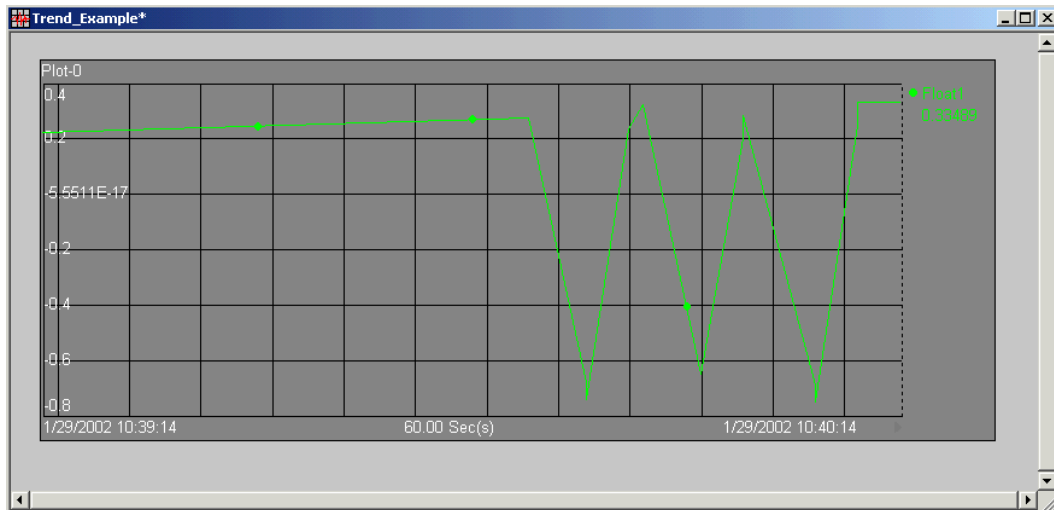
The tag name will now appear in the “Tags in plot” field of the Define Trend dialog. You still have one thing left to do. The tag that you are using in the server changes value about every 100msec. You want to be able to see these changes in the trend. The trend’s default display width is 8 hours. Change the Plot Start value to read “\*-1 Minute”, then click OK.





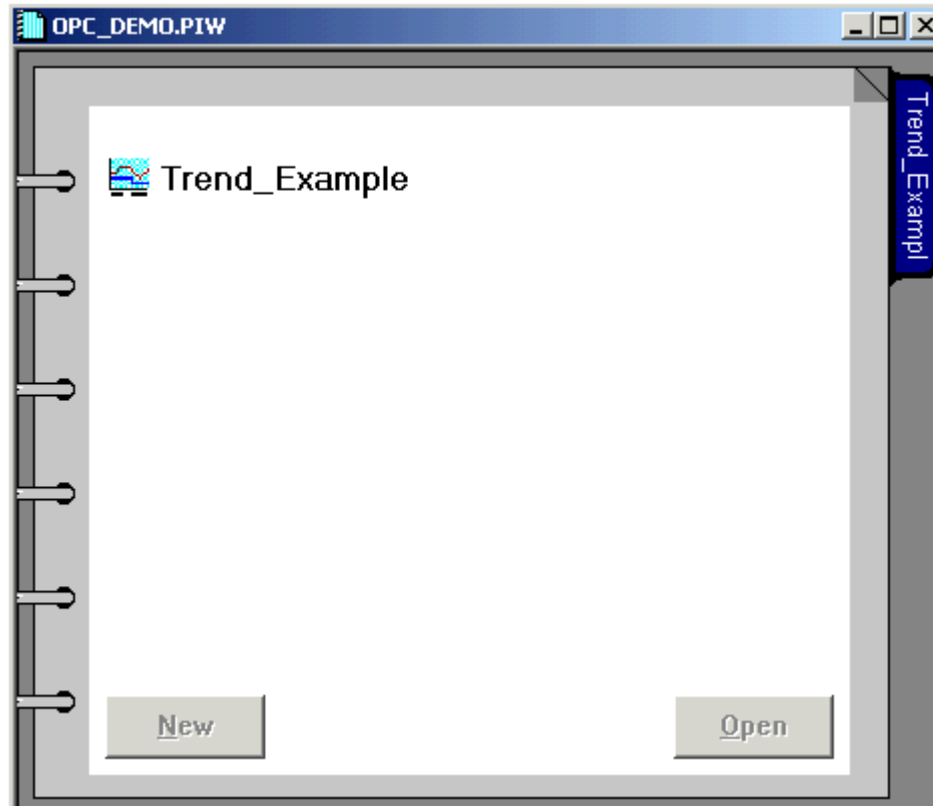
## Viewing Tag Data in the Trend

You should start seeing values plotted in the display right away. You may want to adjust plot times to get your data displayed correctly.



## Save and Close the Project

Once you save and close the display window, you will see it displayed in your process book. Don't forget to save your ProcessBook as well.



You should now be able to connect to, and display data from the KEPServerEX with OSI PI. For more information on using PI you will need to contact your local OSI PI representative.


# Kepware's OPC QuickClient as an OPC Client

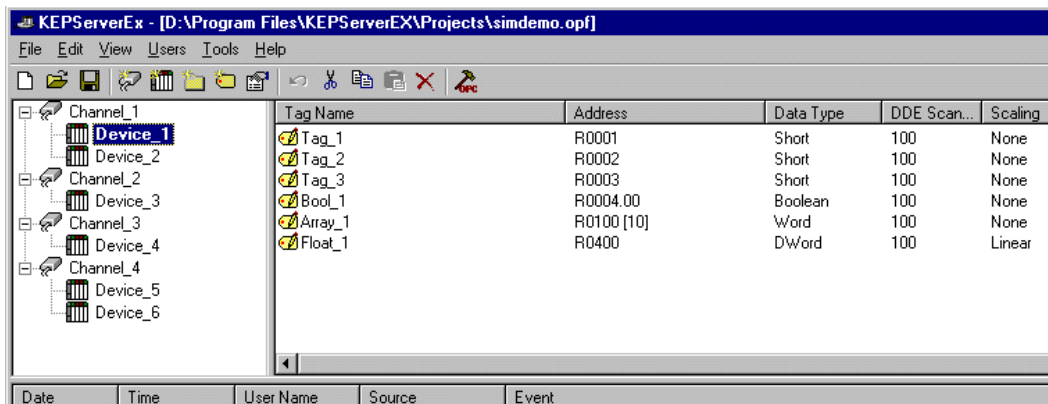
## Connect to KEPServerEX from Kepware's OPC Quick Client

Kepware's OPC Quick Client is an interface that can be used to connect to KEPServerEX. For this example, the OPC Server Version used is V4.41.163 and the OPC client version used is version 4.20.66. The following steps will show you how to create an OPC connection to the KEPServerEX from the Test Client either by using the auto project launch form the server or by manually starting the client and creating a project. Both examples use the "SimDemo.opf" project provided with the server install.

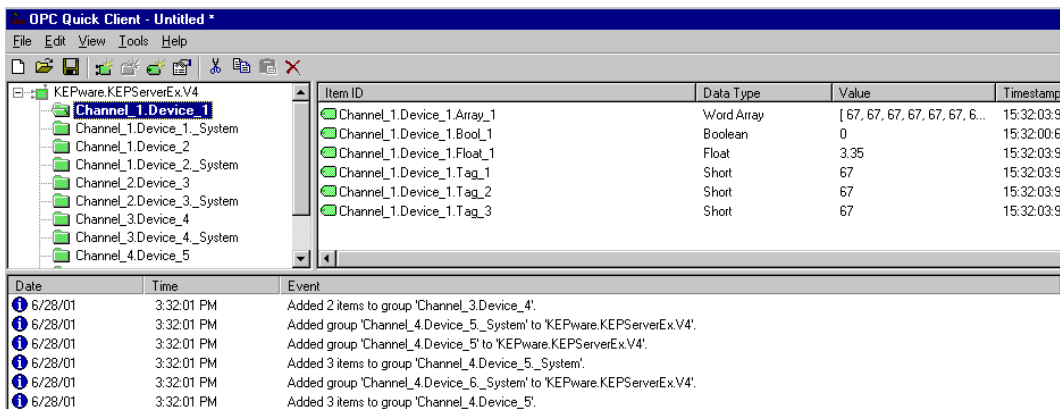
### Auto Launch and Generate a Quick Client Project

In most cases when you are using the Quick Client you are testing to see if you can see all of the tags you have defined in the server. To aid in this process we provide the ability to launch the client from the server and automatically generate the client project.

1. In the Server, once you have added all of the tags you currently want, click on the Launch button  or select "Tools|Launch OPC Quick Client".

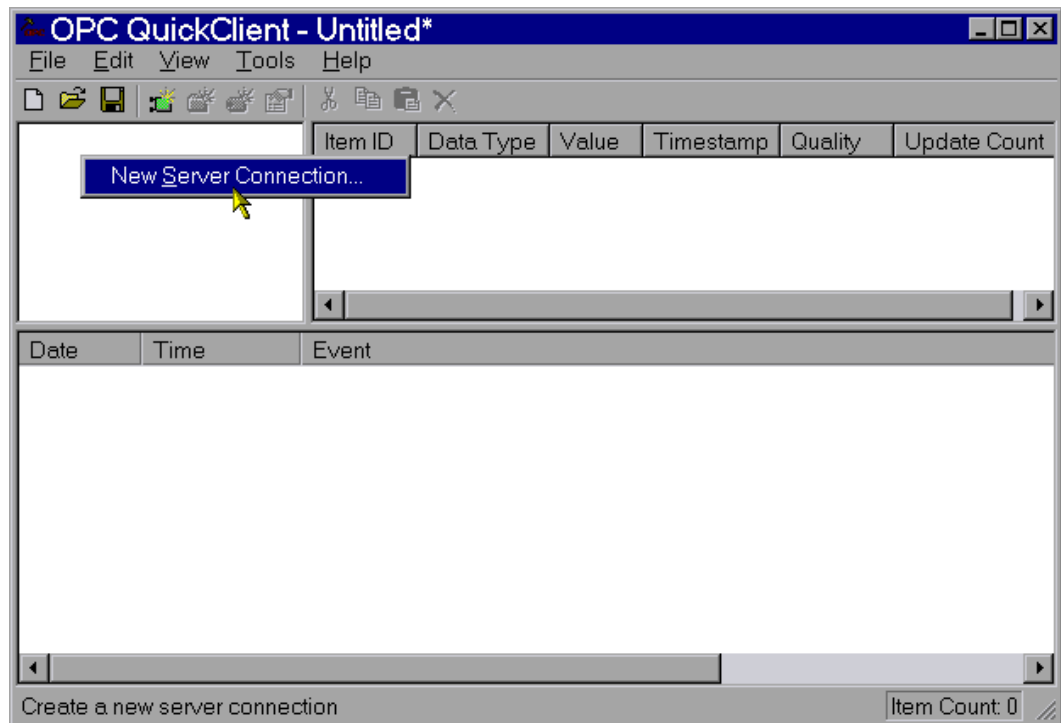


2. You will notice that the client project automatically opens to display the tags in the first Device group.



## Manually Create a Quick Client Project

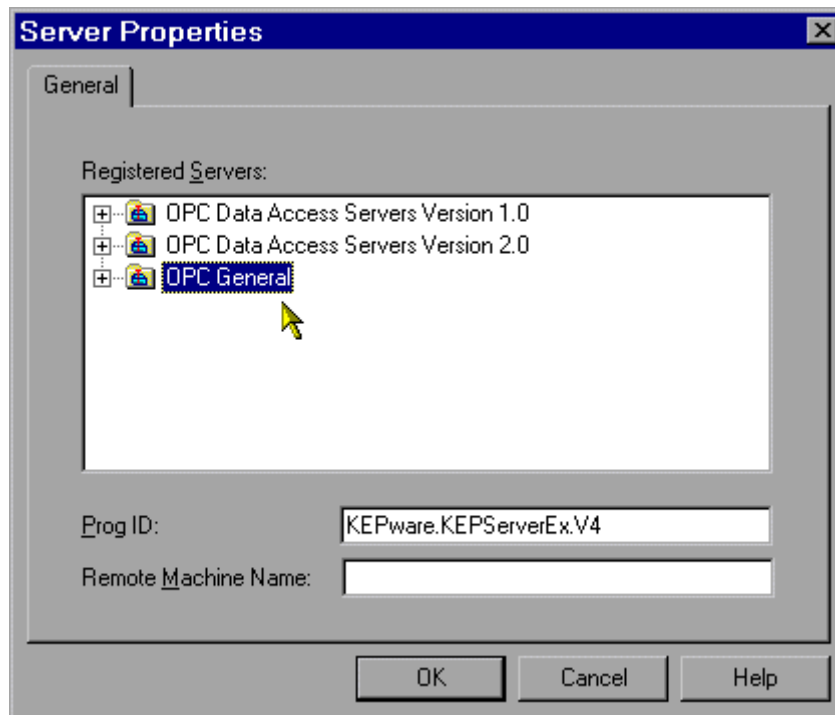
1. To start a new project, select New Server Connection... from the Edit menu or right click in the upper left display box and select New Server Connection...(shown below).



### Browse for KEPServerEX

2. In the Server Properties window, click OK because the KEPServerEX is set as the default server. If you were going to connect to the server running on another PC then you would enter the name of that PC in the Remote Machine Name field of the Server Properties window without any back slashes "\\".

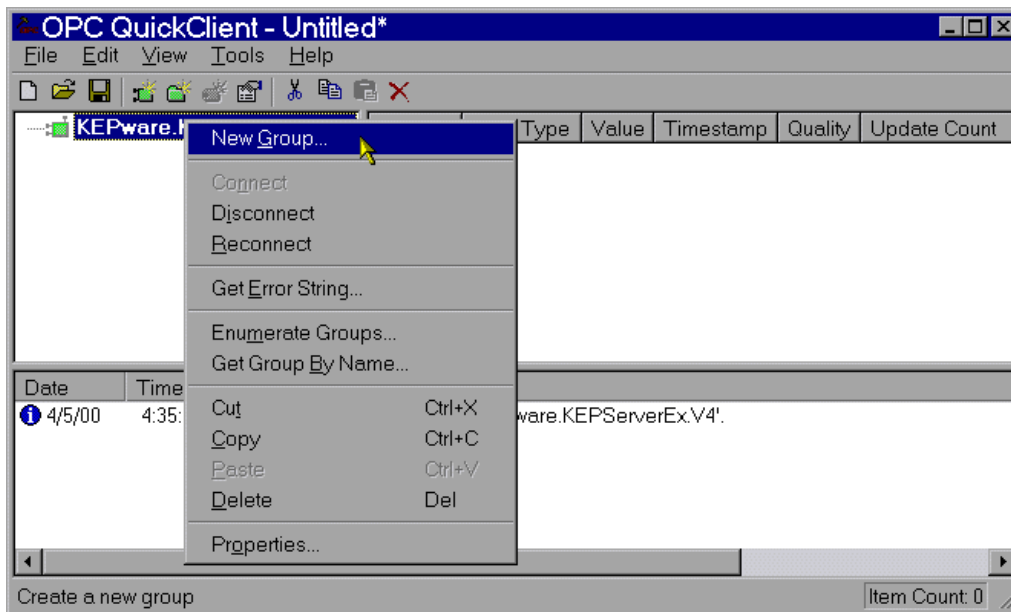
*For remote connections you have to be sure that you have DCOM properly configured. For information on how to configure DCOM see the two guides located either on the installation CD or at the Kepware web site.*



### Add a Tag/Item Group

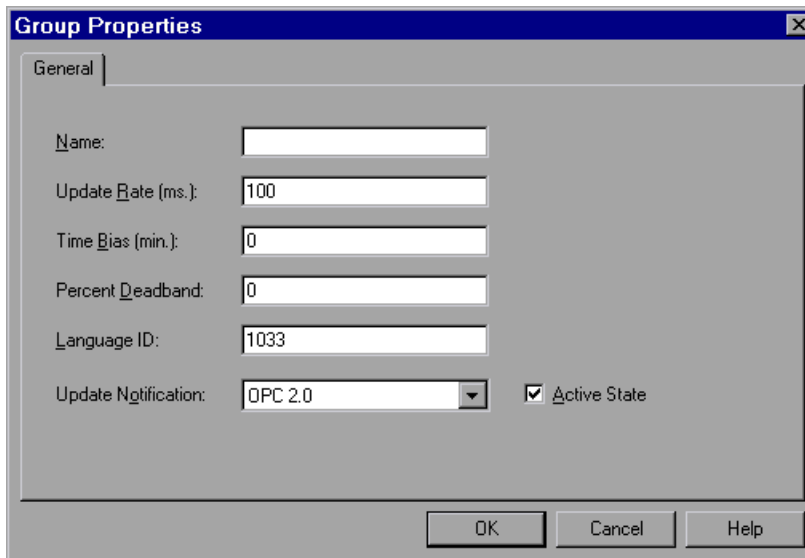
After choosing KEPServerEX, there should be a visual representation of the client connection to the server provided in the upper left-hand display.

3. Right click on this connection and choose New Group or select New Group from the Edit menu.



### Set Group Properties

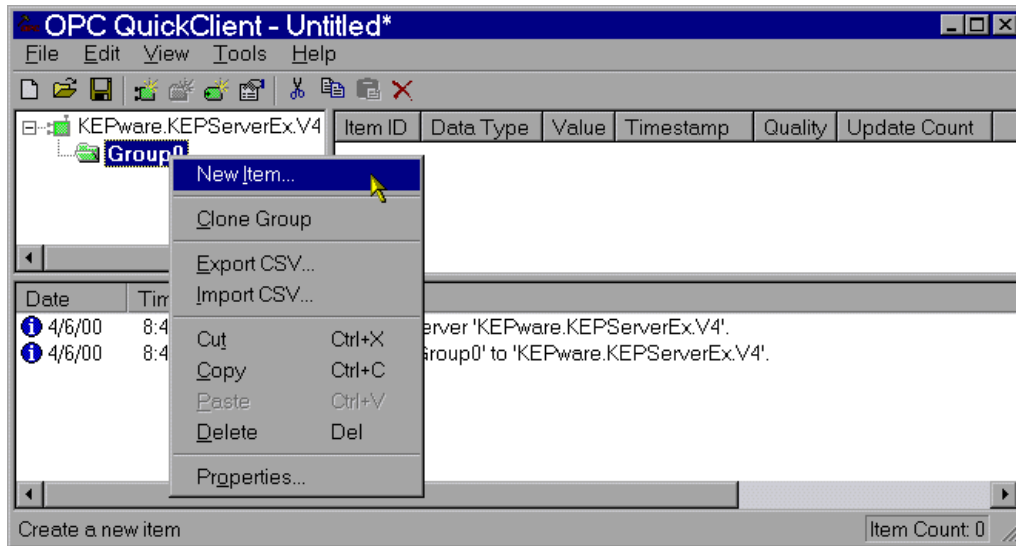
4. In the Group Properties window, enter a Name to identify the group. If no group name is entered, the server will generate a unique name for you. Also, make sure the Active State is enabled so the client will show active items for the group. Finally, click OK.



### Add Tags/Items

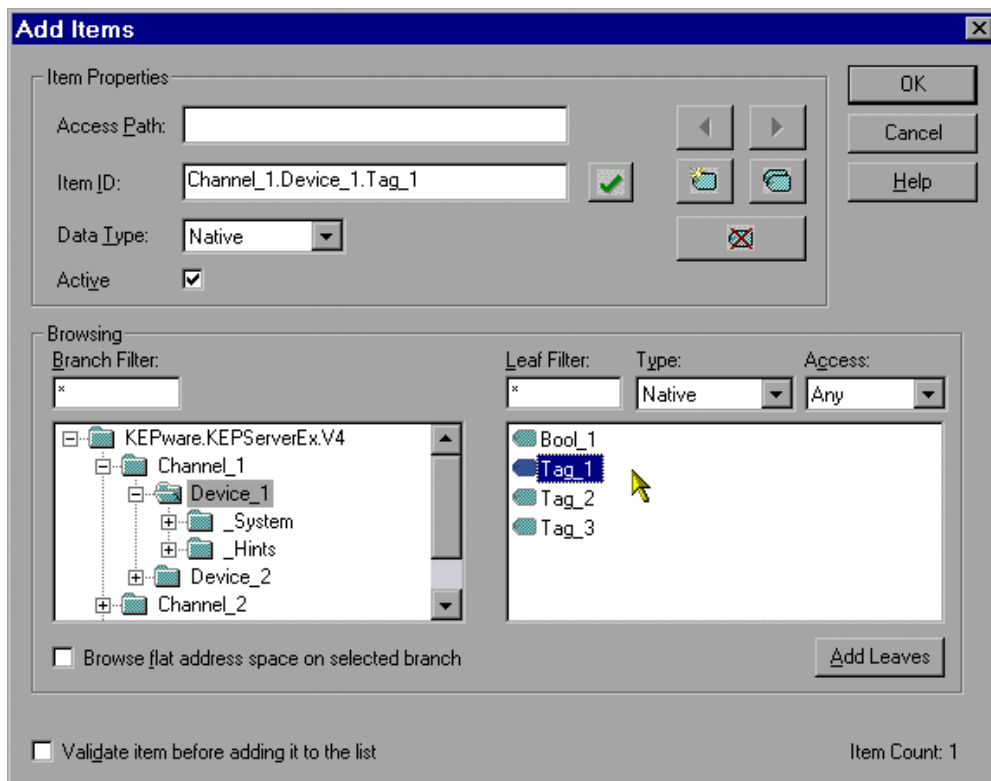
Now that a group has been created it is possible to add items.

5. Right click over the Group Name in the left display or select New Item from the Edit menu to open the Add Items window.



To learn more about using Kepware's OPC QuickClient with KEPServerEX, see the KEPServerEX help file "Designing a Project (Saving and Testing the Project)".

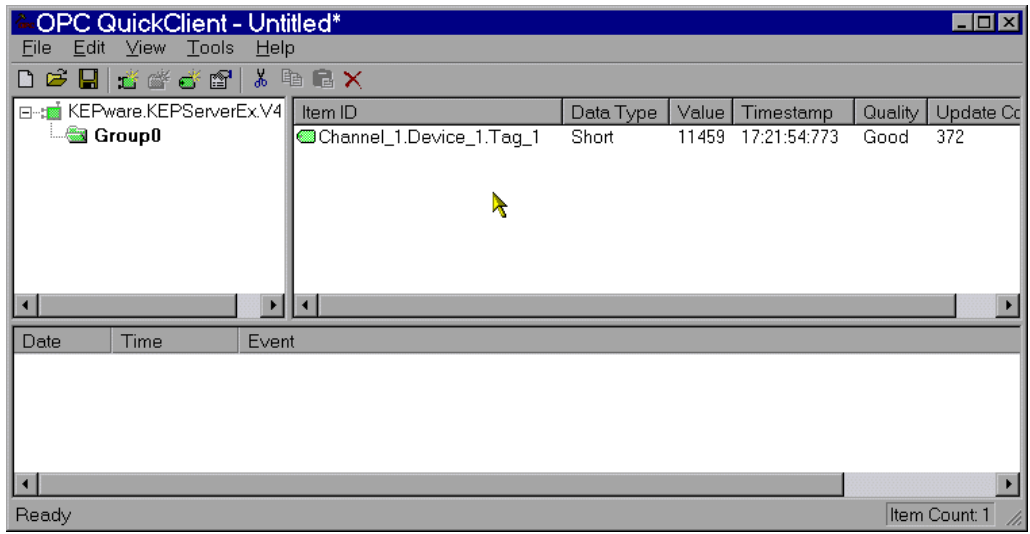
- Using the tree view in the left display of the OPC Address Browser, select the Device or Group from which you wish to select Address items/tags. In the right hand pane, double click to add an item. In this example, Tag\_1 in Device\_1 on Channel\_1 has been selected. Remember that these are items that have already been created in our Server Demo Project (Simdemo.opf). Also, you can view all possible address types and create a dynamic item by choosing "\_Hints" under the device name in the tree menu.



- Once you have selected all of the items to be displayed by the client, click OK

## Check OPC Connection

- Next we are going to verify that we are getting data from KEPServerEX. You should see a Quality of "Good" in the Tag Monitor window for the tag and a Value that is ramping or incrementing very quickly.



9. You now have a good OPC connection to the Server from the OPC Quick Client.