



# Manual

***IBH OPC Server***

***V4.x***

Information in this document is subject to change without notice and does not represent a commitment on the part of IBH softec GmbH. The software and/or databases described in this document are furnished under a license agreement or nondisclosure agreement. The software and/or databases may be used or copied only in accordance with the terms of the agreement. It is against the law to copy the software on any medium except as specifically allowed in the license or nondisclosure agreement. The purchaser may make one copy of the software for backup purposes. No part of this manual and/or databases may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording, or information storage and retrieval systems, for any purpose other than the purchaser's personal use, without the express written permission of IBH softec GmbH.

© Copyright 1993-2007 IBH softec GmbH. All rights reserved.



In case of questions, please don't hesitate to contact our product support..

© <b>IBH softec GmbH</b>	<b>Tel. : +49 6068 3001</b>
<b>Turmstraße 77</b>	<b>Fax. : +49 6068 3074</b>
<b>D-64743 Beerfelden</b>	<b>www : <a href="http://www.ibhsoftec.de">http://www.ibhsoftec.de</a></b>
<b>e-mail:</b> <b><a href="mailto:support@ibhsoftec-sps.de">support@ibhsoftec-sps.de</a></b>	<b>www : <a href="http://www.ibhsoftec-sps.de">http://www.ibhsoftec-sps.de</a></b>

Simatic, S7-200, S7-300, S7-400, WinCC, are trademarks of Siemens AG.

Windows, Visual Basic, Visual C und ActiveX are trademarks of Microsoft Corporation.

<b>Contents:</b>	<b>Page:</b>
1 IBHsofttec GmbH OPC Server V4.x	4
2 Installation and Licensing	5
3 Creating a new project with the OPCEditor	7
4 Variable Syntax	8
5 Optimization of the IBH OPC Server V4.x	11
6 DCOM configuration	12
7 Operation with User rights	13
8 ProTool Pro V6 with the IBHsofttec OPC Server V4.x	15
9 Win CC with the IBHsofttec OPC Server V4.x	19
10 WinCC Flexible with the IBHsofttec OPC Server V4.x	23
11 Intouch® with the IBHsofttec OPC Server V4.x HOWTO create variables manually	25
12 Intouch® with the IBHsofttec OPC Server V4.x HOWTO create variables with the TagCreator	29

## 1 IBHsofttec GmbH OPC Server V4.x

The IBH OPC Server permits OPC client applications like SCADA systems, to connect to PLCs of the Siemens SIMATIC® S5 and the SIMATIC® S7 series. It is also possible to access data simultaneously from several PLC Controls. Even a mixed S5 and S7 operation is possible. All modern SCADA systems are usually designed to be an OPC client.

PLCs may be accessed via the following protocols:

- SIMATIC® S5:
  - AS511 serial / USB
  - IBHLink S5 TCP/IP
  - IBHsofttec SoftPLC internal und external
  - INAT TCP/IP
  - INAT H1
- SIMATIC® S7:
  - MPI Adaptor serial / USB
  - IBHLink S7 / IBHLink Plus (also S7-200)
  - S7 TCP/IP to a CPx43
  - SimaticNet® (Set PG/PC Interface)
  - IBHsofttec SoftPLC internal and external
  - INAT H1

The configuration of the Server is performed via the OPCEditor. A shortcut is installed on your PC.

One of the special features of the **IBH OPC Server** is to allow direct use of the absolute PLC Variables without declaration. It is also possible to use symbolic addressing as defined in the PLC program (Symbolic Table). Even variables used in S7 Data Blocks can be accessed via the **IBH OPC Server** with their symbolic names. These features eliminate the need of double declarations. To do so, the PLC program is assigned with the **IBH OPCEditor** to the PLCName and the required variables are selected within the user interface.

For symbolic addresses, PLC programs in following formats may be used.

- SIMATIC® S5 (.S5D)
- SIMATIC® S7 (.S7P)
- S5/S7 for Windows® project (.S5P)
- S5/S7 for Windows® program file (.S5)
- Symbolic table (.SEQ)

## 2 Installation and Licensing

The **IBH OPC Server** starting with version 4.x can only be used with the 32 Bit Operating Systems Windows 2000, Windows XP and Windows 2003 Server from Microsoft™. Please make sure that the Internet Explorer 5.5 (or a higher version) is installed to allow an errorless execution. Only version 5.5(or higher) has the required system files available.

To install the **IBH OPC Server** please run the program "IBHOPCSetup.exe" and follow the instructions on the screen.

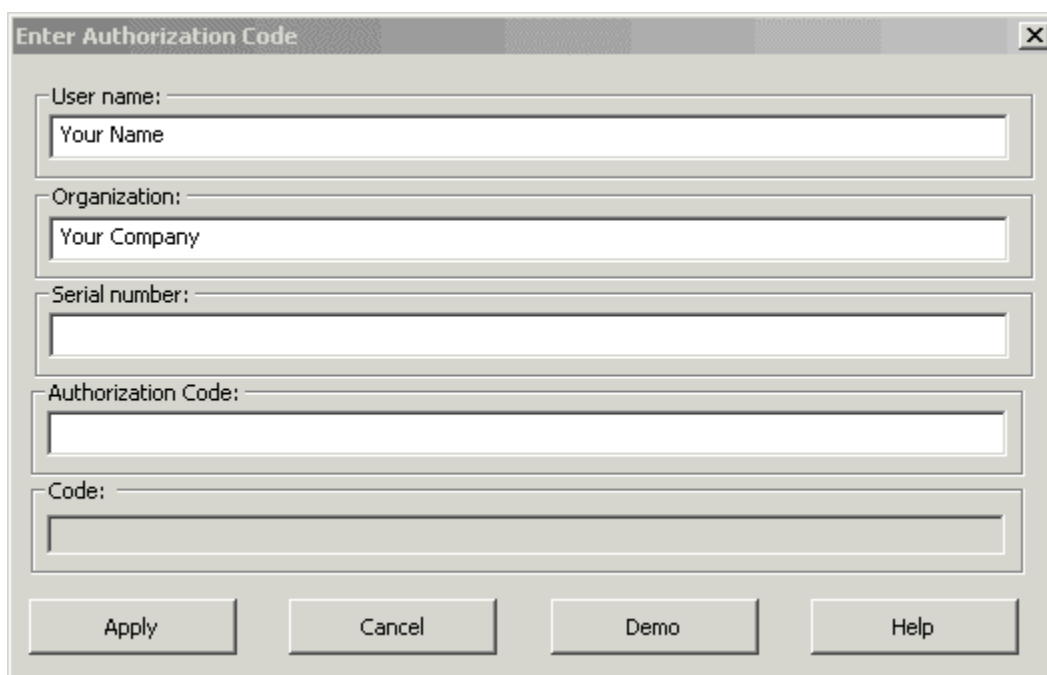
During installation and registration, entries into the Windows Registry are done automatically. To allow these registry entries you must have full administrative rights, when you are installing the **IBH OPC Server** on the operating systems Windows 2000, Windows XP and Windows 2003 Server.

After the installation is completed, the program **IBH OPC Editor** may be started.

If no license has been entered, the **IBH OPC Server** is running in demo mode. A complete configuration and also the use with real PLCs is possible in demo mode. After 1/2 hour the communication with the PLC will be terminated and the **IBH OPC Server** exits. When connecting to a PLC, a demo screen is shown to the user.

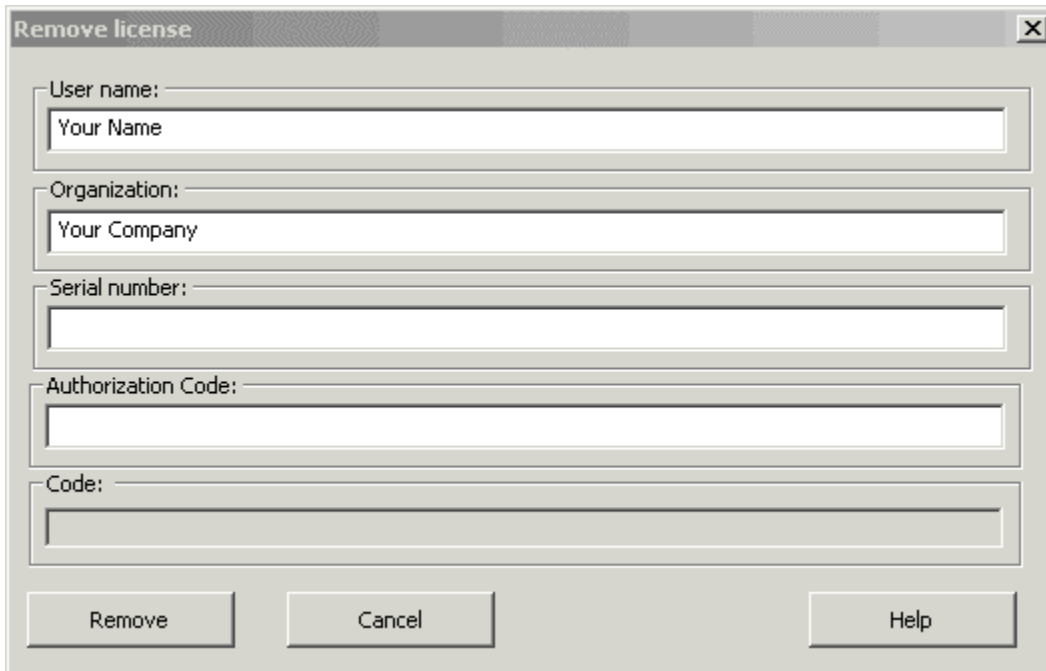
To enter a **License** for the **IBH OPC Server** a "Serial Number and "PIN Code" are required. These numbers can be found on the "Product ID Card" shipped with the IBH OPC Server.

Run the program **IBH OPC Editor**. Via the menu item "Help - License informationen / Enter license ..." the license for the **IBH OPC Server** may be entered. Please note, that the software needs to be restarted after the license has been successfully entered.



The screenshot shows a Windows-style dialog box titled "Enter Authorization Code". It features a close button (X) in the top right corner. The dialog contains five input fields, each preceded by a label: "User name:" (containing "Your Name"), "Organization:" (containing "Your Company"), "Serial number:" (empty), "Authorization Code:" (empty), and "Code:" (empty). At the bottom of the dialog, there are four buttons: "Apply", "Cancel", "Demo", and "Help".

Via the menu item "Help - Remove license ..." the license for the **IBH OPC Server** can also be removed from the PC. Please note, that a license can only be installed once on a specific PC. Installing the same license again on the same PC is not possible. After successful removal of the license, the software needs to be restarted.



The image shows a Windows-style dialog box titled "Remove license". It contains five text input fields stacked vertically, each with a label to its left: "User name:" (containing "Your Name"), "Organization:" (containing "Your Company"), "Serial number:", "Authorization Code:", and "Code:". At the bottom of the dialog, there are three buttons: "Remove", "Cancel", and "Help". The dialog has a standard Windows XP-style border with a close button (X) in the top right corner.

To uninstall the **IBH OPC Server** use the Windows control panel "Software", "Install / Uninstall".

### 3 Creating a new project with the OPCEditor

The configuration of the Server is performed via the OPCEditor. A shortcut is installed on your PC.

#### **HOWTO create a new configuration for the IBH OPC Server for absolute addresses:**

- First create a new configuration via "File - New".
- Insert a new PLC with "Modify - Insert new PLC".
- In the dialog box "PLC Properties" the communication protocol for the PLC may be selected and a name for the PLC may be entered.
- Now the previously chosen protocol needs to be highlighted in the tree view.
- Via "Modify - Connection settings" the communication parameters for the selected protocol can be chosen.
- Please save your changes before transferring the configuration to the server.
- Now the configuration may be transferred to the server via "Modify - Transfer to OPC server".
- In the following dialog the server "IBHSofttec.IBHOPC.DA.1" needs to be chosen.
- Save your configuration.

#### **HOWTO create a new configuration for the IBH OPC Server for symbolic addresses:**

- First create a new configuration via "File - New".
- Insert a new PLC with "Modify - Insert new PLC".
- In the dialog box "PLC Properties" the communication protocol for the PLC may be selected and a name for the PLC may be entered.
- Now the previously chosen protocol needs to be highlighted in the tree view.
- Via "Modify - Connection settings" the communication parameters for the selected protocol can be chosen.
- Highlight the desired PLC and assign the PLC program that contains the symbolic information via "Modify - Assign PLC program".
- Now a window may be opened that permits to pick variables for the server via "Modify - Select variables".
- In the "Variables" view now the variables that may be used with the OPC Server can be selected. Write-protecting variables and changing the default datatype is also possible.
- Please save your changes before transferring the configuration to the server.
- Now the configuration may be transferred to the server via "Modify - Transfer to OPC server".
- In the following dialog the server "IBHSofttec.IBHOPC.DA.1" needs to be chosen.
- Save your configuration.

Inside a project, multiple PLCs with equal and different protocols are possible. Also S5 and S7 PLCs may be mixed.

All commands of the "Modify" menu are also available by right-clicking of the mouse or using the toolbar. Only commands valid for the current node can be selected. All other commands are grayed.

## 4 Variable Syntax

One of the special features of the **IBH OPC Server** is to allow direct use of the absolute PLC Variables without declaration. It is also possible to use symbolic addressing as defined in the PLC program (Symbolic Table). Even variables used in S7 Data Blocks can be accessed via the **IBH OPC Server** with their symbolic names. These features eliminate the need of double declarations.

To do so, the PLC program is assigned with the **IBH OPCEditor** to the PLCName and the required variables are selected within the user interface.

Addressing of variables with absolute addresses in the **IBH OPC Server** is done by using the Simatic® Step®7 or Simatic® Step®5 syntax. Please note that REAL values (Double Word Format ) in S7 must be declared symbolic in order to have the correct format information. If this is not the case, the SCADA application must do the conversion of the "Long variable in the REAL format" into the correct "REAL format".

The IBH OPC Server supports 3 notations for variables.

- Symbolic from the PLC project (Variables are selected via the **IBH OPCEditor**)
- SIMATIC® S7 notation
- SIMATIC® S5 notation

When using absolute addresses from the OPC client, also a mixed S5/S7 notation is possible.

The syntax is always PLCName.VariableName respectively PLCName.DataBlockName.StructureElement.VariableName. A Variable must always begin with the PLCName. If the variables have been selected individually via the **IBH OPCEditor**, the variables can be conveniently browsed from the OPC client.

The configuration of the PLC itself must always be done via the **IBH OPCEditor**, regardless whether absolute or symbolic addressing is to be used. A shortcut to the **IBH OPCEditor** is installed on your PC.

### Common Syntax for S5 and S7:

Operand	Bereich	Datentyp
En.n	Inputbits **	Bool (VT_BOOL)
EBn	Inputbytes *	Unsigned Byte (VT_UI1)
EWn	Inputwords *	Unsigned Word (VT_UI2)
EDn	Inputdoublewords *	Unsigned DWord (VT_UI4)
ETn[Stringlength]	Text from the input area ***	Character Array (VT_BSTR)
An.n	Outputbits **	Bool (VT_BOOL)
ABn	Outputbytes *	Unsigned Byte (VT_UI1)
AWn	Outputwords *	Unsigned Word (VT_UI2)
ADn	Outputdoublewords *	Unsigned DWord (VT_UI4)
ATn[Stringlength]	Text from the output area ***	Character Array (VT_BSTR)
Mn.n	Flagbits **	Bool (VT_BOOL)
MBn	Flagbytes *	Unsigned Byte (VT_UI1)
MWn	Flagwords *	Unsigned Word (VT_UI2)
MDn	Flagdoublewords *	Unsigned DWord (VT_UI4)
MTn[Stringlength]	Text from the flag area ***	Character Array (VT_BSTR)



**Syntax for S5:**

<b>Operand</b>	<b>Bereich</b>	<b>Datentyp</b>
Dn.n,nr	Databits, DB	Bool (VT_BOOL)
DRn,nr	Right Databyte, DB	Unsigned Byte (VT_UI1)
DLn,nr	Left Databyte, DB	Unsigned Byte (VT_UI1)
DWn,nr	Dataword, DB *	Unsigned Word (VT_UI2)
DSn,nr	Dataword, DB *	Signed short Integer (VT_I2)
DDn,nr	Datadoubleword, DB *	Unsigned DWord (VT_UI4)
DVn,nr	Datadoubleword, DB *	Signed long integer (VT_I4)
DFn,nr	Datadoubleword, DB *	Float (VT_R4)
DTn,nr[Stringlength]	Text from a datablock ***	Character Array (VT_BSTR)
DXn.n,nr	Databits, DX	Bool (VT_BOOL)
DXRn,nr	Right Databyte, DX	Unsigned Byte (VT_UI1)
DXLn,nr	Left Databyte, DX	Unsigned Byte (VT_UI1)
DXWn,nr	Dataword, DX *	Unsigned Word (VT_UI2)
DXSn,nr	Dataword, DX *	Signed short Integer (VT_I2)
DXDn,nr	Datadoubleword, DX *	Unsigned DWord (VT_UI4)
DXVn,nr	Datadoubleword, DX *	Signed long integer (VT_I4)
DXFn,nr	Datadoubleword, DX *	Float (VT_R4)
DXTn,nr[Stringlength]	Text from a DX datablock ***	Character Array (VT_BSTR)
Sn.n	S-Flagbits **	Bool (VT_BOOL)
SYn	S-Flagbytes *	Unsigned Byte (VT_UI1)
SWn	S-Flagwords *	Unsigned Word (VT_UI2)
SSn	S-Flagwords *	Signed short Integer (VT_I2)
SDn	S-Flagdoublewords *	Unsigned DWord (VT_UI4)
SVn	S-Flagdoublewords *	Signed long integer (VT_I4)
SFn	S-Flagdoublewords *	Float (VT_R4)
STn[Stringlength]	Text from the extended flag area ***	Character Array (VT_BSTR)

**Syntax for S7:**

Operand	Bereich	Datentyp
DBn.DBXn.n	DB, Databits **	Bool (VT_BOOL)
DBn.DBBn	DB, Databyte *	Unsigned Byte (VT_UI1)
DBn.DBCn	DB, Databyte *	Signed Byte (VT_I1)
DBn.DBWn	DB, Dataword *	Unsigned Word (VT_UI2)
DBn.DBSn	DB, Dataword *	Signed Word (VT_I2)
DBn.DBDn	DB, Datadoubleword *	Unsigned DWord (VT_UI4)
DBn.DBVn	DB, Datadoubleword *	Signed DWord (VT_I4)
DBn.DBRn	DB, Real *	Real (VT_R4)
DBn.DBTn[Stringlength]	Text from a datablock ***	Character Array (VT_BSTR)

\* For these operands the suffix [Count] may be added to read/write arrays. Example: PLCName.DB10.DBB2[5] reads 5 bytes starting from DBB2. PLCName.MW4[7] reads 7 Words starting from the MW4.

\*\* For bit operands also the suffix [Count] may be added to read/write arrays. Arrays of boolean must always start with Bit 0 within the byte. Example: PLCName.DB10.DBX2.0[5] reads 5 Bits starting from DBX2.0. PLCName.M4.0[7] reads 7 Bits starting from M4.0. A variable like PLCName.M4.3[4] is not permitted. When writing boolean arrays, always the entire byte is written.

\*\*\* If strings are entered in absolute notation, there is no verification whether it is a declared string. In S7 PLCs, strings have 2 leading bytes with additional information regarding the string, the declared length and the current length. Since this information lacks in absolute notation, strings are treated as pure character arrays of the length entered in [Stringlength]. If in a write operation the text is shorter as the length entered in [Stringlength], the rest is filled up with zeroes.

The 'signed' datatypes also exist for the S7. Normally, we assume that the type will be set via the configuration tool **IBH OPCEditor**.

**Addressing the DB of a S7-200:**

The syntax for the 200 equals the S7-300 syntax. To access a variable byte of the data block in S7-300 syntax you need to type: VB 2 (S7-200) = DB1.DBB2 (S7-300). The data block is always the DB1. A variable word VW 2 (S7-200) = DB1.DBW2 (S7-300). A variable doubleword VD 2 (S7-200) = DB1.DBD2 (S7-300).

## 5 Optimization of the IBH OPC Server V4.x

The IBH OPC server by default tries to optimize variable read/write operations with the PLC. This means, that variables are automatically arranged into blocks. These blocks are then read at once from the PLC. This happens automatically and has no influence on the client application.

If the client adds the variables Flag 2.0 and Flag 23.5 to the OPC server, the server starts reading the Flagbytes 2 to 23 in one block from the PLC, since this is faster than reading every variable separately from the PLC. The reason for this behaviour is how the PLCs communication protocols work.

The optimization into the blocks is organized by operand areas, that are read in separate requests from the PLC. Each reading from areas like inputs, outputs, flags and every datablock will be organized to a block request. Every block sends it's own request to the PLC.

**Example:** The following variables I2.0, I23.5, Q2.0, Q23.5, M2.0, M23.5, DB10.DBX2.0, DB10.DBX23.5, DB20.DBX2.0, DB20.DBX23.5 are to be read from the PLC.

The IBH OPC server creates 5 requests for the PLC, each request is read in a block operation. In the sample the areas IB2 - IB23, QB2 - QB23, MB2 - MB23, DB10.DBB2 - DB10.DBB23, DB20.DBB2 - DB20.DBB23 are read, because this is faster than reading the single bits.

### The optimization can be taken to absurdity!

**Example:** Only the variables I2.0, I1023.5, Q2.0, Q1023.5, M2.0, M1023.5, DB10.DBX2.0, DB10.DBX1023.5, DB20.DBX2.0, DB20.DBX1023.5 are to be read.

The IBH OPC server also creates 5 requests to the PLC. Each request, again, is read in block operations. In the sample this would be the areas (blocks) IB2 - IB1023, QB2 - QB1023, MB2 - MB1023, DB10.DBB2 - DB10.DBB1023, DB20.DBB2 - DB20.DBB1023. Now per block approximately 1 kByte unnecessary data will be read. For the client and the function itself, this does not have any influence, but it puts a heavy load on the communication line. This strongly reduces performance, since much more data than required is now read.

**In such a case** it makes more sense to "collect" the required variables via the PLCs program into one datablock, which is reduced to the minimum. In this case the IBH OPC server reads and writes only the variables of the "interface" DB. If spreading the written values from the "interface" DB to the PLCs operands is also performed by the PLC program, the internal optimization can work up to full optimization and the maximum performance is reached.

The **throughput** with a **S7-IBHLink** is approximately 100 ms for a variable block of 150 bytes. For a ethernet module CPx43 the throughput is higher. With serial adaptors like a PC adaptor or with a S5, that work at a speed of 9600 baud, the throughput is lower. As we can see, the communication interface in use also has a significant influence on the performance of the server. These consideration should be taken into account when selecting and arranging the variables.

## 6 DCOM configuration

Follow the steps listed in order to enable communication via DCOM.

In case the settings do not match your security requirements, alter the suggested settings accordingly.

### DCOMCNFG

DCOM uses security settings in order to protect Clients and Servers from unauthorized access. The security settings may be altered within the Windows application DCOMCNFG. In order to use the program, administrative rights are required.

For Windows 9x systems, DCOMCNFG is not part of the default installation. It needs to be explicitly added.

Select "Run" and type "DCOMCNFG" in order to run DCOMCNFG. A dialog application appears, which permits to do the security settings.

In the table below the settings are listed, that in general work with OPC Servers and OPC Clients.

Parameter	Setting
Default properties - Authentification Level	None
Default properties - Impersonation Level	Impersonate
Default properties - Launch and Activation permissions	Everyone, System, Administrator und Interactive User
Default properties - Access permissions	Everyone, System, Administrator und Interactive User

The altered settings are applied to DCOM applications after restarting the application.

In case the two PCs are not part of the same domain, the same local user should be added to both systems. The password of this OPC User needs to be identical on both PCs.

### DCOM and Windows XP

When starting DCOMCNFG in Windows XP, a user interface to configure the component services is started. The user may now get to the DCOM configuration dialog by picking "Component Services | Computers | My Computer" and then displays the properties.

The default installation of Windows XP authenticates users from remote computers as Guest. This means, that DCOM Clients can not connect to a server, as long as the Guest access is not activated and the Guest does not have enough permissions to access the server.

### The default behaviour may be changed using the control panel.

Administrative Tools | Local Security Policy | Local Policies | Security Options | Network access: Sharing and security model for local accounts. Set this to: "Classic - local users authenticate as themselves".

### DCOM und Windows XP SP2

Windows XP SP2 makes a difference in the DCOM configuration between local and remote connections. Please make sure to set the permissions also for the remote access.

For Windows XP SP2 are besides the Start and access permissions also settings for the Start and access restrictions. Per default there is a remote access restriction for the user "Everyone". This restriction needs to be removed, if access needs to be granted to someone.

### DCOM and Firewalls

DCOM can not be used via a Firewall. Disable the Firewall to permit a remote communication.

## 7 Operation with User rights

### Windows 2000:

- Log on as Administrator.
- Run the extended registry editor "Regedt32.exe".
- Navigate to the registry key "HKEY\_CLASSES\_ROOT\IBHSoftec.IBHOPC.DA".
- Via the menu item "Security - Permissions" the required permissions for this key and all subkeys need to be set.
- Navigate to the registry key "HKEY\_CLASSES\_ROOT\IBHSoftec.IBHOPC.DA.1".
- Via the menu item "Security - Permissions" the required permissions for this key and all subkeys need to be set.

During the installation of the **IBH OPC Server** the permission "Everyone - Read" is automatically added. This may be altered according to your security requirements.

To transfer a **configuration to the Server**, the user also needs the right to write files in the installation folder of the **IBH OPC Server**. The configuration is stored in the XML file IBHOPC.opx, which is identical to the project file, created in the **IBH OPC Editor**.

- Open the Windows Explorer and highlight the installation folder of the **IBH OPC Server**, typically "C:\Program files\IBH softec GmbH\IBH OPC Server".
- Via "File - Properties" register "Security settings" now "Everyone - Full access" may be added.

Alter the settings according to your security requirements. After these steps are completed, the server may be used with regular User rights.

### Windows XP:

- Log on as Administrator.
- Run the extended registry editor "Regedt32.exe".
- Navigate to the registry key "HKEY\_CLASSES\_ROOT\IBHSoftec.IBHOPC.DA".
- Via the menu item "Security - Permissions" the required permissions for this key and all subkeys need to be set.
- Navigate to the registry key "HKEY\_CLASSES\_ROOT\IBHSoftec.IBHOPC.DA.1".
- Via the menu item "Security - Permissions" the required permissions for this key and all subkeys need to be set.

During the installation of the **IBH OPC Server** the permission "Everyone - Read" is automatically added. This may be altered according to your security requirements.

To transfer a **configuration to the Server**, the user also needs the right to write files in the installation folder of the **IBH OPC Server**. The configuration is stored in the XML file IBHOPC.opx, which is identical to the project file, created in the **IBH OPC Editor**.

- Open the Windows Explorer and highlight the installation folder of the **IBH OPC Server**, typically "C:\Program files\IBH softec GmbH\IBH OPC Server".

- Via "File - Properties" register "Security" now "Everyone - Full access" may be added.

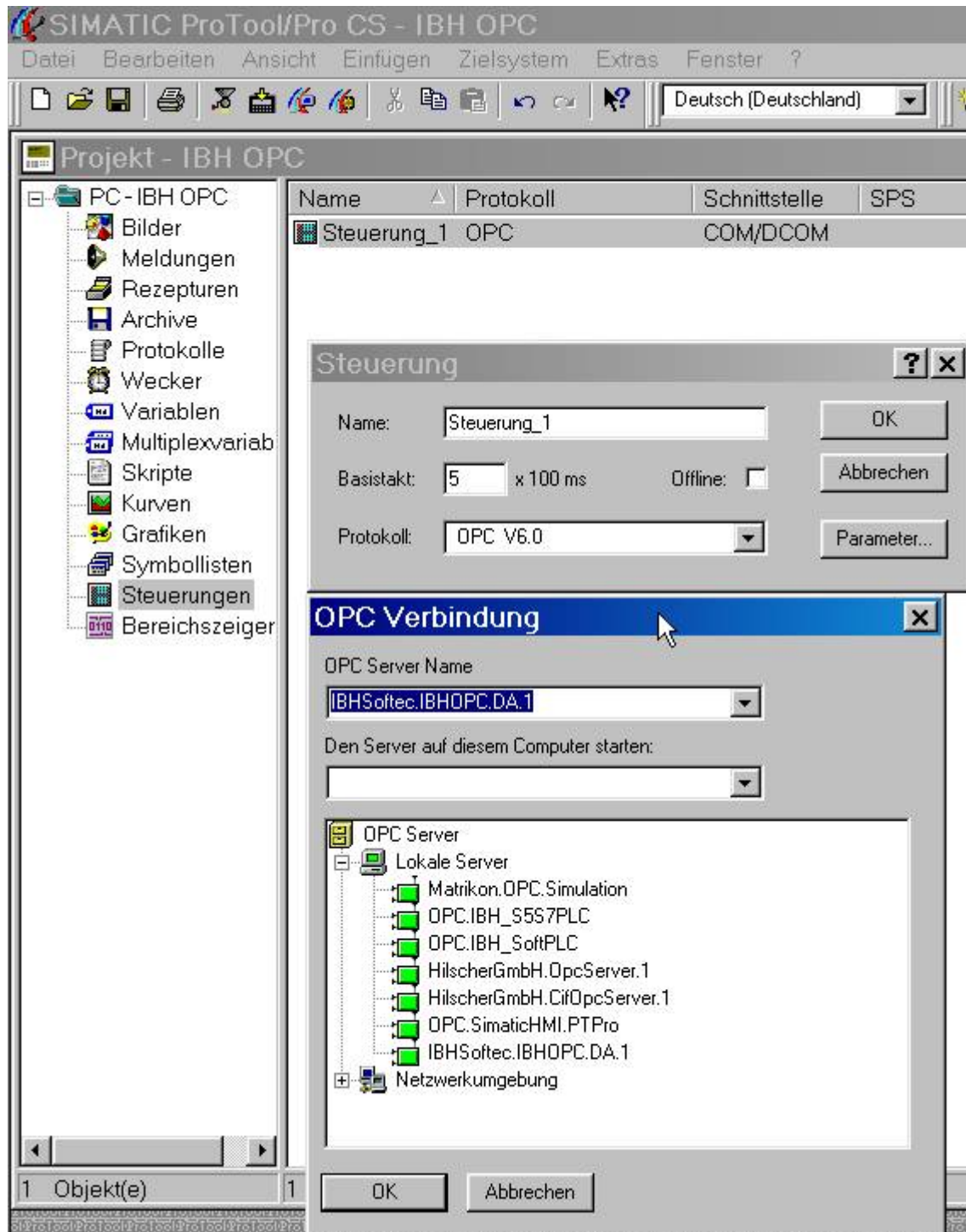
During the installation of the **IBH OPC Server** the permission "Everyone - Full access" is automatically added. This may be altered according to your security requirements. After these steps are completed, the server may be used with regular User rights.

## 8 ProTool Pro V6 with the IBHsofttec OPC Server V4.x

At **<PLC Control>** **<Protocol>** select **OPC V6.0**.

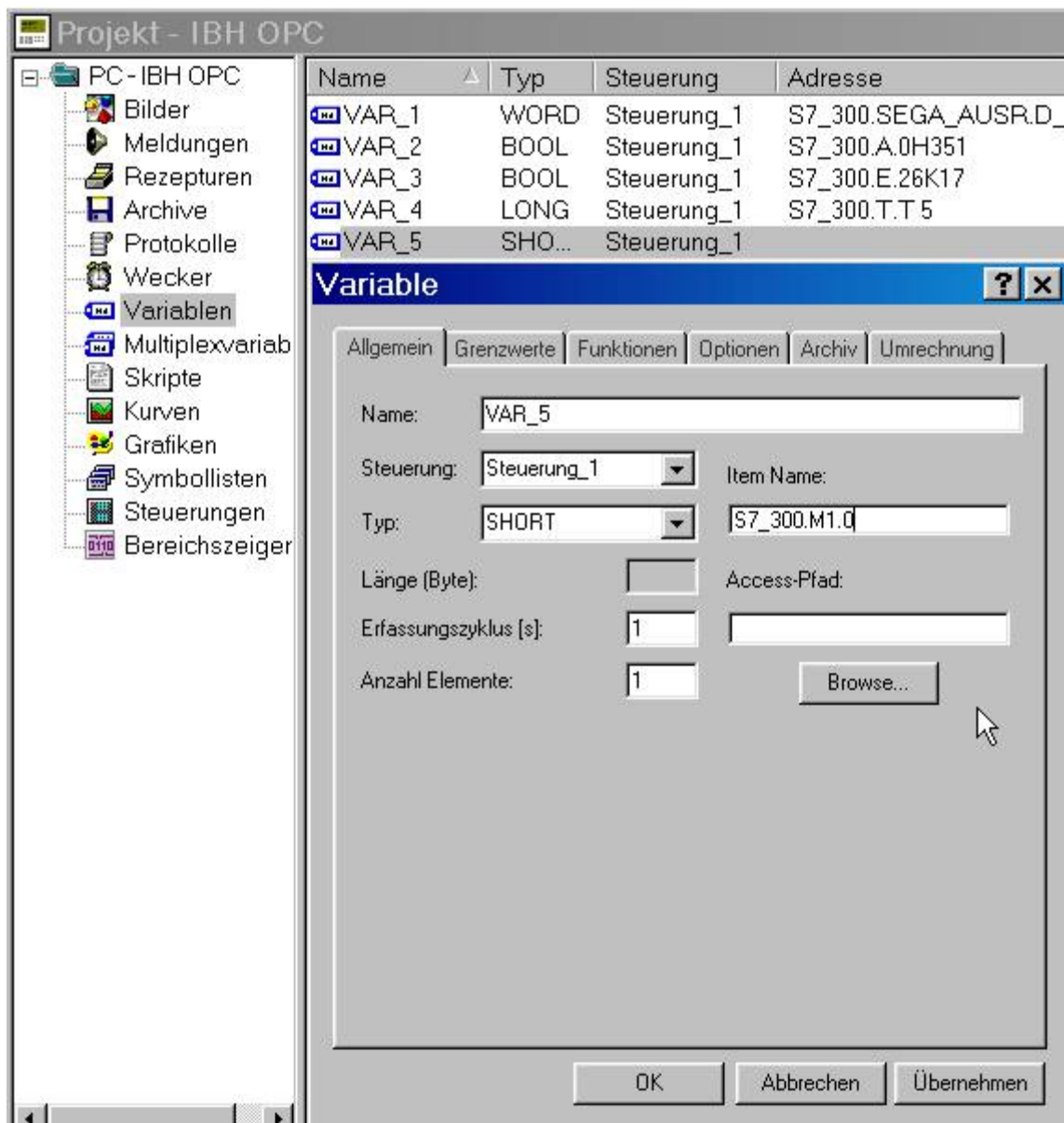
Via **<Parameters...>** open the dialog box **OPC connection**.

There, the server **IBHsofttec.IBHOPC.DA.1** may be selected.



## Create OPC variables manually:

In the menu **<Variables>** now the desired variables can be entered in the field **<Item Name >**:



The notation required for Protocol Pro consists of :

**<Access-Path>.<Item Name>**

In our sample : **S7\_300.M1.0**

The **<Access-Path>** corresponds to the name of the **<PLC>**, as it is defined with the **IBH OPC Editor** :

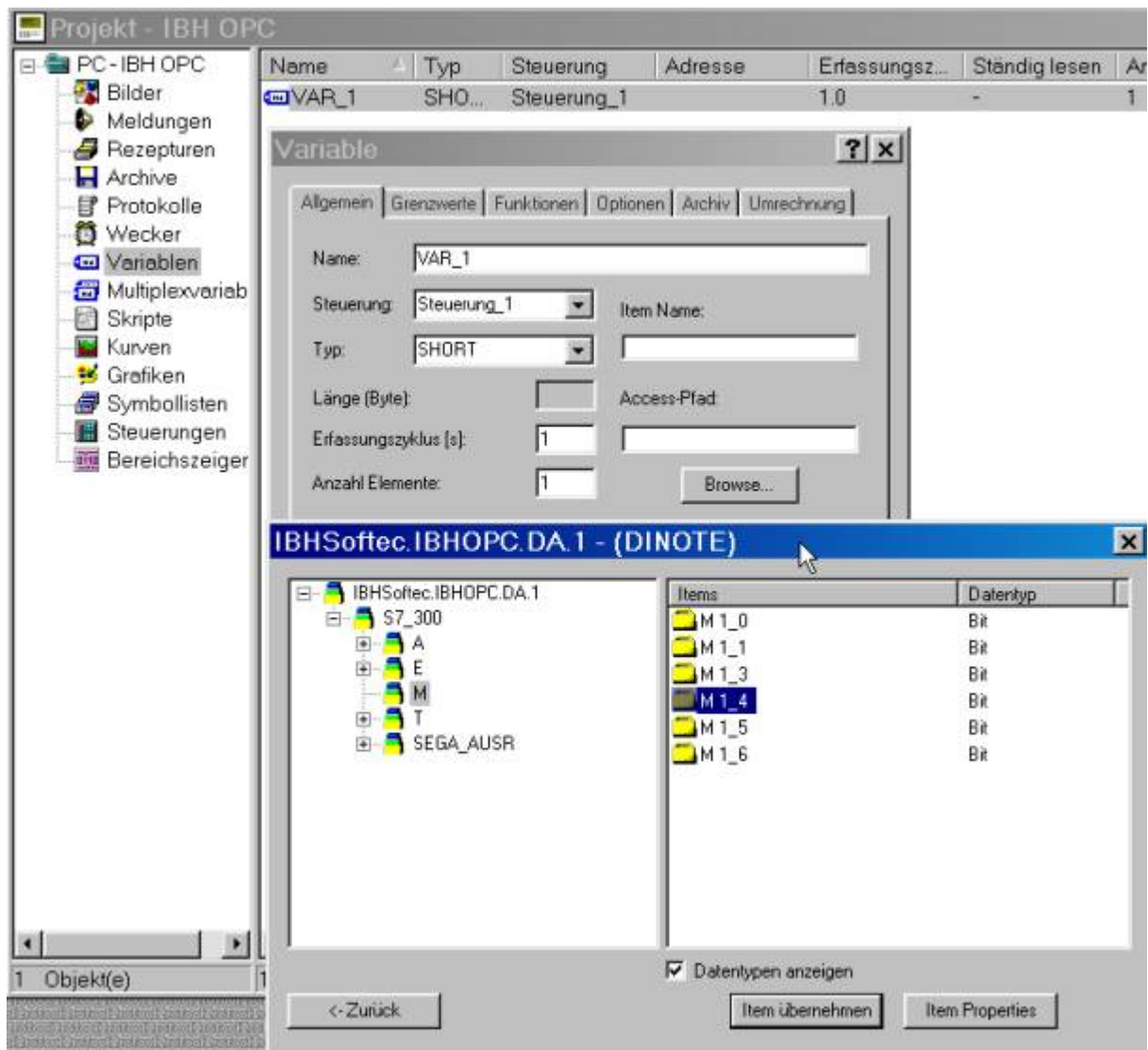


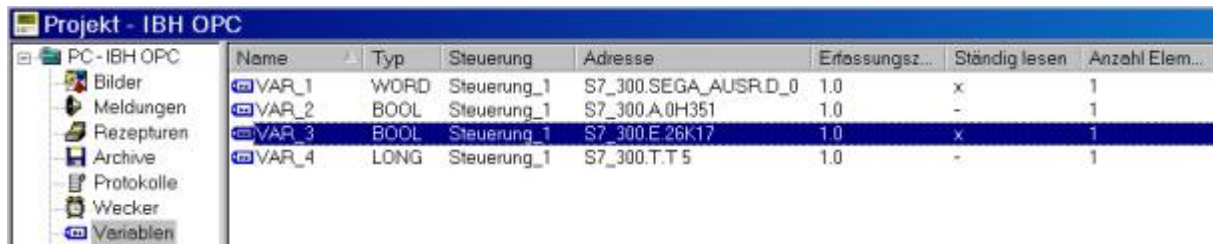


Please leave the entry-field <Access-Path> empty.

### Convenient creation of OPC Variables with the browsing function :

In the menu <Variables> browsing of the servers address space may be done by clicking the button <Browse> :

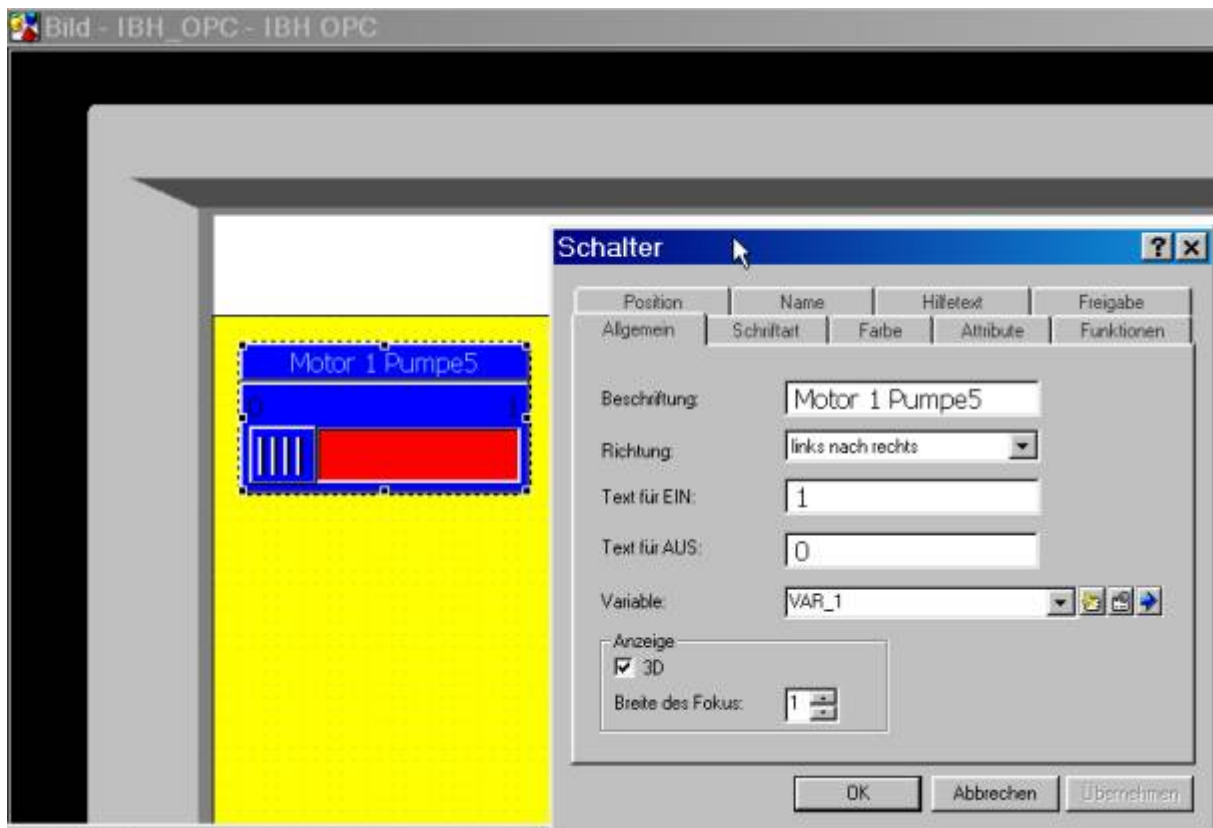




The screenshot shows the 'Projekt - IBH OPC' window. On the left is a tree view with folders: Bilder, Meldungen, Rezepturen, Archive, Protokolle, Wecker, and Variablen. The 'Variablen' folder is selected. On the right is a table listing the created variables.

Name	Typ	Steuerung	Adresse	Erfassungsz...	Ständig lesen	Anzahl Elem...
VAR_1	WORD	Steuerung_1	S7_300.SEGA_AUSR.D_0	1.0	x	1
VAR_2	BOOL	Steuerung_1	S7_300.A.0H351	1.0	-	1
VAR_3	BOOL	Steuerung_1	S7_300.E.26k17	1.0	x	1
VAR_4	LONG	Steuerung_1	S7_300.T.T 5	1.0	-	1

The created **OPC Variables** now may be used:



## 9 Win CC with the IBHsofttec OPC Server V4.x

Right-click <Variable management>

<Select "Add new driver...">



and select the OPC driver :

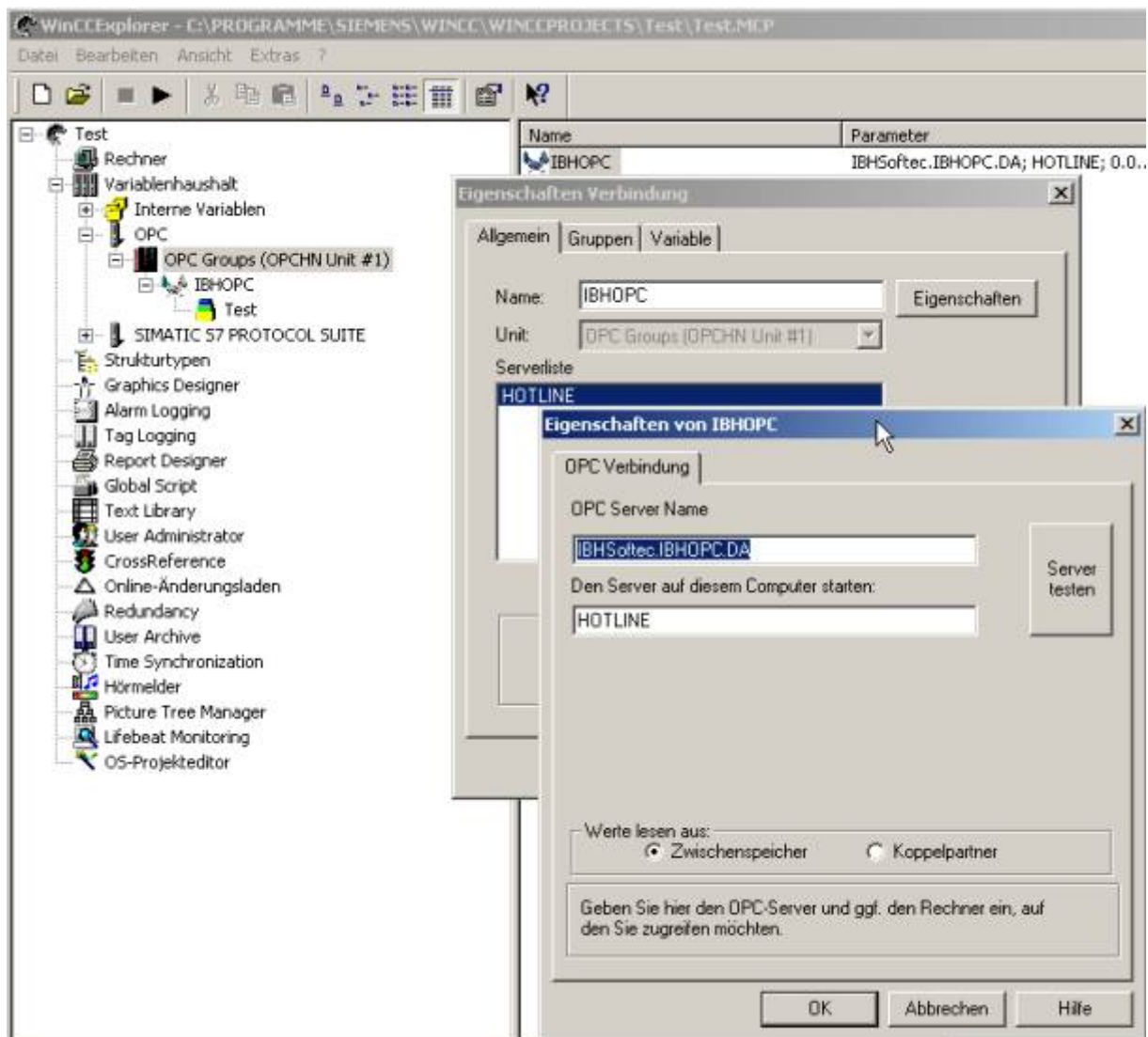


navigate to <OPC Groups> and create a new connection :

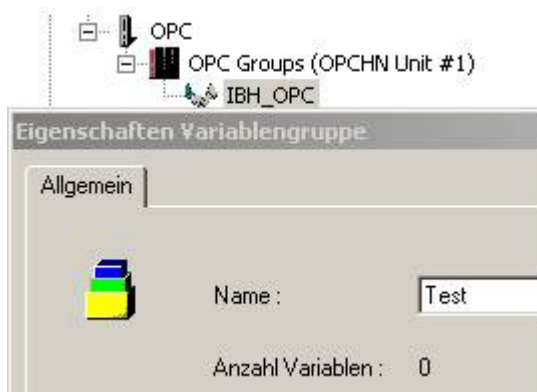


After you have given the connection a name

( in our sample : IBHOPC ) the OPC Server name : **IBHSofttec.IBHOPC.DA** and the host name can be entered in the properties:



Now add a new group.



Then add the new variables in the group :

The **<Item Name>** corresponds to the PLC operand.

*IBH OPC Server*

The entry for WinCC is a combination of PLCName and VariableName :

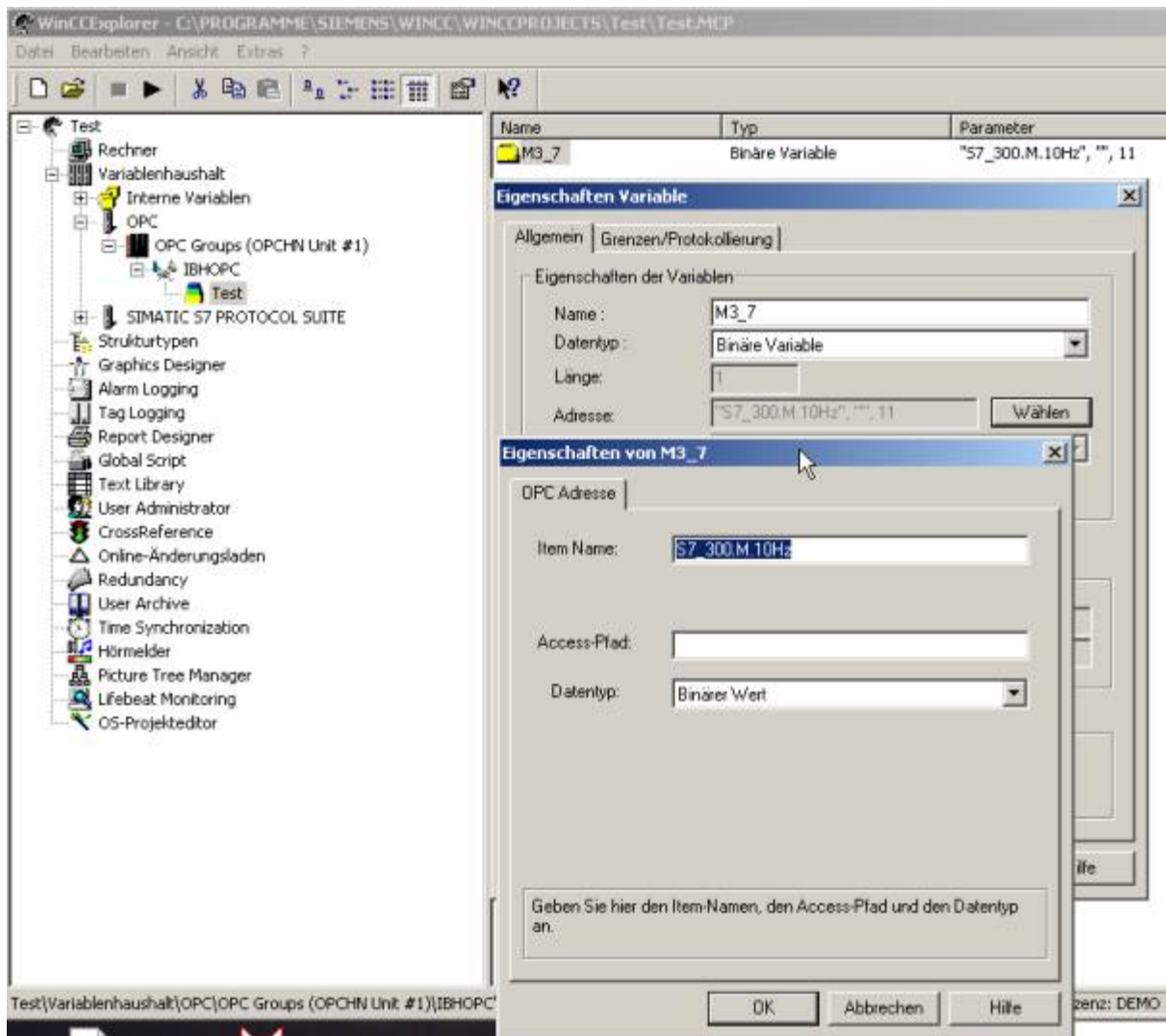
**<Access-Path>.<Item Name>**

In our sample : **S7\_300.M.10HZ**

The **<Access-Path>** corresponds to the name of the **<PLC>**, which was created in the **IBH OPC Editor**:

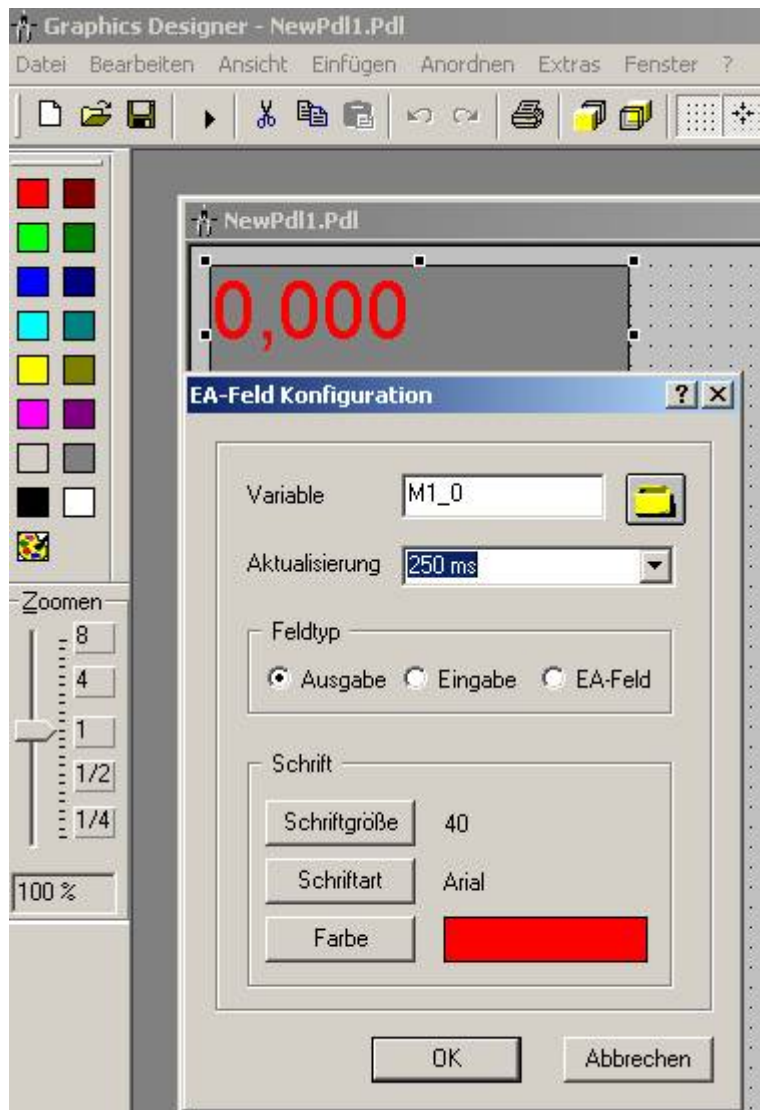


**Note:** Please leave the input box **<Access-Path>** empty.



Now the created variable may be used in the **<Graphics Designer>** :





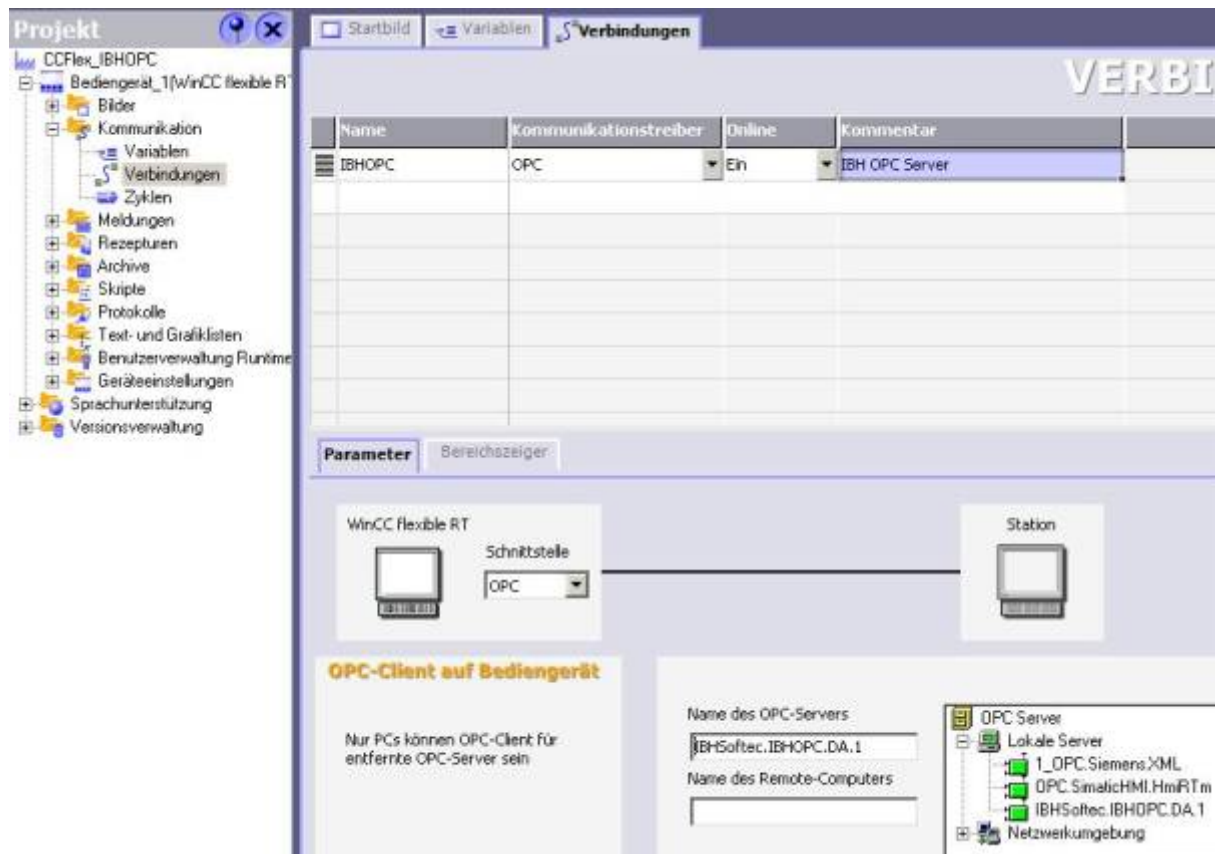
## 10 WinCC Flexible with the IBHsofttec OPC Server V4.x

When creating a project, first select in the column **<PLC>** the protocol **OPC**. Then at **<Connection>** also **OPC** may be chosen.



In the project tree-item **<Connections>** now the connection has to be given a name, in our sample IBHOPC.

Here also the server **IBHsofttec.IBHOPC.DA.1** needs to be chosen.



## Create OPC variables manually:

In the column **<Address>** now the desired variables can be entered:



The notation required for WinCC Flexible consists of :

**<Access-Path>.<Item Name>**

In our sample : **S7\_300.M.Anlauf**

The **<Access-Path>** corresponds to the name of the **<PLC>**, as it is defined with the **IBH OPCEditor** :

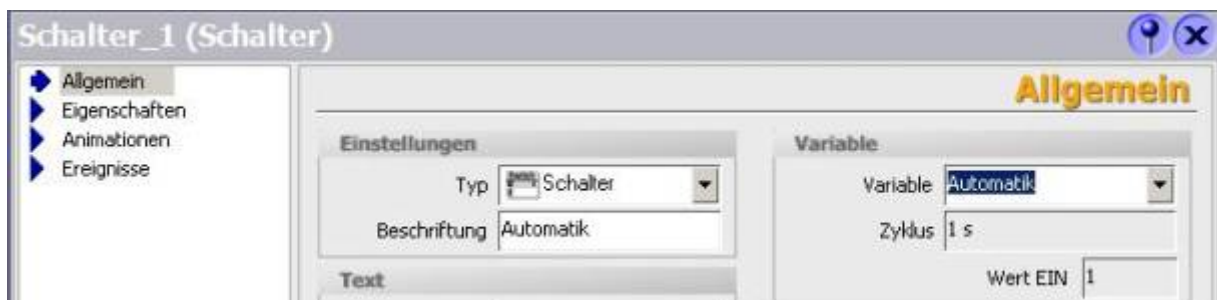


## Convenient creation of OPC Variables with the browsing function :

A click on the dropdown in the column **<Symbol>** opens a window, that permits browsing of the servers address space:



The created **OPC Variables** now may be used:



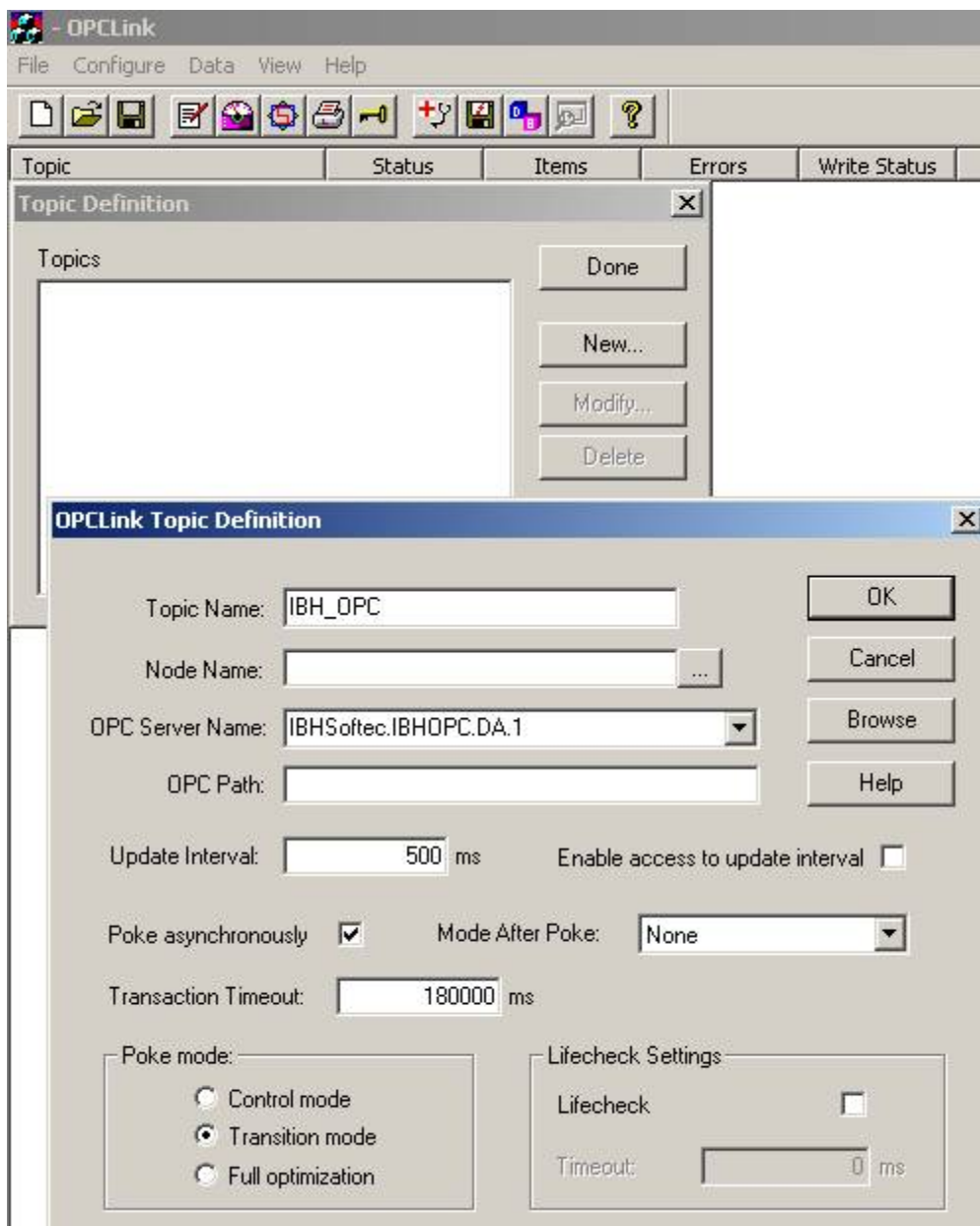
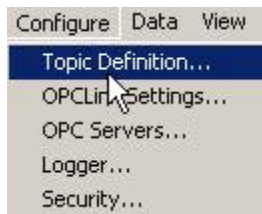


## 11 Intouch® with the IBHsofttec OPC Server V4.x

### HOWTO create variables manually

Please run the Intouch® programm **OPCLink**.

Select <Configure> <Topic Definitions>



Define a topic Name (i.e. IBH\_OPC ).

In the entry field **OPC Server Name** please select the server **IBHsoftec.IBHOPC.DA.1** .

The entry **OPC Path** must stay empty.

Please run the **Intouch WindowMaker**:

Create a new access name :

The screenshot shows the 'Zugriffsnamen' (Access Names) dialog box. The main window has a list box containing 'Galaxy'. To the right of the list are buttons 'Schließen' (Close) and 'Hinzufügen' (Add). A sub-dialog titled 'Zugriffsname hinzufügen' (Add Access Name) is open. It contains the following fields and options:

- Zugriffsname:** IBH
- Knotenname:** (empty)
- Anwendungsname:** OPCLink
- Topic-Name:** IBH\_OPC
- Protokoll:** ☒ DDE, ☐ SuiteLink, ☐ Message Exchange
- Serveranmeldung:** ☐ Alle Items anmelden, ☒ Nur aktive Items anmelden
- ☐ Sekundärquelle aktivieren

Buttons on the right of the sub-dialog: OK, Abbrechen (Cancel), and Failover.

Select the newly created access name.

The entry **<Item>** consists of :

**<D><Access-Path>.<Item Name>**

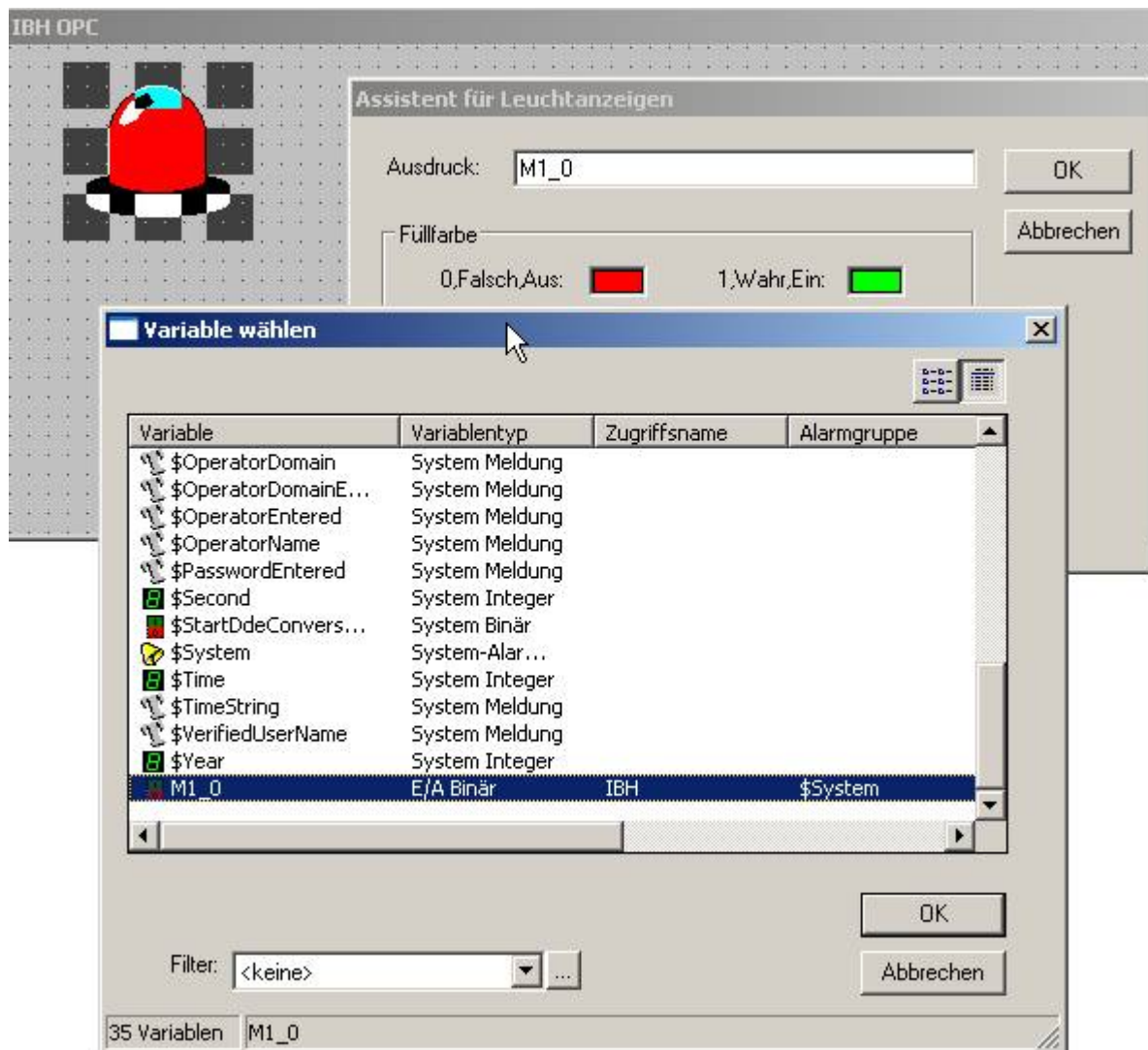
In our sample : **DS7\_300.M1.0**

Whereby the **<D>** is the notation for Discrete.

Following values are possible :

- d discrete value
- i integer value
- r real value
- m message value

The **<Access-Path>** represents the name of the **<PLC>**, which has been created in the **IBH OPC Editor**:



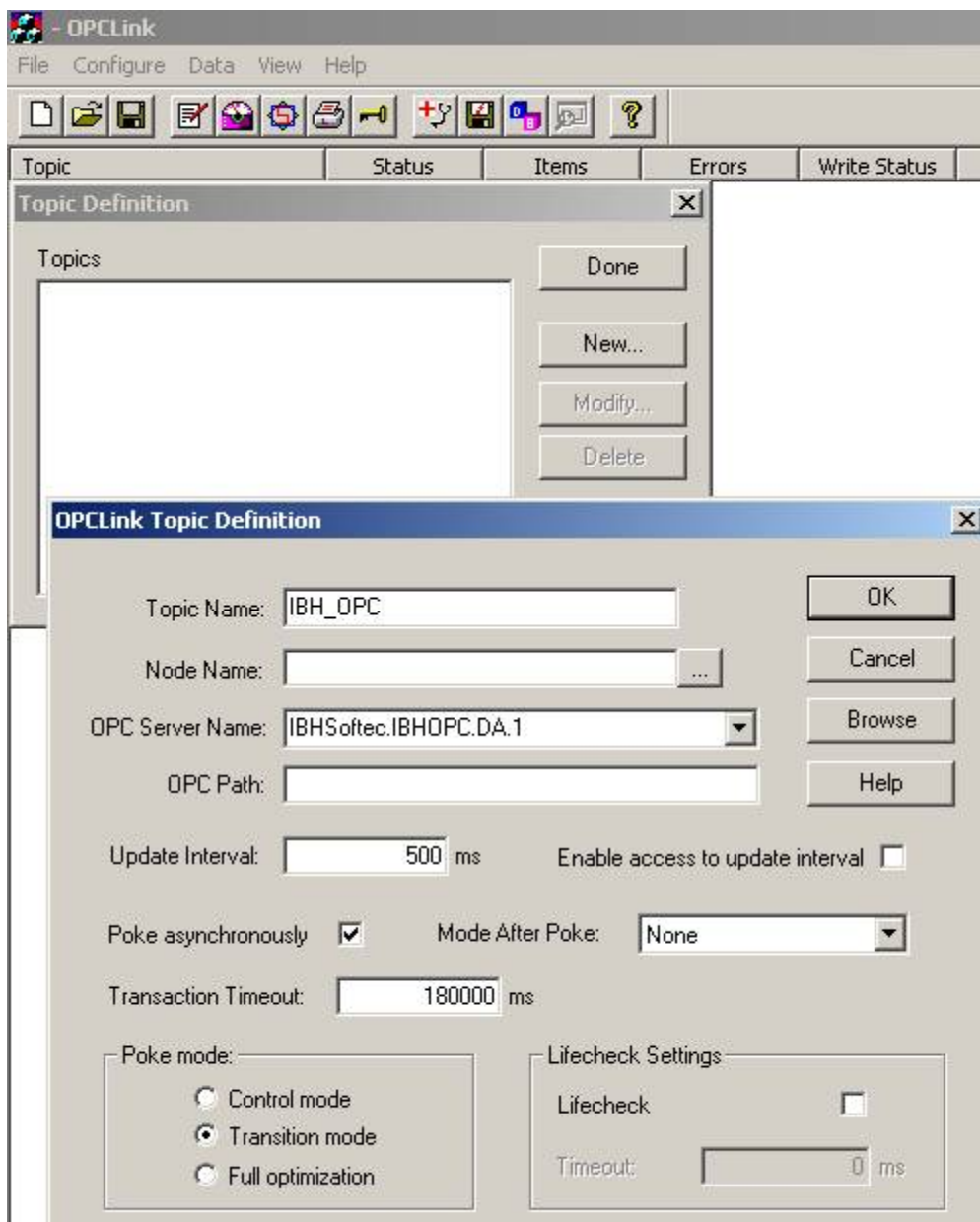
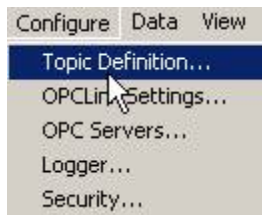
The OPC variables may now be used in the **WindowMaker**.

## 12 Intouch® with the IBHsofttec OPC Server V4.x

### HOWTO create variables with the TagCreator

Please run the Intouch® programm **OPCLink**.

Select <Configure> <Topic Definitions>



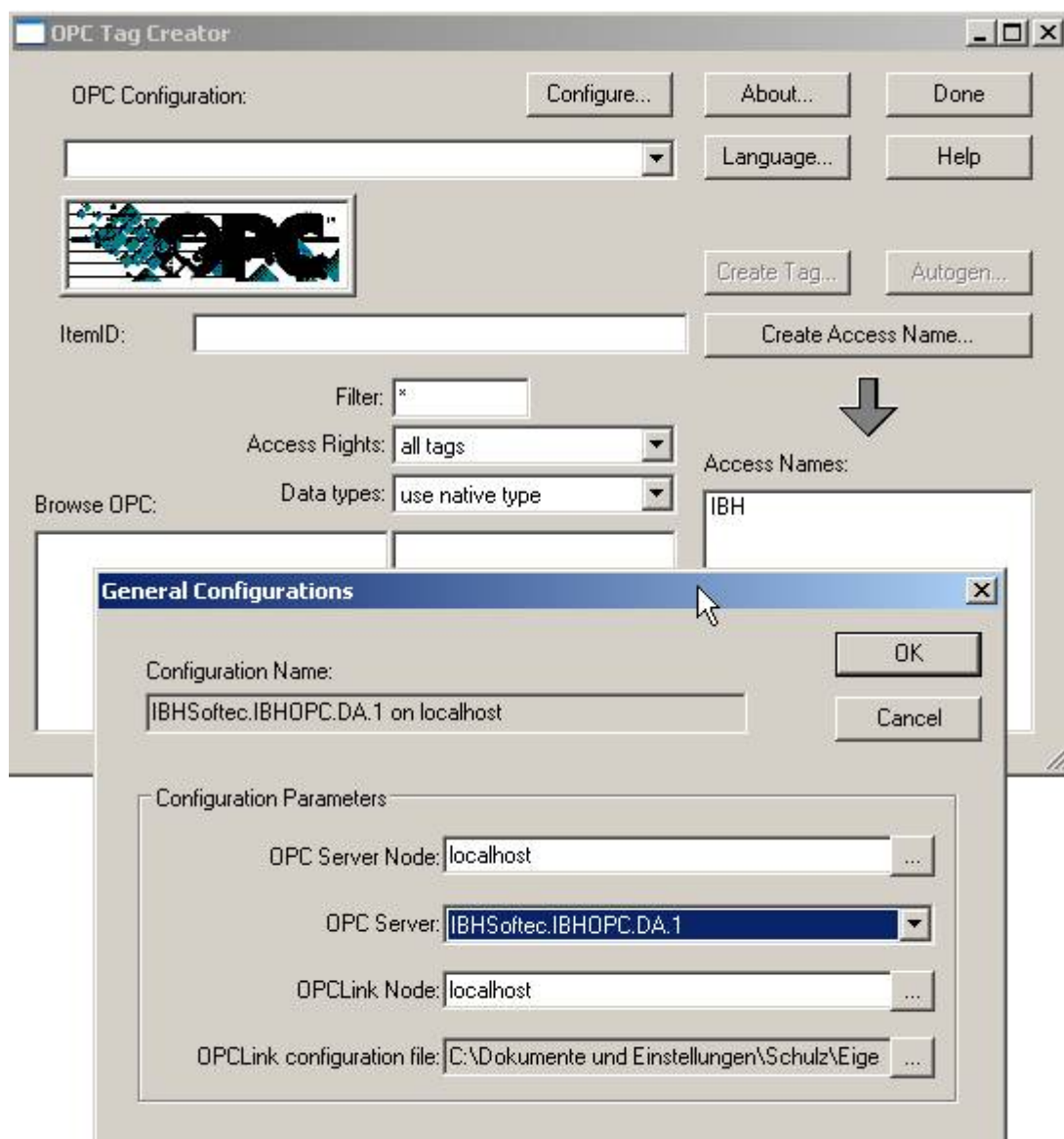
Define a topic Name (i.e. IBH\_OPC ).

In the entry field **OPC Server Name** please select the server **IBHsoftec.IBHOPC.DA.1** .

The entry **OPC Path** must stay empty.

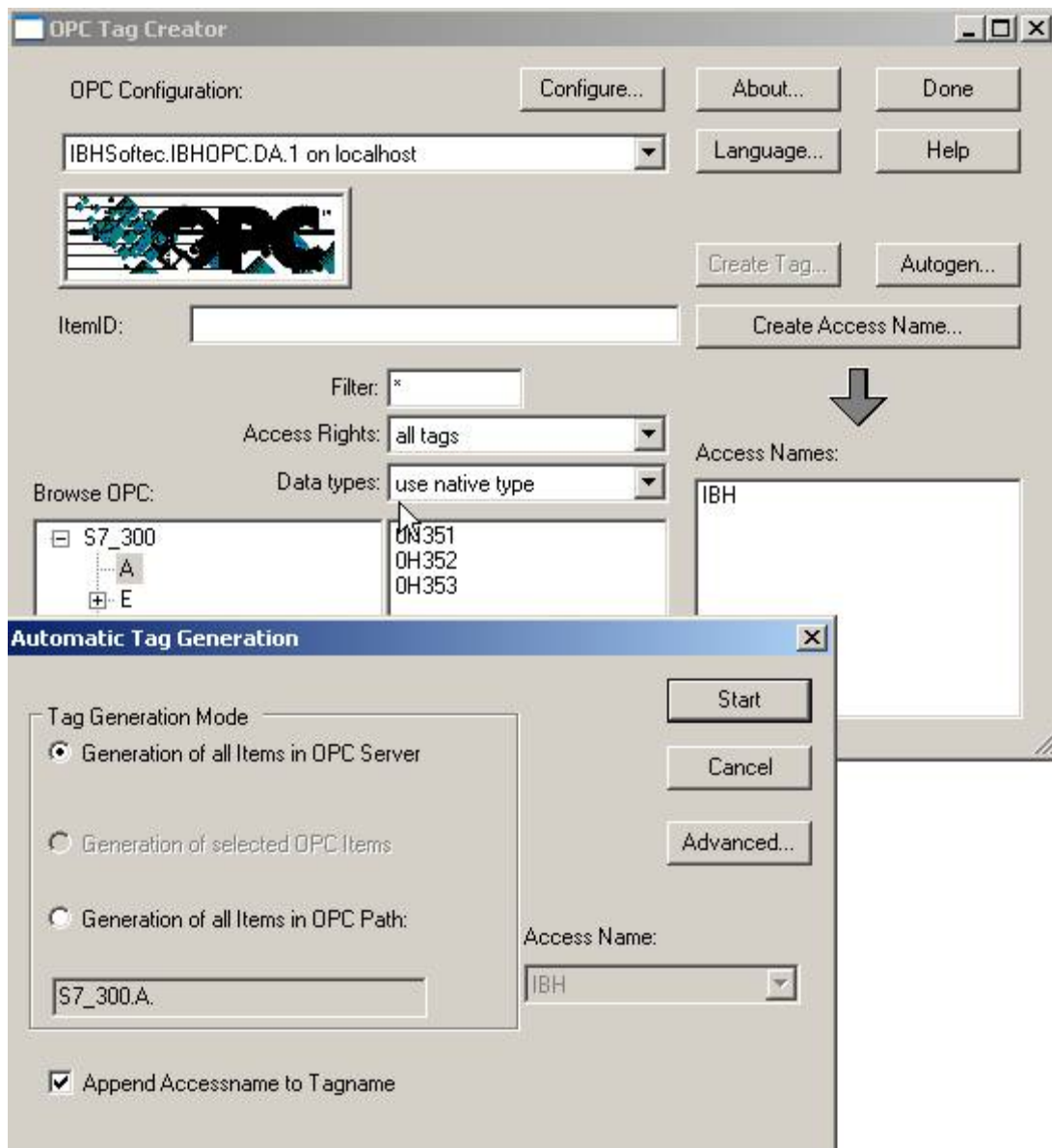
Please run the **Intouch WindowMaker**:

Now open the **OPC Tag Creator**:



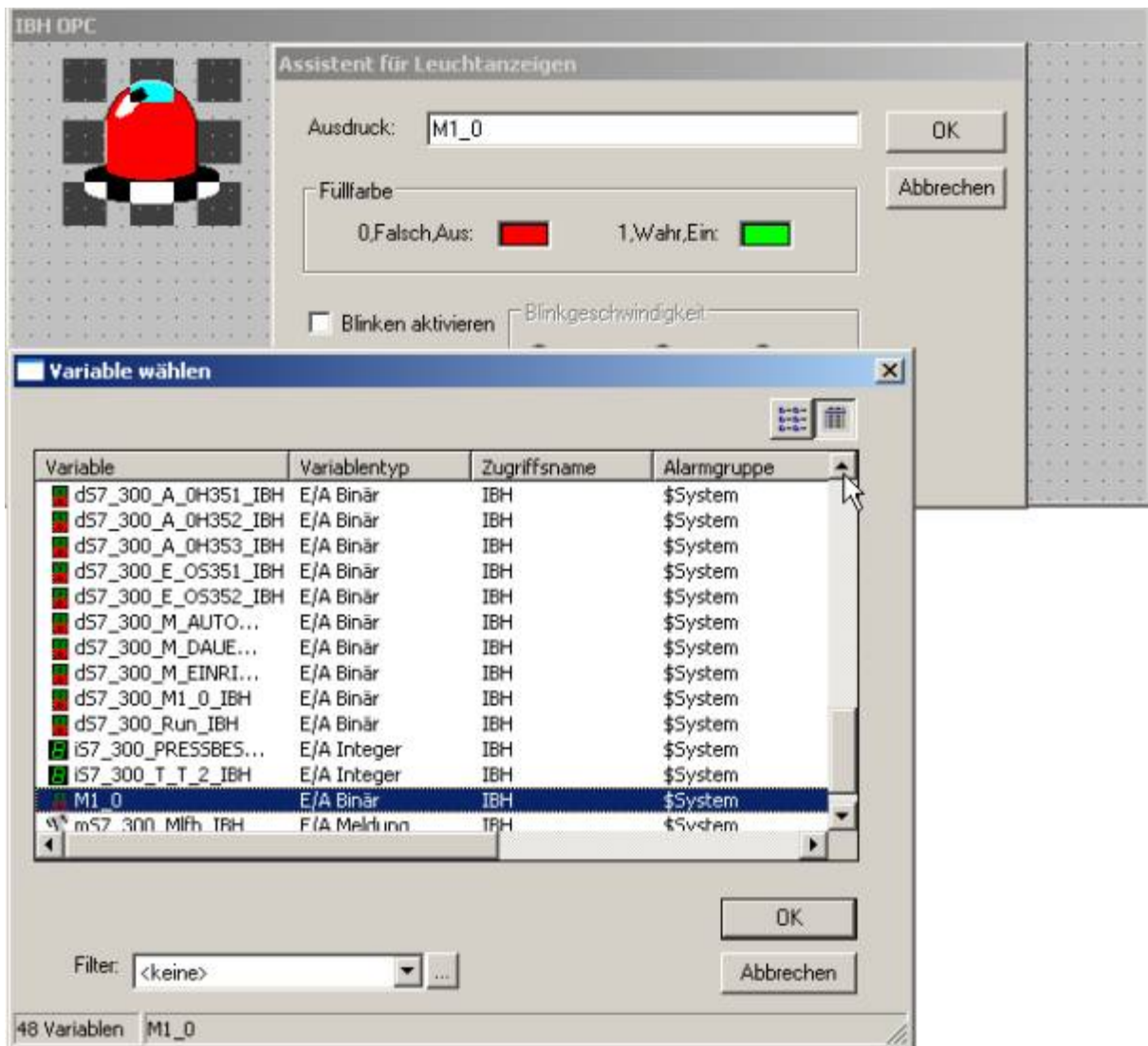
Select the OPC server **IBHsoftec.IBHOPC.DA.1** via <Configure> in the **General Configurations** screen at <OPCServer>.

Then the variables, that are already configured with IBH OPCEditor on the server are available :



Clicking the button **<Autogen>** permits to select, which variables are to be created.





The OPC variables may now be used in the **WindowMaker**.