

OPC Data Access 2.05 & 3.0 Client

OPC Data Access 2.05 & 3.0 Client Communication Protocol

[Supported device types and versions of devices](#)
[Communication line configuration](#)
[Communication station configuration](#)
[I/O tag configuration](#)
[Browsing of the OPC server address space](#)
[Historical Data Access](#)
[Tell commands](#)
[DCOM configuration for connecting a remote OPC server](#)
[Errors and problems](#)
[Literature](#)
[Changes and modifications](#)
[Document revisions](#)

Supported device types and versions

The protocol supports the client-side of communication with OPC (OLE for Process Control) DA (Data Access) servers according to the OPC DA specification ver. 2.05A and 3.0. It also supports the client-side of communication with OPC HDA (Historical Data Access) servers according to OPC HDA specifications ver. 1.20.

Communication line configuration

- Communication line category: **OPC**.
- OPC parameters:
 - **OPC Host**: Only Remote entrance requires it. Computer, where the program [D2000 OPC Server](#) is installed in - it is defined for OPC server type "Remote" only (maximum string: 50 characters). You can define names according to the UNC convention (e.g. \\server or server), DNS domain names (e.g. domain.com, *example.company.com*), or IP address (196.54.23.113).
 - **Backup Host**: Only Remote entrance requires it. It is a backup OPC host. If the parameter is defined, after the communication has failed, the process [D2000 KOM](#) is attempting to establish a connection alternately with the OPC host and Backup host.
 - **OPC Server**: Name ([ProgID](#)) of the OPC server (maximum string: 50 characters).
 - **Server Type**: according to the server type – *InProc*, *Local* or *Remote*.

Communication line parameters

Following communication line parameters can be configured for protocol "OPC Data Access 2.05 & 3.0":

Tab. No. 1

Name	Description	Unit	Default value
OPC HDA: Server if Available	OPC HDA server name (ProgID), if it is available. OPC HDA functions are not to be activated, use empty text.	-	
OPC HDA: Max. Number of Values	Sets the NumItems parameter for the synchronous reading of historical "raw" values. The implicit value 0 (zero) means all values in the given interval.	-	0

Authentication Level	<p>Authentication level used when establishing a connection to the OPC server. The OPC standard defines the following levels:</p> <ul style="list-style-type: none"> • <code>RPC_C_AUTHN_LEVEL_DEFAULT</code> (0) • <code>RPC_C_AUTHN_LEVEL_NONE</code> (1) • <code>RPC_C_AUTHN_LEVEL_CONNECT</code> (2) • <code>RPC_C_AUTHN_LEVEL_CALL</code> (3) • <code>RPC_C_AUTHN_LEVEL_PKT</code> (4) • <code>RPC_C_AUTHN_LEVEL_PKT_INTEGRITY</code> (5) • <code>RPC_C_AUTHN_LEVEL_PKT_PRIVACY</code> (6) <p>Note: The default value <code>RPC_C_AUTHN_LEVEL_CONNECT</code> (2) may no longer be sufficient. Microsoft has implemented security hardening to address the security issues described in CVE-2021-26414. See "KB5004442—Manage changes for Windows DCOM Server Security Feature Bypass (CVE-2021-26414)" for more details.</p>	-	2
Sequenced Transactions	Starts the sequencing of calling the function <i>Refresh2</i> of interfaces <i>Async I/O 2.0</i> and <i>Async I/O 3.0</i> . The calling executes after the first calling has been finished (after it gets the values).	YES /NO	NO
Repeat Failed Group Activation	Failed OPC group activations will be attempted again after a delay specified by the parameter "Group reactivation delay".	YES /NO	YES
Group Reactivation Delay	A delay after which a failed OPC group activations is repeated if repetition is enabled by the parameter "Repeat failed group activation".	sec	30
Reconnect After Failed Group Activation	When an OPC group activation fails, KOM disconnects from the OPC server and reestablishes a connection. The parameter is relevant in a configuration of redundant OPC servers (after a disconnect from an OPC server follows a reconnect to an alternate OPC server where group activation may be successful).	YES /NO	NO
GetGroupState Period	Period of executing the call of <i>GetGroupState</i> of interface <i>IOPCGroupStateMgt</i> . This synchronous call is repeated periodically and detects the problems which could occur in communication with OPC Server.	sec	10
Accept All GetGroupState Errors	<p>All the errors returned by calling of "GetGroupState" of interface <i>IOPCGroupStateMgt</i> are considered to be fatal (they will result in a disconnect from the OPC server and in a reconnect, or in a restart of the KOM process).</p> <p>An example of an error message returned by "GetGroupState": WARNING: Siemens.TXP.OPC caused COM/OPC error 80010108H on IOPCGroupStateMgt::GetState(), Error string : The object invoked has disconnected from its clients.</p>	YES /NO	NO
Stop KOM When OPC Server Fails	Allows stopping the KOM process when a fatal error occurs in communication. See the Note .	YES /NO	NO
Disconnect On Passive	<p>Only the active KOM process (i.e., the active instance connected to the HOT server) communicates with the OPC server. The KOM process, which becomes passive (by changing the active instance or switching the redundancy), closes the connection to the OPC server.</p> <p>Note: This parameter allows to reduce the OPC server load in redundant D2000 systems as well as to solve problems with license limitation of the number of OPC clients.</p>	YES /NO	NO

Note:

Fatal abort of communication with OPC Server causes the COM/OPC errors at callings:

- *GetGroupState* of interface *IOPCGroupStateMgt*,
- "Write" of interface *IOPCAsyncIO2*, *IOPCAsyncIO3*, and *IOPCSyncIO2*,
- "Read" of interface *IOPCAsyncIO2*, *IOPCAsyncIO3*, and *IOPCSyncIO2*.

Communication station configuration

Communication protocol: **OPC Data Access 2.05 & 3.0**.

Station address requires:

- **OPC Group Name** - maximum string: 80 characters. This name is only formal in most cases and has no direct relation to the addressing scheme of devices communicating with the OPC server. The name of the group must be unique, two stations cannot the same *OPC Group Name* within one link.
- **Type** - data access type. You can choose one of the *Async I/O 2.0*, *Async I/O 3.0*, and *Synchronous I/O* options. The *Async I/O 2.0* option - the asynchronous OPC interfaces *IOPCAsyncIO2* and *IOPCDataCallback* (*OnDataChange* and *OnReadCompleted* call-back procedures) are used for data reading. The *Synchronous I/O 2.0* option - OPC interface *IOPCSyncIO* is used. The *Async I/O 3.0* option - OPC interfaces *IOPCAsyncIO3* and *IOPCDataCallback* (*OnDataChange* and *OnReadCompleted* call-back procedures) are used for value reading of OPC items
- **Create Active** - if the option is checked, the OPC group is active and OPC items in this group get values. If the option is not checked, the OPC group and items are initialized, but the OPC items do not get values. It is possible to block the OPC group - station temporarily. The option controls the *pActive* parameter of *SetState* call of the OPC interface *IOPCGroupStateMgt*.
- **Update Rate** - it is given in milliseconds and defines the maximum speed of OPC item value changes for the option *Async I/O 2.0* or *Async I/O 3.0*. It is transferred as *pRequestedUpdateRate* parameter of *SetState* call of OPC interface *IOPCGroupStateMgt*. Warning - the information, set in tab **Time parameters** in dialog window [Communication station - configuration dialog box](#) defines the read rate of OPC items.

- **% Deadband** - the parameter range is from 0.0 up to 100.0%. The parameter can be only used for an analog type of items and determines the minimal change of the item for sending the value from the server to clients. For calculation, the OPC server uses the range defined by parameters **High EU** and **Low EU**, which can be get by using the interface *IOPCItemProperties*. The default value of this parameter is *0.0*. Value of % Deadband is transferred by means of parameter *pPercentDeadband* of the *SetState* call of the OPC interface *IOPCGroupStateMgt*.
Note: The protocol **OPC Client** described in this topic does not support the OPC interface *IOPCItemProperties*. For further information read up the document listed in the section **Literature**.
- **Time Bias** - If the OPC server and OPC client do not run in the same time zone, the parameter (given in minutes) will be used for the correct calculation of the time of the value of the "timestamp" OPC item. The default value of the parameter is 0 (zero). For further information read up the documents listed in the section **Literature**.

Station protocol parameters

The following station protocol parameters can be defined:

Name	Description	Unit	Default value
Do Read after Write	Allows verifying the value after the recording by synchronous reading.	YES/NO	NO
Prefer Synchronous Write	Allows writing data into OPC Server only by synchronous interface <i>IOPCSyncIO2</i> .	YES/NO	NO
Prefer VQT Write	Using the <i>Async I/O 3</i> option allows writing data by <i>WriteVQT</i> function of <i>IOPCASyncIO3</i> interface (i.e. the write that transfers the quality and the timestamp together with the value).	YES/NO	NO
Full Debug	Activates the debug list. It increases the number of information about the communication operation. It is recommended to use only for the detection of the problems and debugging.	YES/NO	NO
Merz OPC Server Type	Special mode for communication with OPC Servers made by Merz company. The conversion to/from <i>MS DOS Date Time Format</i> (2 x WORD) is used for values of TiA and ToA type.	YES/NO	NO
AddItems Delay	Delay which is artificially inserted between the <i>AddItems</i> callings to slow down the initialization phase.	ss:mss	0
QERR Value	An integer value of the "ERROR" state for conversion into Qi - quaternary input.	0,1,2,3	3
QOFF Value	An integer value of "OFF" state for conversion into Qi - quaternary input.	0,1,2,3	2
QON Value	An integer value of "ON" state for conversion into Qi - quaternary input.	0,1,2,3	1
QTRANS Value	An integer value of "TRANS" state for conversion into Qi - quaternary input.	0,1,2,3	0
Keep Values Valid as Weak	If the quality of the OPC item changes into "BAD", the value of the I/O tag will be valid with a <i>Weak</i> flag.	YES/NO	NO
Don't Repeat Failed AddItems	Blocks the effort to call <i>AddItems</i> which still repeats after its failure.	YES/NO	NO
Do AddItems in Single Call	Activates all the items of a group by a single call. It can speed-up the OPC communication start.	YES/NO	NO
Transaction Timeout	All callings of <i>Write</i> and <i>Refresh2</i> functions as registered as separate transactions. If they do not end up (successful or unsuccessful) within this timeout, the error message will occur in the trace file of communication.	sec	120
Reconnect /Reinitialization Delay	Timeout, which delays repetitions of failed operation: <ul style="list-style-type: none"> • creation of <i>AddGroup</i>, • creation of <i>AddItems</i> (see the SA protocol parameter), • repeated connection to OPC Server after it has been stopped or failed or disconnected, • creation of groups and items after repeated connection to OPC Server. 	sec	5
Do Sync Read Before Write If Unk. Type	Enables synchronous reading of the item value before the writing if the KOM process does not know the item data type (i.e. if there is a default value of data type "Empty/Default (VT_EMPTY)" and if the value must be written).	YES/NO	YES
Status Item Name	Address of OPC Item (OPC Item ID), which contains the OPC server or OPC group error status (e.g. depending on the state of communication). An I/O tag with this address must also be configured. If the status reports an error, it will affect the values of all I/O tags on the station (they will have the Weak flag set). The Status Item Inverted Operation parameter specifies what value corresponds to the error status.	-	-

Status Item Inverted Operation	Interpretation of the OPC Item Status Item Name with an error status of OPC server or OPC group. The NO value means that the False or 0 reports the correct state and True respectively non-zero value reports an error. The YES value means that the False or 0 means the error state and True respectively non-zero value reports correct status.	YES/NO	NO
Map NonSpecific as Flag	Mapping the " <i>Non Specific</i> " OPC DA quality flag into the attributes of the I/O tag value.	None, FA, FB, FC, FD, FE, FF, FG, FH, FI, FJ, FK, FL, FM, FN, FO, FP	None
Map LocalOverride as Flag	Mapping the " <i>LocalOverride</i> " OPC DA quality flag into the attributes of the I/O tag value.		
Map ConfigError as Flag	Mapping the " <i>Config Error</i> " OPC DA quality flag into the attributes of the I/O tag value.		
Map NotConnected as Flag	Mapping the " <i>Not Connected</i> " OPC DA quality flag into the attributes of the I/O tag value.		
Map DeviceFailure as Flag	Mapping the " <i>Device Failure</i> " OPC DA quality flag into the attributes of the I/O tag value.		
Map SensorFailure as Flag	Mapping the " <i>Sensor Failure</i> " OPC DA quality <i>flag</i> into the attributes of the I/O tag value.		
Map LastKnown as Flag	Mapping the " <i>Last Known</i> " OPC DA quality <i>flag</i> into the attributes of the I/O tag value.		
Map CommFailure as Flag	Mapping the " <i>Comm Failure</i> " OPC DA quality flag into the attributes of the I/O tag value.		
Map OutOfService as Flag	Mapping the " <i>Out Of Service</i> " OPC DA quality flag into the attributes of I/O tag value.		
Map WaitingForInitialData as Flag	Mapping the " <i>Waiting For Initial Data</i> " OPC DA quality flag into the attributes of I/O tag value.		
Map LastUsable as Flag	Mapping the " <i>Last Usable</i> " OPC DA quality flag into the attributes of the I/O tag value.		
Map SensorCal as Flag	Mapping the " <i>Sensor Cal</i> " OPC DA quality flag into the attributes of the I/O tag value.		
Map EGUExceeded as Flag	Mapping the " <i>EGU Exceeded</i> " OPC DA quality flag into the attributes of the I/O tag value.		
Map SubNormal as Flag	Mapping the " <i>Sub Normal</i> " OPC DA quality flag into the attributes of the I/O tag value.		
Reconnect After Error Count	If a consecutive number of errors during reading equals the value of this parameter, reinitialization of OPC connection will be performed. Value of zero means that reinitialization will not be performed (default behavior). The current implementation handles only errors in synchronous mode (type setting to "Synchronous I/O" in tab Address of Station object).	-	0

I/O tag configuration

Possible I/O tag types: **Ai, Ao, Ci, Co, Di, Qi, Dout, TiA, ToA, TiR, ToR, TxtI, TxtO**.

I/O tag address requires to define **OPC Item ID** (maximum string: 200 characters). If the OPC server supports the IOPCBrowseServerAddressSpace interface, the "OPC Item ID" address may be selected directly from the address list supported by the OPC server (by clicking the "Browse Items..." button, see the [Browsing of OPC server address space](#) section).

Note: if the I/O tag's address is specified as **%IGNORE**, such an I/O tag will be ignored.

Further required parameters (*OPC Item Parameters*) are:

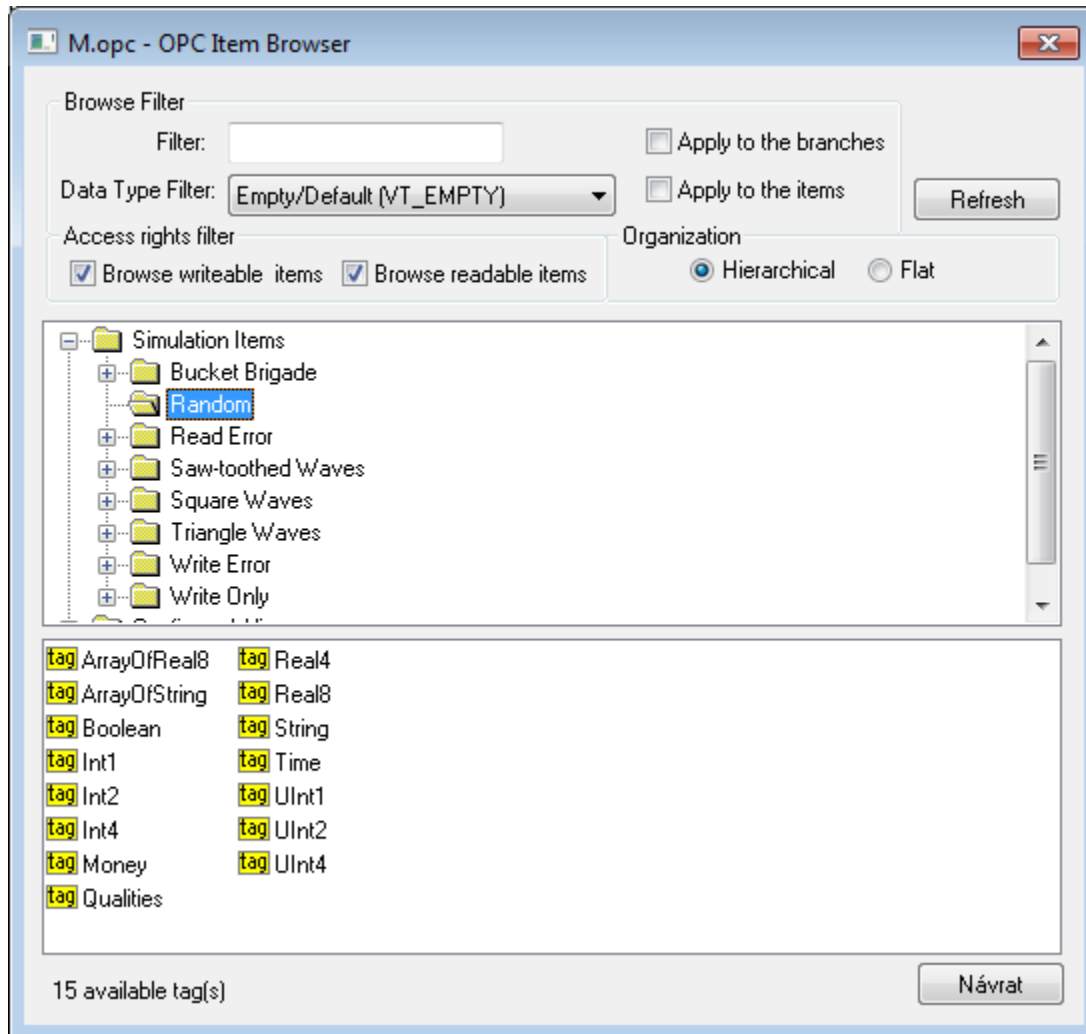
- **Type** - required data type (VARIANT). The default value is *Empty/Default (VT_EMPTY)*, - the OPC server makes a decision on the VARIANT type of the item.
- **Item Active** - if the option is checked, the item will be active and get values from the server. The option is transferred by the parameter *bActive* of the *plItemArray* structure into the *AddItems* call of the OPC interface *IOPCItemMgt*.
- **Bit Index** - bit number. It is defined as a number within the range of 0..31. It can be only used for the Di and Dout value types. Value received from the OPC server will be converted to 32-bit unsigned number and the binary value of the I/O tag is the value of a particular *Bit Index*. The conversion is allowed only for the VARIANT values of integer types (e.g. VT_UI1, see the **Type** parameter).

- **Array Index** - the option (a number within the range of 0..32767) determines the index in the array of received VARIANT value from the OPC server of the Array type. Array value on the defined position is assigned to the value of the I/O tag.

The protocol supports the configuration of the [Destination](#) tab of the I/O tag. If the value of the OPC item is of Array type, the communication protocol copies the values of the array, starting with the *ArrayIndex* item, into a column of a structured variable. The size of the structured variable is taken into consideration. If the VARIANT array is smaller than the number of structured variable rows, the empty rows of the structured variable will be invalidated. If the number of structured variable rows is smaller than the VARIANT array, the values which are over the limit will be ignored.

Browsing of the OPC server address space

Clicking on the "Browse Items..." button in the **Address** tab of the I/O tag dialog box opens another dialog window "OPC Item Browser".



Here the user can choose the items from the OPC server address space. The item representation can be "Hierarchical" or "Flat". Some of the OPC servers do not support the "Hierarchical" representation (the button "Hierarchical" is disabled). If both the representations are supported the "Hierarchical" may be switched over to "Flat" and vice-versa.

- **Hierarchical** - enables browsing of the OPC server address space according to the logical hierarchical groups and subgroups. The dialog box "OPC Item Browser" is divided into two parts:
 - hierarchical tree structure (top part)
 - list of OPC items (OPC tags) belonging to the selected branch (bottom part)
- **Flat** - displays all the available OPC items in one linear list.

OPC Item ID may be selected by double click on it. This item then occurs in the text field OPC Item ID in the Address tab of the I/O tag dialog box and the "OPC Item Browser" dialog window will be closed. To close the window without changes click on the "Cancel" button.

On the top part of the dialog window, there are **filter options**. Text "Filter" allows showing only the items according to the mask (some of the OPC servers support the so-called star convention).

"Data Type Filter" allows showing the items of a specified data type. The "Empty/Default" option is the default and enables viewing all the items.

The user must define whether the filter should be applied on the hierarchical tree structure (check the "Apply to the branches" button) or also on the OPC items (check the "Apply to the items" button).

According to the access rights, only readable items are displayed (check the "Browse readable items" button) or writable ones (check the "Browse writeable items" button). Default settings - both possibilities are enabled.

The changed conditions come into effect by clicking the "Refresh" button.

Copying tags

A keyboard shortcut **Ctrl+C** copies the names of tags in a selected branch into the Windows clipboard. All tags will be copied unless a specific tag is selected.

A keyboard shortcut **Ctrl+Shift+C** copies the names of tags in all browsed branches into the Windows clipboard.

Note: In versions from 20th December 2018 and newer, recycling of browser dialog has been implemented. If the dialog is closed by the Close button or after selecting a tag, it is actually only hidden and it is available for browsing by another I/O tag within the same station so that the tree structure of the browsed objects is preserved. Clicking on the close icon at the top right corner will cause the dialog to be really closed.

Historical Data Access

Implementation of this protocol partially supports the specification of OPC HDA (Historical Data Access). The synchronous reading of raw data is supported. TELL command **GETOLDVAL** activates the reading of historical values. The asynchronous option IOPCAsyncIO of mode for OPC DA should be set to continuously read both the historical data and the current one.

Tell commands

Command	Syntax	Meaning
STWATCH	STWATCH StationName	Tell command sends commands for the synchronous reading of values of all configured I/O tag (regardless of whether data access type is set to "Async I/O 2.0", "Async I/O 3.0" or "Synchronous I/O")
STCOMMAND	STCOMMAND StationName DISCONNECT	Tell command encloses immediately the active OPC connection of line (the parent of "StationName"). Then, the system is restarted and the connection reinitiated. If remote access is used and backup OPC host is configured, the servers are interchanged ("OPC Host" for "Backup Host" or vice-versa).
	STCOMMAND StationName CONNECT_PS	Closes the active OPC connection and enforces connection to primary OPC server "OPC Host". It is important only for remote access.
	STCOMMAND StationName CONNECT_BS	Closes the active OPC connection and enforces connection to backup OPC server "Backup Host". It is important only for remote access.

DCOM configuration for connecting a remote OPC server

Remote browsing/local registration of OPC server

D2000 KOM Process (version 7.01.020 rel. 055 and higher) supports getting GUID OPC Servers from ProgID on remote computers through the DCOM interface by OPCENUM utility (remote browsing). The local registration of the OPC server on the local PC is not required if the OPCENUM utility/windows service has been installed (both on the local PC with **D2000 KOM** process as an OPC client and on the remote PC with the OPC server) and the access rights allow the remote browsing. The OPCENUM utility is a part of the "OPC Core Components Redistributable" package available on <http://www.opcfoundation.org> or as a part of the OPC Server installation package.

If this error occurs (see the section **COM/OPC errors record**):

ERROR: OPCServerName caused COM/OPC error 80040153H on CoCreateInstanceEx(IID_OpcServerList), Error string : Invalid value for registry

it is necessary to install the *OPC Core Components Redistributable* also on the OPC client (KOM process). The registration of the OPC Server is optional. To register, use one of the two ways:

- Some OPC servers are supplied together with installation programs supporting the connection of OPC clients (third side) to the remote OPC server. These installation programs have names such as "OPC Server Connect" and the like. After installation, the OPC server (ProgID) appears in the list of OPC servers on the client's computer. D2000 OPC client uses this information for acquisition of CLSID from given ProgID (see the section **Communication line configuration**). An OPC server registered by this way, of course, can't be run on the client's side.
- Manual registration of the OPC server on the client's side. Proceed as follows:
 - Copy the OPC server (from the computer where has been installed) into an auxiliary directory on the client's side (computer).
 - Run the command line from the directory.
 - Register the OPC server. If the OPC server name is e.g. *OPCSERVER.EXE*, then enter a command:
OPCSERVER.EXE /regserver
and press ENTER. If the OPC server is only a .dll, use the *regsvr32* system utility.
 - The auxiliary directory and files can be deleted.

Always study the OPC server manual from its manufacturer in detail and confront it with the procedures described above.

Setting the access rights to OPC Server

As OPC DA standard uses COM/DCOM technology, the connection to the remote OPC server is checked for Windows operating system access rights. On both the local (OPC client) and remote (OPC Server) computer, the same user must be created (with the same password) and the user must be logged-on the client computer, or the KOM process (running as a Windows service) must be started with this user's credentials.

1. On the computer with the OPC server, use the "dcomcnfg" command to start "Component Services" (or start it in "Administrative Tools").
2. Select "Properties" menu in "Component Services" -> "Computer" -> "My Computer".
3. Make sure that DCOM is enabled, i.e. parameter "Enable Distributed COM on this computer" on the "Default Properties" tab is checked.
4. In the list of components (branch "DCOM Config" under "My Computer"), select the required OPC server and open the dialog box containing its parameters (Properties). Click the **Security** tab.
5. The parameter "Launch And Activation Permissions" set to **Customize** and click the **Edit...** button.
6. Find the required user and if is not in the list, add him/her into..
7. Enable the options "Remote Launch" and "Remote Activation".
8. The parameter **Access permission** set to **Customize** and click the **Edit...** button.
9. In the "Identity" tab check whether "The launching user" or "This user" option is enabled, which is also the checking of the user account that you configure. Typically, we recommend the setting "The launching user". If some problems occur, try direct setting "This user". In any case, be careful about setting "The interactive user", which is absolutely not recommended! The OPC server can be accessible and started only if some user is interactively logged on the computer with the OPC server. This setting leads to the problems such as the unavailability of an OPC server, e.g. after its starting until any user is logged on.
10. If you must use the setting "The system account (services only)", i.e. OPC server works as a Windows service, watch the level of access rights of "SYSTEM" according to the above-mentioned rules.

Wrong setting of access rights will probably cause an error (see the section **Errors and problems - COM/OPC error reports**):

[E]> ERROR: ServerProgID caused COM/OPC error 80070005H on CoCreateInstanceEx(CLSCTX_REMOTE_SERVER), Error string : Access is denied.

Watch logs in the **Event Viewer**.

The problem can be partially solved when you add the **Everyone** group into **Launch Permissions** and **Access Permissions**. To ensure, that the OPC server will be run under some user (not under **SYSTEM account**), open the tab **Identity** and define the data for the option **This user**. In this case, the OPC server is not safe.

If the KOM process runs as a Windows service, it cannot use the "/X1" or "/X2" parameters, because it does not work under the logged-on user but under the **SYSTEM account** and access rights verification executed by the OPC server fails. Use the "/X4" start-up parameter.

If it is not possible to start up the KOM process with the "/X4" parameter, check the user to be included in the "Log on as a service" Policy. Open the "Control Panel" -> "Administrative Tools" -> "Local Security Policy" -> "Local Policies" -> "User Rights Assignment" -> "Log on as a service".

For users working under Windows XP with ServicePack2 or later operating systems, it is recommended to change the parameter **Network access: Let Everyone permissions apply to anonymous users** (Local Security Policy -> Local Policies -> Security Options) to the value of *Enabled*.

If errors like this occur:

[E]> ERROR: ServerName caused COM/OPC error 80070005H on Advise(IID_IOPCDataCallback), Error string : E_ACCESSDENIED Access is denied.

it is necessary to add a user in the context of whose the communication runs. In "Component Services" on the computer with OPC client (KOM process) add this user to the list of users on the "COM Security" tab -> "Edit Limits", for both "Access Permissions" and "Launch and Activation Permissions" parameters and enable "Remote Access" / "Remote Activation". It is a problem connected with establishing the call-back connection with the OPC server. In this case, the roles are reversed and the OPC client (i.e. KOM process) works as a DCOM server. Adding this user and enabling the remote access enables establishing the call-back procedures between OPC client and OPC server. Call-back procedures are necessary for acquiring the values from the OPC server in the "Async I/O 2.0" and "Async I/O 3.0" asynchronous modes.



Note about OPC.SimaticNET and possibly other OPC servers

If the OPC server is configured to run under "The interactive user" on the last tab Identity of the "Properties" dialog window, it may cause the OPC server to be available only when a user is logged on the computer. We recommend changing this setting to "The launching user", "This user", resp. "The system account".



Another note about OPC.SimaticNET

In the specific case for the OPC communication to work, it was required:

- DCOM setting of OPC server OPC.SimaticNET - the option "This user" (e.g. *D2000* user) had to be enabled in the Identity tab. If "The launching user" was enabled, the connection to the OPC server timed out (both for *D2000* KOM and Matricon OPC Explorer), although it could be seen in Task Manager that the OPC server (opcserver.exe) was started under the specified user.
- the specific user with whose credentials the OPC server was running (e.g. *D2000* user) had to be in the "Distributed COM Users" and "Administrators" groups. If the user was not an administrator, the KOM process reported an error *CoCreateInstanceEx (CLSCTX_REMOTE_SERVER), Error string: Class not registered*.

Errors and problems

When starting or during the course of communication, the following error reports could occur. For easier identification of the problem, set the communication tracking level at least to the level **Monitor**, or temporarily to the level **Monitor & disk** in the [configuration dialog box](#) of a particular communication line. If you select the level **Monitor & disk**, the file *line_name.LOG* containing all debug and error logs will be created in the subdirectory **IT RACE** of the application directory on the computer with running process [D2000 KOM](#).

Error:	WriteAsync - FAILED (transactionID) - Item : 'OPCItemID'
Description:	Attempt to write the value using the Write function call of the OPC interface <i>IOPCAsyncIO2</i> failed.
Error:	ShutDown OPC Server : 'ServerProgID' !
Description:	The OPC server was stopped, probably in the correct way even though it has active clients.
Error:	OPC Server 'ServerProgID' is unavailable !
Description:	The remote DCOM OPC server is unavailable. Process D2000 KOM as a client is still attempting to connect the server. Check the PC with installed OPC server (if it is running and is correctly connected to the local network).
Error:	SetCallBack - FAILED, Group : 'OPCGroupName', Server : 'ServerProgID' !
Description:	Fatal error - call-back procedures of the OPC interface <i>OPCDataCallback</i> could not have been initialized for the OPC group. Contact the Ipssoft's technical support.
Error:	SetGroupState - FAILED, Group : 'OPCGroupName', Server : 'ServerProgID' !
Description:	Fatal error - <i>SetState</i> function call of the OPC interface <i>IOPCGroupStateMgt</i> failed. Contact the Ipssoft's technical support.
Error:	EnableSubscribe - FAILED, Group : 'OPCGroupName', Server : 'ServerProgID' !
Description:	Fatal error - <i>SetEnable</i> function call of the OPC interface <i>IOPCAsyncIO2</i> failed. Contact the Ipssoft's technical support.
Error:	RefreshAllAsync - FAILED, Group : 'OPCGroupName', Server : 'ServerProgID' !
Description:	Fatal error - <i>Refresh2</i> function call of the OPC interface <i>IOPCAsyncIO2</i> failed. Contact the Ipssoft's technical support.
Error:	OPCConnectToServer - FAILED, Server : 'ServerProgID' !
Description:	Fatal error - it is not possible to connect to the OPC server. Check the OPC server parameters - the tab OPC in the configuration dialog box of the communication line and the tab Address in the configuration dialog box of the communication stations.
Error:	OPCAddGroup - FAILED, Group : 'OPCGroupName', Server : 'ServerProgID' !
Description:	Fatal error - the OPC group could not be created by calling the <i>AddGroup</i> function of the OPC interface <i>IOPCServer</i> . Contact the Ipssoft's technical support.
Error:	ReadSync - FAILED, Item : 'OPCItemID' !
Description:	<i>Read</i> call of the OPC interface <i>IOPCSyncIO</i> failed.
Error:	AddItems - FAILED (Group not connected), Item : 'OPCItemID' !
Description:	The OPC item could not have been initialized, because creating the OPC group has failed. Contact the Ipssoft's technical support.

Error:	Write - FAILED (transactionID) , Item : 'OPCItemID' !
Description:	Writing the value into the OPC server failed. Writing is performed by the <i>Write</i> function called by the interfaces <i>IOPCSyncIO</i> or <i>IOPCAsyncIO2</i> according to the parameter Type (data access type) of the OPC group.
Error:	Write - FAILED, OPC Server is disconnected, Item : 'OPCItemID' !
Description:	The value could not have been written into the OPC server, because the OPC server is disconnected.
Error:	Group parameters error, Group: 'OPCGroupName', Server : 'ServerProgID' !
Description:	OPC group configuration parameters are wrong. Check the OPC parameters - the tab Address in the configuration dialog box of a particular communication station.

COM/OPC error reports

The error reports described in the "Errors and problems" section are generated by a high level of the OPC client. Most of the errors will also contain an error report on the COM/OPC level. The format of such error reports is as follows:

WARNING/ERROR: ServerProgID caused COM/OPC error ErrorCodeHexadecimal on CallDescription, Error string : ErrorDescription

For example:

WARNING: Matrikon.OPC.Simulation.1 caused COM/OPC error 80010108H on IOPCGroupStateMgt::GetState(), Error string : The object invoked has disconnected from its clients.

ERROR: Matrikon.OPC.Simulation.1 caused COM/OPC error 800706BAH on IOPCSyncIO::Read(), Error string : The RPC server is unavailable.

The error reports are important for the problem analysis and will be required by Ipssoft's technical support if any problem occurs.

Literature

The documents are available on <http://www.opcfoundation.org/>.

- OPC FOUNDATION, Data Access Custom Interface Standard, Version 2.05A, June 28, 2002.
- OPC FOUNDATION, Data Access Custom Interface Standard, Version 3.00, March 4, 2003.
- OPC FOUNDATION, OPC Common Definitions and Interfaces, Version 1.0, October 27, 1998.
- OPC FOUNDATION, Using OPC via DCOM with Microsoft Windows XP Service Pack 2, (C) 2004 OPC Foundation Inc.

Other sources:

- OPC DCOM White Paper, Richard C. Harrison, Intellution Inc. © Intellution Inc. 1998

Changes and modifications

- January 3rd, 2003 - added the parameter UPDATE_RATE, value arrays.
- January 23rd, 2004 – revised parameters group/item, error reports, and DCOM.
- November 22nd, 2005 - new protocol parameters implemented.
- October 10th, 2007 - parameters of protocol Async I/O 3.0 update
- December 3rd, 2007 - added the information about destination column
- March 2009 - added the support of HDA
- February 2010 - OPC Item browsing
- March 2013 - tell commands
- July 2021 - added support for working with 64-bit INT / UINT

Document revisions

- Ver. 1.0 – February 9th, 2000
- Ver. 1.1 - January 3rd, 2002
- Ver. 1.2 - January 23rd, 2004
- Ver. 1.3 - November 22nd, 2005
- Ver. 1.4 - October 10th, 2007
- Ver. 1.5 - December 3rd, 2007
- Ver. 1.6 - March 13th, 2009

- Ver. 1.7 - February 8th, 2010
- Ver. 1.8 - March 5th, 2013
- Ver. 1.9 - July 16th, 2021
- Ver. 1.10 - Feb 7th, 2021 - added parameter "Authentication Level"



Related pages:

[Communication protocols](#)