



Description: How to use TraceWorX32 to troubleshoot OPC ToolWorX 3.1 OPC Server issues

OS Requirement: Windows NT, 2000, XP, or 2003 Server.

General Requirement: Requires OPC ToolWorX v.3.1.

Capabilities

TraceWorX32 (also referred as ReleaseTrace) provides online diagnostics and tuning of all OPC servers build on top of OPC ToolWorX (from version v3.1).

TraceWorX32 tracks the runtime activity for each OPC server application and logs the runtime data to a log file based on user-configured trace levels. The log file provides a thorough, color-coded report detailing all activity for the application, including the time, the date, the severity level, and a description of the event or problem.

TraceWorX32 also features several options for reporting issues to technical support. If you are experiencing problems with any applications, the log file deployment options, such as compressing and e-mailing log files, are ideal for tracking and archiving data and sending detailed reports to technical support. Developers can use these reports to identify the source of the problems.

Runtime activity is logged to the log file based on the trace level specified (value 0 - 1000). The trace level corresponds to the severity level of the runtime activity for the module. Runtime activity with a severity level greater than or equal to the specified trace level will be logged to the log file. For example, a trace level of Normal (600) filters out all activity with a severity level less than 600.

By default, tracing for modules is not enabled (i.e. the trace level is 1000). To log all activity to the log file (without regard to severity), set the level to 0 (logging all).

TraceworX32 registration and unregistration

TraceWorX32 module is automatically registered with the OPC server as part of server's registration. The same applies for unregistration. After you register your server, you will find new ReleaseTraceModuleDesc registry entry in your server's registry key (i.e. under key HKEY_LOCAL_MACHINE\SOFTWARE\companyname\applicationname).

Configuring Trace Parameters for OPC servers

There are configurable registry settings specified for each OPC server that you can change in order to fine-tune TraceWorX32 for the OPC server. You can create the entries in server's registry key (i.e. under key HKEY_LOCAL_MACHINE\SOFTWARE\companyname\applicationname). There are several ways how to configure the TraceWorX32 parameters:

- You can use the TraceWorX32 user interface that you find in Start Menu folder of the OPC ToolWorX.
- You can use the OPCAdmin application. (See more information on the OPCAdmin application in the product documentation.)
- You can create/modify/remove the entries manually.

The following is a list of the registry entries (configurable parameters):

Log File Location

Registry value name: ReleaseTraceLogFile

Registry value type: REG_SZ

Description: Full path to release trace log file. If the entry is missing (default) the following paths are used:

- Win2000: CSIDL_COMMON_APPDATA usually "Documents and Settings\All Users\Application Data\ICONICS\logs"

- WinNT: "WINNT\Profiles\All Users\Application Data\ICONICS\logs"

- Win98: "Windows\Profiles\All Users\Application Data\ICONICS\logs"

Max File Size

Registry value name: ReleaseTraceMaxFileSize

Registry value type: REG_DWORD

Description: Max size of the active logging file (in Megabytes) before archiving. A value of zero means no limit. The default is 1 MB. Archiving consists of deleting the old archive file, renaming the active file with the archive file name and creating a new active file. This registry setting can be altered on-the-fly.



Trace Level

Registry value name: ReleaseTraceLevel

Registry value type: REG_DWORD

Description: Trace level value. This registry setting can be altered on-the-fly.

Values: 0 to 1000 - default is 1000 (i.e. no logging).

Output Debug String

Registry value name: OutputDebugString

Registry value type: REG_DWORD

Description: Specifies whether the TraceWorX messages are also echoed into the debug window. Set it to zero to turn off echo to debug window. Note that this option is (unlike the others) global for whole TraceWorX32 subsystem, so if you change its value, it changes its behavior for all the modules (OPC servers) using TraceWorX32. Also it's registry entry is located on a central place under HKLM\ICONICS\ReleaseTrace\Settings registry key.

Values: 0 or 1

Trace levels used in TraceWorX and OPC ToolWorX

Here are the levels defined by TraceWorX32:

- 1000 - RELTRACE_LEVEL_NONE
- 900 - RELTRACE_LEVEL_SEVERE_ERROR
- 800 - RELTRACE_LEVEL_ERROR
- 700 - RELTRACE_LEVEL_WARNING
- 600 - RELTRACE_LEVEL_NORMAL
- 400 - RELTRACE_LEVEL_DEBUG
- 100 - RELTRACE_LEVEL_DEBUG_VERBOSE
- 0 - RELTRACE_LEVEL_ALL

Here are some more OPC ToolWorX debugging levels defined by OPC ToolWorX:

- 100 - OPCTK_RELTRACE_LEVEL_DEBUG_OK
- 150 - OPCTK_RELTRACE_LEVEL_DEBUG_WARNING

200 - OPCTK_RELTRACE_LEVEL_DEBUG_ERROR

You can also specify your own level in case these definitions do not fit your needs.

Using TraceWorX in your code - TRACEWORX_MSG(...) macro

You can put the TraceWorX32 traces into your code fairly easily. To do so, use the predefined TRACEWORX_MSG macro definition that should be accessible from most places in your code. The syntax of the macro definition is TRACEWORX_MSG(Msg, wLogLevel), where Msg is your trace message, and wLogLevel is the trace level. Here is a typical example:

```
TRACEWORX_MSG( L"This is my trace message",  
RELTRACE_LEVEL_NORMAL)
```

Displaying the XML log file

The XML log file can be displayed in the TraceWorX32 user interface, or in any XML enabled viewer.

Distributing TraceWorX32 with your server

All the OPC servers using TraceWorX (built with TraceWorX32 support) need to install and register the TraceWorX.dll file.