

Description: Guide to running TraceWorX32 to find OPC Communication issues.

OS Requirement: Win 2000, XP Pro, Server 2003, Vista, Server 2008, Windows 7

General Requirement: N/A

Introduction

TraceWorX32 provides online diagnostics and tuning of applications running in the GENESIS32 system. It is designed for system integrators, OEMS, and customers who want tools for doing their own troubleshooting and diagnostics.

TraceWorX32 tracks the runtime activity for each GENESIS32 applications 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 the description of the even or problem.

This document will list the steps you should take to turn on TraceWorX32 to monitor OPC communication issues on your server and client machine.

Setting up TraceWorX32 on the Server

1. Open TraceWorX32 from Start → Programs → ICONICS Tools → TraceWorX32
2. The file opens in your web browser with all the modules that are available for tracing.

NOTE: Internet Explore may have restricted you from viewing this file. You should click on the message and select “Allow Blocked Contents...” to view the modules.

3. Select GenBroker Server in the list of modules.

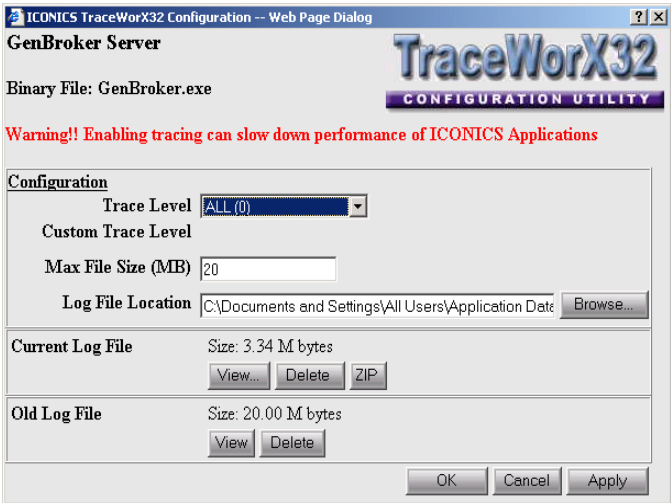
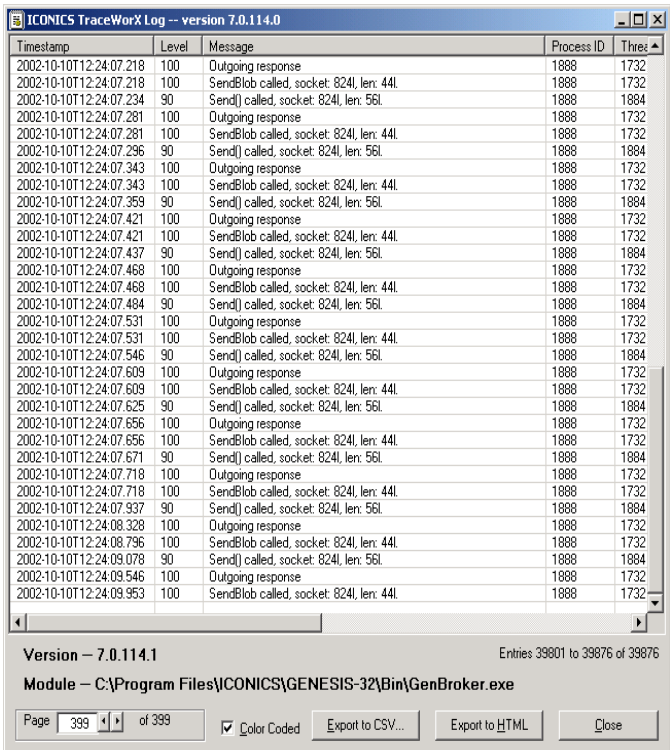


Figure 1 - GenBroker Server Trace Configuration

4. In the window that comes up, set the Trace Level to ALL(0).You can change the Max File Size, by default it is 0, which means that the file can grow infinitely large. You can also change the Log File Location to somewhere that is more convenient for you to find later.
5. Figure 2 shows an example of the GenBroker log file. If your file does not look similar to it or does not continuously increase in size, then there is a communication problem that may be better diagnosed by looking at the client-side.



Timestamp	Level	Message	Process ID	Thread ID
2002-10-10T12:24:07.218	100	Outgoing response	1888	1732
2002-10-10T12:24:07.218	100	SendBlob called, socket: 8241, len: 441.	1888	1732
2002-10-10T12:24:07.234	90	Send() called, socket: 8241, len: 561.	1888	1884
2002-10-10T12:24:07.281	100	Outgoing response	1888	1732
2002-10-10T12:24:07.281	100	SendBlob called, socket: 8241, len: 441.	1888	1732
2002-10-10T12:24:07.296	90	Send() called, socket: 8241, len: 561.	1888	1884
2002-10-10T12:24:07.343	100	Outgoing response	1888	1732
2002-10-10T12:24:07.343	100	SendBlob called, socket: 8241, len: 441.	1888	1732
2002-10-10T12:24:07.359	90	Send() called, socket: 8241, len: 561.	1888	1884
2002-10-10T12:24:07.421	100	Outgoing response	1888	1732
2002-10-10T12:24:07.421	100	SendBlob called, socket: 8241, len: 441.	1888	1732
2002-10-10T12:24:07.437	90	Send() called, socket: 8241, len: 561.	1888	1884
2002-10-10T12:24:07.468	100	Outgoing response	1888	1732
2002-10-10T12:24:07.468	100	SendBlob called, socket: 8241, len: 441.	1888	1732
2002-10-10T12:24:07.484	90	Send() called, socket: 8241, len: 561.	1888	1884
2002-10-10T12:24:07.531	100	Outgoing response	1888	1732
2002-10-10T12:24:07.531	100	SendBlob called, socket: 8241, len: 441.	1888	1732
2002-10-10T12:24:07.546	90	Send() called, socket: 8241, len: 561.	1888	1884
2002-10-10T12:24:07.609	100	Outgoing response	1888	1732
2002-10-10T12:24:07.609	100	SendBlob called, socket: 8241, len: 441.	1888	1732
2002-10-10T12:24:07.625	90	Send() called, socket: 8241, len: 561.	1888	1884
2002-10-10T12:24:07.656	100	Outgoing response	1888	1732
2002-10-10T12:24:07.656	100	SendBlob called, socket: 8241, len: 441.	1888	1732
2002-10-10T12:24:07.671	90	Send() called, socket: 8241, len: 561.	1888	1884
2002-10-10T12:24:07.718	100	Outgoing response	1888	1732
2002-10-10T12:24:07.718	100	SendBlob called, socket: 8241, len: 441.	1888	1732
2002-10-10T12:24:07.937	90	Send() called, socket: 8241, len: 561.	1888	1884
2002-10-10T12:24:08.328	100	Outgoing response	1888	1732
2002-10-10T12:24:08.796	100	SendBlob called, socket: 8241, len: 441.	1888	1732
2002-10-10T12:24:09.078	90	Send() called, socket: 8241, len: 561.	1888	1884
2002-10-10T12:24:09.546	100	Outgoing response	1888	1732
2002-10-10T12:24:09.953	100	SendBlob called, socket: 8241, len: 441.	1888	1732

Figure 2 - Sample GenBroker Log File

Setting up TraceWorX32 on the Client

1. Open TraceWorX32 the same way you launched it on the server machine.
2. Turn on tracing for GenClient, GenClientU, and any ICONICS modules that requests OPC data..

NOTE: You can follow Step 4 in the previous section to setup the tracing level and trace log file path.

3. If there is disruption between communications, you will see message with high severity on your GenClient log similar to Figure 3.



October 2013

[CONICS TraceWorkLog -- version 7.0.114.0]					[] [X]	
Timestamp	Level	Message	Process ID	Thres		
2002-10-10T13:46:43.593	400	RecvThreadFunc finished	1952	872		
2002-10-10T13:46:43.609	700	Failure on CDPCCSecPoint: 0x80004005	1952	1792		
2002-10-10T13:46:43.640	400	CWapper: PassResponse(), DESTROY received	1952	788		
2002-10-10T13:46:43.765	700	Failure on CDPCCSecPoint: 0x80004005	1952	1792		
2002-10-10T13:46:43.781	400	CTcpWapper: Init()	1952	1732		
2002-10-10T13:46:43.781	400	Init() Addr: PANEL, Port: 38080	1952	1636		
2002-10-10T13:46:43.828	700	Init() Socket exception: 11001	1952	1636		
2002-10-10T13:46:43.843	400	Exit()	1952	1636		
2002-10-10T13:46:43.859	700	CTcpWapper: CreateServer()	1952	1636		
2002-10-10T13:46:43.968	400	Exit()	1952	788		
2002-10-10T13:46:43.968	400	OnBreak called, socket: 12521	1952	280		
2002-10-10T13:46:43.984	400	SendThreadSub finished	1952	280		
2002-10-10T13:47:13.875	400	Init() Addr: PANEL, Port: 38080	1952	1636		
2002-10-10T13:47:13.890	700	Init() Socket exception: 11001	1952	1636		
2002-10-10T13:47:13.890	400	Exit()	1952	1636		
2002-10-10T13:47:13.906	700	CTcpWapper: CreateServer()	1952	1636		
2002-10-10T13:47:43.937	400	Init() Addr: PANEL, Port: 38080	1952	1636		
2002-10-10T13:47:43.953	700	Init() Socket exception: 11001	1952	1636		
2002-10-10T13:47:43.953	400	Exit()	1952	1636		
2002-10-10T13:47:43.968	700	CTcpWapper: CreateServer()	1952	1636		
2002-10-10T13:48:13.968	400	Init() Addr: PANEL, Port: 38080	1952	1636		
2002-10-10T13:48:13.984	700	Init() Socket exception: 11001	1952	1636		
2002-10-10T13:48:13.984	400	Exit()	1952	1636		
2002-10-10T13:48:14.000	700	CTcpWapper: CreateServer()	1952	1636		
2002-10-10T13:48:44.000	400	Init() Addr: PANEL, Port: 38080	1952	1636		
2002-10-10T13:48:44.015	700	Init() Socket exception: 11001	1952	1636		
2002-10-10T13:48:44.015	400	Exit()	1952	1636		
2002-10-10T13:48:44.031	700	CTcpWapper: CreateServer()	1952	1636		
2002-10-10T13:49:14.031	400	Init() Addr: PANEL, Port: 38080	1952	1636		
2002-10-10T13:49:14.046	400	RecvThreadFunc starting, Socket: 5121	1952	1944		
2002-10-10T13:49:14.046	400	SendThreadFunc starting, Socket: 5121	1952	1860		
2002-10-10T13:49:14.062	400	CTcpWapper: CreateServer()	1952	1636		
					Previous Next	
Version -- 7.0.114.1						
Entries 201 to 253 of 259						
Module -- C:\Program Files\CONICS\GENESIS-32\Bin\GenClientU.dll						
Page	3	of 3	☒ Color Coded	Export to CSV...	Export to HTML	Close

4. Within the log files created for the modules requesting OPC data, you will also see entries in the log file with high severity similar to Figure 4.

Timestamp	Level	Message
2002-10-09T21:52:31.593	400	Registry: Use Language Aliasing = 1
2002-10-09T21:52:46.078	800	Severly: 750, Message: GraphVox\32\HVAC_AHU1: Failed to get localeID for language
2002-10-09T21:53:31.109	400	Open VBFX Project
2002-10-09T21:53:31.250	400	AddAllClientItems
2002-10-09T21:53:35.365	400	Open VBFX Project
2002-10-09T21:53:35.328	400	AddAllClientItems
2002-10-09T21:53:37.458	800	Severly: 750, Message: GraphVox\32\HVAC_Chiller1: Failed to get localeID for language
2002-10-09T21:54:08.375	400	Open VBFX Project
2002-10-09T21:54:08.365	400	AddAllClientItems
2002-10-09T21:54:10.843	800	Severly: 750, Message: GraphVox\32\LaserDoomMeasure: Failed to get localeID for lang
2002-10-09T21:55:04.125	400	Registry: Use Global Aliasing = 1
2002-10-09T21:55:04.140	400	Registry: Use Language Aliasing = 1
2002-10-09T21:55:04.500	400	Open VBFX Project
2002-10-09T21:55:04.515	400	AddAllClientItems
2002-10-09T21:55:07.250	800	Severly: 750, Message: GraphVox\32\Microsoft_Demo_BikeAssembly: Failed to get loca
2002-10-09T21:55:48.390	400	Open VBFX Project
2002-10-09T21:55:48.421	400	AddAllClientItems
2002-10-09T21:55:48.213	800	Severly: 750, Message: GraphVox\32\Microsoft_Demo_BikePaintShop: Failed to get loc
2002-10-09T22:13:31.140	400	Registry: Use Global Aliasing = 1
2002-10-09T22:13:31.140	400	Registry: Use Language Aliasing = 1
2002-10-09T22:13:31.453	400	Open VBFX Project
2002-10-09T22:13:31.468	400	AddAllClientItems
2002-10-09T22:13:59.953	800	Severly: 750, Message: GraphVox\32\test: Failed to get localeID for language ". Can't fin
2002-10-09T22:15:53.484	800	Severly: 750, Message: GraphVox\32\test: Failed to get localeID for language ". Can't fin
2002-10-10T00:17:59.968	700	UpdateOPCDa[...].point \\\PanelCONICS.Simulator.1\SimulatePLC.Ramp does not exist
2002-10-10T00:17:59.984	700	UpdateOPCDa[...].point \\\PanelCONICS.Simulator.1\SimulatePLC.InlFanSp does not e
2002-10-10T12:15:37.906	400	Registry: Use Global Aliasing = 1
2002-10-10T12:15:37.921	400	Registry: Use Language Aliasing = 1
2002-10-10T12:15:39.062	400	Open VBFX Project
2002-10-10T12:15:39.109	400	AddAllClientItems
2002-10-10T12:15:43.250	800	Severly: 750, Message: GraphVox\32\test: Failed to get localeID for language ". Can't fin
2002-10-10T12:22:56.265	800	Severly: 750, Message: GraphVox\32\test: Failed to get localeID for language ". Can't fin

Version -- 7.0.114.0

Module -- C:\Program Files\ICONICS\GENESIS-32\Bin\Gwx32.exe

Entries 1 to 71 of 71

Page 1 of 1

☒ Color Coded

5. The combination of all the above mentioned log files determined that the communications link between the server and client was severed.
6. In this situation, you will need to check the network communications along with GenBroker and GenClient setup.

7. You can save these files in CSV or html format and email to ICONICS for further analysis and troubleshooting.

Just like we can use TraceWorX32 for machines with GENESIS32 installed, we can use it on WebHMI client machines as well. The steps are similar to those on the server side, with two exceptions: the URL for the TraceWorX32 page is different and the number of options for tracing will be limited to those WebHMI components installed on the client.

The URL for TraceWorX32 on a WebHMI client is as follows:

<http://<NODE>/WebHMI/Samples/TraceWorX32.htm>

where <NODE> is the name of the WebHMI server machine.