

# ***GENESIS32***

## **V8.05 Resolved Anomalies List**

***November 2008***



***OPC Automation at your fingertips***

100 Foxborough Blvd., Foxborough, MA 02035  
508.543.8600 • FAX 508.543.1503  
www.iconics.com • E-mail info@iconics.com

© 2008 ICONICS, Inc. All Rights Reserved.  
Specifications and information subject to change without notice

# GENESIS32 V8.05 Anomalies Resolved

## AlarmWorX32 Resolved Anomalies

#	Summary
1	Resolved an issue with the AlarmWorX32 Logger where it could not create tables in MS Access.
2	Resolved an issue with the AlarmWorX32 Logger where it could crash on a Dual-core system. The Processor Affinity is set to one to address this issue.
3	Enhanced the AlarmWorX32 Report ActiveX automation interface adding ScrolltoHome and ScrolltoEnd OLE Automation functions.
4	Resolved an issue with the AlarmWorX32 Report ActiveX where time filters did not work in Crystal Reports.
5	Resolved an issue with the AlarmWorX32 Report ActiveX where if you used a global alias in the file path, it would load the file and show a chart instead of the data grid.
6	Resolved an issue with the AlarmWorX32 Report ActiveX where the GetCellText sometimes returned "*****".
7	Resolved an issue with the AlarmWorX32 Server where disabling / enabling the limit alarms could remove dynamic OPC tags.
8	Resolved an issue with the AlarmWorX32 Server where it could crash when requesting DA tags from a large tree of areas.
9	Added an INI file set-able option to the AlarmWorX32 Server to disable the generation of "bad alarm tag" and "bad OPC tag" event messages during startup of the server. Enabling this option can significantly reduce the time to start up, especially if some of the OPC servers AlarmWorX32 is communicating to are offline. The INI file setting is: [AWX32\AWXSvr32] NotifyDAErrors=FALSE
10	Enhanced the AlarmWorX32 Server adding support for a DA tag that returns the number of active alarms within an area. The tags "Area_Active_Count" and "Recursive_Active_Count" were added
11	Added an INI file set-able option to the AlarmWorX32 Server that can be used to set the internal scan rate (in ms) used by the DA portion of the alarm server to the AE side of the Alarm server. The example below is for a 10s scan – which means that if an alarm goes active, the DA client might not be updated for 10 seconds. [HKEY_LOCAL_MACHINE\SOFTWARE\ICONICS\AWX32\AWXSvr32] DWord: DAMaxPollRate = 10000
12	Resolved an issue with the AlarmWorX32 Server where the alarm DA tag VRU.Area_UnAcked was returning bad quality . The cause was a conflict with the DA area name VRU.
13	Resolved an issue with the AlarmWorX32 Server where time stamps were not being set correctly for expression inputs when the alarm was set to already ack.
14	Resolved an issue with the AlarmWorX32 Server where it could crash when a client application was requesting DA Tags.
15	Resolved an issue with the AlarmWorX32 Server where it was using too much CPU when there were many areas (i.e. 10,000 digital alarm spread over 1,000 areas)..
16	Resolved an issue with the AlarmWorX32 Server where the Recursive_Active tag had a space between the _ and the A of "active". This was inconsistent with our other OPC-DA tags for areas: Area_Active, Area_UnAcked, and Recursive_UnAcked. Now AlarmWorX32 Server browses the correct name (Recursive_Active, without the space character), but both names (with or without the space character in the name) can be requested (for compatibility with existing applications and graphics).
17	Resolved an issue with the AlarmWorX32 Server where an acknowledged alarm would not go away if it was immediately acknowledged after it went into high alarm and then went into

	low alarm before 10 seconds was up.
18	Resolved an issue with the AlarmWorX32 Server Configurator where the Import/Export of unicode csv files didn't work properly.
19	Resolved an issue with the AlarmWorX32 Server where the ICONICS.AlarmServer_1\Area_Active tag did not exist.
20	Enhanced the AlarmWorX32 Server's DA Server interface adding DA tags that return the severity of the highest severity active alarm in an area. The tags "Area_MaxSeverity" and "Recursive_MaxSeverity" were added to Area objects.
21	Enhanced the AlarmWorX32 Server Configurator to log the ActorID (user ID) when posting GenEvents.
22	Resolved an issue with the AlarmWorX32 Viewer's automation interface where the {{Time}} field was not always returning the correct value.
23	Resolved an issue with the AlarmWorX32 Viewer where the alarm didn't flash if the was not visible at the time of alarm.
24	Resolved an issue with the AlarmWorX32 Viewer where it could crash if the help text size is very large (at least 6 lines of text) – larger than what the tooltips could handle.
25	Resolved an issue with the AlarmWorX32 Viewer where the operator comments were not sorted properly.

### AlarmWorX32 MMX Resolved Anomalies

#	Summary
1	Resolved an issue with the Pager Agent where it did not always post Agent Done status.
2	Resolved an issue with the Pager Agent where it could queue one more page request than the maximum it was configured to handle.
3	Resolved an issue with the AlarmWorX32 MMX Server where it would sometimes and unnecessarily force a disconnect and reload of the MMX Pager Agent, and this would sometime result in a crash of the Agent. An option has been added to prevent the unnecessary restarts / forced disconnects. The Registry setting to invoke this option is: AllowShutdown.
4	Resolved an issue with the Phone Agent where sometimes a different voice than the norm can be heard in different menus of the call-in agent.
5	Enhanced the AlarmWorX32 MMX Server so that it can work with multiple processors.
6	Resolved an issue with the AlarmWorX32 MMX Server where it would restart too often - as a result of the way it was cleaning up database errors. Code was added to handle the manual cleanup of failed SQL commands, making the MMX Server more robust.
7	Resolved an issue with the AlarmWorX32 MMX Server where a failed role id was incorrectly flagged as a database access failure and resulting in a reset. The MMX Server now treats this type of failure properly.
8	Resolved an issue with the AlarmWorX32 MMX Server where
9	Resolved an issue with the AlarmWorX32 MMX Server where Alarms set to not require acknowledge would not correctly trigger the stop condition.
	Resolved an issue with the AlarmWorX32 MMX Server where MMX actions were not reported through GenEvents. The following messages are now reported: <ul style="list-style-type: none"> <li>• Agent Done – Passed</li> <li>• Agent Done – Failed</li> <li>• Receipt Confirmation</li> <li>• Agent Cancel – Passed</li> <li>• Agent Cancel – Failed</li> <li>• Invoke Agent – Passed</li> <li>• Invoke Agent – Not Scheduled</li> <li>• Invoke Agent – Failed.</li> </ul>

### Data Mining Anomalies Resolved

#	Summary
1	Resolved an issue with the Data Mining Control where it would not show more than 257 rows.
2	Enhanced the Data Mining OPC Server (DBOPC Server) adding a INI-set-able option to automatically and periodically re-connect to its Data Source.
3	Resolved an issue with the Data Mining OPC Sever (DBOPC Server) where it was not able to read an integer as an input parameter with the data manipulator and stored procedure.
4	Resolved an issue with the Data Mining OPC Sever (DBOPC Server) with writing some values to a database because of the data type being used. Note, using a string type was OK, however, numbers didn't work.
5	Resolved an issue with the Data Mining OPC Sever (DBOPC Server) where the Data Manipulator's operation could time-out.

### DataWorX32 Anomalies Resolved

#	Summary
1	Resolved an issue with DataWorX32 where it was not supporting Unicode (Chinese in this case) characters.
2	Resolved an issue with DataWorX32 where a second client would not get an update in the case where the first client performs many sync writes to non-active tags, and then at some later time, the second client connects and subscribes to those tags.
3	Resolved an issue with DataWorX32 where it was sometimes not returning the quality of "OPC_QUALITY_OUT_OF_SERVICE" for inactive tags.
4	Resolved an issue with DataWorX32 where it could hang-up when multiple clients disconnect at once .
5	Resolved an issue with DataWorX32 where the NT Service Registry Key for the DataWorX32 runtime was not 'DataWorX32' as it should have been.
6	Resolved an issue with DataWorX32 where the runtime could stop giving data (tags would not load, and if refreshed, the entire tree in the browser became blank).
7	Resolved an issue with DataWorX32 where it was unable to go to runtime mode with an existing (pre version 8.04) configuration database.
8	Resolved an issue with DataWorX32 where it was not providing the range information to an OPC client.
9	Resolved an issue with DataWorX32 where the service could use 100% CPU and locks up when making online changes for very large configurations..

### Expression Editor Anomalies Resolved

#	Summary
1	Resolved an issue with the Expression Editor where the XOR no longer worked with negative numbers. An INI file setting was added to enable this fix (interpretation of how the XOR should function): [HKEY_LOCAL_MACHINE\SOFTWARE\ICONICS\Expressions] "UseSignedBitwiseOperators"=dword:00000001
2	Resolved an issue with the Expression Engine where x=tonumberbase('00',2) could cause it to crash.
3	Resolved an issue with the Expression Engine where the Bittest function was not working for negative numbers.

## GenBroker Anomalies Resolved

#	Summary
1	Resolved an issue with GenBroker where it could display a "busy" message asking you to switch to the busy application or to retry.
2	Resolved an issue with GenClient where it would crash when trying to connect to the Kontor HDA ODBC server.
3	Resolved an issue with GenClient that was causing a memory leak in GraphWorX32 when GraphWorX32 was subscribed to a fast changing array tag from the Database OPC Server,
4	Resolved an issue in GenClient's GenEvent support where the GenEvents were not getting forwarded to secondary server.

## GenTray Anomalies Resolved

#	Summary
1	Resolved an issue with GenTray where it did not close applications when GENESIS32 was installed as a browser node.

## GraphWorX32 Anomalies Resolved

#	Summary
1	Resolved an issue with the Switch ActiveX where the Switch label was not properly aligned to center if the label was a Language Alias value.
2	Resolved an issue with GraphWorX32 where the Digital Selector would show all objects when the layer is linked with a bad quality tag.
3	Resolved an issue with GraphWorX32 where it would not seem to get the updated theme if you changed the theme using scripts.
4	Resolved an issue with GraphWorX32 where if you selected the option to open a popup menu at the mouse position, you would see it open at an arbitrary location rather than at the mouse position.
5	Resolved an issue with GraphWorX32 where it could crash after copying and pasting a 16-bit image.
6	Resolved an issue with GraphWorX32 where an AlarmWorX32 DA Tag was not behaving correctly when the Toggle Pick Action was applied to it.
7	Resolved an issue with GraphWorX32 where the Drag and drop did not work due to it removing the OPC Tag ProgID.
8	Resolved an issue with GraphWorX32 where PPTs could fail to update after repetitive executions of scripts overnight.
9	Resolved an issue with GraphWorX32 where the layers remained in Point Failure colors after a re-Connection of communications to the server.
10	Resolved an issue with GraphWorX32 where the Button 'down' didn't release when moving the mouse off of the button. When clicking on buttons, there are 3 configurable cases: <ul style="list-style-type: none"> <li>• The original one: The button will stay down until the mouse button is released.</li> <li>• Windows like: When the mouse pointer leaves the button's area, the button will pop up. After releasing the mouse button, the mouse pointer has to be in the button's area, otherwise OnUp action is not executed.</li> <li>• Requested one: Same like case #2, except OnUp/OnDown events are fired when the pointer leaves/enters button's area. If the button is up when the mouse button is released, no OnUp action is executed.</li> </ul>

	<p>The following registry keys are used to set up requested behaviour (listed with default values):</p> <ul style="list-style-type: none"> <li>• [HKEY_LOCAL_MACHINE\SOFTWARE\Iconics\Gwx32\Compatibility Properties] "HandleMouseMove"=dword:00000000 "FireEventsOnMouseMove"=dword:00000001</li> <li>• To set up specific cases: 1 - HandleMouseMove = 0 2 - HandleMouseMove = 1, FireEventsOnMouseMove = 0 3 - HandleMouseMove = 1, FireEventsOnMouseMove = 1</li> </ul>
11	Resolved an issue with the GraphWorX32 Viewer ActiveX where Global Aliasing was not working after the ActiveX was dragged into Internet Explorer.
12	Enhanced GraphWorX32 adding support for second level aliasing for the LoadDisplay function. This can be implemented by passing the pick ID into LoadDisplay which passes it into a function CreateAliasMapForCurrentDisplay().
13	Resolved an issue with GraphWorX32 where a popup would not show its frame when the Outer Frame Edge was set to Raised [Fixed size].
14	Resolved an issue with the GraphWorX32 Viewer where it could crash when used in a C# .NET container and when zooming and panning.
15	Resolved an issue with GraphWorX32 where it would not show the last known good value for PPTs when the PPTs stop updating as a result of a communication failure. (The v8.02 GraphWorX did show the last known good value).
16	Resolved an issue with the GraphWorX32 Viewer where it did not render properly in WebHMI. In this particular case, some of the static objects were missing from the display.
17	Resolved an issue with the GraphWorX32 viewer when used in WebHMI where it would sometimes crash when the user clicked on the "back" button in Internet Explorer. In this particular case, the application had code that launched a brand new WebHMI page, rather than just loading a display inside the GWXViewer.
18	Resolved an issue with GraphWorX32 where it could crash while opening/closing popups during LoadDisplay. The following registry keys are used to invoke the solution: [HKEY_LOCAL_MACHINE\SOFTWARE\ICONICS\Gwx32\Compatibility Properties] <ul style="list-style-type: none"> <li>• "UseHierarchicalLock"=dword:00000001</li> <li>• "SynchVBAScripts"=dword:00000001</li> </ul> UseHierarchicalLock - enables hierarchical lock during opening/closing displays SynchVBAScripts – only one VBA script is allowed to run at a time
19	Resolved an issue with GraphWorX32 where the MoveObject did "funny" things with Line objects. For example, in Runtime mode, if you hit any of the buttons the appropriate object will move 15 pixels down and 15 pixels to the right. No problem so far. Try hitting the buttons 5 or 10 times. Exit Runtime mode and you will see some of the objects moved whereas others did not.
20	Enhanced the GraphWorX32 Automation Interface adding support for setting the Format and Engineering units of Process Points.
21	Resolved an issue with GraphWorX32 where confirmation popup dialogs would appear under other popup windows, if the other popup windows were given the property "always on top". The confirmation popup windows are now always on top, independent of how the other popup windows are configured.
22	Resolved an issue with GraphWorX32 where it could crash after a VBScript is removed from a Pick Action button.
23	Resolved an issue with GraphWorX32 where the 'SelectPaletteColorRGB' function did not work on a popup display.
24	Resolved an issue with GraphWorX32 where it could crash after a few login / logout cycles for a particular display. It was found that some additional range checking during the resolving of aliases was needed.
25	Resolved an issue with GraphWorX32 where the DisplayBack automation function did not work in the viewer.

26	Resolved an issue with GraphWorX32 where the viewer ActiveX appeared in opposite order when it went into runtime mode. If you had an Alarm viewer on top of a trend viewer in configuration mode, in runtime, you will see the trend viewer on top of the alarm viewer. This is the same for layers. You would expect the main layer (0) to be on the very bottom, then the next layer on top of it. however, in run time you will see layer 1 below the main layer.
27	Resolved an issue with the GraphWorX32 Viewer where the CreateBitmap function did not work over WebHMI.
28	Resolved an issue with GraphWorX32 where the "Display Back" pick action didn't resolve the previous display's Global Aliasing theme.
29	Resolved an issue with the GraphWorX32 Keypad support where GraphWorX32 could crash if the keypad is set (enabled) in the registry and both the real keyboard and the keypad is used to enter data.

### Language Aliasing Anomalies Resolved

#	Summary
1	Resolved an issue with Language Alias Configurator where it could not export the dictionary to an .mdb file.

### Licensing Anomalies Resolved

#	Summary
1	Resolved an issue with GenRegistrar Server where GetDispatch did not work when IsBrowser was set.
2	Enhanced Licensing to add support for the new (SafeNet) Hardware Licensing Key.

### ProjectWorX32 Anomalies Resolved

#	Summary
1	Resolved an issue with ProjectWorX32 where it would generate a number of errors during the reporting process in the case where there is a GraphWorX32 display with an AlarmWiewer which uses a filter. The ReplaceEm() function did not return a string as expected.
2	Resolved an issue with ProjectWorX32 where it could crash when a check-in comment was greater than 64 characters.
3	Resolved an issue with ProjectWorX32 where it could generate a number of unpacking errors. A fix was made that eliminates the ProjInfo.ini file.

### ScriptWorX32 Anomalies Resolved

#	Summary
1	Resolved an issue with ScriptWorX32 where the application action 'Show VBA Editor' did not work.

## Security Anomalies Resolved

#	Summary
1	Resolved an issue with the Security Configurator where it did not support halting an NT Sync after it had been completed once.
2	Resolved an issue with the Security Indicator ActiveX where it was not print-able.
3	Resolved an issue with the Security Login Application where the keypad would grey out if the user entered in an incorrect password.

## TraceWorX32 Anomalies Resolved

#	Summary
1	Resolved an issue with TraceWorX32 where it was not working with Internet Explorer v7.
2	Enhanced the Tracing capabilities within the DBOPC Server.
3	Resolved an issue with TraceWorX32 with the DlgWrapper.dll where it was susceptible to a buffer overflow condition and consequently was not CERT compliant.

## TrendWorX32 Anomalies Resolved

#	Summary
1	Enhanced TrendWorX32 to support logging to SQL Server 2005 and SQL Server 2008. Note, if using SQL Server 2005 or SQL Server 2008, it should be set to SQL Server 2000 Backward compatibility mode.
2	Resolved an issue with the TrendWorX32 Configurator where Chinese characters would get lost when exporting to CSV file.
3	Added a configurable option to the TrendWorX32 Logger (TWXSQLSvr) to have it wait until it gets a good quality evaluation of the expression before it starts data logging it. The option can be turned on via an IcoSetup.INI file setting. It is a DWORD value under HKLM\Icons\TWXSQLSVR\Data with the name IgnoreFirstBadExpression. Setting it to a "1" turns this option on.
4	Resolved an issue with the TrendWorX32 Report where it was impossible to create reports under Windows Vista.
5	Resolved an issue with the TrendWorX32 Report where under certain conditions, the totalizer value output was zero when it should not have been.
6	Resolved an issue with the TrendWorX32 Report where it could give an extra day in a report. Specifically, when running a daily report using the UseLastDate registry entry, it would give two entries. The first entry was the previous day with no data and the second entry has the data. The previous day report was not needed and has been eliminated.
7	Resolved an issue with the TrendWorX32 Viewer where it could cause a crash in GraphWorX32 when using the Stack Plot function.
8	Resolved an issue with the TrendWorX32 Viewer where it did not provide complete printing details when using the TWXViewer1.PrintTrend function.
9	Resolved an issue with the TrendWorX32 Viewer where the method GetSampleInfo returned the same value if the pen descriptions were identical.
10	Resolved an issue with the TrendWorX32 Viewer where the Alarm lines were one pixel below where they should have been.
11	Resolved an issue with the TrendWorX32 Viewer totalizer support where the totalized pen would get negative values when the user changed the trend period in the TrendWorX32

	viewer.
12	Resolved an issue with the TrendWorX32 Viewer where static pens were no longer getting plotted. In the previous version of the TrendWorX32 Viewer, the last point on a plot was no longer connected to the edge of the graph. This meant that static pens, which have only one point, are no longer drawn as lines. If samples were not shown, pens with unchanging values did not get drawn. The new version of the viewer connects the last point on the Viewer to the edge of the graph to ensure static pens are still shown.
13	Resolved an issue with the TrendWorX32 Viewer where the AddHistoryPen automation method didn't always pull in correct number of samples. This was observed when using iHistorian as the HDA Server.
14	Resolved an issue with the TrendWorX32 Viewer where the pen names would appear not assigned and the display scroll bar would disappear.
15	Resolved an issue with the TrendWorX32 Viewer where Global Aliases were not getting resolved for historical tags.
16	Resolved an issue with the TrendWorX32 Viewer where adjusting columns spacing would not be saved.
17	Resolved an issue with the TrendWorX32 Viewer where the Tagbrowser would freeze when dragging tags into the Tag selection area.

### Unified Data Browser Anomalies Resolved

#	Summary
1	Resolved an issue with the Unified Data Browser where the Extended Point Syntax Editor would drop server name, This occurred when the node contained in a tag corresponded to localhost (browsed like it was remote or typed manually).

### WebHMI Anomalies Resolved

#	Summary
1	Resolved an issue with the WebHMI versioning utility where it could corrupt a (custom) HTML file. Checking for non-unicode files was added to address this issue.
2	Resolved an issue with WebHMI where the user was unable to download values into a Global Alias Tag.
3	Resolved an issue with the GraphWorX32 viewer when used in WebHMI where it would sometimes crash when the user clicked on the "back" button in Internet Explorer. In this particular case, the application had code that launched a brand new WebHMI page, rather than just loading a display inside the GWXViewer.