GENESIS64 - Converting GraphWorX32 to GraphWorX64

APPLICATION NOTE

Description: This document describes what content of a GraphWorX32 display will be converted to a GraphWorX64 display using the GraphWorX32-to-GraphWorX64 Display Translator Utility.

OS Requirement: Windows Server 2003 x64/Vista x64/ Server 2008 x64/Windows 7 x64/ Server 2008 R2 x64

General Requirement: GraphWorX64 / GraphWorX64 File Translator.

Introduction

As the x64 machines become more popular, you may need to upgrade your displays, originally created in GraphWorX32, to GraphWorX64 files. ICONICS has created a Display Translator to help you through this process.

The Display Translator can import and translate a single GraphWorX32 display or multiple displays. A log file is also generated displaying information about the translation process. The GraphWorX64 Display Translator is a utility included in GraphWorX64 that enables you to convert the following types of files to the GraphWorX64 file format:

- GDF: GraphWorX Displays
- TDF: GraphWorX Templates
- SDF: GraphWorX Symbol Categories

Translating Files

If you are interested in converting multiple GraphWorX32 displays to GraphWorX64 format. Then use GraphWorX32-to-GraphWorX64 file Translator.

1. Go to Start → All Programs → ICONICS → GENESIS64
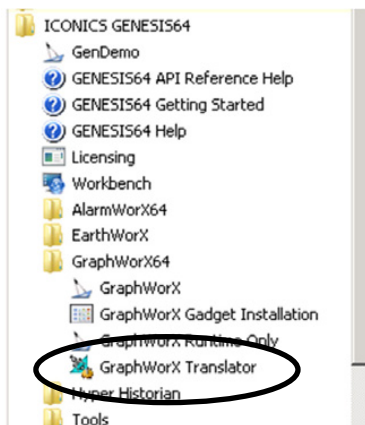2. Open GraphWorX64 → GraphWorX Translator

Figure 1 - GraphWorX64 Translator in Start Menu
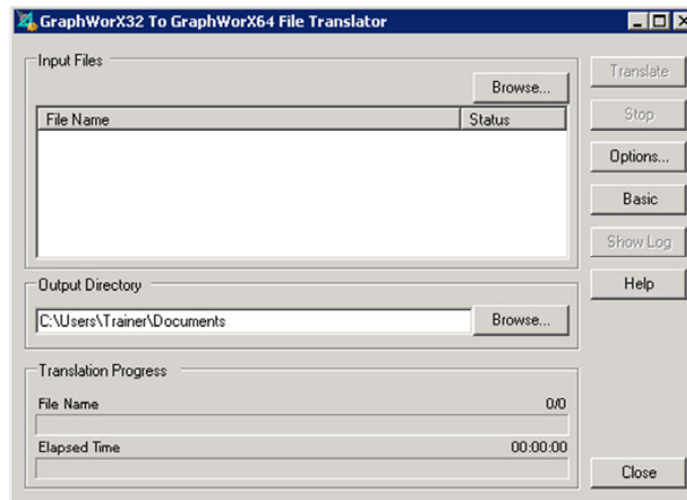
3. Click on Browse to Input GraphWorx32 files.

Figure 2 - GraphWorX32 to GraphWorX64 File Translator

4. Click on Browse… button and browse for the project directory where GraphWorX32 files are located. Select all files and press OK.
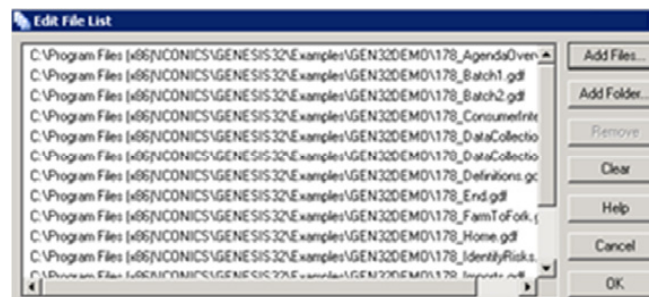

Figure 3 - Adding GraphWorX32 Displays to the Translator

5. Select Output Directory for translated files and then press the "Translate" button to finish translation.
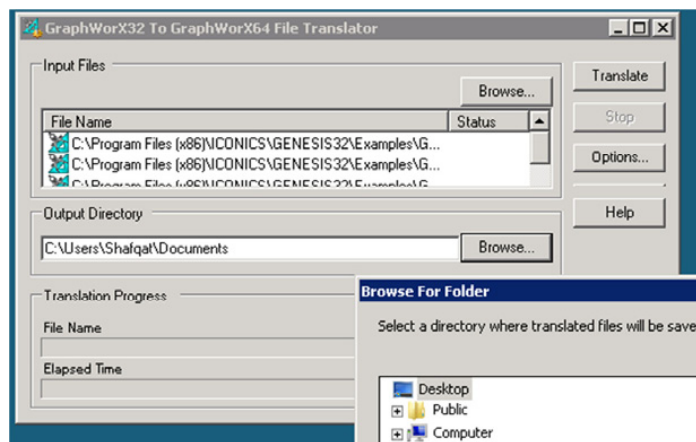

Figure 4 - Choosing Output Directory

6. During translation, the Status will be displayed. Possible status messages include: **Queued; Translating; Incomplete; Failed; Aborted by User or Succeeded:** Files were completely (100%) translated.

NOTE: A detail conversion log will be generated and can be viewed by pressing the "Show Log" button.

Translator Options

The Translator utility allows you to control how the translation is processed by offering you a set of options. Among the options you will find are those in the following categories:
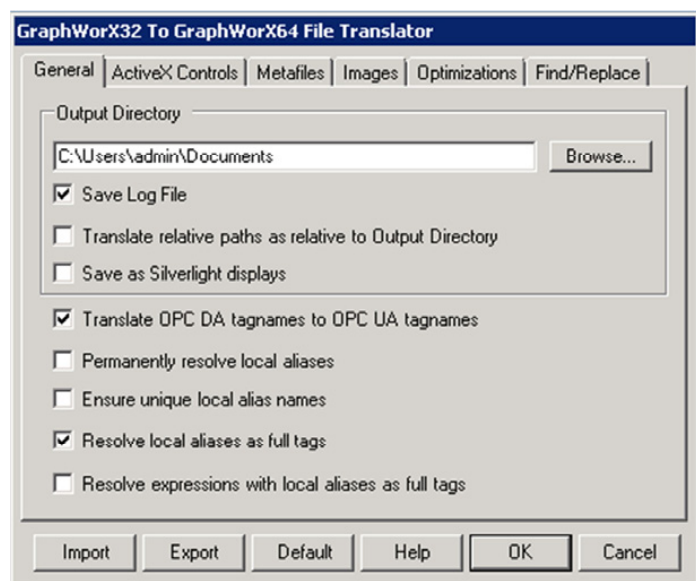

Figure 5 - Translator Options

General Tab

The following functions are accessed through the General tab

- **Output Directory**. This is the directory the translated file is saved to.
- **Save Log File**. When enabled Save Log File writes an XML log file and stores it in the output directory.
- **Translate relative paths as relative to the output directory**. This option is useful when you translate a display with multiple dependencies.
- **Translate OPC DA tag names to OPC UA tag names**. This function translates older OPC Data Access sources into the newer style OPC-Unified Access format that GraphWorX64 supports.
- **Permanently Resolve Local Aliases**. This option is used to translate local aliases. When enabled, tag names in a display will be permanently resolved using the local alias settings in the display. The translated

display would no longer contain any local aliases, only the results of resolving the aliases from the source display. When disabled, tag names will remain unmodified, and local alias definitions will be transferred to the translated display's alias table.
- **Ensure unique local alias names**. This option automatically renumbers any duplicate aliases that it finds in any of a set of dependent multiple display files so that all local aliases are unique.
- **Resolve local aliases as full tags.** This option would be selected if the entire tag is a local alias. The translator does not know when an alias should be treated as a full tag (with the @ and .value), or when it is a partial tag. ActiveX Controls Tab
- **Resolve expressions with local aliases as full tags.** This option would be selected if an entire expression should be treated as a full tag.

Many GraphWorX32 displays contain different types of ActiveX control. You must specify the ActiveX conversion format in the ActiveX Controls tab.

Convert ActiveX Controls to .NET Controls

Upon display file translation, ActiveX controls in the GraphWorX64 display file will be converted to .NET

Convert ActiveX Controls to Placeholders

Upon display file translation, ActiveX controls in the GraphWorX64 display file will be represented as placeholders (text boxes) in the resulting GraphWorX64 display file, which will contain a text box for each ActiveX control.

Images Options Tab

Many GraphWorX32 displays contain different types of image files (e.g., BMP, JPG, etc.). You must specify the image conversion format in the Images tab.

GraphWorX64 displays support the following image file types:
- PNG
- JPG
- GIF
- BMP

Under Convert Images to Format, select one of these image file types from the drop-down list. All four of these formats display correctly inside a browser window

Find / Replace Tab

The Find/Replace tab allows you to perform substitutions after the translation has been completed. It is useful when you need to change the display name, file name, or other items that are part of the display during the translation. For example, if you had a

display titled "Acme Display" and the company was renamed to the New Acme Corporation, you could perform a find and replace on the string "Acme" to change it to "New Acme." Find/Replace is also valuable for altering the data source for the new display and automatically changing the path tag so that it points to the correct Dynamic Data Source.

> **NOTE:** There are other Tabs like Metafiles and Optimizations, which we will not mention here.

Supported Features for Conversion

Static Objects: Common Properties

- Position/Size of Object (Dimensions)
- Rotation Angle - Filled/Unfilled - Fill Color
- Gradient Fill – (Some gradient features cannot be supported and will be converted to the nearest possible WPF approximation.)
- Freeze/Lock Object
- Line Color - Width - Style
- 3D Edge / Shadow/Shadow Color
- Name - Description - Share Keyword - Custom Data

Rectangles

- Rounded Corners

Ellipses

All ellipse properties are included in the "Common Properties" above.

Lines /Polylines

- Single Segment Lines / Multiple Vertex Lines

Text

- Font
- Stretch/Don't Stretch Text on Resize –(some minor differences may be evident)
- Text Alignment - Left - Center –Right.
- Bounds Alignment
- Arc Type - Arc - Pie - Chord
- Start Angle/End Angle

Bitmaps

- Image Type
    - BMP
    - JPG
    - GIF
    - PNG
    - ICO
    - TIF

Metafiles

Metafiles will be translated into the new display as native XAML symbols or Bitmaps (Raster graphics images) rather than as "metafile" objects.

Groups/Symbols

Groups of objects (symbols) in a display will be translated.

Layers

Layers will be translated

Dynamic Objects: Common Properties

- Data Source
- Range Override
- Name - Description - Custom Data
- Global Alias

Location / Rotation / Hide

- Location Path
- Slider
- Start Angle - End Angle
- Clockwise/Counter-Clockwise
- Dial
- Hide or Disable
- Hide when True/False

Digital Color

- Multiple Data Source connections
- Apply Color Changes
  Shadow Color – (No for "high quality" shadows)
- Change Color on True/False

Analog Color

- Apply to Fill Color
- Apply to Line Color
- Apply To Shadow Color – (No for "high quality" shadows)
- Default Color When

> **NOTE:** the GraphWorX.NET Color Dynamic encompasses the functionality of both Digital Color Dynamics and Analog Color Dynamics.

Flash

Note: there is no Flash Dynamic type in GraphWorX.NET; all flash functionality is incorporated into the Hide and Color Dynamics.

- Hide Object
- Change Color
- Flash Rate
- Flash when True/False
- Show Alternate State when not flashing

Digital Selector

- Multiple Data Source connections
- Show Object on True/False

Analog Selector

- Frame Percentage Ranges
- Hide Below Range/Hide Above Range

NOTE: in GraphWorX.NET the functionality of the Analog Selector Dynamic and the Animator Dynamic is consolidated into a single "Range Selector" Dynamic Type.

Animator

- Frame Rate
- Animate When True/False
- Visible/Invisible when Off
- Current Frame When Off/First Frame When Off

Process Point

- Data Entry - Confirm -Format – Initial Value
- Keypad
- Client Display Type
- OPC Request Type
- Update
- Thousands Separators – Leading Zeros
- Show Unit – States

Time / Date

- Display - Time Only
- Date Only
- Time and Date
- Time Format – Date Format

Common Pick Dynamic Properties

- Mouse Button - Type– Force Initial Down State

- Value Tracking
- Execution Trigger - On Down - While Down - While Down Interval - On Up
- Shortcut Key

  (Some key combinations might not be available as shortcut keys in the .NET)

Pick Commands

- Load Display – Filename – Set Aliases.
- Popup Window – Override Window Properties
- Window Properties – (see "Display Properties/Window Properties" Section)
- Modal Popup - Center to Parent.
- Embedded Window
- Override Window Properties
- Center to Parent
- Display Back
- Display Forward.
- Close Window
- Launch Application / Document – Filename.
- Launch Web Page
- Download Value - Toggle Value
- Popup Menu
- Set Local Aliases – **Yes** (display-level only, not object-level)

NOTE: GraphWorX64 uses an entirely new system for integrating custom commands more seamlessly into the product, and this new system is not compatible with GraphWorX32's custom commands).

Display Properties - General

- Background Color – Gradient
- Fixed-Scale/Scalable
- Preserve Aspect Ratio
- Display Dimensions (Width/Height)

Window

Window properties are converted, though there might be some slight variations for a few of the properties.

Runtime

- Runtime Focus Highlights - Show Hand Cursor - Focus Rectangle
- Disabled Object Colors

- Disable Input Controls - Read only Data –Security Write Denied
- Maximum Update Rate
- Minimum/Maximum Zoom

Runtime Advanced

- Global Alias Start Themes
- Enable Runtime Zoom and Pan

ToolTips

- Tooltips for Dynamics
  Object Description – Data Source Name - Data Source Value
- Tooltips for Statics - Object Description

Template Files (.TDF)

Template files (.TDF) will be translated with the same conversion compatibility that is available for normal display files (.GDF).

Applied Templates

The GraphWorX32 template layer will be translated.

> **NOTE:** This object type is not implemented in GraphWorX64 and thus is not translated.

Features NOT SUPPORTED for Conversion

ActiveX Controls

ActiveX(s) can optionally be translated, or a placeholder can be inserted into the translated display instead. If an ActiveX is translated, a 64bit equivalent for the legacy 32bit ActiveX must exist. So, in theory, translation of ActiveX(s) is supported, but in practice this almost never works because 64bit ActiveX(s) are so rare.

Embedded OLE Documents

It is not be possible to translate embedded OLE Documents (Word, Excel, etc.) since the concept of embedded OLE documents is not part of the .NET Framework.

Grid

Grid properties in GraphWorX64 are not stored in the display rather they are saved as application preferences. Consequently, Grid properties are not included in the translation.

Configuration Mode Password

Since the GraphWorX64 file format is XAML, which is an open text/XML format, protecting the display from modifications via a configuration mode password is inherently not possible; hence configuration mode passwords are not translated.

Template Objects

This object type is not implemented in GraphWorX64 and thus is not translated.

Runtime and Print Options

- Print Options – Print options in GraphWorX64 are not stored in the display. Consequently, print options are not included in the translation.
- Display Transparency
- Fill Color – Line Color – color and style is determined by the OS

ToolTips

- Object Name – Tooltips per object type

Pick Commands

- Local Alias Dialog
- Custom Command

Pick Commands

- Drag/Drop Load
- Run Script
- Fill Pattern