

GoDiagram Differences, Windows Forms and Web Forms

Copyright © 2002-2008 Northwoods Software Corporation

GoDiagram™ Win for Microsoft® .NET Windows Forms is a .NET class library containing a set of Windows Forms controls for easily building interactive diagrams in stand-alone .NET-based executable applications.

GoDiagram™ Web for Microsoft® ASP.NET Web Forms is a .NET class library containing a set of Web Forms controls for easily building interactive diagrams in ASP.NET-based web applications.

The two class libraries share much of the design and implementation with each other. The User Guide provides details about GoDiagram, most of which apply to both products. You will need to read that document before you can really make good use of either GoDiagram product.

The goal has been to try to minimize the differences between the two platforms. This allows for maximal code sharing (probably using **#if**) for your **GoDocument** and **GoObject** code.

General Differences

The principal difference between the two products is that GoDiagram Web depends on **System.Web.UI.*** whereas GoDiagram Win depends on **System.Windows.Forms**. Thus the **GoView** class for Web Forms inherits from **System.Web.UI.WebControl**, whereas the **GoView** class for Windows Forms inherits from **System.Windows.Forms.Control**. Both assemblies depend on **System.Drawing.***. Of course the Forms you create will have completely different implementations anyway, so the differences between **Controls** is not the biggest issue when considering porting from one platform to the other one.

The other GoDiagram classes are almost entirely identical, so much of your source code can be transferred between Windows Forms and Web Forms projects with minimal change. You just need to refer to the `Northwoods.GoWeb.dll` assembly instead of the `Northwoods.Go.dll` assembly, and use the **Northwoods.GoWeb** namespace instead of the **Northwoods.Go** namespace.

The **Northwoods.Go.Draw** functionality is not yet supported for GoDiagram Web/ASP.NET.

In general the differences are due to types and concepts that do not apply in an ASP.NET application:

- Uses of **System.Windows.Forms** types: **Cursor**, **ImageList**, **ToolTip**, **ScrollBar**, **MouseEventArgs**, **DragEventArgs**, **KeyEventArgs**
- Notions of in-place editing, mouse-over, hovering, and Windows drag-and-drop and printing

You'll note that GoDiagram provides platform-independent support for cursors and tooltips (by **GoObject.GetCursorName** and **GoObject.GetToolTip**) and scrolling (the **GoView** methods: **ScrollLine**, **ScrollPage**, **CanScroll**, and the UI automatically provided by scroll margins instead of scroll bars). Referring to **MouseEventArgs**, **DragEventArgs**, **KeyEventArgs** is rare in a GoDiagram Win application.

GoDiagram Web offers an implementation of a subset of the **ContextMenu** and **MenuItem** functionality that exists in Windows Forms. So it is often possible to have the same context-menu code for both Win and Web.

The main porting difficulties typically lie with the use of **ImageLists** and the limitations of not-so-interactive images in web pages, i.e., no mouse-over or hover or drag-and-drop between windows or interactive dragging or resizing or snapping. The keyboard is not normally used in web applications, either.

Differences Between the Assemblies

For those of you who are converting a GoDiagram Windows Forms application into a Web Forms application, here is a detailed list of differences. (Caution: this list might not be complete or accurate.)

- Renamed namespace: **Northwoods.GoWeb** instead of **Northwoods.Go**
- **GoView** changes:
 - Inherits from **System.Web.UI.WebControl** instead of **System.Windows.Forms.Control**
 - Implements **ISerializable** (**GetDataObject** method and deserialization constructors for **GoView** or derived class)
 - Events added: **SessionStarted**, **Updated**
 - Events removed: **ObjectHover**, **BackgroundHover**, **ObjectEnterLeave**, **ObjectEdited**, **ExternalObjectsDropped**, **SelectionDropReject**, **SelectionDropped**
 - Methods added:
 - **OnLoad**, **CreateView**, **LoadView**
 - **LoadViewState**, **SaveViewState**
 - **GetSessionViewsTable**, **MakeSessionViewID**, **FindSessionView**, **StoreSessionView**, **ClearSessionView**, **OnSessionStarted**
 - **OnUpdated**, **OnNoPostLoad**
 - **GetBitmap**
 - **Render**, **PreRenderDefinitions**, **RenderScript**, **RenderCss**, **SafeID**, **CreateDataRenderer**
 - **RaisePostBackEvent**, **HandleScroll**
 - Method changed: **Dispose**
 - Methods not used: **UpdateView**, **BeginUpdate**, **EndUpdate**, **Invalidate**, **RaiseChanged**
 - Methods removed:
 - **LayoutScrollBars**, **UpdateScrollBars**, **DoAutoScroll**, **HandleScroll**, **ComputeAutoScrollDocPosition**, **DoAutoPan**, **StopAutoScroll**
 - **DrawXorLine**, **DrawXorRectangle**, **DrawXorBox**
 - **OnKeyDown**, **OnMouseDown**, **OnMouseMove**, **OnMouseUp**, **OnDoubleClick**, **OnMouseWheel**
 - **OnMouseLeave**, **DoMouseOver**, **DoBackgroundMouseOver**, **DoToolTipObject**
 - **DoMouseHover**, **DoHover**, **OnObjectHover**, **RaiseObjectHover**, **OnBackgroundHover**, **RaiseBackgroundHover**, **DetectHover**
 - **OnObjectEnterLeave**, **RaiseObjectEnterLeave**, **DoObjectEnterLeave**

- **OnObjectSelectionDropReject, RaiseObjectSelectionDropReject, OnBackgroundSelectionDropReject, RaiseBackgroundSelectionDropReject, DoSelectionDropReject**
 - **OnObjectSelectionDropped, RaiseObjectSelectionDropped, OnBackgroundSelectionDropped, RaiseBackgroundSelectionDropped, DoSelectionDropped**
 - **OnSizeChanged, OnCreateControl, OnPaint, OnGotFocus, OnLostFocus, OnBackColorChanged, OnBackgroundImageChanged, OnStyleChanged, OnSystemColorsChanged**
 - **OnDragOver, DoInternalDrag, DoExternalDrag, GetExternalDragImage, OnDragEnter, OnDragLeave, OnQueryContinueDrag, OnDragDrop, DoInternalDrop, DoExternalDrop, OnExternalObjectsDropped, RaiseExternalObjectsDropped**
 - **Print, PrintPreview, PrintDocumentPage, PrintView, PrintDecoration, PrintShowDialog, PrintPreviewShowDialog**
 - **DoEndEdit, OnObjectEdited, RaiseObjectEdited**
- Properties changed: **DisplayRectangle, IsPrinting, Enabled**
- Properties added: **ScrollMarginToolTip, ShowsVerticalScrollButtons, ShowsHorizontalScrollButtons, BackgroundImage, Alt, ToolTipText, CursorName, ImagePage, ImageFormat, NoPost, ScriptFile, CssFile, DataRenderer**
- Properties removed:
 - **TopBar, RightBar, BottomBar, LeftBar, TopLeftCorner, TopRightCorner, BottomRightCorner, BottomLeftCorner**
 - **VerticalScrollBar, HorizontalScrollBar, CornerControl, ShowVerticalScrollBar, ShowHorizontalScrollBar, AutoScrollRegion, AutoScrollTime, AutoScrollDelay, AutoPanRegion**
 - **ImageList, BorderStyle, Border3Dstyle**
 - **HoverDelay, ToolTip, DefaultCursor, Cursor**
 - **AllowDragOut**
 - **PrintDocumentSize, PrintDocumentTopLeft, PrintScale, PrintsViewObjects**
 - **EditControl, IsEditing**
- Added enums: **MouseButtons, Orientation, SortOrder** (these enums were defined to have the same name and enumeration members as in **Windows.Forms** for source-compatibility convenience between Windows Forms and Web Forms)
- Removed enum: **GoViewScrollBarVisibility**
- Added classes **GoWebImage, GoWebImageHandler, GoViewDataRenderer, GoPrintView**
- Implemented classes **ContextMenu, GoContextMenu, MenuItem, GoMenuItem**
- **GoInputEventArgs** properties removed: **MouseEventArgs, DragEventArgs, KeyEventArgs**
- **GoToolDragging**: does not initiate any Windows drag-and-drop
- **GoToolPanning** operates with a single middle-mouse-down-drag-up gesture rather than with mouse-click-move-click, as a manual pan rather than autopanning
- **GoObject** methods not used: **OnMouseOver, OnHover, DoBeginEdit, DoEndEdit, CreateEditor**

- **GoDocument.SerializesUndoManager** is true by default, instead of false for GoDiagram Win
- **GoObject** properties not used: **Editor**
- **GoControl** is not used
- **GoImage** properties removed: **ImageList, Index, DefaultImageList**
- **GoIconicNode, GoSimpleNode, GoGeneralNode** methods removed: **Initialize** and **CreateIcon** overloads that use **ImageList**
- **GoButton** does not autorepeat when **GoButton.AutoRepeating** is true

Differences in the Sample Applications

The sample applications are not precisely the same, of course. Besides the differences inherent in the GoDiagram assemblies, many of the features used by the samples that Windows Forms offers have not been reimplemented on Web Forms. This includes features such as:

- MDI child windows
- Property Grid usage
- TreeViews
- Menus (regular menu bars, but many context menus do work in GoDiagram Web)
- Toolbars

Furthermore, some sample applications are not present in Web Forms. For ProtoApp, the functionality that is available for Web Forms is pretty much included in the Demo1 sample application. MovableLinkApp and InteractiveForce do not translate well to the web due to the need for greater real-time feedback during dragging than Web Forms can support.