Architecture

Dynamic handling for showing scroll bars

A dirty boolean is handled in GraphicsList, the canvas document, where it should be. To this Dirtyproperty is associated a DocumentDirtyObserver interface.

Hide   Copy Code

public interface DocumentDirtyObserver

{

void IsDirty(GraphicsList gList);

}

Every time the document is modified, look with the different draw tools, its Dirty property is set to trueand any DocumentDirtyObserver is fired.

Hide   Copy Code

public bool Dirty

{

get

{

return this.dirty;

}

set

{

this.dirty = value;

if (this.dirty)

{

NotifyDirty();

}

}

}

The class DrawArea implements the DocumentDirtyObserver, and every time the document is 'dirty', it calls AdjustRendering, which in turn computes the bounding box of the canvas document (GraphicsList).

Hide   Copy Code

void AdjustRendering()

{

Size docSize;

if (this.GraphicsList != null)

{

docSize = this.GraphicsList.GetSize();

docSize.Width += 20;

docSize.Height += 20;

AutoScrollMinSize = docSize;

}

else

{

AutoScrollMinSize = new Size(0, 0);

}

Invalidate();

}

This way, it implements the Mode/View/Controller design pattern, the GraphicsList being the model,DrawArea being the View and the draw Tools being the controller.

In the future, Dirty property should be handled in only one place (it is also handled in the DrawAreaclass).

Export the graph to JPEG format

The DrawTools 2014 architecture is good enough to be followed to implement a new menu strip option.

1. Add a menu item to the menu bar

Simply open the MainForm designer and add the option

1. Link the MenuItem to a MainForm method to trigger the CommandExportToJpg

Hide   Copy Code

private void CommandExportToJpg()

{

docManager.ExportToJpg();

}

1. Implement the user interface logic in the DocManager ExportToJpg

I let you look at the DocManager.ExportToJpg

1. Subscribe MainForm ExportEvent method implementation to DocManager event.

Hide   Copy Code

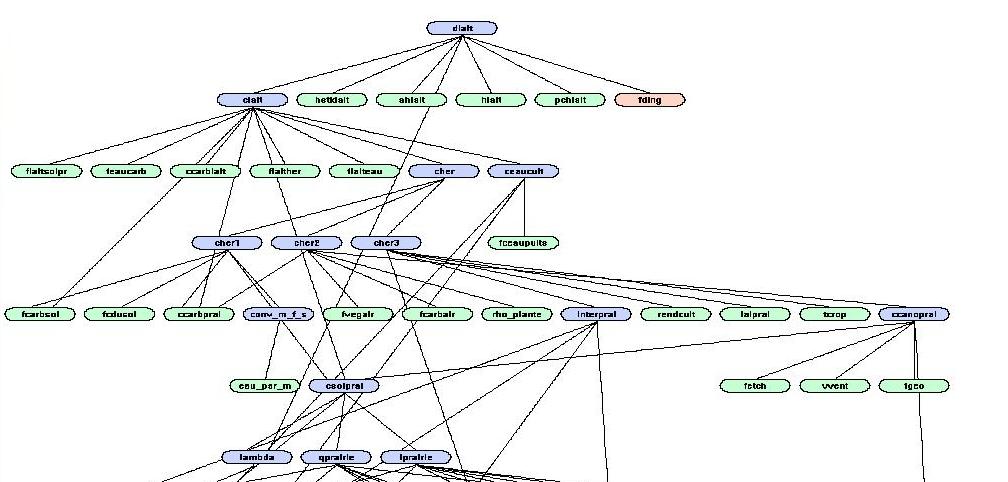
docManager.ExportEvent += docManager\_ExportEvent;

1. Implement MainForm logic to actually transform image bitmap into JPEG file.

All is in the source code, I don't copy it here as it is essentially technical stuff.

Result

The result is best shown by the following graph, obtained with a customized version of the DrawTools.

[](http://www.codeproject.com/KB/cs/711081/result_graph.jpg)

Points of Interest

Architecture is already well thought in the original project, please read the original document for insight about it.

* [DrawTools 2005](http://www.codeproject.com/Articles/8494/DrawTools)

There was a bug where the status bar was hiding the horizontal scroll bar, but after some thorough inspection of the code, it has been fixed.

* [Issue on Stackoverflow](http://stackoverflow.com/questions/21076160/status-bar-hides-usercontrol-scroll-bars)

History

1. Added scroll bars to the draw area
2. Added a status bar
3. Fixed a bug on the context menu when the scroll bar position is not 0
4. Export of the drawing with jpeg format
5. Fixed an OutOfMemoryException with JPEG format export

Special thank you to Alex, who originally posted the **DrawTools** project on CodeProject.

License

This article, along with any associated source code and files, is licensed under [The Code Project Open License (CPOL)](http://www.codeproject.com/info/cpol10.aspx)