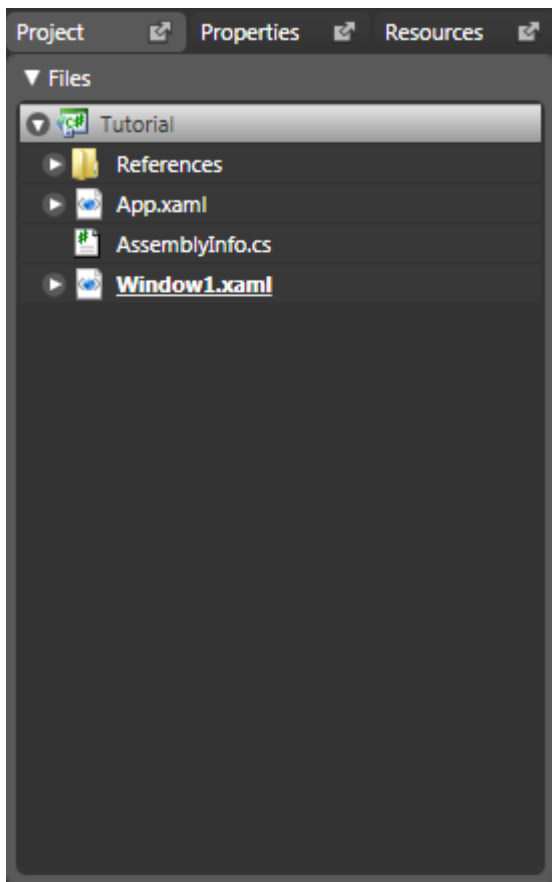




With **Lloyd Humphreys**
(<http://www.lloydhumphreys.com>)

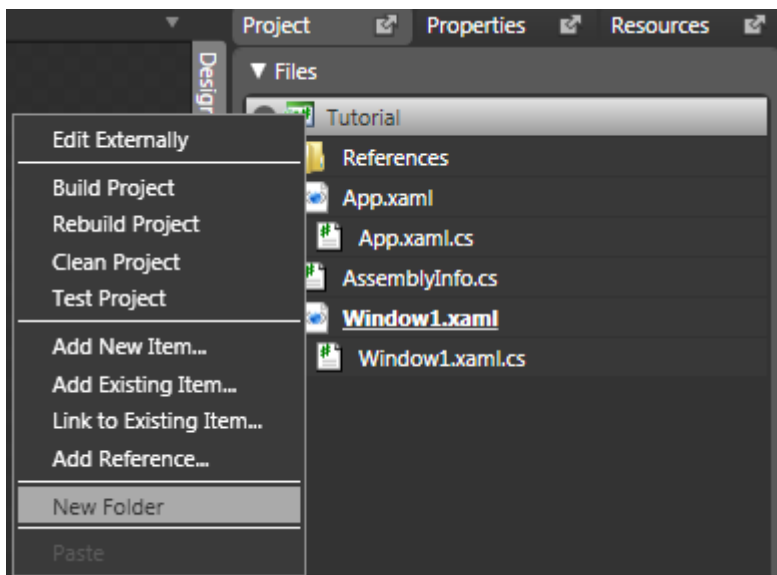
In this first edition of "Expression Blend", I'm going to show you just how Expression Blends file structure works, and how you can edit and manage resources.



Overview

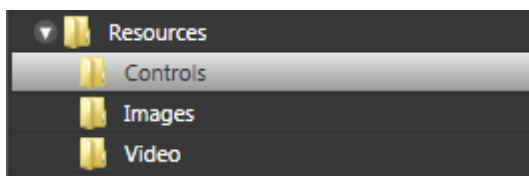
This is the project window. As you can see, the file structure is identical to that of Visual Studio's. The name of this application is "Tutorial", and inside of the solution are numerous files:

- **References** - a list of DLLs (Dynamic Link Libraries) used in the project. As you expand functionality, you expand the list of referenced DLLs.
- **App.xaml** – This is your global application file. Everything here is relevant to your application. It's not editable in the design view, but switching to the XAML view (there's a tab at the side) allows you to make changes. However, editing the XAML in Visual Studio is far superior than editing it in Blend.
- **AssemblyInfo.cs** – This is a file you won't need to touch. It's simply the assembly information that Blend and Visual Studio use. You'll probably not need to touch it.
- **Window1.xaml** – The XAML file of one of your windows. Unless you rename it, this will be called "Window1.xaml". It's the XAML code for your current window. Expanding this further shows us the "Window1.xaml.cs" file – a CSharp file, which can't be edited in Expression Blend – EB isn't designed for it. This file is opened in Visual Studio; and is edited there.



Right clicking on Tutorial allows you to create new folders. This allows you to better organise your project.

You can also add items from this list, if you're going to be messy – otherwise you keep all of your items inside the Resources folder.



It's general practice to make a Resources folder, to keep all of your used items in. For example here, we've got "Controls" for any custom controls you may make, "Images" for any images you use in the project, and "Video" for any Video you might include.

External Media

Adding media to your project couldn't be easier. You can simply add it to the folder created when you create your project, or you can add it directly through Blend. Right click on the folder you want to add content to, then click "Add Existing Media". Navigate to the file you want to add, and then when you click OK, Blend will copy that file into your selected folder. There is another way of adding files, in the form of "Linking". Simply right click and select "Link to Existing Media". This doesn't copy the file you select into a folder, and simply uses the original source. Of course, this has its cons. If you move the file, or deploy your application to a computer which doesn't have the same file, in the same place, then the link will break and your application will not execute, or will break.

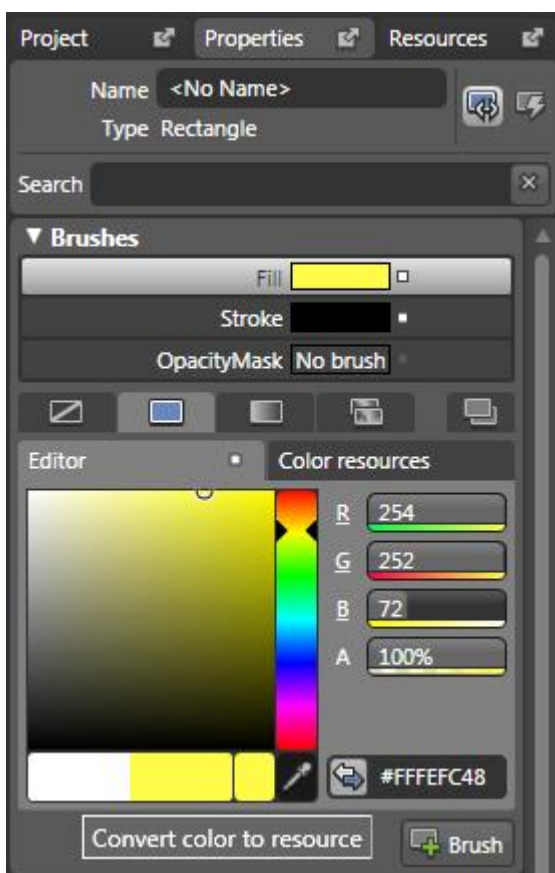
Creating/Using Resources

Creating and using Resources is simple. First, we'll go over the concept of a "Resource". In Blend, resources can be anything – colours, shades, images – *anything*. Generally, Resources save you the time of having to do things over and over again, and I'll show you how to create and use them now.

Creating and Using Colour Resources

Say I create a Square, and I want to colour it a certain shade of yellow. I head over to the Properties panel, and select the colour. But what if I want an Ellipse, with the same colour? I've now got to go back to the Square, grab the hex code, back to the Ellipse and then insert the hex code. Creating a

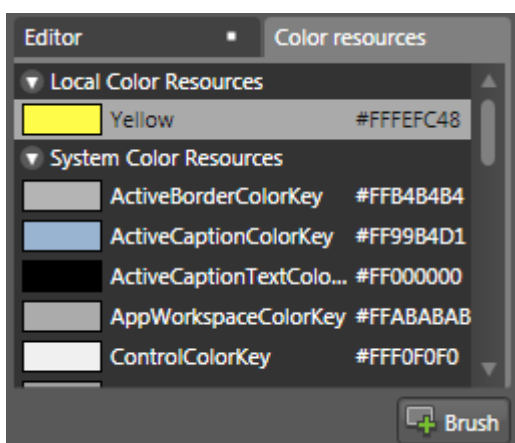
resource makes this a lot easier.



Here, we've got a Rectangle. I've selected the colour I want – a pukey shade of Yellow. To create a Resource, I simply click the two arrows next to the hex code. I'm hovering over it in this picture, and the "Convert colour to resource" tooltip pops up.

After you click this, a window will pop up. Give the colour a name, and where it says "Resource Dictionary", click New. I called mine Colours.xaml.

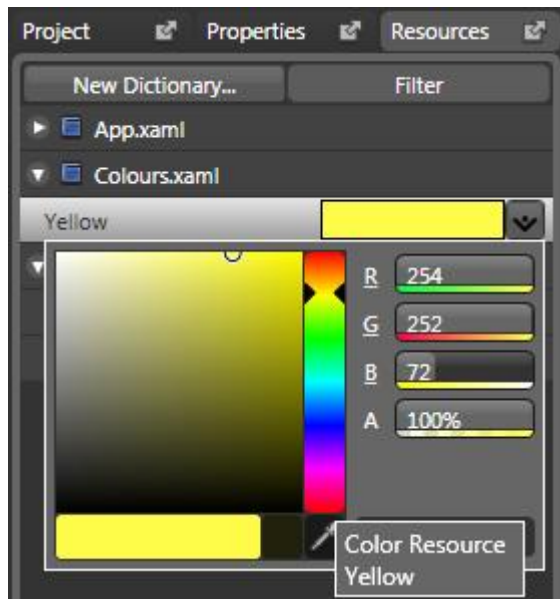
Click OK, and the colour will be renamed to Yellow, and placed in the "Color Resources" tab at the top. It'll open automatically.



This is where the colours you convert to Resources are stored. Mine is called "Yellow" and is placed under "Local Color Resources".

Now, these resources are similar to CSS stylesheets. They're used globally inside your application, and should you want to change that shade a little, you can

do so easily – and any object with the colour applied will also change colour.
Clever, eh?

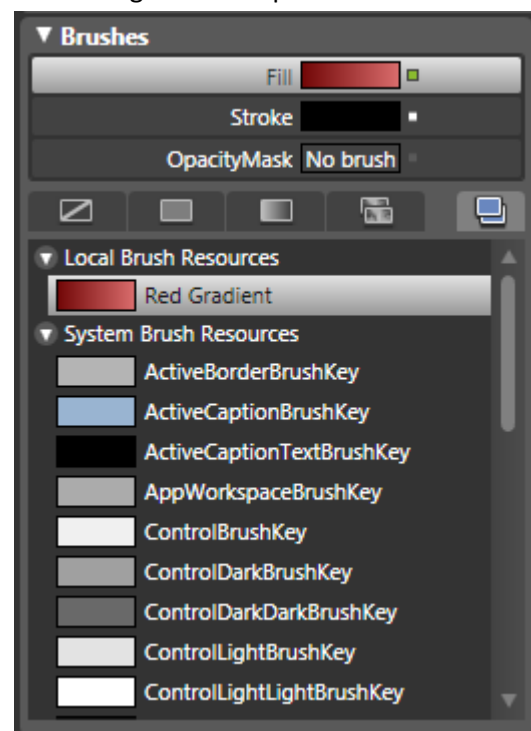


To change the colour, click the “Resources” tab at the very top. Unfold your Resource Library (in my case it’s called “Colours.xml”) and select the colour you want to edit. Changing this colour here will change the colour of “Yellow”, and since my Rectangle has “Yellow” applied to it, the colour of the Rectangle will change.

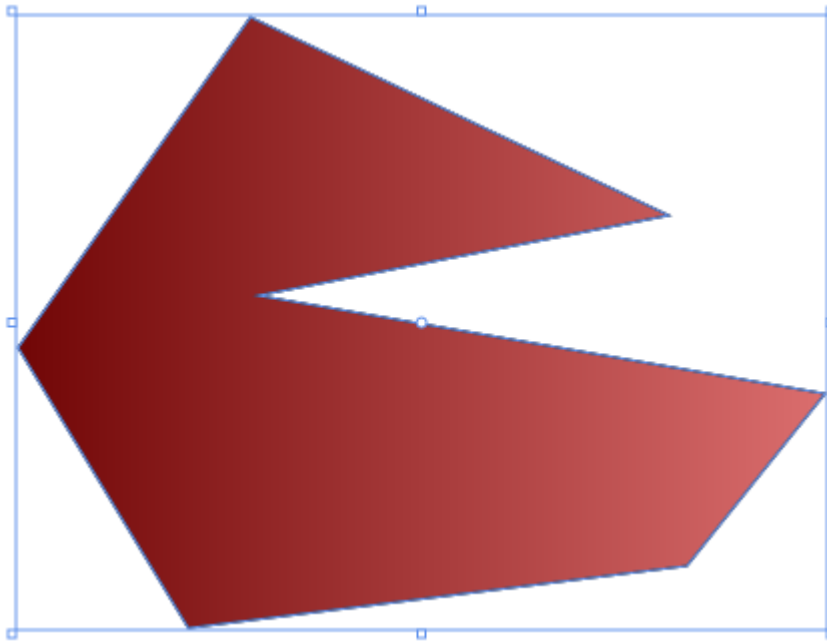
Brushes

Brushes are different from Colour Resources. Brushes are gradients or images - for example, a gradient in Blend is composed of two colours. Clicking “Convert to Color Resource” would simply convert the colour you’re currently altering to a Resource. Creating a Brush captures the whole setting for that Gradient. This time I’ve gone for a more subtle red gradient, and if I click the “+Brush” button at the bottom, another dialog pops up. Save it as we did with the Colour Resource, in a custom library. Again, I’ve saved mine in my “Colours.xml” library.

If we now click on the “Brush Resources” button (the button isn’t very clear in this image, it’s the lighter grey to the right, about a quarter of the way down) in the “Brushes” section of Blend, we can see the gradient under the “Local Brush Resources”. Again, we can apply this to anything – Ellipses, Squares, Triangles....

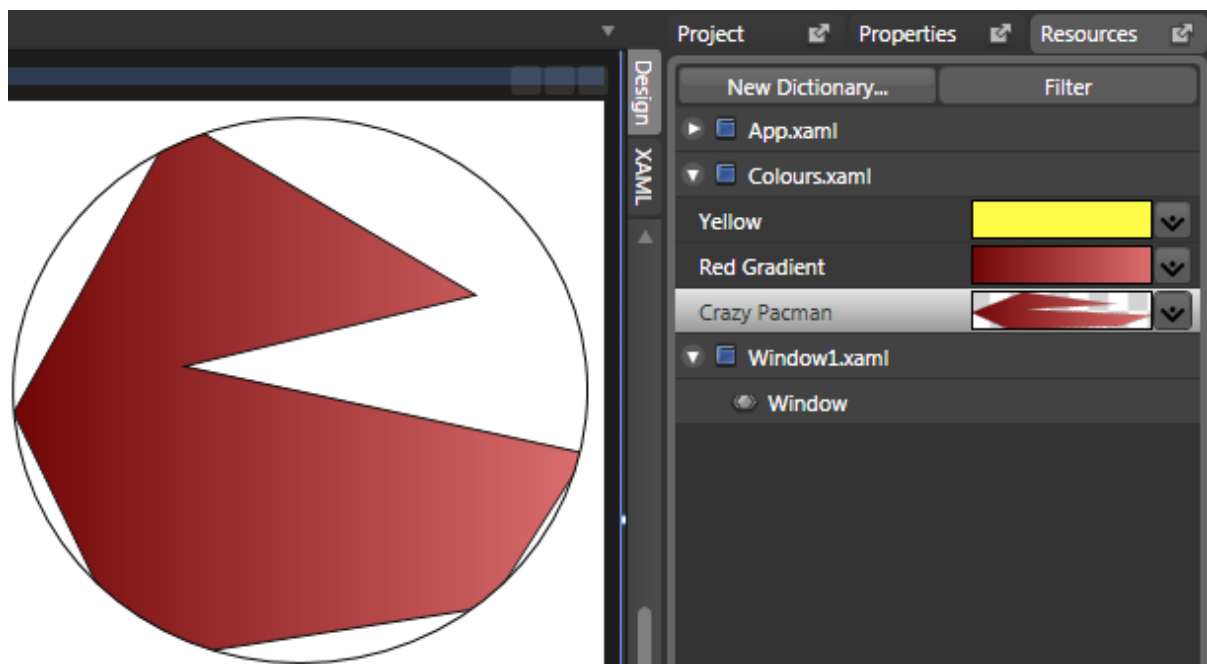


Another type of Brush is an image – for example, I can select my Crazy Pac-man and turn him into a Brush – so I can use him over and over.



This is a simple shape made by the pen tool. If I click on Tools, and then on “Make Brush Resource” while my Crazy Pac-man is selected, I can then click “Make Drawing Brush Resource”. If I save it in the same library, then head up to the “Resources” tab again, a small preview of the item I’ve made is there. Then, to use it, say I created a Square, I could click this Brush, and the square

would be filled with this image – because it’s now a Brush.



Now that the ellipse has a “Crazy Pacman” fill, it looks like this. All the Resources I’ve created are saved in my “Colours.xaml” Library (or Dictionary, as Blend calls them, I feel library is a more fitting name).