Accessing GDAL data provider in DotSpatial

**Tutorial (1)**

Purpose of this tutorial: Become familiar with the access of GDAL data provider in DotSpatial.

**Step 1**: Download the DotSpatial class library and create the DotSpatial tools.

Download the DotSpatial class library.

Add the DotSpatial tools in the Visual Studio toolbox. See the first tutorial if you do not know how to add the controls.

**Step 2:** Add DotSpatial references and an assembly reference.

Add the following DotSpatial references:

DotSpatial.Data.Forms.dll, DotSpatial.Symbology.dll, DotSpatial.Controls.dll, DotSpatial.Projections.dll, DotSpatial.Data.dll, DotSpatial.Topology.dll

Add the following assembly reference:

System.ComponentModel.Composition

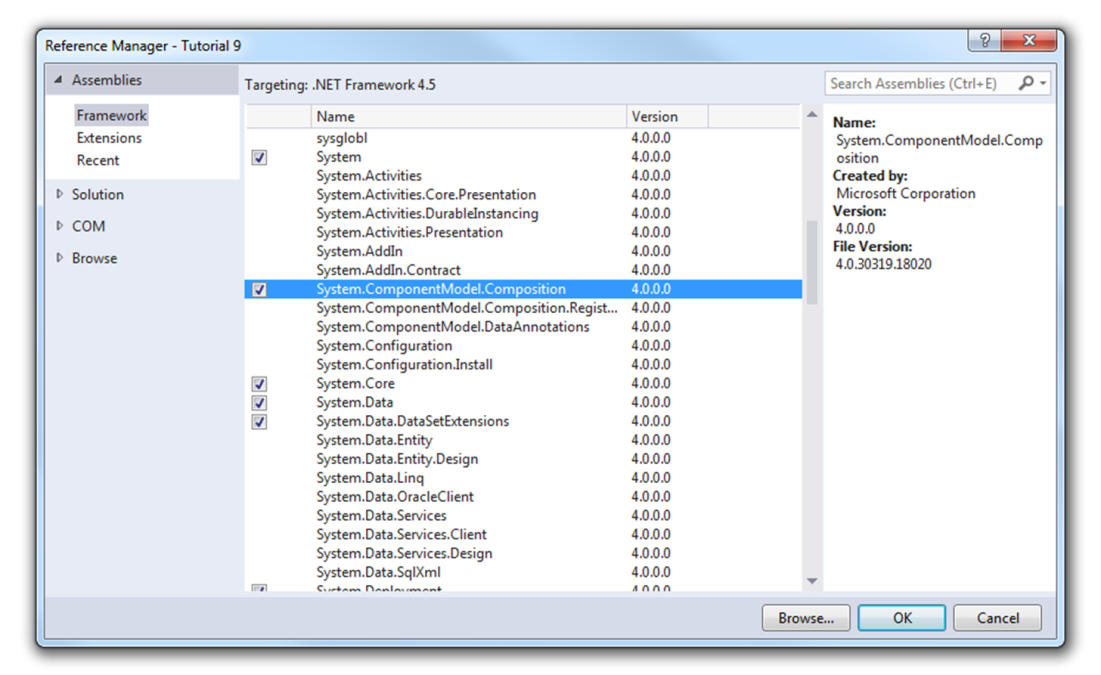


Figure 1: System.ComponentModel.Composition

**Step 3:** Create an Application Extensions folder in your projects bin/Debug folder.

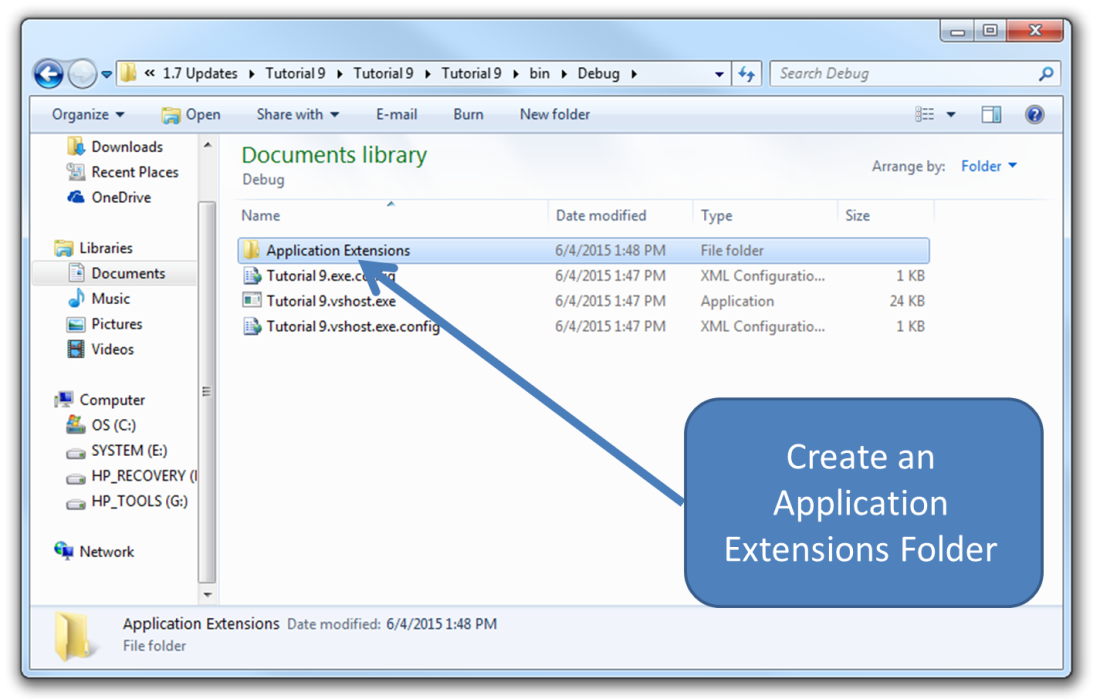


Figure 2: Application Extensions

**Step 4:** Add an AppManger

Drag an "AppManager" control from the DotSpatial controls under toolbox and drop it on the form. Note: This control is necessary for loading different formats of data.

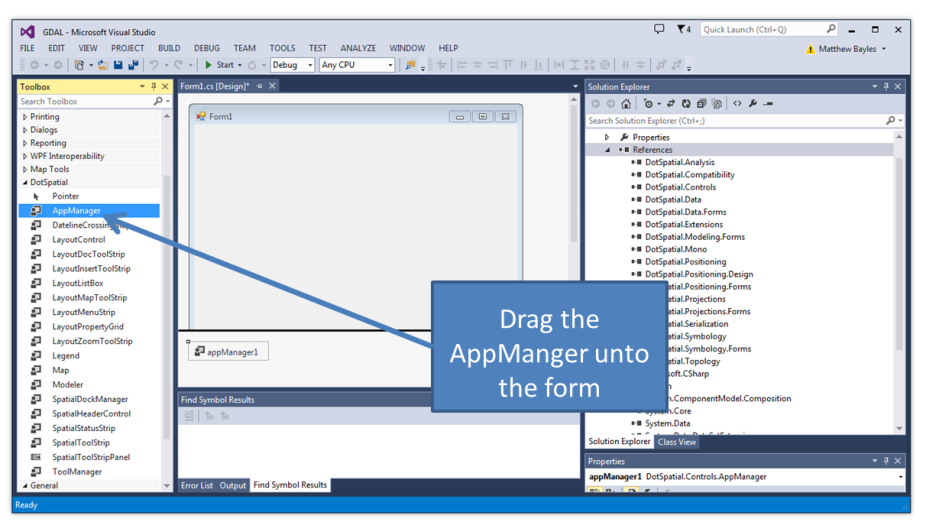


Figure 3: AppManger

**Step 5:** Add additional controls.

In addition to the AppManger we will need to add a SpatialDockManger, Map, Legend, SpatialHeaderControl, and a SpatialStatusStrip, so that we can add a layer to our map.

Set the SpatialDockManger, Map, and Legends Dock to fill.

**Step 6:** Linking the controls.

Now that we have added all the controls we will need to link them to the AppManger. Each of the controls should link to the AppManger through the properties tab. We also need to link our map to our legend through the properties tab as well.

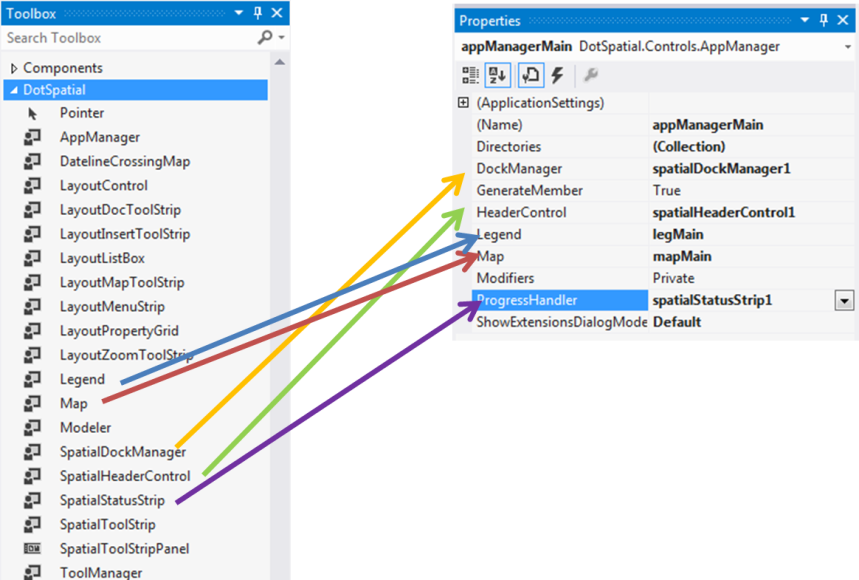


Figure : Linking Controls

**Step 7:** Transfer the GDAL extension folder.

We need to transfer the GDAL extension from the downloaded DotSpatial folder to our Application Extensions folder we created previously. The GDAL extension is located in the “Windows Extension Folder”

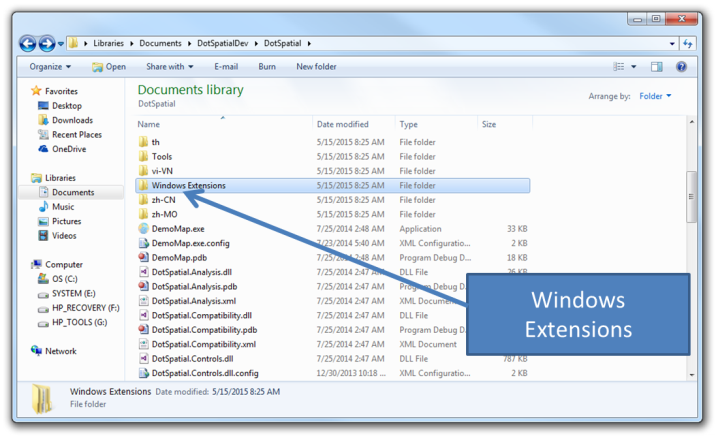


Figure 6: Windows Extensions

Copy the “DotSpatial.Data.Rasters.GdalExtension” and place it in the Application Extensions folder.

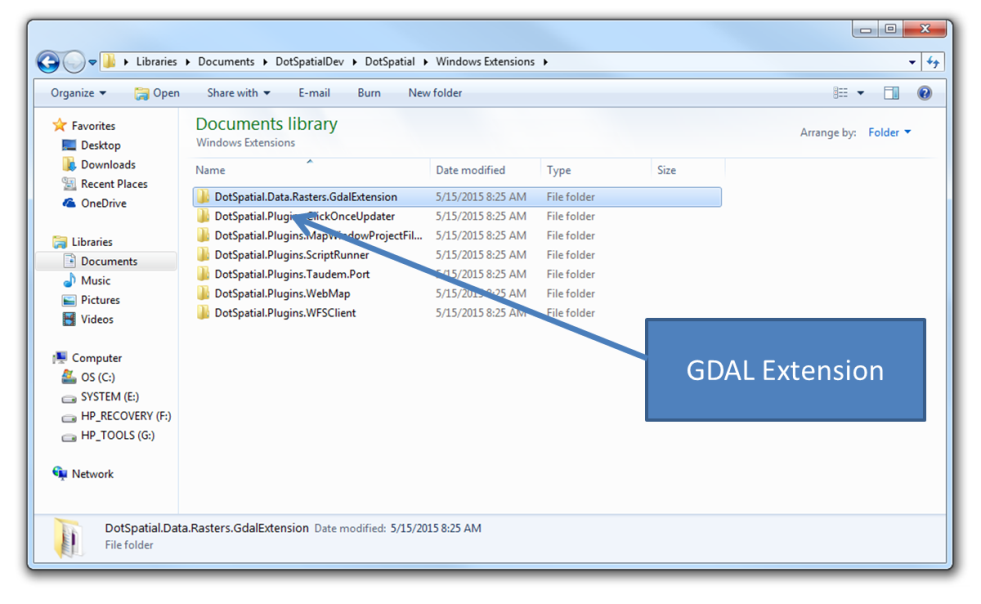


Figure : GDAL Extension

**Step 8:** Code implementation.

Following are required namespaces :

using DotSpatial.Symbology;

using DotSpatial.Controls;

using DotSpatial.Data;

using DotSpatial.Topology;

Add the following code after InitializeComponent();

appManager1.LoadExtensions();

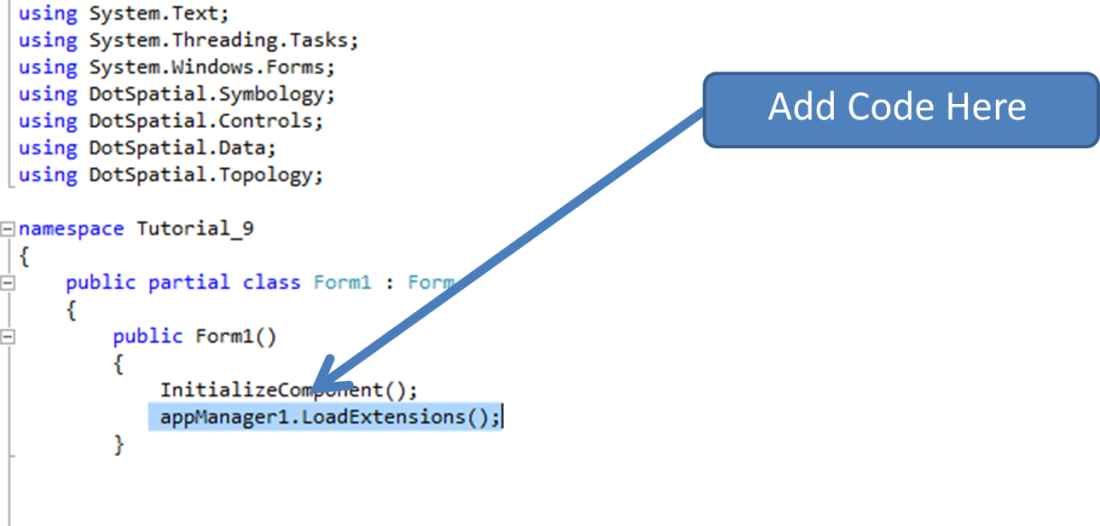


Figure 8: Adding Code

Now that we have added the new line of code, our project has access to many types of files.

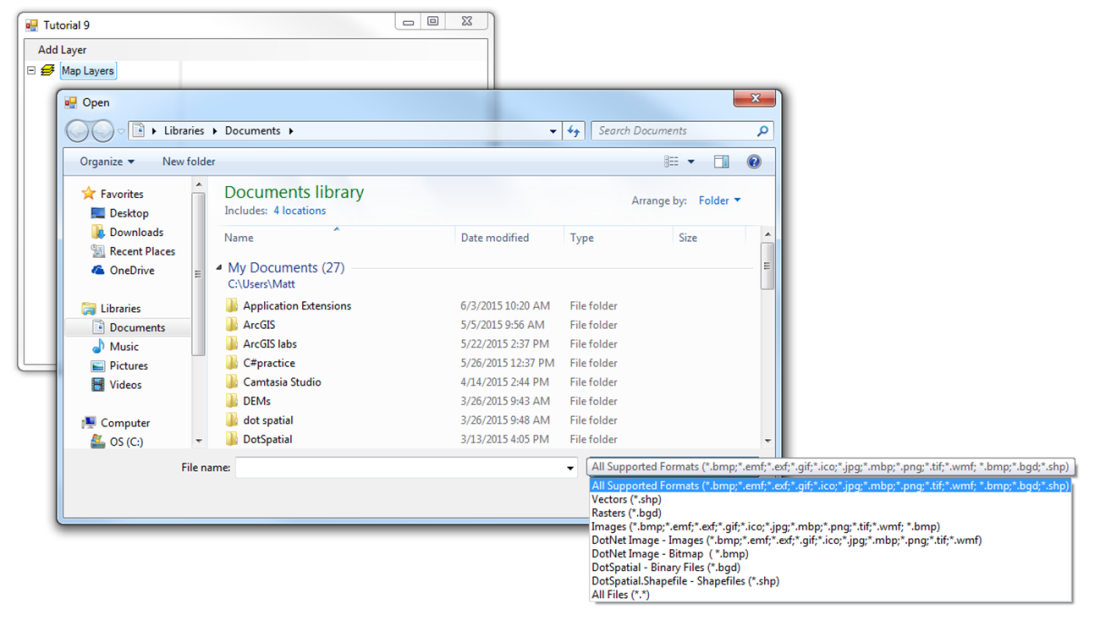


Figure 9: Before Adding GDAL

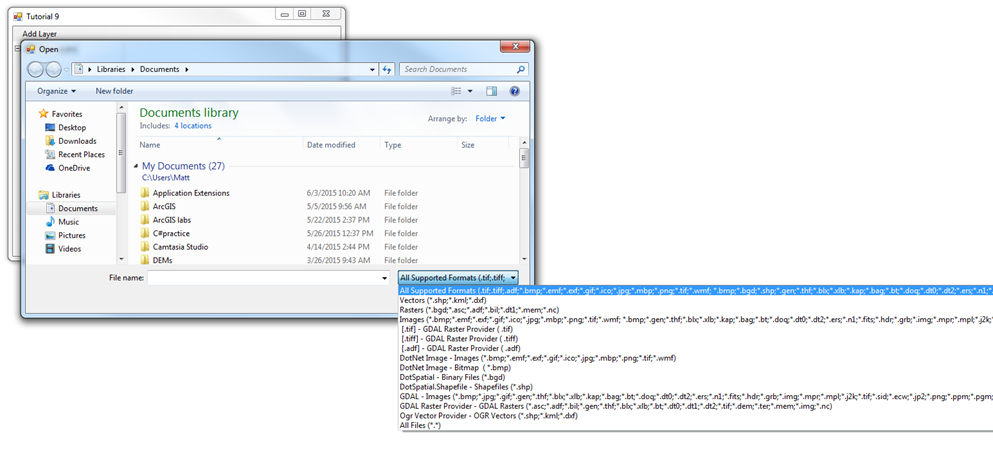


Figure 10: After Adding GDAL