Projection Explorer in DotSpatial

**Tutorial (5)**

Purpose of this tutorial: Become familiar with working with projections in DotSpatial:

Specifically, we will build an application that allows the user to view a polygon shapefile in multiple projections simultaneously and to explore the effect of different projections on area calculations. *.*

**Step 1**: Download the DotSpatial class library

This step is similar to the first Tutorial.

**Step 2:** Add the DotSpatial reference

Add the required DotSpatial reference in the Visual Studio development environment.

Create a new C# application and right click over the project on the solution explorer. On the context menu select the add reference and add the following reference from the DotSpatial folder.

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

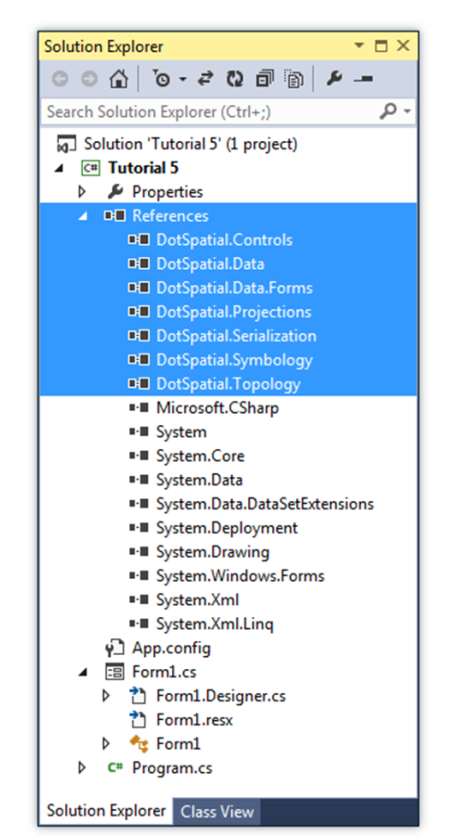


Figure : Required DotSpatial dlls

**Step 3:** Add the DotSpatial Controls into the Visual Studio Toolbox.

This step is as same as the Tutorial # 1 step 3.

**Step 4**: Design the GUI. (Graphical User Interface)

Design the GUI as follows:

Add a panel control and set its properties as follows:

Name: pnlMainAutoscroll: True, Dock: Fill, BorderStyle: Fixed Single

Add 6 more panels on the pnlMain as follows:

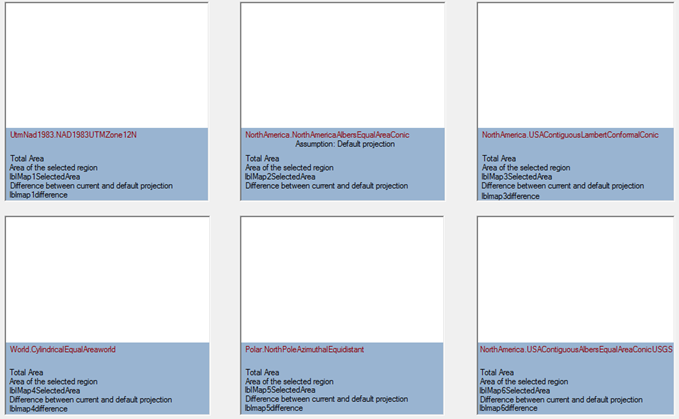


Figure : Main Control Panel

Set the above six panels' properties as follows:

|  |  |  |  |
| --- | --- | --- | --- |
| Controls | Name | BackColor | BorderStyle |
| Panel1 | pnlMap1 | ActiveCaption | Fixed3D |
| Panel2 | pnlMap2 | ActiveCaption | Fixed3D |
| Panel3 | pnlMap3 | ActiveCaption | Fixed3D |
| Panel4 | pnlMap4 | ActiveCaption | Fixed3D |
| Panel5 | pnlMap5 | ActiveCaption | Fixed3D |
| Panel6 | pnlMap6 | ActiveCaption | Fixed3D |

Add 6 map controls into the above panel controls.

Set the map properties as follows:

Dock: top

The following figure shows indicates the order in which the map controls should be placed on their associated panel controls.



Figure : Map Controls

Add the 38 labels described below.

Place the first label at the top of your form to serve as a title for the page. Set its label properties as follows:

Name: lbltitle, ForeColor: Blue, Text: Map Projections

Set the other 37 labels’ properties as follows:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Controls** | **Name** | **ForeColor** | **Text** | **Visible** |
| label1 | lblmap1Projection | DarkRed | UtmNad1983.NAD1983UTMZone12N | True |
| label2 | lblmap2Projection | DarkRed | NorthAmerica.NorthAmericaAlbersEqualAreaConic | True |
| label3 | lblmap3Projection | DarkRed | NorthAmerica.USAContiguousLambertConformalConic | True |
| label4 | lblmap4Projection | DarkRed | World.CylindricalEqualAreaworld | True |
| label5 | lblmap5Projection | DarkRed | Polar.NorthPoleAzimuthalEquidistant | True |
| label6 | lblmap6Projection | DarkRed | NorthAmerica.USAContiguousAlbersEqualAreaConicUSGS | True |
| label7 | lbltotalAreaMap1 | ControlText | Total Area | True |
| label8 | lbltotalAreaMap2 | ControlText | Total Area | True |
| label9 | lbltotalAreaMap3 | ControlText | Total Area | True |
| label10 | lbltotalAreaMap4 | ControlText | Total Area | True |
| label11 | lbltotalAreaMap5 | ControlText | Total Area | True |
| label12 | lbltotalAreaMap6 | ControlText | Total Area | True |
| label13 | lblmap1selectedinfo | ControlText | Area of the selected region | False |
| label14 | lblmap2selectedinfo | ControlText | Area of the selected region | False |
| label15 | lblmap3selectedinfo | ControlText | Area of the selected region | False |
| label16 | lblmap4selectedinfo | ControlText | Area of the selected region | False |
| label17 | lblmap5selectedinfo | ControlText | Area of the selected region | False |
| label18 | lblmap6selectedinfo | ControlText | Area of the selected region | False |
| label19 | lblMap1SelectedArea | ControlText |  | True |
| label20 | lblMap1SelectedArea | ControlText |  | True |
| label21 | lblMap2SelectedArea | ControlText |  | True |
| label22 | lblMap3SelectedArea | ControlText |  | True |
| label23 | lblMap4SelectedArea | ControlText |  | True |
| label24 | lblMap5SelectedArea | ControlText |  | True |
| label25 | lblMap6SelectedArea | ControlText |  | True |
| label26 | lblmap1info | ControlText | Difference between current and default projection | False |
| label27 | lblmap2info | ControlText | Difference between current and default projection | False |
| label28 | lblmap3info | ControlText | Difference between current and default projection | False |
| label29 | lblmap4info | ControlText | Difference between current and default projection | False |
| label30 | lblmap5info | ControlText | Difference between current and default projection | False |
| label31 | lblmap6info | ControlText | Difference between current and default projection | False |
| label32 | lblmap1difference | ControlText |  | False |
| label33 | lblmap2assumption | ControlText | Assumption: Default projection | True |
| label34 | lblmap3difference | ControlText |  | False |
| label35 | lblmap4difference | ControlText |  | False |
| label36 | lblmap5difference | ControlText |  | False |
| label37 | lblmap6difference | ControlText |  | False |

Following figure shows the label controls configuration.

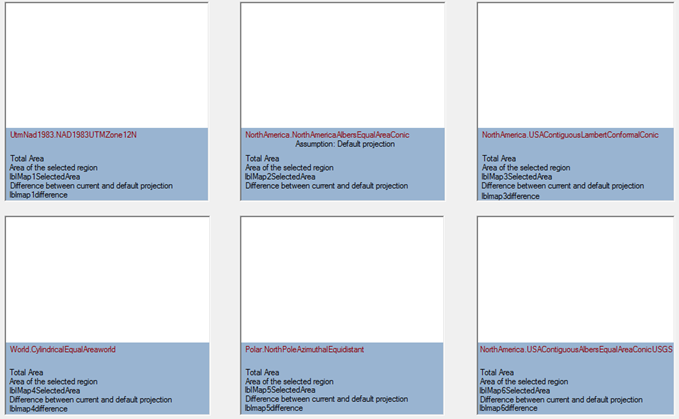


Figure : Label Controls

Add 2 group boxes and set their properties as follows:

First Groupbox's name: gbBasicOperations , Text : Basic Operations

Second Groupbox's name: gbAdvancedOperations , Text = Advanced Operations

Add 2 buttons on the gbBasicOperations and set their properties as follows:

Button 1: Name: btnLoadShapeFile, Text = Load Shapefile

Button 2: Name: btnGetTotalArea, Text = Get Total Area

Add 2 buttons, 2 labels, and 2 combo box on the gbAdvancedOperations as in the following figure:

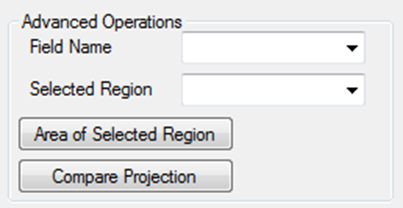


Figure : Advanced Operations

Set the controls properties as follows:

|  |  |  |
| --- | --- | --- |
| Control | Name | Text |
| Label1 | lblFieldName | Field Name |
| Label2 | lblSelectedRegion | Selected Region |
| Combobox | cmbFiledName |  |
| Combobox | cmbSelectedRegion |  |
| Button1 | btnRegionArea | Area of Selected Region |
| Button2 | btnCompareProjections | Compare Projection |

Final Graphical User Interface should look as follows:

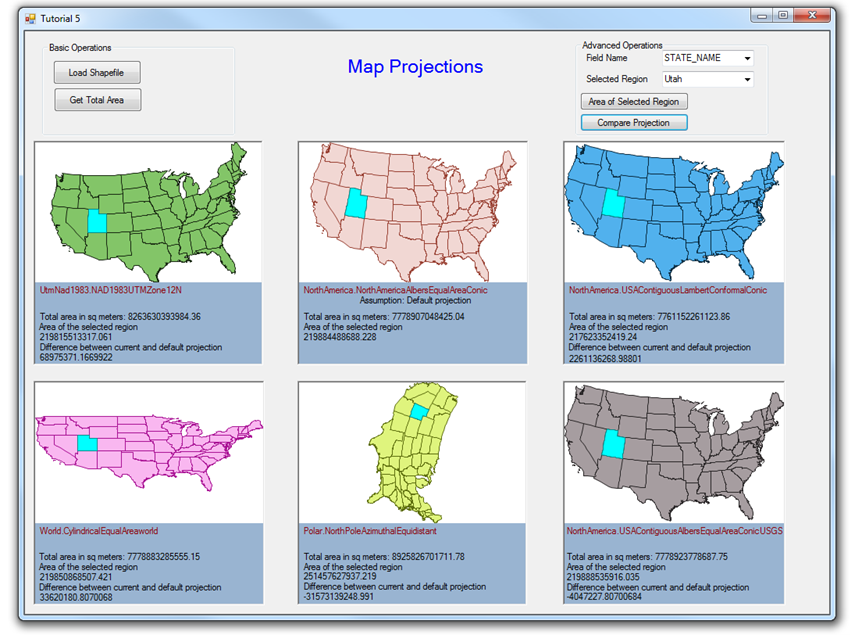


Figure : Final Graphical User Interface

**Step 5: Code implementation**

Add the following namespaces

//Required namespaces

using DotSpatial.Controls;

using DotSpatial.Projections;

using DotSpatial.Data;

Write the following code under btnLoadShapeFile\_click event.

private void btnLoadShapeFile\_Click(object sender, EventArgs e)

{

//define the projections

map1.Projection = KnownCoordinateSystems.Projected.UtmNad1983.NAD1983UTMZone12N;

map2.Projection = KnownCoordinateSystems.Projected.NorthAmerica.NorthAmericaAlbersEqualAreaConic;

map3.Projection = KnownCoordinateSystems.Projected.NorthAmerica.USAContiguousLambertConformalConic;

map4.Projection = KnownCoordinateSystems.Projected.World.CylindricalEqualAreaworld;

map5.Projection = KnownCoordinateSystems.Projected.Polar.NorthPoleAzimuthalEquidistant;

map6.Projection = KnownCoordinateSystems.Projected.NorthAmerica.USAContiguousAlbersEqualAreaConicUSGS;

//add the layers

OpenFileDialog fileDialog = new OpenFileDialog();

fileDialog.Filter = "Shapefiles|\*.shp";

if (fileDialog.ShowDialog() == DialogResult.OK)

{

//add layer to first map

FeatureSet featureSet1 = new FeatureSet();

featureSet1.Open(fileDialog.FileName);

//Populate the FiledName dropdownlist with the help of featureset1.

//We need to pass featureset as an input paramter to FillColumnNames method.

FillColumnNames(featureSet1);

//set the projection

featureSet1.Reproject(map1.Projection);

map1.Layers.Add(featureSet1);

//add layer to second map

FeatureSet featureSet2 = new FeatureSet();

featureSet2.Open(fileDialog.FileName);

featureSet2.Reproject(map2.Projection);

map2.Layers.Add(featureSet2);

//add layer to map3

FeatureSet featureSet3 = new FeatureSet();

featureSet3.Open(fileDialog.FileName);

featureSet3.Reproject(map3.Projection);

map3.Layers.Add(featureSet3);

//add layer to map4

FeatureSet featureSet4 = new FeatureSet();

featureSet4.Open(fileDialog.FileName);

featureSet4.Reproject(map4.Projection);

map4.Layers.Add(featureSet4);

//add layer to map5

FeatureSet featureSet5 = new FeatureSet();

featureSet5.Open(fileDialog.FileName);

featureSet5.Reproject(map5.Projection);

map5.Layers.Add(featureSet5);

//add layer to map6

FeatureSet featureSet6 = new FeatureSet();

featureSet6.Open(fileDialog.FileName);

featureSet6.Reproject(map6.Projection);

map6.Layers.Add(featureSet6);

}

}

Write the following code under btnGetTotalArea\_Click event.

private void btnGetTotalArea\_Click(object sender, EventArgs e)

{

lbltotalAreaMap1.Text = "Total area in sq meters: " + \_getTotalArea(map1).ToString();

lbltotalAreaMap2.Text = "Total area in sq meters: " + \_getTotalArea(map2).ToString();

lbltotalAreaMap3.Text = "Total area in sq meters: " + \_getTotalArea(map3).ToString();

lbltotalAreaMap4.Text = "Total area in sq meters: " + \_getTotalArea(map4).ToString();

lbltotalAreaMap5.Text = "Total area in sq meters: " + \_getTotalArea(map5).ToString();

lbltotalAreaMap6.Text = "Total area in sq meters: " + \_getTotalArea(map6).ToString();

}

Write a private function as follows: This is used to get the total are of the shape file.

private double \_getTotalArea(DotSpatial.Controls.Map mapInput)

{

double stateArea = 0;

if (mapInput.Layers.Count > 0)

{

MapPolygonLayer stateLayer = default(MapPolygonLayer);

stateLayer = (MapPolygonLayer)mapInput.Layers[0];

if ((stateLayer == null))

{

MessageBox.Show("The layer is not a polygon layer.");

}

else

{

foreach (IFeature stateFeature in stateLayer.DataSet.Features)

{

stateArea += stateFeature.Area();

}

}

}

return stateArea;

}

Write the following sub method to populate the fieldnames on the fieldname combo box.

private void FillColumnNames(IFeatureSet featureSet)

{

foreach (DataColumn column in featureSet.DataTable.Columns)

{

cmbFiledName.Items.Add(column.ColumnName);

}

}

Write the following sub method for populating the cmbSelectedRegion combo box.

private void FillUniqueValues(string uniqueField, DotSpatial.Controls.Map mapInput)

{

List<string> fieldList = new List<string>();

if (mapInput.Layers.Count > 0)

{

MapPolygonLayer currentLayer = default(MapPolygonLayer);

currentLayer = (MapPolygonLayer)mapInput.Layers[0];

if ((currentLayer == null))

{

MessageBox.Show("The layer is not a polygon layer.");

}

else

{

DataTable dt = currentLayer.DataSet.DataTable;

cmbSelectedRegion.Items.Clear();

foreach (DataRow rows in dt.Rows)

{

cmbSelectedRegion.Items.Add(rows[uniqueField]);

}

}

}

}

Write the following code in the cmbFiledName\_SelectedIndexChanged event.

private void cmbFiledName\_SelectedIndexChanged(object sender, EventArgs e)

{

FillUniqueValues(cmbFiledName.Text, map1);

}

Write the following sub method.

private void setVisible(Label lbl, bool vis)

{

lbl.Visible = vis;

}

Write the following function for getting the selected region's area.

private double \_getArea(string uniqueColumnName, string uniqueValue, DotSpatial.Controls.Map mapInput)

{

double stateArea = 0;

if ((mapInput.Layers.Count > 0))

{

MapPolygonLayer stateLayer = default(MapPolygonLayer);

stateLayer = (MapPolygonLayer)mapInput.Layers[0];

if ((stateLayer == null))

{

MessageBox.Show("The layer is not a polygon layer.");

}

else

{

stateLayer.SelectByAttribute("[" + uniqueColumnName + "] =" + "'" + uniqueValue + "'");

foreach (IFeature stateFeature in stateLayer.DataSet.Features)

{

if (uniqueValue.CompareTo(stateFeature.DataRow[uniqueColumnName]) == 0)

{

stateArea = stateFeature.Area();

return stateArea;

}

}

}

}

return stateArea;

}

Write the following code in the btnRegionArea\_click event.

private void btnRegionArea\_Click(object sender, EventArgs e)

{

setVisible(lblmap1selectedinfo, true);

setVisible(lblmap2selectedinfo, true);

setVisible(lblmap3selectedinfo, true);

setVisible(lblmap4selectedinfo, true);

setVisible(lblmap5selectedinfo, true);

setVisible(lblmap6selectedinfo, true);

setVisible(lblMap1SelectedArea, true);

setVisible(lblMap2SelectedArea, true);

setVisible(lblMap3SelectedArea, true);

setVisible(lblMap4SelectedArea, true);

setVisible(lblMap5SelectedArea, true);

setVisible(lblMap6SelectedArea, true);

lblMap1SelectedArea.Text = \_getArea(cmbFiledName.Text, cmbSelectedRegion.Text, map1).ToString();

lblMap2SelectedArea.Text = \_getArea(cmbFiledName.Text, cmbSelectedRegion.Text, map2).ToString();

lblMap3SelectedArea.Text = \_getArea(cmbFiledName.Text, cmbSelectedRegion.Text, map3).ToString();

lblMap4SelectedArea.Text = \_getArea(cmbFiledName.Text, cmbSelectedRegion.Text, map4).ToString();

lblMap5SelectedArea.Text = \_getArea(cmbFiledName.Text, cmbSelectedRegion.Text, map5).ToString();

lblMap6SelectedArea.Text = \_getArea(cmbFiledName.Text, cmbSelectedRegion.Text, map6).ToString();

}

Write the following function to calculate the difference between projections.

private double \_calculateDifference(double area1, double area2)

{

double areadiff = 0;

areadiff = area1 - area2;

return areadiff;

}

Write the following code under the compare projection button click event.

private void btnCompareProjections\_Click(object sender, EventArgs e)

{

lblmap1difference.Text = \_calculateDifference(Convert.ToDouble(lblMap2SelectedArea.Text), Convert.ToDouble(lblMap1SelectedArea.Text)).ToString();

lblmap3difference.Text = \_calculateDifference(Convert.ToDouble(lblMap2SelectedArea.Text), Convert.ToDouble(lblMap3SelectedArea.Text)).ToString();

lblmap4difference.Text = \_calculateDifference(Convert.ToDouble(lblMap2SelectedArea.Text), Convert.ToDouble(lblMap4SelectedArea.Text)).ToString();

lblmap5difference.Text = \_calculateDifference(Convert.ToDouble(lblMap2SelectedArea.Text), Convert.ToDouble(lblMap5SelectedArea.Text)).ToString();

lblmap6difference.Text = \_calculateDifference(Convert.ToDouble(lblMap2SelectedArea.Text), Convert.ToDouble(lblMap6SelectedArea.Text)).ToString();

setVisible(lblmap1difference, true);

setVisible(lblmap1info, true);

setVisible(lblmap3difference, true);

setVisible(lblmap3info, true);

setVisible(lblmap4difference, true);

setVisible(lblmap4info, true);

setVisible(lblmap5difference, true);

setVisible(lblmap5info, true);

setVisible(lblmap6difference, true);

setVisible(lblmap6info, true);

}

**Output of the Project**

US states shape file:

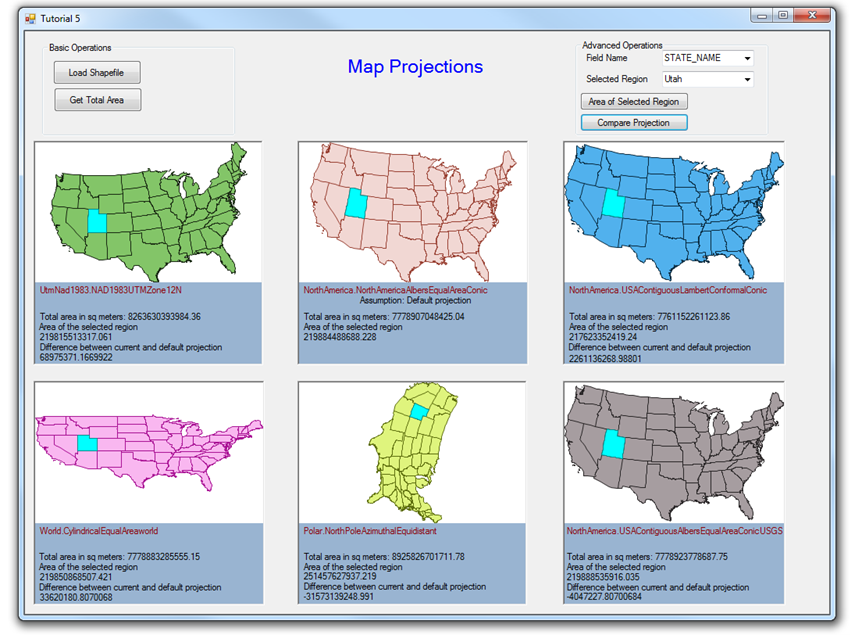


Figure : Final Output