



VDISoft.Analytics

QuickStart Guide



Information Worker Solutions

Introducing VDISoft.Analytics.....	3
Overview.....	3
Where to Use VDISoft.Analytics	3
Getting Started	4
Loading Controls in Visual Studio.net.....	4
Step 1 – Open the toolbox by either clicking Ctrl+Alt+X or selecting View Toolbox from the menu	4
Step 2 – Create a tab for the controls.....	4
Step 3 – Create a tab called VDISoft.Analytics.....	4
Step 4 – Right click in the VDISoft.Analytics tab and select the Choose Items... option	5
Simple Project Example.....	6
Step 1 – Create a new Windows Forms Project in Visual Studio.net.....	6
Step 3 – Drag an Individuals Chart Control on to the form and resize it so that it looks something like the form below:.....	6

Introducing VDISoft.Analytics

Overview

VDISoft.Analytics is a collection of .net-based components for process control, data analysis and Six Sigma solutions.

The VDISoft.Analytics suite consists of nine components and a calculation engine.

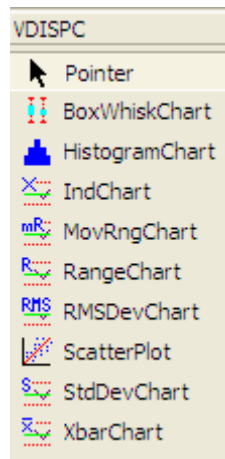


Figure 1 - VDISoft.Analytics Components in Visual Studio.net Toolbox

Each component comes with many custom properties, methods and events that give you complete control over the look and function of your application.

Where to Use VDISoft.Analytics

You can use VDISoft.Analytics in any solution that requires .net-based components. Some examples are:

- Shop-floor process control and analysis systems
- LIMS applications
- Statistical analysis of databases
- Custom SPC solutions
- Six Sigma solutions and applications
- Gauge monitoring
- Real-time process control solutions
- Web-based applications
- Web-enabled applications (e.g. Click Once solutions)
- MES applications
- Quality Management systems

Getting Started

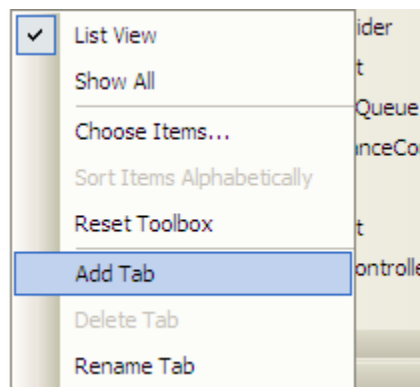
Loading Controls in Visual Studio.net

All of the controls in the VDISoft.Analytics library are contained in a single DLL to ease deployment. Once you have downloaded the DLL from our website (<http://www.vdisoft.net>) you can easily install them into the Visual Studio.net IDE by following these simple steps:

Step 1 – Open the toolbox by either clicking Ctrl+Alt+X or selecting View|Toolbox from the menu

Step 2 – Create a tab for the controls

Right click in the toolbox and select the Add Tab option



Step 3 – Create a tab called VDISoft.Analytics



Step 4 – Right click in the VDISoft.Analytics tab and select the Choose Items... option

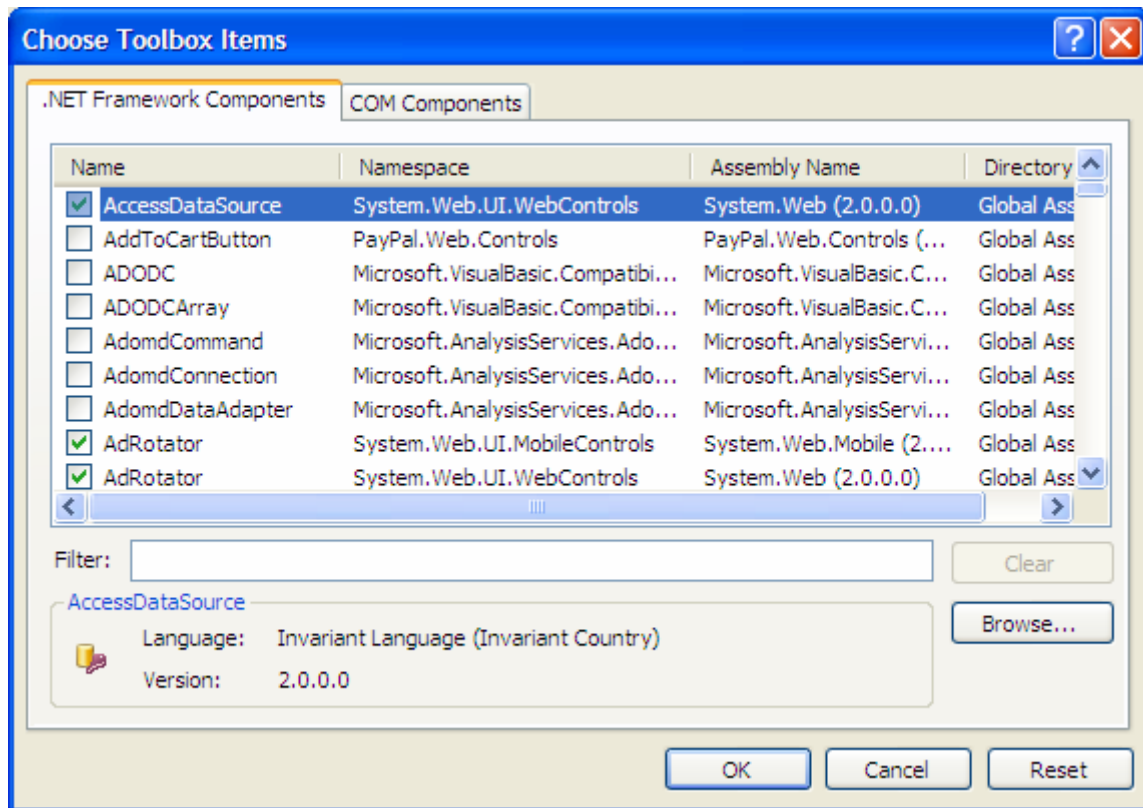
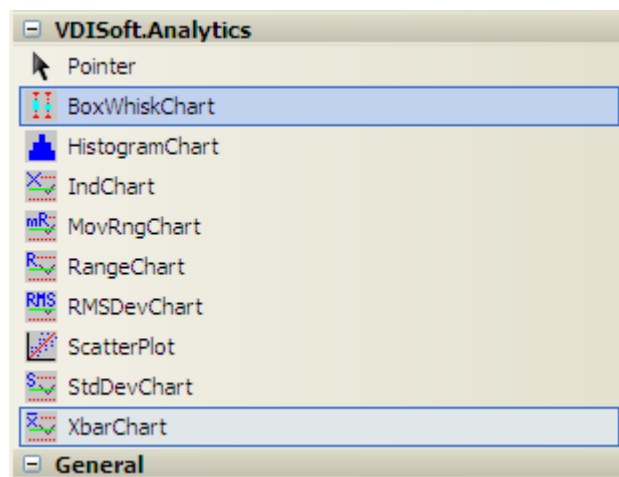


Figure 2 - Click the Browse button to location the VDISPCWIN.DLL library, after you have selected the library click OK.

Once you have loaded the controls they will appear in the Visual Studio.net Toolbox as shown below.



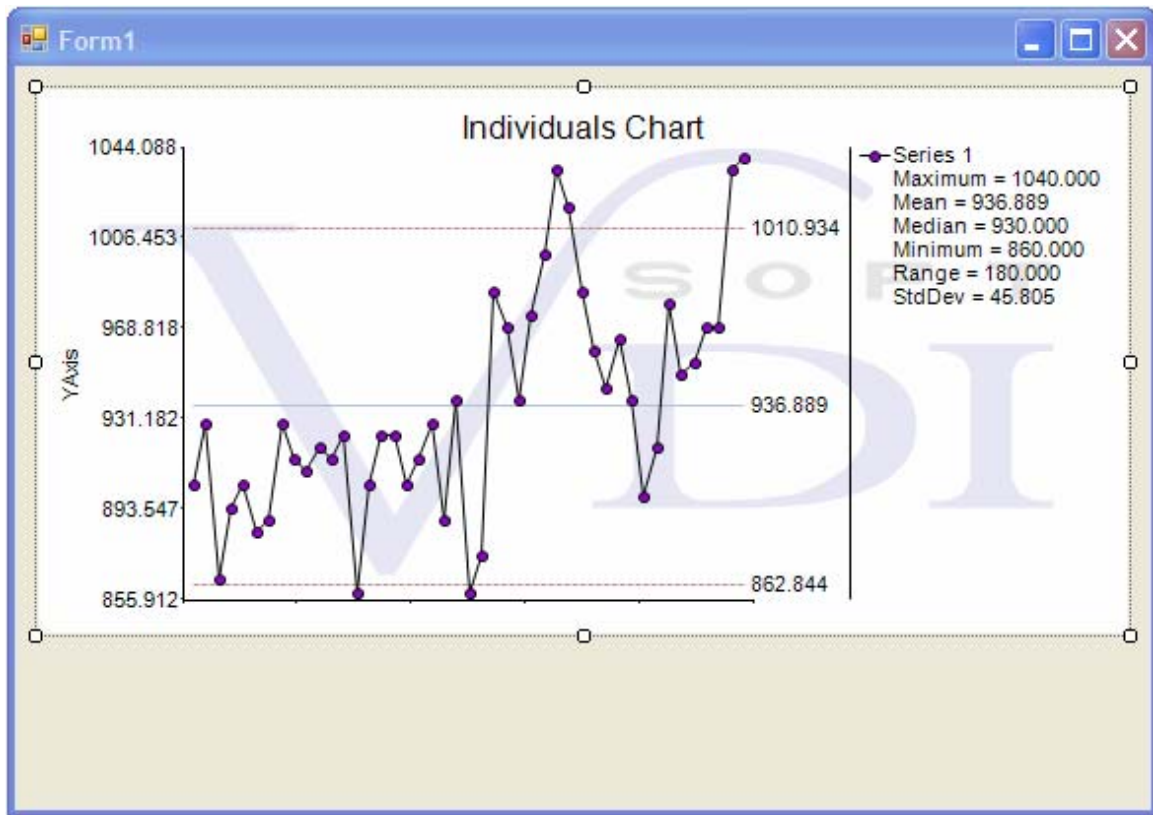
Simple Project Example

In this example we will create a simple Windows form that illustrates how to use one of the controls in the VDISoft.Analytics library.

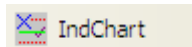
Step 1 – Create a new Windows Forms Project in Visual Studio.net

Note: For this example we will use c# but the procedures would be very similar for a vb.net application. We assume the reader is familiar with how to create a new project in Visual Studio.net. If you are not familiar then consult the Visual Studio.net documentation.

Step 3 – Drag an Individuals Chart Control on to the form and resize it so that it looks something like the form below:



Note: The Individuals Chart is the chart with an icon that looks like this:



Step 4 – Load some data into the control

For our example we will use a dataset that has been saved as an XML file. Copy the XML below to notepad and save it as test.xml somewhere convenient on your hard disk.

```
<?xml version="1.0" standalone="yes"?>
<Sample>
  <Samples>
    <SampleValue>0.22775926870655233</SampleValue>
    <SampleDate>2007-02-01T23:10:49.042784-05:00</SampleDate>
  </Samples>
  <Samples>
    <SampleValue>0.3862580826442028</SampleValue>
    <SampleDate>2007-02-01T23:11:49.042784-05:00</SampleDate>
  </Samples>
  <Samples>
    <SampleValue>0.28617231980253582</SampleValue>
    <SampleDate>2007-02-01T23:12:49.042784-05:00</SampleDate>
  </Samples>
  <Samples>
    <SampleValue>0.44478955140560378</SampleValue>
    <SampleDate>2007-02-01T23:13:49.042784-05:00</SampleDate>
  </Samples>
  <Samples>
    <SampleValue>0.42166995463039258</SampleValue>
    <SampleDate>2007-02-01T23:14:49.042784-05:00</SampleDate>
  </Samples>
  <Samples>
    <SampleValue>0.5660174608072347</SampleValue>
    <SampleDate>2007-02-01T23:15:49.042784-05:00</SampleDate>
  </Samples>
  <Samples>
    <SampleValue>0.53811135913157437</SampleValue>
    <SampleDate>2007-02-01T23:16:49.042784-05:00</SampleDate>
  </Samples>
  <Samples>
    <SampleValue>0.0724422904068801</SampleValue>
    <SampleDate>2007-02-01T23:17:49.042784-05:00</SampleDate>
  </Samples>
  <Samples>
    <SampleValue>0.0956755984088758</SampleValue>
    <SampleDate>2007-02-01T23:18:49.042784-05:00</SampleDate>
  </Samples>
  <Samples>
    <SampleValue>0.7219142283880684</SampleValue>
    <SampleDate>2007-02-01T23:19:49.042784-05:00</SampleDate>
  </Samples>
</Sample>
```

Table 1 - Sample data as xml

Place a button on the form and place this code inside the button's click event:

```
DataSet ds = new DataSet("Sample");

//NOTE: Point to the file on your hard drive
//       wherever it is
ds.ReadXml(@"c:\rc\test.xml");

indChart1.DataSource = ds;
indChart1.DataMember = "Samples";
indChart1.DataValueField = "SampleValue";
indChart1.DataDateTimeField = "SampleDate";
indChart1.DataBind();
```

This example uses data-binding which is built into the controls in order to load the dataset and display it as a control chart.

The resulting chart looks like this:

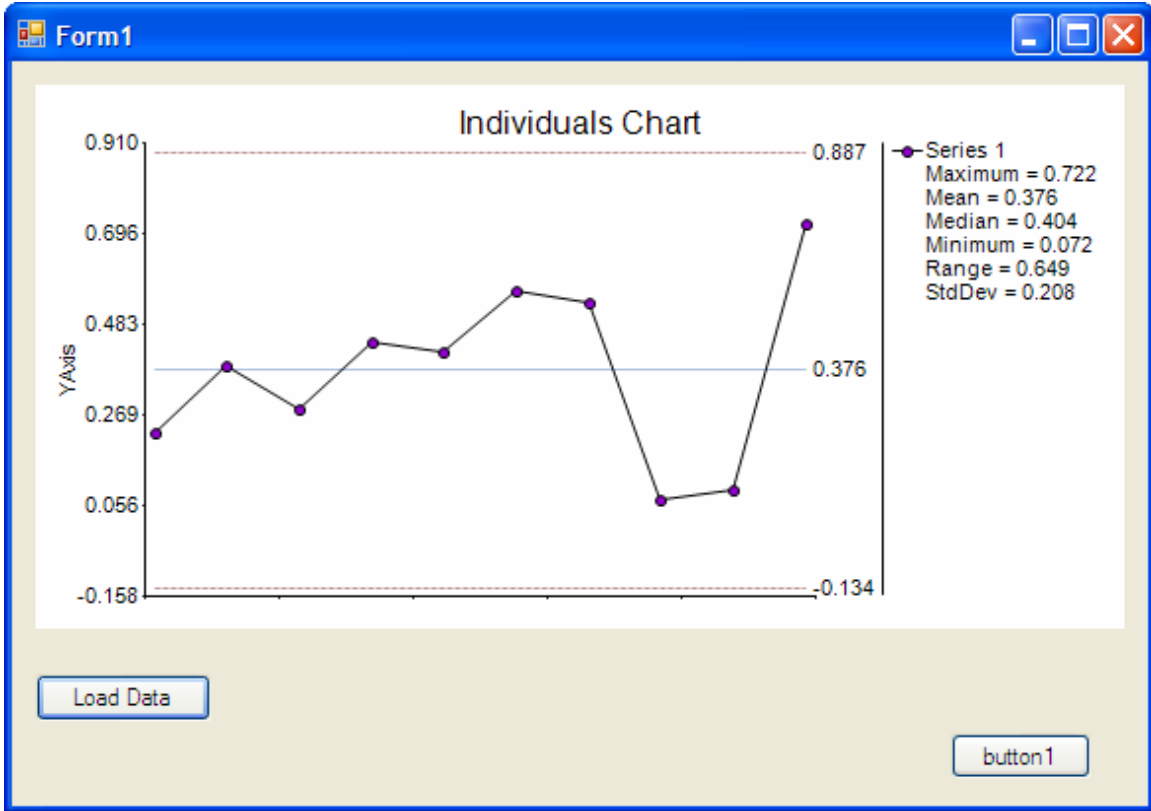


Figure 3 - Sample control chart