



**CUAHSI**  
universities allied for water research

# INGESTING NWIS DATA USING VB.NET AND VISUAL STUDIO 2008

January, 2010

by:

**Tim Whiteaker**  
Center for Research in Water Resources  
The University of Texas at Austin

## Distribution

---

This tutorial and example files are provided free of charge under the Berkeley Software Distribution (BSD) license. Please see the following license information:

Copyright © 2010, Consortium of Universities for the Advancement of Hydrologic Science, Inc.  
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- Neither the name of The University of Texas at Austin nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

## Funding

---

Funding for this document was provided by the Consortium of Universities for the Advancement of Hydrologic Science, Inc. (CUAHSI) under NSF Grant No. EAR-0413265. In addition, much input and feedback has been received from the CUAHSI Hydrologic Information System development team. Their contribution is acknowledged here.

## Support

---

For support with the content of this document, please use the HIS Feedback page at <http://his.cuahsi.org/feedback.html>.

## Table of Contents

---

1	Introduction .....	1
2	Procedure .....	1
2.1	Setting Up the Project .....	1
2.2	Creating the Web Reference .....	2
2.3	Building the User Interface .....	3
2.4	Writing the Code .....	6
2.5	Running the Code.....	9

# 1 INTRODUCTION

This document shows how to connect to WaterOneFlow web services with VB.NET and Visual Studio 2008 by walking you through the creation of a simple program that downloads the latest streamflow values for the Colorado River at Austin, TX. The program accesses real-time streamflow data from the USGS National Water Information System (NWIS) via the WaterOneFlow web service at <http://river.sdsc.edu/wateroneflow/NWIS/UnitValues.aspx>

## Computer and Skill Requirements

To complete this exercise, your computer must meet the following requirements:

- Working Internet connection
- Visual Studio 2008 software

This exercise assumes that you have some familiarity with the following software environments:

- Visual Studio 2008

# 2 PROCEDURE

In this exercise, you will create a windows application with one main window that allows the user to click to see what the average streamflow over the past few days is at the Colorado River at Austin, TX. The application lets the user specify the number of days for which data should be retrieved (up to 30 days back). The application then asks the NWIS Unit Values web service for streamflow values, and then computes the average of the returned values. The NWIS Unit Values web service returns real-time data for roughly the past 31 days. These data typically are recorded at 15-minute intervals.

## 2.1 SETTING UP THE PROJECT

1. Start Visual Studio 2008 (Click on Start | All Programs | Microsoft Visual Studio 2008 | Microsoft Visual Studio 2008).
2. Click File | New | Project...

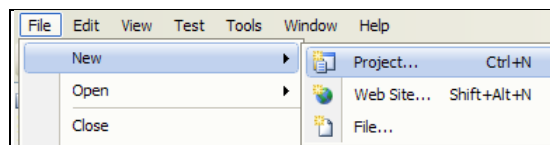
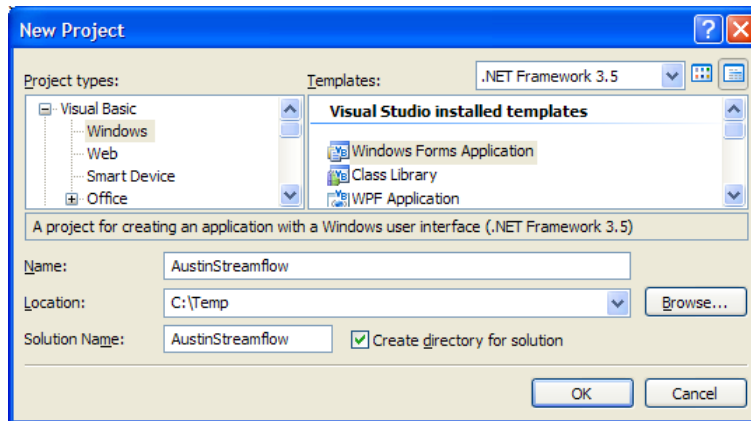


Figure 1 Creating a New Project

3. In the New Project window, set the following properties:
  - a. Choose Visual Basic | Windows from Project Types.
  - b. Select Windows Forms Application from Templates.
  - c. Type "AustinStreamflow" as the Name.
  - d. Set the location where you want to save the project, e.g., C:\Temp.
  - e. Click OK.



**Figure 2 New Project Wizard**

A new project will open with a default form called Form1.

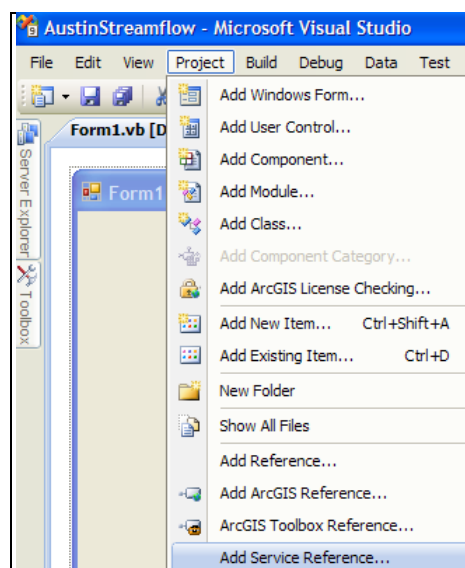
**Note**

If you did not choose Visual Basic as the default language when installing Visual Studio, you might find the template for Visual Basic projects under the “Other Languages” project type.

## 2.2 CREATING THE WEB REFERENCE

This project will make use of the NWIS Unit Values web service to retrieve streamflow values from the USGS stream gage on the Colorado River at Austin. The web service becomes available to the project after making a web reference to the service.

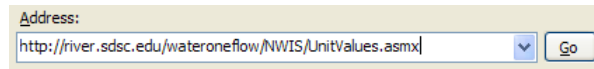
1. Click Project | Add Service Reference...



**Figure 3 Adding a Web Reference**

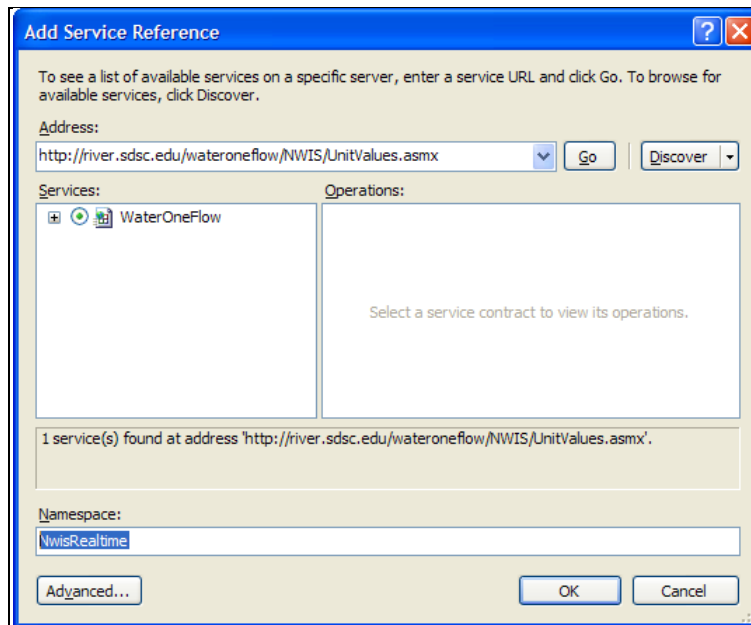
2. In the Add Service Reference window (below Address: ), type in the following URL:

`http://river.sdsc.edu/wateroneflow/NWIS/UnitValues.asmx`



**Figure 4 Address Box**

3. Click Go. Visual Studio will navigate to the URL and verify that a web service is present.
4. Change the namespace from the default to NwisRealtime. This is the name by which you will reference the NWIS web service in your code.



**Figure 5 Add Service Reference Wizard**

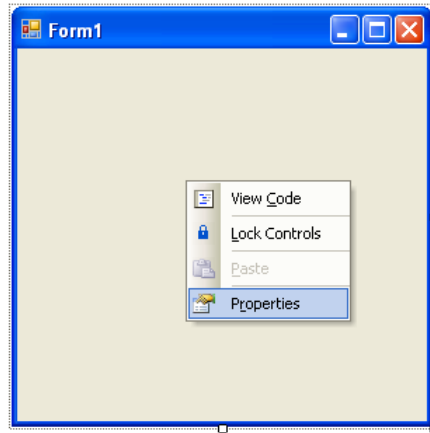
5. Click OK.

The NWIS web service is now available for use within your project.

## 2.3 BUILDING THE USER INTERFACE

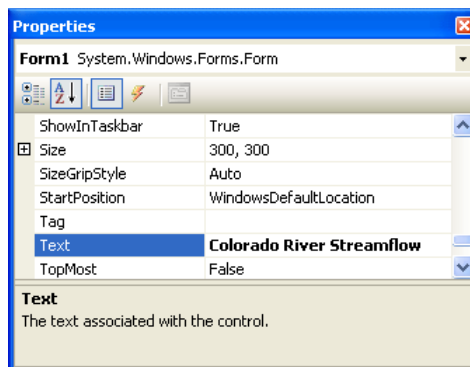
Now that you've set up the project, you'll build the user interface by adding controls to the form. Later, you'll add the code behind those controls which will perform the work.

1. Right click on Form1 and click Properties.



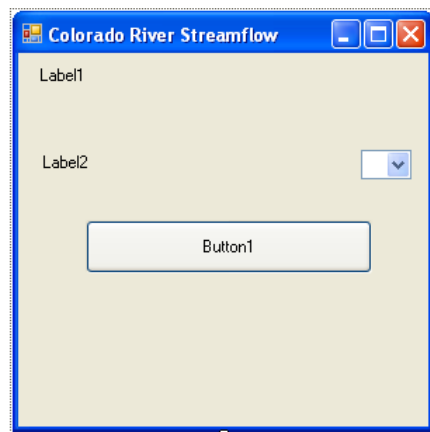
**Figure 6 Accessing Properties of a Form**

2. Change the Text property of the form to "Colorado River Streamflow". This changes the name that appears in the title bar of the form.



**Figure 7 Form Properties**

3. Add two labels, one combo box, and one button to the form, at roughly the same positions as shown in the figure below.



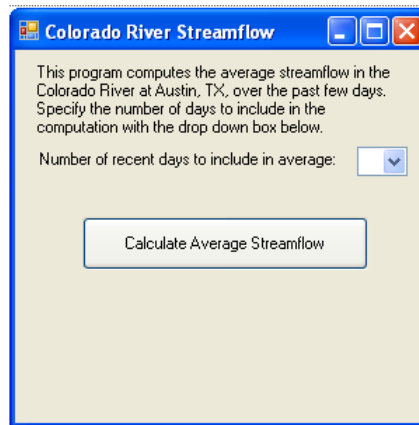
**Figure 8 Adding Components on a Form**

- In a manner similar to setting the Text property of the form, set the properties of the controls as shown below.

Control	Property	Value
Label1	Text	This program computes the average streamflow in the Colorado River at Austin, TX, over the past few days. Specify the number of days to include in the computation with the drop down box below.
	AutoSize	False
Label2	Text	Number of recent days to include in average:
ComboBox1	DropDownStyle	DropDownList
Button1	(Name)	btnCalculate
	Text	Calculate Average Streamflow

**Table 1: Form Properties to be Set**

The form should now look similar to the one below.

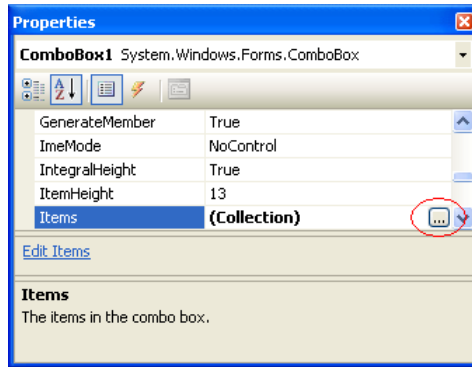


**Figure 9 Final Form**

Now you will add the choice of 1 to 10 days to the combo box.

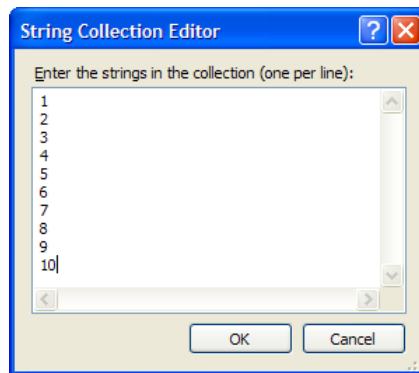
- Click the properties for ComboBox1, and then select the Items property. Click the ellipsis next to (Collection).





**Figure 10 ComboBox1 Properties**

6. Add the numbers 1 through 10 to the String Collection Editor window. This allows the user to select between 1 and 10 days to include in the computation of average streamflow.



**Figure 11 String Collection Editor Window**

7. Click OK to close the String Collection Editor window.

## 2.4 WRITING THE CODE

1. Double click the form (be sure and not to click on any of the controls that you have added to the form.) This opens the code editor, and creates stub code that will be run when the form opens.

```
Public Class Form1
    Private Sub Form1_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
        End Sub
End Class
```

2. Add the following code to the Form1\_Load procedure.

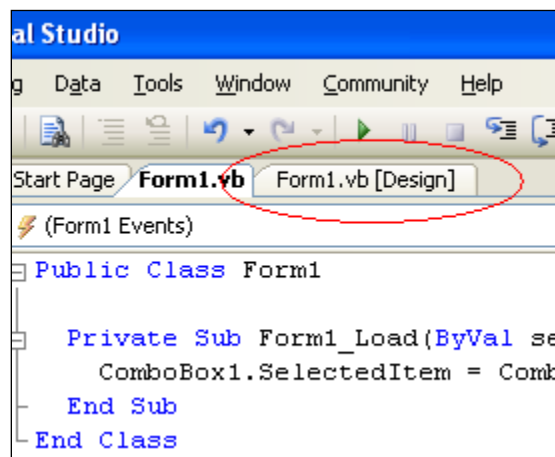
```
ComboBox1.SelectedItem = ComboBox1.Items.Item(9)
```

The result is shown in the screenshot below.

```
Public Class Form1
    Private Sub Form1_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
        ComboBox1.SelectedItem = ComboBox1.Items.Item(9)
    End Sub
End Class
```

In the code above, you are setting the selected item in the combo box to be the 10<sup>th</sup> item (which happens to be the number 10). Indices in VB.NET begin with zero, not one. So the first item in the combo box has an index of zero, while the last item has an index of 9 in this case.

3. At the top of the code editor, click the Form1.vb [Design] tab.



**Figure 12 Code Editor**

This shows the form and the controls that you have placed on it. This is a convenient view for choosing a specific control to write code for. Now you'll add code to the button to compute average streamflow.

4. Double click the Calculate Average Streamflow button to open the code editor and automatically create stub code for the Click event for that button.
5. Add the following code to the btnCalculate\_Click procedure.

```

' Set initial parameters.
' Set the siteCode for our gage of interest.
Dim location As String = "NWISUV:08158000"
' Set the variableCode for streamflow.
Dim variable As String = "NWISUV:00060"
' Set start and end date.
Dim startDate, endDate As String
Dim tmpDate As Date
endDate = Format(Now, "yyyy-MM-dd")
tmpDate = Now.AddDays(-1 * ComboBox1.SelectedItem + 1)
startDate = Format(tmpDate, "yyyy-MM-dd")

' Call the web service.

Dim ws As New NwisRealtime.WaterOneFlowSoapClient
Dim tsResponse As NwisRealtime.TimeSeriesResponseType
tsResponse = ws.GetValuesObject(location, variable, _
                                startDate, endDate, "")

' Process the results.

Dim vals As NwisRealtime.TsValuesSingleVariableType
vals = tsResponse.timeSeries.values
If vals.count = 0 Then
    MsgBox("No values returned")
    Exit Sub
End If

Dim avg As Double = 0

For i As Integer = 0 To vals.count - 1
    avg += vals.value(i).Value
Next

avg = avg / vals.count
MsgBox("The average streamflow is " & _
        FormatNumber(avg, 1) & " cfs")

```

In the code above, you are first preparing the inputs to feed the web service. The tricky part of this is formatting the dates to “yyyy-MM-dd” format (e.g., 2006-12-31), which is what the web service is expecting. Another trick is calculating the start date by adding “negative” days to the current date in the line:

```
tmpDate = Now.AddDays(-1 * ComboBox1.SelectedItem + 1)
```

Next you are creating a new instance of the NWIS Daily Values web service and calling the `GetValuesObject` method from the service with the date inputs from the user. This method returns an Object with the data retrieved from the web service.

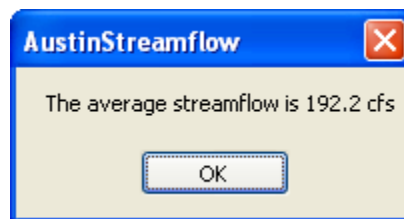
Next, with the results from the `GetValuesObject` call, you are computing the average streamflow from the values returned, and then showing a message box to report the result.

## 2.5 RUNNING THE CODE

The project is now ready to run.

1. Press F5 on your keyboard to run it.
2. Click the Calculate Average Streamflow button.

After a minute or two, a message box appears showing the average streamflow over the past 10 days. Note that your value may be different than the value in the screenshot below, since the value depends on what the streamflow was over the past thirty days when the button is clicked..



**Figure 13 Final Result**

3. Close the form when you are finished.

You have completed the exercise and have learned how to call a web service from Visual Studio 2008. From this point, you could build the solution as an executable file by pressing `Ctrl-Shift-B` on your keyboard. See your Visual Studio help for more information about building solutions.