



CUAHSI
universities allied for water research

WATERONEFLOW WEB SERVICES

Services for publishing hydrologic observations on a HydroServer

VERSION 1.1 SETUP GUIDE AND SOFTWARE MANUAL

August 2011

Prepared by:

Jeffery S. Horsburgh
Utah Water Research Laboratory
Utah State University

David Valentine and Tom Whitenack
San Diego Supercomputer Center
University of California, San Diego

Tim Whiteaker and Ernest To
Center for Research in Water Resources
University of Texas at Austin

Distribution

The WaterOneFlow Web Services and all associated source code and documentation are available at the following URL: <http://his.cuahsi.org/>.

The WaterOneFlow Web Service software, source code, and documentation are provide free of charge under the Berkeley Software Distribution (BSD) license. Please see the following license information:

Copyright © 2010, University of California Regents.

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 University of California Regents 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.

Disclaimers

Although much effort has been expended in the development and testing of the software described in this document, errors and inadequacies may still occur. Users must make the final evaluation as to the usefulness of this software for their application.

Acknowledgements

The WaterOneFlow Web Services and this software manual are based upon work supported by the National Science Foundation (NSF) under Grants No. 03-26064, and 06-10075. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author and do not necessarily reflect the views of NSF.

The team of engineers and software developers that developed the WaterOneFlow Web Services includes:

David Valentine, Research Programmer, San Diego Supercomputer Center, University of California, San Diego.

Tim Whiteaker, Research Associate, Center for Research in Water Resources, University of Texas at Austin.

Technical Support

For questions and support, or if you find errors, have suggestions, or are interested in any later versions, please contact:

David Valentine
University of California, San Diego
San Diego Supercomputer Center, MC 0505
9500 Gilman Drive
La Jolla, CA 92093-0505
valentin@sdsc.edu

Table of Contents

1.0	Introduction and Software Description.....	1
1.1	General Functionality.....	1
1.2	Platform and Minimum System Requirements.....	1
2.0	Before You Install	1
2.1	Installation Prerequisites	1
2.1.1	Setting Up a Domain for Your HydroServer.....	2
3.0	Installing and Configuring the WaterOneFlow Web Services	3

1.0 INTRODUCTION AND SOFTWARE DESCRIPTION

The CUAHSI Hydrologic Information System (HIS) Project is developing information technology infrastructure to support hydrologic science. One of the major components of the HIS is a software stack called HydroServer that can be used for storing and publishing hydrologic data. HydroServer includes a point Observations Data Model (ODM), which is a relational database schema that was designed for storing time series data, a suite of data loaders and tools for working with ODM, the WaterOneFlow Web services that publish data stored in an ODM database on the Internet in WaterML format, and the capability to publish geographic information systems (GIS) datasets as spatial data services. Using the HydroServer software stack, server administrators can create any number of observational data services published using the WaterOneFlow web services as well as any number of spatial data services published as Open Geospatial Consortium (OGC) Web Map Services (WMS), Web Coverage Services (WCS), and Web Feature Services (WFS). This document specifically describes the WaterOneFlow Web Services component of the HydroServer Software Stack.

1.1 GENERAL FUNCTIONALITY

The WaterOneFlow Web Services were created for publishing hydrologic observations on the Internet in a standard way. The WaterOneFlow Web Service for ODM was designed to publish data stored in an ODM database. Each ODM database requires a separate WaterOneFlow Web Service to publish its contents. The WaterOneFlow services respond to user requests for data by querying the underlying ODM database and then translating the results of the query into WaterML format. The data is then communicated back to the user over the Internet using WaterML. The WaterOneFlow services consist of a standard set of data discovery and data delivery methods (e.g., GetSites, GetVariables, GetSiteInfo, GetVariableInfo, GetValues) that can be used by computer programmers and a variety of client software programs to access hydrologic observations stored in an ODM database over the Internet.

1.2 PLATFORM AND MINIMUM SYSTEM REQUIREMENTS

The ODM WaterOneFlow Web Services were designed to be used on HydroServers running Windows Server 2003 or Windows Server 2008 with Microsoft Internet Information Services (IIS) as the web server. An ODM database containing hydrologic observations and that is implemented in either Microsoft SQL Server 2005 or Microsoft SQL Server 2008 is also required. In addition, HydroServers must have the Microsoft .NET Framework Version 2.0 installed prior to installing the WaterOneFlow Web Services. Instructions for obtaining the .NET Framework Version 2.0 from Microsoft are included in the Installation Instructions section below.

2.0 BEFORE YOU INSTALL

2.1 INSTALLATION PREREQUISITES

Prior to configuring the ODM WaterOneFlow Web Services, you must first ensure that the Microsoft .NET Framework Version 2.0 is installed. The .NET Framework Version 2.0 is free and may have already been installed on your computer during normal Windows updates. If you are running Windows Server 2008, the .NET Framework Version 2.0 should already be installed and available on your system. If you are running either the 32 or 64 bit versions of Windows Server 2003, instructions for downloading and installing the .NET Framework Version 2.0 can

be obtained from the Microsoft website via the following URLs. It is recommended that you run a Windows update after installing the .NET Framework to ensure that your server has all of the latest Microsoft security updates.

32 Bit Version of .NET Framework 2.0 for Windows Server 2003

<http://www.microsoft.com/downloads/details.aspx?familyid=0856EACB-4362-4B0D-8EDD-AAB15C5E04F5&displaylang=en>

64 Bit Version of .NET Framework 2.0 for Windows Server 2003 64 bit

<http://www.microsoft.com/downloads/details.aspx?familyid=B44A0000-ACF8-4FA1-AFFB-40E78D788B00&displaylang=en>

The ODM WaterOneFlow Web Services also require that Windows Internet Information Services (IIS), which is included with the Windows operating system, is installed.

Once you have verified that the .NET Framework Version 2.0 and IIS have been installed, you can continue with the installation of the ODM WaterOneFlow Web Services.

NOTE: Implementing the ODM WaterOneFlow Web Services requires that you have an ODM database implemented within an instance of Microsoft SQL Server 2005 or 2008. If you do not already have an ODM database, refer to the documentation for ODM for instructions on how to get started with SQL Server and ODM.

Your ODM database can be implemented within any version of Microsoft SQL Server 2005 or 2008, including the free Express version. Additionally, your ODM database can be hosted on a separate machine than the WaterOneFlow Web Services in the case where you want to separate your database server and your web server. You must create an appropriate read-only SQL Server login within your ODM database for the WaterOneFlow Web services to use.

2.1.1 SETTING UP A DOMAIN FOR YOUR HYDROSERVER

Prior to installing the HydroServer Web applications – i.e., the HydroServer Website, the HydroServer Map Application, the HydroServer Time Series Analyst, the ODM WaterOneFlow Web Services, and the HydroServer Capabilities Web Services – you will want to create a domain for your HydroServer. You will need to work with the Information Technology professionals within your organization to help you create a domain. Once a domain has been created for your HydroServer, you can proceed in the setup of the HydroServer software.

In implementing the HydroServer Web applications, it is helpful to understand the structure of the overall deployment so that each of the pieces can be implemented correctly. The HydroServer Website was designed to be a parent, or root level, application within a domain that you set up for your HydroServer. For example, if you were to create a domain name for your HydroServer called “icewater.usu.edu,” the URL for your HydroServer Website would be at the root level of that domain (e.g., <http://icewater.usu.edu/> would be the URL for your HydroServer Website). Each of the other HydroServer Web applications was designed to be a child application of the HydroServer domain. The following shows how the other HydroServer Web applications would be implemented under the HydroServer Website within the same domain:

<http://icewater.usu.edu/tsa/> - The Time Series Analyst

<http://icewater.usu.edu/map/> - The HydroServer Map application

<http://icewater.usu.edu/HydroServerCapabilities/> - the HydroServer Capabilities Web service

<http://icewater.usu.edu/LittleBearRiver/> - a WaterOneFlow Web Service for the Little Bear River experimental watershed

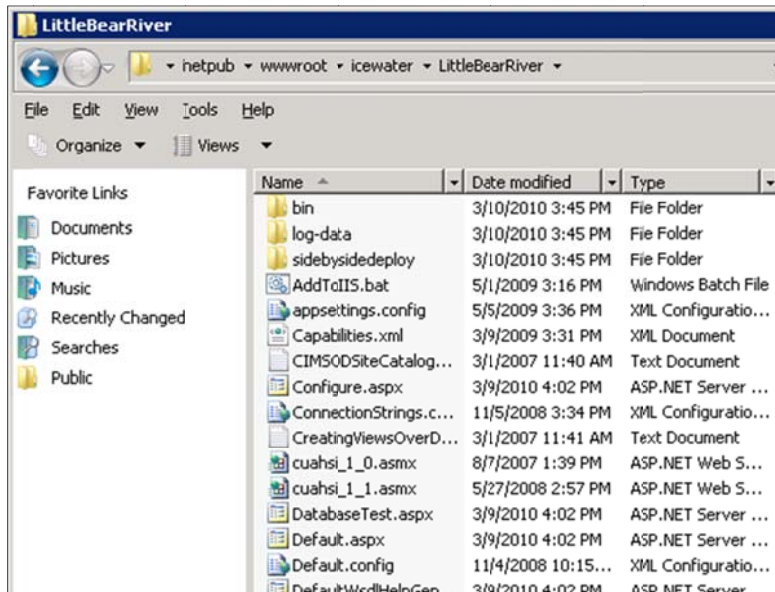
<http://icewater.usu.edu/MudLake/> - a WaterOneFlow Web Service for data collected within Mud Lake at the Bear Lake National Wildlife Refuge

The above example assumes that your HydroServer is serving as the Web server running Microsoft Internet Information Services (IIS), as the database server running Microsoft SQL Server, and as the GIS server running ArcGIS Server. This doesn't have to be the case, though, and there is quite a lot of flexibility for the components of your HydroServer to be spread across multiple machines and implemented within multiple domains. In general, the HydroServer documentation assumes that you are assembling your HydroServer within a single domain.

3.0 INSTALLING AND CONFIGURING THE WATERONEFLOW WEB SERVICES

There are three main tasks associated with installing and configuring the ODM WaterOneFlow Web Services. First you will install the service. Second, you will connect the service to an ODM database, and third, you will configure the application settings for your web service. Use the following steps to install and configure the ODM WaterOneFlow Web Services. The following steps were written for a web server running Windows Server 2008 and IIS Version 7.0.

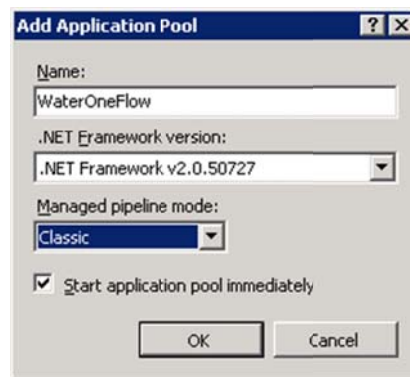
1. Copy the folder containing the ODM WaterOneFlow Web Service program files ("WebApp") from the deployment zip file to a web application directory on your HydroServer. For this example, I have set up a folder called "icewater" under my "c:\inetpub\wwwroot\" path for this application. I copied the "WebApp" folder from the zip file into the "icewater" folder that I created.
2. Rename the "WebApp" folder to something that is consistent with the service that you are creating. For this example, I will rename my "WebApp" folder to "LittleBearRiver" since I am creating a service for my Little Bear River ODM database. When you have renamed the folder, you should have something like the following:



3. Open the Internet Information Services (IIS) Manager by clicking Start → Administrative Tools → Internet Information Services (IIS) Manager. The following window will appear.



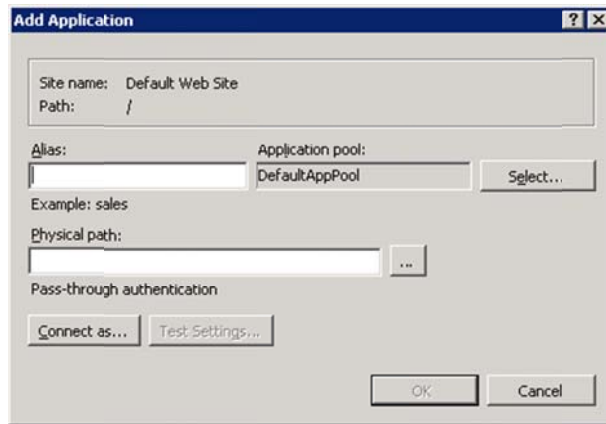
4. Expand your server in the tree view at the left of the IIS Manager form by clicking on the plus sign next to its name. Then, expand the “Sites” element by clicking on the plus sign.
5. We will first create an Application Pool for running the WaterOneFlow Web Services. In the tree view on the left of the IIS Manager, right click on “Application Pools” and select “Add Application Pool”.
6. create a name for the Application Pool in the “Name:” text box. For this example, we will call our Application Pool “WaterOneFlow.” Make sure that “.NET Framework v2.0.50727” is selected in the “.NET Framework version:” text box. From the “Managed pipeline mode:” drop down box, select “Classic.” Ensure that the check box next to “Start application pool immediately” is checked. Your form should look like the following:



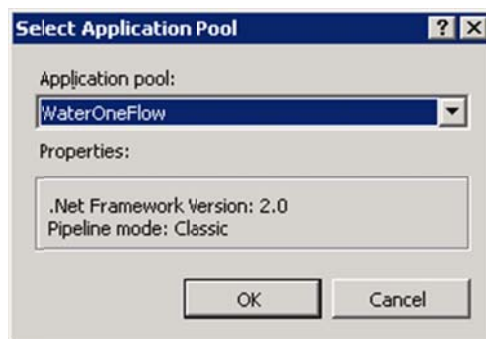
NOTE: In this example we are creating an Application Pool called “WaterOneFlow.” We have chosen the “Classic” Managed pipeline mode because the WaterOneFlow Web Services were created using Visual Studio 2005. It is important to note that the other HydroServer Web applications (e.g., the HydroServer Website, Time Series Analyst, and the HydroServer Map) were developed in Visual Studio 2008 and will require a separate application pool that uses the “Integrated” Managed Pipeline Mode.

If you are creating multiple instances of the WaterOneFlow web services on your server because you have multiple ODM databases, you can reuse the “WaterOneFlow” Application Pool that you just created for those services.

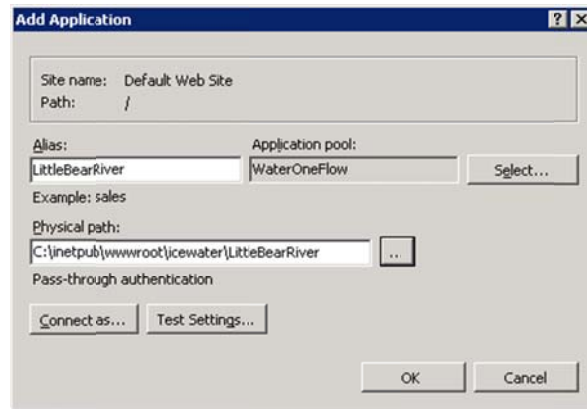
7. For this example, we will implement the WaterOneFlow Web Services under the Default Website in IIS. Right click on the “Default Website” item in the tree view at the left of the IIS Manager and select “Add Application” from the context menu. The following window will open:



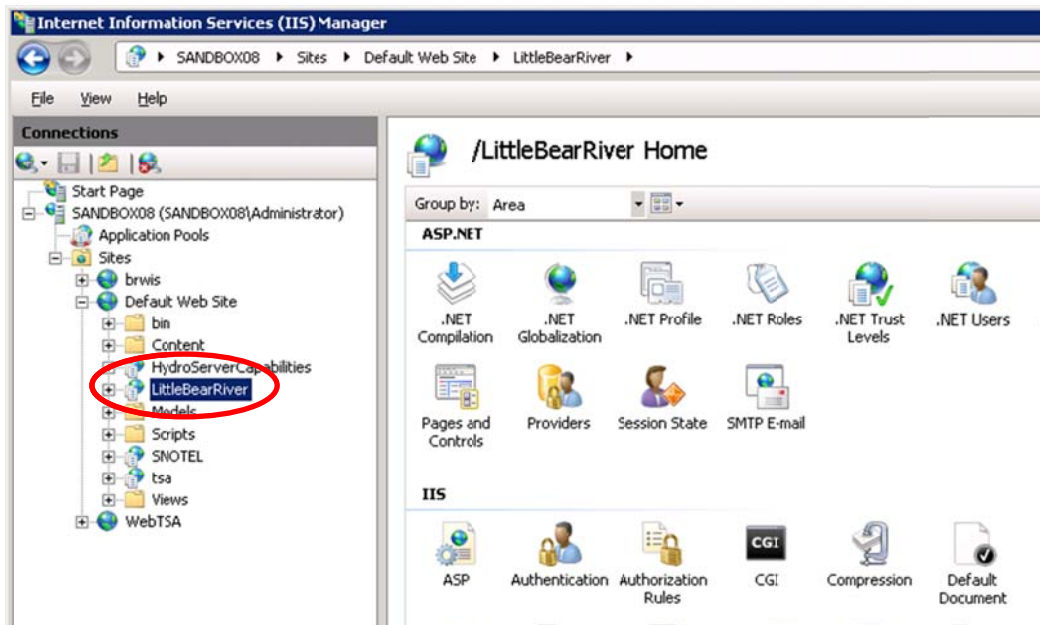
8. In the “Alias” text box, create a name for your WaterOneFlow web service. This name will be part of the URL path to your service. For this example, we will use “LittleBearRiver,” but you will want to set the Alias to something that describes your service. Click the “Select” button next to the “Application pool:” box. In the “Select Application Pool” form that pops up, select the application pool that you just created (e.g., “WaterOneFlow”) from the “Application pool” drop down list and then click “OK”.



9. Click the navigate button next to the “Physical path:” box. Navigate to and select the folder where you extracted the WaterOneFlow Web Service application files (e.g., “c:\inetpub\wwwroot\icewater\LittleBearRiver”). Then click the “OK” button.
10. Your “Add Application” form should now look something like the following. Click the “OK” button to complete this step.

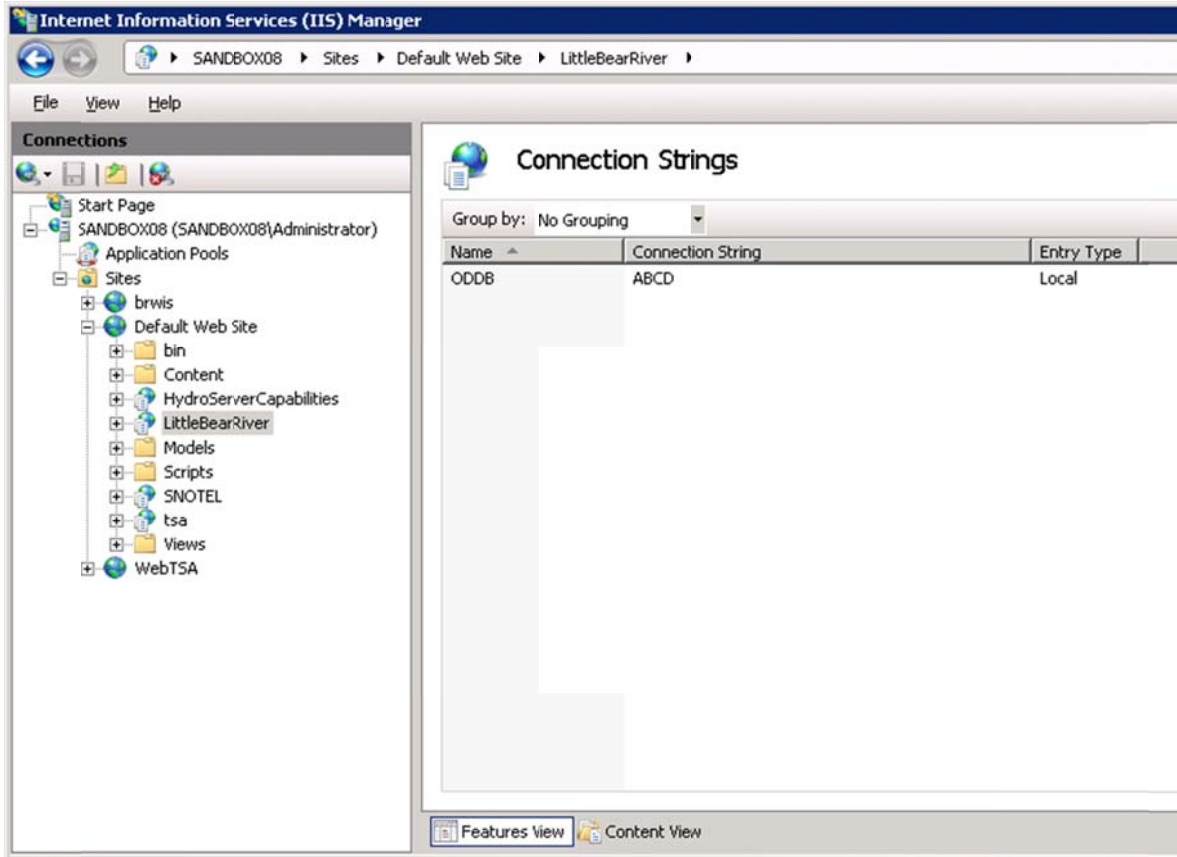


11. You will now notice that an application with the Alias name that you just created (e.g., “LittleBearRiver”) has been created under the Default Website in the IIS Manager.



NOTE: The process for setting up the WaterOneFlow Web Services under a different website is the same as setting them up under the Default Website. However, you must first have created the website in IIS before you can do this. In the example above, since I implemented the WaterOneFlow Web Services under the Default Website on a machine called “sandbox08.uwrl.usu.edu,” my WaterOneFlow Web Service will have a URL path of <http://sandbox08.uwrl.usu.edu/LittleBearRiver/>. If I wanted my WaterOneFlow Web Service to have a different URL, I would first have to set up an appropriate domain name and then set up a website in IIS to handle that domain name. For example, I could register a domain name called “icewater.usu.edu” and have it pointed at this same machine. I would then create a website in IIS called “icewater” that would handle the “icewater.usu.edu” domain. Then, I would follow the steps above to register the WaterOneFlow Web Service under the “icewater” website in IIS. The path for my WaterOneFlow Web Services would then be <http://icewater.usu.edu/LittleBearRiver/>. If you wish to register alternate domain names for your HydroServer, you will need to work with the individuals in charge of your IT infrastructure.

12. Next, we need to tell the WaterOneFlow Web Service how to connect to your ODM database. Make sure that your WaterOneFlow Web Service application is selected in IIS by clicking on its name in the tree view at the left of the IIS Manager form. Under the “ASP.NET” icon group in the middle section of the IIS Manager form, double click on the “Connection Strings” icon. This will open the connection strings editor within the IIS Manager.



13. Double click on the “ODDB” line. On the form that pops up, you need to edit the items to match the location and user credentials for your ODM database. Click the radio button next to “SQL Server” to specify that you will be using a SQL Server database. Then, supply the name of your SQL Server in the “Server:” text box and the name of your ODM database in the “Database:” text box. Your form should now look something like the following:

14. Click on the radio button next to “Specify credentials.” You will notice that the “Set...” button is now enabled. Click the “Set...” button. The following window will open:

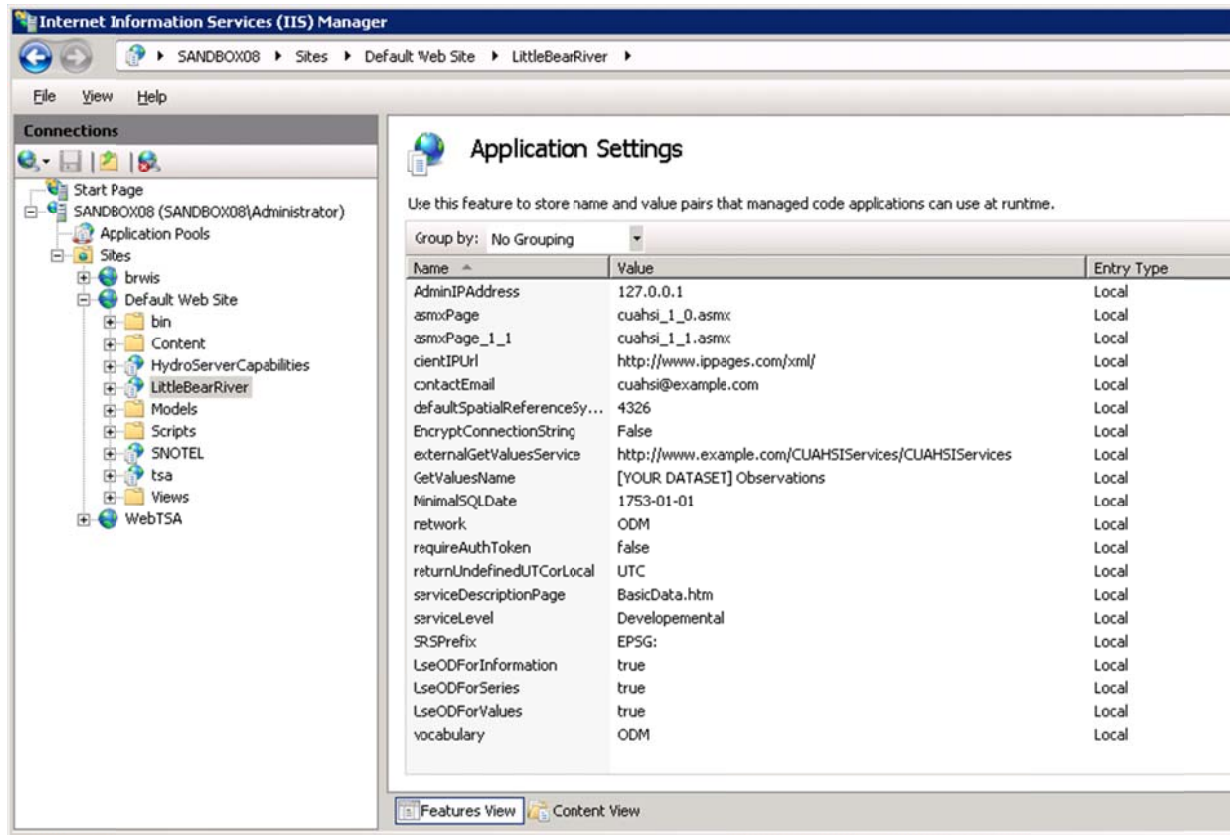
15. Enter your SQL Server user name in the “User name:” text box and your SQL Server password in the “Password” and “Confirm password” text boxes. Click the “OK” button to return to the “Edit Connection String form. Your form should now look something like the following:

The screenshot shows the 'Edit Connection String' dialog box. It has a title bar with a question mark and a close button. The 'Name' field contains 'ODDB'. There are two radio buttons: 'SQL Server' (selected) and 'Custom'. Under 'SQL Server', there are fields for 'Server' (waterdata.usu.edu) and 'Database' (LittleBearRiverODM). Below these is a 'Credentials' section with two radio buttons: 'Use Windows Integrated Security' and 'Specify credentials' (selected). The 'Specify credentials' section has a text box containing 'ODMWebService' and a 'Set...' button. The 'Custom' section has a text box containing the full connection string: 'Server=waterdata.usu.edu;Database=LittleBearRiverODM;User ID=ODMWebService;Password=.....'. At the bottom are 'OK' and 'Cancel' buttons.

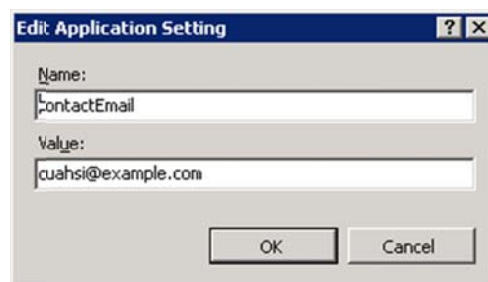
16. In this example, I am connecting to an ODM database called “LittleBearRiver” on a SQL Server machine called “waterdata.usu.edu.” I have created a SQL Server user with read only access to that database called “ODMWebService.” You will notice that the connection string for your database is automatically created for you in the text box at the bottom of the form. Click the “OK” button when you are finished.

NOTE: Since the WaterOneFlow Web Service is a public web application, you want to make sure that you create a SQL Server user within your ODM database that has read only access for use in specifying the connection from the WaterOneFlow Web Service. You can do this within SQL Server Management Studio.

17. Next we need to set a few application settings. Click on the name of your WaterOneFlow service application in the tree view of the IIS Manager (e.g., “LittleBearRiver”) to exit the Connection Strings Editor. Under the “ASP.NET” icon group in the middle section of the IIS Manager form, double click on the “Application Settings” icon. This will open the Application Settings Editor within the IIS Manager.



18. You will notice that there are several items in the list of application settings. You will need to edit some of these to make your WaterOneFlow Web Service work. To edit these items, double click on them. For example, double click on the “contactEmail” setting. The following window will open:



19. You need to set the text in the “Value” text box to what you want your WaterOneFlow Web Service to use. **Do not edit the text in the “Name” text box!** When you have edited the text in the “Value” box (e.g., change the value of the contact email to that of someone that is in charge of the Web service), click the OK button to continue. You need to set the values for several of the application settings in the list using this procedure. The following is a list of the application settings that need to be set and a description of what their value should be set to. You can customize the text of these application settings to meet your needs.

- **contactEmail** – this is the email for a person that is in charge of the web service. Example: “jeff.horsburgh@usu.edu”

- **defaultSpatialReferenceSystemSRSID**– this is the ID for the Spatial Reference System used by your ODM database for the latitude and longitude coordinates of monitoring sites. The default value of “4326” specifies that your latitude and longitude coordinates have a datum of WGS84. Refer to the Sites and SpatialReferences tables in your ODM database for help in setting this value. Other commonly used values include “4267” for NAD27 and “4269” for NAD83.
- **GetValuesName**– this text will be displayed at the top of the Web page that opens when users navigate to your web service. You should remove the part that says “[YOUR DATASET]” and replace it with something that describes your ODM database. Example: “Little Bear River Observations.”
- **network**– this is a short, unique name for the “network” of monitoring sites that is stored in your ODM database. It should be a single word and should not include any special characters (e.g., no “-“, “_“, “:“, “*”, etc.). Example: “LittleBearRiver.”

NOTE: The network setting affects how you ask for site information when calling the ODM web service from a program. Specifically, the web service methods affected are GetSites/GetSitesObject, GetSiteInfo/GetSiteInfoObject, and GetValues/GetValuesObject. The syntax that any WaterOneFlow service call must follow regarding site information, is “Network:SiteCode”, where Network refers to the Network prefix, and SiteCode is a unique identifier for the site in the network. For example, if I’m a program and I am asking a WaterOneFlow web service for information about site “08158000” in the USGS NWIS network(which happens to be the Colorado River at Austin), I would ask for “NWIS:08158000.”

- **serviceDescriptionPage** – this is the name of an HTML page that you can create for describing your data. The default value for this setting is “BasicData.htm.” You can create an HTML page called “BasicData.htm” that describes the data stored in your ODM database and place it in the main web service application directory. When you do this, the WaterOneFlow web service will create a link to that page on the main page of the Web service. You can put whatever content you want in your HTML description page. You can name the HTML file whatever you want, but you must change this application setting to match the name of the file that you create.
- **vocabulary** – this is a short name for the “vocabulary” that is being used to describe the variables in your observation network. It should be a single word and should not include any special characters. Example: “LBR.”

NOTE: The vocabulary is a short identifier for the vocabulary used to describe the variables in your observations network. For example, if I have a variable called “Texonium” with a unique code of “TXN”, I’d like to know if “TXN” is a term that the EPA uses, or perhaps the USGS uses, or perhaps I just made it up and it’s my own vocabulary. The Vocabulary doesn’t have to be the same as the Network. For example, suppose I like the way the USGS describes variables within their NWIS system, and I’ve used the same variable codes in my own ODM database. My Network prefix might be “MyNetwork,” but my Vocabulary prefix could be “NWIS,” indicating that I have adopted the NWIS vocabulary. The syntax for using Vocabulary when calling a WaterOneFlow web service is “Vocabulary:VariableCode”, where VariableCode is the unique identifier for the variable of interest, and Vocabulary refers to the Vocabulary prefix identifying the system to which that given VariableCode belongs. The web service methods affected by Vocabulary are GetVariableInfo/GetVariableInfoObject, and GetValues/GetValuesObject.

20. Congratulations! Once you have set the values of the above application settings, your WaterOneFlow Web Service setup and configuration is complete. You should be able to navigate to your service in a web browser. For this example, the URL for the web service we set up would be <http://sandbox08.uwrl.usu.edu/LittleBearRiver/>.