How to Consume the OWMWDS Web Service

Revision History:

Date	Updated By	Remarks
5/5/2011	P. Clifton	Created
5/12/2011	P. Clifton	Updated with input from Ken

1 Overview ... 3
1.1 Background OWMWDS ... 3
1.2 Related/Reference Documents ... 4
2 OWMWDS Web Service ... 5
2.1 XML Returned ... 5
2.1.1 Well Data ... 5
2.1.2 Return status metadata ... 6
2.2 Sample XML ... 7
2.3 Web Methods ... 8
2.3.1 Select_wells_by_basin (number) ... 8
2.3.2 Select Wells by County (code) ... 8
2.3.3 Select Wells by (Latitude/Longitude) Location ... 9
2.3.4 Select Wells by Region (code) ... 9
2.3.5 Select Wells by TRSQQ ... 10
2.3.6 Select Wells by Watermaster District ... 10
2.3.7 Select Wells by WRD Identifiers ... 11
2.3.8 Select Wells by WWID ... 11
3 How to Use OWMWDS Web Method Example ... 12
3.1 Example Solution Setup ... 12
3.2 Project Library to Access Web Service Web Methods ... 13
3.3 ASP Page – Default page ... 15
3.4 ASP Page – Query Well by Basin ... 16
3.4.1 ObjectDataSource - ods_owmwds_ws Properties ... 18
3.4.2 GridView1 ... 19
3.4.3 Btn_Clear – Click event ... 21
3.4.4 Run the Page ... 21
4 Appendix A – ASP Code ... 21
4.1 Default.aspx ... 21
4.2 Owmwds_well_query_by_basin ... 22

1 Overview

This document describes each of the web methods provided in the OWMWDS web service.

This document also uses an ASP.net/C# example to show how to consume one of the web methods in the OWMWDS. For this example, a simple ASP.net page using C# will be created to search and display data for the 'Select Wells by Basin' web method. This will involve creating:

- A project library to access the web methods in the web service
- An ASP.net page to allow searching the OWMWDS repository and displaying a list of wells

The examples in this document were created with Visual Studio 2008 with ASP.net 3.5.

This is a Phase 1 product in which cooperators can review and use as a discussion starting point.

1.1 Background OWMWDS

Water well data, and their locations, are collected by a variety of public and private entities in Oregon at all levels—local, state, and federal— with little coordination. The Oregon Water Resources Department (OWRD) is the custodian of well logs filed by well drillers when they drill, deepen, or abandon a well. Location information provided by most well logs is, at best, defined by a Public Land Survey quarter-quarter section description. OWRD locates wells using GPS for inspections or ground water studies. Additionally, staff from other government and private organizations visit wells and determine more precise location information for their purposes. The data is not collected or stored according to a standard nor is it easily available to be shared amongst the various users. This can impede and/or increase the cost of development that requires accurately located ground water data.

Well locations portrayed on USGS quadrangle maps, the primary source for the Digital Line Graph (DLG) data files and, subsequently, the NHD (National Hydrography Dataset), are not reliable, either. The maps are dated and most wells were not shown on a given quad map when they were compiled. For example, only representative wells were portrayed on the map where they were part of a large cluster of closely spaced wells (Thompson. p. 42).

The purpose of Oregon Water & Monitoring Well Data Standard (OWMWDS) is to specify a common method for locating and identifying water & monitoring wells in the state. The OWMWDS repository has data from several agencies including OWRD. OWRD is the steward of the repository.

For Phase 1, there are three sources of OWRD well information data that will be used to populate the OWMWDS repository. These sources are ranked by the quality of the data and will be loaded and updated in the OWMWDS repository with "increasing quality":

.

- Well Log – We will load only the "New" well logs that have a lat/long. During the loading we will cross check their lat/long with the specified Quarter/Quarters. If the QQ's match, then the quality of this data will be considered to be 1320 feet. If they do not match, it will be considered to be 5280 feet.

- Well Inspections – Wells existing in this data set will be considered to have a quality of 100 feet. New wells will be added the repository, and existing wells will be updated with the updated quality code.

- Groundwater Located Wells – Wells existing in this data set will have a quality set as the value entered in this data. This should be the most accurate location.

Since each well data source could potentially update the same well, the information of a less quality ranking source will not overwrite the information if a better quality source had updated it previously so that the most accurate (quality) location for the well is kept.

Only new water and monitoring wells (not abandoned and version 1) with latitude and longitude data will be put into the repository. Geotechnical holes will not be included.

OWRD well data will be loaded nightly.

Oregon Water Resources Department (OWRD) has provided a web service with several web methods to query the OWMWDS repository. These include:

- Query by WRD Identifiers
- Query by TRSQQ
- Query by County (code) – this is based on the county of the lat./long. of the well
- Query by Basin (number)
- Query by Latitude/Longitude Location
- Query by Watermaster District
- Query by Region (code)
- Query by WWID – unique OWMWDS repository ID

1.2 Related/Reference Documents

Document Name	**Description**
ORWaterWellDataExchangeStandard_v1Draft.pdf	Document that defines the standard

2 OWMWDS Web Service

The OWMWDS web service provides web methods to access the public well data in the repository. As this is public data, user credentials are not needed to access the data using the web service.

The web service is:
http://apps.wrd.state.or.us/apps/gw/owmwds_ws/owmwds_ws.asmx

2.1 XML Returned

All web methods return the same XML format:

- Well Data for each well returned
- Return Status metadata

2.1.1 Well Data

The data returned as XML for each well is defined in the OWMWDS standards document as:

Name	Type	Width	Description
wwID	Long Integer		Unique feature identifier: sequential, unique ID generated by the horizontal steward.
wwLongitude	Double	13.6	Longitude value of the well location expressed in "degrees-decimal", with a preceding negative sign (World Geodetic Spheroid 1984)
wwLatitude	Double	13.6	Latitude value of the well location expressed in "degrees-decimal" (World Geodetic Spheroid 1984)
wwHorizontalAcc	Double	6.2	Horizontal accuracy assessment of the preceding latitude and longitude coordinate pair.
wwLabel	Text	15	Oregon Water Resources Dept. issued well label number. Also known as the "well tag" or "well ID number".
wwOtherID	Text	50	Collecting entity well identifier.
wwUser	Text	25	Name of the person creating this data record.
wwAgency	Text	50	Name of the agency that the person represents
wwRecordDate	Date		Date that this record was created
wwLogCounty	Text	4	For letter county code abbreviation used at the beginning of the Oregon Waster Resources Dept. well log id (all upper case). For example, , "MARI" in the well log id "MARI99999". The county-code is assigned based on the county the driller entered on the well log. It may not correspond to the actual county that contains the well.
wwLogNbr	Long Integer		The second (numerical) part of the OWRD well log number. For example, "99999" in the well log id "MARI99999". This number is assigned by OWRD when the well log is filed at the Department. There may be more than one well log per well. OWRD assigns a well log id to each well log. Use the number of the first well log (first construction), if known.
wwStartCardNbr	Long		OWRD "start card" number. A startcard is a notice of

	Integer		intent to drill a well that must be submitted to OWRD by a well constructor. Startcards have only been requed since 1989. Therefore, there will be a different startcard for each well log filed after 1989. Use the startcard number of the first well log (first construction).
wwWellName	Text	50	Name of the well. Local reference or agency applied name.
wwOwner	Text	100	Original owner's name or company name on the well log.
wwAddress	Text	100	Street address of the property (including state and zip code) where the well is located. For example, "1 Main St., Salem OR 97301"
wwTaxlot	Text	29	The statewide standard map and tax lot number (ORTaxlot) as defined in the Oregon Casdastral Data Exchange Standard is preferred.
wwTypeofUse	Text	100	Primary use of the well.
wwCompletionDate	Date		The original completion date of the well expressed. If only, the month and year are known use the last day of the month, e.g. "6/30/2001" for "June 2001". If only the year is known use "December 31", e.g. "13/31/2001" for "2001".
wwCompletedDepth	Double	6.2	Completed depth as reported on the (first) well log (in feet).
wwCasingDiam	Integer		Surface casing diameter of the well, or the maximum casing diameter, to the nearest inch.
wwDepth2Water	Double	6.2	Report depth to first water, to the nearest foot.
wwStaticWater	Double	6.2	Static water level, in feet below the land surface, as reported on the (first) well log.
wwDrillingMethod	Text	100	Method of drilling
wwLandSurfaceElevation	Double	6.2	Eleveation of the well at the ground surface (in feet)
wwElevationRemarks	Text	100	Remarks describing where the preceding elevation value was taken with respect to the well.

2.1.2 Return status metadata

The metadata returned in the XML is defined as:

Name	**Type**	**Description**
Return_message	Text	Message that the query was successful or not, if not this will contain the error message
Return_record_count	Long integer	Number of wells actually returned (1 page worth of data)
Total_record_count	Long integer	Total number of wells meeting the criteria
Return_code	Long integer	0 if successful, else an error occurred
Query_parameters (collection) • Parameter_name • Parameter_value	 Text Text	Parameter name/value pairs. Note: numeric values are converted to string before putting them into the XML

2.2 Sample XML

Data is shown in blue. Note: this XML was generated from test data and may not represent actual data for the specific well log shown.


```
<ww_owmwds_list>
    <wells>
        <ww_owmwds>
          <wwID>434879</wwID>
          <wwLongitude>-120.033333</wwLongitude>
          <wwLatitude>45.266667</wwLatitude>
          <wwHorizontalAcc>5280.00</wwHorizontalAcc>
          <wwLabel>77303</wwLabel>
          <wwOtherID>386039</wwOtherID>
          <wwUser>JOEL M PLAHN</wwUser>
          <wwAgency>OWRD - Oregon Water Resources Department</wwAgency>
          <wwRecordDate>2006-08-29T10:20:21.7</wwRecordDate>
          <wwLogCounty>GILL</wwLogCounty>
          <wwLogNbr>50181</wwLogNbr>
          <wwStartCardNbr>184874</wwStartCardNbr>
          <wwWellName/>
          <wwOwner>ANDREW PITT</wwOwner>
          <wwAddress>24165 HWY 206, CONDON</wwAddress>
          <wwTaxlot>1401</wwTaxlot>
          <wwTypeofUse>DOMESTIC</wwTypeofUse>
          <wwCompletionDate>2006-07-26T00:00:00</wwCompletionDate>
          <wwCompletedDepth>280.00</wwCompletedDepth>
          <wwCasingDiam>8</wwCasingDiam>
          <wwDepth2Water>171.00</wwDepth2Water>
          <wwStaticWater>40.00</wwStaticWater>
          <wwDrillingMethod>ROTARY AIR</wwDrillingMethod>
          <wwLandSurfaceElevation xsi:nil="true"/>
          <wwElevationRemarks/>
        </ww_owmwds>
.
.
.
    </wells>
    <return_status>
        <return_message>Query 'Select_wells_by_basin' successful</return_message>
        <return_record_count>10</return_record_count>
        <total_return_count>42</total_return_count>
        <return_code>0</return_code>
        <query_parameters>
          <query_parameter>
            <parameter_name>basin_nbr</parameter_name>
            <parameter_value>5</parameter_value>
          </query_parameter>
          <query_parameter>
            <parameter_name>wwAgency</parameter_name>
            <parameter_value/>
          </query_parameter>
          <query_parameter>
            <parameter_name>startRowIndex</parameter_name>
            <parameter_value>0</parameter_value>
```

```
            </query_parameter>
            <query_parameter>
              <parameter_name>pageSize</parameter_name>
              <parameter_value>10</parameter_value>
            </query_parameter>
            <query_parameter>
              <parameter_name>SortExpression</parameter_name>
              <parameter_value>wwID</parameter_value>
            </query_parameter>
          </query_parameters>
    </return_status>
</ww_owmwds_list>
```


2.3 Web Methods

These are the web methods provided by the web service.

2.3.1 Select_wells_by_basin (number)

Selects wells for a given basin (number).

Parameter Name	Type	Comments
Basin_nbr	Int	Unique number that represents the basin.
wwAgency	String	select wells data for the agency the data originated from, partial words are accepted
startRowIndex	Int	Defaults to 0 if no value so that first record in the list is the starting row, used for paging
pageSize	Int	Defaults to max integer size if no value so all records are in the page, used for paging
SortExpression	String	Defaults to wwID if no value, used for sorting

Basin numbers are:

1 – North Coast	2 – Willamette	3 – Sandy
4 – Hood	5 – Deschutes	6 – John Day
7 – Umatilla	8 – Grande Ronde	9 – Powder
10 – Malheur	11 – Owyhee	12 – Malheur Lake
13 – Goose & Summer Lake	14 – Klamath	15 – Rogue
16 – Umpqua	17 – South Coast	18 – Middle Coast

2.3.2 Select Wells by County (code)

Selects wells from the repository for a given county. This is based on the county the lat./long. is located not the county code for the well log.

Parameter Name	Type	Comments
Location_county	String	For letter county code abbreviation. This is the county associated with the latitude/longitude location and not the alphabetic portion (county code) assigned on the well log
wwAgency	String	select wells data for the agency the data originated from, partial words are accepted
startRowIndex	Int	Defaults to 0 if no value so that first record in the list is the starting row, used for paging
pageSize	Int	Defaults to max integer size if no value so all records are in the page, used for paging
SortExpression	String	Defaults to wwID if no value, used for sorting

2.3.3 Select Wells by (Latitude/Longitude) Location

Selects wells within a distance from a given latitude/longitude point. The point plus the distance (radius) from the point create a geographic area. Wells that fall in the geographic area are returned.

Parameter Name	Type	Comments
wwLatitude	Decimal	Degrees decimal
wwLongitude	Decimal	Degrees decimal
Buffer_distance_in_feet	Decimal	Distance in feet from the lat/long point. This create the radius giving a circular geographic area around the point and all wells within the are will be selected
wwAgency	String	select wells data for the agency the data originated from, partial words are accepted
startRowIndex	Int	Defaults to 0 if no value so that first record in the list is the starting row, used for paging
pageSize	Int	Defaults to max integer size if no value so all records are in the page, used for paging
SortExpression	String	Defaults to wwID if no value, used for sorting

Note: any valid decimal can be sent for the latitude/longitude values. However, the repository only contains well data with latitudes/longitudes values:

- Latitude – between 41.0 and 47.0 degrees decimal
- Longitude – between -125.0 and -116.0 degrees decimal

2.3.4 Select Wells by Region (code)

Selects wells from the repository for a given region. Partial codes are accepted.

Parameter Name	Type	Comments
Wm_region_code	String	1-3 character OWRD region code, partial codes can be accepted
wwAgency	String	select wells data for the agency the data originated from, partial words are accepted
startRowIndex	Int	Defaults to 0 if no value so that first record in the list is the starting row, used for paging
pageSize	Int	Defaults to max integer size if no value so all records are in the page, used for paging
SortExpression	String	Defaults to wwID if no value, used for sorting

OWRD Region codes are:

E – East NC – North Central NW – Northwest
SC – South Central SW – Southwest

Partial codes are accepted:

C – Central N – North S – South W – West

For map of regions: http://wrd.state.or.us/OWRD/offices.shtml

2.3.5 Select Wells by TRSQQ

Selects wells from the repository by the Township, Range, Section, Quarter 160 and Quarter 40 location information.

Township and Range lines were defined by Public Land Surveyors in the late 1800's. They define a grid for Oregon and Washington from which all other land surveys are ultimately based. Townships run North to South and Ranges run East to West. In Oregon Township numbers span from 41 South to 9 North and Range numbers span from 16 West to 50 East. While slightly confusing, the combination of a Township number and a Range number uniquely identify a square called a Township. Each Township is broken down into 36 sections of approximately one-square-block of land containing 640 acres. Each section is divided into 160-acre quarters and those are divided again into quarters (40 acres).

Parameter Name	Type	Comments
Meridian	String	2-3 character code representing the principal meridian lines used by Public Land Surveyors as starting point for sectioning the land: BM – Boise (ID) meridian, MDM – Mount Diablo (CA) meridian WM – Willamette (OR & WA) meridian
Township	Decimal	
Township_char	String	N for North or S for South
Range	Decimal	
Range_char	String	E for East or W for West
Sctn	Int	Section (1 – 36)
Qtr160	String	NE, NW, SE, SW
Qtr40	String	NE, NW, SE, SW
wwAgency	String	select wells data for the agency the data originated from, partial words are accepted
startRowIndex	Int	Defaults to 0 if no value so that first record in the list is the starting row, used for paging
pageSize	Int	Defaults to max integer size if no value so all records are in the page, used for paging
SortExpression	String	Defaults to wwID if no value, used for sorting

2.3.6 Select Wells by Watermaster District

Selects wells from the repository in a given water master district.

For current list of watermasters: http://wrd.state.or.us/OWRD/offices.shtml

Parameter Name	Type	Comments
Wm_district	Int	Watermaster district
wwAgency	String	select wells data for the agency the data originated from, partial words are accepted
startRowIndex	Int	Defaults to 0 if no value so that first record in the list is the starting row, used for paging
pageSize	Int	Defaults to max integer size if no value so all records are in the page, used for paging
SortExpression	String	Defaults to wwID if no value, used for sorting

2.3.7 Select Wells by WRD Identifiers

Selects wells from the repository by the Oregon Water Resources identifiers.

Parameter Name	Type	Comments
wwLogCounty	String	For letter county code abbreviation used at the beginning of the Oregon Waster Resources Dept. well log id (all upper case). For example, , "MARI" in the well log id "MARI99999". The county-code is assigned based on the county the driller entered on the well log. It may not correspond to the actual county that contains the well.
wwLogNbr	Int	The second (numerical) part of the OWRD well log number. For example, "99999" in the well log id "MARI99999". This number is assigned by OWRD when the well log is filed at the Department. There may be more than one well log per well. OWRD assigns a well log id to each well log. Use the number of the first well log (first construction), if known.
wwLabel	String	Oregon Water Resources Dept. issued well label number. Also known as the "well tag" or "well ID number". Note: only include the number portion of the well tag number, exclude the 'L' or 'L-' prefix preceding the well tag number.
wwStartCardNbr	Int	OWRD "start card" number. A startcard is a notice of intent to drill a well that must be submitted to OWRD by a well constructor. Startcards have only been requed since 1989. Therefore, there will be a different startcard for each well log filed after 1989. Use the startcard number of the first well log (first construction).
startRowIndex	Int	Defaults to 0 if no value so that first record in the list is the starting row, used for paging
pageSize	Int	Defaults to max integer size if no value so all records are in the page, used for paging
SortExpression	String	Defaults to wwID if no value, used for sorting

2.3.8 Select Wells by WWID

Selects wells from the repository for a given repository ID (wwID). The wwID is unique so the list of wells returned will either just one well (well was found for the ID) or no wells (well ID was not found). However, the web service data always returns a 'list' of wells so in this case the list may contain at most would be one well.

Parameter Name	Type	Comments
wwID	Int	Unique ID assigned to each well in the repository
startRowIndex	Int	Defaults to 0 if no value so that first record in the list is the starting row, used for paging
pageSize	Int	Defaults to max integer size if no value so all records are in the page, used for paging
SortExpression	String	Defaults to wwID if no value, used for sorting

3 How to Use OWMWDS Web Method Example

This example shows how to implement the 'Select_wells_by_basin' web method provided by the OWMWDS web service. This example was created using Visual Studio 2008 and ASP.net 3.5 with C#. The Visual Studio solution contains two projects:

- One for ASP pages will have 2 pages:
  - Default.aspx – main menu page with hyperlink to query page(s);
  - owmwds_well_query_by_basin.aspx – uses the project library to access the web methods
- One for project library to access the web service web methods

3.1 Example Solution Setup

<u>In Visual Studio:</u>
Create → New → Web Project called 'owmwds_well_query_example'.

This creates the solution and project 'owmwds_well_query_example' with a blank 'Default.aspx' page. Add an additional web page called: 'owmwds_well_query_by_basin.aspx'.

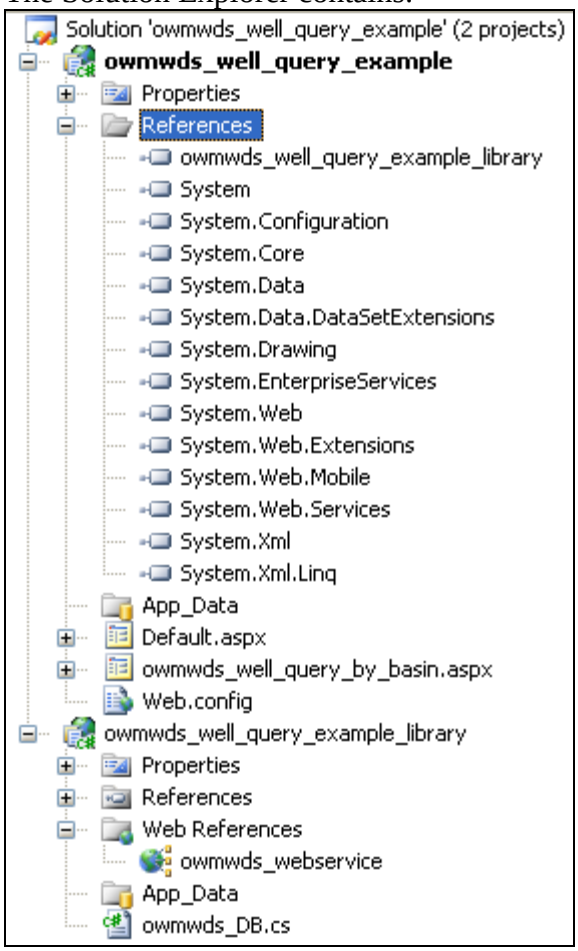
Add another web project to the solution called 'owmwds_well_query_example_library'

This creates the project 'owmwds_well_query_example_library' also with a 'Default.aspx'.
In the library project:

- Delete the Default.aspx and web.config as these are not needed.
- Add a class 'owmwds_DB.cs' which will contain the code for accessing the web service.
- Add reference to System.Web.Services, if not already there
- Add a web reference with name of 'owmwds_webservice to the OWMWDS web service at: http://apps.wrd.state.or.us/apps/gw/owmwds_ws/owmwds_ws.asmx

In the ASP page project ('owmwds_well_query_example') add a reference to the library project.

The Solution Explorer contains:



Note: the owmwds_well_query_example_library project contains the same System references as the owmwds_well_query_example project. These System references were added by default when the projects were created.

3.2 Project Library to Access Web Service Web Methods

Project library 'owmwds_well_query_example' accesses the OWMWDS web service web methods and provides the methods that the ASP page will use.

In this example, an 'ObjectDataSource' will be used as the data source for a GridView that will display the data. A method is needed for selecting the data. When paging is used, the method to select the data only returns a page worth of data, so to support paging, a second select method is used for the total record count.

Since the XML contains the well data and the return status information including the total record count, methods will be created to separate the well data from the return status metadata that is needed (in this case only total_record_count from the metadata is used).

Since this project library may be used elsewhere with or without paging and/or sorting, a base method for selecting well data (no paging and no sorting) with overrides for paging and sorting will be included.
For this example, each web method in the web service will provide a base method to select data with 3 overloads will be created, plus the record count method. This will support using the methods with or without paging and/or sorting:

- Select data method with no paging and no sorting
- Override select data method with sorting, but no paging
- Override select data method with paging, but no sorting
- Override select data method with both paging and sorting
- Select record count method – no overrides as this always needs to return the total record count

The well data returned by the web service is basically an array of wells so the select data methods will just get the data from the web service and stuff it into an array. In addition to putting the well data into an array, the total record count will be read from the metadata and put into a variable so the select record count method can just read if from the variable.

<u>The owmwds_DB.cs code for selecting wells by basin looks like:</u>

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.ComponentModel;

namespace owmwds_well_query_example_library
{
    [DataObjectAttribute]
    public class owmwds_DB
    {
        // Variable to hold the total record count returned in the XML
        // Set by the Select methods that call the web service
        private int _total_records_by_basin;

        // This method just return the count from the variable
        // Variable was set by the Select_wells_by_basin
        public int Select_wells_by_basin_count(int? basin_nbr, String
wwAgency) { return _total_records_by_basin; }

        // These methods call the web service, return the well data and
set the
        // total record count into a variable
        [DataObjectMethodAttribute(DataObjectMethodType.Select, true)]
        public owmwds_webservice.ww_owmwds[] Select_wells_by_basin(int?
basin_nbr, String wwAgency)
        {
            return Select_wells_by_basin(basin_nbr, wwAgency,
int.MaxValue, 0, string.Empty);
        }
```

```
        public owmwds_webservice.ww_owmwds[] Select_wells_by_basin(int?
basin_nbr, String wwAgency, int maximumRows, int startRowIndex)
        {
            return Select_wells_by_basin(basin_nbr, wwAgency,
maximumRows, startRowIndex, string.Empty);
        }
        public owmwds_webservice.ww_owmwds[] Select_wells_by_basin(int?
basin_nbr, String wwAgency, String SortExpression)
        {
            return Select_wells_by_basin(basin_nbr, wwAgency,
int.MaxValue, 0, string.Empty);
        }
        // This is the method that actually calls the web service
        // returns the well data and sets the total count in a variable
        public owmwds_webservice.ww_owmwds[] Select_wells_by_basin(int?
basin_nbr, String wwAgency, int? maximumRows, int? startRowIndex,
String SortExpression)
        {

            if (!basin_nbr.HasValue) return null;
            if (String.IsNullOrEmpty(wwAgency)) wwAgency =
string.Empty;
            if (!maximumRows.HasValue) maximumRows = int.MaxValue;
            if (!startRowIndex.HasValue) startRowIndex = 0;

            owmwds_webservice.owmwds_ws p_db = new
owmwds_webservice.owmwds_ws();

            owmwds_webservice.owmwds_well ws_response =
p_db.Select_wells_by_basin(basin_nbr, wwAgency, startRowIndex,
maximumRows, SortExpression);

            if (ws_response.return_status.return_record_count == 0)
return null;

            _total_records_by_basin =
ws_response.return_status.total_return_count.Value;

            return ws_response.wells;
        }
    }
}
```

3.3 ASP Page – Default page

For this example, there are two ASP pages. The ‘Default.aspx’ is just a ‘main menu’ page with a link to the ‘Query Wells by Basin Page’ (additional query pages can be added for the other web methods). The second ASP page is the ‘Query Wells by Basin Page’.

The Default page is simply a ‘main menu’ page with hyperlink(s) to the actual query pages.

This page has some text and hyperlink(s) to the query page(s).

Oregon Water and Monitoring Well Standard Repository

The 'Oregon Water and Monitoring Well Standard' combines well data from several agencies into a single repository based on an agreed apon list of standard well attriubes. Each agency may contribute data from one or more sources. This can result in multiple records returned for the same well though with more or less attributes populated.

The repository can be queried by:

- Basin

See the Appendix for the ASP code for this page.

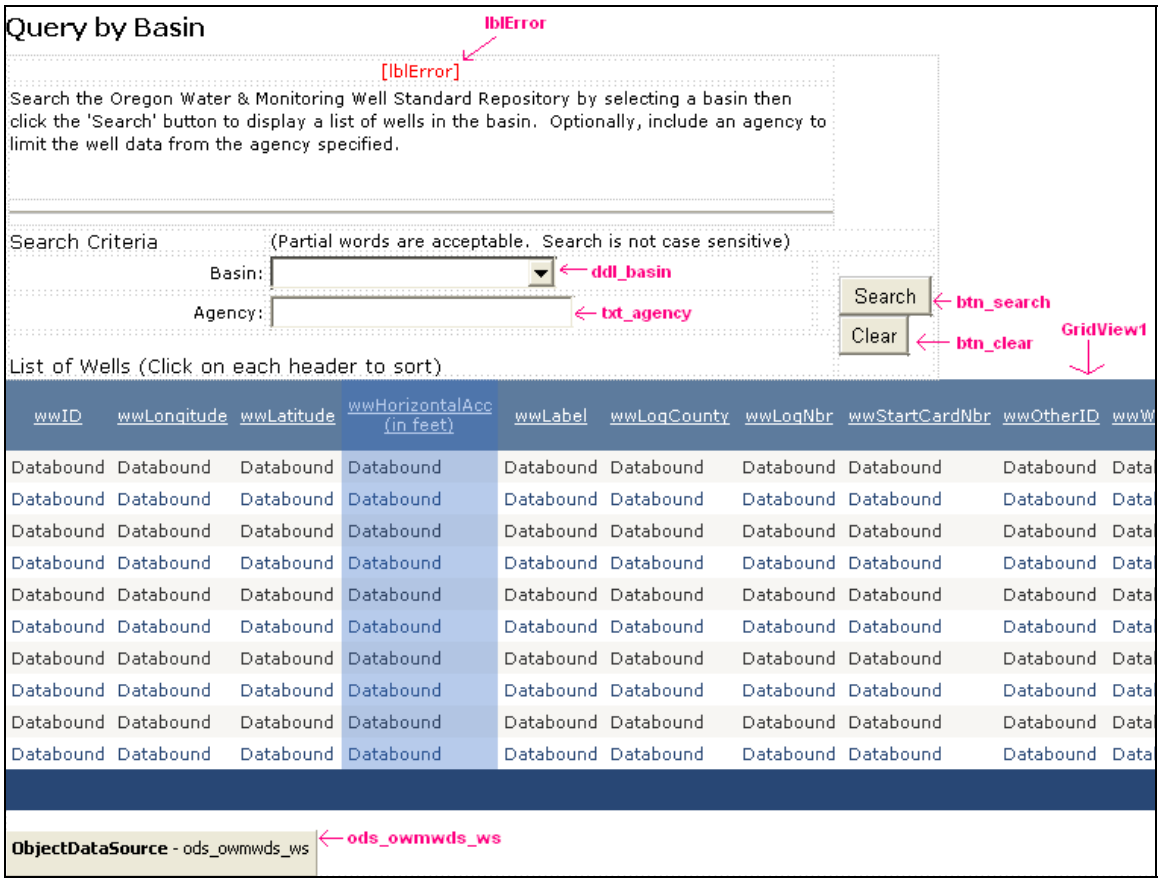
3.4 ASP Page – Query Well by Basin

This page allows the user to display a list of wells for the selected basin. In addition to selecting a basin, the user can specify the agency the data originated from.

The query page will contain:

- Criteria controls the user can set criteria in this case select a basin and also include the agency
- Search button – to run the query and display the results
- Clear button – to clear the criteria controls
- Grid View to display the data
- ObjectDataSource – the grid is bound to for data
- Also, plain text is also displayed on the page for labels, explanations, etc.

Here is the Design View of the 'Query by Basin' ASP page (control names are in pink).



See the Appendix for complete ASP code for this page.

Since the basin numbers are in a drop down list, the Query by Basin ASP page does not have any validations.

The items in the basin drop down list were just manually added to the control:

<u>Basin numbers are:</u>

1 – North Coast	2 – Willamette	3 – Sandy
4 – Hood	5 – Deschutes	6 – John Day
7 – Umatilla	8 – Grande Ronde	9 – Powder
10 – Malheur	11 – Owyhee	12 – Malheur Lake
13 – Goose & Summer Lake	14 – Klamath	15 – Rogue
16 – Umpqua	17 – South Coast	18 – Middle Coast

```
<asp:DropDownList ID="ddl_basin" runat="server">
    <asp:ListItem></asp:ListItem>
    <asp:ListItem Value="1">1 - North Coast</asp:ListItem>
    <asp:ListItem Value="2">2 - Willamette</asp:ListItem>
    <asp:ListItem Value="3">3 - Sandy</asp:ListItem>
    <asp:ListItem Value="4">4 - Hood</asp:ListItem>
    <asp:ListItem Value="5">5 - Deschutes</asp:ListItem>
    <asp:ListItem Value="6">6 - John Day</asp:ListItem>
    <asp:ListItem Value="7">7 - Umatilla</asp:ListItem>
    <asp:ListItem Value="8">8 - Grande Ronde</asp:ListItem>
```

```
    <asp:ListItem Value="9">9 - Powder</asp:ListItem>
    <asp:ListItem Value="10">10 - Malheur</asp:ListItem>
    <asp:ListItem Value="11">11 - Owyhee</asp:ListItem>
    <asp:ListItem Value="12">12 - Malheur Lake</asp:ListItem>
    <asp:ListItem Value="13">13 - Goose & Summer Lake</asp:ListItem>
    <asp:ListItem Value="14">14 - Klamath</asp:ListItem>
    <asp:ListItem Value="15">15 - Rogue</asp:ListItem>
    <asp:ListItem Value="16">16 - Umpqua</asp:ListItem>
    <asp:ListItem Value="17">17 - South Coast</asp:ListItem>
    <asp:ListItem Value="18">18 - Middle Coast</asp:ListItem>
</asp:DropDownList>
```

3.4.1 ObjectDataSource - ods_owmwds_ws Properties

After configuring the data source (Select_wells_by_basin method) and binding the basin and agency controls to the basin_nbr and wwAgency parameters, set the SelectCountMethod to ‘Select_wells_by_basin_count’ and set the SortParameterName to SosrtExpression.

Note: ‘maximumRows’ and ‘startRowIndex’ used here are the default parameter names already set on the ‘ObjectDataSource’ for MaximumRowsParameterName and StartRowIndexParameterName respectively. These are the same as the parameter names in the Select_well_by_basin method in the owmwds_DB.cs.

The properties look like:

ods_owmwds_ws System.Web.UI.WebControls.ObjectDataSource

(Expressions)	
(ID)	**ods_owmwds_ws**
CacheDuration	Infinite
CacheExpirationPolicy	Absolute
CacheKeyDependency	
ConflictDetection	OverwriteChanges
ConvertNullToDBNull	False
DataObjectTypeName	
DeleteMethod	
DeleteParameters	(Collection)
EnableCaching	False
EnablePaging	**True**
EnableViewState	True
FilterExpression	
FilterParameters	(Collection)
InsertMethod	
InsertParameters	(Collection)
MaximumRowsParameterName	maximumRows
OldValuesParameterFormatString	**original_{0}**
SelectCountMethod	**Select_wells_by_basin_count**
SelectMethod	**Select_wells_by_basin**
SelectParameters	(Collection)
SortParameterName	**SortExpression**
SqlCacheDependency	
StartRowIndexParameterName	startRowIndex
TypeName	**owmwds_well_query_library.owmwds_DB**
UpdateMethod	
UpdateParameters	(Collection)

3.4.2 GridView1

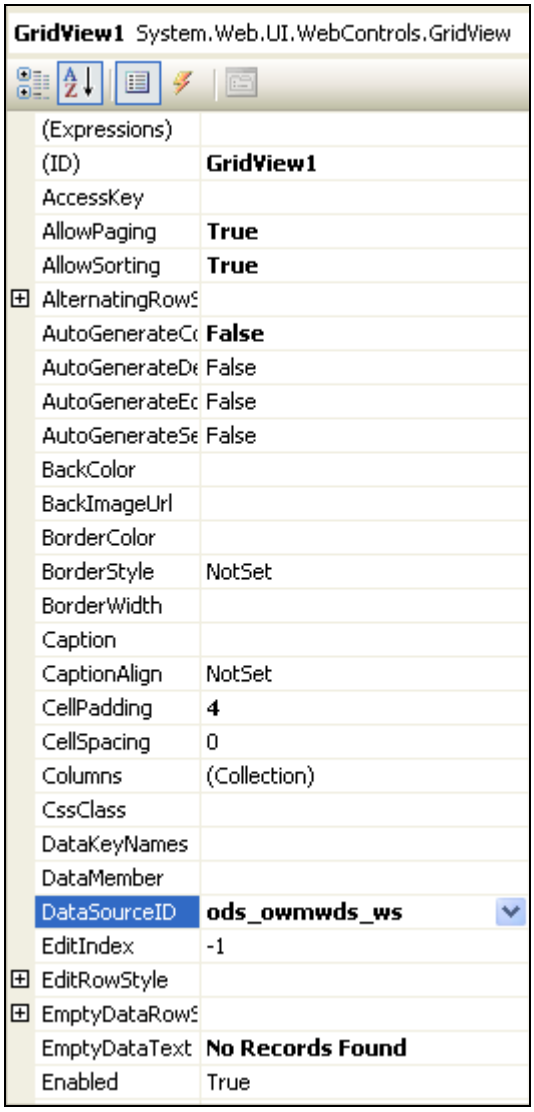
Set the data source for the grid to be the ods_owmwds_ws. Enable paging and sorting.

Add, rearrange or format any columns, as desired. For this example, the columns are:

- wwID
- wwLongitude
- wwLatitude
- wwHorizontalAcc
- wwLabel
- wwLogCounty
- wwLogNbr
- wwStartCardNbr
- wwOtherID
- wwWellName
- wwOwner
- wwAddress

- wwTaxlot
- wwTypeOfUse
- wwCompletionDate - formatted as short date
- wwCompletedDepth
- wwCasingDiam
- wwDepth2Water
- wwStaticWater
- wwDrillingMethod
- wwLandSurfaceElvation
- wwElevationRemarks
- wwUser
- wwAgency
- wwRecordDate – formatted as short date

Here are the primary properties for the GridView in relation to the ObjectDataSource:



GridView1 System.Web.UI.WebControls.GridView

(Expressions)	
(ID)	**GridView1**
AccessKey	
AllowPaging	**True**
AllowSorting	**True**
AlternatingRowS	
AutoGenerateC	**False**
AutoGenerateD	False
AutoGenerateE	False
AutoGenerateS	False
BackColor	
BackImageUrl	
BorderColor	
BorderStyle	NotSet
BorderWidth	
Caption	
CaptionAlign	NotSet
CellPadding	**4**
CellSpacing	0
Columns	(Collection)
CssClass	
DataKeyNames	
DataMember	
DataSourceID	**ods_owmwds_ws**
EditIndex	-1
EditRowStyle	
EmptyDataRowS	
EmptyDataText	**No Records Found**
Enabled	True

Additional properties (not shown) set are:

- Forecolor: #333333
- Gridlines: None

The remaining properties just have the default values.

3.4.3 Btn_Clear – Click event

Add code to clear the controls to the Clear button's click event:

```
ddl_basin.SelectedIndex = -1;
txt_agency.Text = String.Empty;
```

3.4.4 Run the Page

Publish and set up the page to run in your organization or just run the page in debug mode.

Select a basin and click the 'Search' button to display the list of wells for that basin.

4 Appendix A – ASP Code

This appendix contains the full ASP code for the web pages.

4.1 Default.aspx

```
<%@ Page Language="C#" AutoEventWireup="true"
CodeBehind="Default.aspx.cs"
Inherits="owmwds_well_query_example._Default" %>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" >
<head runat="server">
    <title></title>
</head>
<body>
    <form id="form1" runat="server">
    <div>
<table width="100%">
    <tr>
        <td colspan="3" style="font-size: xx-large; font-weight: bold">
        Oregon Water and Monitoring Well Standard Repository
        </td>
    </tr>
    <tr>
        <td colspan="3">
           <p> The 'Oregon Water and Monitoring Well Standard' combines
well data from several
               agencies into a single repository based on an agreed
apon list of standard well
               attriubes.  Each agency may contribute data from
one or more sources. 
```

```
                This can result in multiple records returned for the
same well though with more
                or less attributes populated.  </p>
<p>The repository can be queried by:</p>

            <table style="width: 100%">
                <tr>
                    <td>

                        <asp:HyperLink ID="HyperLink_basin"
runat="server"

NavigateUrl="~/owmwds_well_query_by_basin.aspx"
Target="_self">Basin</asp:HyperLink>

                    </td>
                </tr>
            </table>

       <hr />
        </td>
    </tr>
</table>

    </div>
    </form>
</body>
</html>
```

4.2 Owmwds_well_query_by_basin

```
<%@ Page Language="C#" AutoEventWireup="true"
CodeBehind="owmwds_well_query_by_basin.aspx.cs"
Inherits="owmwds_well_query_example.owmwds_well_query_by_basin" %>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" >
<head runat="server">
    <title></title>
</head>
<body>
    <form id="form1" runat="server">
    <div>
        <table width="100%">
            <tr>
                <td colspan="3">
                    <asp:Label ID="lblError" runat="server"
ForeColor="Red"></asp:Label>
                </td>
            </tr>
            <tr>
                <td colspan="3" style="font-size: xx-large; font-
weight: bold">
                    Oregon Water & Monitoring Well Standard Repository
                </td>
```

```
            </tr>
            <tr>
                <td colspan="3">

                    <asp:HyperLink ID="HyperLink_main_menu"
runat="server"
                        NavigateUrl="~/Default.aspx"
Target="_self">Back to Main Menu</asp:HyperLink>

                </td>
            </tr>            <tr>
                <td colspan="3" style="font-size: x-large; font-weight:
bold">
                    Query Wells by Basin
                </td>
            </tr>
    <tr>
        <td colspan="3">
           <p> Search the Oregon Water &amp; Monitoring Well Standard
Repository by selecting
               one of the 18 Oregon
               basins then click the 'Search' button to display a list
of wells in the basin. 
               Optionally, include an agency to limit the well data
from the agency specified.  </p>

       <hr />
        </td>
    </tr>

         <tr>
            <td style="font-size: small">Search Criteria</td>
            <td colspan="3">(Partial words are acceptable.  Search
is not case sensitive)</td>
        </tr>

        <tr>
            <td style="text-align: right">
                Basin:</td>
            <td>
<asp:DropDownList ID="ddl_basin" runat="server">
    <asp:ListItem></asp:ListItem>
    <asp:ListItem Value="1">1 - North Coast</asp:ListItem>
    <asp:ListItem Value="2">2 - Willamette</asp:ListItem>
    <asp:ListItem Value="3">3 - Sandy</asp:ListItem>
    <asp:ListItem Value="4">4 - Hood</asp:ListItem>
    <asp:ListItem Value="5">5 - Deschutes</asp:ListItem>
    <asp:ListItem Value="6">6 - John Day</asp:ListItem>
    <asp:ListItem Value="7">7 - Umatilla</asp:ListItem>
    <asp:ListItem Value="8">8 - Grande Ronde</asp:ListItem>
    <asp:ListItem Value="9">9 - Powder</asp:ListItem>
    <asp:ListItem Value="10">10 - Malheur</asp:ListItem>
    <asp:ListItem Value="11">11 - Owyhee</asp:ListItem>
    <asp:ListItem Value="12">12 - Malheur Lake</asp:ListItem>
    <asp:ListItem Value="13">13 - Goose & Summer Lake</asp:ListItem>
    <asp:ListItem Value="14">14 - Klamath</asp:ListItem>
    <asp:ListItem Value="15">15 - Rogue</asp:ListItem>
```

```
    <asp:ListItem Value="16">16 - Umpqua</asp:ListItem>
    <asp:ListItem Value="17">17 - South Coast</asp:ListItem>
    <asp:ListItem Value="18">18 - Middle Coast</asp:ListItem>
</asp:DropDownList>
            </td>
            <td> </td>
            <td rowspan="3">
    <asp:Button ID="btn_search" runat="server" Text="Search"
                    ValidationGroup="query" /> <br />
    <asp:Button ID="btn_clear"  runat="server" Text="Clear"
onclick="btn_clear_Click"  />
            </td>
        </tr>
        <tr>
            <td style="text-align: right">
                Agency:</td>
            <td>

           <asp:TextBox ID="txt_agency" runat="server" Width="200px"
></asp:TextBox>

            </td>
            <td> </td>

        </tr>
        <tr>
            <td colspan="3" style="font-size: small">
                <br />
                List of Wells (Click on each header to sort)</td>
        </tr>
        </table>

    </div>

    <asp:GridView ID="GridView1" runat="server" CellPadding="4"
AllowPaging="True"
        EmptyDataText="No Records Found" ForeColor="#333333"
GridLines="None"
        DataSourceID="ods_owmwds_ws" AutoGenerateColumns="False"
        AllowSorting="True" >
        <RowStyle BackColor="#F7F6F3" ForeColor="#333333" />
        <FooterStyle BackColor="#5D7B9D" Font-Bold="True"
ForeColor="White" />
        <PagerStyle BackColor="#284775" ForeColor="White"
HorizontalAlign="Center" />
        <SelectedRowStyle BackColor="#E2DED6" Font-Bold="True"
ForeColor="#333333" />
        <HeaderStyle BackColor="#5D7B9D" Font-Bold="True"
ForeColor="White" />
        <EditRowStyle BackColor="#999999" />
        <AlternatingRowStyle BackColor="White" ForeColor="#284775" />
        <Columns>
            <asp:BoundField DataField="wwID" HeaderText="wwID"
SortExpression="wwID" />
            <asp:BoundField DataField="wwLongitude"
HeaderText="wwLongitude"
                SortExpression="wwLongitude" ItemStyle-Wrap="false" >
```

```
<ItemStyle Wrap="False"></ItemStyle>
            </asp:BoundField>
            <asp:BoundField DataField="wwLatitude"
HeaderText="wwLatitude"
                SortExpression="wwLatitude"  ItemStyle-Wrap="false" >
<ItemStyle Wrap="False"></ItemStyle>
            </asp:BoundField>
            <asp:BoundField DataField="wwHorizontalAcc"
HeaderText="wwHorizontalAcc (in feet)" SortExpression="wwHorizontalAcc"
/>
            <asp:BoundField DataField="wwLabel" HeaderText="wwLabel"
SortExpression="wwLabel" />
            <asp:BoundField DataField="wwLogCounty"
HeaderText="wwLogCounty" SortExpression="wwLogCounty" />
            <asp:BoundField DataField="wwLogNbr" HeaderText="wwLogNbr"
SortExpression="wwLogNbr" />
            <asp:BoundField DataField="wwStartCardNbr"
HeaderText="wwStartCardNbr" SortExpression="wwStartCardNbr" />
            <asp:BoundField DataField="wwOtherID"
HeaderText="wwOtherID" SortExpression="wwOtherID" />
            <asp:BoundField DataField="wwWellName"
HeaderText="wwWellName" SortExpression="wwWellName" />
            <asp:BoundField DataField="wwOwner" HeaderText="wwOwner"
SortExpression="wwOwner" />
            <asp:BoundField DataField="wwAddress"
HeaderText="wwAddress" SortExpression="wwAddress" />
            <asp:BoundField DataField="wwTaxLot" HeaderText="wwTaxLot"
SortExpression="wwTaxLot" />
            <asp:BoundField DataField="wwTypeOfUse"
HeaderText="wwTypeOfUse" SortExpression="wwTypeOfUse" />
            <asp:BoundField DataField="wwCompletionDate"
HeaderText="wwCompletionDate" SortExpression="wwCompletionDate"
DataFormatString="{0:d}" />
            <asp:BoundField DataField="wwCompletedDepth"
HeaderText="wwCompletedDepth (in feet)"
SortExpression="wwCompletedDepth" />
            <asp:BoundField DataField="wwCasingDiam"
HeaderText="wwCasingDiam (max. nearest inches)"
SortExpression="wwCasingDiam" />
            <asp:BoundField DataField="wwDepth2Water"
HeaderText="wwDepth2Water (nearest foot)"
SortExpression="wwDepth2Water" />
            <asp:BoundField DataField="wwStaticWater"
HeaderText="wwStaticWater (in feet)" SortExpression="wwStaticWater" />
            <asp:BoundField DataField="wwDrillingMethod"
HeaderText="wwDrillingMethod" SortExpression="wwDrillingMethod" />
            <asp:BoundField DataField="wwLandSurfaceElevation"
HeaderText="wwLandSurfaceElevation (in feet)"
SortExpression="wwLandSurfaceElevation" />
            <asp:BoundField DataField="wwElevationRemarks"
HeaderText="wwElevationRemarks" SortExpression="wwElevationRemarks" />
            <asp:BoundField DataField="wwUser" HeaderText="wwUser"
SortExpression="wwUser" />
            <asp:BoundField DataField="wwAgency" HeaderText="wwAgency"
SortExpression="wwAgency" />
```

```
            <asp:BoundField DataField="wwRecordDate"
HeaderText="wwRecordDate" SortExpression="wwRecordDate"
DataFormatString="{0:d}" />
        </Columns>

    </asp:GridView>

    <br />

    <asp:ObjectDataSource ID="ods_owmwds_ws" runat="server"
EnablePaging="True"
        OldValuesParameterFormatString="original_{0}"
        SelectCountMethod="Select_wells_by_basin_count"
        SelectMethod="Select_wells_by_basin"
SortParameterName="SortExpression"
        TypeName="owmwds_well_query_example_library.owmwds_DB">
        <SelectParameters>
            <asp:ControlParameter ControlID="ddl_basin"
Name="basin_nbr"
                PropertyName="SelectedValue" Type="Int32" />
            <asp:ControlParameter ControlID="txt_agency"
Name="wwAgency" PropertyName="Text"
                Type="String" />
        </SelectParameters>
    </asp:ObjectDataSource>

    </form>
</body>
</html>
```