XtraShare 2010
for SharePoint Technologies

Development Guide

How to develop custom solutions with XtraShare for SharePoint

RioLinx

support@riolinx.com

1/30/2012

TABLE OF CONTENTS

INTRODUCTION

OBJECTIVES

Beyond its out-of-the-box components, XtraShare for SharePoint is an open platform on which custom solutions can easily be built.

Specifically, XtraShare provides the following development interfaces:

- A full .NET Object Model (or "API")
- A Web Service interface
- An Event Management Framework, which allows developers to capture specific events such as CRUD operations on users or groups, but also more specific events such as when a user self-registers.

This document provides SharePoint developers with detailed processes and steps to develop custom solutions on top of XtraShare for SharePoint.

CONVENTIONS

The following table lists the acronyms used in this document.

Acronym	Stands for
SPF	SharePoint Foundation (v4)
MSS	Microsoft SharePoint Server 2010
CA	Central Administration (refers to a SharePoint server that hosts a Central Administration site)
WFE	Web Front-End (designates a SharePoint server with the Web Application Role)
VS	Visual Studio (2010)

THE XTRASHARE WEB SERVICE

XtraShare comes with a web service that can be used to automatically manage XtraShare objects (such as groups or users) from external systems. For instance, this can be useful if users must be created from or synchronized with a CRM system such as Microsoft Dynamics.

The XtraShare Web Service is physically deployed to the C:\Program Files\Common Files\Microsoft Shared\Web Server Extensions\14\ISAPI\XtraShare but it isn't readily available and can be called at the following url:

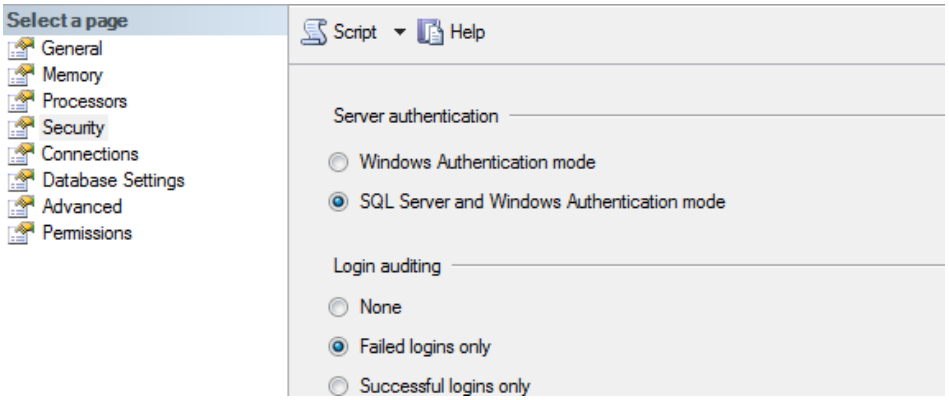
[SharePointSiteUrl]/_vti_bin/XtraShare.om.asmx

where [SharePointSiteUrl] is the url of a SharePoint site.

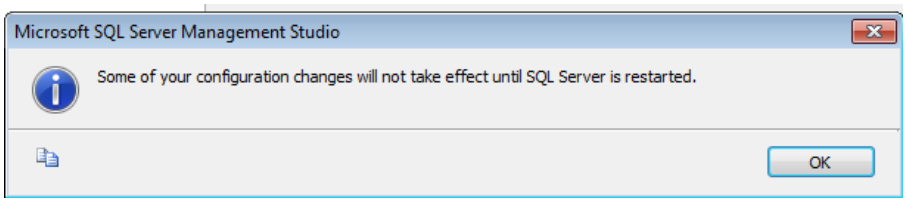
HOW TO CONFIGURE THE XTRASHARE WEB SERVICE?

In order for the XtraShare web service to properly work, you must ensure that the following pre-requisites are met:

1. Enable SQL Server and Windows Authentication mode on the SQL Server instance that hosts the XtraShare database.
    a. This can be achieved in SQL Server Management Studio by right-clicking on the SQL server instance and selecting the Properties menu.
    b. In the Server Properties window, select the Security tab and make sure that "SQL Server and Windows Authentication mode" is selected.



    c. If you have to modify the Server authentication, please note that you may have to restart the SQL Server instance



   d. Create a SQL Server user (such as "xsadmin") to assign this user as db_owner on the XtraShare database used by your SharePoint environment.

2. Configure the Web.config file in C:\Program Files\Common Files\Microsoft Shared\Web Server Extensions\14\ISAPI\XtraShare.
   a. **Important note**: for security reasons, you should only configure this web.config file on a server in your SharePoint farm that is NOT accessible to external users.
   b. Open the Web.config file in the C:\Program Files\Common Files\Microsoft Shared\Web Server Extensions\14\ISAPI\XtraShare folder.
   c. Identify the following section and adjust the `connectionString` property to match the location and user information of your XtraShare database

```
<RLSoft.XtraShare
connectionString="server=.\sql2008;database=XtraShareTest;User
Id=xsadmin;Password=pass" contentDisposition="inline" />
```

HOW TO TEST THE XTRASHARE WEB SERVICE?

In order to verify that the web service is properly configured, you may want to test it from a browser on the local server.

In order to do so, you should modify the following entry of the Web.config file (in C:\Program Files\Common Files\Microsoft Shared\Web Server Extensions\14\ISAPI\XtraShare) from:

```
<remove name="HttpPostLocalhost" />
```

to

```
<add name="HttpPostLocalhost" />
```

Then navigate to [WebAppUrl]/_vti_bin/XtraShare/om.asmx (where [WebAppUrl] is the url of your SharePoint Web Application) and click on the RoleCollectionLoadAll link:

- RoleCollectionLoadAll

Next, click on the Invoke button and a page similar to the following screenshot should appear:

```
<?xml version="1.0" encoding="utf-8" ?>
- <ArrayOfRole xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance'
  - <Role>
      <EntityKey>1</EntityKey>
      <RowVersion>AAAAAAAAB9M=</RowVersion>
      <Id>1</Id>
      <Name>Administrators</Name>
      <Description>XtraShare Administrator group</Description>
      <ScopeId>1</ScopeId>
      <EntityState>Unchanged</EntityState>
    </Role>
  - <Role>
      <EntityKey>2</EntityKey>
      <RowVersion>AAAAAAAAB9Q=</RowVersion>
      <Id>2</Id>
      <Name>Default Group</Name>
      <Description>Default group for newly created users</Description>
      <ScopeId>1</ScopeId>
      <EntityState>Unchanged</EntityState>
    </Role>
  - <Role>
      <EntityKey>3</EntityKey>
      <RowVersion>AAAAAAAAB9g=</RowVersion>
      <Id>3</Id>
      <Name>Registered Users</Name>
      <Description />
      <ScopeId>1</ScopeId>
      <EntityState>Unchanged</EntityState>
    </Role>
</ArrayOfRole>
```

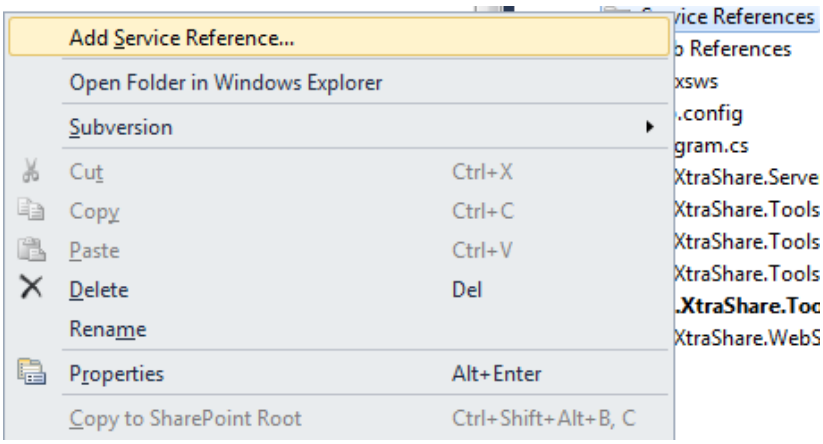

If you don't see a similar page but instead an error message such as "The XML page cannot be displayed", you most likely haven't configured your connection string properly, or the SQL user configured in that connection string doesn't have the appropriate access permissions in the XtraShare database. Please refer to the previous section for detailed configuration steps.

HOW TO CALL THE XTRASHARE WEB SERVICE?

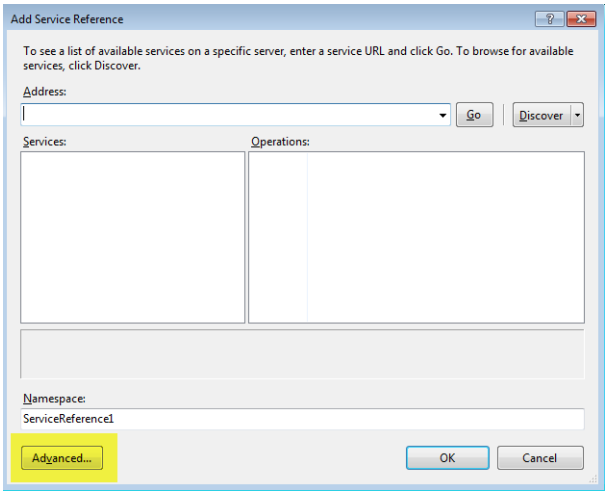
Once the XtraShare web service has been properly configured and tested, you can start calling it from a web service.

In the example below, we will show how to call the XtraShare Web Service from a remote C# Console Application. The principles that we highlight below should be the same for other technologies or types of applications, but we welcome your feedback if you run into any issue.

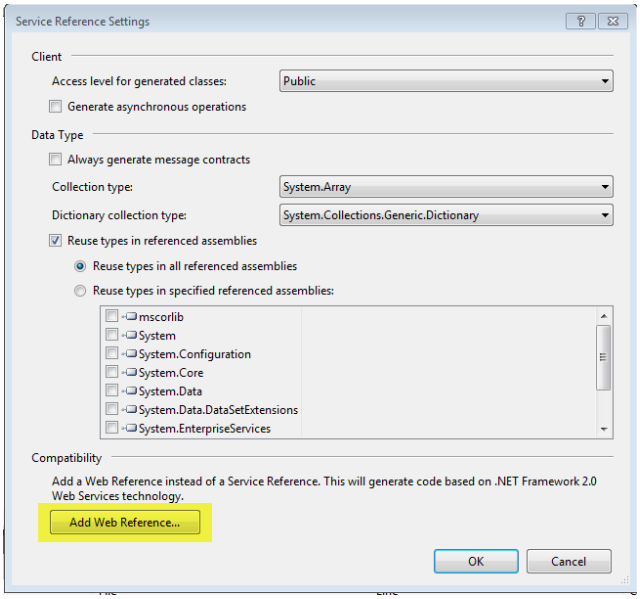
1. Open Visual Studio 2010 and create a new C# Console Application project.
2. Click on ***Service References*** and select ***Add Service Reference***.



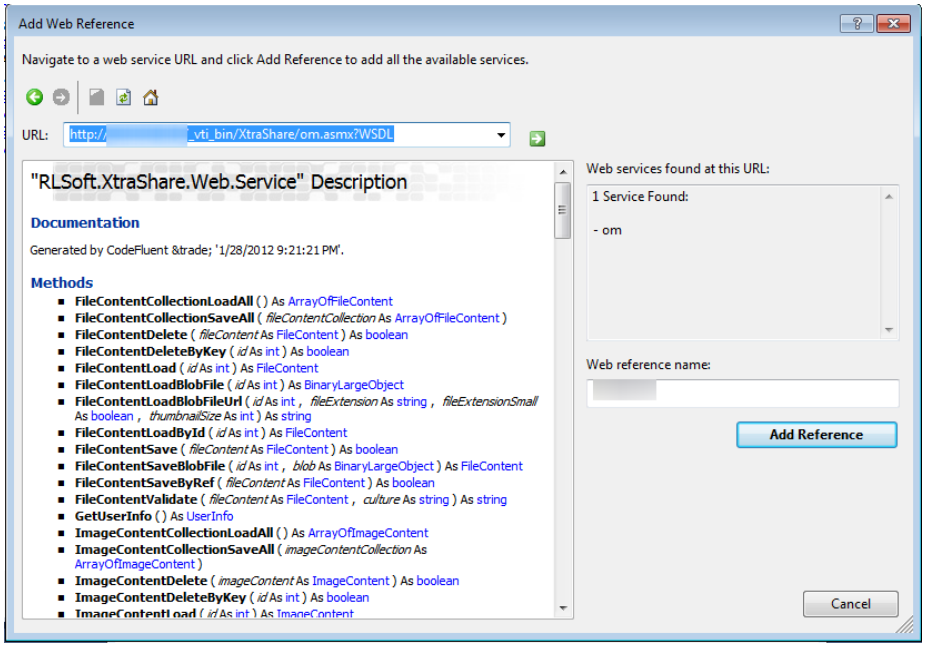
3. In the Add Service Reference window, click on the Advanced button at the bottom of the window:



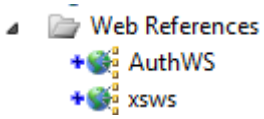
4. In the Service Reference Settings popup window, click on the ***Add Web Reference*** button:



5. In the Add Web Reference window, enter the url of your web application following by **/_vti_bin/XtraShare/om.asmx?WSDL.** For instance, the url you would enter could be http://extranet/_vti_bin/XtraShare/om.asmx?WSDL if http://extranet is the url of your web application.
   Note: Don't forget to appeadl the "?WSDL" suffix, otherwise Visual Studio will be stuck in a loop trying to find the web service description.
   After a few seconds, you should see a windows similar to the following one:



6. Name your web reference (in our sample application, we named it “xsws”) and click on the ***Add Reference*** button.
7. At this stage, you should download our Sample Console Application and open it in Visual Studio 2010.
8. Once it is open, take a look at the Web References. You will notice that there are 2 of them, AuthWS and xsws:



The xsws web reference is the web reference to the XtraShare Web Service, while the AuthWS is the reference to the SharePoint Authentication Web Service. This web service is indeed required when calling a web service accessed through Forms-based Authentication. However, the sample C# Console Application provides 2 methods to connect to the XtraShare Web Service:

a. Through standard Windows Authentication
b. Through Forms-Based Authentication

Depending on the url you use to call the XtraShare Web Service and the available Authentication Providers (by default, XtraShare keeps the Windows Authentication provider but you may disable it manually in Central Administration), you might choose one method or the other.

Take a look at the appSettings section of sample’s app.config file:

```
  <appSettings>
    <!-- Set UseFBA to false to call the XtraShare Web Service through Windows
Authentication.
    If UseFBA is set to false, you must adjust the Domain, UserName and Password keys
below with a Windows account (with at least read permissions on the site specified
below in SiteUrl)
    If UseFBA is set to true, you must  adjust the Username and Password keys below
with an XtraShare account (with at least read permissions on the site specified below
in SiteUrl)-->
    <add key="UseFBA" value="false"/>
    <!-- Active Directory Domain or SharePoint Server Local Name. Only required if
UseFBA is set to false -->
    <add key="Domain" value="DOMAIN"/>
    <!-- Windows or XtraShare User-->
    <add key="UserName" value="user"/>
    <!--Windows or XtraShare Password -->
    <add key="Password" value="password"/>
    <!-- Remote Url of SharePoint site (this could be the Url of the Central
Administration too)-->
    <add key="SiteUrl" value="http://localhost:8000"/>
    <!-- Number of test users to create -->
    <add key="UsersCount" value="1"/>
  </appSettings>
```

9. Adjust the values as mentioned in the comments for each key and check the code of the console application to see how the XtraShare web service is called, depending on the value of the UseFBA parameter.
10. You should now be able to compile and test this sample application which will create some test users in your XtraShare database.