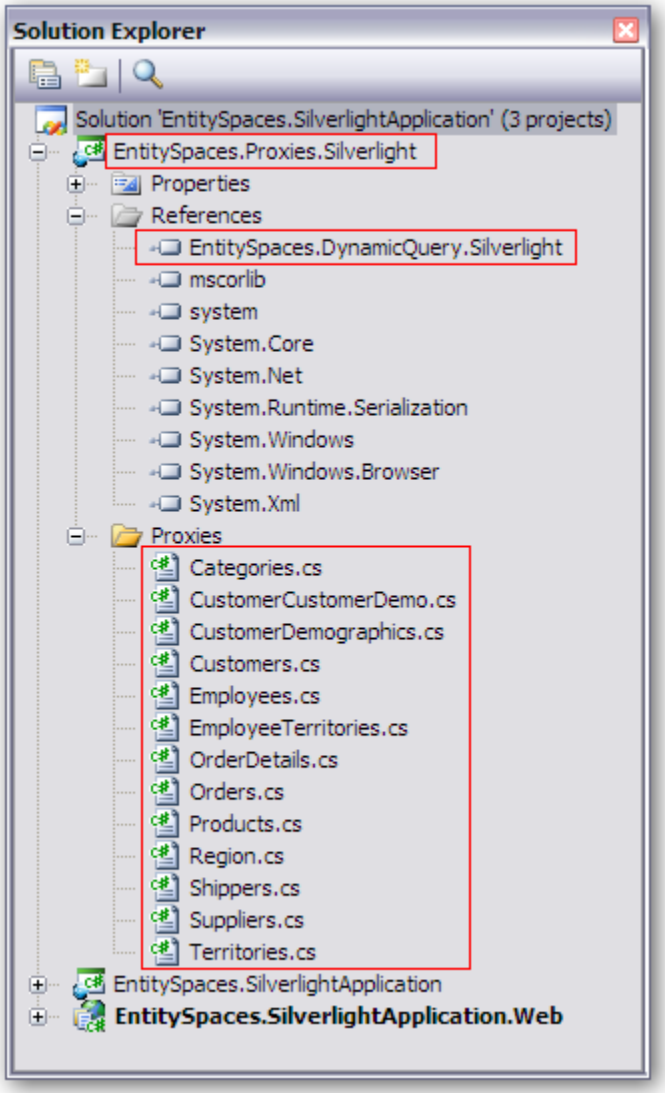
EntitySpaces 2009 Silverlight Demo

Table of Contents

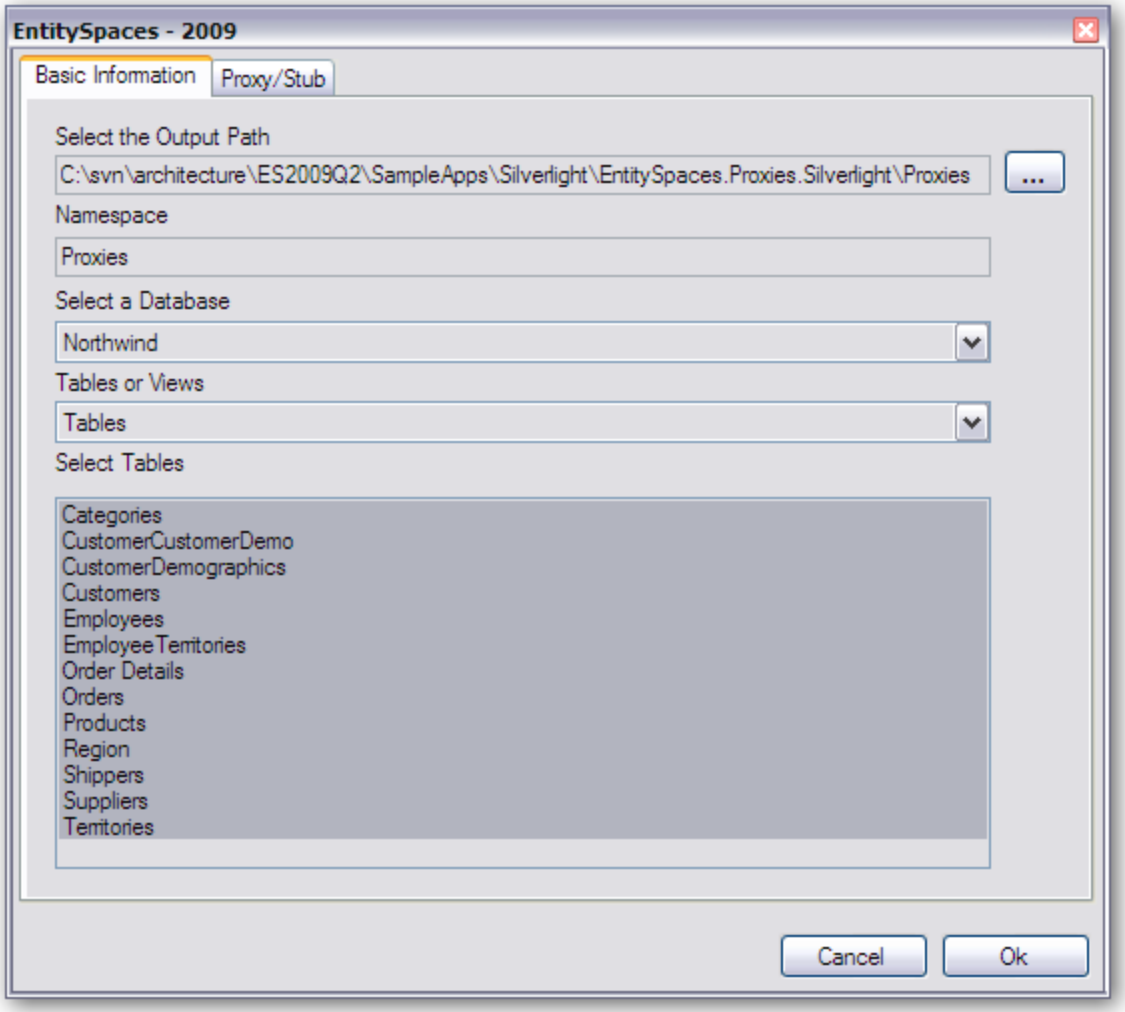
How We Generated the Lightweight Client Side Proxies



This is how we generated the "*EntitySpaces.Proxies.Silverlight"* Assembly. This must be created as a "Silverlight Class Library" when you choose "New Project".

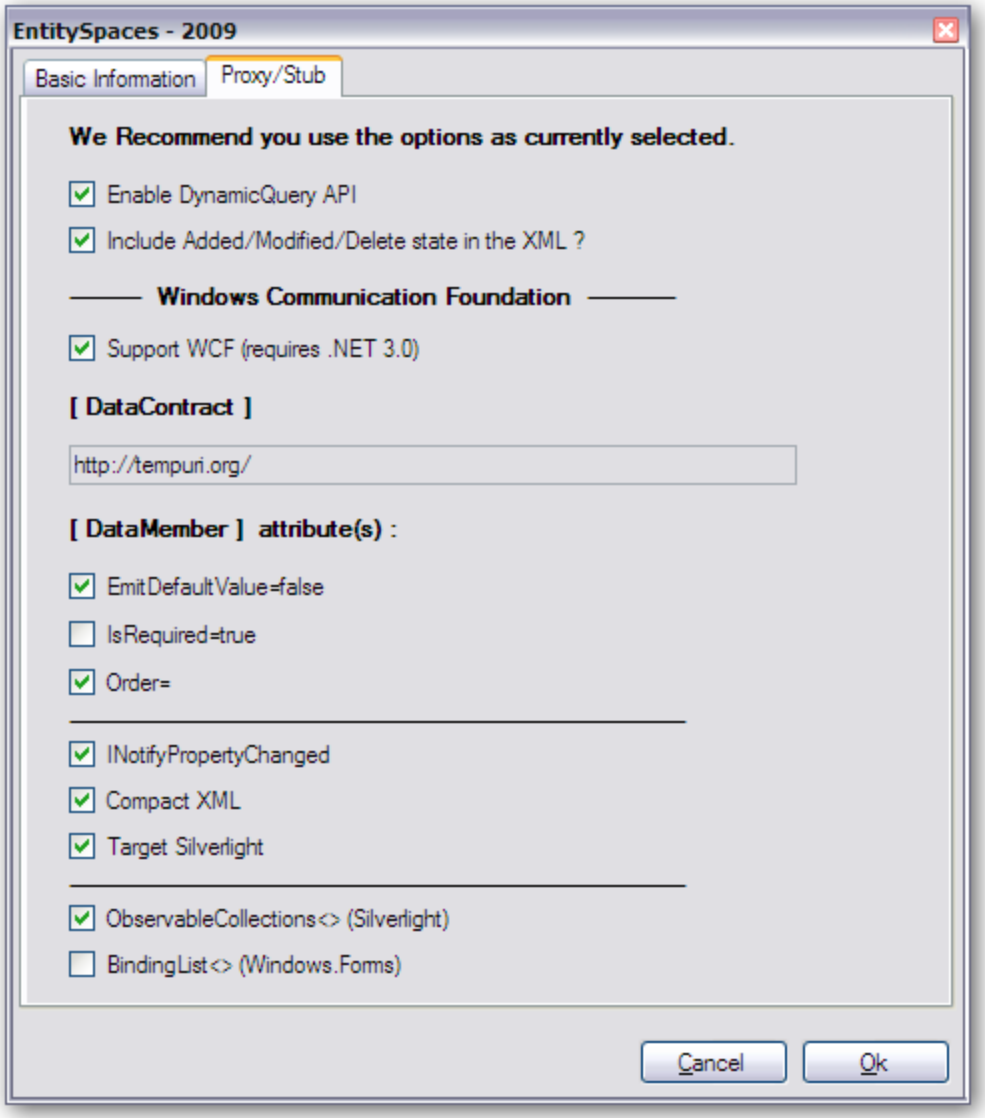
This "*EntitySpaces.Proxies.Silverlight"* proxy class library will run be automatically downloaded into the browser when the user hits your Silverlight application.

Notice that we include the EntitySpaces.DynamicQuery.Silverlight assembly. This assembly allows you to do full EntitySpaces DynamicQueries in your Silverlight application, actually from within the browser. You actually serialize the query and send it to the server to execute. Obviously you do not access your database directly from your Silverlight application.



There wasn't much to do here on the "Basic Information" Tab. We merely selected all of the tables in the Northwind database and we used Proxies as our namespace.

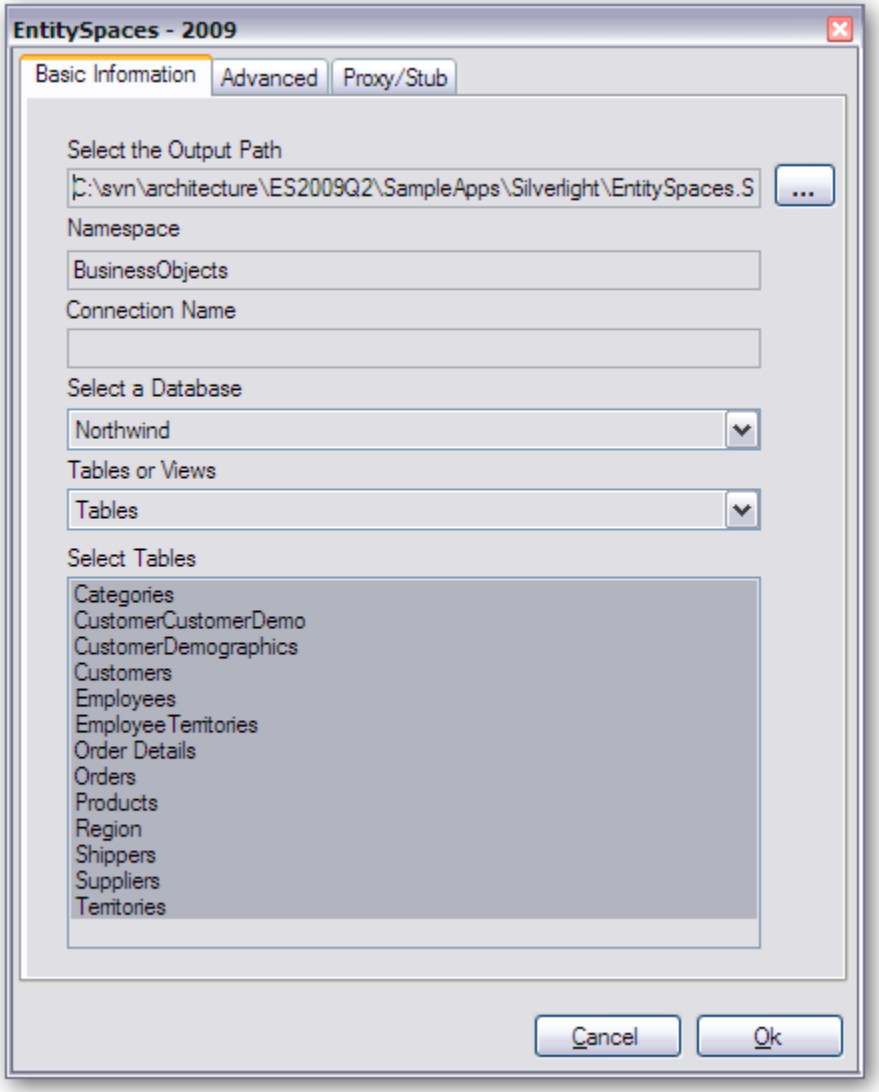
It is important that these be in their own class library. See the next page for the interesting UI choices …

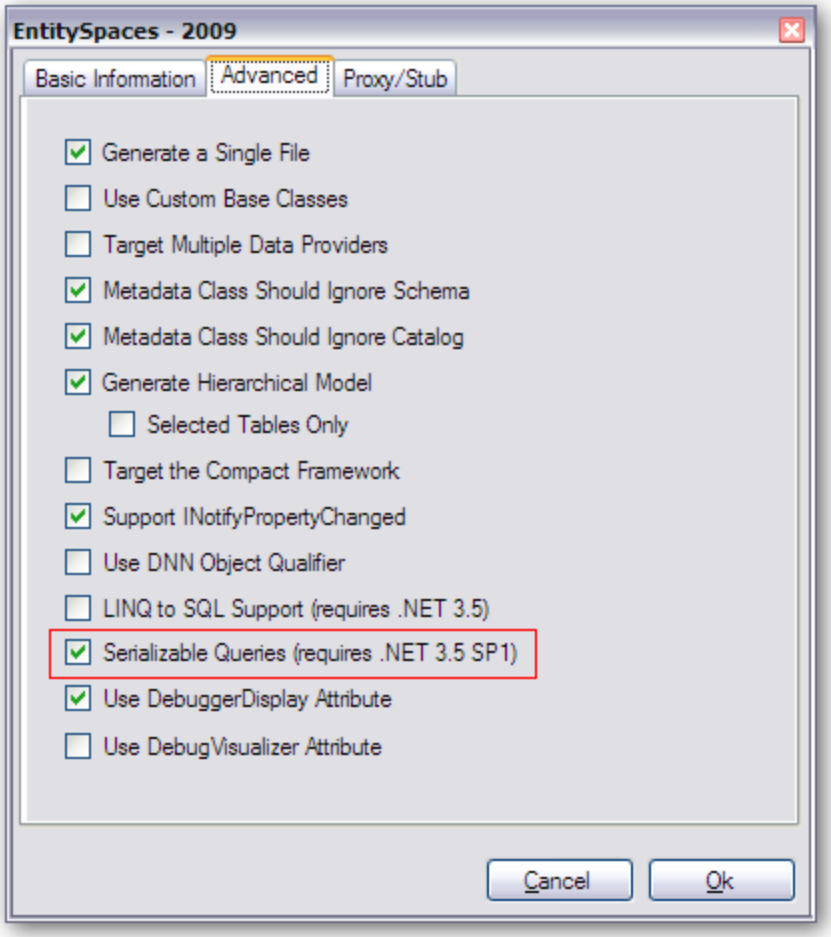


These are the choices we recommend for Silverlight development on the Proxy/Stub tab. That's it, merely generate these classes, include them in your Silverlight Class Library, add a reference to the EntitySpaces.DynamicQuery.Silverlight assembly which you will find in the C:\Program Files\EntitySpaces 2009\Runtimes\.NET 3.5\Silverlight folder and you're off and running. Note, in order to use serializable queries from under Silverlight SP1 of the 3.5 .NET Framework is required on the Server.

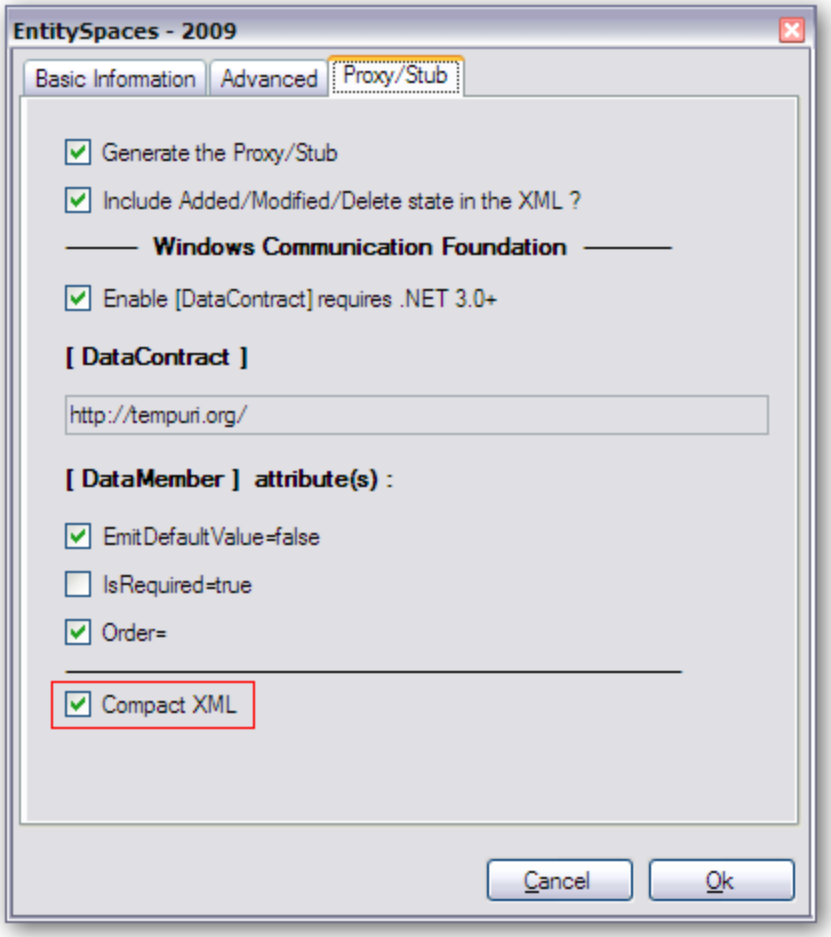
How We Generated the Main Business Objects

These objects are housed in our “EntitySpaces.SilverlightApplication.Web” which also houses our WCF Service.





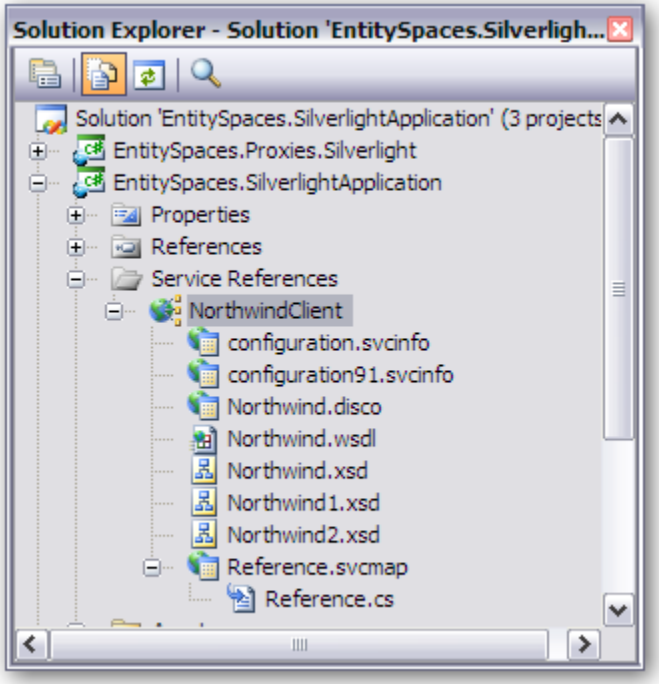
The key things to notice here is that we checked **Serializable Queries** which is necessary if we want to be able to deserialize the queries sent to us from our Silverlight Application.



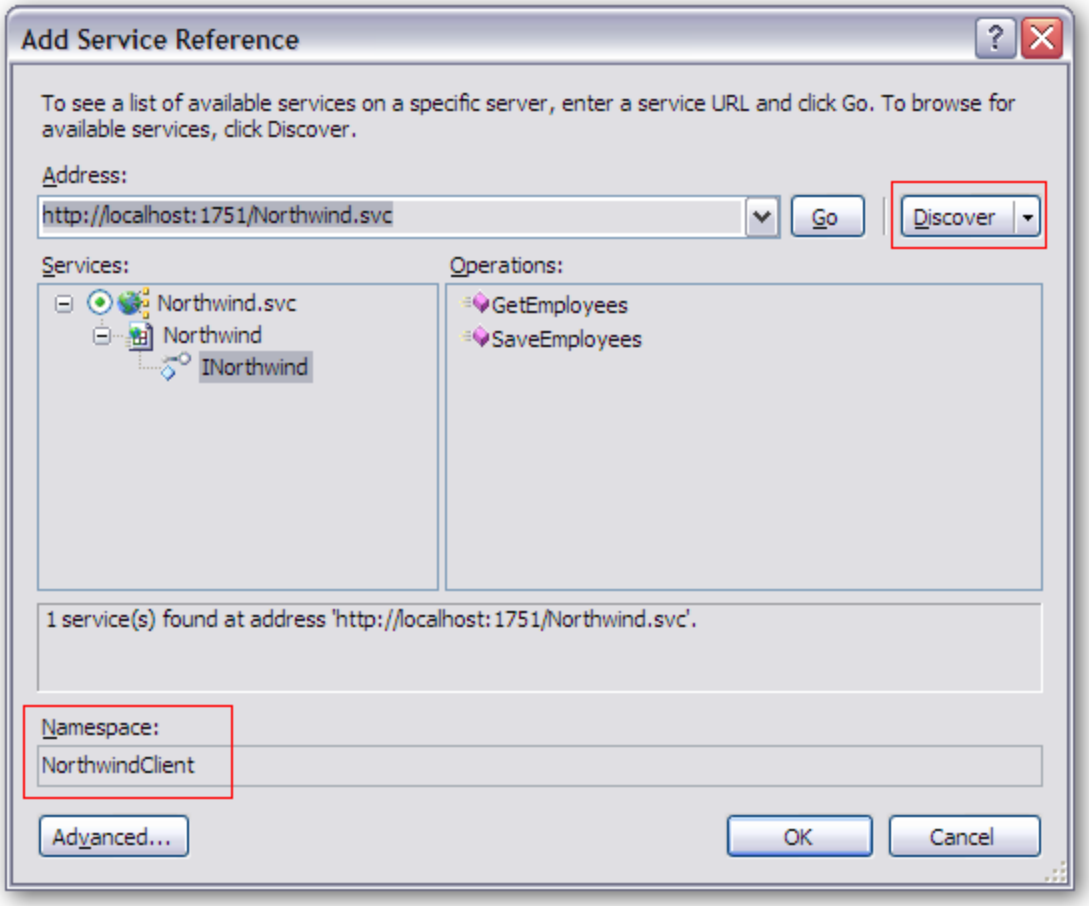
Notice here we make the same choices as we did when created the lightweight proxies. The important thing here is that we need to be sure to check the "Compact XML" checkbox too since we also made this choice for our lightweight proxies.

How We Added the Service Reference

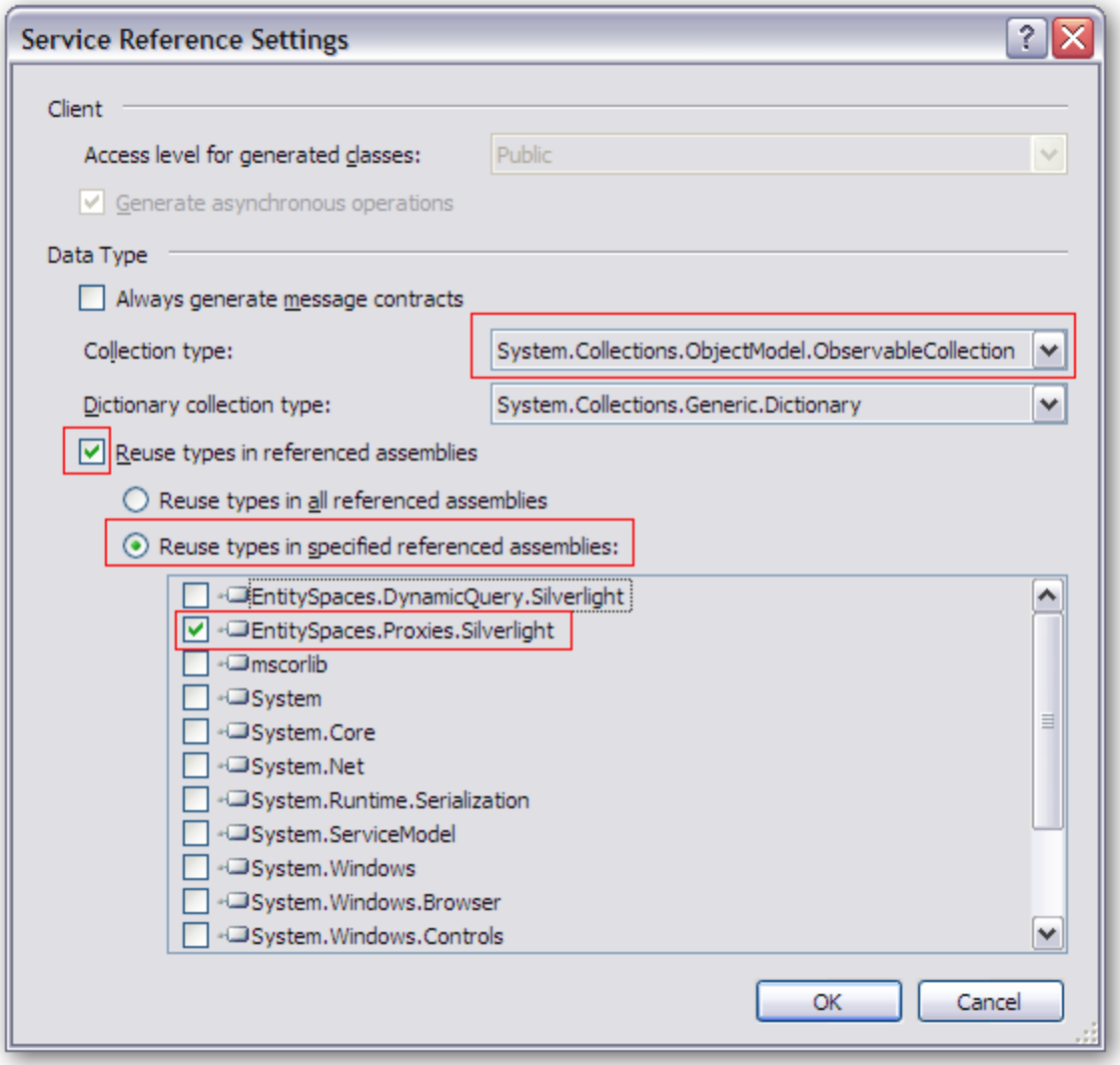
This is what really makes EntitySpaces shine when using it under Silverlight. You can map the full server side proxies to the lightweight client side proxies and totally skip the Visual Studio proxies. This make the programming model very nice.



Let's take a look on the following two pages and see how we added our Service Reference.



The above is pretty standard, since our Service is in the same solution we can merely use the “Discover” button and it will automatically find our Northwind service. Then we changed the Namespace to “NorthwindClient”. The next thing we need to do is let Visual Studio know we want it to use our Proxies and not the Visual studio generated proxies. To do this we must press the “Advanced” Button.



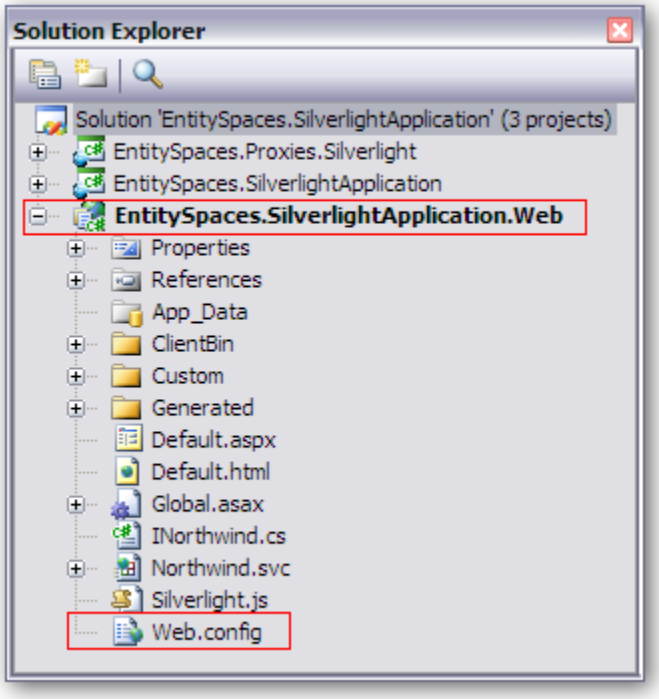
Notice that we set the collection type to “ObservableCollection” the same type that we chose when generating our lightweight proxy stubs. We also indicate that we want it to use the proxies in our EntitySpaces.Proxies.Silverlight assembly (that’s just what we happened to name our assembly that contains the lightweight proxies).

That’s it, click “OK”

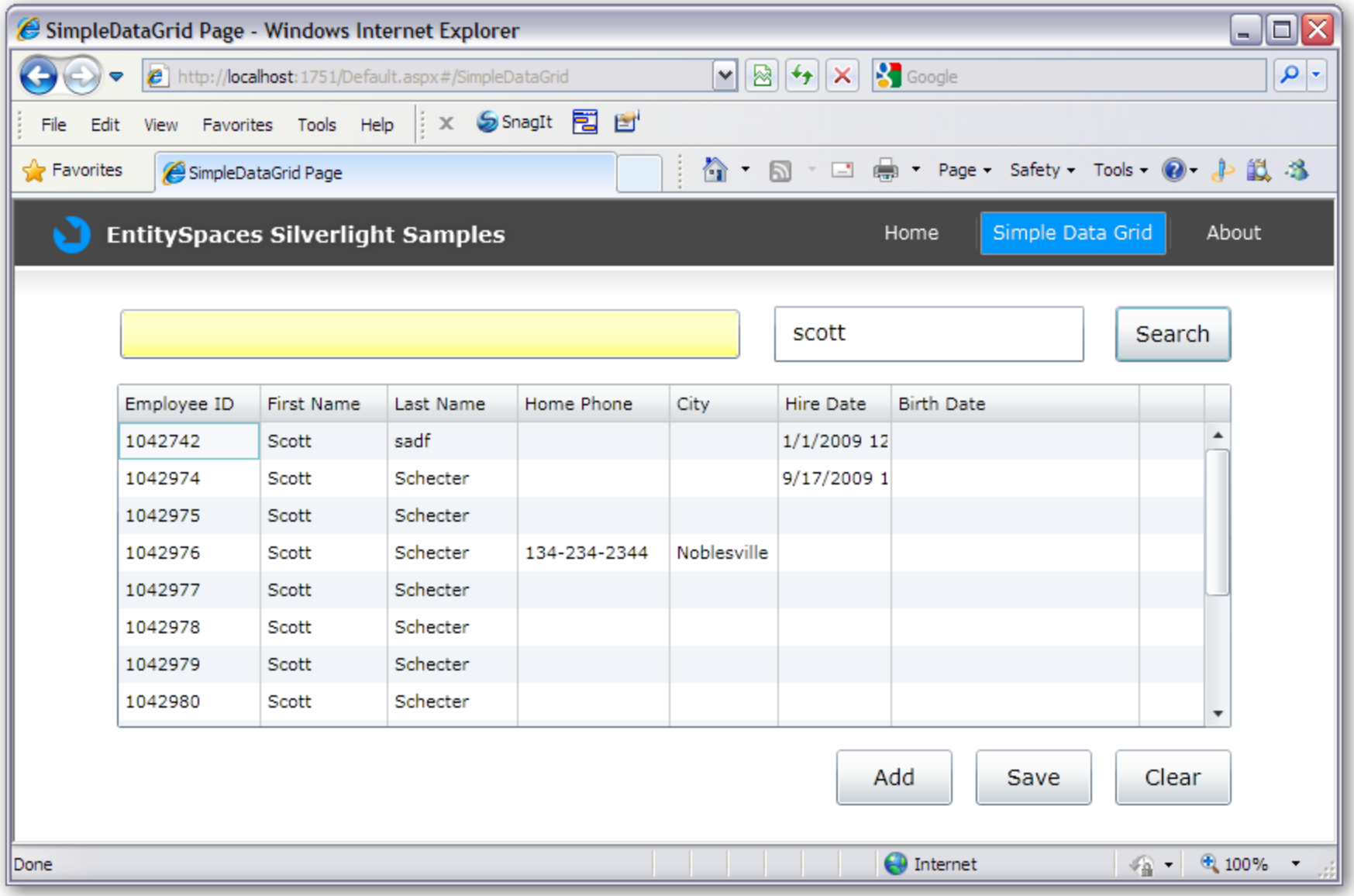
Running the Silverlight Demo

The first thing you need to do is make sure the connection string in the web.config file is correct for your Northwind database. Find the **connectionStrings** section in your web.config file and set it up appropriately.

```
<connectionStrings>
    <add name="Home" connectionString="Password=;User ID=sa;Initial Catalog=Northwind;Data Source=localhost"/>
</connectionStrings>
```



Set the "EntitySpaces.SilverlightApplication.Web" project as your start up project and start the application. Not that it might take a little while on your first search before you see any data. That's it, you should see a screen that looks like this:



The first thing you might try is pressing the “Add” button a few times and then press Save. Then try some searches, the search searches both the FirstName and LastName. You can also edit the values in the grid and press save. To see the Silverlight code itself see the SimpleDataGrid.xaml.cs file in the

EntitySpaces.SilverlightApplication project under “Views”. It’s quite simple since we use EntitySpaces code on both sides and not Visual Studio generated proxies.