Qlikview12 Source Control with Team Foundation Server (TFS) and Visual Studio 2012 (VS2012)

TFS Integration

I had never used Source Control software until a few months ago. Like many of us my source control was a well controlled system of folders on the server and my local drive, with only the ocassional mishap or overwriting and a little pleading with IT to restore a deleted file.
However in my new role I was instructed to get our new Qlik implementation working with TFS. My first stop was obviously the community but I found the information to be quite limited and out dated.

There is an excellent and detailed document on Source Control here [Updated Guide to QlikView and Source Control (TFS & Subversion)](#)
This little document is designed to supplement that as an 'idiots guide to'. I have also included the msscci setup process as this is required in more recent versions of Qlik. Finally I have used VS2012 to do all of the source control rather than using the Qlik screens.

Setup:

In more recent versions of Qlik you need to use the appropriate msscci provider install, this is the 2012 version (only 700kb):
https://visualstudiogallery.msdn.microsoft.com/b5b5053e-af34-4fa3-9098-aaa3f3f007cd

NOTE! It is easier to install this via cmd (run as administrator) altering the following command as necessary:
msiexec /i "C:\Users\adam.davies\desktop\Visual Studio Team Foundation Server 2012 Update 2 MSSCCI Provider (64-bit).msi" ALLUSERS=1
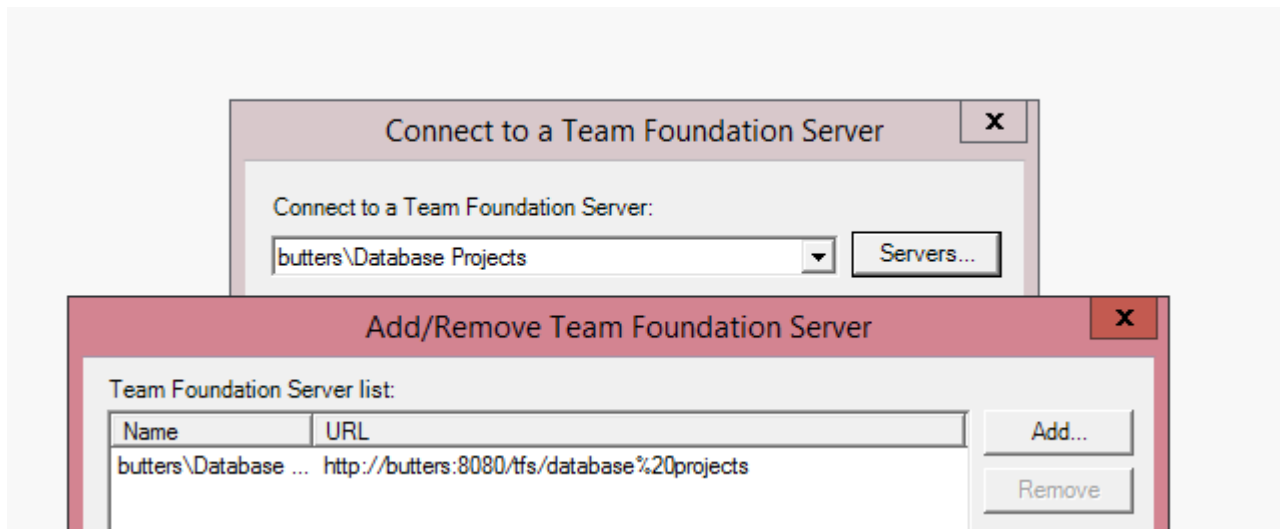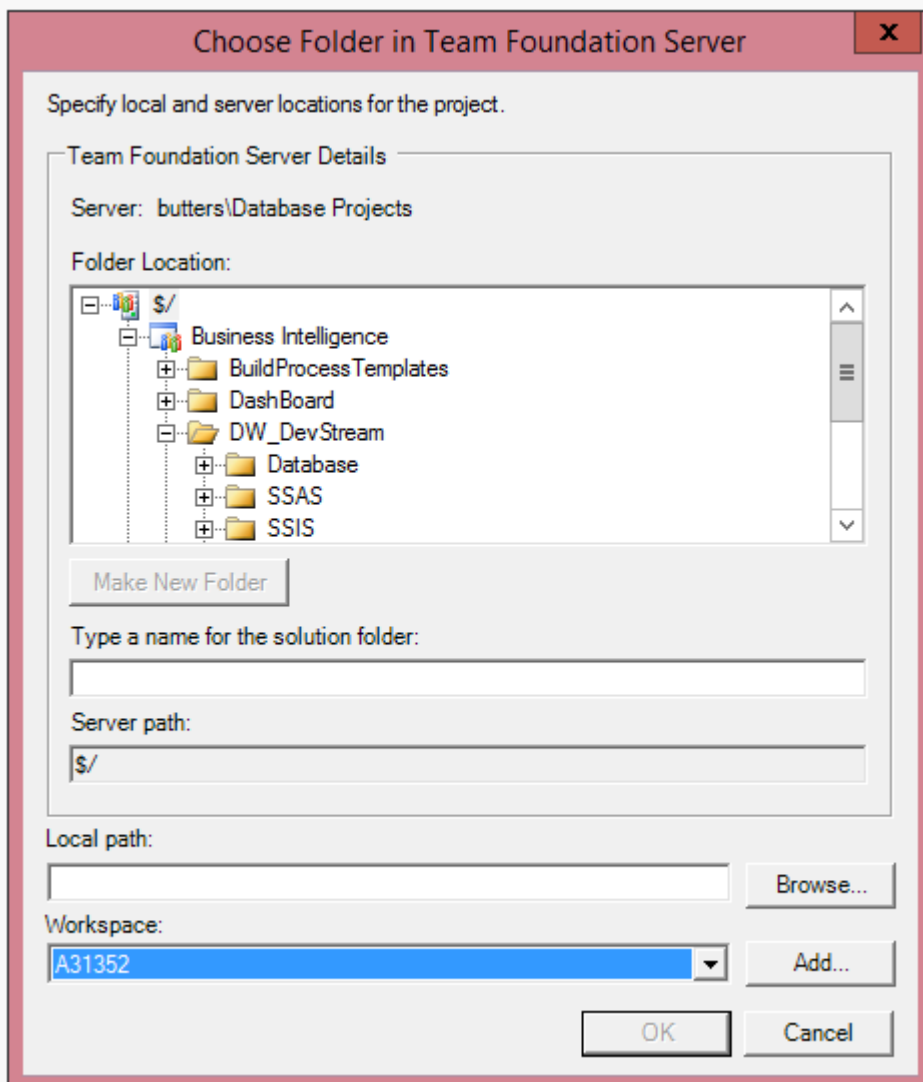In qlikview go to file > source control> settings
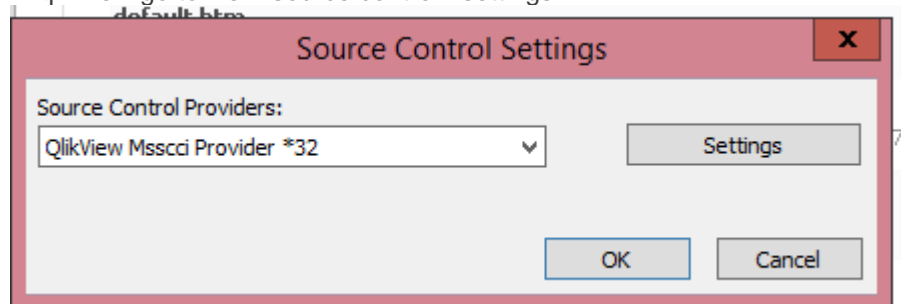
Choose the qlikview msscci provider and then click on settings and Add this (appropriately altered) to the filepath:
C:\Program Files (x86)\Microsoft Team Foundation Server MSSCCI Provider\TfsMsscciProvider.dll
Once this is in then you should be able to connect to butters. Go to source control > get projects
Hopefully you will get a request to enter the server details:

Once this is in then you should be able to browse the projects


Using Source:

In summary the steps are as such:

1. Create a prj folder

2. Open and Save the dashboard

3. Copy the prj folder to your local source folder

4. If your dashboards are small, copy those, otherwise just recreate one with the same name next to the prj folder and it will recreate itself from those details

5. Check these into source using VS2012

Example source control:

This is our test dashboard deployed on our server:

It contains one object and one small item in the script:

1. The first step is to create our prj folder, this is simply an empty folder with the same name as our dashboard and -prj tagged on the end.

2. The prj folder will not have anything in it, until we open and save our dashboard, at which point the files will get generated:

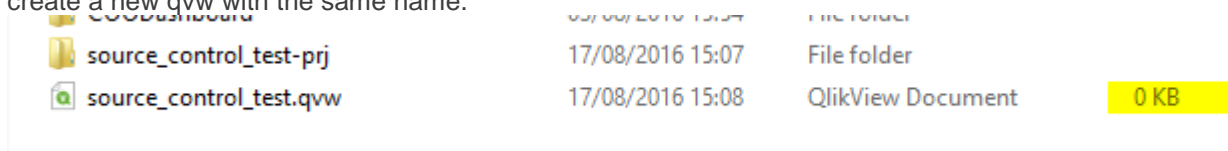|			
AllProperties.xml	17/08/2016 11:05	XML File	1 KB
DocBinary.dat	17/08/2016 11:05	DAT File	1 KB
DocInternals.xml	17/08/2016 11:05	XML File	1 KB
DocProperties.xml	17/08/2016 11:05	XML File	42 KB
LoadScript.txt	17/08/2016 11:05	Text Document	1 KB
Module.txt	17/08/2016 11:05	Text Document	1 KB
QlikViewProject.xml	17/08/2016 11:05	XML File	2 KB
SH01.xml	17/08/2016 11:05	XML File	3 KB
TopLayout.xml	17/08/2016 11:05	XML File	2 KB
TX01.xml	17/08/2016 11:05	XML File	8 KB

Now these are actually all the files which are required to make a dashboard. You could even edit items directly with these files if you wanted, but it's obviously easier to have a dashboard to hand to open in the application.
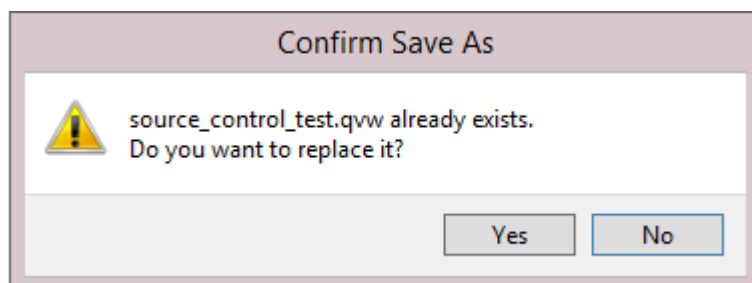
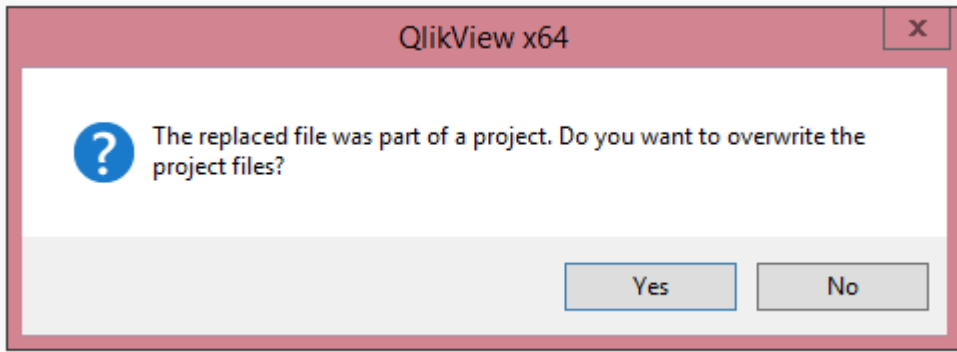3. Now we need to copy the folder to our local source folder:
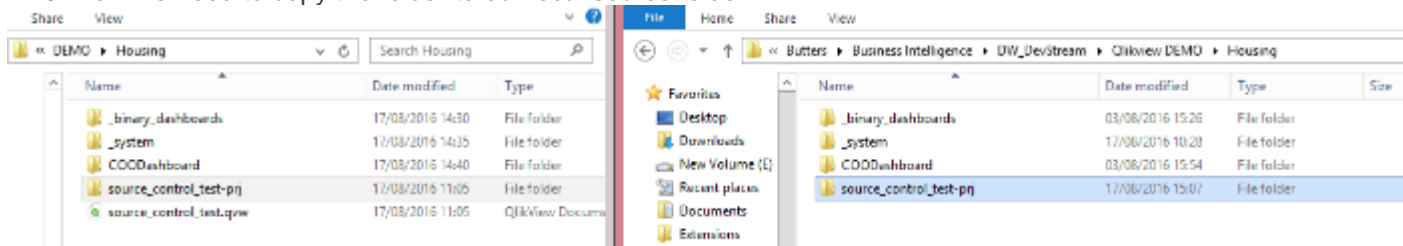
4. Obviously we also want a dashboard locally that we can work on. If the file is small then I usually copy this over as well, however if it is large and you don't want to copy it then you can just create a new qvw with the same name.

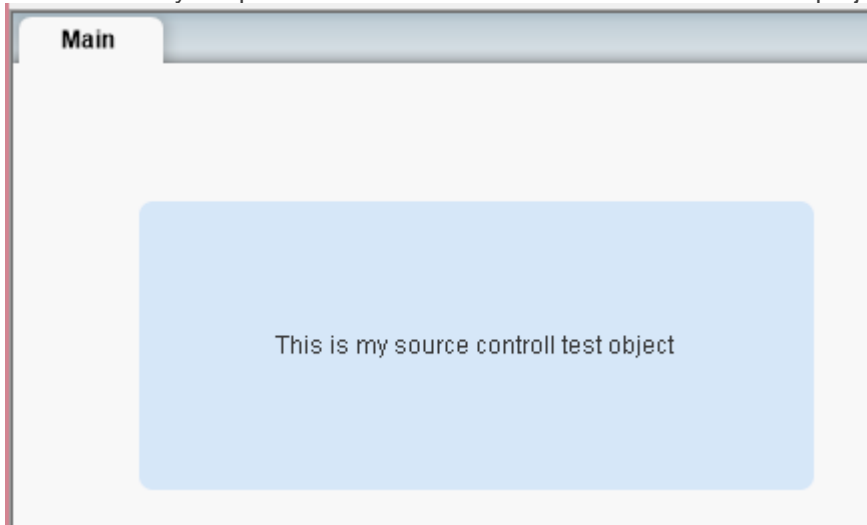|			
CODDashboard	03/08/2016 15:54	File folder	
source_control_test-prj	17/08/2016 15:07	File folder	
source_control_test.qvw	17/08/2016 15:08	QlikView Document	0 KB

Open and save the file, and choose yes to replace the file:
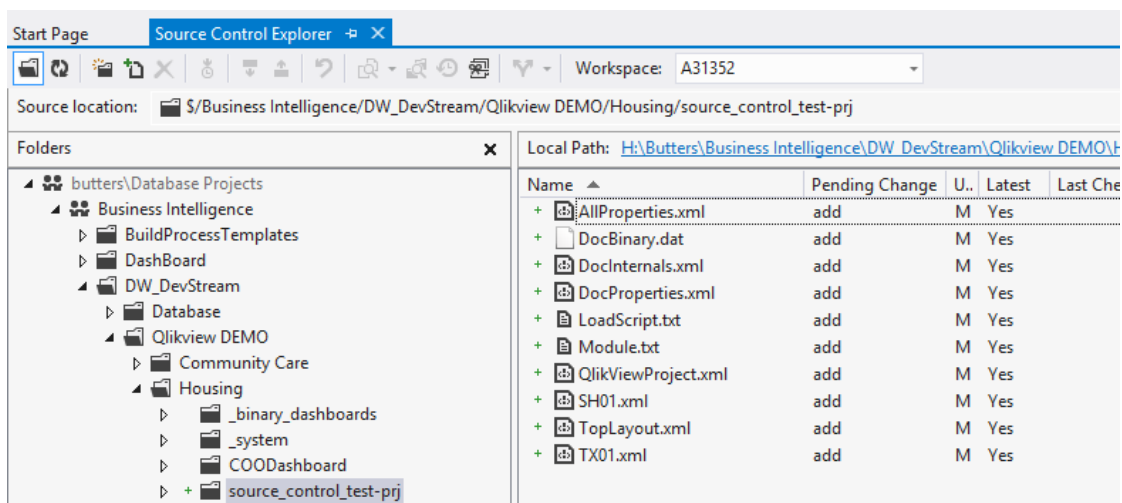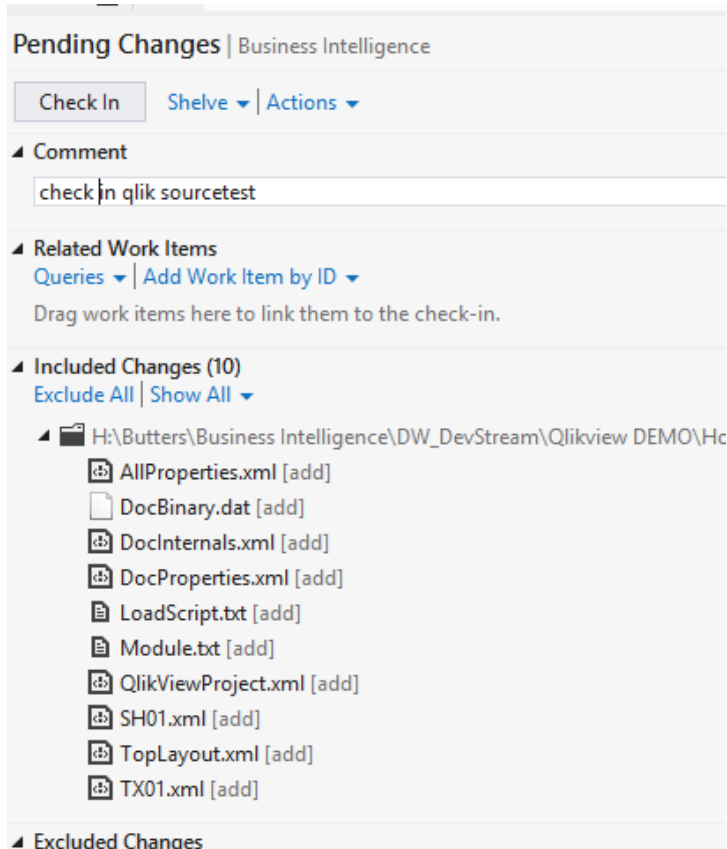

But NO to replace the project files:

The next time you open the file it will have been rebuilt based on the project files

5. Now I normally switch to visual studio to work with source as I find it easier however you can do it via the Qlik screens. I'll explain how to do it with VS2012
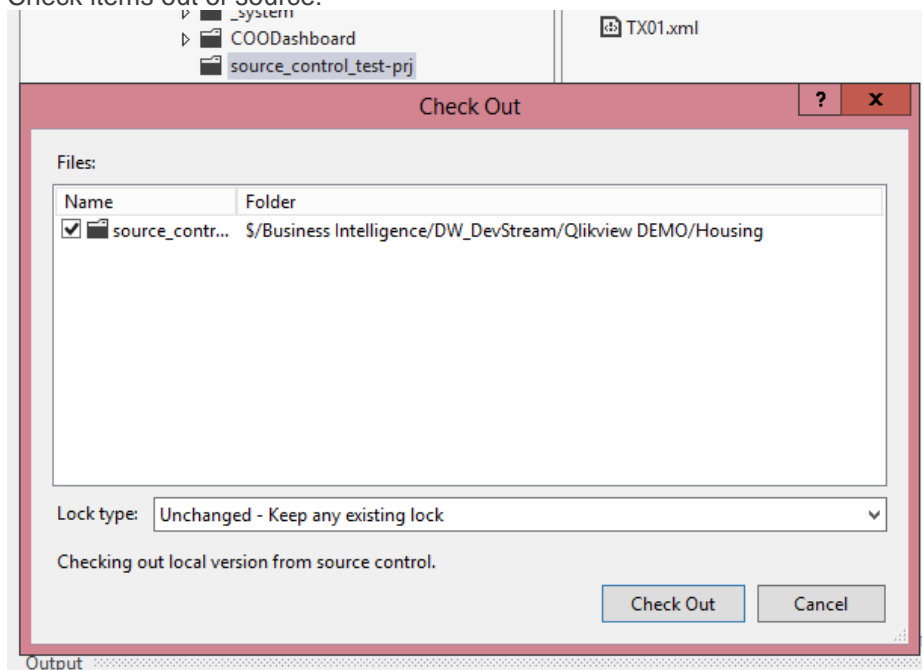
You should see that the files now exist in your source control explorer ready to check in:

To alter the file, you can now just check them out of source, make the changes then check them back in. Once this is done you can copy the prj folder back to your server. Below is the sequence of events:

Check items out of source:

Open your local dashboard and make some changes as required
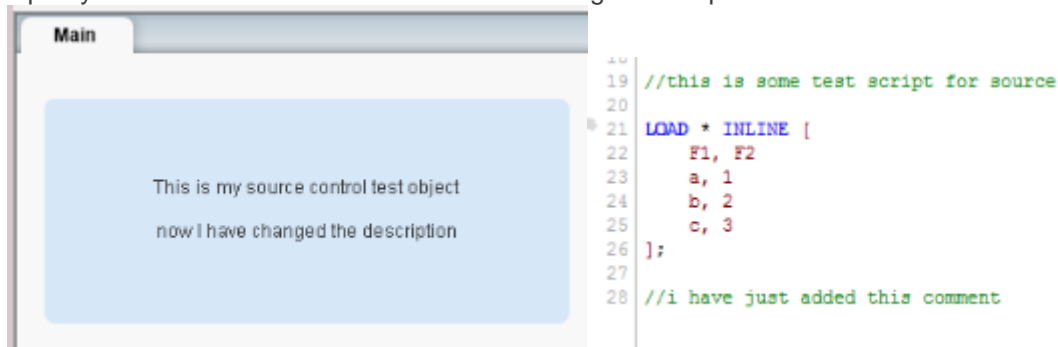
Then save the dashboard

Looking back at our source control in VS2012 we can see that some files have changed (note they are the two things we changed in Qlik) and we need to check them in:

Name ▲	Pending Change	U..	Latest	Last Check-in
AllProperties.xml			Yes	17/08/2016 15:...
DocBinary.dat			Yes	17/08/2016 15:...
DocInternals.xml			Yes	17/08/2016 15:...
DocProperties.xml			Yes	17/08/2016 15:...
✓ LoadScript.txt	edit	M	Yes	17/08/2016 15:...
Module.txt			Yes	17/08/2016 15:...
✓ QlikViewProject.xml	edit	M	Yes	17/08/2016 15:...
SH01.xml			Yes	17/08/2016 15:...
TopLayout.xml			Yes	17/08/2016 15:...
✓ TX01.xml	edit	M	Yes	17/08/2016 15:...

Check these changes in
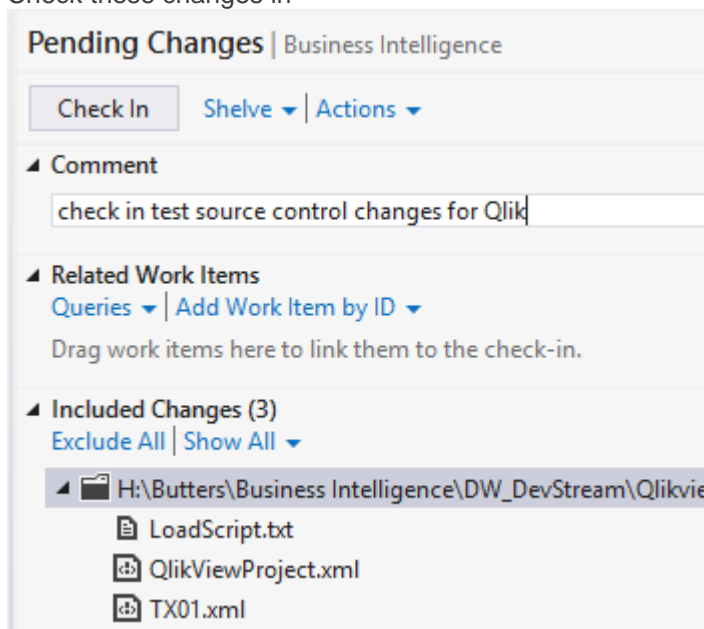
Now copy JUST the prj folder to the Qlikview Server replacing the files on the server.

Then open the server dashboard and you should see your changes:
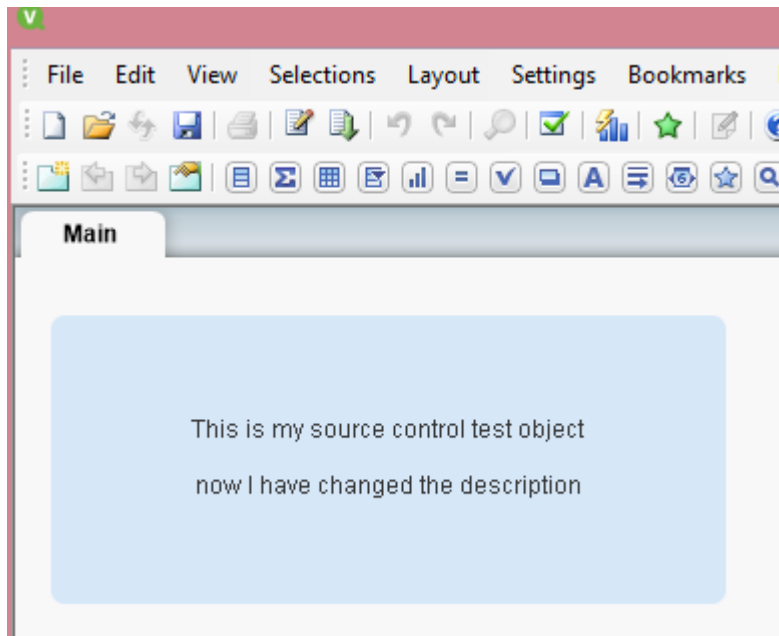
Notes for source control:

The source controll only controls the layout / script etc and NOT the data.

There is no 'deploy' functionality for Qlikview via visual studio so files will still need to be manually copied from your local Source to the server. I suspect in the future I might cobble something together in a batch file to allow me to deploy.