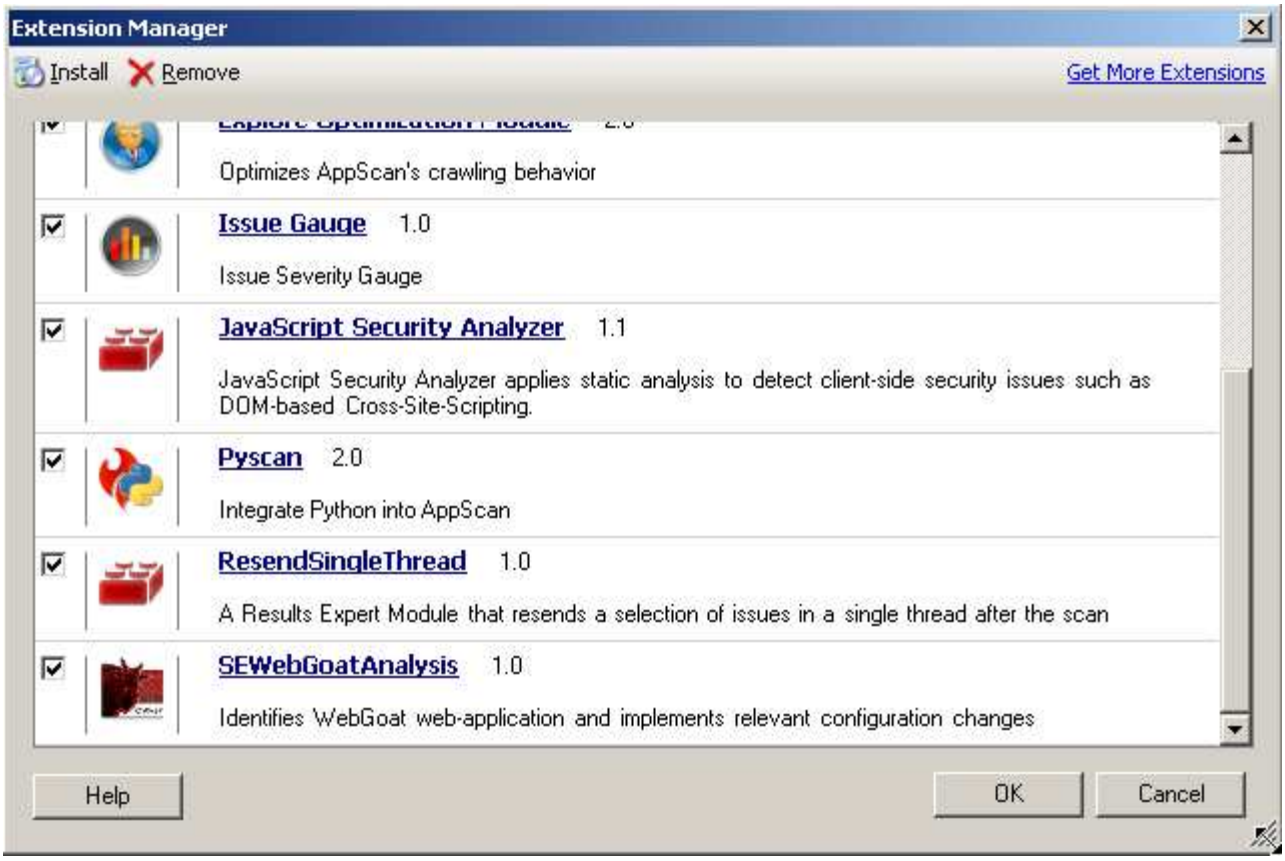
Testing and Creating an O2 Platform AppScan Standard Plug-in

Installing the O2 AppScan PlugIn (described below)

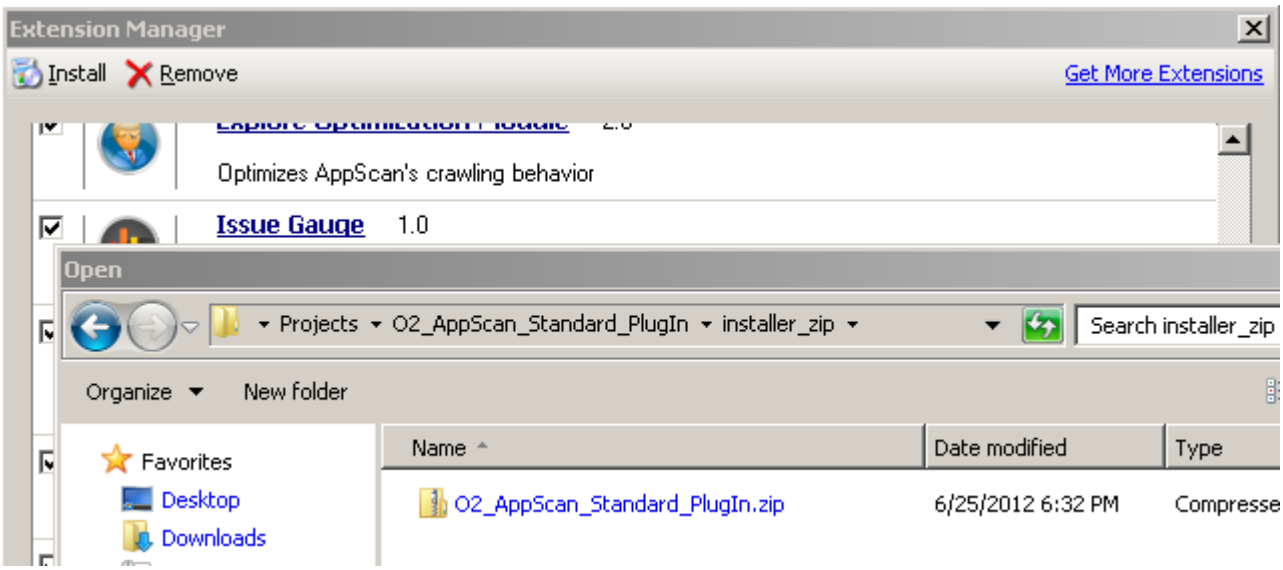
1) download the zip file from: https://github.com/downloads/o2platform/O2_AppScan_Standard_PlugIn/O2_AppScan_Standard_PlugIn.zip

2) open AppScan

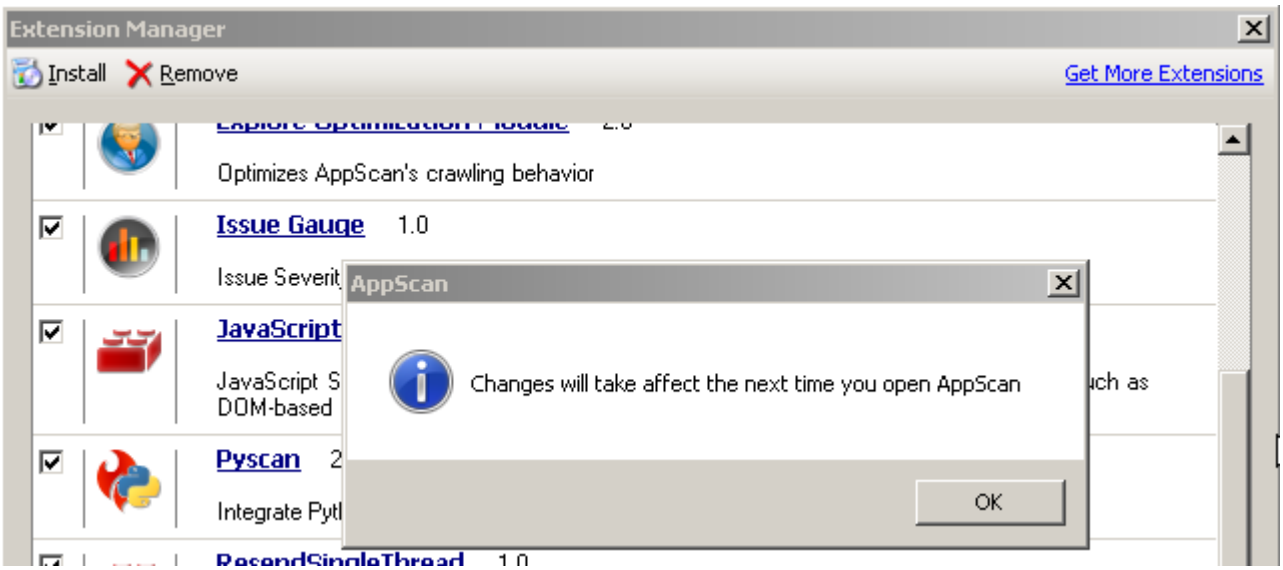
3) open the Extension Manager (in the menu Tools -> Extensions)



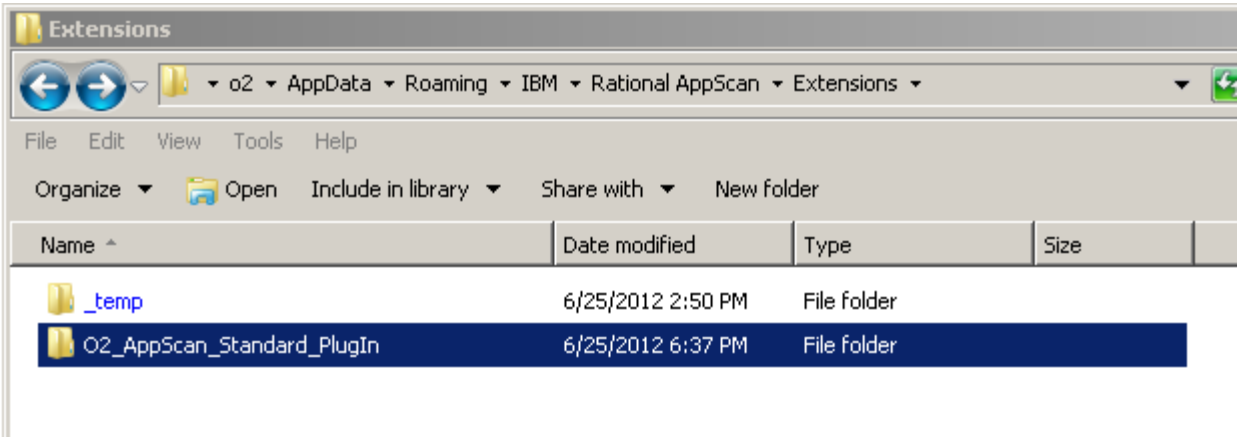
4) click on Install button and select the zip file



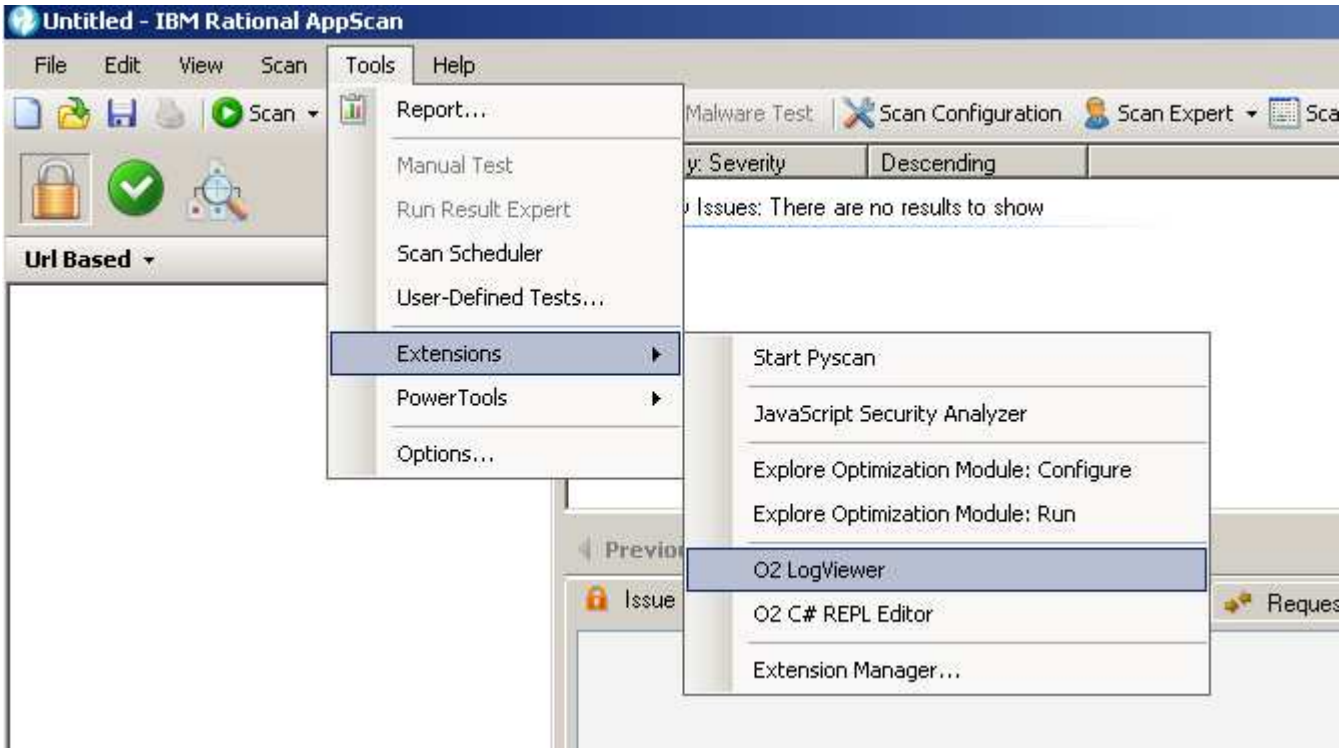
5) reboot:



note: if all went ok, you should have a new folder in the *C:\Users\o2\AppData\Roaming\IBM\Rational AppScan\Extensions'* folder



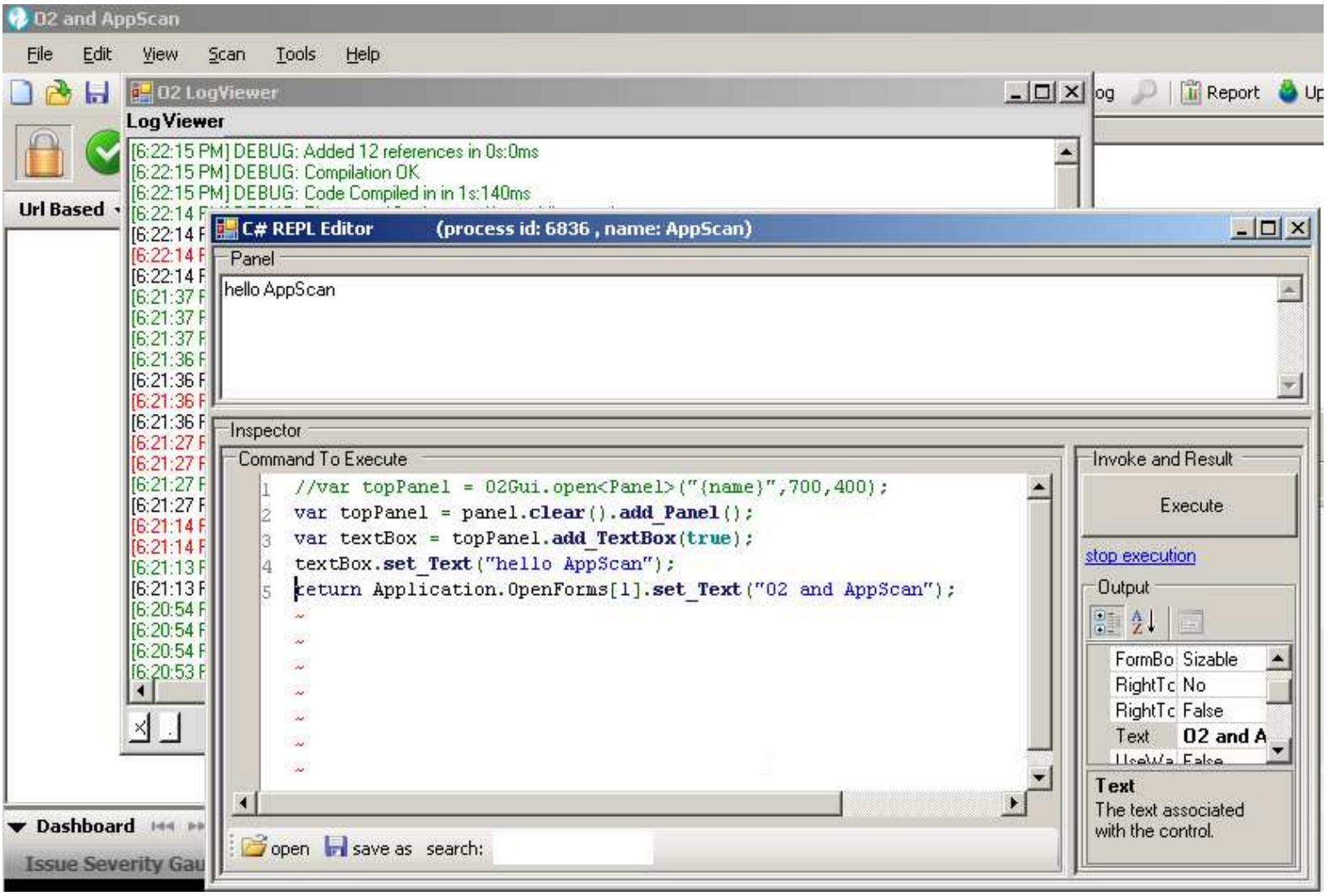
6) after reboot there will be two new menu items in the Tools -> Extensions menu:



7) Which when executed will provide access to the main AppScan form:

```
var topPanel = panel.clear().add_Panel();
var textBox = topPanel.add_TextBox(true);
textBox.set_Text("hello AppScan");
return Application.OpenForms[1].set_Text("O2 and AppScan");
```

Executing the script shown above will add a textbox to the O2 C# REPL environment and change the title of the AppScan Form



Development (of the PlugIn show above)

Based on the info from : http://www.ibm.com/developerworks/rational/downloads/08/appscan_ext_framework/#PUBLIS81

The source code of this PlugIn it at: https://github.com/o2platform/O2_AppScan_Standard_PlugIn **but its is better if you follow the step-by-step workflow provided below**

github Search... Explore Gist Blog Help o2platform

o2platform / O2_AppScan_Standard_PlugIn Admin Pull Request Unwatch 1 Fork 1

Code | Network | Pull Requests 0 | Issues 0 | Wiki | Graphs

O2 AppScan Standard PlugIn — Read more

Clone in Windows | ZIP | HTTP | SSH | Git Read-Only | git@github.com:o2platform/O2_AppScan_Standard_P | Read+Write access

branch: master | Files | Commits | Branches 1 | Tags | Downloads 1

Latest commit to the **master** branch

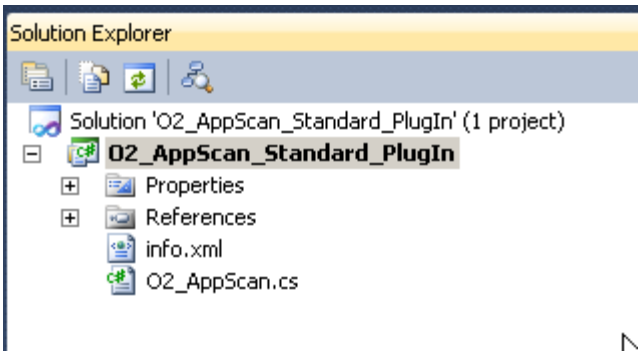
First Commit

DinisCruz authored 14 minutes ago commit 1d6f35cf6a

O2_AppScan_Standard_PlugIn /

name	age	message	history
Properties	14 minutes ago	First Commit [DinisCruz]	
_External_Dlls	14 minutes ago	First Commit [DinisCruz]	
.gitignore	15 minutes ago	Initial commit [o2platform]	
O2Logo_Small.gif	14 minutes ago	First Commit [DinisCruz]	
O2_AppScan.cs	14 minutes ago	First Commit [DinisCruz]	

Create a visualStudio project



with this code on O2_AppScan.cs

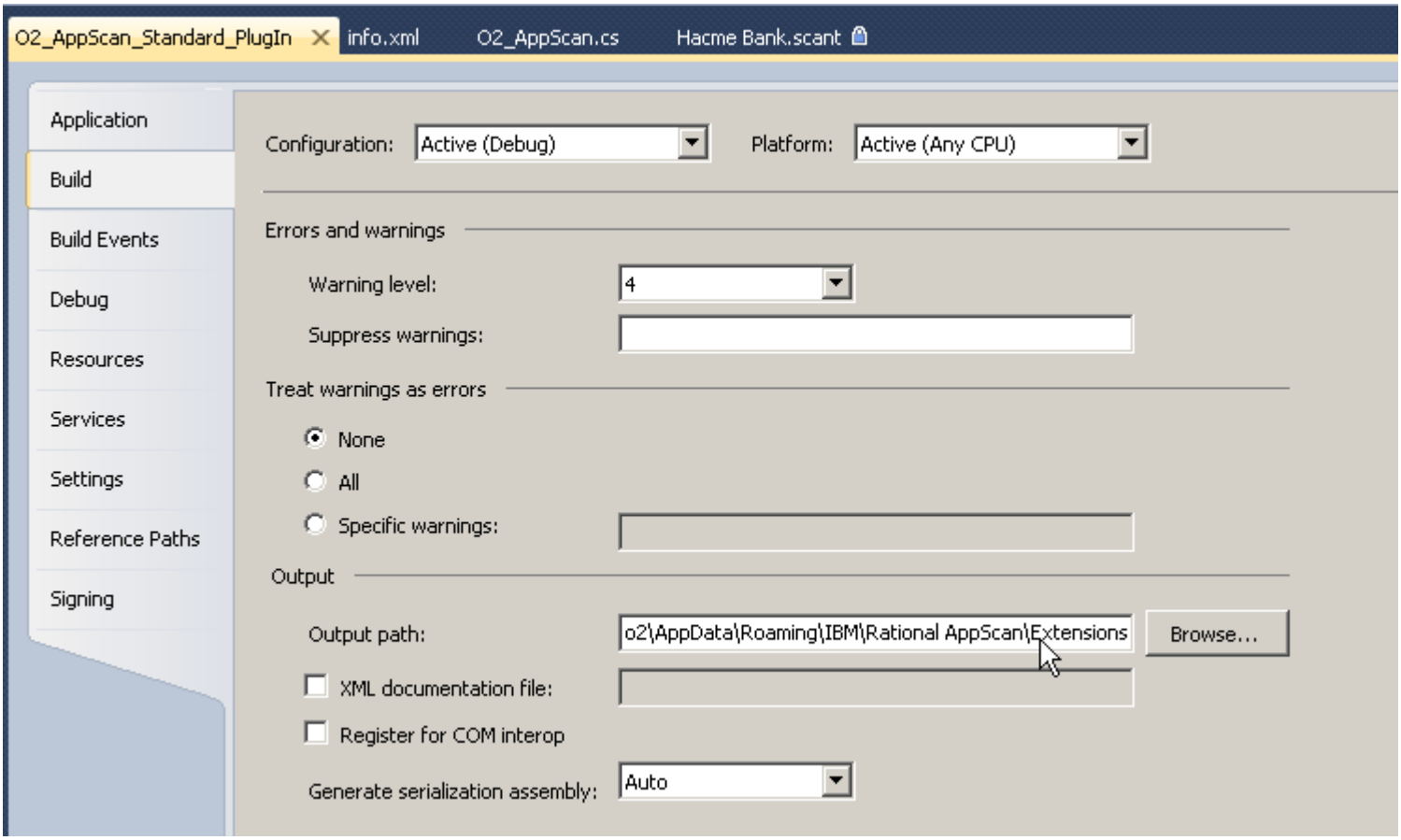
```
using System;
using System.Collections.Generic;
using System.Windows.Forms;
using AppScan;
using AppScan.Scan;
using AppScan.Events;
using AppScan.Extensions;
using AppScan.Scan.Events;

namespace MyExtension1
```

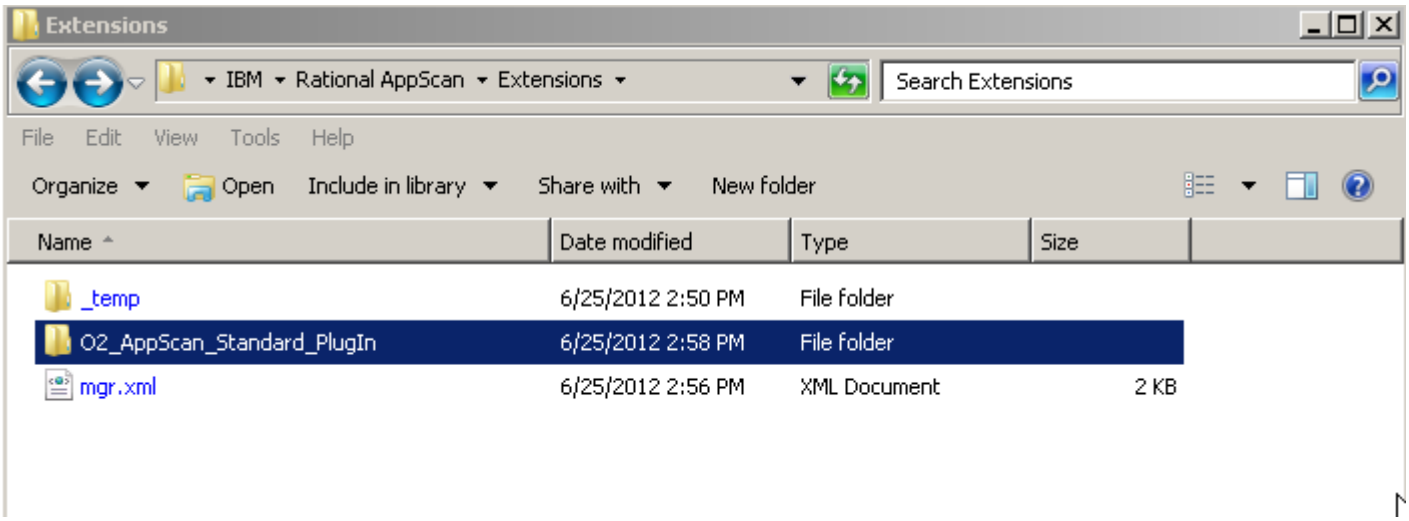
```
{
    public class MyExtension : IExtensionLogic
    {
        IAppScan appScan;

        public void Load (IAppScan appscan, IAppScanGui appScanGui, string extensionDir)
        {
            MessageBox.Show("In AppScan Plug-in Load");
            appScan = appscan;
            RegisterToAppScanEvents();
        }

        private void RegisterToAppScanEvents()
        {
                appScan.Scan.StateChanged += Scan_StateChanged;
        }

        private void Scan_StateChanged(object sender, StateChangedEventArgs e)
        {
                if (e.CurrentState == ScanOperationState.Exploring)
                {
                    MessageBox.Show("Now Exploring");
                }
        }



        public ExtensionVersionInfo GetUpdateData(Edition targetApp, Version targetAppVersion)
        {
                return null;
        }
      }

}
```

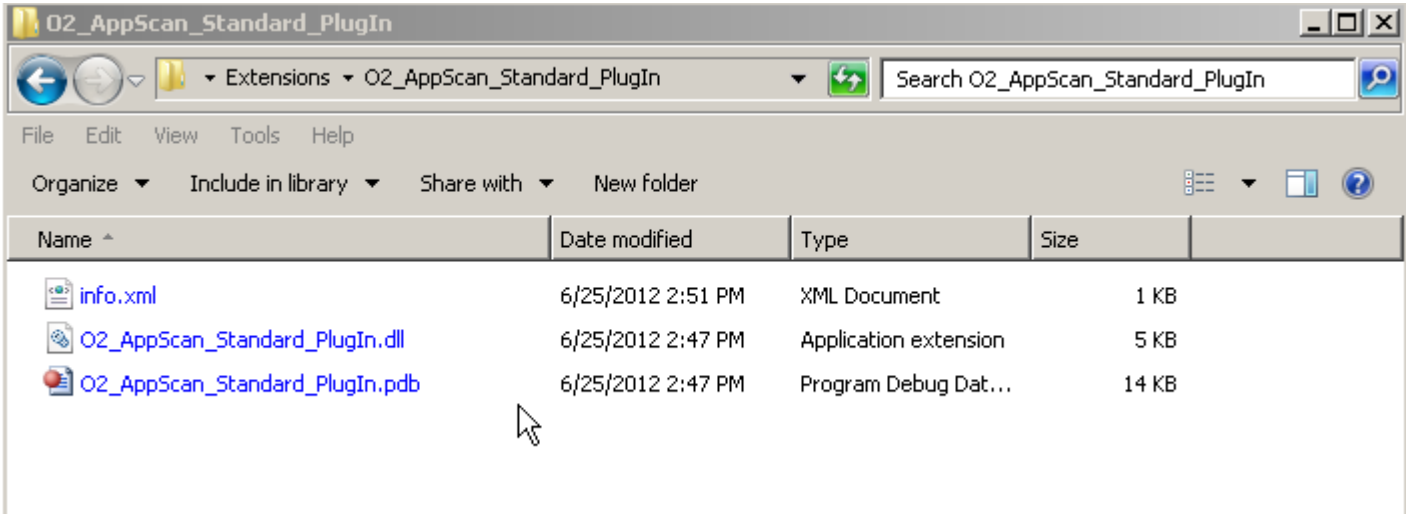
and this on info.xml file:

```
<?xml version="1.0"?>
<AppScanExtension>
  <FullName>O2 Platform</FullName>
  <Description>O2 Platform Plug-in for AppScan</Description>
  <Version>1.0</Version>
  <Author>Author Name</Author>
  <Copyright>My Copyright info</Copyright>
  <HomepageURL>http://o2platform.com</HomepageURL>

  <IconFile>O2_Logo.gif</IconFile>

  <MainDllFile>O2_AppScan_Standard_PlugIn.dll</MainDllFile>
  <TargetEdition>

    <ID>All</ID>
    <MinCompatibleVersion>7.5</MinCompatibleVersion>
  </TargetEdition>
</AppScanExtension>
```

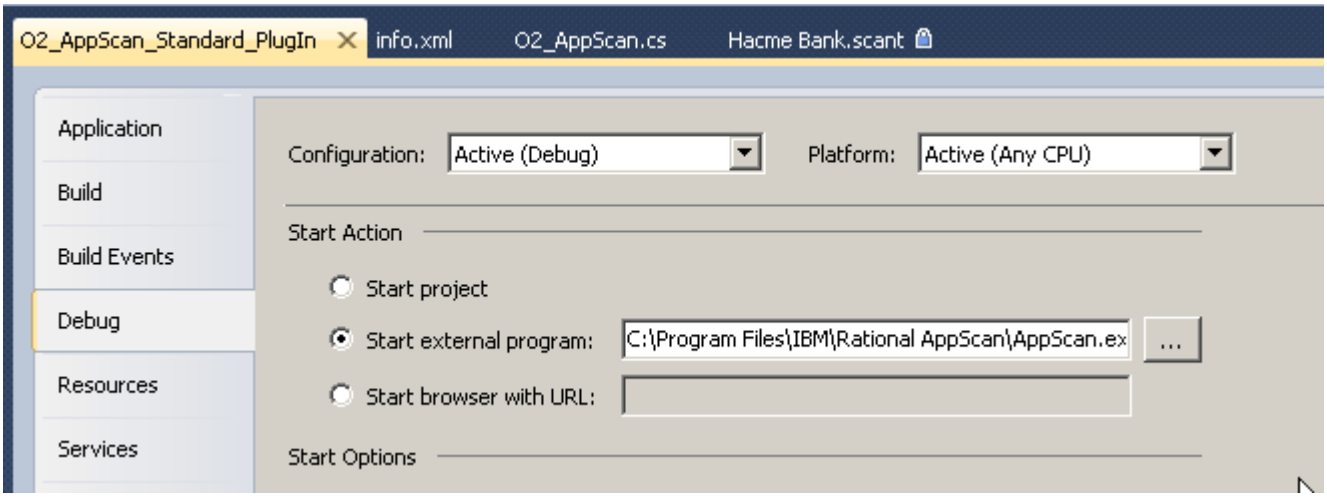
There some unzip issues (the import doesn't seem to like the windows zip) so to help the process, I set the VisualStudio compilation to the **C:\Users\o2\AppData\Roaming\IBM\Rational AppScan\Extensions\O2_AppScan_Standard_PlugIn** folder (this will also allow the use of the compiled plug-in without needing to zip it)


O2_AppScan_Standard_PlugIn
info.xml
O2_AppScan.cs
Hacme Bank.scant
Application
Build
Build Events
Debug
Resources
Services
Settings
Reference Paths
Signing
Configuration: Active (Debug)
Platform: Active (Any CPU)
Errors and warnings
Warning level: 4
Suppress warnings:
Treat warnings as errors
None
All
Specific warnings:
Output
Output path: o2\AppData\Roaming\IBM\Rational AppScan\Extensions
Browse...
XML documentation file:
Register for COM interop
Generate serialization assembly: Auto


Extensions
IBM ▸ Rational AppScan ▸ Extensions
Search Extensions
File Edit View Tools Help
Organize Open Include in library Share with New folder
Name
Date modified
Type
Size
_temp
6/25/2012 2:50 PM
File folder
O2_AppScan_Standard_PlugIn
6/25/2012 2:58 PM
File folder
mgr.xml
6/25/2012 2:56 PM
XML Document
2 KB



Configure appscan to run from visual studio,



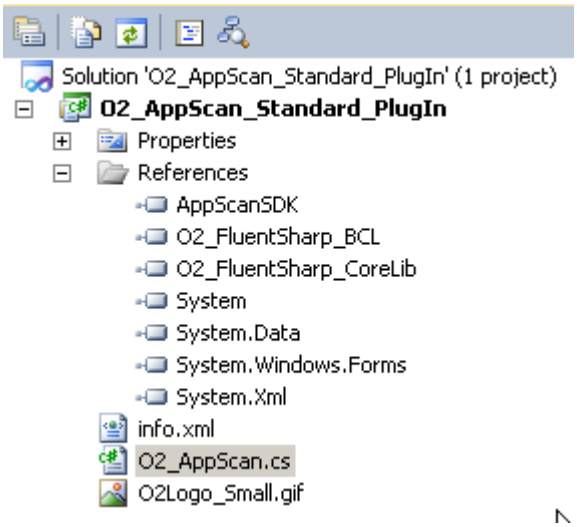
and if all goes well, press F5, and after a while, we will get a breakpoint hit here:

O2_AppScan_Standard_PlugIn | info.xml | O2_AppScan.cs | Hacme Bank.scant

MyExtension1.MyExtension | Load(IAppScan appscan, IAppScanG

```
 8  using AppScan.Scan.Events;
 9
10  namespace MyExtension1
11  {
12      public class MyExtension : IExtensionLogic
13      {
14          IAppScan appScan;
15
16          public void Load (IAppScan appscan, IAppScanGui appScanGui, string extensionDir)
17          {
18              MessageBox.Show("In AppScan Plug-in Load");
19              appScan = appscan;
20              RegisterToAppScanEvents();
21          }
22
23          private void RegisterToAppScanEvents()
24          {
25                  appScan.Scan.StateChanged += Scan_StateChanged;
26          }
27
```

Since that works, lets add the references the the core O2 libs



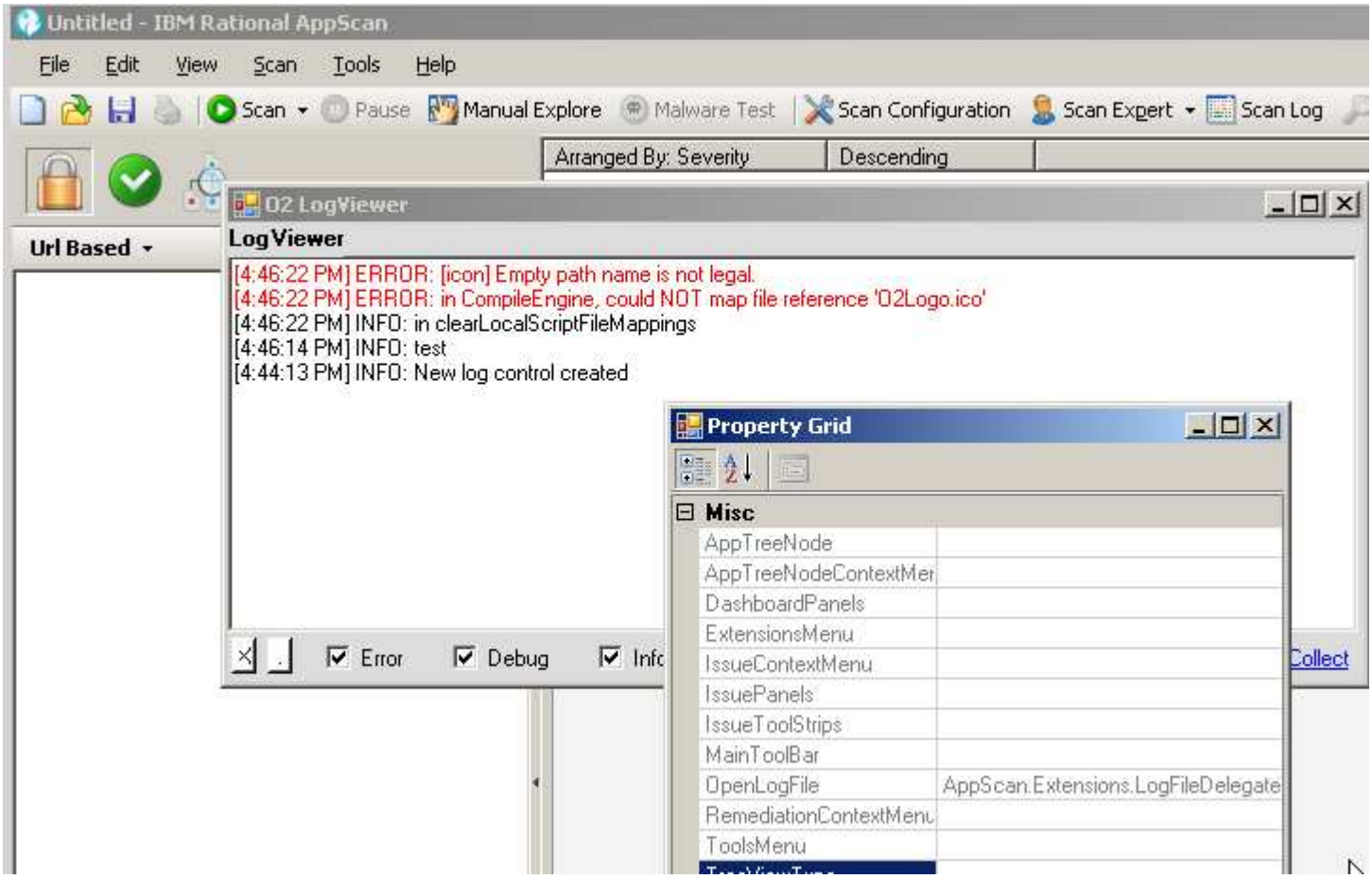
and show an O2 Log Viewer on load:

```
public void Load (IAppScan appscan, IAppScanGui appScanGui, string extensionDir)
        {
            "O2 LogViewer".popupWindow().add_LogViewer();
            appScan = appscan;
            RegisterToAppScanEvents();
        }
```

And see the properties of the appScanGui object

```
public void Load (IAppScan appscan, IAppScanGui appScanGui, string extensionDir)
        {
            "O2 LogViewer".popupWindow().add_LogViewer();
            "test".info();
```

```
            appScan = appscan;
            RegisterToAppScanEvents();

            O2.Kernel.show.info(appScanGui);

        }
```

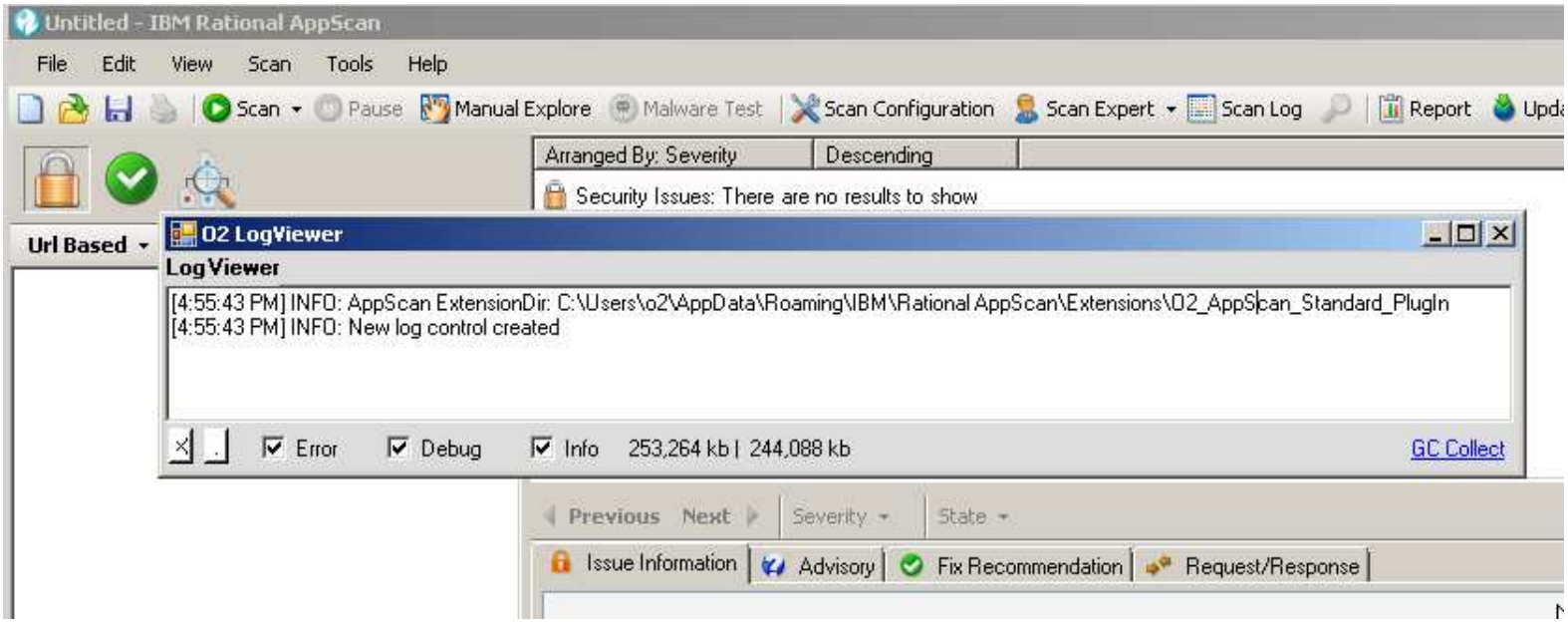


Simple version with a log viewer a number global static variables to hold the main object

```
using System;
using System.Collections.Generic;
using System.Windows.Forms;
using AppScan;
using AppScan.Scan;
using AppScan.Events;
using AppScan.Extensions;
using AppScan.Scan.Events;
using O2.DotNetWrappers.ExtensionMethods;

namespace O2.AppScan
{
    public class O2_AppScan_Standard_PlugIn : IExtensionLogic
    {
        public static IAppScan      AppScan          { get; set; }
        public static IAppScanGui   AppScanGui       { get; set; }
        public static string        ExtensionDir    { get; set; }

        public void Load (IAppScan appscan, IAppScanGui appScanGui, string extensionDir)
        {
            "O2 LogViewer".popupWindow().add_LogViewer();
            "AppScan ExtensionDir: {0}".info(extensionDir);

            AppScan = appscan;
            AppScanGui = appScanGui;
            ExtensionDir = extensionDir;
```

```
        }


        public ExtensionVersionInfo GetUpdateData(Edition targetApp, Version targetAppVersion)
        {
                return null;
        }
      }

}
```

Which looks like this:

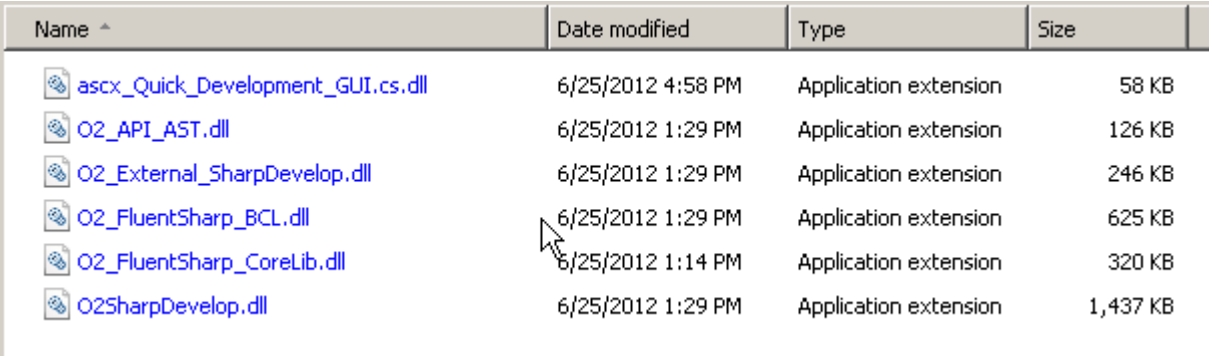


Using this script to compile O2's Main REPL C# script environment:

```
var targetDir = "O2_REPL".tempDir(false);
return "ascx_Quick_Development_GUI.cs.o2".local().compileIntoDll_inFolder(targetDir);
```

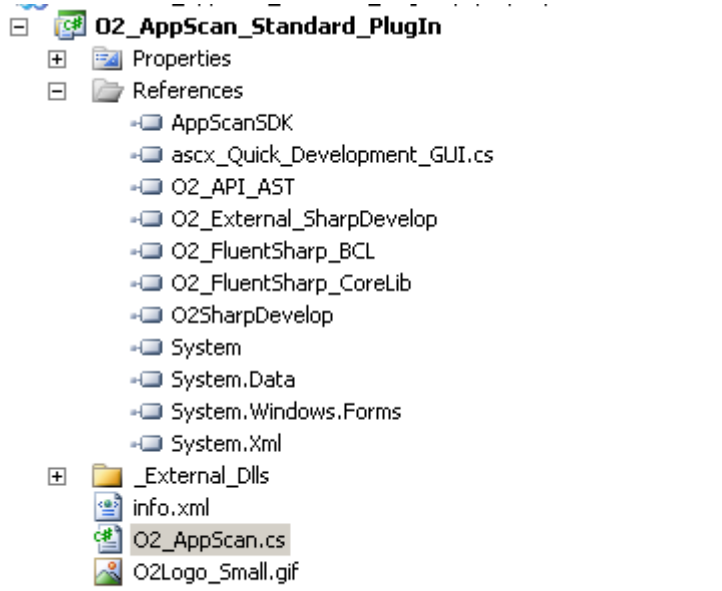
And this one to get a list of all main dependencies

```
var file =@"C:\_WorkDir\O2\O2 Install\_O2_V4_TempDir\6_25_2012\O2_REPL\ascx_Quick_Development_GUI.cs.dll";
var targetDir = file.directoryName();
foreach(var assemblyName in file.assembly().referencedAssemblies())
{
       var location = assemblyName.assembly().Location;
       if (location.lower().contains("microsoft.net").isFalse())
       {
              Files.Copy(location,targetDir);
       }
}
```



Name	Date modified	Type	Size
ascx_Quick_Development_GUI.cs.dll	6/25/2012 4:58 PM	Application extension	58 KB
O2_API_AST.dll	6/25/2012 1:29 PM	Application extension	126 KB
O2_External_SharpDevelop.dll	6/25/2012 1:29 PM	Application extension	246 KB
O2_FluentSharp_BCL.dll	6/25/2012 1:29 PM	Application extension	625 KB
O2_FluentSharp_CoreLib.dll	6/25/2012 1:14 PM	Application extension	320 KB
O2SharpDevelop.dll	6/25/2012 1:29 PM	Application extension	1,437 KB

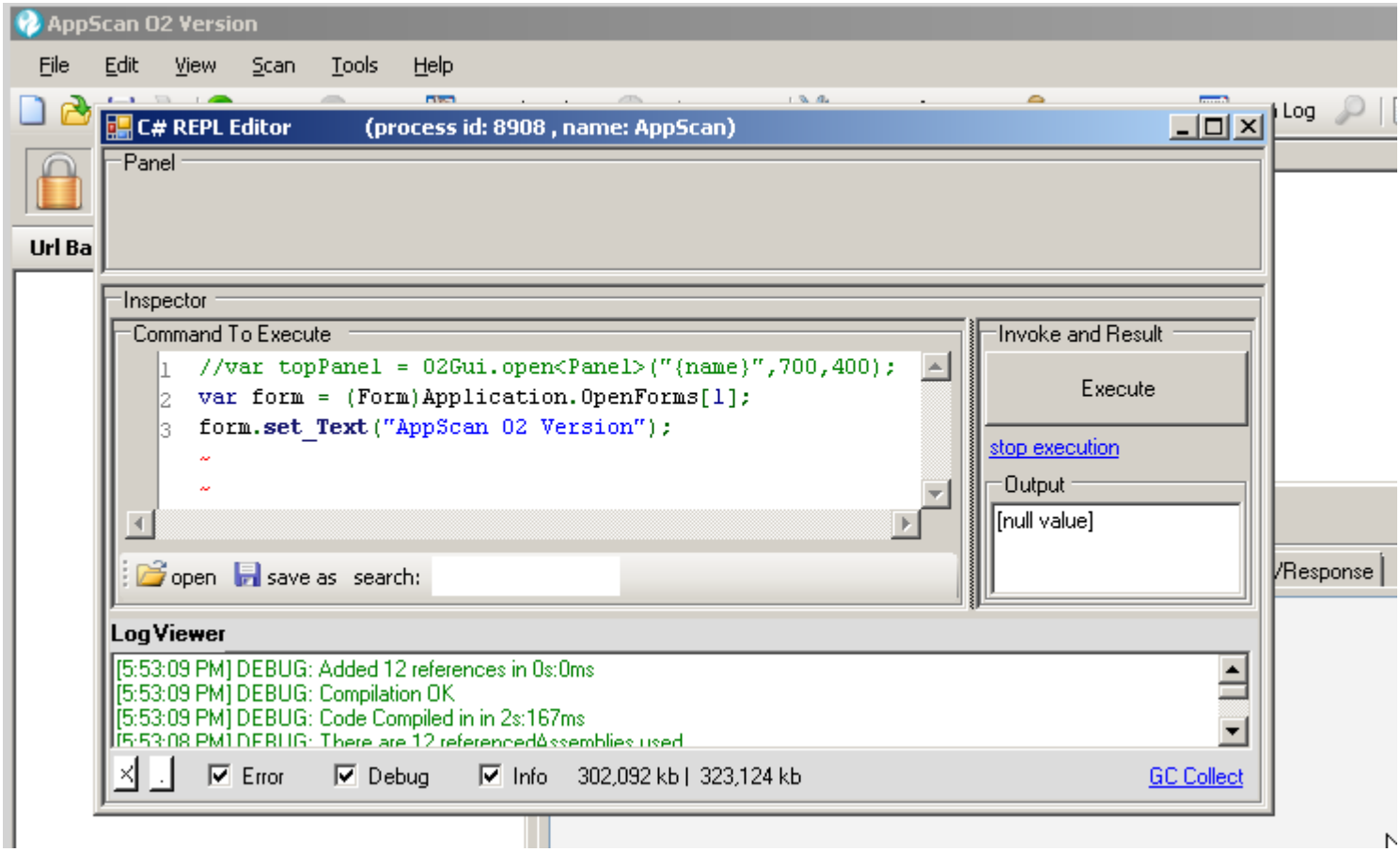
Which can be added as references to the PlugIn Project



And consumed like this

```
public void Load (IAppScan appscan, IAppScanGui appScanGui, string extensionDir)
{
    "O2 LogViewer".popupWindow().add_LogViewer();
    "AppScan ExtensionDir: {0}".info(extensionDir);
    ascx_Panel_With_Inspector.runControl();
    AppScan = appscan;
    AppScanGui = appScanGui;
    ExtensionDir = extensionDir;
}
```

When executed this will give us O2's REPL Script inside the AppScan process:



We can add a menu item like this:

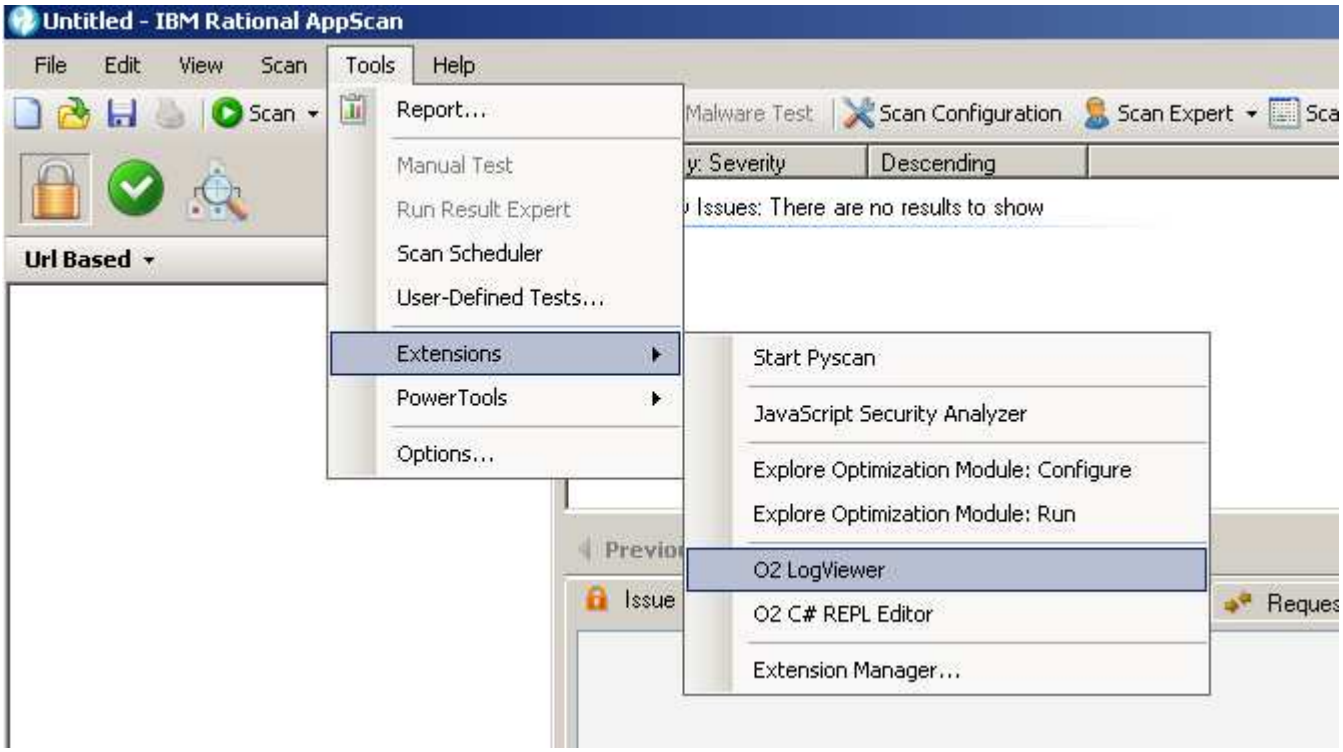
```
var menuItem_LogViewer= new AppScan.Extensions.MenuItem<EventArgs>("O2 LogViewer", (sender)=> "O2
LogViewer".popupWindow().add_LogViewer());

            AppScanGui.ExtensionsMenu.Add(menuItem_LogViewer);
```

Here is the whole script that adds two menu items to the main AppScan Tools -> Extensions menu:

```
    using System;
    using System.Collections.Generic;
    using System.Windows.Forms;
    using AppScan;
    using AppScan.Scan;
    using AppScan.Events;
    using AppScan.Extensions;
    using AppScan.Scan.Events;
    using O2.DotNetWrappers.ExtensionMethods;
    using O2.XRules.Database.Utils;

    namespace O2.APIs
    {
        public class O2_AppScan_Standard_PlugIn : IExtensionLogic
        {
            public static IAppScan      AppScan          { get; set; }
            public static IAppScanGui   AppScanGui       { get; set; }
            public static string        ExtensionDir    { get; set; }

            public void Load (IAppScan appscan, IAppScanGui appScanGui, string extensionDir)
            {
                "AppScan ExtensionDir: {0}".info(extensionDir);

                AppScan = appscan;
```

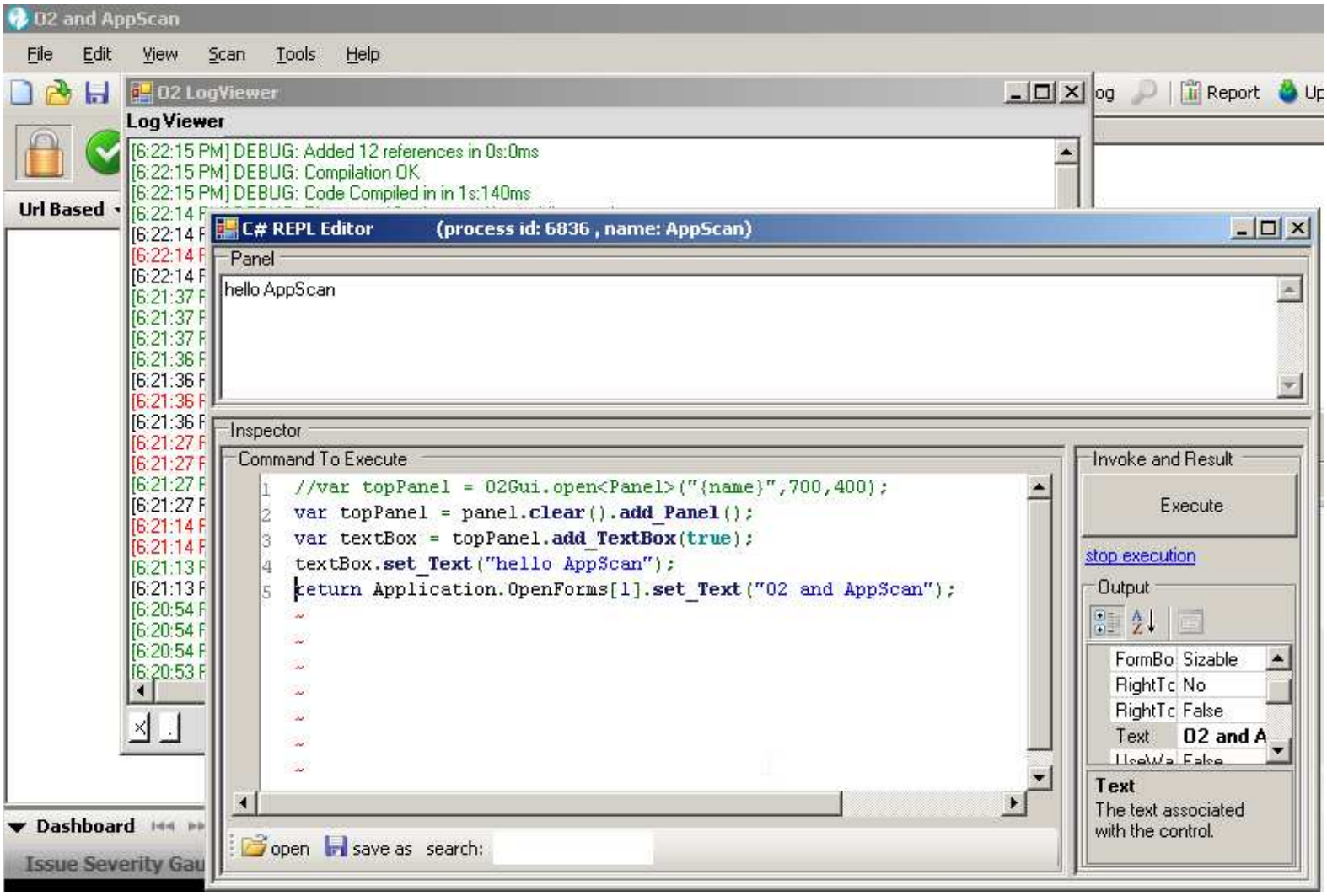
```
                AppScanGui = appScanGui;
                ExtensionDir = extensionDir;
                addO2MenuItems();
            }

            public void addO2MenuItems()
            {
                var menuItem_LogViewer= new AppScan.Extensions.MenuItem<EventArgs>("O2 LogViewer", (sender)
=> "O2 LogViewer".popupWindow().add_LogViewer());
                AppScanGui.ExtensionsMenu.Add(menuItem_LogViewer);

                var menuItem_ScriptEditor = new AppScan.Extensions.MenuItem<EventArgs>("O2 C# REPL Editor",
(sender)=> ascx_Panel_With_Inspector.runControl());
                AppScanGui.ExtensionsMenu.Add(menuItem_ScriptEditor);
            }

            public ExtensionVersionInfo GetUpdateData(Edition targetApp, Version targetAppVersion)
            {
                    return null;
            }
        }

    }
```

i.e. here:



And now we have access to the main AppScan form:

```
var topPanel = panel.clear().add_Panel();
var textBox = topPanel.add_TextBox(true);
textBox.set_Text("hello AppScan");
return Application.OpenForms[1].set_Text("O2 and AppScan");
```


O2 and AppScan
File Edit View Scan Tools Help
O2 LogViewer
Log Viewer
[6:22:15 PM] DEBUG: Added 12 references in 0s:0ms
[6:22:15 PM] DEBUG: Compilation OK
[6:22:15 PM] DEBUG: Code Compiled in in 1s:140ms
Report
Url Based
C# REPL Editor (process id: 6836 , name: AppScan)
Panel
hello AppScan
Inspector
Command To Execute
1 //var topPanel = O2Gui.open<Panel>("{name}",700,400);
2 var topPanel = panel.clear().add_Panel();
3 var textBox = topPanel.add_TextBox(true);
4 textBox.set_Text("hello AppScan");
5 return Application.OpenForms[1].set_Text("O2 and AppScan");
Invoke and Result
Execute
stop execution
Output
FormBo Sizable
RightTc No
RightTc False
Text O2 and A
Text
The text associated with the control.
Dashboard
Issue Severity Gau
open
save as
search: