Scripting DbgHostLib

This lib was created from the Debug Diagnostic Tool which includes a wrapper for DbgEng (WinDbg Extension)

Related posts:

- **Debug Diagnostic Tool v1.2** www.microsoft.com/en-us/download/details.aspx?id=26798
- **How to Use the Debug Diagnostic Tool v1.1 (DebugDiag) to Debug User Mode Processes** http://msdn.microsoft.com/en-us/library/ff420662.aspx
- **Using Managed Code to debug Memory Dumps** http://naveensrinivasan.com/2010/11/11/using-managed-code-to-debug-memory-dumps/
- **How to control a debugger engine?** http://codenasarre.wordpress.com/2011/06/14/how-to-control-a-debugger-engine/
- Anatomy of a DbgEng Extension DLL http://msdn.microsoft.com/en-us/library/windows/hardware/ff538069(v=vs.85).aspx
- SOS.dll (SOS Debugging Extension) http://msdn.microsoft.com/en-us/library/bb190764(v=VS.100).aspx
- S.A.D. or S(imple) A(fter) D(ump) http://codenasarre.wordpress.com/2011/06/26/s-a-d-or-simple-after-dump/
- Using sosex within windbg to understand IL and Assembly code http://naveensrinivasan.com/2010/11/29/using-sosex-within-windbg-to-understand-il-and-assembly-code/
- Updating .NET String in memory with Windbg http://naveensrinivasan.com/2011/06/14/updating-net-string-in-memory-with-windbg/
- Now Available: Psscor4 – Debugger Extension for .NET 4.0 http://blogs.msdn.com/b/tom/archive/2011/04/28/now-available-psscor4-debugger-extension-for-net-4-0.aspx
- LeakShell or how to (almost) automatically find managed leaks http://codenasarre.wordpress.com/2011/05/18/leakshell-or-how-to-automatically-find-managed-leaks/#comment-38

Stand-alone Script to create the dll

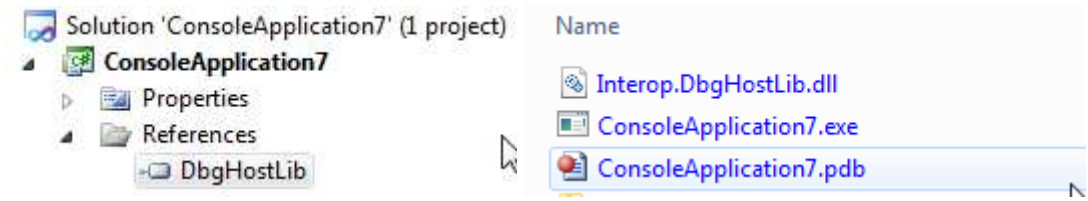
```
var tlbimp = @"C:\Program Files (x86)\Microsoft SDKs\Windows\v7.0A\Bin\x64\tlbimp.exe";
var fileToConvert = @"C:\Program Files\DebugDiag\DbgHost.exe";
var tmpDir = "_dbgLib".tempDir(false);
tlbimp.startProcess_getConsoleOut("\"" + fileToConvert + "\"", tmpDir).info();
```

Script that is part of of the Installer_Debug_Diagnostic_Tool.cs

```
                    if (dgbHostDll.fileExists().isFalse())
                    {
                          var tlbimp = (this.ProgramFilesFolder.contains("x86"))
                                                ? this.ProgramFilesFolder.pathCombine(@"\Microsoft SDKs\Windows\v7.0A\Bin\x64\tlbimp.exe")
                                                : this.ProgramFilesFolder.pathCombine(@"\Microsoft SDKs\Windows\v7.0A\Bin\tlbimp.exe");
                          var fileToConvert = this.Install_Dir.pathCombine(@"DbgHost.exe");
                          tlbimp.startProcess_getConsoleOut("\"" + fileToConvert + "\"",
```

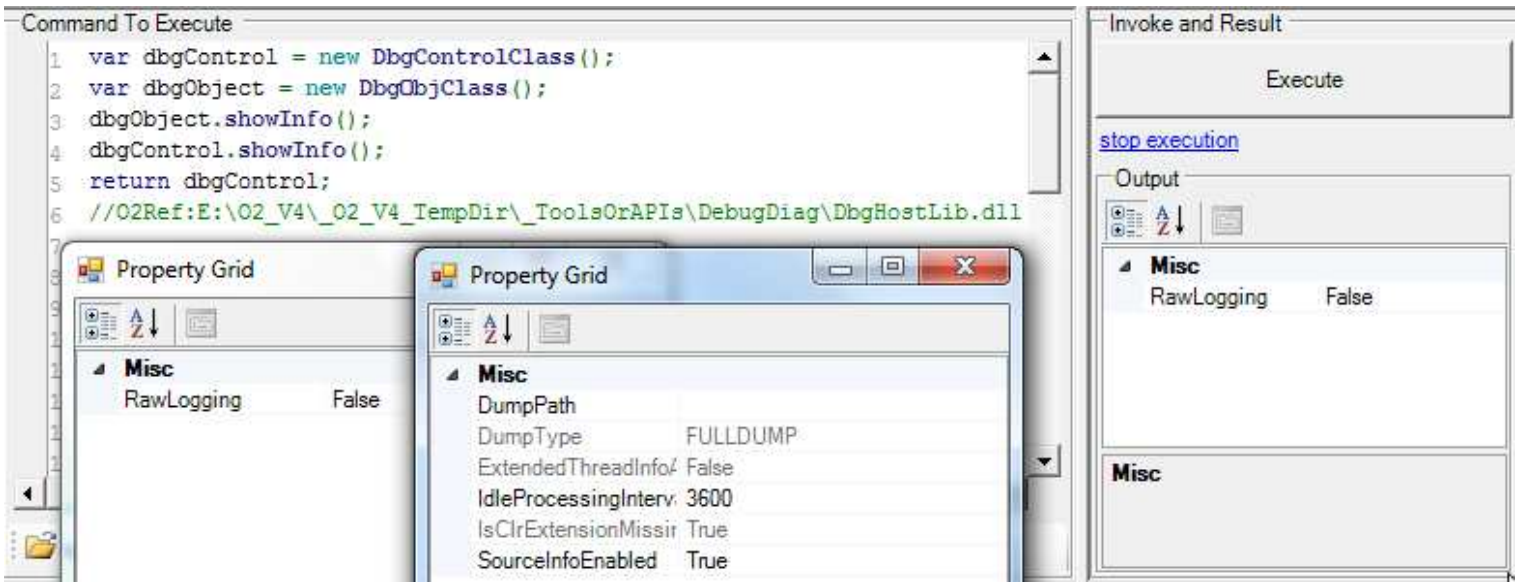
```
debugDiagDir).info();

                        "Created file: {1}".info(dgbHostDll);
                }
```

Note that if you create this using VisualStudio (like the example in the blog posts) you will get a dll called Interop.DbgHostLib.dll



We can now access the DbgHostLib.dll from an O2 Script (with direct reference to the Dll):

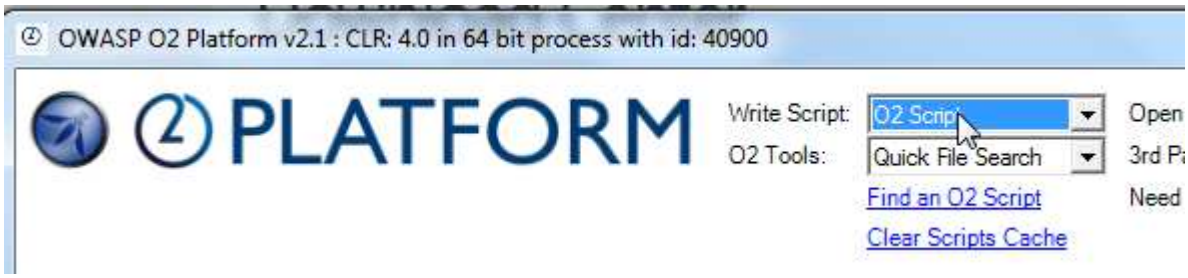
```
var dbgControl = new DbgControlClass();
var dbgObject = new DbgObjClass();
dbgObject.showInfo();
dbgControl.showInfo();
return dbgControl;
//using DbgHostLib
//O2Ref:E:\O2_V4\_O2_V4_TempDir\_ToolsOrAPIs\DebugDiag\DbgHostLib.dll
```



Or using a relative reference

```
var dbgControl = new DbgControlClass();
return dbgControl;
//using DbgHostLib
//O2Ref:DebugDiag\DbgHostLib.dll
```

Note that you will need to start the O2 REPL editor from the version of x64 version of O2 (if you are in a 64 bit box):



Now that we have the DbgHostLib.dll we can use it start attaching to a running process

```
var exe = @"E:\O2_V4\_O2_V4_TempDir\6_30_2012\Util - LogViewer [08145]\Util - LogViewer.exe";
var process = exe.startProcess();
var scriptPath = "_dbgctl".tempDir();
var symbolPath = "C:\\symcache";
var dumpPath = scriptPath;
var dbgControl = new DbgControlClass();

dbgControl.AttachToProcess(process.Id, scriptPath,symbolPath,dumpPath);
return dbgControl;

//using DbgHostLib
//O2Ref:DebugDiag\DbgHostLib.dll

//O2Tag_DontAddExtraO2Files
```

Attaching and killing the host process

```
var exe = @"E:\O2_V4\_O2_V4_TempDir\6_30_2012\Util - LogViewer [08145]\Util - LogViewer.exe";
var process = exe.startProcess();
//process.closeInNSeconds(2);
var scriptPath = "_dbgctl".tempDir();
var symbolPath = "C:\\symcache";
var dumpPath = scriptPath;
var dbgControl = new DbgControlClass();
var dbgObject = new DbgObjClass();

dbgControl.AttachToProcess(process.Id, scriptPath,symbolPath,dumpPath);
this.sleep(1000);
dbgObject.Kill();
return dbgObject;

//using DbgHostLib
//O2Ref:DebugDiag\DbgHostLib.dll

//O2Tag_DontAddExtraO2Files
```

.

.

.

.