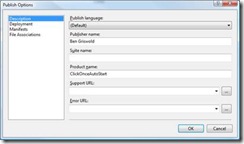
As you may have gathered from [my last post](http://johnnycoder.com/blog/2009/02/18/clickonce-getting-started-sample/), I’ve been playing around with ClickOnce a bit lately.  The latest head-scratcher is how to get a ClickOnce deployed application to run at startup.  It should be a no-brainer (just copy a shortcut in the startup folder, right?) but when you add Vista to the mix all hell breaks loose (and when I say “all hell breaks loose” I mean nothing happens at startup.)  I finally sat down this weekend and put together a sample application and some documentation to support my findings.  Any help or feedback is greatly appreciated.  The main goal is to determine a reliable, reusable way to add AutoStart functionality to any ClickOnce deployed application for any Windows operating system.  Again, piece of cake, right?

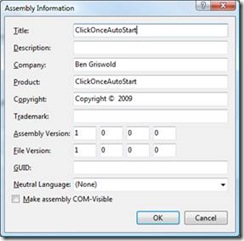
#### Quick Overview

An application shortcut is copied to the Start Menu Programs directory as part of the installation process for a ClickOnce application which is configured to be available offline1. For example, for me on Vista, the attached sample application's shortcut is placed in the following location: C:\Users\Ben Griswold\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Ben Griswold\ClickOnceAutoStart.appref-ms.

The shortcut location and file name are based on the Publisher name (Ben Griswold) and the Product name (ClickOnceAutoStart) as defined in the project's Publishing Options2.

[](file:///C:\Users\Ben%20Griswold\AppData\Local\Temp\WindowsLiveWriter-429641856\supfiles1F29E4B\clip_image002%5b3%5d.jpg)

It is important to note that the shortcut's location and file name are NOT based on the executing assembly's Company and Product values3, however, the Publishing Options and Assembly Information values are often identical (as is the case in the attached sample application.)

[](file:///C:\Users\Ben%20Griswold\AppData\Local\Temp\WindowsLiveWriter-429641856\supfiles1F29E4B\clip_image004%5b3%5d.jpg)

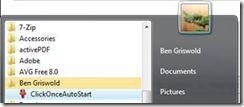
Though Visual Studio provides a means to add an application shortcut to the products folder in the Start Menu Programs directory as well as the Desktop4, there does not appear to be an out-of-the-box means to add a shortcut to Startup Folder. In other words, **there is no configuration setting to enable the ClickOnce application to run at startup**.

#### ClickOnce AutoStart Solutions

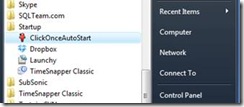
There are two sited ways to implement run at startup functionality. Both options, however, come with drawbacks and issues.

##### Solution 1

Programmatically copy the application’s Start Menu Program shortcut (the Application Reference file) into the Startup folder.

[](file:///C:\Users\Ben%20Griswold\AppData\Local\Temp\WindowsLiveWriter-429641856\supfiles1F29E4B\clip_image006%5b3%5d.jpg)

*Start Menu\Programs\Ben Griswold*

[](file:///C:\Users\Ben%20Griswold\AppData\Local\Temp\WindowsLiveWriter-429641856\supfiles1F29E4B\clip_image008%5b3%5d.jpg)

*Start Menu\Programs\Startup*

*Issue 1* - There doesn't appear to be a reliable way to determine where the ClickOnce installer placed the Start Menu Program .appref-ms shortcut. As stated above, the shortcut location is based on the Publishing Options. Can one programmatically read the Publisher and Product name? Is this information is available via the AppDomain.ActivationContext Property? The bottom-line is, if the shortcut location can’t be determined then the file can’t be copied.

Arguably, this isn’t a serious issue since the shortcut location could be “pre-calculated” and stored elsewhere in the application. For example, in Application Settings or within AssemblyInfo which seems to be the de facto standard. All the same, since this Publisher and Product are displayed on the Deployment Web Page5, it is reasonable to assume the values must be exposed.

[](file:///C:\Users\Ben%20Griswold\AppData\Local\Temp\WindowsLiveWriter-429641856\supfiles1F29E4B\clip_image010%5b3%5d.jpg)

*Issue 2* – Copying the Application Reference shortcut into the Startup folder doesn’t work reliably in Windows Vista. This is the case whether UAC is enabled or not. Interestingly, the same shortcut can successfully launch the application if it is manually executed by the user, but under standard OS startup the shortcut is never triggered.

*Issue 3* – Another issue is tied to the uninstall. Is there a way to hook into the uninstall of a ClickOnce application? If not, there doesn’t appear to be a way to remove any/all artifacts which may have been programmatically added to the end users system. In other words, after the product uninstall, it would become the end user’s responsibility to remove the shortcut file which was programmatically copied into the Startup folder. Coincidently, the Start Menu Programs shortcut and desktop shortcut are all removed when the ClickOnce application is uninstalled.

##### Solution 2

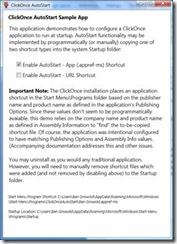
Programmatically create a URL shortcut file which maps to the ClickOnce application's update location and place it in the Startup folder. I am happy to say **this method does work reliably in Windows Vista**.

*Issue 1* - Though creating an URL shortcut file which maps to the ApplicationDeployment.CurrentDeployment.UpdateLocation.AbsoluteUri works consistently, an empty browser window is launched when the Uri is being queried. Again, this solution works but it isn't particularly clean.

*Issue 2* – Same uninstall issue as outline above.

#### Sample Application

The accompanying sample application illustrates how to enable run at startup using either of the two techniques stated above. One may validated the functionality and issues by launching the application (after installing via ClickOnce) and then enabling the AutoStart options, verifying shortcut are valid and testing the AutoStart functionality per a system restart.

[](file:///C:\Users\Ben%20Griswold\AppData\Local\Temp\WindowsLiveWriter-429641856\supfiles1F29E4B\clip_image012%5b3%5d.jpg)

Again, any help or feedback is greatly appreciated. The main goal is to determine a reliable, reusable way to add AutoStart functionality to any ClickOnce deployed application for any Windows operating system.

References:

[ClickOnce Startup Folder](http://social.msdn.microsoft.com/Forums/en-US/netfxbcl/thread/75d2112c-707c-4051-a5fc-eb51802558bb/)

[Tip: ClickOnce Deployment](http://www.keithelder.net/blog/archive/2006/07/28/Tip-ClickOnce-Deployment.aspx)

1 Project Properties > Publish > Install Model and Settings

2 Project Properties > Publish > Install Model and Settings > Options... > Description

3 AssemblyInfo.cs or Properties > Application > Assembly Information...

4 Project Properties > Publish > Install Model and Settings > Options... > Manifest > Create desktop shortcut

5 Project Properties > Publish > Install Model and Settings > Options... > Deployment.