

RE: Installer issue in deployment project VS2005

Source:

<http://www.tech-archive.net/Archive/DotNet/microsoft.public.dotnet.general/2007-08/msg00536.html>

- *From:* Bo McIlvain <bo5278@xxxxxxxxxxxxxxxx>
 - *Date:* Tue, 28 Aug 2007 08:12:00 -0700
-

Linda,

Thanks for getting back to me. I think we've almost got the problem solved, after some intense work on it yesterday.

First, your questions:

1. The MSI package was indeed created with VS2005. The package installs an Outlook 2003 COM Addin, used by a law firm client of mine to add buttons on their email so that standard disclaimers can be easily added/removed from outgoing email. I had given it to their IT dept to install, and we found a bug in it after they installed it into their standard desktop image and rolled it out to 500 users, so we wanted to uninstall/reinstall via remote process driven by Altiris, which must be a non-interactive process, hence the need to use command-line tools. While I was struggling with the failure of msiexec.exe to do the uninstall, I remembered that many setup.exe's pass command line params on to msiexec, so I tried that and got the error message that the command line params were bad. I looked in the log file, and found that the setup.exe produced by VS2005 was passing command line params, unmodified, as a string to the MsiInstallProduct API call. The MSDN documentation describes this command line as a set of MSI property settings, so that's what you saw in my example. It works pretty well, although it doesn't quite have the functionality of the msiexec.exe's normal command line switches. For example, you can't get the full range of UI choices you have with msiexec.exe. LIMITUI=TRUE works well enough for our purposes, however.

2. This is the heart of the issue. I realized after a while that the only desktops where the uninstall using setup.exe and command line params was working at all were on ones where I had uninstalled the old package at least once using Add/Remove programs. We couldn't get uninstall to work ANYWHERE except in the Windows Add/Remove Programs interface, which was unacceptable as we needed to do this un-attended.

I have concluded that somehow we have in fact a mismatch between the .msi package file I'm using for the uninstall and the one currently installed on the desktops. So, what I did (and you anticipate me in your final comment) was to go into the registry on a machine where the package was installed, search for my package name under

RE: Installer issue in deployment project VS2005

HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Uninstall, locate the uninstall string (which uses the Product Code, and not a .msi install file!), copy it to a .bat file and change the command line switch from /I to /x and added /qb to do it quietly (but not silently). This worked at the local machine's command prompt, and we are testing today to see if it works with Altiris, and I have high hopes for success at this point. So, if we ignore msixec.exe's error message about a corrupt file and address it as a product mismatch, things are beginning to make sense again, and in my experience that's the best indication that we're on the right path.

So, what appeared to be a problem with msixec opening the .msi file actually turns out to be that msixec couldn't locate the package referenced in the .msi file as an installed package on the machine, probably because the product codes don't match. I think you should pass this along not as bug in function, but as an erroneous error message. There was nothing wrong with the .msi file except that it wasn't the right one!

Thanks for your help, this has been a tough one...

Bo McIlvain

"Linda Liu [MSFT]" wrote:

Hi Bo,

Thank you for your prompt response and detailed feedback!

I'm currently working on the following work-around: use setup.exe to pass

"command line" parameters to the installer API,

Could you please tell me with what tool you create the MSI package and setup.exe? I don't remember we could pass parameters to setup.exe if the MSI package and setup.exe is created with VS2005.

This has shown some promise, but we've found instances where it fails to remove the .dll the package installs from the Global Assembly Cache.

How do you remove the DLL the package installs from the Global Assembly Cache? What's the error message when you fail to remove the DLL from GAC? For DLL that is installed by MSI, we could only uninstall it via MSI.

The following is a KB article discussing this topic:
<http://support.microsoft.com/kb/873195/en-us>

I would prefer to know I can use msixec.exe with VS2005 deployment

RE: Installer issue in deployment project VS2005

projects!

Yes, we could use msiexec.exe with a VS2005 setup project. For example, to add an uninstall program for your application, add the msiexec.exe file in the setup project and create a shortcut for this file. Set the Arguments property of the shortcut to "/x {ProductCode}".

Hope this helps.

Sincerely,
Linda Liu
Microsoft Online Community Support