

How to create a simple install system for VB6 on XP/Vista and newer?

[closed]

Asked 16 years, 4 months ago Modified 10 years, 3 months ago

Viewed 6k times



7



As it currently stands, this question is not a good fit for our Q&A format. We expect answers to be supported by facts, references, or expertise, but this question will likely solicit debate, arguments, polling, or extended discussion. If you feel that this question can be improved and possibly reopened, [visit the help center](#) for guidance.

Closed 12 years ago.

Heavy emphasis on simple. I've never made an installer and I'd rather not have to learn much. A system that I could hand a pile of files to and it would make some smart guesses about where to put them would be ideal.

Go ahead and answer the general question.

However In my cases I'm stuck with some extra constraints. The program to be installed is written in VB6 (or is it 5?) and a few previous versions of VB, so it's *not* going to be updated any time soon. I have a running install and will have a Clean VM to play with So I'll be

doing a loop of: run the install, find where it's broken, fix it, add that to the installer, revert the VM, try again. If anyone has a better approach I'm open to suggestions.

I *MUST* get it working on XP and I'd really like to also have something that will work on newer versions of Windows as well.

windows

vb6

installation

Share

Improve this question

Follow

edited Sep 18, 2014 at 9:49



Deanna

24.2k ● 7 ● 73 ● 159

asked Aug 23, 2008 at 1:50



BCS

78.3k ● 69 ● 194 ● 298

possible duplicate of [What is the best choice for building Windows installers?](#) – [saschabeaumont](#) Jun 4, 2010 at 0:20

@sascha Not a duplicate, because installing VB6 programs is a little different to installing .Net programs – [MarkJ](#) Jan 12, 2011 at 9:20

7 Answers

Sorted by:

Highest score (default)



[InnoSetup](#) or [NSIS](#), whichever seems easier to you. [ISTool](#) is a nice GUI tool for InnoSetup which makes creating setup scripts even easier.

11

Share Improve this answer

answered Aug 23, 2008 at 2:16

Follow



Imran

90.8k ● 26 ● 100 ● 131



I'm using Inno (I didn't try the other) and find it easy to use. A bit light on the docs though. – [BCS](#) Sep 18, 2008 at 22:25

NSIS is excellent! been using it for years. you can make an extremely simple installer but you can also get as involved as you want/need. – [Nerdtron](#) Nov 10, 2011 at 12:18



5

I've used InnoSetup several years ago, before Vista, and was very happy with it then. I only had a few files to install and a Start menu icon. It worked great, and was easy to learn.



Share Improve this answer

answered Aug 23, 2008 at 3:26

Follow



Brian Ensink

11.2k ● 3 ● 51 ● 63



4

[Dependency Walker](#) is super useful for finding out which dll is missing from the installer. Once you know the dll, you can find what merge module it is in using the [Merge Module Finder](#).



Share Improve this answer

answered Aug 23, 2008 at 1:57

Follow



Nick

13.4k ● 17 ● 66 ● 100



Dependency Walker was /Really/ handy – [BCS](#) Sep 18, 2008 at 22:24



1

I have worked with NSIS and getting past some of its minor complexities its a fantastic system. its free, offers tons of plugin ability and managed to do everything I needed to do.



Share Improve this answer

answered Aug 23, 2008 at 2:19

Follow



Tanerax

5,856 ● 5 ● 30 ● 28



1

Creating a full setup package for a program is almost a subject area in itself. There are many factors to consider and most of us aren't running Windows 95 anymore. The world is not as simple as it once was.



There are a lot of things that need to be addressed, and some of these "setup" issues mean changing the program too. For example the "protected folders" concept that seemed to be new to people when Vista UAC came on the scene. I guess they were all running as admin or something? In its simplest form it means you don't put



writable files next to the EXE in Programs (aka "Program Files") anymore.

Another factor is that the way the registry is used has changed. I'm not talking about registry virtualization, though that's part of it as well. But COM registration can be done both per-machine and per-user and even turning UAC off can muck this up. See [Per-User COM Registrations and Elevated Processes with UAC on Windows Vista SP1](#). The result is that a setup package shouldn't be running regsvr32 (or otherwise calling the self-reg entrypoint of a COM library). See "Remarks" at [SelfReg Table](#).

Windows Installer is *the* way to go forward in most cases. VB6 programmers have Visual Studio Installer 6.0 version 1.1 available as a free download for creating MSI packages. See "COM Servers" at the VFP article [Using Microsoft Visual Studio Installer for Distributing Visual FoxPro 6.0 Applications](#) for some valuable information.

This isn't the easiest option but there is a VB Setup Wizard in VSI 1.1 to help get the basics right. Doing advanced things like creating a [CommonAppData] subfolder and setting Everyone rights on it has to be done in a post-build step outside the IDE. That's where 3rd party tools can be useful to give you more control without resorting to Orca or post-build Installer scripts.

Those guys making scripted "legacy" installers try to keep up, but the scripting gets more and more complicated.

The results are sometimes iffy. Windows 7 introduces a few new wrinkles of its own.

While ClickOnce isn't really the best option for VB6, nothing says you can't use reg-free COM for XCopy installs of many programs. Reg-free COM can even be a good option for use in an Installer package for that matter.

So in the end the "simplest" way to deploy VB6 programs is probably going to be reg-free COM XCopy packages wrapped in a self-extracting EXE that will fire off a script to create a Start Menu shortcut. If you can live without the shortcut this is even easier: just unzip the package where it needs to go!

See [Make My Manifest](#) or alternative tools for reg-free COM packaging.

This requires that the target systems be running XP (preferably SP2) or later. The only possible glitch here is that XP did not include the VB6 SP6 runtimes until XP SP3, so you'll want to test your program against the VB6 SP5 runtimes first. Well one more glitch: you can't use ActiveX EXEs this way, they still require registration.

Share Improve this answer

Follow

answered Aug 25, 2009 at 13:32



Bob77

13.3k ● 1 ● 30 ● 37



0



My advice is this. Try to keep the installer as simple as possible. Windows Installer is a very complicated piece of software and when things don't work right it can be hard to figure out what's going on. I'm sure we have all experienced the endless loop of Windows Installer trying to repair a file that you no longer have the source .msi file for.

Most of the time using Windows Installer is like using a sledge hammer to crack a nut.

I use InnoSetup for my own stuff and InstallShield at work (against my will). Start with a simple script based installer and only use Windows Installer if you have a good reason to.

Note that support for installing assemblies to the GAC may be missing for some non Windows Installer setup tools (such as InnoSetup).

Share Improve this answer

answered Aug 30, 2008 at 2:44

Follow



[jmatthias](#)

7,595 ● 7 ● 30 ● 37



0



I **used** to LOVE Inno Setup. Emphasis on "used to".

When you run the single file installer (what you'd typically do), it unpacks the real setup program into a folder under the temp folder and then tries to execute it. The problem is... some anti-virus programs don't allow this.



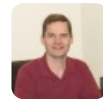
The author is aware of this and refuses to do anything about it. The folder name is random, so cannot be added to any exemption list your anti-virus program may use.

Again. The author is aware of this and suggests that I tell my users to turn off their anti-virus programs during installation. (Like that's going to happen)

Share Improve this answer

answered Dec 11, 2008 at 11:51

Follow



Brad Bruce

7,807 ● 3 ● 42 ● 61

-
- 5 That's a problem with the anti-virus software. What INNO setup is doing is perfectly reasonable and valid thing to do. It is also superior to the unload everything into the temp folder and run from there. – [bruceatk](#) Dec 11, 2008 at 11:58
-