

# Re: How to Deploy MFC only based application?

---

*Source:* <http://www.tech-archive.net/Archive/VC/microsoft.public.vc.mfc/2006-03/msg01929.html>

---

- *From:* Uwe Mundry <UweMundry@xxxxxxxxxxxxxxxxxxxxxxxxxxxxxx>
  - *Date:* Thu, 23 Mar 2006 16:04:01 -0800
- 

This is all fine, except I am one of the folks who absolutely needs a single EXE distribution. One of my apps can self-replicate, that is store a copy of itself to an autorunning CD, which when placed in a computer elsewhere, needs to install itself and run.

It must be my choosing, not Joe's or Microsoft's, if I choose to go this path, or if I have no choice but to go this path. The argument of increased EXE size means zero, zippo, nothing to me and my hundreds of customers, we all don't care if an exe is 2MB or 4MB.

If I link statically, then that is what I am doing. Then I will test this exe on the configurations I know the users have, and if all is well, the exe gets shipped, easy. Since this concept has now been broken with VS2005, I have been forced to go back to VS2003, where this was not an issue, and life is good there. Unless there is a solution, that means bye bye to VS 2005 for me for a while to come.

"Joseph M. Newcomer" wrote:

Generally, the idea that copy==install has been obsolete for about a decade. Perhaps longer. It was only by accident that this ever worked for you, and you got away with it. It doesn't mean that it is supposed to work (in fact, there are excellent chances that it wouldn't). The only sane way to deploy an application is to use an installer.

While you could do static linking, this leads to numerous other problems and should not be treated seriously as a solution except in the incredibly restrictive situation where there is only a single executable without any DLLs at all. For example, you can no longer use MFC Extension DLLs if you do static linking.

By the way, did you know that typical apps still use between four and two dozen system DLLs? And that you often need the latest version of these as well?

History has finally caught up with you. You are, surprise, going to be forced to do the job right. Why is running an install package is more complex than a simple copy? I've got clients with literally thousands of client app and server app installations (I know one product delivered over 5,000 installations since I delivered it), all of which are installed using standard installer packages, and no one has ever experienced any problems.

## Re: How to Deploy MFC only based application?

Note that VS.NET allows what are called "side-by-side" installations so there is no need to try to write these DLLs into the Windows\System32 directory, which is often considered a problem with older install systems. Often servers had tighter security and the installers would fail. That hasn't been true since the first VS.NET release which could use the new .msi technology.

Since each version of MFC came with a set of DLLs that were part of the distribution, and these DLLs often were improved versions of the existing DLLs. MFC42 and MSVCRT were two of perhaps a dozen DLLs that constituted a valid distribution. There were about a dozen versions of MFC42.DLL, and if you had linked to use the newer ones, the program would not load with an older version. If there had been an MFC50.DLL, then MFC60..MFC65, representing the various upgrades, people would have been conditioned out of the copy==install myth years ago.

This question keeps coming up, and there is one simple answer: use the installer. Anything that gave the illusion of working in the past was just an illusion; most people got away with it because other install packages had updated the necessary Windows components. That has not been guaranteed, ever; it just happened that way. mostly. On good days, except alternate Tuesdays when the moon was full (most of the install problems that we've had reported have been when people did a copy instead of a full install, and components were out-of-sync. Check on the descriptions of "DLL Hell". That's what installers were designed to avoid!)

joe

On Mon, 6 Feb 2006 12:19:42 +0100, "Carlo Capelli" <carlo.capelli@xxxxxxxxxx> wrote:

Hi all.

I upgraded the development of our application to Visual Studio 2005, and now

I can't deploy it to some server configuration.

The application comprises some .EXEs and .some .DLLs (MFC extension DLLs).

Up to Visual Studio 2003, installing the application on the server required only to copy the EXEs & DLLs in the same folder, with the following minimal

MFC-MSVC DLLs set (MFC71.dll, MSVCP71.dll, MSVCR71.dll).

This worked well on both Windows2000 server and Windows2003 server.

Using VS2005, this minimal set is (mfc80.dll, msvcp80.dll, msvcr80.dll). I found this coping just 1 of my EXEs on a Windows2000 server (with my MFC

extensions DLLs), then attempting to start my EXE, I got the usual 'can't find DLL etc etc'. Copying the three files listed above solved the problem.

On the customer test server (Windows2003 server) instead, I get a message like 'Incorrect system configuration: run the setup program etc etc'.

## Re: How to Deploy MFC only based application?

It is to be noted that both the Windows2000 server and the Windows2003 server have the .NET framework 1.1 installed.

I'd like to be able gain the same behaviour as VS2003 (simple copy), because of the complexity (or impossibility) to run installation packages on our customers' servers.

Anyone could help?

Joseph M. Newcomer [MVP]  
email: [newcomer@xxxxxxxxxxxxx](mailto:newcomer@xxxxxxxxxxxxx)  
Web: <http://www.flounder.com>  
MVP Tips: [http://www.flounder.com/mvp\\_tips.htm](http://www.flounder.com/mvp_tips.htm)