Subject: On Windows: Creating VERSIONINFO for your binaries...
Posted by tvanriper on Sun, 16 Sep 2007 15:18:28 GMT
View Forum Message <> Reply to Message

I noticed no topic regarding this subject in the forum, and figured some folks might want some information about it here, rather than having to hunt around for it.

I'll describe a fairly simple way to provide version information for your Windows application, such that one can right-click on the executable in Windows Explorer, and click 'Properties' to see what version of the product they're running. Note that if you intend to use MSI (Microsoft Installer) to install your application, you will need this if you want to properly support patching your product.

The linker links your file version information from resources. This means you need to create a resource file bearing your file version information. Note that you also need to do this to have your application show an icon for itself in Windows Explorer. To include a resource file in your project in TheIDE, just tap Ctrl-I (or right-click in the files area and select 'Insert any file(s)...'), then type in any filename ending with ".rc". TheIDE is smart enough to know that this is a resource file, and will run your resource compiler for this file.

The version information block should look much like this:

```
#include <windows.h>

VS_VERSION_INFO VERSIONINFO
FILEVERSION 1, 0, 0, 1
PRODUCTVERSION 1, 0, 0, 1
#ifdef DEBUG
FILEFLAGSMASK VS_FF_DEBUG | VS_FF_PRERELEASE
#else
FILEFLAGSMASK 0
#endif
FILEOS VOS__WINDOWS32
FILETYPE VFT_APP
FILESUBTYPE VFT2_UNKNOWN
BEGIN
    BLOCK "StringFileInfo"
    BEGIN
        BLOCK "040904b0"
        BEGIN
            VALUE "Comments", "My Super Amazing Application\0"
            VALUE "CompanyName", "Cheese Olfactory Workshop\0"
            VALUE "FileDescription", "Provides a /dev/null device for Windows.\0"
            VALUE "FileVersion", "1.00.00.01\0"
            VALUE "InternalName", "DEVNULL\0"
            VALUE "LegalCopyright", "Copyright (C) 1967-2007, Cheese Olfactory Workshop, All
rights reserved\0"
            VALUE "OriginalFilename", "devnull.exe\0"
            VALUE "ProductName", "DevNull\0"
            VALUE "ProductVersion", "1.00.00.01\0"
        END
    END
    BLOCK "VarFileInfo"
    BEGIN
        VALUE "Translation", 0x0409, 0x04B0
    END
END
```

Notice the trailing \0 characters... those are important. Without them, you won't have a proper information block, and while the numeric fields seem to be in place, nobody can see the string information (they'll be dropped, I think).

Also notice that I've included Windows.h. This allows you to use the VS_FF_DEBUG, VS_FF_PRERELEASE, VOS__WINDOWS32, VFT_APP, and VFT2_UNKNOWN 'keywords'. If you do not want to include windows.h, you'll have to figure out these values for yourself.

If you're making a DLL instead of an EXE, you want VFT_APP to be changed to VFT_DLL.

If your application is intended to run on specific OSes (not just any WIN32 machine), you might want to change VOS__WINDOWS32 to something more specific. You can choose from the following:

VOS_UNKNOWN The operating system for which the file was designed is unknown.
VOS_DOS File was designed for MS-DOS.
VOS_NT File was designed for Windows Server 2003 family, Windows XP, Windows 2000, or Windows NT.
VOS__WINDOWS16 File was designed for 16-bit Windows.
VOS__WINDOWS32 File was designed for 32-bit Windows.
VOS_DOS_WINDOWS16 File was designed for 16-bit Windows running with MS-DOS.
VOS_DOS_WINDOWS32 File was designed for 32-bit Windows running with MS-DOS.
VOS_NT_WINDOWS32 File was designed for Windows Server 2003 family, Windows XP, Windows 2000, or Windows NT.

You can find more detailed information on this subject at Microsoft's web site. At the moment, this URL seems to present the topic nicely:

http://msdn2.microsoft.com/en-us/library/aa381058.aspx

Subject: Re: On Windows: Creating VERSIONINFO for your binaries...
Posted by mirek on Wed, 19 Sep 2007 19:11:49 GMT
View Forum Message <> Reply to Message

Actually, I think we should extend the application templates to generate this AND also add a reference to .iml icon to .rc (and place export flag and single icon to .iml).

Maybe on option(s).

Mirek

Subject: Re: On Windows: Creating VERSIONINFO for your binaries...
Posted by tvanriper on Wed, 19 Sep 2007 20:19:20 GMT
View Forum Message <> Reply to Message

If you search on this topic through Google, you'll find that a lot of people would like to auto-increment the file version information (distinct from the product version information) based on something that ensures each updated version of the file is of a 'greater' version than the previous one. Developers often forget to increment the file version information on release, leading to problems when deploying the application (something I've experienced personally).

Microsoft uses the current date/time to help generate that number. I think that's a decent approach. Perhaps a combination of the product version and the date/time, somehow.

I think, if you did this in a somewhat automated fashion, you'd make an awful lot of developers out there happy.

I'd also be concerned about having to remember other twiddly bits... getting the FILEOS or FILETYPE/FILESUBTYPE right, etc.

On the other hand, I can see where some folks might not want to have this automated.

I'm not one of them, though. Heh.

Subject: Re: On Windows: Creating VERSIONINFO for your binaries...
Posted by mirek on Thu, 20 Sep 2007 12:37:35 GMT
View Forum Message <> Reply to Message

Well, yes. I think that the main trouble is when you are using this in collaboration with concurrent versioning system -> then there can be two equal version numbers for two different releases.

Except this, it would be really trivial to implement version number increment e.g. for each release build. Or perhaps for each release full SHA-1 hash But concurrent access makes this impossible (IMO).

Subject: Re: On Windows: Creating VERSIONINFO for your binaries...
Posted by tvanriper on Thu, 20 Sep 2007 13:51:45 GMT

View Forum Message <> Reply to Message

You may notice that there are two different version numbers.

The product version differentiates between products.

The file version differentiates between files.

When people talk about auto-incrementing a version number, it's the file version they're generally talking about. And it's essential that this version increments (and never decrements), as the Microsoft Installer requires this in order to work properly (if it ever decrements, the decremented file will not install over the previous version).

The product version doesn't matter, with regards to the installer.

So, when you mention how this collaborates with a concurrent versioning system, I'm unsure if you mean product version or file version.

Some folks may not use a concurrent versioning system, so you probably don't want to make that a requirement (but perhaps an option). A handy alternative is the date. Or manual.

I think the most trivial way to implement version number increment (for product and/or file versions) may involve integrating the versioning mechanism with Esc, then exposing certain base choices in Esc. Add a dash of GUI on top of all this, and it might not be so bad for the user.

Subject: Re: On Windows: Creating VERSIONINFO for your binaries...
Posted by nixnixnix on Tue, 04 Mar 2008 16:18:20 GMT

View Forum Message <> Reply to Message

An option to have automatic build numbering and to autogenerate the VERSIONINFO (once it has been setup) would be great.

Mirek, I wasn't sure from the above whether you will implement this in the near future. If so, will you post in this thread?

Nick

p.s. I don't see mention on MSDN of how to set the application icon.

Subject: Re: On Windows: Creating VERSIONINFO for your binaries...
Posted by michael on Tue, 04 Mar 2008 19:18:08 GMT

View Forum Message <> Reply to Message

Has anybody ever tried to use the get the versioninfo on vista?

I compiled an app using versioninfo and the versioninfo from the exe-file looks good on windows xp. But on vista there are only a few informations left like filetype and fileversion.

whats the reason why vista dosn't display all informations like xp does?

Subject: Re: On Windows: Creating VERSIONINFO for your binaries...
Posted by tvanriper on Fri, 14 Mar 2008 19:43:10 GMT
View Forum Message <> Reply to Message

I've seen it show up on Vista without issue.

Right-click on application in Windows Explorer, select 'Properties', click on 'Details' tab, and I can see the the following:

File description
Type
File version
Product name
Product version
Copyright
Size
Date modified
Language

I develop on a Vista system (despite my dislike for the OS).

Subject: Re: On Windows: Creating VERSIONINFO for your binaries...
Posted by deep on Sat, 09 Mar 2013 16:12:39 GMT
View Forum Message <> Reply to Message

Hi,

I am bumping this rather old message.

When I try to add icon via .rc file or version information as first post of this thread I get error in linking.

Linking...
LINK : fatal error
LNK1123: failure during conversion to COFF: file invalid or corrupt

I checked this couple of times with "clean/rebuild".

I get same linking error if I follow this for Application Icon
 http://www.ultimatepp.org/forum/index.php?t=msg&goto=206 11&

Win7, MSC10, UPP-5800

This application works well in debug, optimal, speed mode compilation without .rc file.

Same code compiles and links fully with MINGW using .rc file and shows application icon in file browser as well as application running. It also shows version information in file properties.

MINGW Debug mode compiled application runs ok
MINGW optimal release mode links ok but crashes on run(with or without .rc file)

```
Problem signature:
  Problem Event Name: APPCRASH
  Application Name: GeoFun.exe
  Application Version: 1.0.0.1
  Application Timestamp: 513b5ace
  Fault Module Name: GeoFun.exe
  Fault Module Version: 1.0.0.1
  Fault Module Timestamp: 513b5ace
  Exception Code: c0000005
  Exception Offset: 0015ab49
  OS Version: 6.1.7601.2.1.0.256.1
  Locale ID: 2057
  Additional Information 1: 0a9e
  Additional Information 2: 0a9e372d3b4ad19135b953a78882e789
  Additional Information 3: 0a9e
  Additional Information 4: 0a9e372d3b4ad19135b953a78882e789
```

Subject: Re: On Windows: Creating VERSIONINFO for your binaries...
Posted by deep on Sun, 10 Mar 2013 16:46:17 GMT
View Forum Message <> Reply to Message

Hi,

I checked it with example/clock also

Linking error is same with MSC10. (upp 4193 and 5800). Problem may be with MSC10 linker.

clock example compile well with MINGW debug and optimal mode. Runs ok with both options without crash . clock.rc file used with version info and icon.

Subject: Re: On Windows: Creating VERSIONINFO for your binaries...
Posted by deep on Sat, 16 Mar 2013 11:20:38 GMT
View Forum Message <> Reply to Message

Hi,

This issue is resolved now.

The problem was with .Net version and compiler tools.
Followed this.

It happened because of VS2012 install and removal.

1. Installed .Net version 4.
2. Re-installed Windows SDK 7.1
3. Installed VS 2010 Compiler update

http://stackoverflow.com/questions/10888391/link-fatal-error-lnk1123-failure-during-conversion-to-coff-file-invalid-or-c

http://www.microsoft.com/en-us/download/details.aspx?id=4422