

# ActiveForms

by Bob Swart

ActiveForms are a quick way to create a browser-based application and still be able to use all of Delphi's visual components and design-time support. If you have an existing Win32 GUI application and want to move to the web with it, then ActiveForms are certainly an option to consider (although you must be aware of the downsides: it only runs natively inside Internet Explorer, and really requires Win32 as the client operating system). And apart from the code signing (security!) issue, which can take some time to setup, ActiveForms are really a quick solution to put a first version of your app in the browser. Until you want to deploy an updated version, that is, at which time you may find that the client (running Internet Explorer) does not always automatically use the new version of the ActiveForm. In fact, the current way Delphi generates ActiveForm deployment files and Internet Explorer "interact", it would be a miracle if your real-world ActiveForm was automatically updated even once at the client side. Fortunately, I've found an easy work-around, but let's take a closer look at the cause of this problem first..

Before we start looking at the problem, I assume you've all started an ActiveForm with the "include version information" enabled, as well as all other "auto-increment build number" and "auto-increment release number" options inside the Project Options and Web Deployment options dialogs. If you don't include version information with your ActiveForm, then you don't even need to read the remainder of this little article, as there will be no way whatsoever that Internet Explorer on the client will be able to determine that a new version of the ActiveForm exists.

Which means clients need to stick with the original version (unless you can instruct them to either manually delete either the old one, or download and install the new one - yeah, right, and maybe Borland will buy Microsoft to fix these mis-communication problems next time).

Anyway, when deploying an ActiveForm, Delphi (and C++Builder) generates a number of files. If you do not use .CAB file compression and also do not deploy any packages or additional files, then Delphi generates a single .HTM file and a single .OCX file (both with filenames equal to your projectname). The .HTM file will contain a section that identifies the ActiveForm to be loaded, which also includes the version number, as follows:

```
<OBJECT
classid="clsid:42424242-4242-4242-4242-
          424242424242"
codebase=" ./DrBob42X.ocx#version=1,0,0,0"
width=640
height=480
align=center
hspace=0
vspace=0
>
```

Now, if you either specify that you want to use .CAB file compression (always a good idea) or you need to deploy packages or additional files, then Delphi will generate an .HTM file that will not directly specify the ActiveForm with a version number, but rather an .INF file (without a version number - that's the problem). This .INF file will, in its turn, contain the specifications of the ActiveForm and all other deployed files, including detailed version information, for example as follows:

```
[Add.Code]
DrBob42X.ocx=DrBob42X.ocx
VCL50.bpl=VCL50.bpl

[DrBob42X.ocx]
file= ./DrBob42X.cab
clsid={42424242-4242-4242-4242-424242424242}
RegisterServer=yes
FileVersion=1,0,4,2

[VCL50.bpl]
file= ./VCL50.cab
FileVersion=5,0,6,18
DestDir=11
```

The problem when using an .INF file (which you get - again - even if only using .CAB file compression), is that Internet Explorer only looks at the version information inside the .HTM file in order to determine whether or not to download a new set of files from the server. Even if we supply a new .INF file with higher version numbers, Internet Explorer will not check these version numbers and download a new ActiveForm. This has been a problem for a long while now, and a potential showstopper for real-world applications using ActiveForms.

This is how I work around it: just specify the version number inside the HTM file again. When using an .INF file, the .HTM file will originally contain the following snippet:

```
<OBJECT
classid="clsid:42424242-4242-4242-4242-
          424242424242"
codebase=" ./DrBob42X.inf "
width=640
height=480
align=center
hspace=0
vspace=0
>
```

As soon as you enter a version number to this ./DrBob42.inf file, then Internet Explorer will be triggered to compare it to the local version, and download a new ActiveForm (and, if needed, packages and additionally deployed files as well). In short, the "working" OBJECT code snippet would look as follows (for the same 1.0.4.2. version number inside the DrBob42.inf file):

```
<OBJECT
classid="{clsid:42424242-4242-4242-4242-
          424242424242}"
codebase=" ./DrBob42X.inf#version=1,0,4,2"
width=640
height=480
align=center
hspace=0
vspace=0
>
```

The only downside is that you need to manually adjust the .HTM file every time you deploy your ActiveForm. But that's not really a problem, since I doubt you'd want to use the .HTM file generated by Delphi (or C++Builder) in a real-world situation anyway. It only means you need to look in the .INF file to find out the actual version number (which, in my case, increases after every build and every deployment) and modify the .HTM file to reflect that new version. And that will make sure all my clients will get the right new version of the ActiveForm next time they load Internet Explorer, as they should.

One final word of caution: if you've selected the "auto-increment build number" and "auto-increment release number" options, then you should know that the release

number will be auto-incremented right after you've deployed the ActiveForm. In other words: right after you've deployed it for the first time, the ActiveForm version on disk (and on the server) will be 1,0,0,0.

However, the file version information in your current project will then show 1,0,1,0 - i.e. the release number (third number of the series) in the IDE will always be one higher than the release number of the actually deployed ActiveForm. This is not so with build numbers (fourth number of the series) - these will always be in sync. Remember this when your client calls and specifies the version number (and it might also be a good idea to have the ActiveForm display the version number somewhere in a lower-left corner, for example)...



---

*Bob Swart (aka Dr.Bob - [www.drbob42.com](http://www.drbob42.com)) is an @-Consultant, Delphi "Clinic" Trainer and co-founder of the "Delphi OplossingsCentrum" of TAS Advanced Technologies ([www.tas-at.com](http://www.tas-at.com)) in Best, The Netherlands. Bob is co-author of the *Revolutionary Guide to Delphi 2*, *Delphi 4 Unleashed* and *C++Builder 4 Unleashed*, and technical author for a number of computer magazines, including *The Delphi Magazine*, *Delphi Developer*, *UK-BUG Newsletter* and *SDGN Magazine*.*