

A Crash Course In Wine

Unless you have never ever been involved with Windows, you will probably have one or two applications that need Windows to run. In my case that includes in particular BibleWorks, a Van Dale dictionary and a Latin dictionary, but also two nostalgic tools to extract icons from executable files and define colours on your display. Almost daily I use one or two of them and they need to be integrated into my Linux system. A dual boot configuration with Windows would not work.

There are two options: run Windows as a guest operating system from your Linux platform (which is called virtualisation, e.g. with Xen, KVM or VMware), or run the applications directly under Linux using a tool to make them 'talk' with Linux (usually called emulation, e.g. with Wine). Since Microsoft is not really interested in these cunning plans, it is always a bit of trial and error to find what works. Currently I am trying to get virtualisation alive, but for the time being Wine is the way. It's not perfect, but it does the job good enough for my purposes.

But Wine too has its trials and tribulations. Here is a crash course in how to get it running properly. The Linux environment is openSUSE 11.0 and KDE 3.5, but it should not be very different on other distros or desktop managers.

1. *Install Wine*

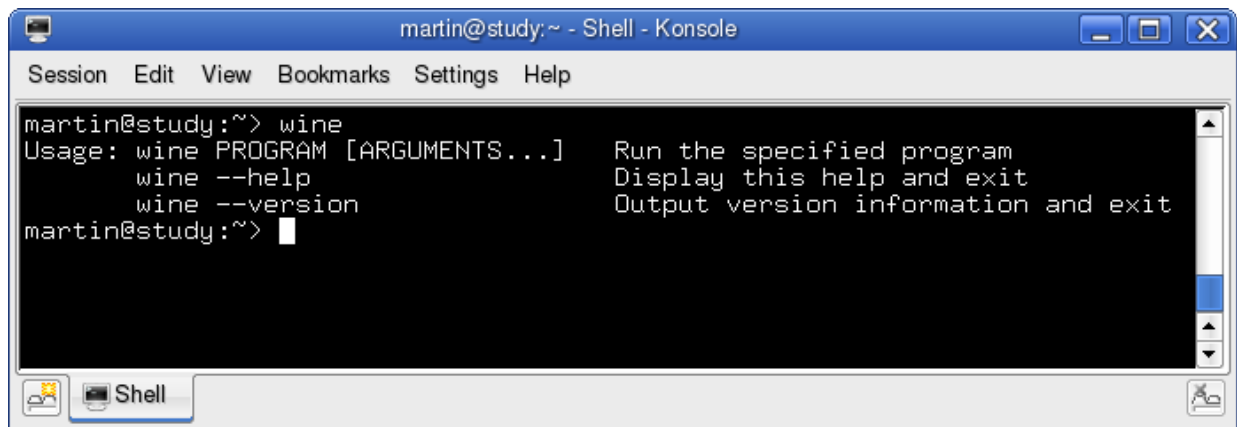
I only install the Wine package that comes with openSUSE, nothing extra (like wine-doors), nor do I use the Wine CVS builds repository. Just to keep on the safe side. The current Wine version is 1.1.21. Of course, I do keep it updated.

2. *Configure Wine*

After installation, open a terminal (aka as console) and type:

```
$ wine
```

Now wine will create a `/.wine` directory in your Linux home directory. In case you don't already know: the dot makes it invisible in your file manager, unless you select 'Show hidden files'.

A screenshot of a terminal window titled "martin@study:~ - Shell - Konsole". The terminal shows the command "wine" being entered, followed by its usage information: "Usage: wine PROGRAM [ARGUMENTS...] Run the specified program", "wine --help Display this help and exit", and "wine --version Output version information and exit". The prompt "martin@study:~>" is visible at the end of the line.

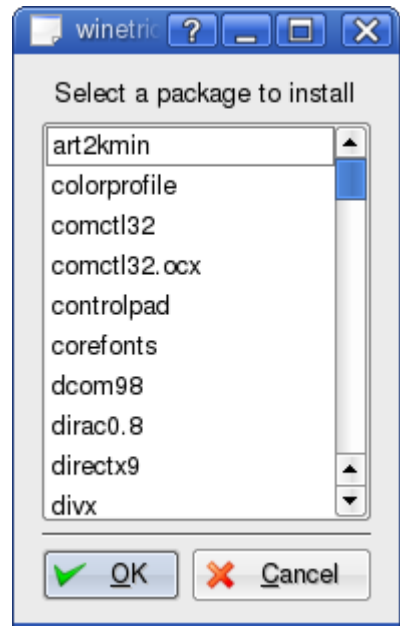
```
martin@study:~> wine
Usage: wine PROGRAM [ARGUMENTS...]  Run the specified program
      wine --help                    Display this help and exit
      wine --version                 Output version information and exit
martin@study:~> █
```

Next it is wise to install a few extras. Yes, it is a bit like the real Windows: after installation you need to add some more before you can start working. In a terminal type:

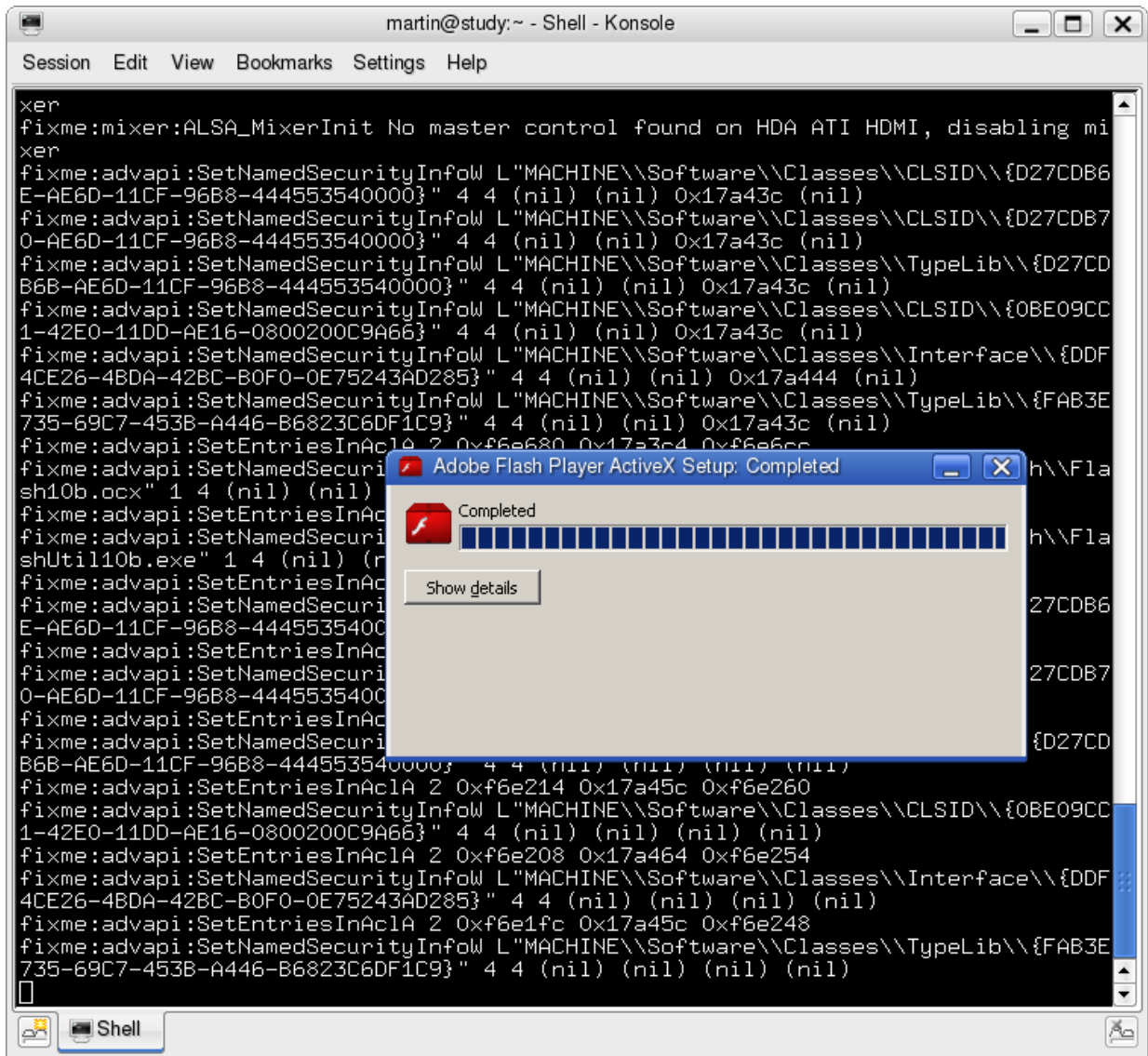
```
$ winetricks
```

A nifty tool will start which allows you to easily install additional components. Be sparsely in adding! You can select all you want using Ctrl and the mouse, but it seems to be safer to add one package at a time and then restart winetricks. Recommended additions:

- allfonts [gives you the default Microsoft fonts]
- gecko
- ie6 [quite some applications need it]
- flash
- quicktime72
- vb[3/4/5/6]run [can all be installed simultaneously; some applications need it]
- allcodecs [if you want sound & video enabled; make your own choice of codecs]
- winxp [that's just a script, not a package]

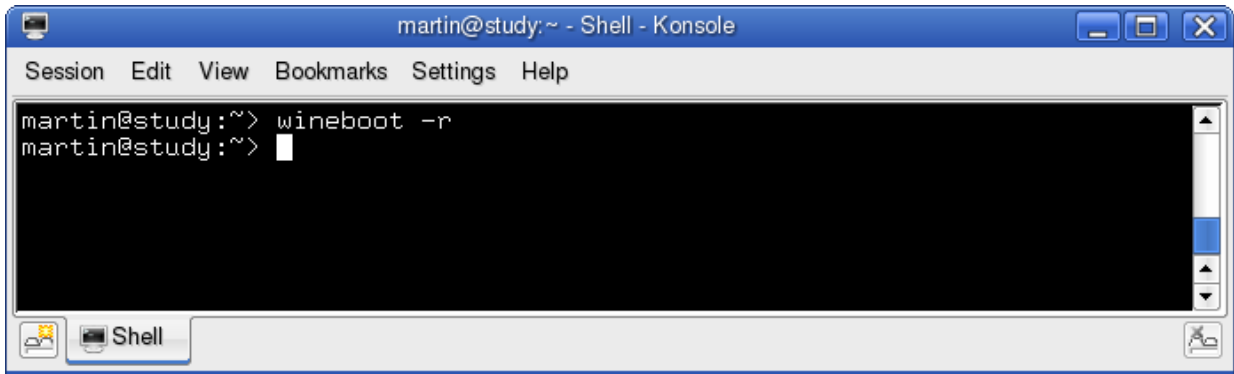


In the terminal packages will automatically be downloaded and installed. True Linux stuff! Keep an eye on the screen: since it is Windows, dialog boxes & questions may pop up.



Probably you will be asked to restart your computer at some point. No need for that, but after you have finished installing all you want, just type:

```
$ wineboot -r
```



3. Tweak Wine

After installation and configuration you can tweak a few things by typing in a terminal:

```
$ winecfg
```

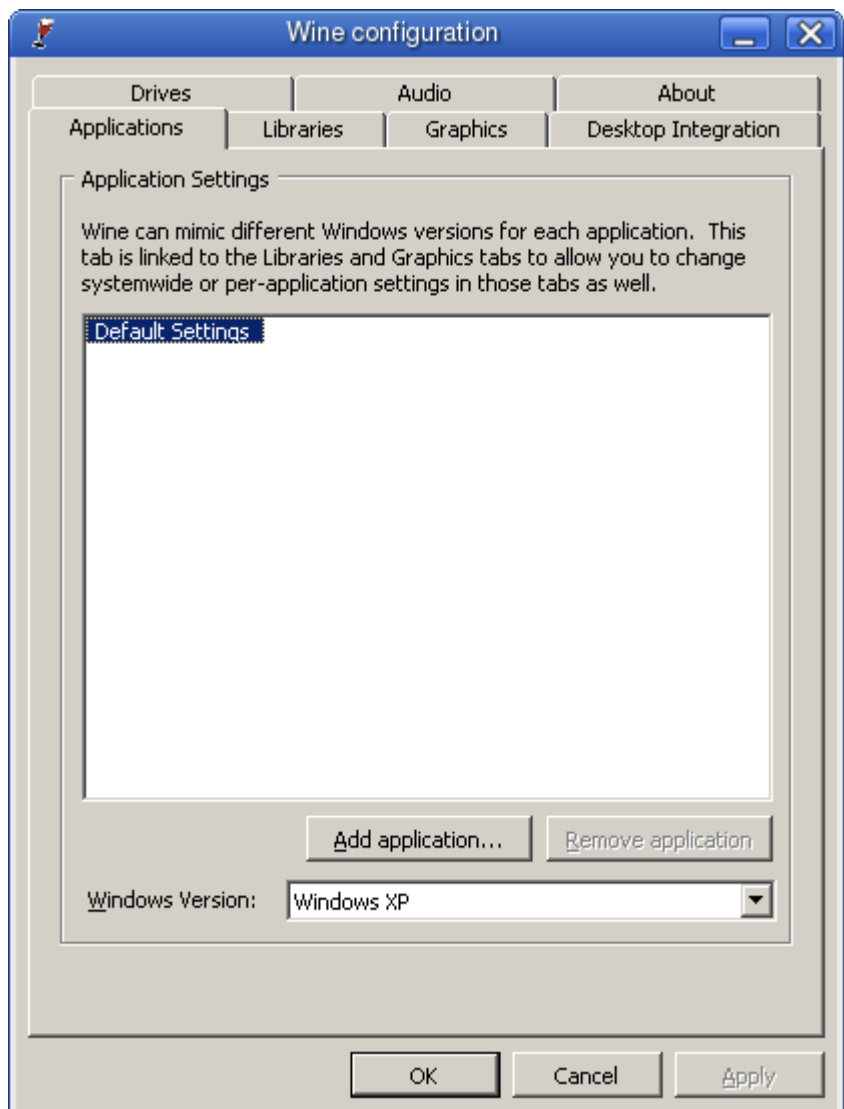
Another nifty tool starts. However, this is still work in progress and you will get a 'Not Implemented Yet' message on a number of options. I would suggest you tweak two things:

- + on the 'Desktop Integration' tab link the default 'Windows folders' to the actual directories in your Linux home directory;
- + on the 'Audio' tab select a sound driver (ALSA is usually the best choice).

4. Install your applications

Now you can install your Windows applications.

Make sure *.exe files (aka Windows executables) are associated with Wine. In KDE open Control Centre, go to KDE Components and then to File Associations. Type 'exe' in the search box, expand the 'applications' tree and select 'x-dosexec'. If no application is selected, click 'Add',



just type 'wine' and click 'OK'. Finally click 'Apply' to save these changes.

For the rest it's a piece of fruitcake. Insert the CDROM, open in a New Window and double-click the Setup.exe (or something like it). Just like Windows – sorry for that. However, you will probably come across a few glitches. In the case of my Latin dictionary, for example, some things did not work. It runs on the basis of HTML files, which opened fine with the help of Internet Explorer, but through lack of an Acrobat Reader plug-in the PDF help files would not open. Well, that was just a case of adding KDE menu entries to start them separately in Kpdf (or Xpdf, or Acoread). But for BibleWorks and Van Dale Woordenboek everything runs fine out of the box. And yes, I payed for all of this proprietary software.

That's it. Now you can use your useful Windows applications as in the old days. Until a better alternative becomes available, of course...