

COMP231: INSTALLING PCSPIM, XSPIM, AND SPIM

1 Windows 9x/NT Installation - PCspim

Insert the companion CD that came with the text. If the CD doesn't automatically launch a web browser, goto *My Computer* and double click the CD drive. Click on *Software* in the left navigation frame and then select *Spim, PCSpim and xspim* from the right top links. You will see a link for the Windows software, click it. Save the .zip file somewhere on your machine (for example, the desktop) then extract it with an unzip utility, such as WinRAR. Browse to the resultant folder. Click on `Setup.exe` to install the simulator. The default location is `C:\Program Files\PCSpim`. When setup finishes, there should be a link to the executable in your Start menu. You can of course always browse to the folder and make additional shortcuts for your desktop or quicklaunch toolbar yourself.

2 Linux Installation

2.1 Debian GNU/Linux and Debian based distributions:

You should use a precompiled .deb package. This will have both the command line `spim` and X11 `xspim` executables as well as the necessary configuration files and some brief documentation. Debian does not consider `spim`'s license to be free since it requires Prof. Larus' name stay on the source and his approval for commercial use, so in Debian (Sarge, Etch) you will likely need to add a `non-free` option to your preferred Debian mirror(s) in your `/etc/apt/sources.list` file as a default Debian installation will not be able to see this package. For example you could add this whole line:

```
deb http://mirrors.kernel.org/debian/ etch main contrib non-free
```

Once you know you have a proper repository in `sources.list`, run:

```
apt-get update
apt-get install spim
```

On Debian based distributions (i.e. Kubuntu, Xandros, etc.) you should check your distribution's repository first. For example, Kubuntu has its own `spim` package in the Ubuntu `multiverse` which is the equivalent of Debian's `non-free` section. If a search on your distribution's package manager fails to bring up a version of `spim`, the next best thing to do is install the Debian package since `spim` and `xspim` have only basic dependencies. You can manually download the Debian package from here:

```
http://packages.debian.org/testing/electronics/spim
```

and install the resulting with:

```
dpkg -i spim_7.3-1_i386.deb
```

Note if you're on ppc you need to install the older version (spim 6.5) at:

```
http://packages.debian.org/stable/electronics/spim
```

You might also want to try this if the newer package doesn't work for some reason.

2.2 Fedora Linux and Red Hat based distributions:

You should use a precompiled `.rpm` package. It appears that no major distribution (Fedora, CentOS, RHEL, SUSE, etc.) includes spim in its default repositories, hence all packages come from 3rd parties. For newer distributions you should use:

```
http://www.cs.wisc.edu/cbi/downloads/rpm/fedora-4-i386/RPMS.apps/spim-7.2.1-1.sam.1.i386.rpm
```

Once downloaded, you can install with:

```
rpm -i spim-7.2.1-1.sam.1.i386.rpm
```

and if this fails, you can try this older package (spim 6.5):

```
ftp://fr2.rpmfind.net/linux/falsehope/home/rathann/apt/7.3/RPMS.stable/spim-6.5-1.i386.rpm
```

2.3 Gentoo Linux, Gentoo based distributions, and BSD:

Gentoo has spim and xspim source in its portage repositories. You should be able to install and compile both with an:

```
emerge spim
```

Depending upon which BSD flavor you're using, you may be able to do the same thing.

3 Compiling from source (Linux, BSD, Mac OSX, etc.)

If your OS doesn't have a precompiled package, or none of the precompiled packages worked on your system, you are basically left with no other option than to attempt to compile from source. First, **do not** use the code provided with the book on your first try. You are much more likely to have success with the updated code from Prof. Larus' web page (spim-7.3) at <http://www.cs.wisc.edu/~larus/SPIM/spim.tar.gz> as a number of the bugs that hindered compilation have been addressed. To compile this version of spim you need the following tools:

- gcc (3.3 or higher, 3.4, 4.0, and 4.1 all seem to work)
- flex (version is extremely important, more on this later)
- bison

- `xmkmf` (imake support)
- `xlibs` (and development files)
- `libxaw` (Athena widgets for X11 and development files, version 6 or 7 work)

You need to make sure these are installed these for your particular distribution. Luckily, these tools are rather common and with the exception of `flex` the version used doesn't seem to matter much. To compile, first extract the code from the `.tar` archive and go into the resulting `/xspim` subdirectory:

```
tar -xvzf spim.tar.gz
cd spim-7.3/xspim
```

Then adjust the `Imakefile` to make sure you have proper installation paths for your distribution. The defaults work fine on most systems, it puts the executable in `/usr/local/bin` and the `exception.s` file in `/usr/local/lib`. However, your might want to put these files elsewhere. Run the autoconfiguration tools to generate a customized `Makefile` for your system with correct `Xlibs` locations. Then use the resulting `Makefile` to compile:

```
./Configure
xmkmf
make
make install
```

If you want to install the command line `spim` also, you must compile it separately:

```
cd ../spim
make
make install
```

If all went well, you now should be able to launch `xspim` and `spim` as a normal user. You may need to open a new terminal window or `rehash` to do so. Generally if things didn't compile, you should be able to track down the problem. Compiler complaints that `.h` files are missing, or that `-lXaw` and/or `-lXmu` are not valid flags indicate that not all your X11 development libraries are installed or being seen. Not having `bison` installed will prevent the compile from starting. If `xmkmf` is not available you can try using the standard `Makefile`, but this may fail if your distribution puts your X11 libraries in odd locations. However, you can search and try to change these in the `Makefile` manually. However, if `xspim` compiles and launches into a semiworking state, you have an issue with `flex`. Examples of such a situation is if you get a window, but no buttons, or a window and buttons but no registers. The most amusing is an output of an X11 font string on the command line when you type `xspim`. A known working version for `spim 7.3` is `flex 2.5.31`. The later `flex 2.5.1.33` does not work. You may need to downgrade `flex` to compile. Since `flex` is only needed to compile, and not needed to run `xspim` or `spim`, you can safely restore your distributions default version afterwards. If your distribution only supports earlier versions (i.e. `< 2.5.31`), you might want to try the code included with the book if the newer code fails to produce a working `xspim`. If all else fails, `spim` will always compile correctly, `flex` is used only for `xspim`.

<p>Note: <code>xspim</code> is known to not work on <code>solaris</code>.</p>
--