

Boo on Ubuntu

NOTE: This information is very much out of date. Ubuntu now has packages for everything you need - the "boo" package takes care of core installation, and the "monodevelop-boo" package will allow you to develop boo files on mono. Note, however, that the version of boo included in these packages is usually a bit behind. The latest-and-greatest features of boo are currently only available through SVN (see the bottom of this document).

Notes on installing everything you need to use boo, mono and Monodevelop on Ubuntu.

These instructions install the very latest bleeding edge versions (from SVN) of mono and monodevelop. This is very time consuming. You might prefer to install mono and gtk# binaries instead, and wait until the next version of MonoDevelop is released that includes boo support.

I found [these instructions](#) in French half way through writing this, so try that if you like.

See also [these notes](#) on compiling mono from svn.

Installing Support Applications

From the Applications menu, choose System tools -> Terminal

We will use apt-get to install some supporting applications. If you aren't using a Debian-based version of Linux, try finding rpm versions of this software instead. In Ubuntu, if you get any errors that a package cannot be found using apt-get, enable the "universe" repository by editing /etc/apt/sources.list.

Type

```
sudo apt-get install subversion autoconf automake1.9 libtool
libgtksourceview-dev intltool libcairo1-dev libpng12-dev libjpeg62-dev
libtiff4-dev libungif4-dev bison g++-3.3
```

Enter your password when prompted, and the software will be downloaded and installed. Answer 'y' for any questions. You may have to re-insert your Ubuntu CD, too.

Also install an old version of mono to use for compiling the new version later:

```
sudo apt-get install mono
```

Type "mono -V" (no quotes) at the command line to test mono works, and note the version number (1.0.5, at the time of this writing).

NOTE If you are not planning on installing Mono from SVN later, there is a way to get debian packages for a more recent version of Mono (1.1.6 and higher). See [these notes](#) for info on getting backported mono packages.

Download source code from SVN

```
cd      #to go to your home directory
mkdir svn

#now checkout the sources we need: (will take a while to download)

svn co svn://svn.myrealbox.com/source/trunk/monodevelop

#this will take some time, especially mcs
svn co svn://svn.myrealbox.com/source/trunk/ libgdiplus gtk-sharp
gtksourceview-sharp gtkmozembed-sharp monodoc mono mcs

#type
#svn list svn://svn.myrealbox.com/source/trunk
#to see a list of other available software, such as
#the Tao OpenGL wrapper or the Cecil assembly IL tool
```

Compile and Install Mono

Note You may want to replace the /usr/local prefix with just /usr in all the autogen commands on this page.

```
cd libgdiplus
./autogen.sh --prefix=/usr/local
make
sudo make install

cd ../mono
./autogen.sh --prefix=/usr/local --with-preview=yes
#the "with-preview" enables C# 2.0 features like generics
make
sudo make install
```

Type "/usr/local/bin/mono -V" to see that mono is working and the version number is higher than before.

Remove the old mono

```
sudo dpkg --purge mono mono-mcs
```

Add mono to your path:

Type:

```
gedit ~/.bashrc
```

And add this line to the end, then save and close:

```
PATH=/usr/local/bin:$PATH
LD_LIBRARY_PATH=/usr/local/lib/:$LD_LIBRARY_PATH
PKG_CONFIG_PATH=/usr/local/lib/pkgconfig:$PKG_CONFIG_PATH
```

We need to set the LD_LIBRARY_PATH permanently:

```
sudo gedit /etc/ld.so.conf

#And add this line:
/usr/local/lib

#then type
sudo ldconfig
```

Now when you use terminal again, you can just type mono to get the new version instead of having to type /usr/local/bin/mono.

If you have problems along the way try the #mono irc channel.

Compile and install GTK# and other mono libs

Now compile and install other mono software we need for MonoDevelop:

GTK#:

```
sudo apt-get install libpango1.0-dev libgtk2.0-dev libxml2-dev
libart-2.0-dev libgnomecanvas2-dev libgnomeui-dev libgnomeprintui2.2-dev
libgda2-dev librsvg2-dev libgtkhtml3.2-dev libvte-dev libgnomedb2-dev
libpanel-applet2-dev

cd ../gtk-sharp
./bootstrap --prefix=/usr/local
make
sudo make install
```

Check that the new gtk-sharp assembly was installed to the mono GAC:

```
gacutil -l gtk-sharp
```

install monodoc:

```
cd ../monodoc
./autogen.sh --prefix=/usr/local
make
sudo make install
```

install gtksourceview-sharp:

```
cd ../gtksourceview-sharp
./autogen.sh --prefix=/usr/local
make
sudo make install
```

install gtkmozembed-sharp:

```
cd ../gtkmozembed-sharp
./autogen.sh --prefix=/usr/local
make
sudo make install
```

Install Boo

Here are instructions for either installing the latest boo binary release (easier), or compiling boo from svn yourself.

Installing boo binary release:

Installing boo binary release: Change the boo version to most recent available at the [Download](#) site.

```
wget http://dist.codehaus.org/boo/distributions/boo-0.5.3.1544.tar.gz
tar xzvf boo-0.5.3.1544.tar.gz
cd boo-0.5.3.1544
./configure --prefix=/usr/local
make
sudo make install
```

Installing boo from svn

See the [Source](#) and [How To Build](#) pages for more info.

Requires [nant](#) and also [java](#) or [ikvm](#). I used ikvm so that installing java wasn't necessary.

Installing nant and ikvm. cd into your svn folder and type (check the ikvm and nant websites for the most up to date version):

```
#ikvm
wget
http://easynews.dl.sourceforge.net/sourceforge/ikvm/ikvmbin-0.12.0.0.zip
unzip ikvmbin-0.12.0.0.zip
chmod 755 ikvm/bin/*.exe

#nant
wget http://nant.sourceforge.net/release/0.85-rc3/nant-0.85-rc3-bin.tar.gz
tar xzvf nant-0.85-rc3-bin.tar.gz
mv nant-0.85-rc3 nant
chmod 755 nant/bin/*.exe
```

Add shell script wrappers for nant and ikvm in /usr/local/bin/ (so you can just type "nant" or "ikvm" to get them to work)

Type "sudo gedit /usr/local/bin/nant" and paste in these contents (change path to your path to nant):

```
#!/bin/sh
env /usr/local/bin/mono /home/yourname/svn/nant/bin/NAnt.exe "$@"
```

Then type "sudo chmod 755 /usr/local/bin/nant".

Type "sudo gedit /usr/local/bin/ikvm" and enter:

```
#!/bin/sh
env /usr/local/bin/mono /home/yourname/svn/ikvm/bin/ikvm.exe "$@"
```

Then type "sudo chmod 755 /usr/local/bin/ikvm". (type "sudo ln -s /usr/local/bin/ikvm /usr/local/bin/java" if you don't plan on installing java)

Now we download, compile and install boo itself.

```
#from svn folder
svn co svn://svn.boo.codehaus.org/boo/scm/trunk/ boo
cd boo
```

We need to change one little thing in the build file: (unless you have java installed, or you created a "java" alias to ikvm in /usr/local/bin)

```
gedit default.build

#replace the java property line with ikvm instead of java:
<property name="java" value="ikvm" />
```

Now compile by typing:

```
nant
sudo nant install -D:install.prefix=/usr/local
```

Type "gacutil -I Boo.Lang.Compiler" to test that it installed.

booish, booi, and booc will be under /usr/local/lib/boo/
You can test the booish interpreter like so:

```
booish
```

It should work if /usr/local/bin was added to your PATH.

Compile and Install MonoDevelop

```
cd ../MonoDevelop
./autogen.sh --prefix=/usr/local --enable-boo
make
sudo make install
```

If you get errors compiling, I had to change SetDefaultSortFunc(...) to DefaultSortFunc = ... in a couple of files.

Note also, MonoDevelop sources may not be in sync with the very latest svn version of boo obviously.

Test by entering "monodevelop &" on the command line.

Updating

cd into the svn folder and type:

```
svn up boo MonoDevelop
svn up mcs mono gtk-sharp libgdiplus gtksourceview-sharp gtkmozembed-sharp

#plus whatever other svn software folders you have
```

to update the sources. You then have to repeat the steps to recompile and install the software.

Note Run "sudo nant uninstall" in the boo folder to remove the old boo dlls before recompiling and reinstalling a new version of boo.

Get Help

If you have errors, please ask for help on our [IRC Channel](#) or the #monodevelop IRC channel on irc.gimp.net.