

Step 1 – The dependencies

The Ardour website has a list of [Ardour's dependencies](#). You'll notice that there are a few libs which they have modified versions of. You don't *need* these special ones – you can get by without them. I learned yesterday that APT has very neat mechanism for installing these dependencies. A package maintainer can specify the list of dependencies required to *build* the package, not just to install it. So, to install the dependencies do the following:

```
> sudo apt-get build-dep ardour3
```

On Debian, that would actually give you all the dependencies for Ardour 4 (their Ardour3 package is actually version 4.x). On Ubuntu there are a couple more dependencies you need to install:

```
> sudo apt-get install vamp-plugin-sdk libtag1-dev libaubio-dev liblrd0-dev  
librubberband-dev
```

Now you should have all the dependencies for Ardour.

Step 2 – Get the code

Assuming you have already changed to the directory where you want to clone ardour

```
> git clone git://git.ardour.org/ardour/ardour.git
```

Alternatively you could go to their [github mirror](#) and fork that, and then clone that to your machine. If you want to submit changes doing them via github PRs is by far the easiest way.

Step 3 – Build

Next change into the ardour directory that was cloned

```
> cd ardour
```

Then we build

```
> ./waf configure  
> ./waf
```

If you are missing any dependencies then you should find out during the *waf configure* step.

Step 4 – Run

To run the version you just built

```
> cd gtk2_ardour  
> ./ardev
```

[Waf](#) also lets you do install/uninstall/clean etc.

Stop!

Do you really need to do this? We provide ready-to-run packages of Ardour. Unless you are a developer with experience compiling and building applications from source, this document is not for you. Please go back to the [download page](#)

We do not provide support for building from source. We do not make regular efforts to keep this page up to date. Please do not ask for help with this process.

Before You Start

1. make sure you have gcc/g++ installed. `gcc --version` will check this. You should ideally have version 4.3 or above.
2. You will need git installed
3. You will need Python version 2.6 or later installed
4. If you want JACK support, you must have a suitably new version of [JACK](#) installed. For JackOSX, version 0.89 or newer. For JACK1, 0.121 or newer.

Getting Required Libraries and Tools

Ardour uses a **lot** of software libraries to provide functionality needed by the program. Before you can build Ardour, you will need to make sure that your system has all of [the current dependencies installed](#). You can do this using your system's software update/install tool, or if you are comfortable compiling source code, from the source code of each dependency. Ardour developers in general do not provide assistance with this task, so please don't ask us for help.

Building GTK+ (the graphical toolkit we use) from source is a monumental task and can require a lot of other libraries to be built and installed along the way. This is particularly true on OS X, where many of GTK's own dependencies are not easily (or correctly) available.

If you use your system software update/install tool, you must have the "development" versions of all packages. The "normal" versions are there to be able to run software that uses the package, but they do not work when trying to *build* other software.

Building Ardour 5.x

From here on, we will refer to the directory where your Ardour source code is located as `$AD`. It does not matter where it is located on your system. Typically it will be a location such as `~/ardour`

If building from git, checkout Ardour

```
cd $AD
git clone git://git.ardour.org/ardour/ardour.git
cd $AD/ardour
```

OR If building from a source tarball, unpack it

```
cd $AD
tar xf /where/you/put/the/src/tarball
cd ardour-<VERSION>
```

Now, the build

```
./waf configure
./waf
```

You **do not need to install** in order to use your new build of Ardour. You can run it from within the build tree:

```
cd gtk2_ardour
./ardev
```

To install the results:

```
./waf install
```

To uninstall:

```
./waf uninstall
```

To clean up results of a build (objects, libraries, etc) use

```
./waf clean
```

Creating an Application Bundle

Ardour is distributed by ardour.org in the form of "bundles", which are nothing more than a directory tree which contain everything the app needs to run.

```
cd tools/linux_packaging
./build --public --strip some
./package --public --singlearch
```

You now have a functioning binary bundle, in the form a .tar file.