

# Systrip compilation instructions

## ***Dependencies List***

- libxml2
- zlib
- freetype
- glew
- libjpeg
- libpng
- Qt 4.6 including at least the following modules: QtCore, QtGui, QtNetwork, QtOpenGL, QtXML, QtAssistantClient (not installed by default with Qt 4.7 you can get it here : <ftp://ftp.qt.nokia.com/qt/source/qt-assistant-qassistantclient-library-compat-src-4.6.3.tar.gz> see <http://labs.qt.nokia.com/2010/06/22/qt-assistant-compat-version-available-as-extra-source-package/> if you need help to compile it)
- cmake
- cmake-gui (On various platform, this is directly provided by the cmake package)
- OpenBabel 2.2.3
- FFMPEG 0.5
- Gsoap 2.7
- libSBML 4.3
- libMYSQL ++ (mysql c++ client library)

## **Dependencies List - Ubuntu**

LibSBML is not packaged on Ubuntu so you can get it here : <http://sbml.org/Software/libSBML>.  
Deb file are available in the download section.

libqt4-dev libqtassistantclient-dev libfreetype6-dev zlib1g-dev libglew-dev libjpeg-dev libpng12-dev  
build-essential cmake doxygen libxml2-dev libopenbabel-dev gsoap libmysql++-dev

## ***Basic Installation***

These are generic installation instructions.

CMake is the main build system, and the configure scripts along with all the autoconf/automake tools are deprecated and may not work.

The simplest way to compile this package is:

1. `cd` to the directory containing the package's source code, and create a `build` directory. Enter this `build` folder, and type `cmake ..`. This will check for all of Systrip's dependencies, and warn if some are missing.  
If you want to build with debug information, or with optimisations, you need to tell CMake to do so.  
You can either use `cmake -i`, which will ask you what kind of build you want (Debug/Release), or invoke cmake like this:

```
`cmake .. -DCMAKE_BUILD_TYPE=Release` (or Debug).
```

2. Type `make` to compile the package. (if you have more than one core, use `-jX` where X is your number of cores + 1.
3. Type `make install` to install the programs and any data files and documentation.
4. You can remove the program binaries and object files from the source code directory by typing `make clean`.

There is an issue when linking with the dynamic version of OpenBabel : when closing Systrip an error can occur during the destruction of OpenBabel plug-ins. This error don't affect the program usability. To fix it link with the static version of OpenBabel see “OpenBabel compilation” for more information.

## ***Compilers and Options***

You can pass options to cmake, either by using `cmake .. -i` or `cmake-gui ..`, which will provide you with some control over the build.

You can change the build type (Debug Vs. Release) and compilation options.

If Cmake cannot find some external libraries you can set yourself the location of these libraries. The next table contains the Cmake variables to set the location of the main libraries used in systrip.

Cmake variable	Description
TULIP_DIR	Tulip installation directory.
OPENBABEL2_DIR	OpenBabel installation directory.
GSOAP_PATH	Gsoap installation directory.
LIBSBML_DIR	Libsbml installation directory.
FFMPEG_DIR	FFMPEG installation directory.
MYSQL_DIR	MYSQL installation directory.

If you need more options read CMake find modules located in the Systrip source code (systrip\_source\_directory/controller/Systrip/CMakeFiles).

## ***Installation Names***

By default, `make install` will install the package's files in `/usr/local/bin`, `/usr/local/man`, etc. You can specify an installation prefix other than `/usr/local` by giving cmake' the option `-DCMAKE\_INSTALL\_PREFIX=PATH`.

## ***Running Systrip***

To run Systrip launch the 'tulip' executable located in the 'bin' directory of the installation directory.

```
'installation_directory/bin/tulip'
```

## ***LibSBML compilation***

LibSBML uses the autotools to configure it's build. You don't need to set special build options to use it with Systrip.

The simplest way to compile this package is:

1. Go into the source directory of LIBSBML.
2. Configure the build process with the configure script : `./configure .`  
Check the build configuration log to see if there is some errors.
3. Type `make` to compile the package.
4. Type `make install` to install it.
5. You can remove the program binaries and object files from the source code directory by typing `make clean`.

If you need more help with LibSBML compilation see the it's website :  
<http://sbml.org/Software/libSBML> .

### ***OpenBabel compilation***

OpenBabel uses Cmake as build system. To avoid the error when closing Systrip just set the `BUILD_SHARED` variable to OFF.

If you're compiling on 64 bit system you will need to add `-fPIC` to ``CMAKE_CXX_FLAGS'`.

The simplest way to compile this package is:

1. Create a directory where the package will be build.
2. Configure the build process with cmake :  
``cmake buildDirectory sourceDirectory cmakeOptions -DBUILD_SHARED=OFF'`  
Check the build configuration log to see if there is some errors.
3. Go to the build directory if you're not in.
4. Type `make` to compile the package.
5. Type `make install` to install OpenBabel.
6. You can remove the program binaries and object files from the source code directory by typing ``make clean'`.

Checks the OpenBabel site if you need more help for building it : [http://openbabel.org/wiki/Main\\_Page](http://openbabel.org/wiki/Main_Page)

### ***FFMPEG compilation***

FFMPEG uses the autotools to configure it's build. There is some special options to set to use FFMPEG with Systrip : ``--enable-gpl --enable-memalign-hack --enable-swscale'`.

The simplest way to compile this package is:

Go into the source directory of FFMPEG.

- Configure the build process with the configure script : ``./configure --enable-gpl --enable-memalign-hack --enable-swscale'`. Check the build configuration log to see if there is some errors.
- Type `make` to compile the package.
- Type `make install` to install it.
- You can remove the program binaries and object files from the source code directory by typing

make clean.

If you need more help with FFmpeg compilation see the it's website : <http://ffmpeg.org/>

### ***GSOAP compilation***

You don't need to compile GSOAP to use it with Systrip. Just set the GSOAP\_PATH to the source path of GSOAP.