

## Build Instructions (Windows)

Build Jil in Windows with VC.

### Install wxWidgets

下载最新版（当前最新版是 2.9.4）：<http://wxwidgets.org/>

解压后，build\msw 子目录下有 VC++ solution（或 dsw）文件，打开，build 整个 solution（这里为方便起见，其实很多子程序库 Jil 并不需要）。

注：如果后续编译 Jil 时遇到 wxGraphicsContext 相关错误，请检查 include\wx\msw\setup.h 文件，看 wxUSE\_GRAPHICS\_CONTEXT 是不是 0。

添加环境变量（环境变量是为 CMake 作准备，下同）：

```
wxWidgets_ROOT_DIR = <Root of wxWidgets>
```

比如：

```
wxWidgets_ROOT_DIR = D:\lib\wxWidgets-2.9.4
```

### Install C++ Boost Library

下载最新版：<http://www.boost.org/>

目前 Jil 用到的 boost 程序库，都是 header-only 的，不需要编译。

直接解压后，添加环境变量：

```
Boost_DIR = <Root of Boost>
```

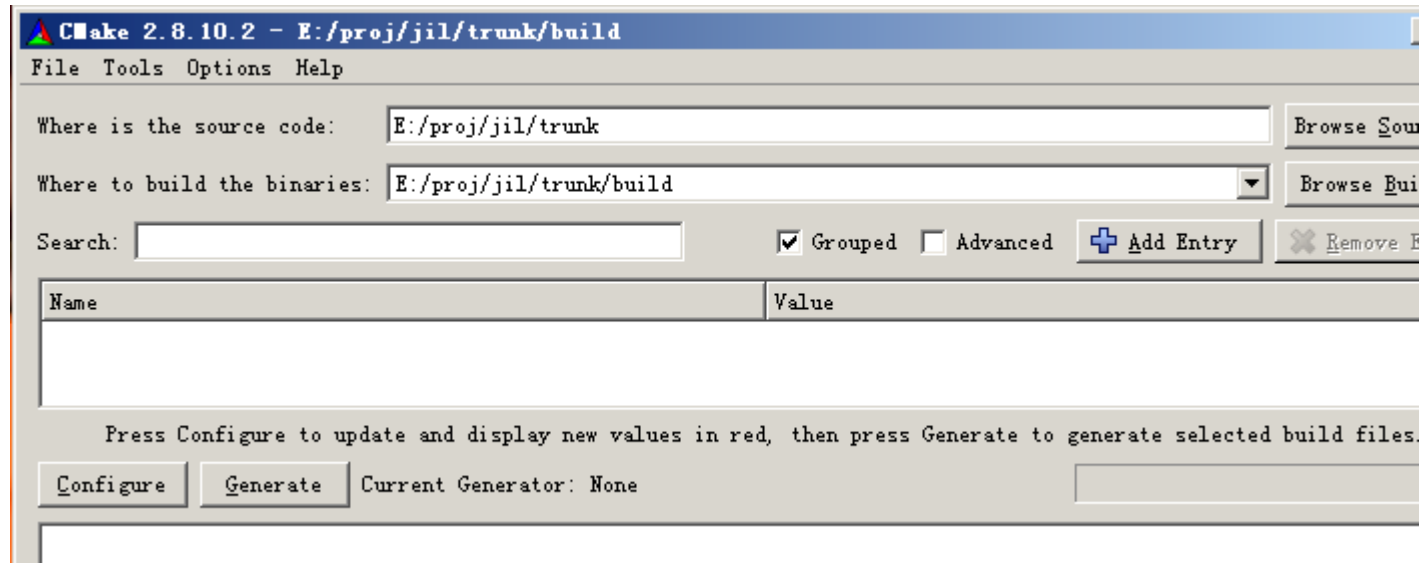
比如：

```
Boost_DIR = D:\boost
```

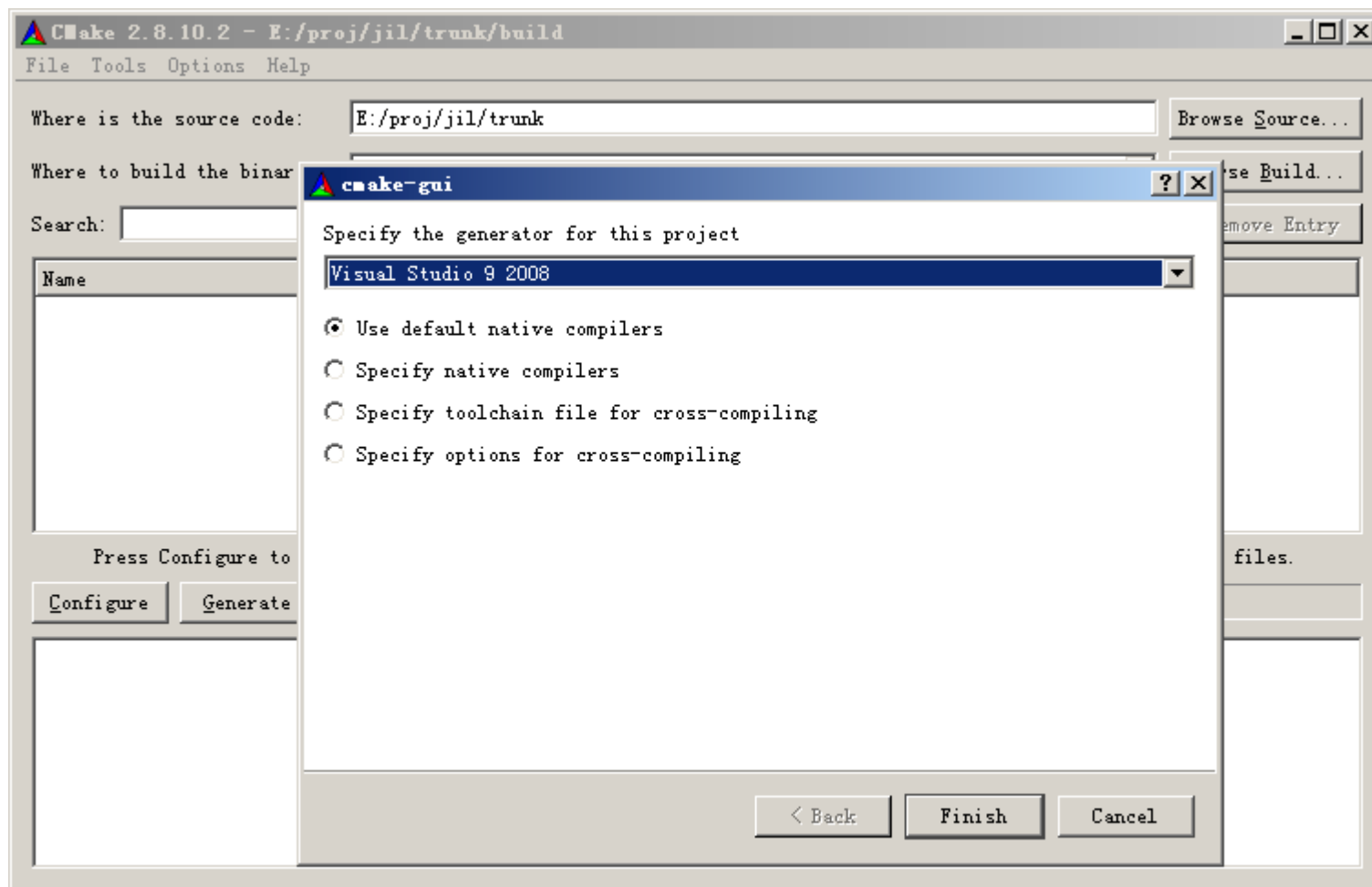
## Install CMake

下载最新版并安装: <http://www.cmake.org/>

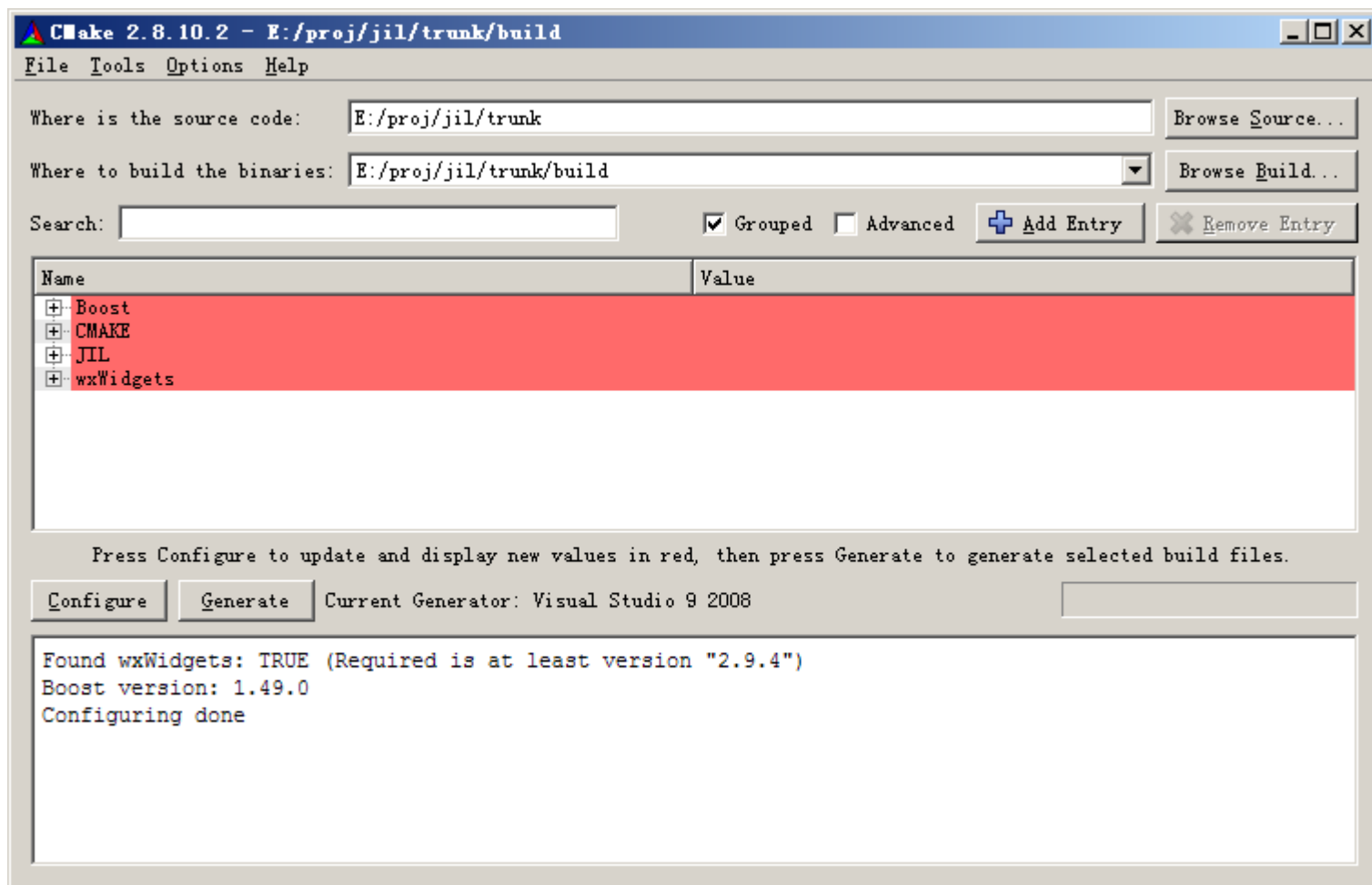
启动 CMake。



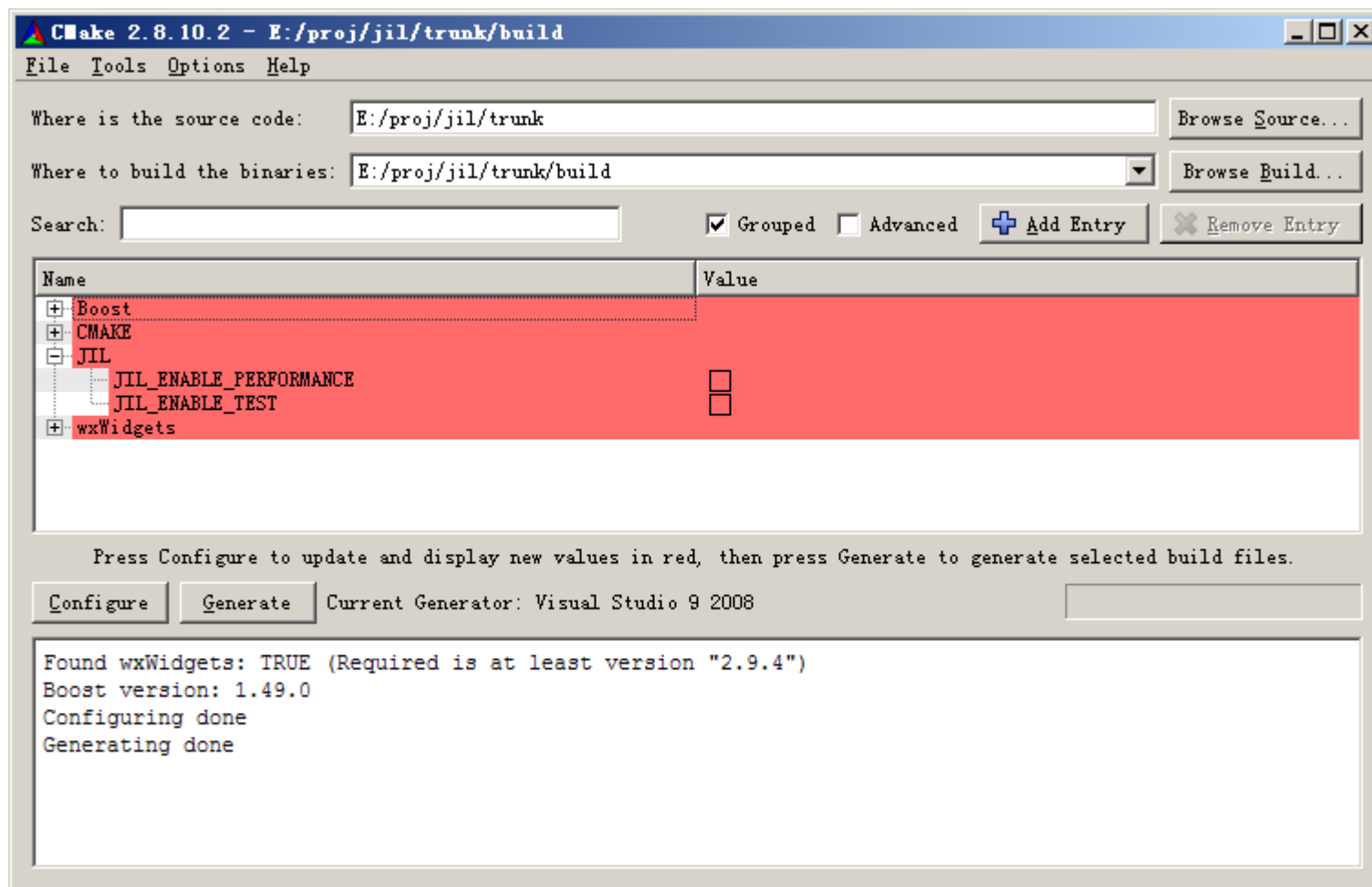
根据代码存放位置，输入 Jil 的根目录和 build 目录，如上图所示。然后按 Configure。



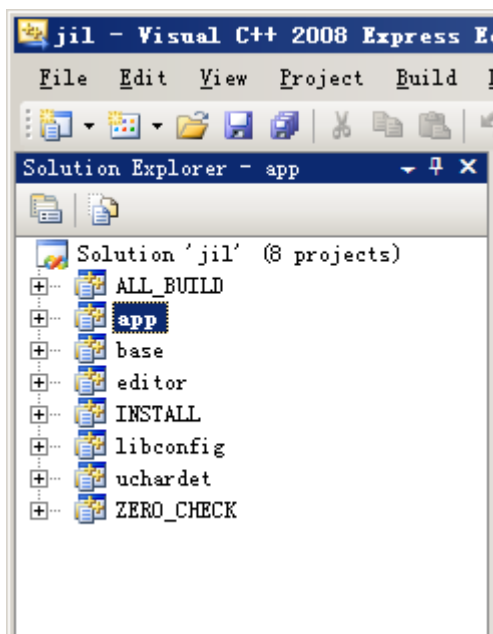
选择 Visual Studio 版本，比如我用的是 VC++ 2008 Express，那么就选 Visual Studio 9 2008。然后按 Finish。



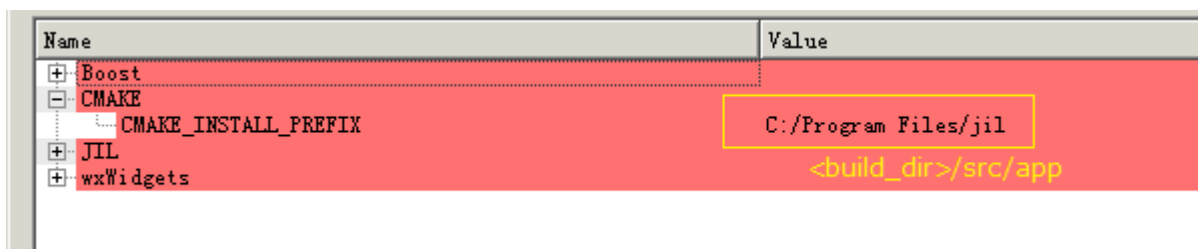
如果前面各环境变量设置无误的话，这一步应该直接通过。CMake 的消息框里显示 Configuration done，还有它所找到的 wxWidgets 和 Boost 各为什么版本。接下来按 Generate 生成 VC 的 solution 文件。



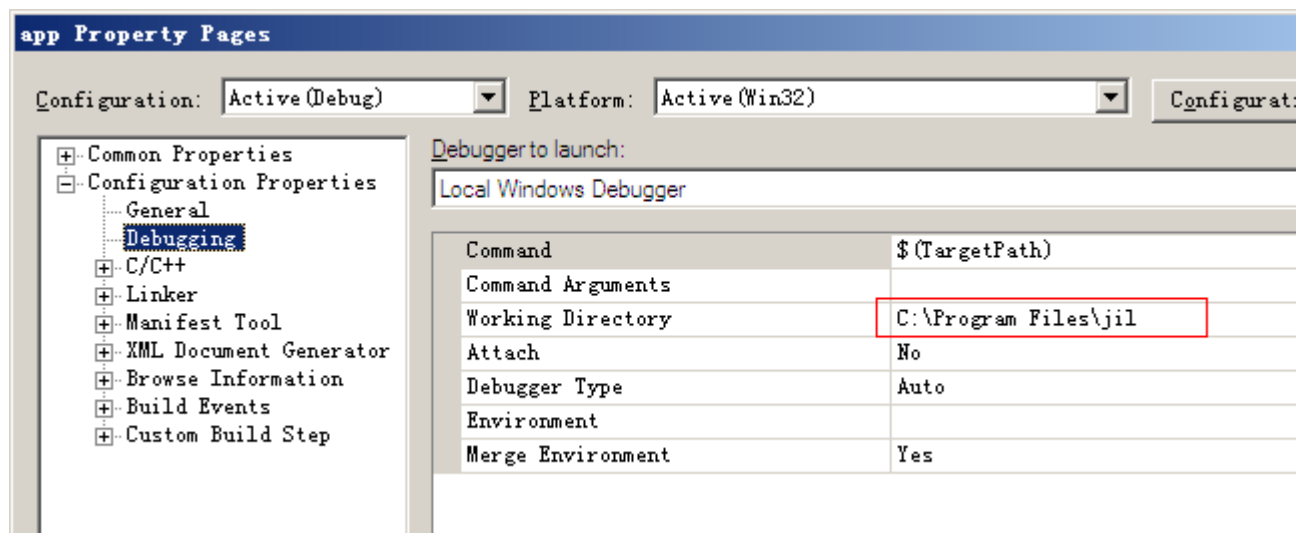
这一步没问题的话，solution 文件就生成了。打开 build 目录，打开 jil.sln 文件。



Cmake 自动生成了几个额外的 projects，不用管 ZERO\_CHECK，INSTALL 则相当于 make install，ALL\_BUILD 通常也不用，因为开发过程中 INSTALL build 一次基本上就够了，所以一般设置 app（生成 exe）为 start-up project。INSTALL 这个 project 会把可执行文件、动态库什么的，拷贝到一个指定目录（缺省为 C:\Program Files\jil，见下图）。对 Windows 平台来说，INSTALL 没有太大用处。我一般设置 CMAKE\_INSTALL\_PREFIX 为 <build\_dir>/src/app:



如果保留 CMAKE\_INSTALL\_PREFIX 的缺省值，为了在 VC 里直接运行程序，需要相应地设置 Working Directory:



但是不建议这样做，因为虽然 Working Directory 设置的是安装目录，但是每次生成的 exe 却还在 build\src\app\Debug|Release，而 Jil 认为缺省配置文件和 exe 同目录，所以在 VC 里直接运行或调试时，缺省配置文件就读不到了。

## Config Files

如前所述，缺省配置文件和 exe 同目录，用户配置文件放在用户目录。

如果是 XP，用户目录为：C:\Documents and Settings\<User>\Application Data\jil。Win7 什么的，类似。目前，你可以拷贝缺省配置文件到用户目录，然后就可以改动并测试效果了。