

###### Autoscan

用来扫描源代码目录生成configure.scan文件的。autoscan可以用目录名做为参数，但如果你不使用参数的话，那么autoscan将认为使用的是当前目录。autoscan将扫描你所指定目录中的源文件，并创建configure.scan文件。configure.scan包含了系统配置的基本选项，里面都是一些宏定义。我们在实际使用过程中，往往把configure.scan文件改成configure.in或者configure.ac文件（mv configure.scan configure.in）。

###### **Aclocal**

是一个perl 脚本程序。aclocal根据configure.in文件的内容，自动生成aclocal.m4文件。aclocal的定义是：“aclocal - create aclocal.m4 by scanning configure.ac”。 （若autoscan.in文件缺少了AC\_INIT\_AUTOMAKE宏，就无法生成aclocal.m4文件)

注：aclocal.m4文件。当时在学习的时候，非常纠结于这个m4后缀到底是什么意思，虽说是个无关痛痒的问题，但是每次看到这个文件，总会去琢磨这个后缀到底代表了一个什么东东。其实非常简单，m4就是代表macro的意思，也就是说m后面有4个字母

###### **autoconf**

是用来产生自动配置软件源代码脚本configure文件的。configure是一个脚本，它能设置源程序来适应各种不同的操作系统平台，并且根据不同的系统来产生合适的Makefile，从而可以使你的源代码能在不同的操作系统平台上被编译出来。configure.in文件的内容是一些宏，这些宏经过autoconf 处理后会变成检查系统特性、环境变量、软件必须的参数的shell脚本。configure.in文件中的宏的顺序并没有规定，但是你必须在所有宏的最前面和最后面分别加上AC\_INIT宏和AC\_OUTPUT宏。要生成configure，必须用到aclocal.m4。

###### **Automake**

configure脚本创建后，若这时候执行./configure命令，会提示出错,如下：

config.status: error: cannot find input file: `Makefile.in'

因为还没有把Makefile.in文件生成出来，那么如何生成Makefile.in呢？首先需要编写Makefile.am文件，看看后缀名.am就可以知道，这个Makefile.am是给automake工具使用的，通过Makefile.am文件，把Makefile.in生成出来，然后通过./configure命令读取执行Makefile.in，把Makefile生成出来。最后才是我们的"make"和“make install"

automake只要按部就班，依照顺序敲几个命令是非常容易的，唯一麻烦的就是写Makefile.am文件，需要在每个参与编译的子目录下都创建一个Makefile.am文件。

逐个的把所有Makefile.am整好后，就可以在software目录下执行automake --add-missing命令了，执行完成之后，会在每个目录下面自动生成Makefile.in文件。最后执行linux安装文件三部曲，"configure","make","make install"，可以随便执行了。

###### **configure**

这一步一般用来生成 Makefile，为下一步的编译做准备，你可以通过在 configure 后加上参数来对安装进行控制，比如代码:

./configure --prefix=/usr

上面的意思是将该软件安装在 /usr 下面，执行文件就会安装在 /usr/bin （而不是默认的 /usr/local/bin)，资源文件就会安装在 /usr/share（而不是默认的/usr/local/share）。