

emerge

虽然我们能够使用[autoconf](#)、[automake](#)、[make](#)等工具实现自动化编译，但这种针对单个软件包的编译系统，在编译多个软件时仍然十分繁琐

假设需要编译[emacs](#)和[vim](#)，使用 [xft](#)字体、图形界面支持，并去掉调试符号，需要分别作如下配置

```
emacs ./configure --prefix=/usr/local/ \--no-debug \--with-xft \--with-x-toolkit=gtk \--with
```

这就像点菜时，你必须告诉厨师：鱼香肉丝(多放辣椒、不放蒜)、宫保鸡丁(多放辣椒、不放蒜)、麻婆豆腐(多放辣椒、不放蒜).....

实际上，大多数人这样点菜：鱼香肉丝、宫保鸡丁、麻婆豆腐.....多放辣椒、不放蒜

[emerge](#)就是这样一种点菜方式，它是[gentoo](#)的包管理系统，提供了更为现代化的编译方式

可以通过指定**USE**标记[xft](#)、[gtk](#)、[-debug](#)来确定所有软件的编译方式

设置 USE 标记

以下方法按优先级由低到高排列：

[/etc/make.profile/](#)目录是一个符号链接，里面包含一些[make.defaults](#)文件，放置开发者设置的 **USE**标记[\[31\]](#)：

```
/usr/portage/profile/base/make.defaults/usr/portage/profile/default-linux/make.defaults/usr/portage
```

在[/etc/make.conf](#)文件中声明永久 **USE**标记(推荐)

```
USE="nptl nptlonly nls cjk php mysql -kde -qt3 -qt4"
```

- 带**-**的 **USE**标记，表示排除

在[/etc/portage/package.use](#)文件中为单个包声明 **USE**标记

```
app-editors/emacs-cvs xft www-servers/lighttpd fastcgidev-lang/php mysql cgi gd ctype pcre session
```

使用环境变量声明临时 **USE**标记

```
USE="-java" emerge seamonkey
```

查看使用的 **USE**标记：

```
merge --pretend --verbose seamonkey Calculating dependencies ...done![ebuild R ] www-client/
```

编译选项

[/etc/make.conf](#)

```
CFLAGS="-O2 -march=i686 -pipe" ❶CXXFLAGS="-O2 -march=i686 -pipe" ❷CHOST="i686-pc-linux-gnu" ❸MAKEO
```

- ❶ 针对C语言的优化选项，**-march=**设置目标架构
- ❷ 针对C++语言的优化选项
- ❸ 进行编译工作的机器架构
- ❹ 编译选项
- ❺ [emerge](#) 特性。并行下载、使用 [ccache](#) 缓冲编译结果

- ⑥ ccache 缓存目录
- ⑦ ccache 缓存大小
- ⑧ 通过关键字选择分支。**x86**表示 x86 架构的稳定分支，**~x86**表示 x86 架构的不稳定分支
- ⑨ USE标记
- ⑩ 使用**axel**加速下载

gentoo支持多种架构：x86、sparc、amd64、ppc、ppc64、alpha、hppa、mips、ia64、arm，我们使用的 PC 多为 x86 架构

假设你主要使用 x86 稳定分支，但少数软件要使用最新版本，在 `/etc/portage/package.keywords`文件中为单个包设置关键字

```
app-editors/emacs-cvs ~x86x11-misc/emacs-desktop ~x86app-il8n/fcitx ~x86app-editors/vim x86app-edit
```

微调

在 `/etc/portage/`目标下包含一些文件，可以在软件包级别上进行调节。前面已经介绍了 `package.use`和 `package.keywords`

- package.keywords**
还未被确认适合你的系统或架构，但是你能希望能安装的软件包
- package.use**
特定软件包而不是整个系统使用的 USE标记
- package.provided**
屏蔽的软件包(需要指明版本号)
- package.mask**
永远不希望 Portage 安装软件包。
- package.unmask**
被 Gentoo 开发者屏蔽的软件包，但是你能希望能安装软件包。

软件包可能由于以下原因被屏蔽

~架构 keyword	意味着这个软件没有经过充分的测试，不能进入稳定分支，请等待一段时间后在尝试使用它
-架构 keyword 或 -* keyword	意味着这个软件不能工作在您机器的体系结构中
missing keyword	意味着这个软件还没有在您机器的体系结构中进行过测试
package.mask	意味着这个软件被认为是损坏的，不稳定的或者有更严重的问题，它被故意标识为“不应使用”
profile	意味着这个软件不适用于您的 profile 。安装这样的应用软件可能会破坏您的系统，或者只是不能与您使用的 profile 相兼容

例如：

```
gnome-base/gnome-2.8.0_pre1 (masked by: ~x86 keyword)lm-sensors/lm-sensors-2.8.7 (masked by: -sparc
```

使用 **emerge**



注意

本部分内容来源于**gentoo 中文手册**

查找.

查找名字包含 pdf 的软件包

```
emerge --search pdf
```

查找与 pdf 相关的软件包

```
emerge --searchdesc pdfemerge -S pdf
```

查看软件拥有的 USE 标记

```
emerge -vp 软件包名称
```

管理。

安装软件包

```
emerge 软件包名称
```

模拟安装软件包

```
emerge --pretend 软件包名称
```

下载软件包的源代码包

```
emerge --fetchonly 软件包名称
```

从系统中删除软件包

```
emerge --unmerge 软件包名称
```

更新。

更新系统

```
emerge --update --ask world      emerge -ua world
```

更新整个系统

```
emerge --update --deep world      emerge -uD world
```

使用新的 USE 标记 重新构建系统

```
emerge --update --deep --newuse world      emerge -uDN world
```

移除孤立依赖的软件包

```
emerge --update --deep --newuse world❶ emerge --depclean❷ revdep-rebuild❸
```

- ❶ 重新构建系统
- ❷ 清除孤立依赖包
- ❸ 重新构建依赖关系

revdep-rebuild 工具由 gentoolkit 包提供；使用前别忘了首先 emerge 它：

```
emerge gentoolkit
```

[31] 在升级 Portage 的时候，这些文件将会被覆盖，请不要在这里设置