

An Emacs Tutorial for Vim User

w0mTea

November 28, 2014

Contents

1 前言	2
2 为什么离开 vim 改用 emacs	2
3 emacs 的安装	2
4 基础知识	3
4.1 快捷键的约定	3
4.2 常用快捷键	3
5 简单配置	3
6 过渡——evil-mode	4
6.1 安装	4
6.2 启用	6
7 emacs 中的包管理	6
7.1 el-get	6
7.2 ELPA	6
7.3 手动安装	6
8 保护你的手指	6
8.1 用手掌外缘按 ctrl	7
8.2 改键	7
8.3 踏板	7
9 编程语言配置	8
10 重量级应用——org-mode	8
11 文档和资料	8
12 结尾	8

1 前言

写这篇教程的起因在于向许多 `vim` 推荐 `org-mode` 时，他们总是觉得虽然 `org-mode` 功能强大，可是使用 `emacs` 总是有着一些障碍。作为一个同样从 `vim` 转向 `emacs` 的人，我觉得或许分享我的经验可以让他们更快的接受 `emacs`，从而体验 `emacs` 的美妙。

本文默认读者所使用的是类 `unix` 操作系统，因此对于 `windows` 用户来说，如果某些操作在你的电脑上无法执行，请不必大惊小怪。

2 为什么离开 `vim` 改用 `emacs`

在绝大多数 `unix/linux` 教程都教我们用 `vi/vim` 的情况下，许多人从开始就习惯了 `vim` 的一切：它的简洁、高效，它的模式，它简单直接的按键绑定……于是就这样，一批批的 `vim` 用户诞生。然而对于一直和 `vim` 并称的 `emacs`，我们往往只是知道它很强，却不清楚它到底强在哪里。如果你想清楚的了解这个 `vim` 的老对手到底凭什么能和 `vim` 分庭抗礼，那么，你可以试着使用它。

如果你是一个 `lisp` 爱好者，那么你绝对不能错过 `emacs`。或许经过漫长的配置，`vim` 同样可以很好的支持 `lisp`，可是在 `emacs` 上，`lisp` 天然就可以被良好的支持。同时，`emacs` 也可以被 `lisp` 扩展。因此，你写的每一行 `lisp` 代码，或许都可以让你的 `emacs` 变的更好用。写 `lisp`，也是让我离开 `emacs` 转向 `vim` 的契机。

如果你厌倦了 `unix` 哲学，厌倦了做一件事只能用无数的小工具来组合；或者你喜欢一个大而全的东西，可以帮你做各种各样的事情，那么 `emacs` 无疑是一个更好的选择。事实上，往往 `emacs` 用的越久，每天对着它的时间也越长，它帮你做的事情也越超出简单的文本编辑。

3 `emacs` 的安装

`emacs` 并不像 `vim/vi` 那样几乎被所有类 `unix` 系统内置，因此我们往往需要手动安装 `emacs`。

对于有包管理器的系统，使用包管理器通常都可以成功安装 `emacs`。需要注意的是，某些发行版的仓库默认不安装 `emacs` 的 GUI 版本，因此需要手动安装 `emacs-x11` 或类似名字的包。

需要注意的是，`emacs` 的 GUI 版本并不是 `xemacs`。我们使用的 `emacs` 实际上是 `gnu emacs` 这个分支，而 `xemacs` 则是另一个 `emacs` 的分支。虽然 `xemacs` 和 `gnu emacs` 有着相当的兼容性，可是在某些时候难免会碰到奇怪的问题。

对于没有包管理器或仓库没有 `emacs` 的系统，可以从 <http://ftp.gnu.org/gnu/emacs/> 下载。

4 基础知识

毫无疑问，对于一个从未是用过 vi/vim 的人来说，使用 **emacs** 并不是一个特别让人困惑的事，最起码开始不是，因为 **emacs** 和他之前用过的任何文本编辑器（比如 windows 下著名的 **notepad**）从表面上看并没有太大的不同。然而对于习惯了 vi/vim 模式操作的人来说，这样的操作模式真是让人无比烦恼：如果不使用一系列快捷键，效率就会变的无比低下；可是如果使用快捷键……那都是什么鬼东西！

为了让我们可以更快的接受 **emacs**，我们需要了解一些最最基础的东西。这些知识并不繁琐，但却总是有很大的帮助

4.1 快捷键的约定

由于没有 vim 那样的模式之分，**emacs** 的快捷键总是需要使用组合键。可是网上查到的 C-n, C-x C-s 都是啥意思呢？

emacs 中，快捷键的表示都遵循了一些固有的约定。C-x 表示同时按着 ctrl 和 x，C-x C-s 表示先按 ctrl+x，然后按 ctrl+s。当然，也可以按着 ctrl 不放，然后依次按 x 和 s 咯。

同样的，还有 M-系列的快捷键。M-x 表示同时按 alt 和 x（alt 在不同键盘上可能不同，有可能也叫 meta 之类的）。在这样的约定里，还有一些其他的特殊键，比如 ESC、RET（回车）等

4.2 常用快捷键

这里列举一些最简单同时也最最常用的快捷键。

快捷键	功能
C-x C-f	打开某文件
C-g	取消正在输入的命令
C-x C-c	关闭 emacs
C-x C-s	保存当前文件
M-x	运行命令

5 简单配置

emacs 的配置文件可以是下列三个中的一个：

- ~/.emacs
- ~/.emacs.el
- ~/.emacs.d/init.el

虽然说使用哪一个配置文件都可以，可是我还是建议使用最后一种。因为这种方案下，你可以把 **emacs** 相关的所有配置都放在 **.emacs.d** 这个文件夹

下，而不是零散的东堆西散。尤其在你的配置文件变的很大的时候，你可以轻松的把一些部分拆分成单独的模块放在这个文件夹里。

另外，下文提到的包管理中，最好也把其相关文件放在`.emacs.d`文件夹下。

至于具体的配置，可以根据自己的需求来弄。后文会提供一些好的站点帮助大家完成自己的配置文档。而一些简单的配置，会在后面的内容里提到。

6 过渡——evil-mode

对于刚接触 `emacs` 的 `vimer` 来说，最难习惯的估计就是光标移动了。如果还能像 `vim` 那样操作无疑会与愉快的多。而像 `vim` 一样操作 `emacs` 并不是你一个人的想法，因此也有现成的帮手来帮助我们完成目标，那就是 `evil-mode`。

6.1 安装

`emacs` 有着若干种安装扩展的方法，具体的会在下一节讲到。这里只讲一种我最常用到的也是感觉最方便的方法：`el-get` 安装。

在你的配置文件中加入下列部分：（需要注意的是，`el-get` 的默认位置也在`.emacs.d`文件夹内）

```
(add-to-list 'load-path "~/.emacs.d/el-get/el-get")

(unless (require 'el-get nil t)
  (url-retrieve
   "https://github.com/dimitri/el-get/raw/master/el-get-
install.el"
   (lambda (s)
     (end-of-buffer)
     (eval-print-last-sexp))))

;; now either el-
get is `require'd already, or have been `load'ed by the
;; el-get installer.

;; now set our own packages
(setq
 my:el-get-packages
 '(el-get                                ; el-get is self-
 hosting
   switch-window                        ; takes over C-x o
   auto-
 complete                               ; complete as you type with overlays
   zencoding-mode                       ; http://
www.emacswiki.org/emacs/ZenCoding
   color-theme                          ; nice looking emacs
```

```

    color-theme-tango))                ; check out color-
theme-solarized

;
;; Some recipes require extra tools to be installed
;;
;; Note: el-get-
install requires git, so we know we have at least that.
;;
(when (el-get-executable-find "cvs")
  (add-to-list 'my:el-get-packages 'emacs-goodies-
el)) ; the debian addons for emacs

(when (el-get-executable-find "svn")
  (loop for p in '(psvn                ; M-x svn-status
yasnipet                ; powerful snippet mode
)
do (add-to-list 'my:el-get-packages p)))

(setq my:el-get-packages
  (append my:el-get-packages
    (mapcar #'el-get-source-name el-get-sources)))

;; install new packages and init already installed packages
(el-get 'sync my:el-get-packages)

```

上述代码段会自动检查是否安装了 **el-get**，并自动在未安装的情况下安装。注意，这段代码需要系统中安装过 **git** 才能运行。同时为了在安装其他扩展时不会出问题，建议安装 **svn** 或 **cvs**。把上述代码段保存后，重新运行 **emacs**，就会自动安装 **el-get**

el-get 安好了，那么怎么安装 **evil-mode** 呢？回到上面那段代码，可以看到

```

;; now set our own packages
(setq
  my:el-get-packages
  '(el-get
    switch-window
    auto-complete
    zencoding-mode
    color-theme
    color-theme-tango))

```

只要在这段代码内添加上我们想要的扩展，而且这个扩展恰好在 **el-get** 的仓库内，那么我们就可以自动的安装并启用对应扩展。大多数常见扩展都可以被 **el-get** 自动找到，**evil-mode** 也不例外。因此只要在这段代码中加上 **evil-mode** 就可以。搞定后和下面的差不多：

```
(setq
my:el-get-packages
'(el-get
  switch-window
  auto-complete
  evil-mode
  zencoding-mode
  color-theme
  color-theme-tango))
```

之后重启 **emacs**，就可以安装了。

6.2 启用

安装成功后，只需要在配置文件中加入

```
(require 'evil)
(evil-mode 1)
```

就可以全局启用 **evil-mode**。如果想手动启动 **evil-mode**，把上面的 **1** 改成 **0**，在需要启动的时候按 **M-x evil-mode RET** 即可。

现在，**vim** 熟悉的操作，不就回来了吗？

7 emacs 中的包管理

在上一节，我们已经使用了 **el-get** 来安装扩展。只需要在列表中加入你需要的包名就可以自动安装，岂不是爽的很？这一节会介绍一些 **el-get** 的其他用法，以及其他的安装方法。

7.1 el-get

7.2 ELPA

7.3 手动安装

8 保护你的手指

emacs 需要大量使用 **ctrl** 和 **alt** 两个键，但在大部分 **qwerty** 键盘上，**ctrl** 的位置都在很难按到的角落里。据说，如果长期使用小指按角落的 **ctrl** 会很容易导致手部健康出现问题。因此，我专门加入了这一章来列举一些常用的方法来避免 **ctrl** 和 **alt** 位置不当带来的伤害。

8.1 用手掌外缘按 ctrl

由于许多键盘中 `ctrl` 处于左下角，所以可以把左手外翻向左下角压去，这样就可以按到 `ctrl`。

- 优点：简单，不需要特别的准备
- 缺点：笔记本键盘很难用，按着 `ctrl` 时左手几乎无法按其他任何键
- 推荐度：2/5

8.2 改键

仔细观察，不难发现我们的键盘上总有一些位置很黄金却很少用到的键，这之中典型的例子之一 `caps lock` 键。因此，我们不妨更改键位设置，把使用频度更高的键更换到这些位置上。

一些常用的改键方案包括：

1. 左 `ctrl` 和 `caps lock` 交换：似乎是网上流传最广的改键方法
2. 右 `alt` 和右 `ctrl` 交换：这种改法最适合空格两边都是 `alt` 的键盘，这样大拇指稍稍移动就可以按到 `ctrl` 和 `alt`

改键方法视具体环境不同而有所不同。在 `windows` 下，可以使用各种改键软件完成这一工作。在 `*nix` 下，对于使用 `xorg` 的用户来说也可以使用 `xmodmap`。如果使用 `DE`，那么很有可能在设置中心内也有调整键盘布局的选项。

- 优点：效果不错，可以根据自己的情况自由配置
- 缺点：需要自己进行一些准备；偶尔使用被改的冷门键可能会不方便
- 推荐度：4/5

8.3 踏板

有一类被称为踏板的神奇道具，可以定义踩下时发出的按键信号，这类踏板用于 `emacs` 那真真是极好的，可以极大的减轻手部负担。

- 优点：简单方便，效果超群，直接减少手的工作量
- 缺点：相比上面的方法来说最贵；不同系统驱动可能有潜在问题
- 推荐度：3/5

9 编程语言配置

10 重量级应用——org-mode

11 文档和资料

- emacs manual: http://www.gnu.org/software/emacs/manual/html_node/emacs/index.html
- emacs wiki: <http://www.emacswiki.org/emacs/>

12 结尾

本文仓促写成，错漏颇多，还望各位指出错误，让这份教程可以帮助更多的人。