

從 Vi 到 Emacs 筆記

Author

April 27, 2017

© 龔存

1 轉向 Emacs 陣營

轉向 Emacs 的原因主要是我在寫 RDT 協議的相關文檔時，由於採用近似 RFC 格式的 txt 文檔，發現使用 Vim 確實很不順手，由於寫中文本身就要頻繁的切換輸入法，與 Vim 本身的編輯模式和輸入模式的切換混合在一起，就非常的痛苦。使用其他的編輯器，要麼不能跨平台，要麼不能定制和可編程，所以最後我只能嘗試使用 Emacs。

Emacs 帶給我驚喜，使用它寫類 RFC 文檔真的非常方便，不僅沒有模式切換帶來的痛苦，而且縮進方面做得非常好，本來我只想寫文檔時用用 Emacs，其他時候還是用 Vim 好了，但是漸漸我就是想用 Emacs，寫文檔用它，寫程序用它，記事本用它，寫 Markdown 用它，甚至寫郵件也用它。

但是有很多 Vi 的快捷鍵真的根深蒂固了，當然可以使用 evil-mode 或者 viper-mode，但是這樣來說就沒太大意義，畢竟我就是厭倦了 Vi 的模式切換。所以花一點時間，學習一下 Elisp 編程，或者到 stack-overflow 上抄抄別人的答案并試著理解，我根據一份“Emacs for vi users”的清單，定制了若干 Vi 鍵綁定，大部分以 **ESC-ESC** 作為前綴，列在下面，供大家參考。（關於區域內搜索和替換，我還沒寫出鍵綁定。）

另外，Emacs 的初始化配置文件 `.emacs` 也是經常變動的，所以一併加入版本管理。

2 Emacs for vi users

E-E corresponds to *ESC-ESC*, **C-a** corresponds to *Control+A*, and **M-f** corresponds to *Meta+F*

Action	Vi	Emacs	Notes on Emacs
Go to beginning of buffer	1G	C-up	Mac OS X don't have Home key
Go to end of buffer	G	C-down	Mac OS X don't have End key
Go to line n	n G	E-E g	
Go to beginning of line	0	C-a	
Go to end of line	\$	C-e	
Set mark x	m x	E-E m	
Go to mark x	' x	E-E '	
Go to first displayed line	H	E-E H	
Go to last displayed line	L	E-E L	
Go to middle displayed line	M	E-E M	
Delete n lines	n dd	M- n E-E dd	
Yank n lines	n yy	M- n E-E yy	
Open line above	O	C-M-o	
Open line below	o	C-o	
Join lines	J	E-E J	
Search forward for character x in current line	f x	E-E f	Input x
Search forward to character before x in current line	t x	E-E t	Input x
Delete forward for character x in current line	df x	E-E d f	Input x
Delete forward to character before x in current line	dt x	E-E d t	Input x
Delete to the end of line	D	C-k, or E-E D	
Delete to the beginning of line	d0	E-E d 0	
Delete to the end of buffer	dG	E-E d G	
Delete to the beginning of buffer	:1,d	E-E d g	
Yank from <i>beginning</i> to <i>end</i> lines	: <i>beginning</i> #, <i>end</i> #y	E-E r y	Input <i>beginning</i> #, <i>end</i> #
Delete from <i>beginning</i> to <i>end</i> lines	: <i>beginning</i> #, <i>end</i> #d	E-E r d	Input <i>beginning</i> #, <i>end</i> #
Auto-complete single word	C-x C-n	M-/	
Auto-complete whole line	C-x C-l	C-x C-l	
Compile command	:make	F8	
Move to the next error message and visit the corresponding source code	:cnext	C-x '	Use ctags or etags to make file tags first
Run shell command	!: <i>shell-command</i>	E-E ! <i>shell-command</i>	
Read from shell command	:r! <i>shell-command</i>	C-u E-E ! <i>shell-command</i>	
Write to shell command and read back	: <i>beginning</i> , <i>end</i> ! <i>shell-command</i>	M-h M- <i>shell-command</i>	