

Vim

Pierre Cosquer

Association Actux

23 mars 2011

- 1 Introduction
- 2 Prise en main
- 3 Un peu plus compliqué
- 4 Vim pour coder

Vim : Éditeur de texte puissant et efficace. Antonyme : Emacs, nano, notepad.

Vim : Éditeur de texte puissant et efficace. Antonyme : Emacs, nano, notepad.

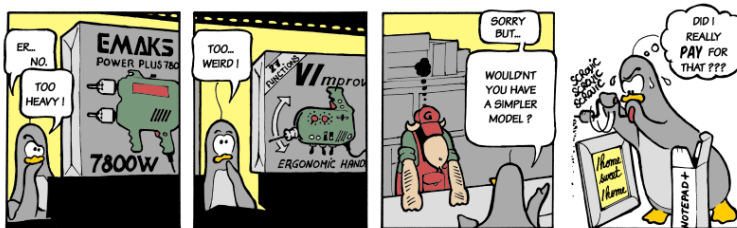


FIGURE: Copyright (c) 2007 Laurent Gregoire

- ① *ed*, développé par Ken Thompson au début des années 1970, éditeur modal ligne par ligne
- ② *vi*, éditeur modal visuel développé par Bill Joy en 1976
- ③ *Vim* (*VI iMprove*), clone de *vi* développé par Braam Moolenaar pour Amiga, puis porté sur de nombreux OS.

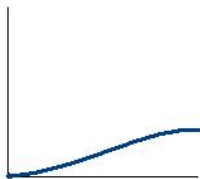
Classical learning curves for some common editors

TT
11-17-09

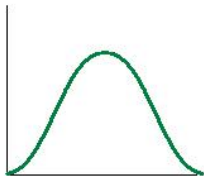
Notepad



Pico



Visual Studio



vi



emacs



Vim réagit différemment suivant dans quel mode il se trouve. Il y a 6 principaux modes :

Normal ou Commande Touche « Echap » Mode par défaut de Vim.

Insertion Touche « i ». Permet d'insérer du texte.

Ligne-de-commande Touche « : ». Permet de lancer une commande puis de retourner en mode normal.

Visuel Touche « v ». Permet de sélectionner une zone de texte et d'appliquer une commande dessus.

Sélection Touche « Ctrl-G » en mode visuel. Supprime la zone sélectionnée et passe en mode Insertion.

Ex Touche « Q ». Permet de lancer plusieurs commandes à la suite.

Déplacements

h, j, k, l ou les flèches
et beaucoup d'autres comme w pour se placer sur le mot à droite

Suppression

x pour supprimer un caractère
d*déplacement* pour supprimer du curseur vers le déplacement
dd pour supprimer la ligne entière

Copier/Coller

y*déplacement* pour copier
p ou P pour coller

Quelques commandes

Sauvegarde de fichier

:w nom du fichier

Ouverture de fichier

:e nom du fichier

Recherche

/*motif* recherche *motif* en avant

Substitution de texte

:s/*motif*/*remplacement*/ remplace une fois *motif* par *remplacement* sur la ligne courante

:%s/*motif*/*remplacement*/g remplace toutes les occurrences de *motif* par *remplacement* dans tout le fichier

Quelques raccourcis sympathique

Annulation

En mode commande, « u » annule une commande, « U » restaure la ligne entière.

Complétion

En mode insertion, « ^p » et « ^n » affiche une pop-up d'auto-complétion.

Vim permet d'éditer plusieurs fichiers (buffers) à la fois

- `:e fichier` ouvre un buffer
- `:badd fichier` ajoute un buffer sans l'ouvrir
- `:bp` et `bn` ouvre le buffer précédent/suivant
- `:ls!` liste les buffers
- `:b bufname` ouvre le buffer *bufname*
- `:bd bufname` supprime le buffer *bufname*

Le plugin MiniBufExplorer affiche en permanence la liste des buffers

Vim permet d'afficher plusieurs fichiers, voire plusieurs fois le même fichier

- `:split` et `:vsplit` découpent la fenêtre
- `Ctrl-w h`, `Ctrl-j`, `Ctrl-w k` et `Ctrl-l` pour se déplacer
- `Ctrl-w +` et `Ctrl-w -` pour redimensionner la fenêtre active
- `:help windows` pour plus de détails

Vim supporte de nombreuses options

- `:set opt` active l'option *opt*
- `:set opt=valeur` assigne *valeur* à l'option *opt*
- `:set noopt` désactive l'option *opt*
- `:help opt` affiche l'aide de l'option *opt*

On peut sauvegarder ses préférences dans le vimrc :

- `:e $MYVIMRC`

Quelques options à mettre dans son vimrc :

```
set nocompatible " désactive la compatibilité avec vi
syntax on " active la coloration syntaxique
filetype plugin indent on
set number " affiche les numéros de lignes

:colorscheme permet de changer le thème de couleur
```

gg=G : réindente proprement tout le code