

Premiers pas en Emacs

Christophe Pallier*

22 décembre 2000

Résumé

L'utilisation d'un système GNU/Linux nécessite de modifier ou d'écrire fréquemment des fichiers textes. Il est donc nécessaire de connaître et de savoir utiliser au moins un programme éditeur de texte. Vi et emacs sont sans doute les deux programmes les plus puissants, mais aussi les plus ésotériques. Pico et nedit sont deux éditeurs simples à utiliser mais assez limités, et qui, de plus, ne sont pas forcément disponibles sur tous les systèmes. Le but de cet article est de faciliter la prise de contact avec Emacs. Il s'adresse donc à ceux qui voudraient bien l'essayer mais se sentent un peu "perdus" face à l'austérité de son écran d'accueil.

Pourquoi utiliser Emacs ?

Emacs (prononcer "imax") est l'éditeur de texte créé, en 1984, par Richard Stallman, l'instigateur principal du projet GNU. Il est disponible dans tout système GNU/Linux qui mérite ce nom. Par rapport à des éditeurs plus modernes, Emacs, il faut bien le reconnaître, est d'un abord rébarbatif. L'auteur de ces lignes a essayé pendant des années d'éviter d'utiliser Emacs, en choisissant d'autres éditeurs en apparence plus conviviaux, mais qui s'avéraient toujours plus ou moins limités ou buggés. Finalement il a fini par adopter Emacs quand il a réalisé qu'il

*Copyright (c) 2000 Christophe Pallier. Permission est donnée de copier, distribuer et modifier ce document selon les termes de la licence GNU pour les documentations libres, version 1.1, publiée par la Free Software Foundation (www.fsf.org). L'original de ce texte est accessible sur <http://www.pallier.org>.

était possible, en quelques heures, d'acquérir suffisamment de notions pour que ses avantages contre-balancent nettement ses défauts.

L'un des principaux intérêts d'Emacs réside dans les modes qui spécialisent son comportement en fonction du type de fichiers sur lequel on travaille. Cela se révèle très appréciable lorsque, par exemple, on produit un document en Latex ou en HTML, écrit un script shell, un Makefile, ou rédige un programme en C ou en Perl. Emacs n'est bien sûr pas le seul éditeur présentant cette caractéristique. Toutefois, durant sa déjà longue existence, de nombreux contributeurs ont perfectionné ces modes et écrit des extensions d'Emacs, de façon que pour quelque problème qu'on se pose, il existe pratiquement toujours déjà une solution (au point qu'Emacs possède des extensions qui peuvent sembler un peu "farfelues": certains packages permettent de convertir du texte en code morse, de faire du calcul formel (p.ex. dériver $\sin(x)/x$), ou de jouer au morpion (gomoku)..).

Un autre aspect intéressant d'Emacs est qu'il peut servir d'environnement au sein duquel l'on commande certains programmes interactifs (par exemple le puissant package statistique 'R'). Cela permet d'utiliser divers programmes à l'intérieur d'un environnement avec lequel on est familier. On peut également lire son mail sur un serveur POP à l'intérieur d'Emacs, ou bien encore "explorer" son système de fichiers avec le mode "dired"... En fait Emacs est, plutôt qu'un simple éditeur de texte, un véritable environnement de travail. Ainsi, lorsqu'on travaille sous Windows, le mode shell d'Emacs se trouve être une formidable solution pour pallier les terribles limitations des boîtes DOS (absence d'historique des sorties, pas de complétion automatique des noms de fichier..)

Le principal écueil du débutant, sans nul doute, ce sont les séquences de touches à mémoriser. En réalité, pourtant, une petite vingtaine de séquences de touches (que nous présentons, parmi d'autres, dans cet article) suffissent pour réaliser 99% des tâches; de plus certaines de ces combinaison de touches sont utilisables dans la plupart des applications interactives sous unix/linux (bash, netscape...). Après quelques heures d'utilisation elles deviennent automatiques. Je déconseille d'utiliser les menus, ou la souris, ce qui s'avère inefficace à long terme et ralenti l'apprentissage.

Créer, ouvrir et fermer des fichiers

Sur la ligne de commande d'un terminal, tapez 'emacs'. Quand emacs apparaît, appuyez sur la touche 'Ctrl' et, tout en la maintenant enfoncée, appuyez

sur 'x' puis 'f' (combinaison de touches qu'on notera C-x C-f; notez qu'il n'est pas nécessaire de relacher la touche Ctrl). Le message 'Find file:' s'affiche sur la dernière ligne (appelée 'minibuffer' selon la terminologie emacs). Entrez 'test1' et appuyez sur 'ENTER', 'test1' est le nom du fichier créé. Entrez quelques lignes en frappant normalement au clavier (par exemple 'Ceci est un essai ENTER d'Emacs'). Puis appuyez sur C-x C-s pour sauver votre fichier, puis C-x C-c pour sortir d'Emacs. Le tableau suivant décrit pas à pas la suite d'actions que vous venez d'effectuer.

| | |
|-------------------------|--|
| C-x C-f test1 ENTER | ouvre un nouveau fichier au nom de test1 |
| Ceci est un essai ENTER | écrit sur la première ligne |
| d'Emacs | écrit sur la deuxième ligne |
| C-x C-s | sauve le fichier |
| C-x C-c | quitte emacs |

Vous pouvez vérifier le contenu du fichier que vous venez de créer en entrant la commande 'cat test1' sur une ligne de commande du shell. Vous devriez alors voir s'afficher les deux lignes 'Ceci est un essai' et 'd'Emacs'.

Relancez Emacs et ouvrez votre fichier avec 'C-x C-f test1 ENTER'. Fermez-le immédiatement en tapant 'C-x k ENTER' (relacher la touche Ctrl avant de frapper 'k'). Ouvrez-le à nouveau, faites quelques modifications et tapez 'C-x k ENTER'. Emacs vous demande confirmation avant de détruire ('kill') le buffer 'test1'. Si vous répondez 'yes', vos modifications seront perdues; si vous répondez 'no', l'opération est abandonnée et vous aurez l'opportunité de sauver votre fichier avec 'C-x C-s'. Sachez également que si vous appuyez sur C-x C-c trop rapidement et qu'emacs vous demande si vous voulez ou non sauver des fichiers, vous pouvez renoncer à sortir en tapant 'C-g'. Plus généralement 'C-g' permet d'abandonner l'action courante. C'est une combinaison de touches très utile! Effectivement, si Emacs ne répond plus, ou si vous ne comprenez plus où vous êtes, tapez sur 'C-g', éventuellement à plusieurs reprises; il y aura alors toutes les chances que vous vous retrouviez dans une situation mieux maîtrisée.

Il y a d'autres façons d'ouvrir le fichier 'test1'. Tout d'abord vous auriez pu écrire 'emacs test1' sur la ligne de commande. Encore une autre façon, de l'intérieur d'Emacs, est d'utiliser 'C-x C-f' puis la touche TAB pour obtenir la complétion automatique de nom de fichier: Essayez 'C-x C-f te TAB'; si 'test' est le seul nom de fichier qui commence par 'te', il s'affichera directement. Si par contre il y a d'autres fichiers dont le nom commence par 'te', alors leur liste s'affichera dans une fenêtre. Vous pouvez déplacer le curseur dans cette fenêtre avec la séquence 'C-x o' (tapez cela plusieurs fois, et observez comment le curseur

se déplace de fenêtre en fenêtre); utilisez les flèches pour déplacer le curseur sur le nom du fichier qui vous intéresse et tapez “Enter”.

Emacs permet bien sûr d’ouvrir plusieurs fichiers simultanément ; pour cela, utilisez ’C-x C-f’ plusieurs fois. Chaque fichier s’ouvre dans un nouveau ‘buffer’. Pour passer d’un buffer à l’autre, on peut utiliser ’C-x b’ et taper le nom du fichier destination (sans oublier que la complétion du nom par TAB fonctionne). Une autre manière consiste à afficher la liste des buffers avec ’C-x C-b’; on peut alors se déplacer vers cette liste avec un ’C-x o’ suivi de déplacements avec les flèches. Finalement, ’C-x 1’ supprime les fenêtres supplémentaires et permet de retrouver le buffer courant dans une seule fenêtre dans l’écran d’Emacs. Faites l’exercice suivant:

| | |
|-------------------------------------|--|
| C-x C-f test1 ENTER | ouvre le fichier test1 |
| C-h t | ouvre le tutorial d’emacs dans un nouveau buffer |
| C-x b ENTER | retourne a test1 |
| C-x C-b C-x o C-n ENTER C-x o C-x 1 | se promener |

Finalement, voici un récapitulatif de commandes utiles pour naviguer d’un fichier ou d’une fenêtre à l’autre. Dans la dernière colonne, nous avons indiqué le nom de la fonction Emacs associée à la séquence de touche. Cela est utile si vous désirez changer les fonctions des touches. (En fait, Emacs possède de nombreuses fonctions dont seule une minorité sont associées à des séquences de touches. On peut cependant exécuter n’importe quelle fonction en tapant Esc-x suivit du nom de la fonction).

| | | |
|---------|--|-------------------------|
| C-x C-c | quitte emacs | save-buffers-kill-emacs |
| C-g | abandonne l’action courante | keyboard-quit |
| C-x C-f | ouvre ou crée un fichier | find-file |
| C-x C-s | sauve le buffer courant | save-buffer |
| C-x C-w | sauve dans un nouveau fichier | write-file |
| C-x k | ferme le buffer courant | kill-buffer |
| C-x b | change de buffer | switch-to-buffer |
| C-x C-b | affiche la liste des buffers | list-buffers |
| C-x 1 | ferme toutes les fenêtres sauf la courante | delete-other-windows |
| C-x o | déplace le curseur vers une autre fenêtre | other-window |
| C-x 5 f | ouvre un fichier dans une autre fenêtre X | find-file-other-frme |
| C-x 5 0 | détruit la fenêtre X (frame) | delete-frame |

Se déplacer et éditer un fichier

Le tableau suivant liste les touches de déplacement les plus utilisées. Sous la plupart des systèmes Linux, la touche Alt fait office de touche 'META' (dans le langage de la documentation d'Emacs) donc 'M->' signifie appuyer sur Alt et '>' simultanément. Notez que ces combinaisons de touches fonctionnent de la même façon sur les lignes de commande des shells bash et tcsh, ainsi que dans de nombreux programmes interactifs tels que gnuplot, octave... Ainsi votre apprentissage sera vite rentabilisé.

| | | |
|----------|--------------------------------------|------------------------|
| C-v, M-v | avancer vers le bas ou vers le haut | scroll-up, scroll-down |
| M-> | aller en fin de fichier | end-of-buffer |
| M-< | aller en début de fichier | beginning-of-buffer |
| C-a | aller en début ligne | beginning-of-line |
| C-e | aller en fin de ligne | end-of-line |
| C-p | ligne précédente | previous-line |
| C-n | ligne suivante | next-line |
| C-d | effacer le caractère sous le curseur | delete-char |
| C-k | effacer jusqu'à la fin de la ligne | kill-line |
| C-x u | annuler l'action précédente | undo |

Couper/Copier/Coller

Ouvrez un fichier ou bien créez-le avec Emacs. Déplacez le curseur au début d'un passage que vous souhaitez couper ou coller. Appuyez sur 'C-<SPC>': cela pose une "marque" au début du passage. Déplacez vous jusqu'à la fin du passage, puis tapez 'C-w' pour couper le texte (ou 'M-w' pour le copier). Allez là où vous voulez le coller, puis tapez 'C-y' (yank). Vous pouvez coller le texte (avec C-y) à plusieurs reprises.

Une propriété intéressante du "presse-papier" d'Emacs est qu'il souvient non seulement du dernier passage copié (ou coupé), mais également des précédents: si vous avez coupé plusieurs passages, presser C-y affiche le dernier, mais si vous appuyez ensuite sur M-y, l'avant-dernier est inséré. On peut appuyer plusieurs fois sur M-y pour remonter plus en arrière dans l'historiques des passages coupés. Cela peut s'avérer très utile si l'on a effacé par inadvertance un long passage, et beaucoup travaillé depuis de façon que la commande undo (C-x u) n'est plus efficace.

On a souvent besoin de couper le texte jusqu'à la fin de la ligne courante; pour cela on peut utiliser 'C-k'; Plusieurs 'C-k' successifs effacent autant de lignes.

Vous remarquerez que, contrairement à d'autres éditeurs de texte, le texte placé entre la marque et le curseur (i.e. le texte "sélectionné") n'apparaît pas en video-inverse (en fait cela est vrai pour un Emacs d'origine, votre configuration peut varier). Cela peut paraître gênant au premier abord, mais il y a une bonne raison à cela (on peut poser plusieurs marques : cf. la section 'mark' dans la documentation d'Emacs, accessible par 'C-h i m emacs Enter m mark Enter'). Pour savoir où vous avez posé la dernière marque (i.e. le début de la région "sélectionnée"), vous pouvez utiliser l'une des deux commandes: (1) C-x C-x, qui échange la position de la marque et du curseur; ou (2) C-u C-SPACE, qui amène le curseur à la position de la marque.

Si vous tenez absolument à voir le texte entre la marque et le curseur en vidéo inverse, exécutez la commande 'transient-mark-mode' (tapez 'Alt-x', puis 'transi', puis TAB, vous devez voir s'afficher 'transient-mark-mode'); Appuyez alors sur la touche 'Enter'. Dans ce mode, le texte sélectionné change de couleur. Quand vous voudrez sortir du mode 'sélection', utilisez 'C-g'.

Quelques remarques à propos des marques sont intéressantes. Premièrement, on peut poser plusieurs marques dans un texte, et revenir successivement de l'une à l'autre par 'C-u C-SPC'. Deuxièmement, certaines commandes posent implicitement des marques. Par exemple allez au milieu d'un fichier texte, puis tapez 'M->' pour aller en fin de fichier. Si vous tapez ensuite 'C-u C-SPC', vous revenez automatiquement au point de départ. Une marque est également posée implicitement quand vous effectuez une recherche de chaîne de caractères.

Il se peut que votre système permette de copier/coller avec la souris. Par exemple, sous X-Window: cliquer à gauche et glissez pour sélectionner du texte (si vous avez relâché le bouton avant d'avoir atteint la fin du texte à sélectionner, allez en cette position, puis cliquez sur le bouton de droite pour étendre la sélection); pour coller, cliquez sur le bouton du milieu. Ce mécanisme est également utile pour copier du texte entre Emacs et une autre application X, par exemple une fenêtre terminal.

Chercher/Remplacer

Pour chercher un mot, tapez 'C-s' puis les premières lettres de ce mot. Observez bien le comportement d'Emacs : à chaque fois que vous tapez une lettre, il avance dans le texte de façon incrémentale. Utilisez C-g pour interrompre la recherche. Une fois le mot localisé, si vous désirez trouver les occurrences suivantes, tapez à nouveau C-s. Quand vous serez habitué à cette recherche incrémentale, vous aurez du mal à vous en passer sous d'autres éditeurs. Si vous désirez revenir à l'endroit d'où vous avez démarré la recherche, tapez 'C-u C-SPC' (Emacs a posé une marque implicitement à l'endroit où vous avez commencé la recherche). C-s cherche en avant dans le texte; pour chercher en arrière, utilisez C-r ('C-r' est bien utile sous bash: il permet de rechercher une ligne dans l'historique des commandes!).

Pour chercher et remplacer des chaînes de caractères, utilisez 'M-%'. Notez une possibilité intéressante: il est possible d'entrer des retours chariots dans la chaîne recherchée avec 'C-q C-j'. Par exemple, pour ajouter systématiquement un signe \$ à la fin de chaque ligne, tapez 'M-% C-q C-j ENTER \$ C-q C-j ENTER'.

Il est également possible de chercher et de remplacer en utilisant des expressions régulières (grosso modo des "wildcards"). Pour chercher une expression régulière, utilisez 'M-C-s'. Supposons par exemple que vous vouliez rechercher toutes les initiales de noms (une lettre majuscule suivie d'un point) dans votre texte; il vous suffit de taper 'C-M-s SPC [A-Z]' Si voulez chercher et remplacer utilisez la command 'M-x query-replace-regexp'. La description détaillée des expressions régulières nécessiterait plusieurs pages. Pour bien comprendre comment elles fonctionnent, il vous faudra lire la documentation en-ligne d'Emacs, ce que le paragraphe suivant va nous apprendre à faire.

| | |
|-----------|--|
| C-SPC | pose une marque (p.ex. début de sélection) |
| C-u C-SPC | va à la dernière marque |
| C-x C-x | échange le curseur et la marque |
| C-w | coupe le texte entre la marque et le curseur |
| M-w | copie le texte entre la marque et le curseur |
| C-y | colle le presse-papier |
| M-y | remonte dans l'historique des textes coupés |
| C-s, C-r | recherche incrémentale |

Utiliser l'aide

Il est très chaudement recommandé d'apprendre à se servir des différents systèmes d'aide en-ligne d'Emacs, auxquels on accède par C-h.

Par exemple, tapez 'C-h t'. Vous devez voir s'afficher un "tutorial", c'est à dire une documentation interactive pour débutant. Parcourez la rapidement, en utilisant les touches C-v et M-v pour descendre ou monter dans le texte. Ce Tutorial est un bon complément du présent article. Quand vous en aurez assez, fermez-le avec 'C-x k'.

La séquence de touche 'C-h i m emacs' vous emmène dans la documentation d'Emacs, au format 'info'. Par exemple, tapez 'm regexps' pour lire les quelques pages décrivant la syntaxe des expressions régulières (déplacez vous en appuyant sur la touche espace, remontez en arrière avec la touche 'u') Tapez 'C-x k' pour fermer cette documentation. Plus généralement 'C-h i' permet d'accéder à toutes les documentations au format info, notamment aux utilitaires GNU, et à la documentation de la librairie C ('C-h i m libc'), ce qui est quasiment indispensable pour programmer en C sous Linux. 'C-h i h' vous présentera des explications plus complètes pour utiliser efficacement le mode 'info'.

Les fonctions suivantes sont également très utiles:

'C-h k' décrit la fonction associée à une touche; par exemple 'C-h k C-k' affiche la description de la commande 'kill-line'.

'C-h f' décrit une fonction. Essayez, par exemple, 'C-h f iso-accents-mode'.

'C-h a' liste les fonctions dont le noms contient un mot clé. Par exemple, 'C-h a spell' vous listera les fonctions liées à la détections des fôtes d'orthographe.

'C-h p' liste les modules d'extensions Emacs (fichiers .el ou .elc). Pour charger un module, on doit utiliser la commande 'load-library' et fournir le nom du module. Par exemple si vous chargez le module 'autorevert' puis tapez M-x autorevert-mode: emacs detectera alors toute modification sur le disque du fichier associé au buffer courant.

Les modes

On n'attend pas le même comportement d'un éditeur selon qu'on travaille sur un texte ou un programme en C. Emacs possède des modes distincts pour travailler plus efficacement sur les différents type de fichiers. Le mode est affiché au milieu de la ligne d'état en bas de l'écran. Bien qu'Emacs tâche de détecter le type de fichier à l'ouverture, il faut néanmoins savoir changer manuellement de mode, par exemple lorsqu'on crée un nouveau fichier. Comme toujours sous Emacs, il suffit d'exécuter la commande adéquate, par exemple: `M-x text-mode`, `M-x latex-mode`, `M-x perl-mode`... Pour obtenir la liste des modes accessibles, tapez `'C-h a mode'`. Pour avoir de l'aide sur le mode courant, tapez `'C-h m'`.

Emacs possède également la notion de 'mode mineur', qui spécifie, en quelque sorte, des sous-modes. Par exemple `'auto-fill-mode'` spécifie que emacs doit revenir automatiquement à la ligne quand une certaine colonne est atteinte. La remise en forme d'un paragraphe se fait manuellement avec `'M-q'`.

Un des modules d'extension les plus remarquables d'Emacs fournit des modes pour TeX et LaTeX. Nommée AUCTeX, il rend l'édition de documents LaTeX très confortable : les mots clés LaTeX sont colorés, le reformatage des paragraphes tient compte des commandes latex (par exemple, un environnement verbatim ne sera pas reformaté accidentellement!), `'C-c-c'` lance la compilation par latex ou le visualiseur xdvi..., etc.

Quelques "trucs" en vrac

Et voici pour finir, quelques fonctions intéressantes:

| | |
|-------------------------------|--|
| M-x ispell-buffer | appelle le correcteur orthographique |
| M-x iso-accents-mode | permet de taper des caractères accentués avec une clavier américain |
| M-! | exécute une commande shell |
| M- | exécute une commande shell sur la région cou- rante |
| C-x r k et C-x r y | Couper/coller des colonnes |
| C-u 72 * | entrer 72 fois le caractère '*' |
| C-u 3 M-x sort-numeric-fields | trier un tableau sur la troisieme colonne |
| M-x global-set-key | permet d'associer une fonction à une touche |
| M-x compile | execute 'make -k' |