# Journal : Bash et les raccourcis claviers

Posté par Bienvenue le 19/09/17 à 21:54. Édité 2 fois
mots-clés : bash, raccourcis, bienvenue

## Sommaire


- Préambule
- Présentation du journal
- Historique et exécution de commandes
- Manipuler le tableau
- Auto-complétion
- Nettoyer et gérer l'affichage
- Gérer les processus
- Exécuter une ancienne commande
- Bonus, Bonus


Bonjour nal

Aujourd'hui je vais parler des raccourcis clavier sur Bash. Par ce que j'ai vu ailleurs.

C'est très pratique et nécessaire pour pouvoir travailler Bash en bouquant le moins possible aux mains. Tant qu'on écrit du texte, c'est facile. Mais très souvent, on a besoin de bouquer à l'intérieur d'histoire, de supprimer un caractère, d'avoir l'auto-complétion, etc. Pour cela, le même set d'outiner les touches dédiées de clavier existe. Et ça peut vraiment faire gagner du temps. Et ça rend le travail beaucoup plus agréable et moins fatiguant pour les articulations.

Bash supporte une série de raccourcis clavier hérités de l'emacs. Et, ceux de l'emacs ou de la vieille vi(1), ceux en caractère. Chaque raccourcis a sa fonction que nous verrons tout au long de l'article.

Les touches alt et control sont alors aussi bien du clavier, mais elles sont utilisables facilement sans trop bouger ou tordre ses mains.

**Important**

Pour la suite de ce tutoriel, je retrouve les raccourcis. M-x pour les raccourcis avec la touche Alt et C-x pour ceux avec la touche Control.

**Manipuler le curseur**

Commençons par le commencement. Comment avancer et reculer le curseur de bash?

- C-f : Avancer d'un caractère
- C-b : Reculer d'un caractère

F, fait référence au mot anglais "Forward" (vers l'avant) et B à "Backward" (vers l'arrière).

Il est aussi possible de déplacer le curseur d'un mot grâce aux d'un caractère :

- M-f : Avancer d'un mot, l'arrêter à la fin de celui-ci
- M-b : Reculer d'un mot, l'arrêter au début de celui-ci

Pour aller au début ou en fin de ligne, utiliser :

- C-a : Aller au début de la ligne
- C-e : Aller au fin de la ligne
- C-xx : Utiliser entre la position actuelle du curseur et le début de la commande

**Historique et exécution de commandes**

Pour exécuter une commande que vous venez de taper, presser :

- C-m : Retour chariot

Si vous voulez naviguer dans l'historique des commandes, comme avec les touches fléchées :

- C-p : Commande précédente (p en référence à "previous")
- C-n : Commande suivante (n pour "next")

Il est aussi possible de rechercher une commande dans l'historique :

- C-r : Recherche dans l'historique
- C-s : Si vous avez trouvé la commande, vous avez modifié une ligne nappable de l'historique, ce raccourcis ia remet en l'état (telle qu'elle se trouve dans l'historique).

Lorsque vous recherchez dans l'historique avec C-r, et que plusieurs occurrences sont trouvées, appuyez plusieurs fois sur C-r pour passer de l'une à l'autre. Le raccourcis ia fait de la commande la plus récente à la plus ancienne.

Lors de l'histoire d'une commande, vous pouvez mettre le ter mot de la précédente commande exécutée. Pour cela :

- M-. : Rapporte le ter mot de la précédente commande.

**Manipuler le texte**

Il est possible de supprimer un caractère avant ou après le curseur avec les raccourcis :

- C-d : Supprimer le caractère qui se trouve après le curseur
- C-h : Supprimer le caractère qui se trouve avant le curseur

Ces fonctions sont également disponibles pour supprimer les mots et les lignes :

- M-d : Supprimer le mot qui se trouve après le curseur
- C-w : Supprimer le mot qui se trouve avant le curseur
- C-k : Supprimer le texte qui se trouve après le curseur
- C-u : Supprimer le texte qui se trouve avant le curseur

A noter que les mots ions supprimés avec les commandes précédentes se retrouvent dans le presse papier de Bash. Ce presse papier est différent de celui du système. Il est accessible avec plusieurs commandes. Il offre toutefois la fonction de "Mot-Kring", comme Emacs. Lorsque vous remettrez le presse papier, ce dernier modifie pas ce qui l'entoure.

Tout est gardé en mémoire et vous pouvez choisir ce que vous voulez "coller".

Les raccourcis du presse papier de Bash sont :

- C-y : "Coller" le dernier élément présent dans le presse papier
- M-y : Remplacer le mot "collé" par le mot précédent. Cela utilise alors ce qui se trouvait précédemment dans le presse papier.

Bash permet également d'interpréter 2 lettres de 2 mots :

- C-t : Intervertit la lettre sous le curseur avec celle qui la précède
- M-t : Permet, avec deux mots.

Enfin, il est possible de transformer le cassé d'un mot :

- M-u : Convertit un mot en majuscule
- M-l : Convertit un mot en minuscule
- M-c : Capitalise un mot.

**Auto-complétion**

Faire appel à l'auto-complétion de Bash peut se faire avec les raccourcis :

- C-i : Équivalent à la touche TAB

**Annuler**

Il existe 2 raccourcis pour annuler :

- C-_ : Annule la dernière action
- C-x C-u : Annule une séquence dans l'historique

**Gérer les processus**

Quand un processus est lancé en premier plan depuis Bash, les raccourcis suivants sont disponibles :

- C-z : Envoyer le processus en arrière plan (équivalent à bg)
- C-c : Mettre le processus en premier plan (équivalent à fg). Il est possible pour ce ce article pas avec des commandes. Tg ou bg. La commande jobs liste les processus en pause.

A noter que lors de l'écriture d'une commande, C-c fonc soulor le curseur sur une nouvelle ligne de commande vide.

**Nettoyer et gérer l'affichage**

Il est possible de maintenir la sortie écran de Bash à l'écran avec les raccourcis :

- C-l : Nettoie l'écran, pour ne maintenir qu'une invite de commandes vide. Sinonera à la commande 'clear'.
- C-s : Stop l'affichage. Très utile quand un programme très verbeux s'exécute.
- C-q : Reprendre l'affichage stoppé avec C-s.

**Exit Of File**

Il existe un raccourci pour envoyer le caractère EOF (End Of File) :

- C-d

Si ce raccourcis est appelé sur une ligne de commande, cela équivaut à la commande exit.

**Bonus: Termux**

En bonus, j'en profite pour présenter un émulateur de terminal pour Android. Termux. Assez complet, léger, intégrant Bash. Je vous invite à aller voir leur site web.

## Y'a les mauvaises touches au bord du clavier et les bonnes touches au bord du clavier

Posté par Thomas (page perso) le 19/09/17 à 23:54. Édité 1 fois

Déjà, "Retour chariot", "Retour chariot", ça. Ces touches étant aux bords du clavier, cela demande quelques mouvements qui pourraient être évités.

Les touches alt et control sont elles aussi aux bords du clavier, mais elles sont utilisables facilement sans trop bouger ou tordre ses mains.

Pour le <TRB> et <Ctrl> je ne sais pas du tout d'accord.

> Tel est très accessible du demandant aux doigts extension du petit doigt vers l'extérieur.
> Ctrl, c'est pas pour ça, mais je comprends pas pourquoi tu dis que c'est pas pratique. Ctrl est pas loin du tab.

Bill pouvant souvent être atteint par le pouce droit, ce qui rend le souffle dans que du coup du Ctrl. Je parel directement sur un clavier de portable est souvent tout aussi inaccessible que le Ctrl.

Bill : Si tu m en a optimiser le frappe pas des raccourcis clavier ex-tu passé au Bépo qui te fera gagner bien plus de ce côté ?

Ps : Deux mots pour aller entre les mots : Ctrl+Left/Ctrl+Right.

## Re: Y'a les mauvaises touches au bord du clavier et les bonnes touches au bord du clavier

Posté par andpascalou le 20/09/17 à 00:24. Édité 1 fois

> Si tu m en a optimiser le frappe pas des raccourcis clavier ex-tu passé au Bépo qui te fera gagner bien plus de ce côté ?
>
> Yes. Sur un clavier TypeMatrix :
>
> Mais je suis en train de construire un clavier mécanique matriciel de 5 x 10 avec :
> - les touches "Retour chariot" en 2u,
> - les touches "Control" et "Retour arrière" en pavé et disposé de la touche d'espace
> - les touches "Alt" en pavé et disposé de la touche "Retour arrière"
> - les touches "Alt" de pavé et d'autre des touches "Ctrl"
> - les touches "Alt" et "Ctrl" de pavé en 1.25u
> - Pas de touches dédiées F1, F2, etc (F1 = Ctrl + 1)
> - Les touches "Escape" et "Tab" plus bas
>
> Ce qui demande, pour la dextérité, de jouer du clavier :
> - Ctrl+Tab+Esc+Alt+Retour arrière pour mettre entrer/Ctrl/AltGr/Ctrl/Alt+1/2/3/4/5/6

## Re: Y'a les mauvaises touches au bord du clavier et les bonnes touches au bord du clavier

Posté par Thomas (page perso) le 20/09/17 à 00:36. Édité 1 fois

Enfin un clavier ou Backspace n'est pas juste à côté de Enter.

## Re: Y'a les mauvaises touches au bord du clavier et les bonnes touches au bord du clavier

Posté par andpascalou le 20/09/17 à 00:44. Édité 1 fois

Ctrl, mais avec 60 touches, c'est dure de tout avoir.

## Re: Y'a les mauvaises touches au bord du clavier et les bonnes touches au bord du clavier

Posté par Thomas (page perso) le 20/09/17 à 00:56. Édité 1 fois

Ta phrase entre les caractères de contrôle disaccentuent, "M de T", ah mince...

## Readline

Posté par Gabu (Tilman (page perso)) le 19/09/17 à 23:51. Édité 3 fois

A noter que ce ne sont pas vraiment des raccourcis clavier de bash mais de readline et donc utilisable dans plein d'outils.

D'autre part, ces raccourcis sont configurables et il existe un mode vi, ce qui est vraiment plus pratique (certes echap est au cou du clavier mais ça évite les combinaisons ambidextres et puis avec un peu de bricolage, on peut le remplacer par une pédale).

## Re: Readline

Posté par Bienvenue (page perso) le 19/09/17 à 23:54. Édité 1 fois

Merciqui du renseigné ! Cet ai pas pensé à le mentionné et que très cher.

http://www.gnu.org/software/bash/manual/bashref.html#Bindings

http://www.gnu.org/software/bash/manual/bashref.html#Vi-mode

http://www.ms.uky.edu/public/pub/gnu/readline/readline.ps.gz

http://www.ms.uky.edu/public/pub/gnu/readline/readline.pdf

## Re: Readline

Posté par Bienvenue le 19/09/17 à 23:55. Édité 1 fois

Ou mieux, voici par exemple mon fichier ~/.inputrc :

```
$include /etc/inputrc
set bell-style none
"\e[1;5C": end of line
"\e[1;5D": beginning of line
"\e[3;5~": delete function
$endif
$if term=linux
# Alpha the current or previous word
"\C-w": "\eb\ed\C-y"
$endif
```

Les 2 premières commandes me permettent d'utiliser A-Left et A-Right pour me déplacer en début et fin de ligne équivalent à C-a et C-e.

La troisième commande permet d'utiliser C-Suppr pour effacer les mots. Les fonctions de readline sont leur raccourcis clavier. Sur mon terminal, "\e[3;5~" correspond à C-F3 (Suppr + Control). Je préfère les utiliser. Et pratique, ce raccourcis est équivalent à la commande bash **bind -P** qui est C-F3. Jeud thus utilisé alors ceuxer le commande en cours d'écriture.

Le quatrième commande est un fait une macro qui transforme **C-w** en **b.bashrc** ; **d** (bashrc) ; **C-y** (bashrc) ; **C-y** (bashrc) ; **C-y