

Pour lire un fichier texte

Les utilitaires en ligne de commande pour VISUALISER le contenu d'un fichier

En premier lieu, les outils présentés dans cette rubrique ne permettent pas de modifier le contenu d'un fichier. Leur seule fonction est de lire et d'afficher le contenu d'un fichier sur la sortie standard de la console.

La commande **cat** permet de lire le contenu du fichier passé en paramètre et affiche le texte sur la sortie standard.

```
$cat fichier.txt
```

La commande **tac** affiche quant à elle le contenu mais en commençant par la dernière ligne. C'est-à-dire que le fichier est lu en sens inverse.

```
$tac fichier.txt
```

Pour ajouter des numéros au début de chaque ligne, l'utilitaire **nl** est plus approprié. Il permet une visualisation des lignes du fichier sur la sortie standard.

```
$nl fichier.txt
```

D'autres utilitaires comme **head** ou **tail** donnent respectivement les dix premières et dix dernières lignes d'un fichier.

```
$head fichier.txt
```

```
$tail fichier.txt
```

Cette dernière commande est très pratique si on veut observer ce qui se passe en temps réel avec une commande. L'option **-F** est nécessaire (attention passez en **root**).

```
#tail -F /var/log/syslog
```

La commande se lance puis attend un événement. On ouvre alors une autre console pour exécuter une commande ou un programme afin de voir ce qui se passe. Par exemple, nous allons désactiver un service :

```
#service sudo stop
```

La console précédente doit alors afficher quelques informations en temps réel de ce qui vient d'être effectué. On se sert souvent de cette commande lors du plantage d'un programme pour voir où le bug se situe. Pour sortir de la commande **Ctrl-C** redonne le prompt.

On peut également demander à **head** ou **tail** d'afficher plus de dix lignes en renseignant le nombre *n* souhaité. Ainsi, la commande suivante permettra d'afficher les trente premières lignes du fichier

```
$head -30 fichier.txt
```

Deux programmes en lecture seule permettent également de pouvoir lire les fichiers à partir d'une interface texte. Le premier **more** permet une navigation de page en page alors que le second **less** donne accès à une lecture ligne par ligne. Ce dernier est très complet et ouvre à de multiples possibilités : navigation par raccourcis, ancrages, édition de plusieurs documents,

comparaisons, etc. (cf. `$less --help`). Une fonction très intéressante est l'ouverture de l'éditeur de texte du système par l'appui sur la touche `v` (`$echo $EDITOR`).

```
$more fichier.txt
```

```
$less fichier.txt
```

Les programmes en *mode texte* pour **ÉDITER** le contenu d'un fichier

On nomme ces programmes des éditeurs de texte. Ils permettent d'éditer du texte brut (aucun symbole n'est caché). Leurs fonctions se résument à **lire**, **couper**, **copier**, **coller** et **sauvegarder** du texte. Ces programmes ne formatent pas le texte. L'*italic*, le gras, la « couleur », la taille des caractères et les polices ne sont possibles. Nous entendrons par couleur non possible, celle du formatage. Par exemple : mettre un mot en rouge. En revanche, les couleurs syntaxiques sont très utilisées. Elles servent à donner du relief au code afin de se repérer plus aisément dans le fichier.

Les plus connus sont NANO, JOE, VIM, EMACS. Cependant il y en a beaucoup d'autres.

```
$vim fichier.txt
```

Les programmes en *mode graphique* pour **ÉDITER** le contenu d'un fichier

Il existe également de nombreux éditeurs de texte en mode graphique. Ils ne s'exécutent que dans un environnement graphique (sous X). L'environnement *GNOME* propose généralement **gedit**, *KDE* emploie **Kate**. Il y a également **leafpad**, etc. Chaque éditeur a ses avantages et inconvénients. **gvim** est une interface graphique à l'éditeur **VIM**.

```
$leafpad fichier.txt
```

Les programmes en *mode graphique* pour **TRAITER** le contenu d'un fichier

Ces nouveaux programmes, ne s'exécutent — comme les éditeurs en mode graphique — qu'en présence d'un environnement graphique (sous X). On les nomme **traitements de texte** car ils permettent notamment d'éditer du texte mais également de mettre en forme le texte, c'est-à-dire de le formater. Ainsi, il sera possible de mettre le texte en *italic*, en **gras** ou encore de proposer une police spéciale et de lui donner une taille plus ou moins importante.

Sous Linux, comme sous MAC et Windows, **LibreOffice Writer** est le plus connu actuellement. C'est un *fork* d'OpenOffice Writer qui n'est plus libre aujourd'hui. Pour lancer le programme à partir de la ligne de commande, lancer la commande suivante :

```
$lowriter fichier.txt
```

En résumé

Il existe des utilitaires en ligne de commande pour visualiser du texte. Ces programmes ne font que renvoyer sur la sortie standard de la console le contenu des fichiers passés en paramètres. Les plus utilisés sont **cat**, **tac**, **nl**, **head**, **tail**, **more** et **less**.

Ensuite, des programmes en *mode texte* permettent d'éditer du texte. On parle d'**éditeurs de texte**. Leurs fonctions d'édition sont très basiques : **lire**, **couper**, **copier**, **coller**, **sauvegarder**. Aucun formatage de texte n'est possible. Cependant, ces quelques fonctions peuvent être déclinées pour en faire des outils très adaptés. Les plus connus sont **NANO**, **JOE**, **VIM** et **EMACS**.

Un nombre plus important d'éditeurs de texte œuvrent en *mode graphique* ou interface graphique. Ainsi, selon l'environnement employé, des programmes comme **Gedit**, **Kate** **Leafpad** ou encore **Gvim** seront très utilisés.

Pour finir, les **traitements de texte** offrent la possibilité de formater le texte en lui donnant une forme plus agréable visuellement. Ainsi, l'italic, le gras, la taille du texte et les polices seront très largement employés dans un cadre de non accessibilité. Un des plus employé dans le monde du libre actuellement est le programme **LibreOffice Writer**.

D'autres outils en ligne de commandes comme **Sed** ou **Awk** étendent les possibilités des simples utilitaires. Ils feront l'objet d'une fiche complémentaire.