

# TP4 : Inodes et liens

---

## 1. Manipulation sur les fichiers

1. Allez dans votre répertoire *travail/unix* et créez un fichier *f1.txt* avec votre éditeur de texte favori (nedit, emacs, xemacs, vi, gedit, kwrite, etc.) Utilisez votre imagination pour écrire quelques lignes dans votre fichier et enregistrez-le (CTRL-s avec nedit, CTRL-x puis CTRL-s avec emacs).
2. Visualisez le contenu du fichier avec les commandes *cat*, *more* et *less* (man si vous voulez en savoir plus sur ces commandes).
3. Renommez votre fichier avec la commande *mv f1.txt f2.txt*. Observez le résultat avant et après avec *ls* et *ls -i*.
4. Créez un répertoire *test* ainsi qu'un sous-répertoire *test/sousTest* et déplacez-y le fichier avec la commande *mv f2.txt test/sousTest*. Tapez *ls -l* puis *ls -l test/sousTest* pour vérifier.
5. Chaque fichier est associé à un inode (noeud d'information). Un inode est donc une structure de données enregistrée sur le disque et contenant toutes les informations relatives au fichier (tout ce qui est produit pas *ls -l*) sauf son nom. Tapez *ls -lai*. Observez les numéros des inodes dans la colonne de gauche.
6. Déplacez/renommez le répertoire *sousTest* avec la commande *mv test/sousTest nouveauTest*. Utilisez *ls -lai* pour observer que les inodes n'ont pas changé. Ils n'ont pas changé car le répertoire est toujours le même : on l'a simplement déplacé, en changeant son nom, donc son chemin relatif. C'est la même commande qui permet sous Unix de renommer et de déplacer un fichier. On notera que *nouveauTest* contient toujours le fichier *f2.txt* (*ls -l nouveauTest*).
7. Dupliquez le fichier avec la commande *cp nouveauTest/f2.txt f3.txt*. Utilisez *ls -lai* pour vérifier que le nouveau fichier a bien un nouveau numéro d'inode. Modifier l'un des deux fichiers. L'autre est-il modifié ?

## 2. Liens physiques et symboliques

1. Créez un lien physique sur le fichier *f3.txt* avec la commande *ln f3.txt phyln.txt*. Comparez les inodes de ces deux fichiers. Observez également la seconde colonne du résultat de *ls -l*. Le numéro d'inode est-il le même ? Modifiez l'un des deux fichiers. L'autre est-il modifié ?
2. Créez un lien symbolique sur le fichier *f3.txt* avec la commande *ln -s f3.txt symln.txt*. Comparez les inodes de ces deux fichiers. Observez aussi comment ce nouveau fichier apparaît au travers des commandes *ls* et *ls -l* et les modifications éventuelles au niveau de *f3.txt*. Modifiez l'un des deux fichiers. L'autre est-il modifié ?
3. Supprimez maintenant le fichier *f3.txt* (*rm f3.txt*). Peut-on encore accéder à *phyln.txt* et *symln.txt* ?

Expliquez ce que sont les liens physiques et les liens symboliques, leurs avantages et inconvénients. En particulier, comment sont gérés les droits sur les liens physiques et symboliques ?

## 3. Redirection de la sortie standard

1. Exécutez la commande *date*
2. Exécutez la ligne de commandes suivante : *date > monFichier*
3. Affichez le contenu du fichier *monFichier* (*cat monFichier* ou *more monFichier*). Réfléchissez à ce qui s'est produit. Au besoin, réitérez ces deux lignes, et modifiez éventuellement le nom du fichier de sortie. Vous venez de *rediriger la sortie standard de la commande date vers le fichier monFichier*.
4. Exécutez les lignes de commandes suivante :

- echo Bonjour > monFichier
  - more monFichier.
- Que fait la commande echo (man echo)?

## 4. Retour aux liens

1. Enchaîner les commandes suivantes avec et sans l'option -s pour ln, puis expliquer.  
echo toto > fichier1; ln fichier1 fichier2; rm fichier1; more fichier2
2. Les "raccourcis" de Windows correspondent-ils plutôt à des liens physiques ou à des liens symboliques?

## 5. Commande file

1. Tester la commande file sur plusieurs types de fichiers différents (fichier normal, liens physiques et symboliques, répertoires, fichiers du répertoire /dev). A quoi sert-elle?