

2014-2015

---

## TD 1 - Système Linux

---

contacts : `mathias.gauduchon@univ-amu.fr`, `melika.baklouti@univ-amu.fr`,  
`xavier.giraud@univ-amu.fr`, `nicolas.barrier@univ-amu.fr`

Dès qu'il s'agit de manipuler des commandes, n'hésitez pas à utiliser la commande **man**.

Pour la suite des exercices, vous devez avoir ouvert un terminal linux. Pour ce faire, cliquez sur le menu "Applications" en haut à gauche de votre bureau puis sélectionnez l'onglet "Applications Incontournables" et enfin "Terminal".

### Exercice 1

Comme on l'a vu dans le cours, certaines commandes permettent d'explorer ou de modifier votre environnement de travail. À l'aide de **man** ou en exécutant la commande directement, analyser ce que produisent les commandes suivantes (commandes à taper dans l'ordre, ne pas retaper le symbole "\$" en début de ligne) :

```
$ pwd
$ ls
$ ls -a
$ ls -al
$ mkdir Essai
$ ls -l
$ cd Essai
$ pwd
$ ls
$ touch toto
$ ls -l
$ touch toto
$ ls -l
$ cd ..
$ rm Essai
```

Que se passe-t-il lors de l'exécution de la dernière commande ? Pourquoi ? Que faudrait-il faire pour effacer le répertoire `Essai` ? Le vérifier dans le terminal.

Que représente `..` ?

Faire la même chose que précédemment pour les commandes suivantes :

```
$ pwd
$ ls -l
$ cd /etc
$ pwd
$ ls -l
$ which date
$ cd /bin
$ pwd
$ ls -l
$ cd
$ pwd
$ ls -l
$ cd /
$ mkdir MonFichierSysteme
```

Que se passe-t-il lors de l'exécution de cette dernière commande ? Pourquoi ?

## Exercice 2

Revenir au répertoire personnel en tapant **cd**

Créer un répertoire **essai**. Dans ce répertoire, nous allons créer un fichier **texte.txt** contenant la phrase "Le ciel est bleu". Pour ce faire, ouvrir l'éditeur de texte **kate**. Taper dans le terminal

```
$ kate &
```

Le "&" permet de lancer l'application dans le **background** (l'arrière-plan) ce qui permet de ne pas bloquer le terminal pendant que le logiciel tourne encore. Taper alors dans la fenêtre qui apparaît la phrase : *Le ciel est bleu* et sauvegarder ce texte dans un fichier **texte.txt** dans le répertoire **essai**.

Vérifier que le fichier **texte.txt** a bien été créé dans le répertoire **essai** et vérifier son contenu.

Modifier alors le nom du fichier **texte.txt** en **textebis.txt** en utilisant la commande

```
$ mv texte.txt textebis.txt
```

Taper

```
$ ls
```

Ouvrir **texte.txt**. Que se passe-t-il ?

Revenir sous le répertoire personnel. Créer un deuxième répertoire **test** et un sous répertoire **tt**.

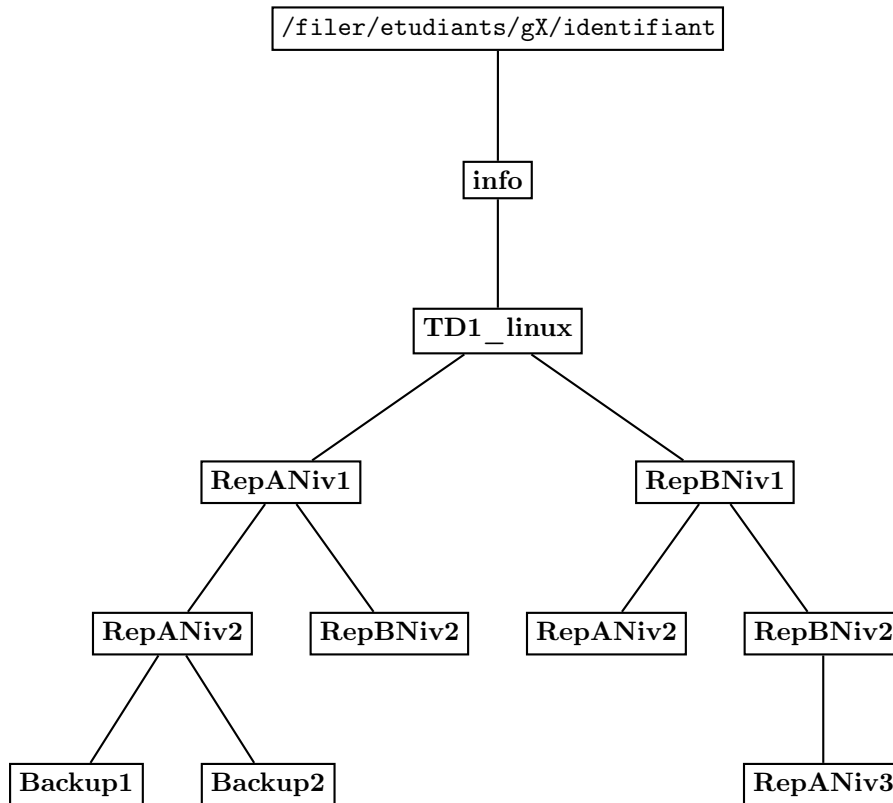
Déplacer alors le fichier **textebis.txt** dans le sous répertoire **test/tt** en utilisant la commande **mv**

En utilisant la commande **cp**, copier le fichier **textebis.txt** dans votre répertoire personnel. Vérifier.

Effacer alors le fichier **textebis.txt** dans le répertoire **/test/tt** en utilisant **rm**.

### Exercice 3

Créer dans son répertoire personnel l'arborescence décrite par :



A l'aide de firefox, aller chercher une photo de son acteur (ou actrice) préféré(e) et la sauvegarder dans le répertoire `RepANiv3`. Dans un terminal, copier ensuite ce fichier dans le répertoire `Backup2`. Pour finir, déplacer cette copie dans le répertoire `Backup1` en changeant son nom en `maphoto.jpg`.

### Exercice 4 (un peu plus difficile)

Le symbole “>” placé à la suite d’une commande permet de sauvegarder le résultat de cette commande dans un fichier. Si le fichier existe déjà, son contenu est détruit et est remplacé par le nouveau. La syntaxe est

`commande [-options] [arg1] [arg2] [...] > nom_fichier` (ne pas omettre les espaces autour de “>”).

Créer un fichier `essai.txt` contenant le résultat de `ls -l` (utiliser le symbole “>”). Le vérifier.

Le symbole “>>” se prête à la même utilisation que “>” à la différence qu’il ne détruit pas le contenu du fichier mais rajoute les nouvelles lignes de texte à la suite de l’ancien contenu.

Compléter le fichier `essai.txt` avec le contenu du fichier `textebis.txt` crée dans l’exercice 3 (utiliser la commande `cat` et le symbole “>>”). Vérifier.

Copier le contenu du fichier `textebis.txt` dans `essai.txt` (ne pas utiliser la commande `cp` mais la commande `cat` et le symbole “>”). Quelle est la différence entre les deux opérations ?