

Linguistique de corpus

Outils et méthodes de traitement de corpus

Patrick Paroubek

LIMSI-CNRS
Dépt. CHM - Groupe LIR
Bât. 508 Université Paris XI, 91403 Orsay Cedex
pap@limsi.fr

mercredi 19 octobre 2016 / L3 - Cours 5

Les fichiers

Gestion des droits d'accès

Sur un système, il y a trois classes d'utilisateurs

- 1 l'utilisateur seul (**u**), la liste des utilisateurs sur un système UNIX est contenue dans le fichier **/etc/passwd**
- 2 un groupe d'utilisateurs (**g**), notion administrative définie par l'administrateur système de la machine en fonction des projets, un même utilisateur peut appartenir à plusieurs groupes, la liste des groupes sur un système UNIX est contenue dans le fichier **/etc/group**
- 3 tous les utilisateurs (**a** pour "all").

Les fichiers

Gestion des droits d'accès

- **ls -l** pour afficher les propriétés d'un fichier, produira par exemple l'affichage suivant :

-rw-r--r-- 1 pap users 114 Oct 19 17 :55 Makefile

qui indique les droits d'accès au fichier (**r**=read, **w**=write, **x**=execute, **-**=droit refusé) successivement pour l'utilisateur, le groupe et tous, ensuite vient le nombre de sous-repertoire ou de liens contenus (ici 1), l'id du propriétaire du fichier, le groupe auquel appartient le fichier, la taille du fichier (114), la date et l'heure de dernière modification du fichier, puis le nom du fichier.

Les fichier

Gestion des droits d'accès

- **chown pap foo.txt** attribue la propriété du fichier foo.txt à l'utilisateur pap.
- **chgrp users foo.txt** attribue la propriété du fichier foo.txt au groupe users.
- **newgrp projet2**, pour adopter l'identité du groupe projet2 (à condition que l'on soit listé comme membre de ce groupe dans /etc/group)

Programmer en shell

Structure d'un fichier de commandes

- expérimentation avec un programme shell.
- un programme shell est un fichier de texte contenant des commandes shell
- les commentaires sont du texte libre situées sur des lignes commençant par le caractère #
- en shell Unix, la première ligne d'un fichier de commandes est un commentaire qui indique quel interprète de commande doit être utilisé pour exécuter le programme, ici un des interprètes de commande shell les plus courant :
!/bin/bash
- si ce commentaire est absent, on suppose que le fichier est un fichier de commandes shell

Exemple, dans le fichier */home/tnmle/bin/affiche* on a le fichier texte suivant :

```
#!/bin/bash
```

```
ls -l
```

- En shell UNIX, les commandes peuvent être écrites soit une commande (avec ses arguments) par ligne,
- soit plusieurs commandes sur une même ligne, dans ce cas, elles sont séparées par des points virgules.
- Le nom de la commande que l'on ajoute au vocabulaire de l'interprète sera le nom du fichier qui contient les commandes qui sont exécutées pour réaliser la nouvelle commande, e.g. *affiche*

Une fois la sémantique de la nouvelle commande décrite en terme des commandes existantes dans un fichier texte, la nouvelle commande peut être exécutée sous 2 conditions :

- 1 les informations administrative de fichier autorisent son execution (mention 'x' pour le possesseur, le groupe ou tout le monde). Rappel, pour ajouter un droit d'exécution pour tout le monde au fichier *affiche*, il faut faire :

```
chmod a+x /home/tnml3/bin/affiche
```

- 2 l'interprète de commande sait où trouver le fichier *affiche*

Cette dernière condition peut être remplie si :

- le fichier affiche est mentionné avec son chemin lors de l'appel à l'interprète de commande, par ex.

```
/home/tnml3/bin/affiche
```

ou bien

```
cd /home/tnml3/bin; ./affiche
```