

Substitutions de noms, de commandes, de variables

Laurent Tichit

6 avril 2011

Plan

- 1 Shell : variables
- 2 Bash : caractères spéciaux

Shell : variables

Variables d'environnement

Le Shell possède un *langage de commandes*. Il est donc possible de définir des variables. De plus il existe déjà un certain nombre de *variables d'environnement* :

- GDMSESSION=LXDE
- HOME=/home/tichit
- LANG=fr_FR.UTF-8
- LOGNAME=tichit
- OLDPWD=/home/tichit
-
- PATH=/usr/local/sbin :/usr/local/bin :/usr/sbin :/usr/bin :/sbin :/bin
- PWD=/home/tichit/Desktop/sofia/unix
- SHELL=/bin/bash
- TERM=xterm
- USER=tichit
- ...

La liste des variables d'environnement est obtenue par la commande **printenv**.

Utilisation des variables

```
HOME=/home/tichit
```

- nom de variable : HOME
- valeur de la variable : \$HOME
- afficher la valeur de la variable : **echo \$HOME**
- créer et affecter une variable : **VAR=toto**
- supprimer une variable : **unset VAR**
Attention : pas d'espace autour de l'opérateur "="

Bash : caractères spéciaux

Substitution de noms de fichiers

Permet de pouvoir indiquer un ensemble de noms de fichiers en utilisant des expressions régulières.

- $u*v$ est remplacé par la liste des noms de fichiers situés dans le répertoire de travail si leur nom commence par u et fini par v . u et v peuvent être vides, auquel cas $*$ est remplacé par la liste des noms de tous les fichiers situés dans le répertoire de travail.
- $u?v$ idem que ci-dessus, sauf que "?" est remplacé par un SEUL caractère.
- $[\alpha\beta\gamma]$ est remplacé par un caractère de l'ensemble $\{\alpha, \beta, \gamma\}$.
- $[\alpha - \omega]$ est remplacé par un caractère de l'intervalle $[\alpha, \omega]$.
- $[\sim\alpha - \omega]$ est remplacé par un caractère qui n'est pas dans l'intervalle $[\alpha, \omega]$.
- $[\sim\alpha\beta\gamma]$ est remplacé par un caractère n'appartenant pas à l'ensemble $\{\alpha, \beta, \gamma\}$.

Séparation de mots

Les caractères ESPACE et TABULATION sont des caractères spéciaux. Ils permettent de découper une ligne de commande en mots distincts.

Si l'on veut considérer la chaîne "université technique" comme une chaîne et non deux mots distincts, il faudra utiliser :

VAR="université technique" car

VAR=université technique produira une erreur.

parenthèses

- Les parenthèses simples permettent principalement de grouper des commandes (et les exécuter dans un sous-shell).
- Les doubles parenthèses permettent de rentrer en mode arithmétique et faire du calcul en nombre entier :
En supposant que a , b , c sont des variables on peut effectuer ce genre d'operations :
 $((a=b+c/2))$
 $((a++))$

Substitutions de variables et commandes

- **\$VAR** : remplacé par le contenu de la variable VAR.
- **\$(cmd)** : remplacé par la chaîne de caractères résultat (sur la sortie standard) de la commande *cmd*. Ex :

```
MYLS=$(ls)
```

On peut aussi utiliser les anti-quotes (') pour la substitution de commandes. Ex :

```
MYLS='ls'
```

Caractères bloquant la substitution

- \ le caractère suivant est considéré comme un caractère normal.
- ' (quotes, apostrophes) idem que ci-dessus, empêche la substitution de TOUS les caractères spéciaux situés entre elles.
- " (doubles-quotes, guillemets) idem que ci-dessus, sauf pour la substitution de variables et de commandes (`$ $ () ' '`).