code de retour

- valeur à laquelle le processus père peut accèder
- 0: terminaison normale
- autre valeur: situation anormale
- commande1 && commande2:la commande2 est exécutée si la commande 1 réussit
- commande1 || commande2: la commande2 est exécutée si la commande 1 échoue
- exemple: commande test
- exemple: construction if/then/else/fi

Entrées-sorties

- Entrées-sorties
  - entrée standard: 0
  - sortie standard: 1
  - sortie d'erreur standard: 2

Héritage

- les descripteurs d'un processus enfant sont initialement les mêmes que ceux du processus père.
- si on ne les modifie pas, en sortie, les affichages s'entrelacent.
- il est possible de changer la valeur de entrée standard et des sorties standard et en erreur
  - pour les rediriger depuis/vers un fichier : <, >, >>, 2>, 2>>
  - pour les rediger depuis/vers un processus : |

redirection des sorties

- >: le contenu du fichier est remplacé par la sortie de la commande
- >>: la sortie s'ajoute à la fin du fichier
- exemples:
  - ls /etc > /tmp/foo.txt
  - cat ls
  - ls /usr/bin >> /tmp/foo.txt
  - du -sk /var/* > /tmp/bar.txt
  - date > /tmp/bar.txt (noter que le No d'inode est inchangé)

redirection de la sortie d'erreur standard

- commande 2> fichier
- commande 2>> fichier
- pratique pour isoler messages d'erreur et sortie
- /dev/null: le trou noir: pour éliminer les messages d'erreur
- du -sk /var/* 2> /dev/null

redirection simultanées

- on peut rediriger plusieurs descripteurs sur une même ligne de commande
- ls > /tmp/f1 2> f2
- les redirections sont traîtées de gauche à droite
- du -sk /var/* > /tmp/resultat 2> /tmp/erreur
- en cas de redirection simultanée avec cette syntase: impérativement vers des fichiers différents

redirection en entrée

- commande < fichier
- exemples:
  - mail petit < texte
  - wc -l < /etc/passwd

redirections avancées

- &1: valeur du descripteur de fichier 1
- &2: valeur du descripteur de fichier 2
- 1>&2: l'entrée standard est redirigée vers le même fichier que la sortie standard
- sert pour des redirection simultanées vers un même fichier
- Exemples:
  - ls > /tmp/test 2>&1 #OK
  - ls 2>&1 >/tmp/test #pas OK: stderr est toujours lié au terminal

<<

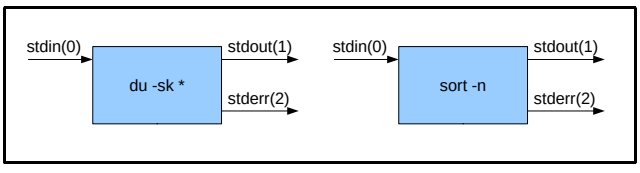
- dans une version ultérieure de ce document

fermeture d'un descripteur

- dans une version ultérieure de ce document

enchaînement de commandes

du -sk * |sort -n



L'ensemble forme une nouvelle commande

Filtres

- commande lisant leurs données sur l'entrée standard et envoyant leur sortie sur la sortie standard
- pratique pour les enchaînements de commandes
- philosophie unix: des commandes simples que l'on combine entre elles

commande test

- réalise des tests simple, le code de retour indique que le test est positif ou négatif
- test -d /var/tmp : teste si /var/tmp est un dossier
- test -x /bin/ls: teste si /bin/ls est un exécutable
- test 1 = 2: teste l'égalité de deux chaînes
- forme alternative :
  - test -d /var/tmp
  - [-d /var/tmp]

structure de contrôle if

- syntaxe:

```
if commande1
then
    commande2
[elif commande3
then commande4]
...
[else
    commande5]
fi
```

- si commande1 retourne 0, on exécute commande2 sinon, si commande3 retourne 0, on exécute commande4 ... sinon

sort

- selon SUSv3, sort a trois fonctions sur son entrée standard ou des fichiers textes constituées de lignes contenant un ou plusieurs champs :
  - trier les données (par défaut)
  - fusionner des fichiers triées en une sortie globale triée (option -m)
  - vérifier que les données sont triées (option -c)

Sort: options courantes:

- -t car_sep: permet de préciser le caractère qui sépare les champs du fichier
- -k : précise les champs sur lesquels portent le tri
- -o: indique un fichier de sortie (par défaut: sortie standard)
- -d: supprime les doublons
- -c : vérifie si un fichier est trié. Le résultat est indiqué uniquement par le code de retour: 0 si trié, 1 sinon.
- -m : fusionne des fichiers supposés déjà triés

uniq

- supprime les doublons d'une liste triée
- exemple: cat /tmp/test.txt |sort|uniq
- voir manuel pour les autres options
-

tail/head

- queue/tête d'un fichier

wc

- compte le nombre de lignes, de mots et de caractères
- wc -l : nombre de lignes
- wc -w : nombre de mots
- wc -c : nombres de caractères
- ls | wc -l : donne le nombre de fichier du dossier courant

grep, egrep & Co

- grep chaine: sélectionne les lignes qui contiennent la chaine
- grep petit /etc/passwd: sélectionne les lignes de /etc/passwd contenant la chaîne petit
- caractères spéciaux de la commande egrep:
  - ^: début de ligne
  - $: fin de ligne
- grep '^petit:' /etc/passwd: sélectionne les lignes commençant par petit:

cut

- sélectionner certaines colonnes

tr

more/less

commande find

- find permet de chercher récursivement les fichiers vérifiant une ou plusieurs conditions
- outre les expression simples, l'expression que doit vérifier un fichier peut être de la forme (par priorité décroissante) :
  - (expression)
  - ! expression
  - expression1 -a expression2 : ET logique
  - expression1 expression2: ET logique
  - expression1 -o expression2: OU logique

commande find

- expressions élémentaires à argument numérique:
  - +n: toutes les valeurs supérieures ou égales à n
  - -n: toutes les valeurs inférieures ou égales à n
  - n: n exactement
  - par la suite, partout où on verra un argument numérique n, on pourra utiliser +n, n ou -n
  - exemples:
    - -size 1024k: les fichiers de taille égale à 1024 Ko
    - -size -1024k: les fichiers de taille inférieure égale à 1024 Ko
    - -size +1024k: les fichiers de taille supérieur ou égale

commande find: quelques expressions élémentaires

- quelques expressions élémentaires:
  - -name motifProtégé: les fichiers vérifiant le motif
  - -size n: les fichiers de taille n
  - -mtime n, -ctime n, -atime n
  - -perm p avec p ayant la forme numérique ou symbolique des arguments de chmod
  - -type c avec c=b,c,d (dossier),l (lien symbolique),p,f (fichier ordinaire),s
  - -user u
  - -group g
  - -link n : nombre de liens physique sur le fichier
  - -print: provoque l'affichage des noms des fichiers vérifiant l'expression (par défaut sur le Gnu find)
  - ...

commande find: -exec

- -exec commande;
  - pour chaque fichier trouvé, la commande est exécutée.
  - si {}apparaît parmi les arguments de la commande, il est remplacé par le nom du fichier trouvé
- -exec commande arguments {} +
  - syntaxe POSIX/SUSv3 (standard mais pas disponible sur toutes les plateformes)
  - les noms des fichiers trouvés sont accumulés dans une liste l

commande find : exemples

- fichiers ordinaire nommés core de plus de 1024Ko
  - find -size +1024k -type f -name core -print
- fichier ordinaires de l'utilisateur petit ou fichiers ordinaires de taille supérieure à 1024 Ko et de nom core
  - find -type f \(-user petit -o \(-size +1024k -name core \) \) -print
- fichier dont le nom commence par C
  - find -name c* -print

commande find: -exec et les espaces (&Co)

- rm est une commande qui ne lit pas sur son entrée standard. 3 méthodes pour effacer un ensemble de fichiers sélectionnés par find :
  - on lance un rm par fichier (lourd)
    - find -name *.bak -exec rm -f {} \;
  - syntaxe POSIX: un seul rm global est lancé :
    - find -name *.bak -exec rm -f {} \+
  - une solution avec les options spécifiques de Gnu find:
    - find -name *.bak -print0 |xargs -0 rm -f
  - ne marche pas avec les noms de fichiers contenant des caractères à problème (espace, saut de ligne, ', etc.):

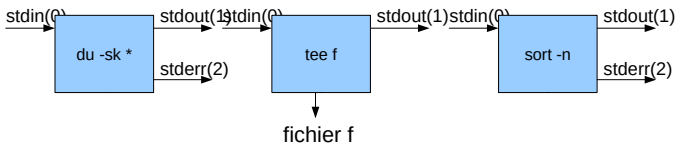
commande xargs

- xargs [options] commande
- xargs rassemble ce qu'elle reçoit sur son entrée standard dans une liste l et exécute « commande l »
- xargs s'utilise avec des commandes qui n'acceptent pas de données sur leur entrée standard
- Exemple:
  - grep -l perso * | xargs chmod 700

xargs : options utiles

- -p: prompt mode. une confirmation est demandée à l'utilisateur pour chaque invocation de la commande
- si le nombre ou la taille des arguments transmis via l'entrée standard est important, il est possible d'indiquer à xargs d'exécuter plusieurs fois la commande avec une liste limitée:
  - -n nombre: la commande est invoquée plusieurs fois et chaque invocation a au plus nombre arguments
  - -s taille: la commande est invoquée plusieurs

commande tee



- la commande tee envoie simultanément son entrée standard vers un fichier et vers sa sortie standard.
- options:
  - -a: ajoute au fichier
  - -i : ignore le signal

commandes qui ne lisent pas leur entrée standard

- ls, who, find
- chmod, cp, mv, rm, ln, mkdir
- date
- kill
- file, type
- echo

Scripts shell

- fichier texte contenant des commandes
- nouvelle commande

Editeur de texte

- vi : toujours présent
- emacs : couramment présent
- kedit: sous kde
- wxd: spécifique à notre parc