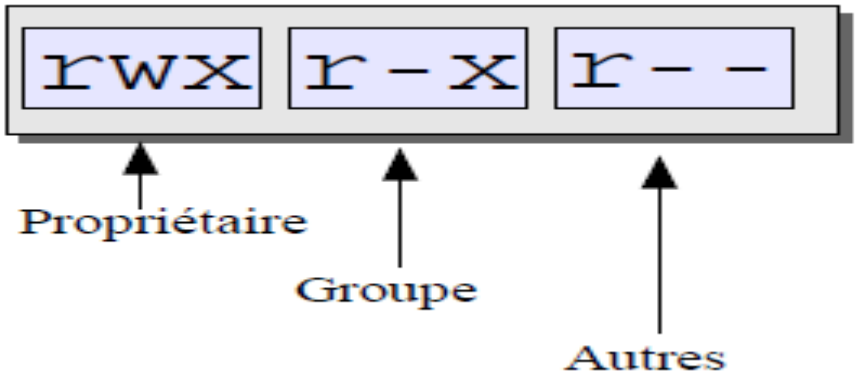
Les droits d'accès

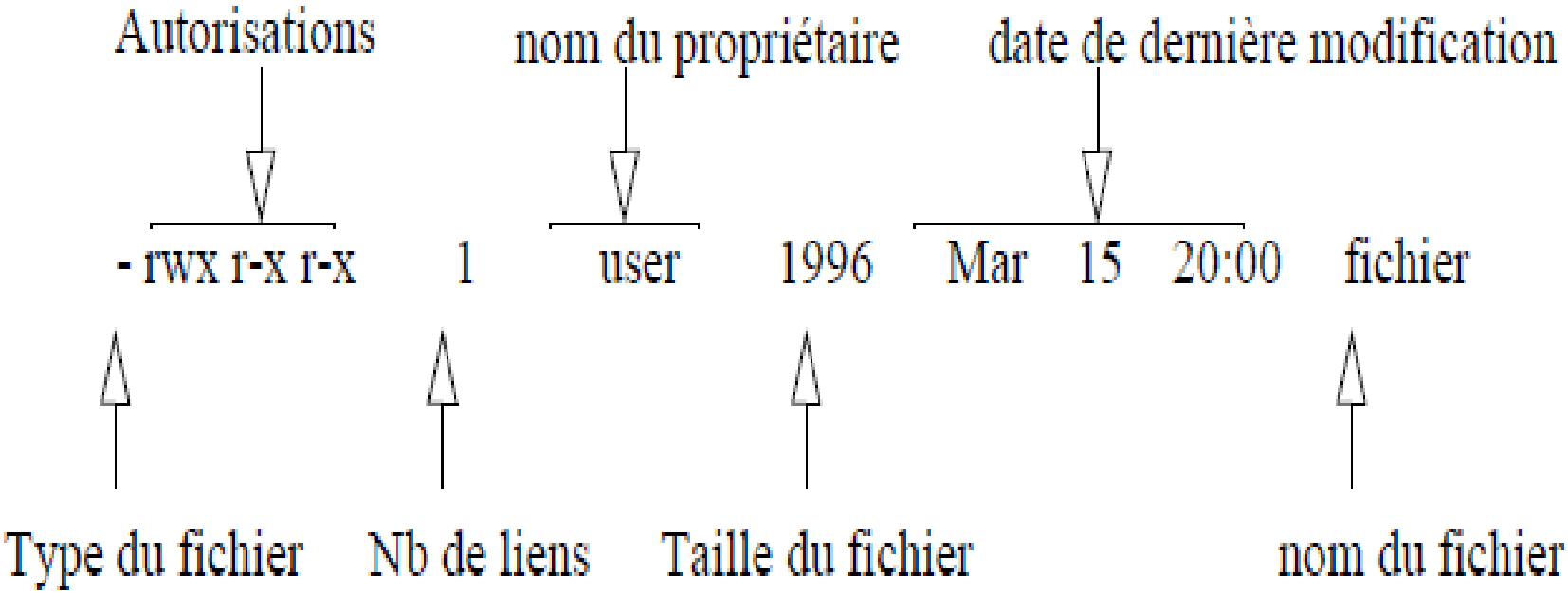


Sortie de la commande ls -l :

d	rwxr-xr-x	1	oracle	dba	466	Feb 8 2001	oracle
-	rwxrwxrwx	1	oracle	dba	14536	Feb 8 2001	output.log
-	rwxr-x---	1	oracle	dba	1456	Feb 8 2001	launch.sh

- Sur la première ligne du tableau, le répertoire oracle appartient à *l'utilisateur oracle* et au groupe dba, et possède les droits rwxr-xr-x.

Les droits d'accès



Les droits d'accès

La première information concerne le type du fichier.

Cet indicateur peut prendre les valeurs suivantes:

- **-** pour un fichier ordinaire,
- **d** pour un répertoire,
- **b** pour un fichier spécial de type bloc (périphériques...),
- **c** pour un fichier spécial de type caractère (disque, streamer...),
- **l** pour un lien symbolique,
- **s** pour une socket.

Les droits d'accès

- La seconde concerne les droits (utilisateur-propriétaire, groupe de travail, autres utilisateurs),
- Ensuite le nombre de liens sur le fichier (ou le répertoire),
- Puis le propriétaire effectif,
- la taille,
- la date de dernière modification
- Et le nom du fichier.

Les droits d'accès

Droit	*Signification*
	Général
r	Readable (lecture)
w	Writable (écriture)
x	Executable (exécutable comme programme)
	Fichier normal
r	Le contenu du fichier peut être lu, chargé en mémoire, visualisé, recopié.
w	Le contenu du fichier peut être modifié, on peut écrire dedans. La suppression n'est pas forcément liée à ce droit (voir droits sur répertoire).
x	Le fichier peut être exécuté depuis la ligne de commande, s'il s'agit soit d'un programme binaire (compilé), soit d'un script (shell, perl, ...)

Les droits d'accès

Répertoire	
r	Les éléments du répertoire (catalogue) sont accessibles en lecture. Sans cette autorisation, le ls et les critères de filtres sur le répertoire et son contenu ne sont pas possibles. **Ce droit ne suffit pas pour entrer dans le catalogue**.
w	Les éléments du répertoire (catalogue) sont modifiables et il est possible de créer, renommer et supprimer des fichiers dans ce répertoire. On voit donc que c'est ce droit qui contrôle l'autorisation de suppression d'un fichier **même si on est pas propriétaire des fichiers du répertoire**.
x	Le catalogue peut être accédé par cd et listé. Sans cette autorisation il est impossible d'accéder et d'agir sur son contenu qui devient verrouillé. Avec uniquement ce droit les fichiers et répertoires inclus dans celui-ci peuvent être accédés mais il faut alors obligatoirement connaître leur nom.

Les droits d'accès

Ainsi pour un fichier :

rwx	r-x	r--
Droits de l'utilisateur, en lecture, écriture et exécution	Droits pour les membres du groupe et lecture et exécution	Droits pour le reste du monde en lecture uniquement

Les droits d'accès

Modification des droits

- Lors de sa création, un fichier ou un répertoire dispose de droits par défaut.
- On utilise la commande **chmod** (**change mode**) pour modifier les droits sur un fichier ou un répertoire.
- Il existe de méthodes pour modifier ces droits :
  - par la forme symbolique
  - par la base 8.
- Seul le propriétaire d'un fichier peut en modifier les droits (sauf l'administrateur système).
- Le chmod sur un lien symbolique est possible comme sur tout autre fichier, mais cela ne modifie pas les droits du lien par lui-même mais les droits du fichier pointé.

Les droits d'accès

Par symboles

- La syntaxe est la suivante :
- chmod modifications Fic1 [Fic2...]
- S'il faut modifier les droits de l'utilisateur, on utilisera le caractère « **u** ».
- Pour les droits du groupe, le caractère «**g**», pour le reste du monde le caractère «**o**», pour tous le caractère «**a**».
- Pour ajouter des droits, on utilise le caractère « **+** », pour en retirer le caractère « **-** », et pour ne pas tenir compte des paramètres précédents le caractère « **=** ».
- Enfin, le droit d'accès par lui-même : « **r** », « **w** » ou « **x** ».

Les droits d'accès

On peut séparer les modifications par un espace, et cumuler plusieurs droits dans une même commande.

```
$ ls –l
-rw-r--r-- 1 oracle system 0 Aug 12 11:05 fic1
-rw-r--r-- 1 oracle system 0 Aug 12 11:05 fic2
-rw-r--r-- 1 oracle system 0 Aug 12 11:05 fic3
$ chmod g+w fic1
$ ls -l fic1
-rw-rw-r-- 1 oracle system 0 Aug 12 11:05 fic1
$ chmod u=rwx,g=x,o=rw fic2
$ ls -l fic2
-rwx--xrw- 1 oracle system 0 Aug 12 11:05 fic2
$ chmod o-r fic3
$ ls -l fic3
-rw-r----- 1 oracle system 0 Aug 12 11:05 fic3
```

Les droits d'accès

Par base 8

- La syntaxe est identique à celle des symboles. A chaque droit correspond une valeur octale c'est à dire de zéro (0) à sept (7), positionnelle et cumulable.

Propriétaire			*Groupe*			*Reste du monde*		
r	w	x	r	w	x	r	w	x
400	200	100	40	20	10	4	2	1

- Pour obtenir le droit final il suffit d'additionner les valeurs.

Les droits d'accès

Par exemple si on veut rwxrw-rw-

➔ alors on fera 400 + 200 + 100 + 40 + 20 + 4 + 2 = 766

Pour rw-r--r--

➔ 400 + 200 + 40 + 4 = 644.

```
$ chmod 766 fic1
$ ls -l fic1
-rwxr-xr-x 1 oracle system 0 Aug 12 11:05 fic1
$ chmod 644 fic2
$ ls -l fic2
-rw-r--r-- 1 oracle system 0 Aug 12 11:05 fic2
```

Les droits d'accès

Masque des droits

- Lors de la création d'un fichier ou d'un répertoire et qu'on regarde ensuite leurs droits, on obtient généralement rw-r—r-- (644) pour un fichier et rwxr-xr-x (755) pour un répertoire.
- Ces valeurs sont contrôlées par un masque, lui-même modifiable par la commande **umask.**
- La commande prend comme paramètre une valeur octale qui sera soustraite aux droits d'accès maximum.
- Par défaut tous les fichiers sont créés avec les droits 666 (rw-rw-rw) et les répertoires avec les droits 777 (rwxrwxrwx), puis le masque est appliqué.

Les droits d'accès

- Sur la plupart des Unix, le masque par défaut est 022, soit ----w--w-. Pour obtenir cette valeur, on tape umask sans paramètre.
- Pour un fichier :
  - Maximum rw-rw-rw- (666)
  - Retirer ----w--w- (022)
  - Reste rw-r--r-- (644)
- Pour un répertoire :
  - Maximum rwxrwxrwx (777)
  - Retirer ----w--w- (022)
  - Reste rwxr-xr-x (755)

Les droits d'accès

- **Pour un fichier:**
- Si vous tapez umask 022, vous partez des droits maximum 666 et vous retranchez 022, on obtient donc 644,

➔ par défaut les fichiers auront comme droit 644 (-rw-r-r--).

- Si vous tapez umask 244, vous partez des droits maximum 666 et vous retranchez 244, on obtient donc 422,

➔ par défaut les fichiers auront comme droit 422 (-rw--w--w-).

- **Pour un répertoire :**
- Si vous tapez umask 022, vous partez des droits maximum 777 et vous retranchez 022, on obtient donc 755,

➔ par défaut les fichiers auront comme droit 644 (-rwxr-xr-x).

Les droits d'accès

- Si vous tapez umask 244, vous partez des droits maximum 777 et vous retranchez 244, on obtient donc 533,

➔ par défaut les fichiers auront comme droit 422 (-rwx-wx-wx).

umask n'est utilisatable que si on est propriétaire du fichier.

Les droits d'accès

- Pour un fichier :
- Maximum rw- rw- rw- (666)
- Retirer --- r-x rw- (056)
- Reste rw- -w- --- (620) **et PAS 610 !**

ATTENTION : la calcul des droits définitifs (effectifs) n'est pas une simple soustraction de valeurs octales ! Le masque retire des droits mais n'en ajoute pas.

Les droits d'accès

- ***Changement de propriétaire et de groupe***
- Il est possible de changer le propriétaire et le groupe d'un fichier à l'aide des commandes **chown** (**change owner**) et **chgrp (change group).**
  - chown utilisateur fic1 [Fic2…]
  - chgrp groupe fic1 [Fic2…]

- Sur les UNIX récents seul **root** peut utiliser **chown**. La commande **chgrp** peut être utilisée par n'importe qui à condition que cet utilisateur fasse aussi partie du nouveau groupe**.**

Les droits d'accès

- En précisant le nom d'utilisateur (ou de groupe), le système vérifie d'abord leur existence.
- On peut préciser un UID ou un GID, dans ce cas le système n'effectuera pas de vérification.
- Pour les deux commandes on peut préciser l'option **-R**, dans ce cas les droits seront changés de manière récursive.
- Les droits précédents et l'emplacement du fichier ne sont pas modifiés.

Les droits d'accès

- Sous certains Unix il est possible de modifier en une seule commande à la fois le propriétaire et le groupe.

  - chown utilisateur[:groupe] fic1 [fic2...]
  - chown utilisateur[.groupe] fic1 [fic2...]
  - $ chgrp dba fic1
  - $ ls -l fic1
  - -rwxr-xr-x 1 oracle dba 0 Aug 12 11:05 fic1

La commande find

- La commande find permet de retrouver des fichiers à partir de certains critères.
- La syntaxe est la suivante :

Find <Répertoire de Recherche> <Critère de Recherche>

- Les critères de recherche sont les suivants :
- name recherche sur le nom du fichier,
- perm recherche sur les droits d'accès du fichier,
- links recherche sur le nombre de liens du fichier,
- user recherche sur le propriétaire du fichier,

La commande find

- group recherche sur le groupe auquel appartient le fichier,
- type recherche sur le type (d=répertoire, c = caractère, f = fichier normal),
- size recherche sur la taille du fichier en nombre de blocs (1 bloc= 512octets),
- atime recherche par date de dernier accès en lecture du fichier,
- mtime recherche par date de dernière modification du fichier,
- ctime recherche par date de création du fichier.

La commande find

Critères

- Les options permettent de définir les critères de recherche. Elles peuvent être combinées et groupées entre elles, sous forme et ET et de OU logiques.

-name, -iname

- L'option -name permet une sélection par noms de fichiers.
- Il est possible d'utiliser les critères de recherches (caractères spéciaux) déjà vus.
- Le critère est idéalement placé en guillemets.
- Avec -iname, on n'effectue pas de différences entre les minuscules et les majuscules.

La commande find

-type

- L'option -type permet une sélection par type de fichier.
- Nous avons vu au début du cours qu'outre les exécutables, liens, répertoires et fichiers simples étaient présent d'autres types de fichiers.

Code fichier	*Type de fichier*
b	Fichier spécial en mode bloc
c	Fichier spécial en mode caractère
d	Répertoire (directory)
f	fichier normal
l	lien symbolique
p	tube nommé (pipe)
s	Socket (sur certains Unix)

La commande find

Remarque: La commande **find** doit être utilisé avec l'option **-print**. Sans l'utilisation de cette option, même en cas de réussite dans la recherche, find n'affiche rien à la sortie standard (l'écran, plus précisément le shell).

```
$ find . -name "re*" -type d -print
./rep1
./rep2
```

- La commande find est récursive, c'est à dire où que vous tapiez, il va aller scruter dans les répertoires, et les sous répertoires qu'il contient, et ainsi de suite.

La commande find

-user et -group, -uid et -gid

- Les options **-user et -group** permettent une recherche sur le propriétaire et le groupe d'appartenance des fichiers.
- Il est possible de préciser le nom (utilisateur, groupe) ou l'ID (UID, GID).
- On peut aussi préciser **-uid et -gid avec les numéros UID et GID.**

```
$ find . -type f -user oracle -group dba -print
./fic1
./fic3
```

La commande find

Recherche par nom de fichier

- Pour chercher un fichier dont le nom contient la chaîne de caractères toto à partir du répertoire /usr,
- vous devez tapez : `find /usr -name toto -print`
- En cas de réussite, si le(s) fichier(s) existe(nt), vous aurez comme sortie `toto`
- En cas d'échec, vous n'avez rien.
- Pour rechercher tous les fichiers se terminant par .c dans le répertoire /usr, vous taperez :

```
find /usr -name " *.c " -print
```

- Vous obtenez toute la liste des fichiers se terminant par .c sous les répertoires contenus dans /usr (et dans /usr lui même).

La commande find

Recherche suivant la date de dernière modification

- Pour connaître les derniers fichiers modifiés dans les 3 derniers jours dans toute l'arborescence (/),

  vous devez taper :

```
find / -mtime 3 -print
```

Recherche suivant la taille

- Pour connaître dans toute l'arborescence, les fichiers dont la taille dépasse 1Mo (2000 blocs de 512Ko), vous devez taper :

```
find / -size 2000 -print
```

La commande find

Recherche combinée

- Vous pouvez chercher dans toute l'arborescence, les fichiers ordinaires appartenant à olivier, dont la permission est fixée à 755, on obtient :

```
find / -type f -user olivier -perm 755 -print
```

Redirection des messages d'erreur

- Vous vous rendrez compte assez rapidement qu'en tant que simple utilisateur, vous n'avez pas forcément le droit d'accès à un certain nombre de répertoires, par conséquent, la commande find peut générer beaucoup de messages d'erreur (du genre permission denied), qui pourraient noyer l'information utile.

La commande find

- Pour éviter ceci, vous pouvez rediriger les messages d'erreur dans un fichier poubelle (comme /dev/null), les messages d'erreur sont alors perdus (rien ne vous empêche de les sauvegarder dans un fichier, mais ça n'a aucune utilité avec la commande find).

- $ find / -name toto2 > /dev/null

  ➔ recherche d'un fichier et envoi des erreurs retournées dans le fichier null (poubelle)

La commande find

- L'option -print est une commande que l'on passe à find pour afficher les résultats à la sortie standard. En dehors de print, on dispose de l'option -exec.
- **find** couplé avec **exec** permet d'exécuter une commande sur les fichiers trouvés d'après les critères de recherche fixés.
- Cette option attend comme argument une commande, celle ci doit être suivi de {}\ ;.

La commande find

- Exemple recherche des fichiers ayant pour nom core, suivi de l'effacement de ces fichiers.

```
find . -name core -exec rm {}\   ;
```

- Tous les fichiers ayant pour nom core seront détruits, pour avoir une demande de confirmation avant l'exécution de rm, vous pouvez taper :

```
find . -name core -ok rm {}\   ;
```

Bon courage