

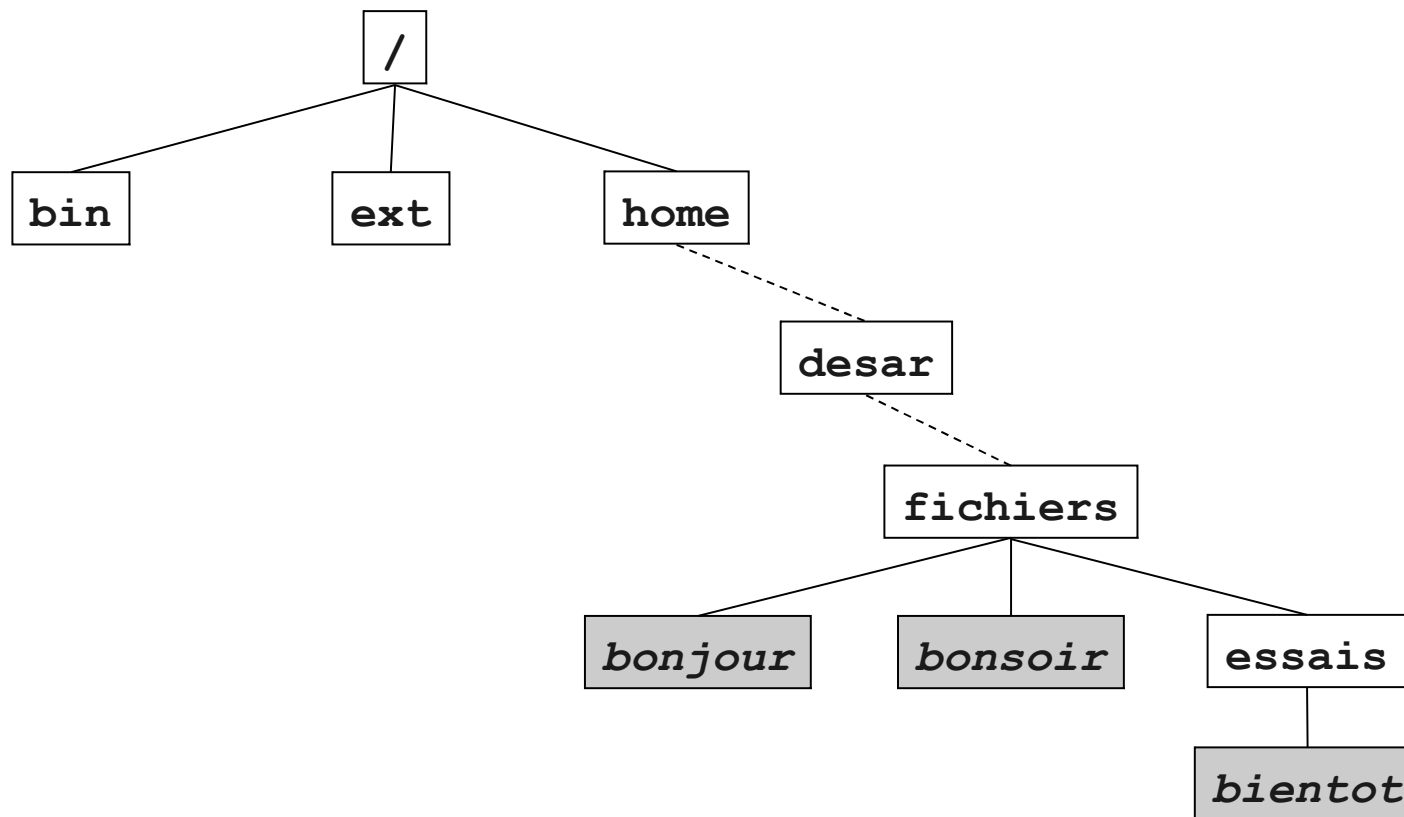
CH.2 SYSTÈMES DE FICHIERS SOUS UNIX

- 2.1 L'arborescence des fichiers
- 2.2 Les commandes utiles
- 2.3 Les droits
- 2.4 La structure : les i-noeuds
- 2.5 Le montage

2.1 L'arborescence des fichiers

Fichiers organisés de manière hiérarchisée,
regroupés dans des répertoires (arborescence).

Exemple partiel de hiérarchie de fichiers :



Gestion des fichiers : en général, via l'interface graphique.

- Navigation dans la hiérarchie
- Création de répertoires
- Création de fichiers
- Destruction des répertoires et des fichiers
- Duplication de répertoires et de fichiers
- Renommage de répertoires et de fichiers
- Déplacement des répertoires et des fichiers
- Création de liens symboliques (raccourcis)
- Gestion des droits
- Gestion du propriétaire et du groupe

Opérations obtenues par copier-coller, par clic-droit et navigation dans le menu, en faisant des sauvegardes dans les éditeurs de textes, en examinant les propriétés, ...

2.2 Les commandes utiles

Commandes en mode console (en ouvrant une fenêtre de shell)

- Localisation dans la hiérarchie : **pwd**
- Vision du contenu d'un répertoire : **ls** *<nom>*
- Navigation : **cd** *<destination>*
- Création de répertoire : **mkdir** *<nom>*
- Destruction d'accès à un fichier : **rm** *<nom>*
- Destruction d'accès à un répertoire : **rmdir** *<nom>*
- Duplication : **cp** *<nom d'origine>* *<destination>*
- Renommage et déplacement : **mv** *<nom d'origine>* *<destination>*
- Création de lien symbolique : **ln -s** *<nom d'origine>* *<destination>*
- Gestion des droits : **chmod**
- Gestion du propriétaire et du groupe : **chown** (et **chgrp**)

Exemple de session :

```
[desar@riesling fichiers]$ pwd
```

```
/home/institut/desar/fichiers
```

```
[desar@riesling fichiers]$ ls --help
```

```
Usage: ls [OPTION]... [FICHIER]...
```

Afficher les informations au sujet des FICHIERS (du répertoire courant par défaut). Trier les entrées alphabétiquement si aucune des options `-cftuSUX` ou `--sort` n'est utilisée.

.....

- `-a, --all` afficher les noms cachés débutant par `.`
- `-i, --inode` afficher le numéro d'index de chaque fichier
- `-l` utiliser le format long d'affichage
- `-R, --recursive` afficher les sous-répertoires récursivement

```
[desar@riesling fichiers]$ ls
```

```
bonjour bonsoir essais
```

```
[desar@riesling fichiers]$ ls -l
```

```
total 3
```

```
-rw-r--r--    1 desar    institut    10 mar 20 15:57 bonjour
-rw-r--r--    1 desar    institut    10 mar 20 15:58 bonsoir
drwxr-xr-x    2 desar    institut  1024 mar 20 15:58 essais
```

```
[desar@riesling fichiers]$ ls -lR
```

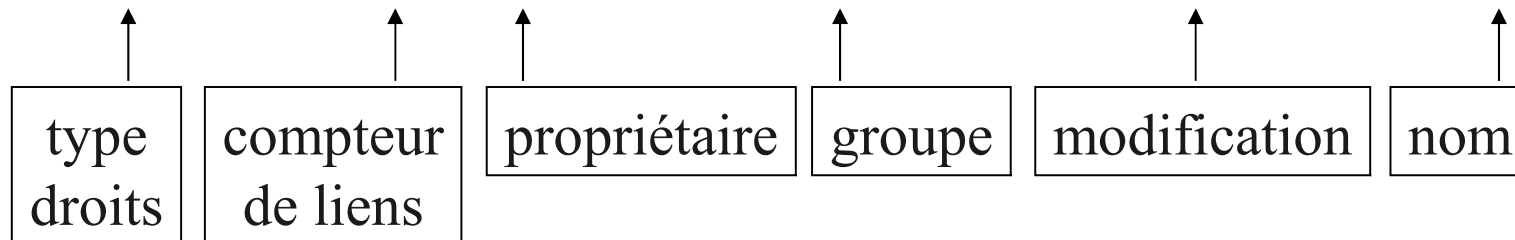
```
total 3
```

```
-rw-r--r--    1 desar    institut    10 mar 20 15:57 bonjour
-rw-r--r--    1 desar    institut    10 mar 20 15:58 bonsoir
drwxr-xr-x    2 desar    institut  1024 mar 20 15:58 essais
```

```
essais:
```

```
total 1
```

```
-rw-r--r--    1 desar    institut    12 mar 20 15:58 bientot
```



La navigation se fait avec la commande `cd`. L'utilisateur est dans un répertoire. Des pseudonymes désignent divers répertoires :

- Celui où on se trouve est `.`
- Son père est `..`
- Son répertoire de base est `~`
- La racine est `/`

Exemple de navigation :

```
[desar@riesling fichiers]$ cd essais
[desar@riesling essais]$ pwd
/home/institut/desar/fichiers/essais
[desar@riesling essais]$ cd ../..
[desar@riesling desar]$ pwd
/home/institut/desar
[desar@riesling desar]$ cd /
[desar@riesling /]$ cd ~
[desar@riesling desar]$ pwd
/home/institut/desar
```

Exemple de visualisation du contenu de fichiers-texte : cat et more

```
[desar@riesling fichiers]$ cat bonjour
Bonjour !
[desar@riesling fichiers]$ more bonsoir
Bonsoir !
```

Exemple de créations, liens, renommage, destruction

cp recopie le fichier bonjour en bonsoir ; l'information est dupliquée.
Le fichier bonjour existe encore.

```
[desar@riesling fichiers]$ cp bonjour salut
[desar@riesling fichiers]$ cat salut
Bonjour !
```

ln -s crée un raccourci ou alias.

```
[desar@riesling fichiers]$ ln -s bonjour coucou
[desar@riesling fichiers]$ cat coucou
Bonjour !
```

On peut utiliser un répertoire comme destination

Crée un lien bonsoir dans le répertoire essais.

```
[desar@riesling fichiers]$ ln -s bonsoir essais
```

mv renomme bonsoir en bonsoir_n.

```
[desar@riesling fichiers]$ mv bonsoir bonsoir_n
```

Déplace bonsoir en dans le répertoire essais.

```
[desar@riesling fichiers]$ mv bonsoir essais
```

rm efface l'accès à bonsoir.

```
[desar@riesling fichiers]$ rm -f bonsoir
```

L'option -f fait qu'aucune confirmation n'est demandée (attention !)

Pour les répertoires, on utilise rmdir (vider le répertoire auparavant) ou on efface récursivement avec rm -r (voire rm -rf : grande prudence !)

On peut utiliser des caractères passe-partout (*, ?)

2.3 Les droits

Considérons la situation suivante :

```
[desar@riesling fichiers]$ ls -l
total 4
-rw-r--r--   3  desar   institut   10 mar 20 15:58 adieu
-rw-r--r--   1  desar   institut   10 mar 20 16:18 bonjour
-rw-r--r--   3  desar   institut   10 mar 20 15:58 bonsoir
drwxr-xr-x   2  desar   institut  1024 mar 20 16:18 essais
```

Le premier caractère identifie le type de fichier, puis viennent deux groupes de trois rwx, certains remplacés par un trait.

Le premier se réfère au propriétaire u, le second au groupe g, le troisième aux autres o.

Pour un fichier, r autorise la lecture, w l'écriture et x l'exécution.

Pour un répertoire, r autorise la lecture de la liste des fichiers contenus, w la modification de cette liste et x l'accès aux fichiers contenus.

Le propriétaire et *lui seul* (et le superutilisateur) peut modifier ces droits par la commande `chmod`.

```
[desar@riesling fichiers]$ chmod --help
Usage: chmod [OPTION]... MODE[,MODE]... FICHIER...
       ou: chmod [OPTION]... MODE_OCTAL FICHIER...
```

On peut utiliser comme option soit des modes par symboles, soit en octal.

En symbolique, `chmod [ugoa][+ -=][rwx] <nom>`

Par exemple la commande `chmod a-w <nom>` protège en écriture pour tous, `chmod u+x <nom>` rend un fichier exécutable par le propriétaire.

Rien n'est changé pour les utilisateurs non mentionnés

En mode octal, trois groupes correspondent successivement au propriétaire, à son groupe et aux autres. Les bits correspondent, du fort au faible, à la lecture, à l'écriture et à l'exécution. Par exemple, $6=1*4+1*2$ correspond à `rw-`.

Par exemple, `chmod 644 <nom>` donne l'état par défaut pour un fichier.

Pour ce qui est des répertoires, les droits par défaut sont 755. Le propriétaire peut tout faire. Les autres peuvent lire les fichiers contenus.

La commande `chmod go-x <nom de répertoire>` (mode 744) empêche les tiers d'accéder aux fichiers contenus dans un répertoire ; ils peuvent toujours en consulter la liste.

La commande `chmod go-r <nom de répertoire>` (mode 711) empêche les tiers de consulter la liste des fichiers contenus dans un répertoire ; mais ils peuvent y accéder, s'ils en connaissent l'adresse (car le droit x est activé).

Enfin, la commande `chmod go-rwx <nom de répertoire>` (mode 700) assure la confidentialité complète vis-à-vis des tiers.

Attention : si un fichier est interdit en écriture, son contenu ne peut être modifié, mais le fichier peut être renommé ou son accès supprimé.

L'utilisateur peut seul changer ces droits. Si vous copiez un fichier chez un autre utilisateur, ce dernier ne pourra en général rien faire sur ce fichier. Il faut penser à modifier le propriétaire et le groupe par la commande `chown <propriétaire:groupe> <nom>`

2.4 La structure : les i-noeuds

On peut pratiquer des fonctions de base d'UNIX sans connaître plus de détails sur l'organisation des fichiers.

Une certaine compréhension permet néanmoins une plus grande efficacité de programmation et explique des particularités des commandes.

Les fichiers contiennent des *données*, stockées sur un *support physique* (disque). Ces données sont accessibles à travers des *noms* de fichiers.

Pour permettre une manipulation plus facile des données et des noms de fichiers, ainsi qu'une structure uniforme du système de fichiers, les données sont repérées de façon interne non pas par un nom, mais par un numéro, le *i-noeud*, indépendant de l'adresse réelle des données sur le disque.

Ce i-noeud assure une interface entre l'utilisateur et le système physique de fichiers.

L'option *i* de la commande *ls* permet de consulter les numéros de i-noeuds.

Exemple d'affichage des i-noeuds :

Le nombre qui figure en tête est l'index ou i-nœud.

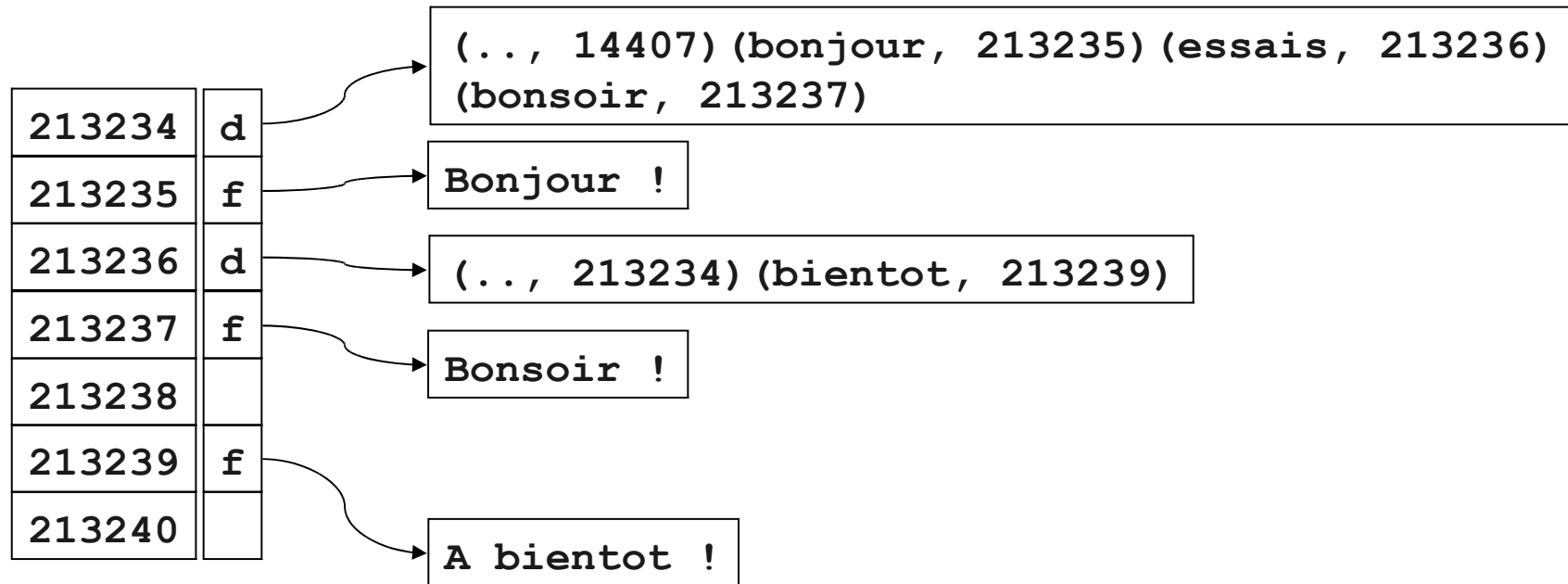
```
[desar@riesling fichiers]$ ls -liR
total 3
 213235 -rw-r--r--    1 desar    institut    10 mar 20 15:57 bonjour
 213237 -rw-r--r--    1 desar    institut    10 mar 20 15:58 bonsoir
 213236 drwxr-xr-x    2 desar    institut   1024 mar 20 15:58 essais

essais:
total 1
 213239 -rw-r--r--    1 desar    institut    12 mar 20 15:58 bientot
```

La structure est la suivante :

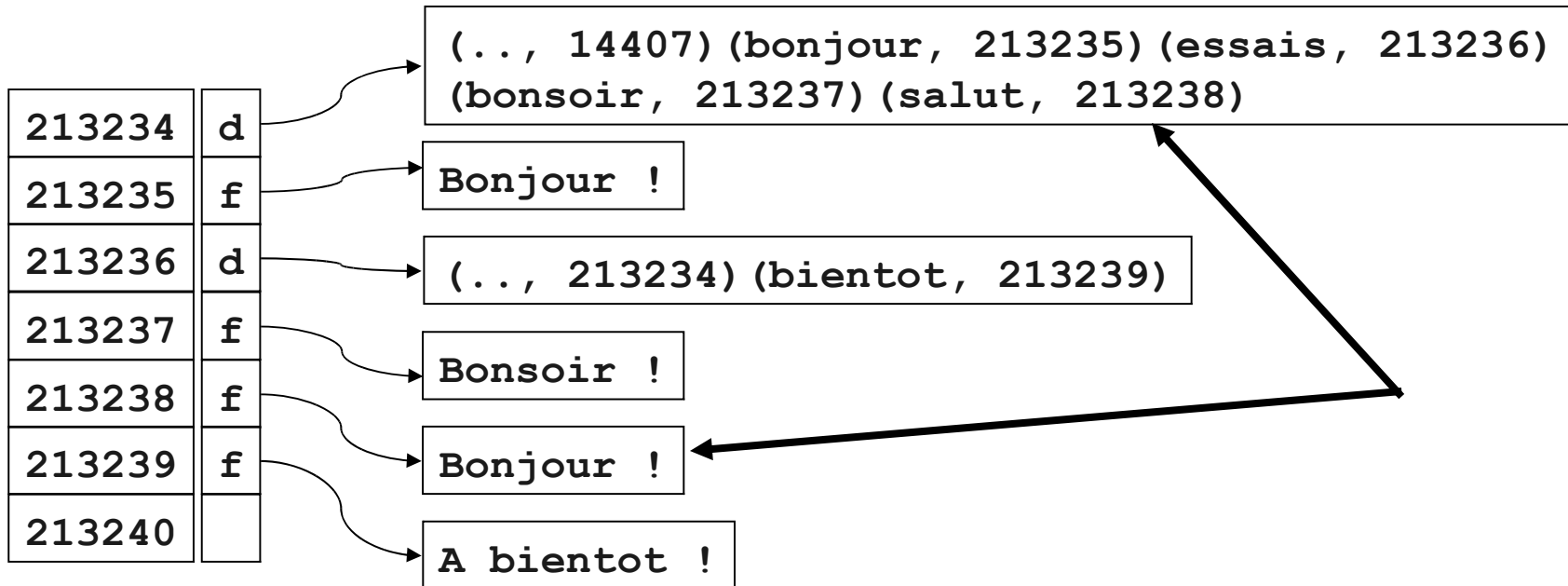
- Une table des i-noeuds contient pour chaque numéro un pointeur vers l'information stockée ; elle contient aussi l'information sur le type de fichier, les droits, le propriétaire et le groupe, le compteur de liens, la date de modification, ...
- L'information relative aux répertoires contient, pour chacun des éléments qu'il contient, le numéro du i-noeud correspondant, et celui de son père.

Exemple de table de i-noeuds :



De cette manière, il est possible de changer l'information sans changer le nom, ou de changer le nom sans modifier l'information, de créer, de supprimer, de renommer, de lier fichiers et répertoires.

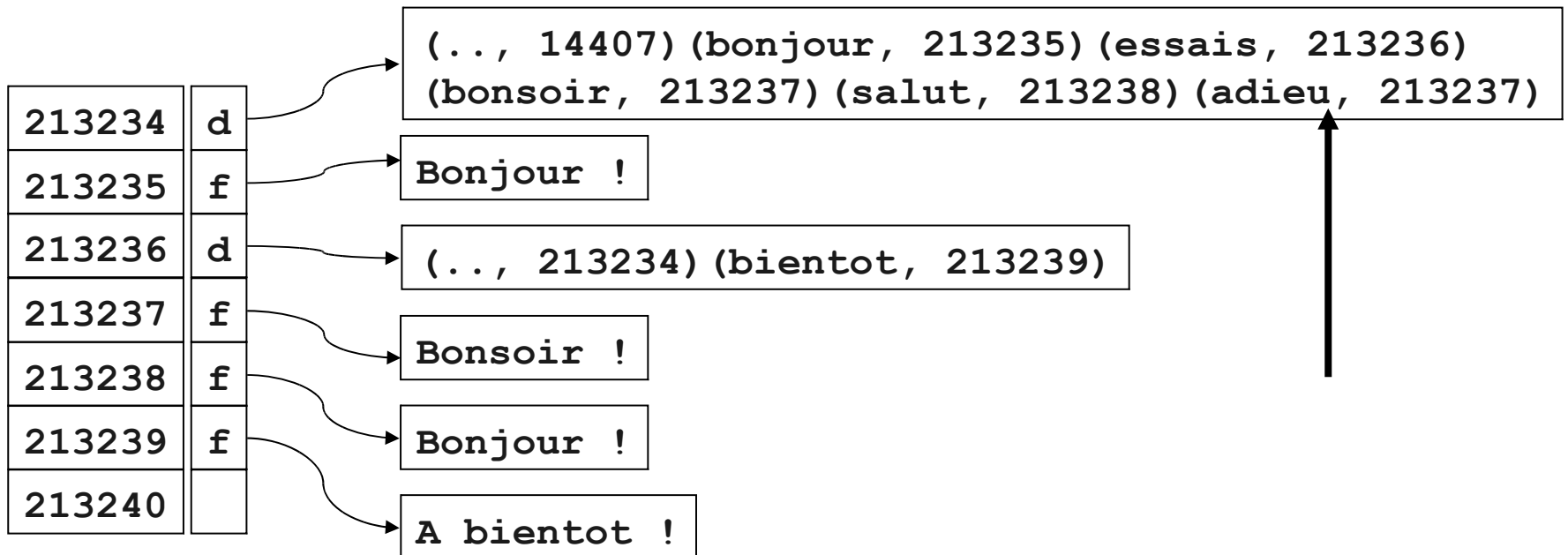
```
[desar@riesling fichiers]$ cp bonjour salut
```



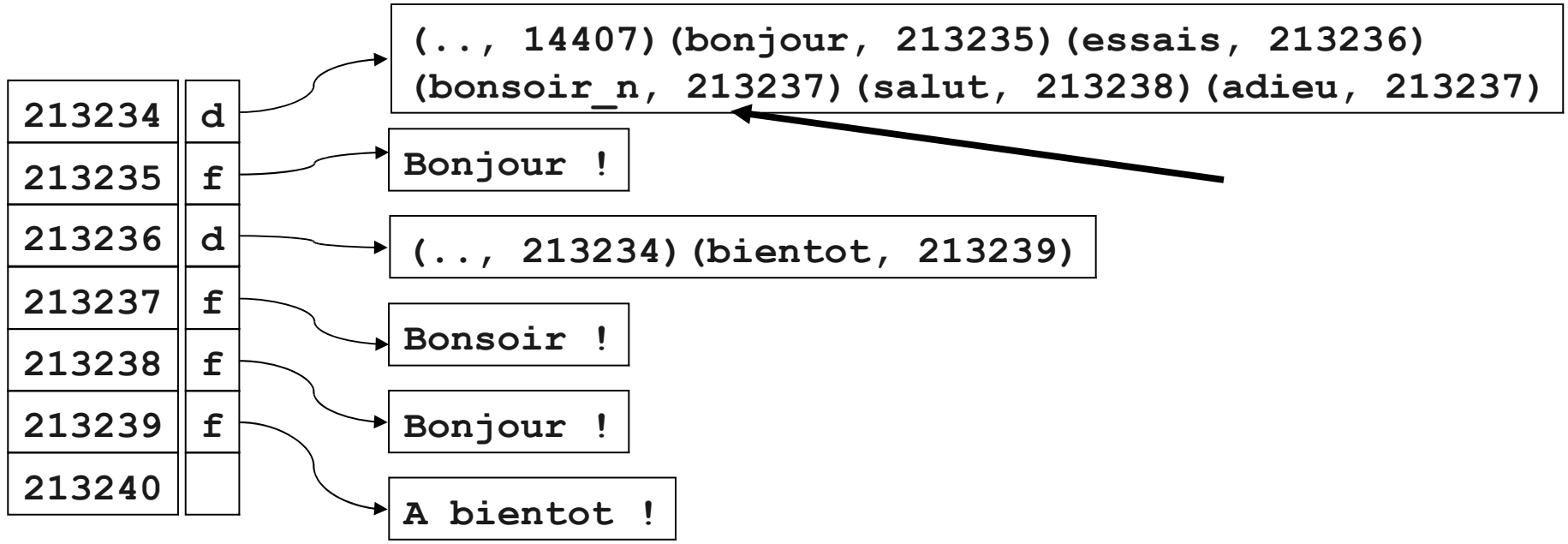
Le contenu est dupliqué.

On peut créer un nouvel accès vers la même information en utilisant la commande `ln` (sans l'option `-s` qui crée un lien symbolique).

```
[desar@riesling fichiers]$ ln bonsoir adieu
```



```
[desar@riesling fichiers]$ mv bonsoir bonsoir_n
```



```
[desar@riesling fichiers]$ ls -liR
```

```
total 5
```

```
213237 -rw-r--r--  2 desar  institut    10 mar 20 15:58 adieu
213235 -rw-r--r--  1 desar  institut    10 mar 20 15:57 bonjour
213237 -rw-r--r--  2 desar  institut    10 mar 20 15:58 bonsoir_n
213236 drwxr-xr-x  2 desar  institut  1024 mar 20 16:18 essais
213238 -rw-r--r--  1 desar  institut    10 mar 20 16:18 salut
```

```
essais:
```

```
total 1
```

```
213239 -rw-r--r--  1 desar  institut    12 mar 20 15:58 bientot
```

La commande `rm` permet d'effacer non pas un fichier mais l'accès à ce fichier :

- Le compteur de liens du i-noeud est diminué d'une unité ;
- Si ce compteur vaut 0, le i-noeud et l'espace mémoire vers lequel il pointe sont récupérés dès que possible.

```
[desar@riesling fichiers]$ rm --help
```

```
Usage: rm [OPTION]... FICHIER
```

```
Enlever (unlink) les FICHIER(s).
```

Par exemple, dans la situation précédente,

```
[desar@riesling fichiers]$ rm -f adieu
```

```
[desar@riesling fichiers]$ ls -liR
```

```
total 4
```

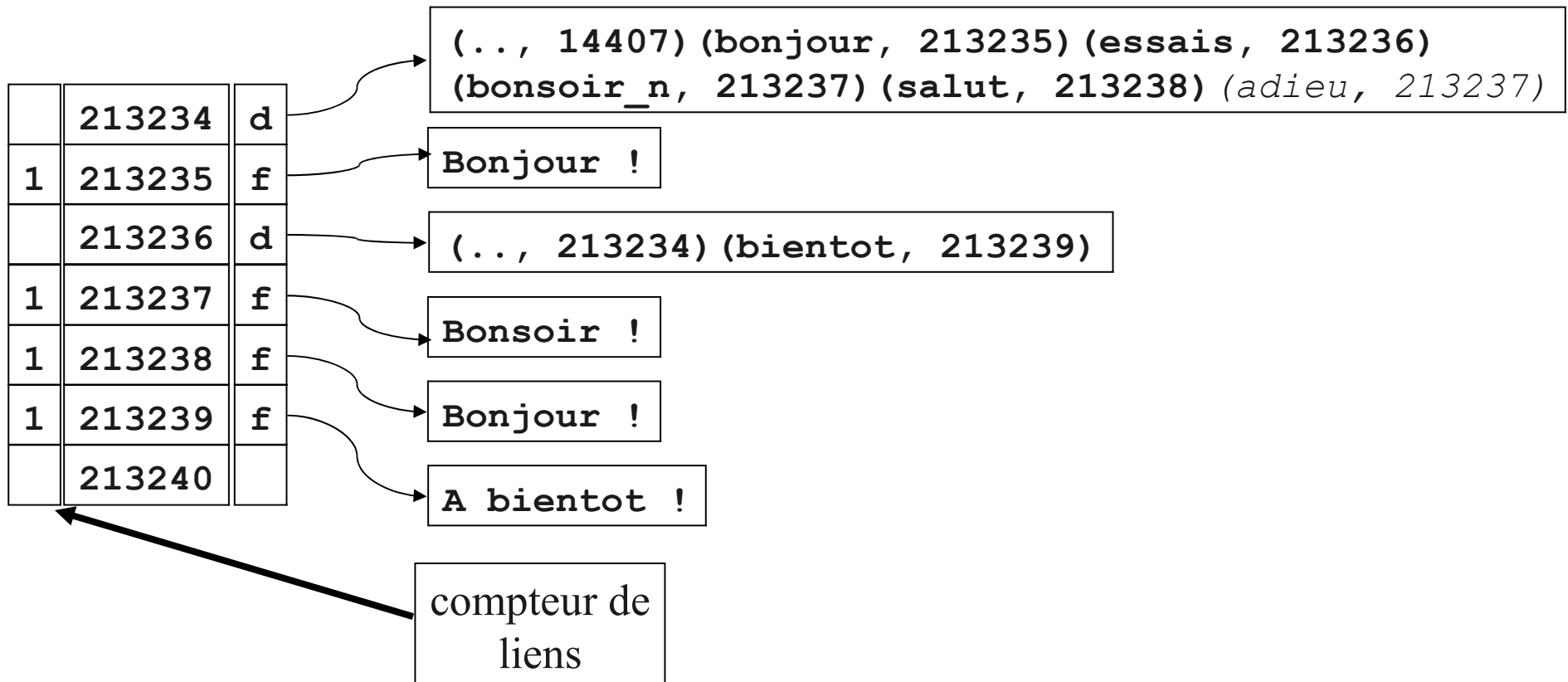
```
213235 -rw-r--r--  1 desar  institut      10 mar 20 15:57 bonjour
213237 -rw-r--r--  1 desar  institut      10 mar 20 15:58 bonsoir_n
213236 drwxr-xr-x  2 desar  institut    1024 mar 20 16:18 essais
213238 -rw-r--r--  1 desar  institut      10 mar 20 16:18 salut
```

```
essais:
```

```
total 1
```

```
213239 -rw-r--r--  1 desar  institut      12 mar 20 15:58 bientot
```

```
[desar@riesling fichiers]$ rm adieu
```



En conséquence, pour éviter d’effacer l’accès à un fichier, il faut interdire le droit en écriture du répertoire où il se trouve.

Les liens symboliques.

Les liens ordinaires sont une façon de créer plusieurs références au même objet.

Une fois créées, ces références sont rigoureusement équivalentes.

Ces liens ne sont applicables qu'aux fichiers (non aux répertoires).

Il existe un autre système de liens dits *liens symboliques*, obtenus par la commande `ln -s`, déjà vue. Le résultat obtenu ressemble aux alias de Windows.

Un lien symbolique est un nouveau fichier qui contient l'adresse à laquelle renvoie ce fichier (cela peut être un répertoire).

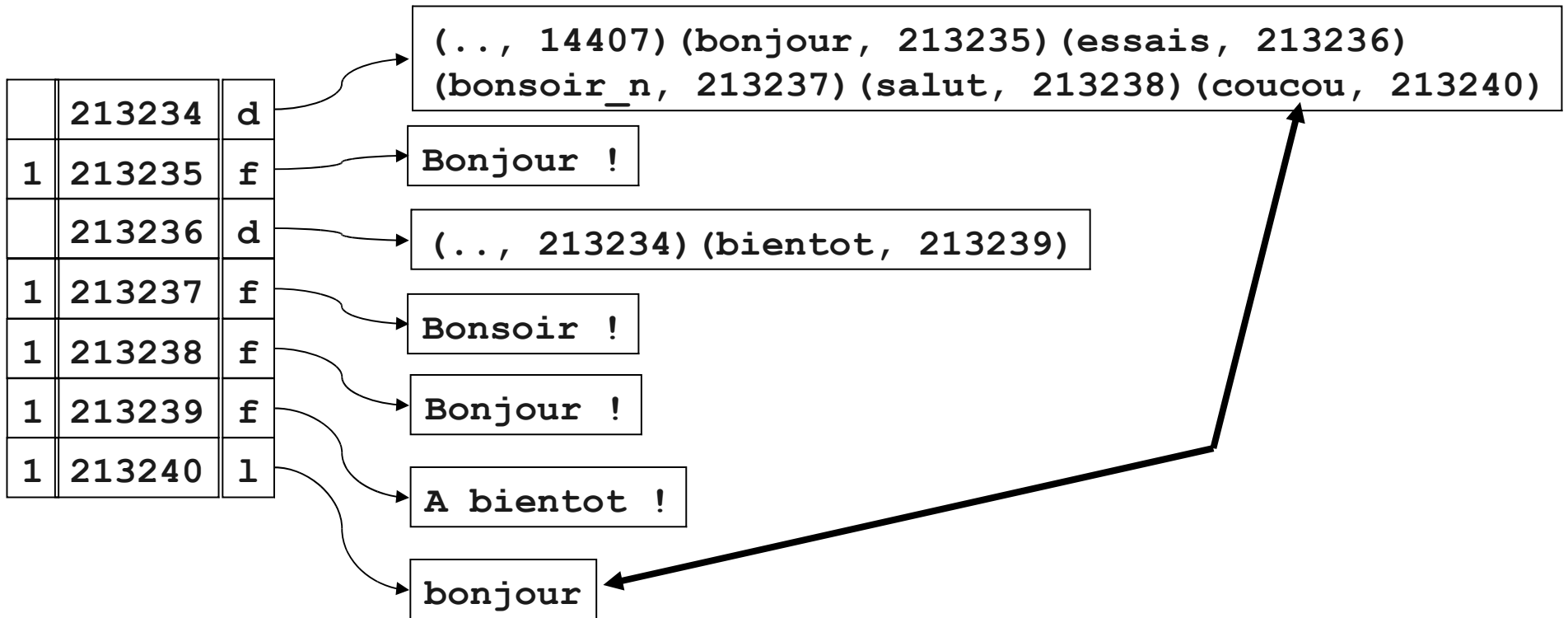
```
[desar@riesling fichiers]$ ln -s bonjour coucou
```

```
[desar@riesling fichiers]$ ls -li
```

```
total 5
```

```
213235 -rw-r--r--  1 desar  institut      10 mar 20 15:57 bonjour
213237 -rw-r--r--  1 desar  institut      10 mar 20 15:58 bonsoir_n
213236 drwxr-xr-x  2 desar  institut    1024 mar 20 16:18 essais
213240 lrwxrwxrwx  1 desar  institut       7 mar 20 16:23 coucou ->
                                     bonjour
213238 -rw-r--r--  1 desar  institut      10 mar 20 16:18 salut
```

```
[desar@riesling fichiers]$ ln -s coucou bonjour
```



Le nom coucou renvoie à bonjour.

```
[desar@riesling fichiers]$ cat coucou  
Bonjour !
```

Mais il n'existe aucune garantie d'intégrité :

```
[desar@riesling fichiers]$ rm bonjour  
[desar@riesling fichiers]$ ls -li  
total 4  
 213237 -rw-r--r--    1 desar    institut    10 mar 20 15:58 bonsoir  
 213240 lrwxrwxrwx    1 desar    institut     7 mar 20 16:23 coucou ->  
                                bonjour  
 213236 drwxr-xr-x    2 desar    institut  1024 mar 20 16:18 essais  
 213238 -rw-r--r--    1 desar    institut    10 mar 20 16:18 salut  
[desar@riesling fichiers]$ cat coucou  
cat: coucou: Aucun fichier ou répertoire de ce type.
```

2.5 Le montage

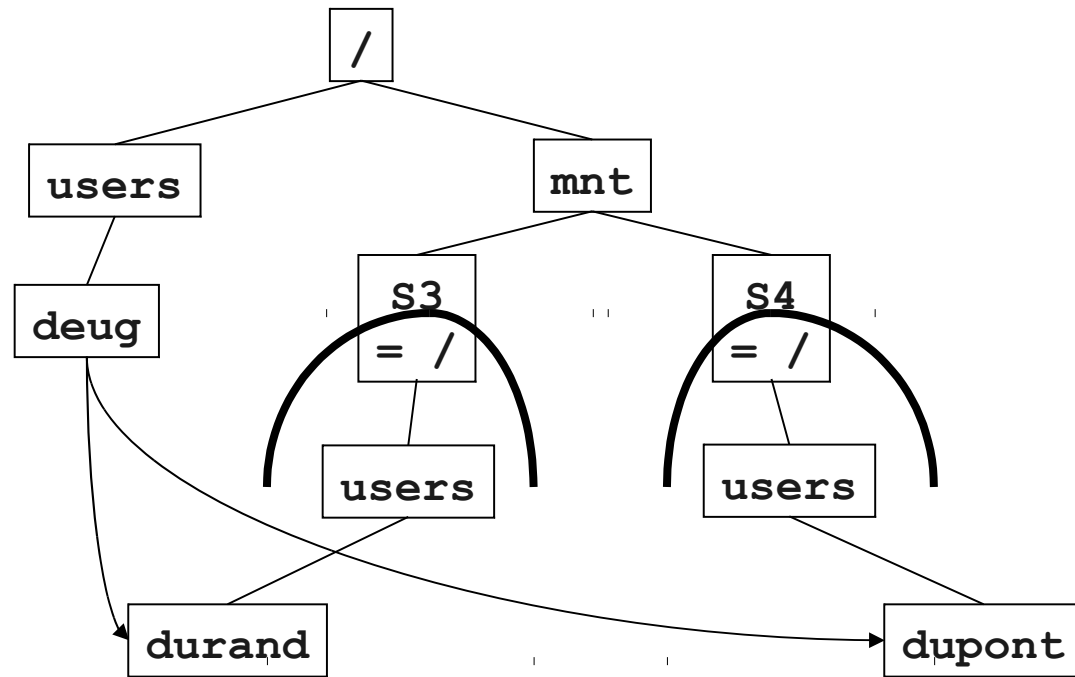
Les systèmes de fichiers différents peuvent être assemblés en un seul : disques physiques, disquettes, CD-rom, disques amovibles (zip).

Chacun est un système de fichiers complet, avec une table de i-noeuds par disque. L'assemblage est fait par le superutilisateur avec la commande mount, qui attache la racine d'un système de fichiers en un point de l'arborescence d'un autre système.

Les liens ordinaires ne permettent pas de passer d'un système de fichiers à un autre monté. Les liens symboliques le permettent de façon transparente pour l'utilisateur.

Voici un exemple où l'utilisateur Durand a ses fichiers sur le disque S3 et l'utilisateur Dupont a les siens sur le disque S4.

Les disques externes S3 et S4 sont montés et des liens symboliques renvoient sur les répertoires de base de Durand et de Dupont.



Il suffit de créer deux liens :

```
ln -s /mnt/S3/users/durand /users/deug/durand
ln -s /mnt/S4/users/dupont /users/deug/dupont
```

si bien que Durand se retrouve chez lui par `cd /users/deug/durand`
 et Dupont par `cd /users/deug/dupont`