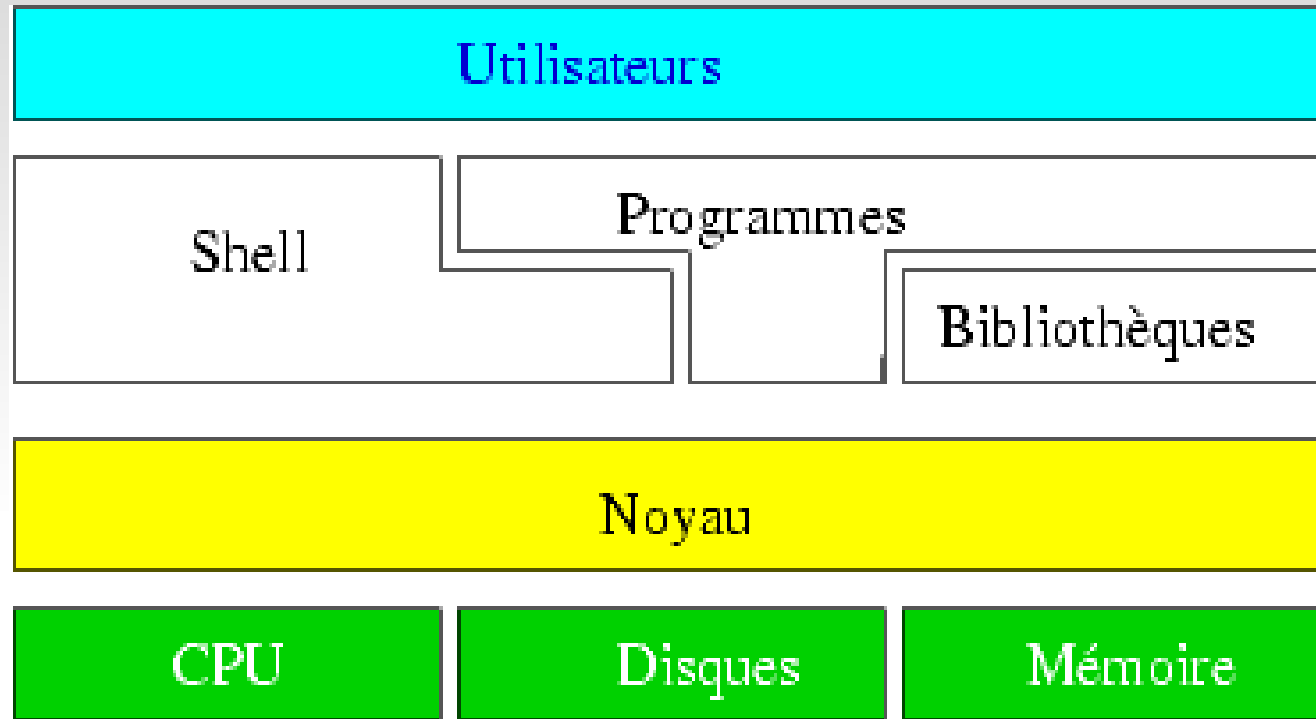


Unix-Intro

- ◆ Créée en 1969. (pas vieux: mûr ;-)
- ◆ Pas UN unix mais DES unix
 - ◆ un arbre genealogique compliqué ...
 - ◆ Plus connus: Linux, xxBSD, MacOS ...
- ◆ Caractéristiques:
 - ◆ Multitaches (temps partagé)
 - ◆ Multi-utilisateurs (droits+extinction).
 - ◆ Interactif (et batch).
 - ◆ Avec ou sans interface graphique.
 - ◆ question de goûts ...Et d'usage

Vue générale



- aucun process n'accède directement à une ressource matérielle → sous le contrôle du kernel
 - fichiers spéciaux
- Cours = Shell = Interpréteur de commandes

Syntaxe générale des commandes

- Les commandes se tapent dans un terminal ou shell
 - `prompt$ cde [options] [arguments]`
- `prompt$` : ce qui figure au debut de chaque ligne
- `[]` : facultatif
- Les commandes acceptent ou non des options et/ou des arguments
- `$man commande` pour tout savoir

Syntaxe générale des commandes (2)

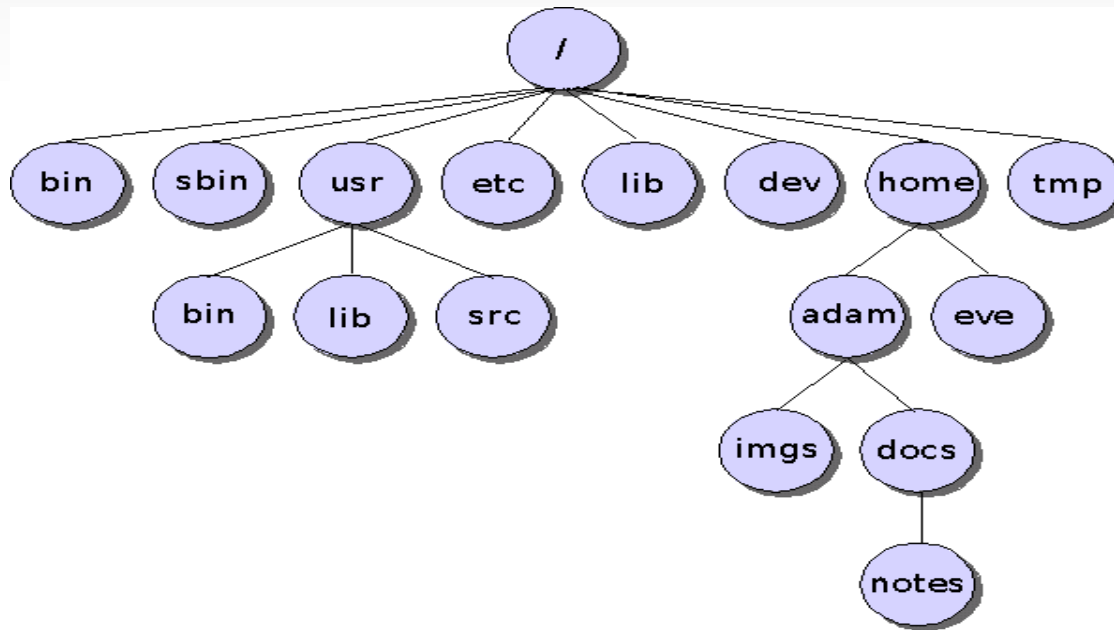
- Qqs exemples:
- `$cde [[-a value] [-b] fichier1 [fichier2]]`
 - `$cde`
 - `$cde fichier1`
 - `$cde fichier1 fichier2`
 - `$cde -b fichier1`
 - `$cde -a fichier1` → pas valide

Fichiers

- Sous Unix, tout est fichier !!!
 - Fichier “normaux”, repertoires, peripheriques ...
- Tout fichier à un nom.
 - Long max: 256 caracteres
 - Peuvent être composés de majuscules, minuscules, chiffres, symboles ...
 - On évite -{ } [] # & ?” car ils ont un sens
 - “case sensitive”: **toto** <> **Toto**
 - Extension non-significative
 - Fichiers “cachés”

Fichiers

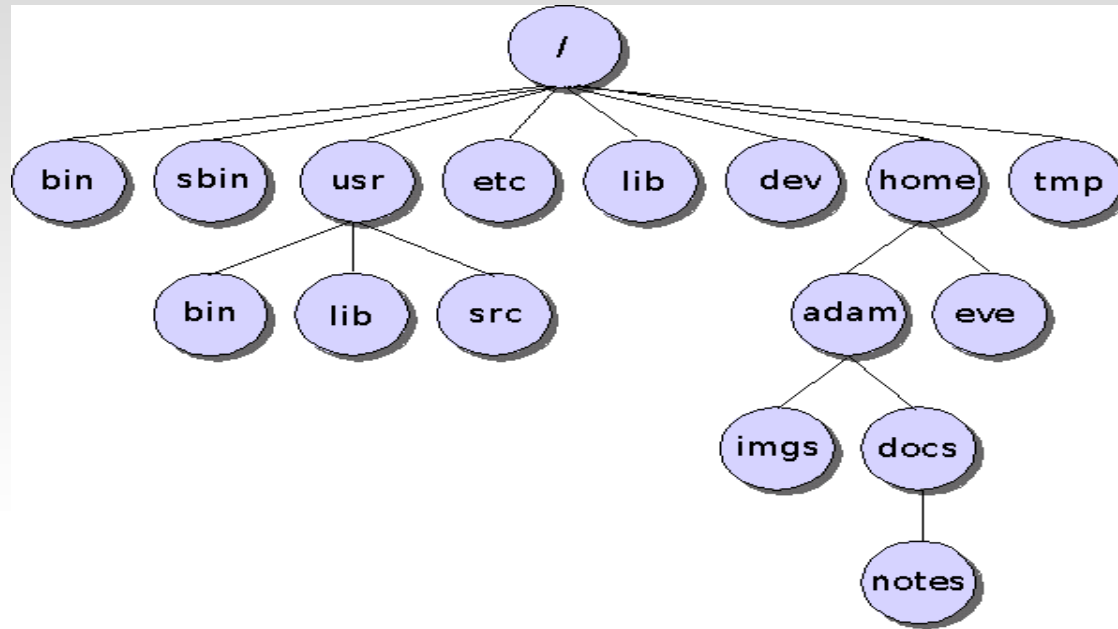
- Tout fichier est dans l'arborescence
 - Arborescence unique \leftrightarrow Windows
 - Racine: /



Fichiers

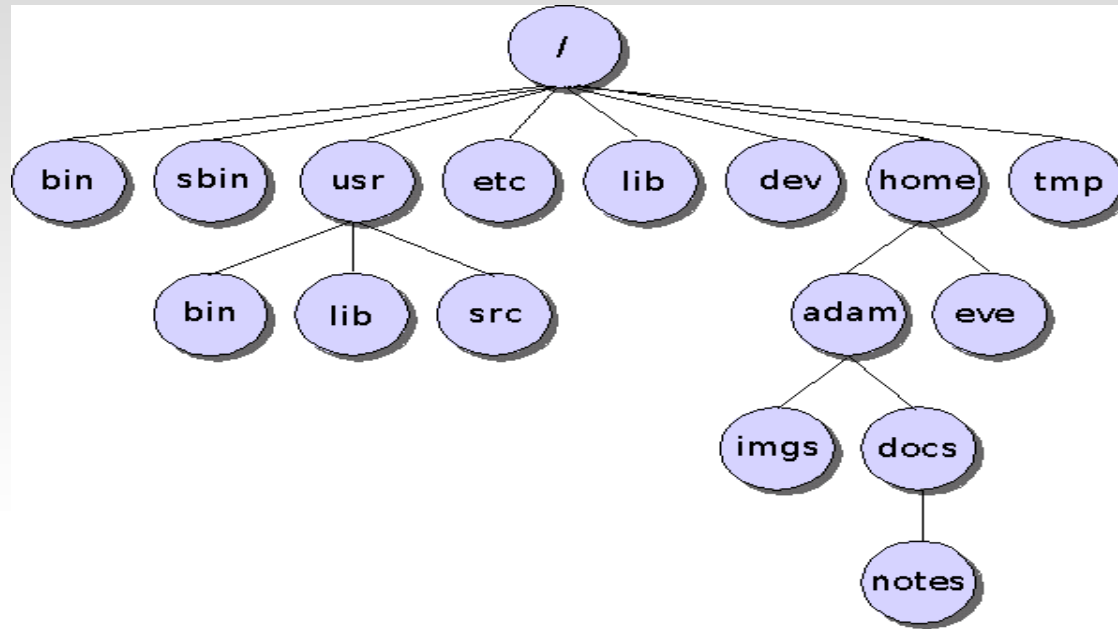
- Tout fichier est dans l'arborescence
 - Clef USB, disquettes, disques → repertoires
 - /media
 - Automontage
 - Il faut accéder à un périphérique pour le voir dans /media
- pour voir la liste des périphériques montés:
 - **%mount [nombreuses options]**
- idem + tailles (total, used, available)
 - **%df [-k] [-h]**

Fichiers – chemins absolus



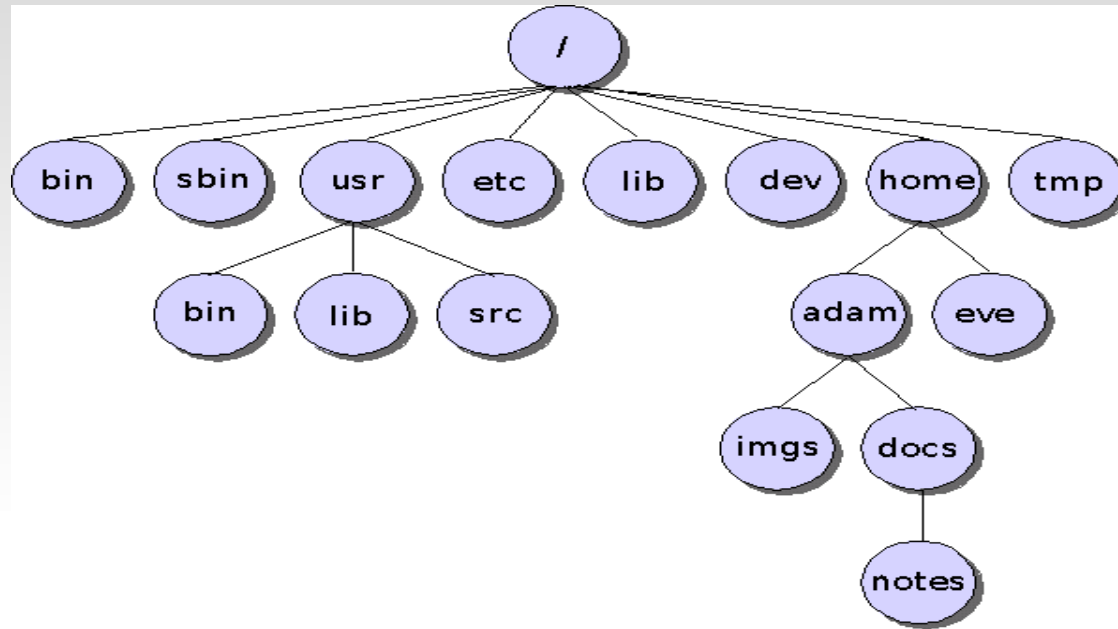
- `/home/adam/docs/notes/machin`: chemin absolu
- désigne qqchose (repertoire ? fichier ?) qui est dans le rep notes qui est dans le rep docs qui est ...

Fichiers – chemins relatifs



- docs/notes/machin: chemin relatif
- N'a de sens que “par rapport” à /home/adam

Fichiers – chemins relatifs

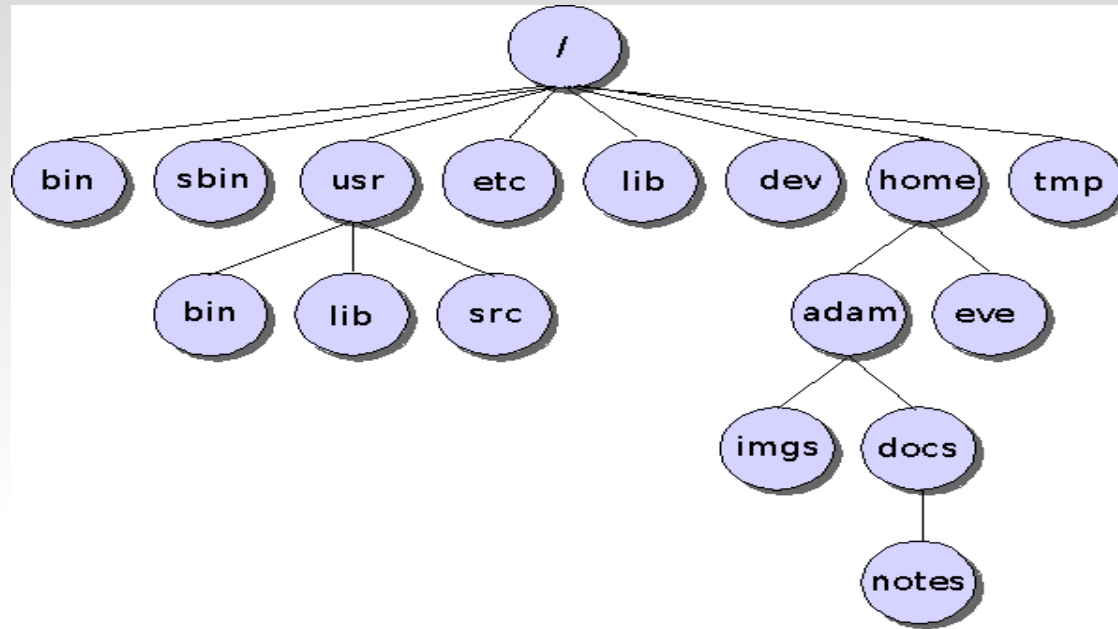


- `..` = le repertoire “parent”
- `../eve` est equivalent à `/home/eve`
 - N'a de sens que dans un sous rep de home
 - `../..`/etc idem

Fichiers – chemins relatifs

- . = Répertoire courant: le repertoire dans lequel on est ...
 - ./machin : le fichier ou repertoire nommé “machin” qui est dans le repertoire dans lequel je suis
 - machin : idem
 - utilisé comme argument de commande pour designer le rep courant. Ex: `$cde arg1 .`
- ./qqchose: surtout utilisé pour exécuter un programme situé dans le rep courant (c.f + tard)

Fichiers – chemins relatifs



- Home directory (maison ...)
 - La partie de l'arborescence qui “est à vous”.
- /home/adam = “chez adam”
- ECM : pas /home

Fichiers – chemins relatifs

- ~ = votre Home Directory
- ~tartempion = le Home directory de tartempion
- ~tartempion/../../truc_tordu
 - Un fichier ou rep nommé truc_tordu situé 2 repertoires “au dessus” de “chez tartempion”

Fichiers – Metacaractères

- * : remplace tte chaine de caractères
 - `$cde toto*` : Lance une commande sur tout ce qui commence par toto et qui est situé dans le rep courant (.)
 - `$cde /users/*toto*` ...
 - `$cde */toto` ...
 - ...

Fichiers – Metacaractères

- ? : remplace un caractère
 - \$cde t?t?
- [range] : remplacé par tous les caractères compris dans le range
 - \$cde t[aeiouy]t[aeiouy]
 - \$cde [A-Z]*
 -

Commandes

- Pour commencer, qqs questions existentielles:
 - Qui suis je ?: **\$whoami / \$id [-a]**
 - Je veux devenir un autre !!!:
 - **\$su - login**
 - **\$logout ou CTRL D**
 - Ou suis je ?(sur quelle machine): **\$hostname**
 - Ou suis je ?(dans quel repertoire): **\$pwd**
 - Qui est sur la même machine ?: **\$who**
 - Quelle heure est il ?: **\$date**

What's the meaning of life ? :h 42 sous vi

Cdes de manipulation fichiers & repertoires

- `$pwd` : “quel est le rep courant ?”
- `$cd [rep]` : aller dans le repertoire spécifié
- `$mkdir [options] rep [rep2 ...]`
- `$ls [options] [arguments]`: afficher ce qu'il y a dans un repertoire
 - `$ls`
 - `$ls /`
 - `$ls *.doc`

Cdes de manipulation fichiers & repertoires

- `$cp [options] nom1 [nom2 ... nom3] nom4`
 - `$cp fic1 fic2`
 - `$cp fic1 fic2 rep`
 - `$cp rep1/* rep2`
 - `....`
- `$mv [options] nom1 [nom2 ... nom3] nom4`
 - `$mv fic1 fic2`
 - `$mv fic1 fic2 rep`
 - `$mv rep1/* rep2`
 - `....`

Cdes de manipulation fichiers & repertoires

- `$rmdir [options] rep1 [rep2 ... rep3]`
 - Peu utilisé car limitations (c.f TP)
- `$rm [options] nom1 [rep2 ... rep3]`
 - `$rm ../toto*`
 - `$rm *`