

# **Administration Unix**

---

**Le cas de Solaris 7**

**Ronan Keryell**

---

**Centre de Recherche en Informatique de  
l'École des Mines de Paris**

**25–26 janvier 2000**

- Unix est un système d'exploitation performant, portable et très complet
- ... et complexe
- ↗ Administration pouvant être assez compliquée car tout est possible
-  Sirènes graphiques : une interface graphique ne fait que cacher la complexité qui réapparaît en cas de problème...
- Nécessité de comprendre comment cela fonctionne !
- Cours plutôt ciblé Solaris 7 mais ceux avec des machines SVR4 et même les autres peuvent être concernés



- Années 60 : collaboration de Bell Telephone Laboratories de AT&T, General Electric et MIT sur le projet de système d'exploitation multi-utilisateur *Multics*
- Projet Multics arrêté en 1969
- En attendant la suite, Ken Thomson de BTL écrit un jeu *Space Travel* qu'il fait tourner sur un PDP-7
- Problème : pas d'environnement de développement sur PDP-7 et nécessité de faire de l'assemblage croisé sur Honeywell 635 roulant GECOS
- Pour faciliter le développement du jeu, développement d'un système d'exploitation pour le PDP-7 : système de fichier simple (*s5fs*), système de gestion de processus, interpréteur de commande (*shell*)
- Le système devient auto-suffisant et est nommé *Unix* en 1969,



jeu de mot en opposition à *Multics*

- Portage d'Unix sur PDP-11 et développement de l'éditeur de texte `ed` et du système de composition de texte `runoff`
- Développement de langage interprété *B* utilisé pour développer les outils
- Dennis Ritchie fait évoluer le langage en *C* dont le succès a largement dépassé le cadre d'Unix
- 1972 : 10 machines sous Unix...
- Unix réécrit en *C* en 1973 et la distribution version 4 contient elle-même `cc`
- L'université de Berkeley récupère une licence (gratuite à cause d'un procès antitrust de 1956 entre AT&T et Western Electric Company)



- La version 7 de 1979 est la première version réellement portable
- Beaucoup d'améliorations fournies par les utilisateurs eux-mêmes (de même que BSD & Linux maintenant) favorisé par le côté non commercial
- MicroSoft et Santa Cruz Operation collabore sur un portage pour i8086 : Xenix
- Portage sur machine 32 bits (Vax-11) en 1978 : UNIX/32V qui est récupérée par Berkeley
- Rajout d'utilitaires (csh de Bill Joy) et d'un système de pagination
- La DARPA donne un contrat à Berkeley pour implémenter IP
- Dernière version en 1993 : 4.4BSD. En tout : apport des *socket*, d'IP, d'un *fast file system*, des signaux robustes, la mémoire virtuelle



- Société BSDI créée pour vendre 4.4BSD *lite* en 1994, débarrassé de tout code d'origine AT&T
- 1982 : loi antitrust qui éclate AT&T en baby-Bell dont le AT&T Bell Laboratories qui peut alors commercialiser Unix
  - ▶ 1982 : System III
  - ▶ 1983 : System V
  - ▶ 1984 : System V release 2 (SVR2)
  - ▶ 1987 : System V release 3 (SVR3) introduit les IPC (InterProcess Communications : mémoire partagée, sémaphores), les STREAMS, le *Remote File Sharing*, les bibliothèques partagées,...
  - ▶ Base de nombreux Unix commerciaux
- 1982 : Bill Joy quitte Berkeley pour fonder Sun Microsystems. Adaptation de 4.2BSD en SunOS qui introduit le *Network File*



*System*, interface de système de fichier générique, nouveau mécanisme de gestion mémoire

- Milieu des années 1980 à Carnegie-Mellon University développe Mach, un micro-noyau avec des serveurs implémentant une sémantique 4BSD. OSF/1 & NextStep sont basés sur Mach
- 1987 : AT&T achète 20% de Sun ↵ prochaines version de SunOS basées sur System V : SunOS 5 (Solaris 2)
- 1989 : co-développement AT&T-Sun de SVR4 : inclut les fonctionnalités de SVR3, 4BSD, SunOS & Xenix. Crédit d'Unix Systems Laboratories pour développer et vendre Unix
- Novell achète une partie d'USL en 1991 pour développer UnixWare (Unix + Netware) et tout USL en 1993
- Arrivée de NT



- Linus Torwald récupère en 1991 Minix, un Unix jouet, sur PC i80386 et le développe en Linux
- 1994 : première version publique de Linux : 1.0.  
Développements pris en main par des programmeurs répartis sur Internet. Intégration des utilitaires GNU
- Novell cède la marque Unix au X/Open puis Sun rachète les droits de SVR4 à Novell en 1994
- Chorus, société française, développe un micro-noyau
- Rachat de Chorus par Sun. ↗ JavaOS ?
- Un système Unix ≡ programmes utilisateurs + bibliothèques + utilitaires + système d'exploitation qui fournit le support d'exécution et les services
- Unix tourne sur toutes les plates-formes depuis les systèmes



embarqués jusqu'aux supercalculateurs massivement parallèles



- La documentation papier « *Solaris 7 System Administrator Collection* » avec en particulier
  - ▶ *System Administration Guide, Volume I*
  - ▶ *System Administration Guide, Volume II*
  - ▶ *NFS Administration Guide*
  - ▶ *Solaris Naming Administration Guide*
- Les manuels et l'« *AnswerBook* » local
- <http://docs.sun.com> : site de référence de la documentation de Sun. Contient aussi l'*AnswerBook*
- Les FAQ (*Frequently Asked Questions* ou encore *Foire Aux Questions*) <http://www.pasteur.fr/other/computer/FAQ/>
- Les News `news:comp.sys.sun`, `news:comp.unix.solaris`
- <ftp://ftp.lps.ens.fr/pub/users/besancon/cookbook/> : le livre «

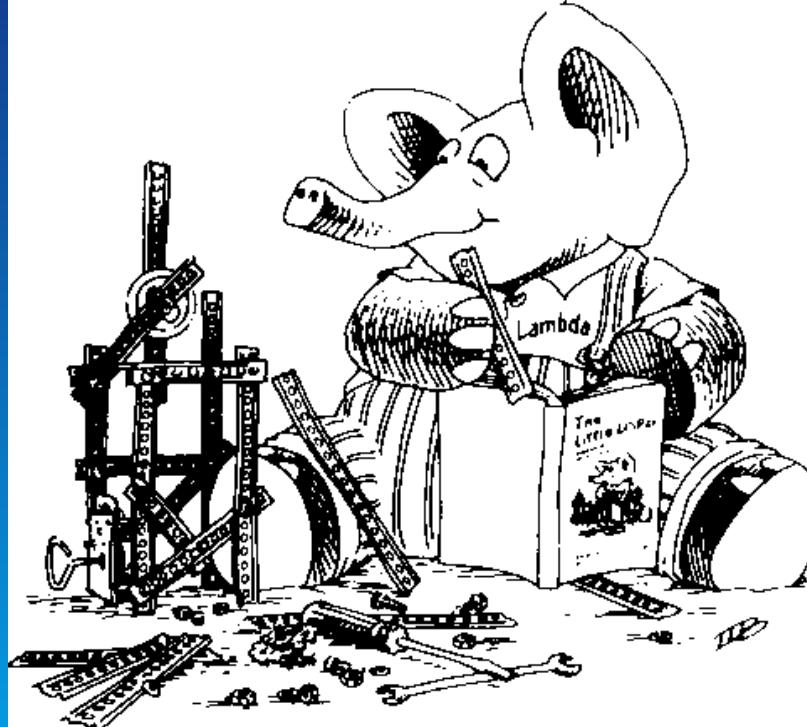


*Administration Système Unix » de Thierry Besançon, Pierre David et Joël Marchand*

- <http://www.freenix.fr/~dumas/linux/Guide/> : « *Le guide du ROOTard pour Linux* »
- Livre *Essential System Administration*, Æleen Frisch, 2nd Edition September 1995 1-56592-127-5, Order Number: 1275 788 pages, O'Reilly, \$34.95 <http://www.oreilly.com/catalog/esa2/>
- Livre « *Unix Internals — The New Frontiers* », Uresh Vahalia, Prentice Hall



- Rappels sur Unix
- Administration
- Installation automatique  
(jumpstart de Solaris)
- Configuration automatique  
(cfengine de GNU)
- Mise en pratique



Processus de *boot* : démarre le noyau système (ou tout autre programme autonome : programme de test,...)

- *Bootstrap loader* : petit programme suffisant pour charger un gros programme (lorsque les ordinateurs se démarraient en programmant explicitement les instructions de démarrage...)
- `man boot`
- Différences entre un Sun et un PC...
- Sun :
  - ▶ Programme (*firmware*) de démarrage écrit en PROM
  - ▶ Exécution d'un programme de test à l'allumage
  - ▶ Chargement du système automatique si indiqué dans la mémoire non volatile du système (`man eeprom`)
  - ▶ Boot primaire depuis



- Disque : bloc de boot primaire (`ufsboot`) dans les blocs 1 à 15 du disque
- Réseau : requête RARP pour récupérer le numéro IP à partir du numéro Ethernet de la machine puis diffusion d'une requête TFTP pour récupérer `inetboot` depuis le réseau qui lui-même fait une requête RARP puis diffuse une requête en protocole `bootparams` pour savoir où trouver son noyau qu'il monte alors par NFS
- PC :
  - ▶ Boot primaire implémenté dans la ROM du BIOS et dans les ROMs des cartes d'extension
  - ▶ Contrôle possible des périphériques et E/S via interruptions
  - ▶ Charge le premier secteur physique du disque ou de la disquette (*Master Boot Record*) et l'exécute en mode réel



- ▶ Le boot secondaire (`boot.bin`) est chargé par le boot primaire et s'exécute en mode protégé et paginé 32 bits en appelant du BIOS en mode réel et est capable de charger un programme depuis un disque UFS, un CD-ROM ou le réseau
- ▶ Le boot secondaire peut lancer le *Configuration Assistant* pour changer la configuration matérielle si l'utilisateur tape ESC rapidement
- ▶ Ensuite, si la variable `auto-boot` de `eeprom` l'indique, le boot secondaire commence et interprète `/etc/bootrc` qui contrôle le processus de démarrage et charge le noyau par défaut. `b` permet alors de spécifier un autre fichier de démarrage et `i` fait entrer en mode interactif
  - Enfin démarrage du noyau par défaut
  - Chargement des modules nécessaires au fonctionnement du



noyau

- Montage des disques nécessaires décrits dans /etc/vfstab
- Démarrage de /sbin/init qui amène le système dans l'état par défaut
- Comme système d'enregistrement des messages non fonctionnel dès le début, regarder messages initiaux avec dmesg



man init

- Système V divise l'état du système en plusieurs niveau pour clarifier la situation
- 0 : retourne sous moniteur ou BIOS
- 1 : met le système en mode administrateur. Les disques locaux sont montés
- 2 : mode multi-utilisateur
- 3 : mode multi-utilisateur et exportation de services vers le monde extérieurs (disques,...)
- 4 : libre pour tout usage
- 5 : arrête le système et la machine (si possible par le matériel)
- 6 : arrête le système et reboot dans l'état par défaut



- a, b et c : pseudo-état pour lancer des commandes correspondant à ces niveaux sans changer l'état
- S ou s : mode *single-user* pour la maintenance (ne monte pas les disques locaux autres que ceux nécessaire au système de base)

Le comportement des niveaux est décrit dans le fichier /etc/inittab qui est relu en tapant init q

```
ap::sysinit:/sbin/autopush -f /etc/iu.ap
ap::sysinit:/sbin/soconfig -f /etc/sock2path
fs::sysinit:/sbin/rcS sysinit >/dev/console 2<>/dev/console </dev/console
is:3:initdefault:
p3:s1234:powerfail:/usr/sbin/shutdown -y -i5 -g0 >/dev/console 2<>/dev/console
sS:s:wait:/sbin/rcS >/dev/console 2<>/dev/console </dev/console
s0:0:wait:/sbin/rc0 >/dev/console 2<>/dev/console </dev/console
s1:1:respawn:/sbin/rc1 >/dev/console 2<>/dev/console </dev/console
s2:23:wait:/sbin/rc2 >/dev/console 2<>/dev/console </dev/console
s3:3:wait:/sbin/rc3 >/dev/console 2<>/dev/console </dev/console
s5:5:wait:/sbin/rc5 >/dev/console 2<>/dev/console </dev/console
s6:6:wait:/sbin/rc6 >/dev/console 2<>/dev/console </dev/console
```



```
fw:0:wait:/sbin/uadmin 2 0 >/dev/console 2<>/dev/console </dev/console  
of:5:wait:/sbin/uadmin 2 6 >/dev/console 2<>/dev/console </dev/console  
rb:6:wait:/sbin/uadmin 2 1 >/dev/console 2<>/dev/console </dev/console  
sc:234:respawn:/usr/lib/saf/sac -t 300  
co:234:respawn:/usr/lib/saf/ttymon -g -h -p `uname -n` console login: -T AT386 -d /dev/console
```

Le premier champ est juste mnémotechnique. `initdefault` indique l'état normal

`/etc/default/init` positionne les variables d'environnement de tous les processus

`TZ=MET`

`LC_COLLATE=fr.ISO8859-15`

`LC_CTYPE=fr.ISO8859-15`

`LC_MESSAGES=fr`

`LC_MONETARY=fr.ISO8859-15`

`LC_NUMERIC=fr.ISO8859-15`

`LC_TIME=fr.ISO8859-15`



qui précise le fuseau horaire et le fait qu'on veut du français et un encodage avec Euro



- Permet de ne pas être déranger par les utilisateurs lors de tâches d'administration lourdes (mise à jour du système,...)
- Niveau de démarrage par défaut en cas de gros problèmes (disques immontables car systèmes de fichiers corrompus) et attente d'une correction par l'administrateur
- Peut passer en mode mono-utilisateur depuis un état mono-utilisateur
- Boot avec `boot -s` (Sun) ou `b -s` (PC)
- Mode qui pose plus de questions avec `boot -as` et `b -as`
- Si on sort du shell mono-utilisateur (`^D`) on lance le mode multi-utilisateur



- D'abord savoir où on en est... `who -a` (pas la version GNU)
- Le plus violent (ne prévient pas les utilisateurs...) : `init niveau`
- Des équivalents rapides
  - ▶ `reboot` (BSD) pour `init 6`
  - ▶ `halt` (BSD) pour `init 5`
  - ▶ `poweroff` pour `init 0`
- Plus conciliant avec les utilisateurs : `shutdown`. Exemple :  
`shutdown -y -g secondes -i niveau message`  
envoie un message régulier à tous les utilisateurs annonçant la fin prochaine et la raison. Messages envoyés aussi aux utilisateurs des machines utilisant des disques de la machine locale



- Chaque niveau  $n$  lance l'exécution d'un script  $/sbin/rcn$  (*Run Control script*)
- Grossièrement, chaque script exécute les programmes du répertoire  $/etc/rcn.d$  commençant par la lettre K par ordre alphabétique et avec stop comme argument puis les scripts commençant par la lettre S par ordre alphabétique et avec start comme argument

Exemple de  $/etc/rc2.d$  :

K07dmi	S30sysid.net	S73nfs.client	S80PRESERVE
K07snmpdx	S47asppp	S74autofs	S80spc
K28nfs.server	S69inet	S74syslog	S85power
README	S70uucp	S74xntpd	S88sendmail
S01MOUNTFSYS	S71rpc	S75cron	S88utmpd
S05RMTMPFILES	S71sysid.sys	S75savecore	S92volmgt
S20sysetup	S72autoinstall	S76nscd	S93cacheos.finish
S21perf	S72inetsvc	S80kdmconfig	S99audit
S22acct	S73cacheefs.daemon	S80lp	S99dtlogin

- Pour simplifier, comme un script peut être utilisé par plusieurs



niveaux, ils sont tous mis dans `/etc/init.d` et on utilise en fait des liens vers eux depuis les `/etc/rcn.d`. En plus les scripts K et S pointent vers les mêmes fichiers

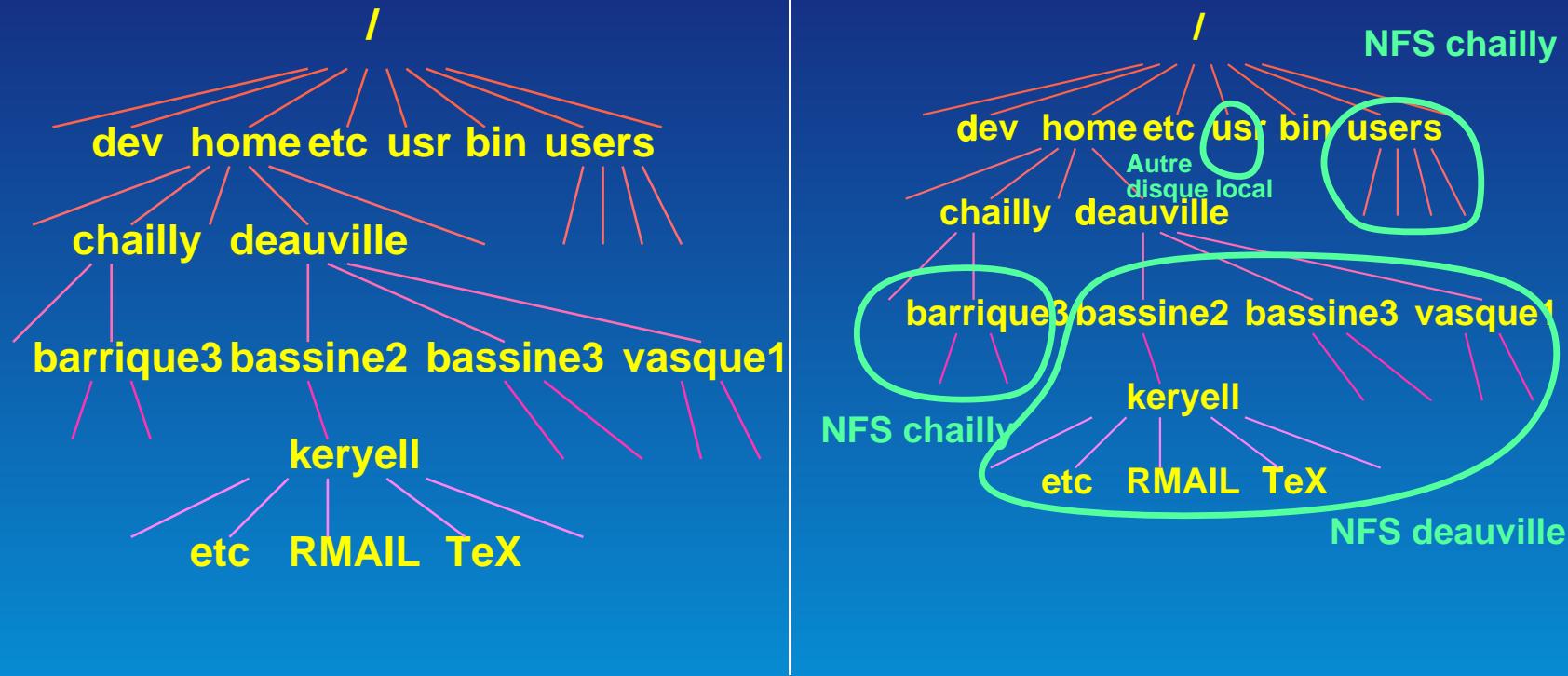
- Si un script se termine par `.sh` il est « sourcé » par le `sh` courant du `/sbin/rcn` au lieu d'être exécuté dans un autre `sh` afin de propager des effets de bords (variables de `sh`,...)
- Les comportements varient en fonction des niveaux et des niveaux précédents
- ↗ Simple de rajouter quelque chose de nouveau en gérant les dépendances. Exemple de `pppd` : faire un `/etc/init.d/ppp`

```
#!/bin/sh
state=$1
case $state in
'start')
```



```
/etc/ppp/ppp-on &
;;
'stop')
/etc/ppp/ppp-off
;;
esac
```





- Organisation hiérarchique simplifiant l'administration
- « Montage » ≡ accrocher une hiérarchie à un répertoire de la hiérarchie déjà existante
- Morceaux de hiérarchie privés (configuration et fichiers de la



machine elle-même)

- Morceaux de hiérarchie partagés (partage de fichiers)
- Pour avoir un système fonctionnel il faut au moins
  - / : La racine de la hiérarchie Unix. Partition montée par le processus de démarrage
  - /usr : Hiérarchie partageable montée par un script lors du démarrage
- La majorité des choses sont configurables à partir de fichiers texte (pas de *Registry*...)



man filesystem

/boot : version PC : contient la configuration de boot et le système d'aide à la configuration matérielle. Émulation de l'EEPROM des Sun

/dev : fichiers spéciaux permettant d'accéder au matériel. Les fichiers *devices* sont typiquement construits pour refléter la configuration logicielle et matérielle de la machine

/dev/dsk : disques accédés en mode bloc (pas de système de fichier)

/dev/pts : pseudo-terminaux

/dev/rdsk : disques accédés en mode bloc (pas de système de fichier) mais de manière brute (par blocs de 512 octets forcément)



/dev/rmt : bandes en mode brut

/dev/sad : gestion des STREAMS (programmes qu'on peut insérer dans des flots de données)

/dev/term : pour accéder aux terminaux et modems

/dev/cua : pour accéder aux terminaux et modems en outrepassant la détection de porteuse

/devices : idem /dev mais non plus classé par fonction mais par contrôleur (pseudo/, sbus@1f,0/). En général les fichiers de /dev sont des liens vers des fichiers de /devices

/etc : fichiers et bases de données de configuration spécifiques à la machine. Définit l'identité de la machine

/etc/acct : configuration du système de comptabilité système

/etc/cron.d : configuration du système de lancement de processus



à instants déterminés cron

/etc/default : configuration par défaut pour divers programmes

/etc/dfs : information de configuration sur les systèmes de fichiers exportés vers (utilisables par) d'autres machines

/etc/fs : binaires permettant de monter différents types de systèmes de fichier (CD-ROM, disque, NFS) lors du démarrage avant le montage de /usr

/etc/inet : fichiers de configuration d'Internet et de ses services

/etc/init.d : scripts utilisés pour passer d'un niveau d'exécution du système à un autre

/etc/lib : bibliothèques partagées utilisées lors du démarrage

/etc/lp : fichiers de configuration du système d'impression



/etc/mail : configuration du courrier électronique

/etc/net : configuration de certains services réseaux  
indépendamment du niveau transport

/etc/opt : informations de configuration sur les packages  
optionnels

/etc/rc0.d : scripts pour entrer ou sortir du niveau d'exécution 0 du  
système (moniteur/BIOS de la machine)

/etc/rc1.d : scripts pour entrer ou sortir du niveau 1  
(administration système)

/etc/rc2.d : scripts pour entrer ou sortir du niveau 2  
(multi-utilisateur)

/etc/rc3.d : scripts pour entrer ou sortir du niveau 3  
(multi-utilisateur et exporte les ressources)



/etc/rcS.d : scripts pour entrer ou sortir du niveau S (*single-user*)

/etc/saf : configuration de l'utilitaire d'accès à des services  
(login,...)

/etc/skel : fichiers de configuration par défaut pour le rajout  
d'utilisateur via useradd

/etc-tm : fichiers Trademark affichés lors du démarrage du système

/etc/uucp : information de configuration d'UUCP

/export : racine par défaut des disques qui seront exportés

/floppy : pour accéder au lecteur de disquette

/home : racine par défaut des répertoires des utilisateurs

/mnt : répertoire vide utilisable pour des montages temporaires de  
systèmes de fichiers



/opt : répertoire contenant des applications additionnelles sous forme de package

/proc : contient le système de fichier des processus

/sbin : exécutables nécessaire lors de la phase de démarrage ou d'une phase de récupération système après catastrophe

/tmp : répertoire des fichiers temporaires :  effacé pendant la phase de démarrage

/var : répertoire contenant des fichiers propres à la machine mais qui varient (grandissent...) au cours du temps

/var/adm : contient des fichiers de log et de comptabilité

/var/log : contient des fichiers de log et de comptabilité

/var/cron : fichiers de log de cron



/var/mail : boîtes aux lettres des utilisateurs

/var/nis : base de donnée NIS+

/var/opt : fichiers variants associés à des logiciels optionnels

/var/preserve : contient les fichiers de sauvegarde pour vi et ex

/var/sadm : base de données utilisée par le système de gestion des paquets de logiciel

/var/saf : messages et comptabilité de saf

/var/spool : répertoire de spool

/var/spool/cron : contient les bases de données cron et at par utilisateur

/var/spool/locks : contient des fichiers de verrouillage

/var/spool/lp : zone de stockage des fichiers à imprimer



/var/spool/mqueue : zone de stockage du courrier électronique en cours de traitement

/var/spool/pkg : zone de stockage de paquets de logiciels

/var/spool/uucp : opérations uucp en cours de traitement

/var/spool/uucppublic : fichiers déposés par uucp

/var/tmp : fichiers temporaires. Répertoire non effacé lors du démarrage du système

/var/uucp : fichiers de messages et statut d'uucp

/var/yp : base de donnée NIS

/vol : répertoire de périphériques gérés par le gestionnaire de volume

/kernel : hiérarchie contenant la partie du noyau qui est



indépendante des spécificités de la machine

/kernel/drv : contrôleurs du matériel 32 bits

/kernel/drv/sparcv9 : contrôleurs du matériel 64 bits pour SPARCv9

/kernel/genunix : le noyau lui-même, indépendant de l'architecture de la machine

/platform : répertoire contenant les objets spécifiques à la plate-forme. Structure calquée sur celle de /

/platform/*platform-name*/kernel : contient les parties du noyau spécifiques au matériel

/platform/*platform-name*/kernel/unix : noyau 32 bits

/platform/*platform-name*/kernel/sparcv9/unix : noyau 64 bits pour SPARCv9



*/platform/platform-name/lib* : bibliothèques spécifiques

*/platform/platform-name/sbin* : exécutables spécifiques



- Garder / petit
- ↵ gros disques montés dans /export, /usr, /opt, /var,...
- /usr contient des fichiers partageables indépendants de l'architecture (/usr/share) ou pas
- Possible de faire des serveurs de /usr (en lecture seulement)

/usr/4lib : bibliothèques de compatibilité Solaris 1 (SunOS4)

/usr/bin : contient les utilitaires systèmes standard

/usr/ccs : système de compilation C

/usr/demo : programmes de démonstration avec leur données

/usr/dt : racine de CDE Motif.

/usr/dt/bin : utilitaires CDE de base

/usr/dt/include : entêtes CDE



/usr/dt/lib : bibliothèques CDE

/usr/dt/man : manuels CDE

/usr/games : des jeux ! Mais vide...

/usr/include : les entêtes pour la compilation C

/usr/java : compilateur et machine virtuelle Java

/usr/kernel : morceau du noyau indépendant de la plate-forme  
non indispensable au démarrage

/usr/lib : bibliothèques ainsi que certains exécutables tels que des démons

/usr/lib/64 : bibliothèques 64 bits

/usr/lib/acct : scripts et binaires utilisés pour la comptabilité

/usr/lib/class : classes d'ordonnancement



/usr/lib/font : fontes pour troff

/usr/lib/fs : modules et commandes spécifiques à des systèmes de fichier

/usr/lib/iconv : tables de codage de jeux de caractères pour iconv

/usr/lib/libp : bibliothèques système utilisée lors du *profiling*

/usr/lib/locale : bases de données de localisation en fonction de la langue

/usr/lib/lp : bases de données et exécutables utilisés par le processus d'impression

/usr/lib/mail : fichiers utilisés pour la génération de la configuration de sendmail V8

/usr/lib/netsvc : programmes et démons utilisés par les services



réseau Internet

/usr/lib/nfs : démons NFS

/usr/lib/sa : commandes utilisées par le système d'analyse  
d'activité sar

/usr/lib/saf : programmes et démons pour saf

/usr/lib/sparcv9 : bibliothèques 64 bits SPARCv9

/usr/lib/spell : bases de données du correcteur orthographique

/usr/lib/uucp : programmes auxiliaires pour uucp

/usr/local : tout ce qui est local à un site. Reproduit l'organisation  
de /usr pour mettre le logiciels du domaine public, etc. dans  
/usr/local/bin, /usr/local/man, /usr/local/src, etc. ou  
dans des sous-répertoire comme /usr/local/corba contenant  
lui-même un bin, man, include, etc. si l'ensemble est gros



/usr/old : des vieilleries

/usr/openwin : contient le logiciel de fenêtrage OpenWindows basé sur X11

/usr/platform : toute une arborescence de type /platform contenant des choses spécifique à une plate-forme mais non nécessaire au démarrage

/usr/sadm : fichiers et répertoires pour l'administration système

/usr/sadm/install : programmes de gestion des packages

/usr/sbin : commandes d'administration

/usr/sbin/static : version statiquement compilée de certaines commandes en cas de problème avec les bibliothèques dynamiques

/usr/share : fichiers partageables et indépendants de l'architecture



/usr/share/lib : bases de données indépendantes de l'architecture

/usr/share/lib/terminfo : fichiers de description de terminaux pour terminfo

/usr/share/lib/zoneinfo : fichiers décrivant les fuseaux horaires

/usr/share/man : manuels en ligne

/usr/share/src : les sources si disponibles

/usr/ucb : commandes compatible Berkeley (BSD)

/usr/ucbinclude : entêtes à la Berkeley

/usr/ucblib : bibliothèques Berkeley

/usr/xpg4 : commandes POSIX.2



- Une machine peut exporter des disques systèmes vers des machines sans disque ou avec peu de disque
- Définition d'une hiérarchie d'exportation standard

*/export/exec/architecture-name* : /usr pour l'architecture donnée et la version courante du système d'exploitation

*/export/exec/architecture-name.release-name* : /usr pour l'architecture donnée et une version précise du système d'exploitation

*/export/exec/share* : version exportée de /usr/share

*/export/exec/share.release-name* : pour une autre version du système

*/export/root/hostname* : le / de la machine *hostname*

*/export/swap/hostname* : le fichier de swap de la machine



*hostname*

/export/var/hostname : /var de la machine *hostname*



-  Certaines ressources sont centralisées (NIS, NIS+, FNS) et définies sur des serveurs : les administrer sur ces serveurs
- Les fichiers de configuration sont au format texte
- Facile de les modifier avec un éditeur de texte
- Utilisation d'outils graphiques de Sun (admintool, solstice) ou d'autres plus génériques (<http://www.webmin.com/webmin/>). Jolis et pratiques, mais pas toujours portables
- Écriture de scripts de manipulation automatique des fichiers de configuration
- Outils d'automatisation tout faits (cfengine)

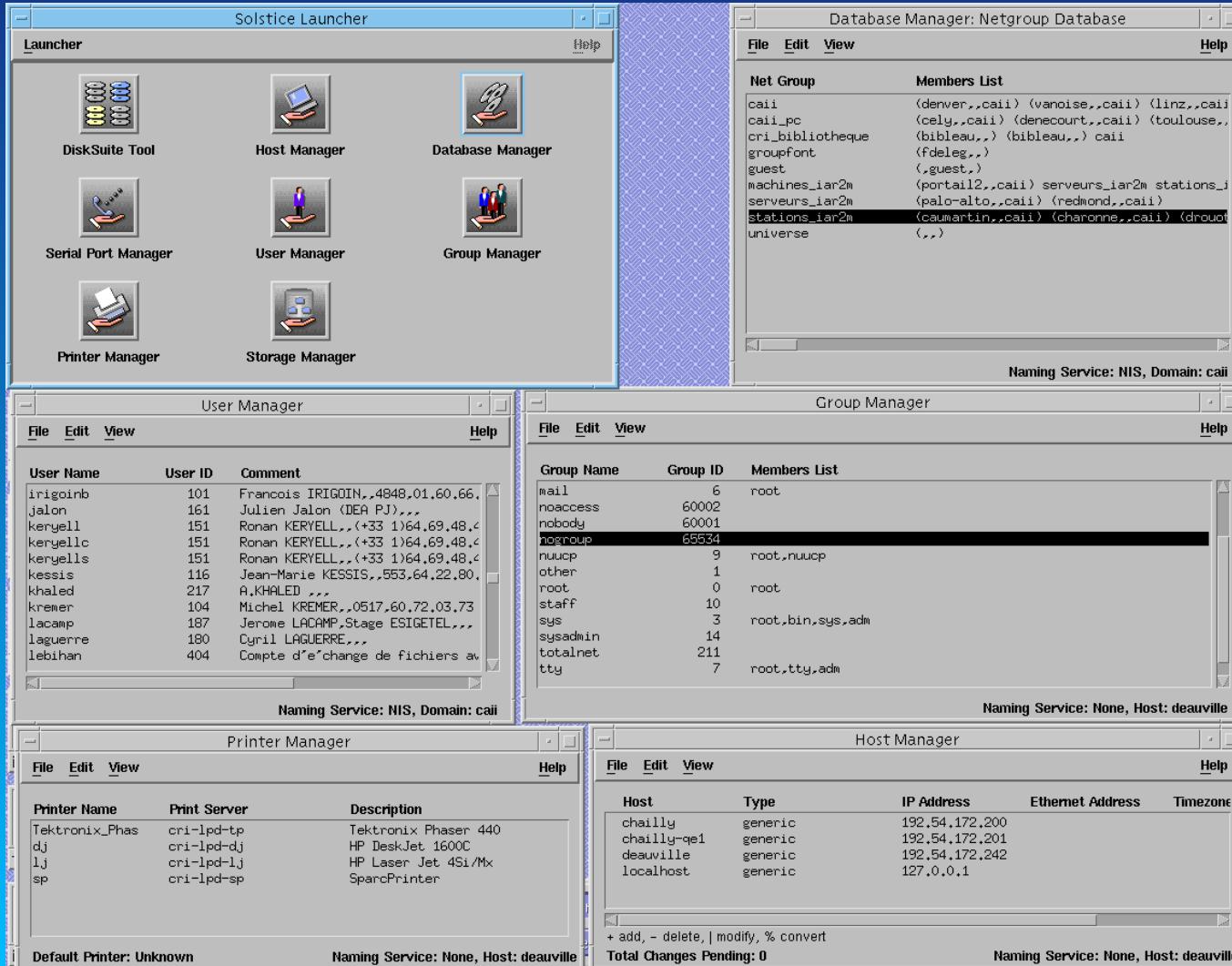


The screenshot displays several windows from the Admintool application:

- Admintool: Users**: A table showing user information. One row for "root" is selected.
- Admintool: Hosts**: A table showing host names and IP addresses. Several rows are selected.
- Admintool: Modify User**: A dialog for modifying a user. It includes fields for User Name (root), User ID (0), Primary Group (1), Secondary Groups (root,bin,sys,adm,uucp,mail,tty,lp), Comment (Super-User), and Login Shell (Other /sbin/sh).
- Admintool: Printers**: A table showing printer configurations. One row for "telegraphe" is selected.
- Admintool: Software**: A table showing software packages. One row for "AutoFS, (Root) 5.7/Generic" is selected.
- Admintool: Add Local Printer**: A dialog for adding a local printer. Fields include Printer Name, Print Server (voltaire.ensmp.fr), Description, Printer Port (/dev/term/a), Printer Type (PostScript), File Contents (PostScript), and Fault Notification (Write to superuser). An "Options" section contains checkboxes for Default Printer and Always Print Banner. A User Access List field shows "all".
- Admintool: Groups**: A table showing group information. One row for "bin" is selected.
- Admintool: Modify Group**: A dialog for modifying a group. It includes fields for Group Name (bin), Group ID (2), and Members List (root,bin,daemon). Buttons for OK, Apply, Reset, Cancel, and Help are present.



## Plus complet qu'admintool (local, NIS & NIS+)



Un compte utilisateur est défini par :

- Nom d'utilisateur utilisé à la connexion. 2 à 8 caractères, sans espace ni souligné, unique à travers tout le site administré (conflits sinon...)
- Mot de passe restreignant l'accès. 6 à 8 lettres. Éviter des mots de passe « trouvables ». Éviter tout ce qui peut être écrit quelque part ou combinaison triviale. Faire changer les mots de passe régulièrement. Ne pas réutiliser son mot de passe dans un autre contexte
- Répertoire personnel (`$HOME` ou `~`) qui contiendra les fichiers de l'utilisateur (qui pourra bien sûr avoir aussi d'autres répertoires à lui ailleurs)
- Des fichiers d'initialisation lus par le shell lors d'une connexion
- Un numéro d'utilisateur utilisé en interne pour faire référence à



l'utilisateur.  Éviter les conflits dans une grosse organisation.

## Allocation des numéros

- ▶ 0 à 99 : comptes réservés au système, root (0), daemon, bin, sys,...
- ▶ 60001 : nobody
- ▶ 60002 : noaccess
- ▶ Le reste de 100 à 2147483647 : utilisateurs standards.

Utiliser ce grand espace de numérotation pour instaurer une hiérarchie (regrouper les utilisateurs par services, etc.).

Exemple 100 $xy$  pour les utilisateur de l'équipe système, etc.).

 Quelques problèmes de compatibilité avec de vieux Unix pour les gros numéros  $\leq 60003\dots$

Éviter de ré-allouer un ancien numéro : il peut rester des fichiers de l'ancien utilisateur qui seront alors accessibles au nouvel



utilisateur...

- Un numéro de groupe primaire d'utilisateurs partageant une unité de droit d'accès. Exemple : tous les membres d'une même équipe sont mis dans le groupe `equipe` et peuvent accéder aux fichiers réservés à l'équipe en lecture seulement pour le groupe `equipe`
- `id -a` permet d'afficher son identificateur et ses groupes
- Par défaut, lorsqu'un utilisateur crée un fichier il lui appartient : il possède son numéro d'utilisateur et son numéro de groupe primaire



- Déclare un compte utilisateur

- Syntaxe (`man -s 4 passwd`) :

*username : password :uid :gid :gecos-field :home-dir :login-shell*

*username* : nom de login

*passwd* : mot de passe crypté de l'utilisateur (/etc/passwd est lisible par tout le monde...). Si \* : compte verrouillé. Si x, il faut accéder aux vrais mots de passe cryptés dans le fichier /etc/shadow lisible seulement par root

*uid* : numéro de l'utilisateur

*gid* : numéro de groupe primaire de l'utilisateur

*gecos-field* : nom réel de l'utilisateur (apparaît dans le champ From: des mails par exemple) (nom du temps où ces informations étaient utilisées pour identifier les commandes envoyées au General Electric Computer Operating System



chez Bell Labs...)

- La commande `passwd` permet de changer son mot de passe ou le mot de passe de quelqu'un d'autre (si on est `root`...)
- `su` permet de prendre les droits d'un autre utilisateur, `su -` lance en plus un nouveau shell de login



- Utilisé en plus de /etc/passwd pour éviter que tout le monde puisse accéder aux mots de passes cryptés et fasse tourner un craqueur de mot de passe
- Utilisé localement ou via NIS+ (NIS n'a pas de protection de lecture)
- Syntaxe (`man -s 4 shadow`) :  
*username : password : lastchg : min : max : warn : inactive : expire*  
*username* : nom de login  
*password* : mot de passe crypté de l'utilisateur  
*lastchg* : jour de dernière modification du mot de passe  
*min* : nombre de jours minimum nécessaires entre des changements de mots de passe  
*max* : nombre de jours de validité du mot de passe. Après, un nouveau est demandé



*inactive* : nombre de jours d'inactivité tolérable avant blocage

*expire* : jour d'expiration du compte



- Déclare les groupes
- Donne un nom aux groupes utilisés dans /etc/passwd
- Rajoute les utilisateurs dans d'autres groupes, les groupes secondaires. Utile pour déclarer des entités transversales (appartenance à des projets,...). Limite par défaut de 16 groupes secondaires par utilisateur
- newgrp permet de se connecter sous un nouveau groupe
- groups affiche la liste des groupes auxquels on appartient
- Syntaxe (`man -s 4 group`) :  
*groupname : password : gid : user-list*  
*groupname* : nom du groupe  
*password* : mot de passe permettant à des utilisateurs de se connecter avec `newgrp` sous un autre groupe dont il ne fait



pas partie

*gid* : numéro de groupe avec les même contraintes que pour le numéro d'utilisateur

*user-list* : liste des utilisateurs appartenant au groupe



-  Attention à la notion de comptes locaux ou de comptes dont la définition est centralisée par un service de nommage (NIS, NIS+, FNS). Dans ce dernier cas il faut demander à aller modifier l'information directement à la base
- Modifier le fichier `/etc/passwd` à la main et éventuellement `/etc/group` (fichiers texte !).  Conflit si plusieurs personnes modifient ces fichiers simultanément. Penser à mettre à jour aussi `/etc/shadow` le cas échéant
- Outils graphiques : `admintool` (utilisateurs locaux) et `solstice` (utilisateurs locaux et centralisés)
- Outils textuels
  - ▶ Utilisateurs : `useradd`/`usermod`/`userdel`
  - ▶ Groupes : `groupadd`/`groupmod`/`groupdel`
  - ▶ Changement de mot de passe avec `passwd`



- ▶ Avec NIS+ utilisation de `nistbladm` et `niscat` pour visualiser une ressource
- ▶ Avec NIS visualiser la liste via `ypcat passwd` et `ypcat group`. Pour propager des changements depuis le serveur, penser à faire un `make` dans `/var/yp` sur le serveur



## Où les mettre ?

- Sun : */home/utilisateur* (et donc sur le serveur dans */export/home/utilisateur*). Simple mais peu hiérarchique
- Au CRI :
  - ▶ */users/centre/utilisateur* est un lien symbolique vers */home/...* pour cacher l'éclatement possible d'un centre sur plusieurs disques
  - ▶ */home/machine/disque centre/utilisateur* contient réellement le répertoire utilisateur
  - ▶ Intérêt à choisir de mettre dans */etc/passwd* le dernier type pour résister à une disparition catastrophique de */users/...*



- 2 religions principales existent : les shells à syntaxe à la sh (sh, jsh, ksh, bash) et ceux à syntaxe à la « C » (csh, tcsh)
- Fichiers de configuration trouvés dans le répertoire principal de l'utilisateur
- 2 états de connexion possibles (shell de connexion principale (login) ou simplement interactif (autres fenêtres ou autre shell) + 1 état non interactif (shell utilisé pour exécuter un script))
- sh
  - ▶ /etc/profile (*a priori* non partagé...) puis .profile si shell de login
- bash : lecture dans l'ordre de
  - ▶ /etc/profile si shell de login
  - ▶ premier fichier trouvé dans l'ordre .bash\_profile,



- .bash\_login et .profile si shell de login
- ▶ .bashrc si shell interactif mais pas de login
- ▶ le fichier spécifié par \$BASH\_ENV si elle existe lors du lancement d'un shell non interactif
- ▶ .bash\_logout lors de la déconnexion si shell de login
- csh et tcsh : lecture dans l'ordre de
  - ▶ /etc/csh.cshrc
  - ▶ /etc/csh.login pour un shell de login
  - ▶ .cshrc ou .tcshrc à la place (s'il existe et tcsh)
  - ▶ .login pour un shell de login
  - ▶ .logout lors de la déconnexion si shell de login
- ↵ Nécessité de fournir des fichiers tout configurés aux utilisateurs mais configurables et en plus avec un mécanisme



d'évolution centralisée... Inclure par exemple un  
source /usr/local/share/env/tcshrc  
/etc/skel contient des fichiers standard de configuration



Quelques unes des variables d'environnement changeant le comportement

DISPLAY : nom de l'écran X11 à utiliser pour l'affichage

HOME : nom du répertoire personnel

LANG : langage et codage des messages du système. Variables LC\_... pour changer seulement des parties (cf. /etc/init/default pour exemple)

LD\_LIBRARY\_PATH : liste des répertoires où trouver les bibliothèques dynamiques

LOGNAME : nom d'utilisateur (gecos)

LPDEST : imprimante par défaut

MAIL : nom du fichier boîte aux lettres



MANPATH : liste des répertoires où trouver les manuels

MANSECTS : liste des sections de manuels possibles

PATH : liste des répertoires où trouver les commandes

SHELL : shell à utiliser si besoin est par certains programmes

TERM : modèle de terminal à utiliser

TZ : fuseau horaire



- Développement des
  - ▶ Disquettes
  - ▶ Cartes PCMCIA (PC-card)
  - ▶ CD-ROMs surtout
- Monter un système de fichier sous Unix est *a priori* compliqué : insérer le médium, passer super-utilisateur, créer un point de montage, monter le contrôleur du médium sur le point de montage, quitter le mode super-utilisateur. Idem pour enlever le médium.
- Pénible et nécessité d'être super-utilisateur
- ↗ outil automatique : *Volume Manager*



- CD-ROM
  - ▶ Insérer
  - ▶ Patienter
  - ▶ Accéder au système de fichiers dans `/cdrom/cdrom0`,  
`/cdrom/cdrom0` (second lecteur),...
  - ▶ Accéder aux données brutes dans  
`/vol/dev/aliases/cdrom0`,...
- Disquette
  - ▶ Insérer
  - ▶ Taper `volcheck` (pas de détection d'insertion)
  - ▶ Accéder au système de fichiers dans `/floppy/floppy0`,  
`/floppy/floppy1` (second lecteur),...
  - ▶ Accéder aux données brutes dans



/vol/dev/aliases/floppy0,...

- Possibilité d'en faire bénéficier d'autres machines en exportant via NFS le système de fichier comme un autre en rajoutant share cdrom\* dans /etc/rmmount.conf pour que ce soit automatique



- Vérifier que plus aucun processus n'utilise le médium
- `eject cdrom` ou `eject floppy` (même si cela ne peut pas éjecter physiquement la disquette, cela en avertit le système)
-  Si médium encore utilisé, impossible d'éjecter... Trouver la liste des processus l'utilisant avec `fuser -u /cdrom/cdrom0` par exemple et employer des mesures correctives
  - ▶ Insérer
  - ▶ Taper `volcheck` (pas de détection d'insertion)
  - ▶ Accéder au système de fichiers dans `/floppy/floppy0`,  
`/floppy/floppy1` (second lecteur),...
  - ▶ Accéder aux données brutes dans  
`/vol/dev/aliases/floppy0`,...
- ↵ outil automatique : *Volume Manager*



Peut stocker de 360 Ko à 2,88 Mo de données

- Format MS-DOS et NEC-DOS
  - ▶ Formattage physique  
`fdformat -v -U + options de densité, nom,...`

- Format UFS

- ▶ SPARC
- ▶ Intel

Incompatibilités liées au stockage grand indien/petit indien des octets

- ▶ Formattage physique  
`fdformat -v -U + options de densité, nom,...`
- ▶ Rajout d'un système de fichier UFS  
`newfs -v /vol/dev/aliases/floppy0`



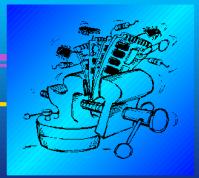
- Avertir le volume manager qu'il peut monter la disquette dans /floppy  
`volrmmount -i floppy0`



- Besoin de gérer l'ajout et l'enlèvement de logiciels complets déjà compilés
- Système V possède la notion de paquets (*packages*)
- Fichier contenant le logiciel lui-même ainsi que ses fichiers de configuration mais aussi scripts d'installation et de dés-installation, de gestion de dépendance
- Solaris est lui-même découpé en paquets. Permet de spécialiser l'installation du système
- Site de logiciels libre <http://smc.vnet.net/solaris.html>
- Gestion par admintool ou pkgadd et pkgrm
- Gestion d'une base de donnée des paquets et fichiers installés. Affichage des logiciels installés par pkginfo [-l]
- Convention de nommage des paquets : commence par un



identificateur du producteur. Paquets de Sun commencent par SUNW



- Installations peuvent poser des questions
- Faire un fichier de réponse avec pkgask
-  Interaction entre installation et partage de ressources entre machines
- Faire le tri entre les paquets pour savoir si l'installation est locale ou centralisée
- Installer les parties centralisées sur les serveurs (typiquement dans /opt, /usr/opt si partagé et /usr/local) et les parties non partagées sur toutes les machines (/etc, /usr/opt si non partagé)
- Problème si installation centralisée pour une machine dont les fichiers ne sont pas au même endroit sur le serveur (`/export/machine/root` au lieu de `/`) : nécessité de rajouter une option pour changer la position de la racine



```
pkgadd -R /export/machine/root SUNWntpr
```



- Mises à jour de morceaux du système d'exploitation (extensions, corrections de bugs,...)
- Extension du système de paquets. Regroupe plusieurs paquets
  - patchadd/patchrm
  - showrev -p affiche la liste de tous les patches installés
  - Nommage : *numéro-de-patch.numéro-de-version*
  - Récupération des patches depuis le site de Sun



- Système de reconfiguration dynamique du matériel (rajout de cartes sans arrêter le système) sur Sun haut de gamme
- Beaucoup de matériels (PC...) ↗ beaucoup de *device drivers*
- Néanmoins la majorités sont déjà disponibles sur un système mais ne sont chargés qu'à la demande lors d'une auto-reconfiguration (gain de mémoire, évite de reconstruire le noyau à chaque fois)
- Conducteurs de matériels placés dans /kernel\_drv et /platform/`uname -m`/kernel\_drv et fichiers de configuration associés en extension .conf en cas de besoin spécifique
- Rajout de conducteurs via pkgadd par exemple



- Vérifier que le conducteur existe pour le périphérique et le rajouter si besoin est
- Arrêter en demandant une reconfiguration du noyau : redémarrage après création d'un fichier `/reconfigure` ou bien redémarrage avec option `-r`
- Éteindre la machine puis les périphériques
- Rajouter le périphérique
- Rallumer les périphériques puis la machine
- Redémarrer le système
- Vérifier que le nouveau périphérique est pris en compte et apparaît par exemple dans `/dev`

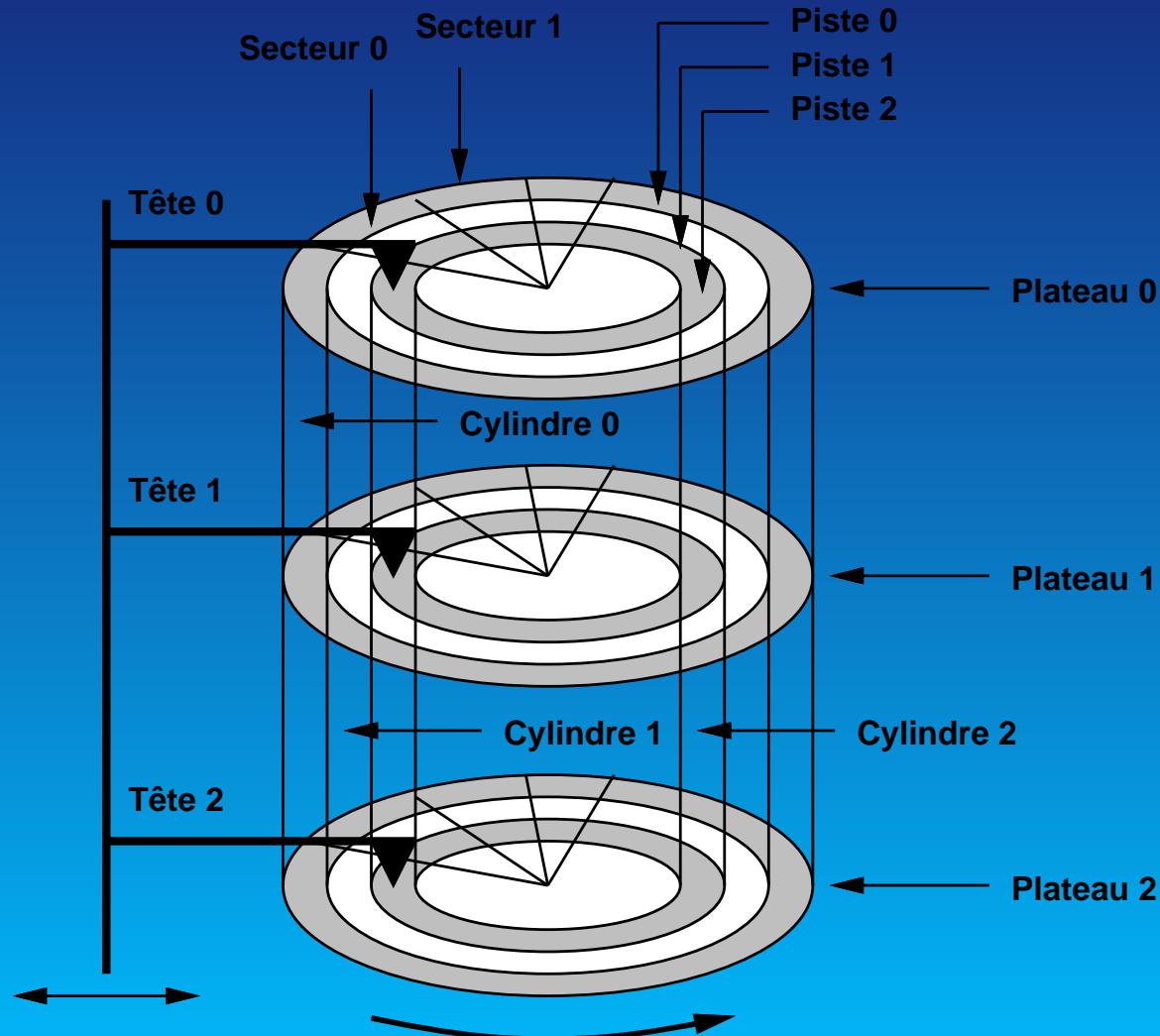


- **dmesg** affiche messages début du boot. Contient entre autre liste des périphériques reconnus

```
Jan 17 18:18
SunOS Release 5.7 Version Generic [UNIX(R) System V Release 4.0]
Copyright (c) 1983-1998, Sun Microsystems, Inc.
mem = 65136K (0x3f9c000)
avail mem = 51163136
root nexus = i86pc
isa0 at root
pcio0 at root: space 0 offset 0
    IDE device at targ 0, lun 0 lastlun 0x0
        model ST36531A, stat 50, err 0
[...]
```

- **sysdef** affiche information sur paramètres du noyau : device drivers, paramètres du noyau, modules chargeables, mémoire
- **prtconf** affiche la liste des périphériques





FFS optimisé pour les disques :



- Partition (ou tranche) : ensemble de cylindres consécutifs ↗ ↘ localité
- Allocation dans des cylindres consécutifs
- Allocation dans des secteurs consécutifs avec un saut (temps de rotation)
- Laisse des cylindres vides régulièrement pour allouer plus rapidement de nouveaux secteurs

Problèmes des caches dans les contrôleurs disque qui éloignent de la réalité...



- Découpe des disques pour des usages différents
- Augmente la localité (et donc performances) des accès au sein de chaque partition
- Limites infranchissables (contre certains utilisateurs expansifs)
- Fournit des zones brutes pouvant avoir chacune leur système de fichier (indépendant et même de type différent) voir sans (swap, base de donnée)
- Peuvent avoir des politiques d'exportation différentes sous NFS
-  Éviter d'avoir 2 partitions qui se recouvrent sans raison...
- Partitionnement fait automatiquement et graphiquement par la procédure d'installation
- Mais en cas de problème sur un disque, de remplacement, de changement du partitionnement : connaissance utile



- Chaque système d'exploitation a sa convention de partitionnement (n'est pas déterminé au niveau du disque lui-même)
- Solaris 7 découpe en 8 ou 10 partitions avec comme convention l'usage courant
  - 0 : contient /
  - 1 : du swap
  - 2 : tout le disque (déborde sur les autres...)
  - 3 : /export sur un serveur
  - 4 : /export/swap sur un serveur
  - 5 : /opt
  - 6 : /usr
  - 7 : /home OU /export/home



- 8 : sur PC contient le système de boot et pointe au début du disque
- 9 : sur PC contient les blocs alternatifs utilisés à la place d'autres en panne et pointe après la partition 8
- Sur PC nécessité d'une « convention collective des OS » pour faire du multi-OS. Convention de partitionner un disque jusqu'en 4 partitions via `fdisk`. Solaris prend une de ces partitions et la repartitionne avec son propre système
-  La loi de Murphy veut que le partitionnement choisi n'est jamais le bon... En général, / et le swap sont trop petits
-  Loi de Murphy numéro 2 : difficile de changer le partitionnement dynamiquement...



- `/dev/dsk` pour accéder aux disques en mode bloc
- `/dev/rdsk` pour accéder aux disques en mode bloc et brut de fonderie (que par blocs de 512 octets, moins pratique mais plus rapide)
- Contrôleur avec un disque relié directement dessus (IDE, Xylogics)
  - ▶ SPARC : `/dev/[r]dsk/cx dy sz`  
 $x$  : numéro logique contrôleur  
 $y$  : numéro disque  
 $z$  : numéro partition Solaris (0 à 7)
  - ▶ Intel : `/dev/[r]dsk/cw dx pz`  
 $w$  : numéro logique contrôleur  
 $x$  : numéro disque  
 $z$  : numéro partition PC (`fdisk`) (1 à 4, partition 0 représente



≡ tout le disque)

`/dev/[r]dsk/cwdxsy` est le partitionnement Solaris à l'intérieur de la partition PC Solaris

$y$  : numéro partition Solaris (0 à 9)

- Contrôleur orienté bus (SCSI, IPI) : plusieurs systèmes indépendants connectés au bus, chaque système peut avoir des disques
  - ▶ SPARC : `/dev/[r]dsk/cwtxdySz`  
 $w$  : numéro logique contrôleur  
 $x$  : numéro cible sur le bus (numéro unité logique SCSI typiquement)  
 $y$  : numéro disque  
 $z$  : numéro partition Solaris (0 à 7)
  - ▶ Intel : `/dev/[r]dsk/cwtxdypt`



*w* : numéro logique contrôleur

*x* : numéro cible sur le bus (numéro unité logique SCSI typiquement)

*y* : numéro disque

*t* : numéro partition PC (`fdisk`) (1 à 4, partition 0 représente ≡ tout le disque)

`/dev/[r]dsk/cwtxdy[sz]` est le partitionnement Solaris à l'intérieur de la partition PC Solaris

*z* : numéro partition Solaris (0 à 7)

- Par convention, partition numéro 2 de Solaris contient tout le disque (ou toute la partition PC `fdisk` réservée à Solaris)
- CD-ROM : comme un disque standard. Généralement caché par le volume manager



Utilisation :

- Installation d'un nouveau disque :
  - ▶ Formattage
  - ▶ Partitionnement
- Affiche les disques reconnus sur le système
- Affiche des informations et leur partitionnement
- Test d'un disque
- Réparation d'un disque
- Destruction du contenu (sensible...) avant renvoi



## Les commandes sont abrégables

- partition gère et affiche le partitionnement (prtvtoc donne aussi l'information)

```
partition> p
```

```
Current partition table (original):
```

```
Total disk cylinders available: 253 + 2 (reserved cylinders)
```

Part	Tag	Flag	Cylinders	Size	Blocks
0	root	wm	3 - 28	203.95MB	(26/0/0) 417690
1	swap	wu	29 - 170	1.09GB	(142/0/0) 2281230
2	backup	wm	0 - 252	1.94GB	(253/0/0) 4064445
3	unassigned	wm	0	0	(0/0/0) 0
4	unassigned	wm	0	0	(0/0/0) 0
5	unassigned	wm	0	0	(0/0/0) 0
6	usr	wm	171 - 252	643.23MB	(82/0/0) 1317330
7	unassigned	wm	0	0	(0/0/0) 0
8	boot	wu	0 - 0	7.84MB	(1/0/0) 16065
9	alternates	wu	1 - 2	15.69MB	(2/0/0) 32130



Adresses aussi en *cylindre/tête/bloc*

Format propose un partitionnement par défaut

- current décrit le disque courant

```
format> cu
```

```
Current Disk = c0t4d0: bassine
<SEAGATE-ST39102LW-0004 cyl 6922 alt 2 hd 12 sec 214>
/sbus@1f,0/espdma@e,8400000/esp@e,8800000/sd@4,0
```

- format reformate le disque
- backup récupère un label (la table des matières du disque) de secours en cas de perte du label principal
- analyse permet de tester le disque avec un effet plus ou moins destructeur
- repair répare 1 bloc du disque en le rajoutant dans la liste des



défectueux et en remet un autre à la place. En cas de problème matériel

- `defect` permet de gérer la liste des défauts (1 gros disque est rarement parfait...)
- `volname` donne un nom au disque. Au CRI on donne des noms de conteneurs pour s'y retrouver. `goutte` aura moins d'octets que `bassine`. Même nom qu'on retrouve monté
- `label` entérine les modifications

`/etc/format.dat` contient les paramètres de formatage (géométrie, etc) des disques connus

- Disque récent (SCSI-2) en bon état : informe directement `format`
- Sinon, lire la documentation ou récupérer un `format.dat` récent (ou le contraire !)



- Partage du disque entre plusieurs OS

Total disk size is 788 cylinders

Cylinder size is 16065 (512 byte) blocks

Partition	Status	Type	Cylinders				%
			Start	End	Length	====	
1		IFS: NTFS	0	50	51		6
2		DOS-BIG	51	101	51		6
3	Active	Solaris	102	356	255		32
4		UNIX System	357	592	236		30

SELECT ONE OF THE FOLLOWING:

1. Create a partition
2. Specify the active partition
3. Delete a partition
4. Exit (update disk configuration and exit)
5. Cancel (exit without updating disk configuration)

Enter Selection:



- 1 seule partition Solaris par disque
- Partition Solaris alignée sur 1 cylindre
- Épargner le *Master Boot Record* sur le cylindre 0
- Subtilité
  1. On formate le disque avec `format`
  2. On partitionne globalement avec `fdisk` appellable directement depuis `format`
  3. On partitionne la partition Solaris générée avec `format` à nouveau...
- Mode non interactif pour extraire des configurations et configurer de manière précise un disque brut style `/dev/rdsk/c0t0d0p0`. Intérêt pour faire des installations automatiques multi-système d'exploitation



- Utilise le Virtual File System : définit une interface permettant de rajouter assez simplement un nouveau type de système de fichiers
- Masque les détails : possibilité de lire, écrire, consulter, etc. quel que soit le type de système de fichiers (local, distant,...)
- Systèmes de fichiers de type disque local

**UFS** : Unix File System, basé sur le FFS 4.3BSD, Système de fichier par défaut

**HSFS** : High Sierra et ISO-9660 (version officielle de la précédente) pour CD-ROM. Lecture seule. Extension Rock Ridge fournissant la sémantique UFS (sauf les liens durs et... l'écriture !)

**PCFS** : lecture et écriture sur des disques au format MS-DOS (typiquement disquettes)



**S5FS** : lecture et écriture sur des disques au format System V sur PC

- Système de fichiers de type accès distant

**NFS** : Network File System pour accéder à des fichiers distants comme s'ils étaient locaux (modulo des différences de performance)

- Système de fichiers virtuels

**CacheFS** : Cache File System pour stocker localement une copie rapide. CD-ROM, Intranet distant,...

**TMPFS** : Temporary File System pour stocker en mémoire pour aller très vite. Configuration par défaut de /tmp (accélération des compilations...) qui est doublement volatil

**LOFS** : Loopback System pour faire apparaître à un autre endroit une partie de la hiérarchie (y compris montages NFS)



**PROCFS** : Process System montre la liste des processus en train de tourner sous forme de répertoires. Utilisé par des outils de debug et d'analyse

5 autres systèmes de fichiers à usage interne sans administration particulière

- Tâche de l'administrateur ?
    - ▶ Créer de nouveaux systèmes de fichiers
    - ▶ Rendre les ressources locales et distantes accessibles aux utilisateurs
    - ▶ Connexion et ajout de nouveaux disques
    - ▶ Mise en place d'une *excellente* politique de sauvegarde
    - ▶ Vérification et correction des fichiers endommagés
- Pour les hackers : `fsdb` un debogueur de système de fichiers pour récupérer un accident...



- Commandes générique : `mount`, `umount`, `mkfs`, `fsck`,... acceptent l'option `-F fs-type` et appellent en fait `mount`, `umount_fs-type`, `mkfs_fs-type`, `fsck_fs-type`,... Voir les documentations de ces dernières commandes pour les détails intrinsèques



- Pour accéder à un système de fichier : montage pour attacher le système à un répertoire de la hiérarchie préexistante
- / est toujours monté (lancement du noyau) et indémontable
- Montage cache les fichiers préexistants dans le répertoire
- Démontage du système de fichier fait réapparaître d'éventuels fichiers préexistants
- Démontage possible seulement si plus aucun process n'utilise le système de fichier
- Démontage utile pour faire une sauvegarde d'une partition en étant sûr que personne ne la modifie
- Arrêt du système utilise une procédure de démontage



## /etc/mnttab fichier en écriture seulement

```
/proc      /proc      proc      rw,suid,dev=2940000      915634700
/dev/dsk/c0d0s0  /        ufs       rw,suid,dev=1980000,largefiles  915634700
/dev/dsk/c0d0s6  /usr      ufs       rw,suid,dev=1980006,largefiles  915634700
fd        /dev/fd    fd      rw,suid,dev=2a00000      915634700
swap      /tmp       tmpfs    rw,dev=1          915634702
palo-alto:/opt   /opt      nfs       ro,dev=2b40001      915634709
palo-alto:/usr/local /usr/local nfs       ro,dev=2b40002      915634710
auto.home     /home      autoofs  ignore,indirect,hard,bg,intr,dev=2b80001 915634711
-xfn       /xfn       autoofs  ignore,indirect,dev=2b80002      915634711
-hosts     /net       autoofs  ignore,indirect,nosuid,nobrowse,dev=2b80003 915634711
auto.direct   /var/mail    autoofs  ignore,direct,dev=2b80004      915634711
auto.direct   /users      autoofs  ignore,direct,dev=2b80005      915634711
voltaire.ensmp.fr:vold(pid246) /vol      nfs       ignore,noquota,dev=2b40003 915634726
palo-alto:/export/verre1 /users  nfs       rw,hard,intr,dev=2b400ca 916307121
palo-alto:/export/verre2/var/mail /var/mail nfs  rw,hard,intr,actimeo=0,dev=2b402aa 916663055
```

df affiche aussi la liste des systèmes de fichiers montés avec de l'information sur la place disponible

Filesystem	1024-blocks	Used	Available	Capacity	Mounted on
/dev/dsk/c0d0s0	195580	34359	141663	20%	/
/dev/dsk/c0d0s6	618904	481994	81209	86%	/usr



swap	1083936	616	1083320	0%	/tmp
palo-alto:/opt	2733423	2312742	407014	85%	/opt
palo-alto:/usr/local	2733423	2312742	407014	85%	/usr/local
auto.direct	2061938	20124	2034941	1%	/var/mail
auto.direct	2061938	1892023	163042	92%	/users
palo-alto:/export/verre1					
	2061938	1892023	163042	92%	/users
palo-alto:/export/verre2/var/mail					
	2061938	20124	2034941	1%	/var/mail



- Déclaration dans la table /etc/vfstab pour montage au démarrage

#device	device	mount	FS	fsck	mount	mount
#to mount	to fsck	point	type	pass	at boot	options
fd	-	/dev/fd	fd	-	no	-
/proc	-	/proc	proc	-	no	-
/dev/dsk/c0d0s1	-	-	swap	-	no	-
/dev/dsk/c0d0s0	/dev/rdsk/c0d0s0	/	ufs	1	no	-
/dev/dsk/c0d0s6	/dev/rdsk/c0d0s6	/usr	ufs	1	no	-
palo-alto:/opt	-	/opt	nfs	-	yes	ro
palo-alto:/usr/local	-	/usr/local	nfs	-	yes	ro
swap	-	/tmp	tmpfs	-	yes	-

- ▶ `mount -a` permet de monter tout ce qui est dans la table
- ▶ `mountall` idem mais seulement ce qui est marqué comme montable lors du boot et lance avant un `fsck` si marqué
- ▶ `mount mount-point` monte le système de fichier correspondant décrit dans la table
- ▶ `mount -p` affiche la table de montage courante en format



vfstab pour ré-injection facile

- Montages infréquents : à la main avec  
`mount [-F FSType] [-o options] [-O] special mount-point` : -O pour monter au dessus d'un point de montage  
`mount -F pcfs /dev/dsk/c0d0p0:c /mnt` pour la partition « C: »
- Auto-monteur : montage à la demande lors de l'utilisation d'un système de fichier  

```
miromesnil-keryell > ls /net
miromesnil-keryell > ls /net/cri.ensmp.fr
export/  usr/
```
- Volume manager : monte automatiquement des média temporaires



- Détache un système de fichier lorsque inutile
- Met à jour (finalise) les structures de données du système de fichier
- Lors de l'arrêt du système (seul moyen de démonter /...)
- `umount mount-point`
- `umountall` essaye de démonte tout sauf le nécessaire vital (/,,  
/proc, /var et /usr)
- `umountall -k` tue les processus qui utilisent les partitions à démonter (cf fuser)
- Par l'auto-monteur lorsqu'un système de fichier n'est pas utilisé pendant un certain temps



- Unix File System est celui utilisé par défaut sous Solaris.  
Extension du FFS 4.3BSD. La partition est divisée en groupes de cylindres
- Boot Block** 8 Ko permettant le démarrage. Existe même si pas partition de boot

**Superblock** contient les informations sur le système de fichier : taille, statut, label, taille des blocs, date de dernière modification, nom du dernier répertoire de montage, etc.

Contient des drapeaux précisant le fonctionnement

**État** *clean, stable, active, logging et unknown*. Permet de savoir où en est le disque lors d'un accident. *clean, stable* ou *logging* ne nécessite pas de *fsck*

**Extended Fundamental Type (EFT)** pour avoir des numéros d'utilisateurs, de groupes et de devices sur 32 bits

**Large file systems** système de fichiers de 1 To en tout.



Pratique si stripping/RAIDs à la DiskSuite

**Large files** pour fichiers dépassant les 2 Go. Par défaut  
Comme l'information des superblocs est critique, elle est  
répliquée dans tous les groupes de cylindres et décalée de  
telle manière qu'elle soit répartie en plus sur tous les plateaux

**Inodes** contiennent toutes les informations sur un fichier sauf  
son nom : type (normal, répertoire, device,...), mode,  
propriétaire et groupe, taille, dates,... et tableau de 15  
adresses de blocs de données. L'adresse 13 pointe vers un  
bloc d'adresses, l'adresse 14 pointe vers un bloc d'adresses  
de blocs d'adresses et l'adresse 15 encore un niveau de plus  
pour les très gros fichiers

**Blocs de données** stockent le contenu des fichiers et des  
répertoires (fichiers de noms et d'adresses d'inodes). Blocs  
de taille 8 Ko ou 1 Ko (fragments) par défaut



**Blocs libres** blocs non utilisés (ni inodes, ni données, ni blocs d'adresse) par groupe de cylindre. Garde trace de la fragmentation pour limiter sa propagation

Pour des raisons de performance, on arrête le remplissage du disque à 90 % de la capacité pour ne pas perdre trop de temps à chercher de la place

- Journalisation
  - ▶ Penser les modifications aux fichiers sous forme de transactions
  - ▶ Stocker les transactions dans un journal
  - ▶ Appliquer (plus tard) les transactions au système de fichier
  - ▶ Après accident, lors du redémarrage les transactions incomplètes sont éliminées mais les transactions complètes sont prises en compte ↵ cohérence maintenue



- ▶ Plus besoin de faire tourner de longs `fsck` au démarrage
- ▶ Démarré par option `-o logging` au montage
- ▶ Le journal est alloué dans la liste de blocs vides
- `mkfs -F ufs` permet de créer un système de fichier en spécifiant tous les paramètres
- `newfs` crée un système de fichier standard en appelant `mkfs -F ufs` avec des paramètres par défaut

```
deauville-root > newfs -v /dev/rdsk/c0t4d0s5
newfs: construct a new file system /dev/rdsk/c0t4d0s5: (y/n)? y
mkfs -F ufs /dev/rdsk/c0t4d0s5 5926944 214 12 8192 1024 64 2 167 6144 t 0 -1 8 128
/dev/rdsk/c0t4d0s5:      5926944 sectors in 2308 cylinders of 12 tracks, 214 sectors
                      2894.0MB in 61 cyl groups (38 c/g, 47.65MB/g, 7936 i/g)
super-block backups (for fsck -F ufs -o b=#) at:
  32, 97840, 195648, 293456, 391264, 489072, 586880, 684688, 782496, 880304,
  978112, 1075920, 1173728, 1271536, 1369344, 1467152, 1561376, 1659184,
  1756992, 1854800, 1952608, 2050416, 2148224, 2246032, 2343840, 2441648,
  2539456, 2637264, 2735072, 2832880, 2930688, 3028496, 3122720, 3220528,
  3318336, 3416144, 3513952, 3611760, 3709568, 3807376, 3905184, 4002992,
  4100800, 4198608, 4296416, 4394224, 4492032, 4589840, 4684064, 4781872,
```



4879680, 4977488, 5075296, 5173104, 5270912, 5368720, 5466528, 5564336,  
5662144, 5759952, 5857760,

Il peut être utile de stocker cette information pour avoir l'adresse des superblocs en cas de coup dur.

- `tunefs` permet de fignoler les paramètres après coup. Moins utile avec tous les caches des disques



- Beaucoup de choses sont faites de manière asynchrone pour accélérer un système de fichiers. `fsflush` effectue les écritures en tâche de fond
- `sync` re-synchronise les disques avec ce que pense l'utilisateur (utile si obligé d'arrêter salement une machine)
- Suite à un reboot intempestif ou une panne matérielle, structures de données incohérentes dans le système de fichier : fichiers à moitiés effacés, superbloc endommagé,...
- Lancement d'un `fsck` au démarrage si un système de fichier n'est pas marqué *clean* (démonté proprement à l'arrêt), *stable* (non démonté proprement à l'arrêt mais non modifié après le dernier `sync` ou `fsflush` avant l'arrêt) ou *log* (système de fichier journalisé). Parcours de toute la structure du disque : long ! Mais analyse de plusieurs disques en parallèle



- ▶ Corrige le superblock (taille, nombre d'inodes, nombre de blocs et d'inodes libres)
- ▶ Peut récupérer un superblock de secours. Si le système est trop HS pour savoir où le trouver, chercher si vous n'avez pas la sortie de `newfs` quelque part sinon faire un `newfs -N` du disque pour faire un système de fichiers pour de faux
- ▶ Vérification des inodes (nombre de liens vers l'inode, taille, blocs de données référencés 2 fois)
- ▶ Correction des répertoires « . » et « .. » dans les répertoires
- ▶ ...
- ▶ Si fichiers et répertoires (inodes) non référencés dans un répertoire : reliés à `lost+found`
- Certains problèmes sont insolubles automatiquement : choix ↗ questions à l'utilisateur. Possibilité de faire un `fsck` à la main



(sur un système de fichier inactif !) avec

`fsck /dev/rdsk/device-name`

-  `fsck` n'a aucun moyen de réparer le *contenu* des fichiers...
-  Ne pas monter a priori de disque local via `/etc/vfstab` sans préciser que le `fsck` doit être fait au démarrage. Un - dans `/etc/vfstab` indique pas de `fsck`, 1 pour `fsck` séquentiel dans l'ordre du `/etc/vfstab` et plus que 1 pour dire que les `fsck` sont ensuite faits en parallèle sur les disques
-  `fsck` ne remplace pas les RAID et encore moins les sauvegardes ! Évite juste les restaurations en cas de problèmes mineurs
- Pour hackers et pompiers le débogueur de système de fichiers :  
`fsdb`, `fsdb_ufs`,...



- Utiliser un cache dans disque local pour servir de mémoire cache de fichiers distants
- Application diminution de trafic
  - ▶ Réseau local pour cacher du trafic NFS
  - ▶ Distant (PPP) et Intranet. Possibilité d'exporter le cache en local pour le factoriser entre plusieurs clients locaux
  - ▶ Accélérer la lecture d'un CD-ROM sur un serveur de CD-ROMs (prévoir au moins 650 Mo de cache par CD-ROM...)
- Extension du concept avec *AutoClient* pour cacher même / et /usr et faire des *network computer* : machine sans système de fichiers local mais un disque qui sert de cache à ses fichiers servis par une machine distante. Utilisation des fichiers du cache aussi pour le reboot



- Créer un répertoire qui contiendra la base de données du cache avec

```
cfsadmin -c répertoire-du-cache
```

```
cfsadmin -c /cache
```

- Créer un montage de type CacheFS avec

```
mount -F cachefs -o  
backfstype=fstype,cachedir=répertoire-du-cache  
back-fsystem mount-point
```

```
mkdir /b
```

```
mount -F cachefs -o backfstype=nfs,cachedir=/cache palo-alto:/export /b
```

- Contrôler l'efficacité du cache sur un système de fichiers avec

```
root@miromesnil /: cachefsstat /b  
/b
```

```
taux de succès : 80% (663 sélectionné, 161 manquant)
```

```
contrôles de cohérences : 163 (163 réussite, 0 échec)
```

```
modifie : 0
```

```
défragmentation de la mémoire : 0
```

- Statistiques sur une base de données de cache avec



`cfsadmin -l répertoire-du-cache`

- Pour améliorer le fonctionnement on peut conseiller au cache de précharger/garder certains fichiers dans le cache
  - ▶ `cachefspack -p liste-de-fichiers`
  - ▶ `cachefspack -f[r] fichier-de-listes` pouvant contenir des expressions régulières, des résultats de commandes exécutées au vol,...
  - ▶ `cachefspack -u liste-de-fichiers`  
`cachefspack -uf fichier-de-listes` pour déverrouiller  
`cachefspack -U répertoire-du-cache` pour déverrouiller tous les fichiers du cache
  - ▶ `cachefspack -i fichier` donne le statut d'un fichier
- `cachefslog` permet de demander au CacheFS de mettre des messages d'information dans un fichier journal de



fonctionnement



- Permet d'avoir plus de mémoire apparente (virtuelle...) que de mémoire physique dans l'ordinateur : moins cher
- Opérations de traduction d'adresse (mémoire virtuelle vers mémoire physique) et d'échange de pages entre la mémoire physique et les fichiers de swap (échange)
-  pour avoir plus de 4 Go par processus nécessite de fonctionner en mode 64 bits et d'avoir un processeur 64 bits (SPARCv9, pas x86)
- Solaris introduit la notion de swap virtuel (SWAPFS) permettant de bénéficier entre autres d'un espace presque égal à la mémoire physique + les fichiers de swap
- Possibilité de rajouter en cours de route du swap mais aussi d'en *enlever* (changements à chaud de matériel, etc)
- Partition (sans système de fichier pour la vitesse) de swap



allouée sur disque lors de l'installation d'une machine

```
/dev/dsk/c0d0s1 - - swap - no -
```

Possibilité d'un nombre arbitraire de partitions de swap dans  
`/etc/vfstab`. ↗ débit en parallélisant sur plusieurs disques

- Surveillance de la mémoire

- ▶ En tout (y compris mémoire physique) avec `swap -s`

```
deauville-keryell > swap -s
total: 329232k bytes allocated + 27936k reserved = 357168k used, 3080200k available
```

- ▶ Par partition avec `swap -l`

```
deauville-keryell > swap -l
swapfile          dev  swaplo blocks   free
/dev/dsk/c0t0d0s1 32,1      16 2307344 2017344
/dev/dsk/c0t3d0s1 32,25     16 4194272 3895568
```

- Rajout de fichiers de swap avec `swap -a fichier` qui peut être une partition brute ou un fichier standard (machine sans disque, pas de partition disque disponible, besoin temporaire). Fichier



standard créé avec mkfile :

```
deauville-root > mkfile 1g /export/vasque2/gros-swap
deauville-root > swap -a /export/vasque2/gros-swap
deauville-keryell > swap -s
total: 329304k bytes allocated + 28168k reserved = 357472k used, 4128408k available
deauville-keryell > swap -l
swapfile          dev  swaplo blocks   free
/dev/dsk/c0t0d0s1    32,1      16 2307344 2017344
/dev/dsk/c0t3d0s1    32,25     16 4194272 3895568
/export/vasque2/gros-swap -       16 2097136 2097136
```

- Suppression d'une zone de swap avec swap -d *fichier*

```
deauville-root > swap -d /export/vasque2/gros-swap
```



- Beaucoup d'applications créent de nombreux fichiers temporaires : compilateurs, bases de données, etc
- Idée : utiliser un système de fichiers en mémoire (TMPFS) pour les répertoires stockant ces fichiers temporaires
- Par défaut /tmp est un TMPFS

swap	-	/tmp	tmpfs	-	yes	-
------	---	------	-------	---	-----	---

-  Si un TMPFS est plein c'est la mémoire virtuelle qui est pleine... Possibilité de limiter la taille d'un TMPFS avec `-o size=taille`
- Inconvénient : si la machine est éteinte le contenu du TMPFS est perdu. En accord avec la sémantique de /tmp
- Sinon autre possibilité pour /var/tmp : utiliser un CacheFS d'un /var/tmp sur disque dont le fichier de cache serait lui même en TMPFS...



-  Les fichiers représentent la propriété intellectuelle. Toute la vie d'une entreprise sous forme de fichiers...
- Les pannes existent, les utilisateurs (`rm -rf *`) et les bugs aussi...
- La majorité des entreprises ne survivent pas à moyen terme à la perte de leur informatique
- Nécessité de faire des sauvegardes régulières (1 fois par jour)
- Et sur le long terme (années) pour retrouver des fichiers perdus par erreur et dont on ne s'aperçoit pas tout de suite
- Stocker les média de sauvegarde à différents endroits le plus loin possible du système de sauvegarde (autres bâtiments, villes)
- Penser à crypter les sauvegardes en milieu sensible
- Ne pas oublier que les média ne sont pas éternels ( $\approx 10$  ans)



pour les bandes magnétiques)

- Que faire de média dont plus aucun lecteur n'existe ? Légende disant qu'il existe un service de l'État qui possède tous les types de lecteurs possibles au cas où...



- Choisir un système matériel : DAT (12 Go et 1 Mo/s), 8mm, AIT, DLT, DTF, SD-3, D-1, D-2 (330 Go et 15 Mo/s)
  - RAIDs de bandes ( $\nearrow$  tolérance,  $\nearrow$  débit,  $\nearrow$  capacité), automates de plein de bandes
  - Choisir une logiciel de sauvegarde
    - ▶ Outils bruts par partition et par machine :  
`ufsdump/ufsrestore,...`
    - ▶ Outils automatisant la sauvegarde d'un réseau et utilisant des outils bruts
      - *Solstice Backup*
      - Amanda ([www.amanda.org](http://www.amanda.org)) gratuit et utilisé au CRI & IAR2M : 80 partitions (1 centaine de Go) sauvegardées chaque nuit sur 1 cassette 8 mm de 7 Go,
- Utiliser un outil de ce type !



- Choisir une politique de sauvegarde :
  - ▶ Faire un état des lieux des machines et des disques
  - ▶ Quoi sauvegarder ? Tout si possible ! En priorité les partitions utilisateurs avant celles du système (récupérables en gros sur CD)
  - ▶ Quand sauvegarder ?
  - ▶ Durée nécessaire aux sauvegardes
  - ▶ Centralisation du système de sauvegarde
  - ▶ Débit du réseau : ne pas écrouler le réseau pendant le fonctionnement normal
  - ▶ Éviter de faire des sauvegardes sur des systèmes de fichiers très actifs ↵ sauvegardes la nuit. En théorie, devrait être en mode mono-utilisateur...
  - ▶  Problème de cohérence avec des systèmes de base de



donnée : sauvegarder plutôt une image instantanée de la base plutôt que la base de donnée

- ▶ Utiliser astucieusement les sauvegardes incrémentales : plutôt que de sauvegarder toute une partition, on la sauve une fois complètement et ensuite on ne sauve que les différences par rapport à la sauvegarde complète. Récursion possible sur le concept. Compromis à trouver entre sécurité, facilité de restauration et place occupée sur bande. De toute manière, besoin supérieur à l'incrément quotidien.

Exemple à allocation statique pour une partition disque

- Vendredi soir : niveau 0
- Lundi soir : niveau 1
- Mardi soir : niveau 2
- Mercredi soir : niveau 3
- Jeudi soir : niveau 4



- Si on perd 1 niveau 0 on perd la semaine
- Si on veut récupérer 1 jeudi il faut lire 5 bandes

Rotation d'un jour sur toutes les partitions de manière cyclique pour répartir la quantité (niveaux 0) sur les bandes de sauvegarde ↗ rapidement compliqué... Cf. outils



- Pas d'accès direct à l'information
  - ▶ Fichiers séparés par des marques de fin de fichier
  - ▶ Possible de sauter plus rapidement d'une marque à l'autre
  - ▶ Concaténation possible de fichiers si pas de rembobinage
- Utilisation de la commande `mt` pour contrôler la bande
- Conducteur matériel : `/dev/rmt/x y [n]` *Raw Magnetic Tape*  
(blocs de 512 octets)
  - $x$  : numéro dérouleur
  - $y$  : lettre de densité (l, m, h, u) plus de compression c optionnelle
  - $n$  : optionnel indiquant qu'on ne veut pas de rembobinage automatique



- Outils de sauvegarde de base de UFS (gestion via inodes...)
  - Sauvegarde d'abord la table des matières ↗  si gros changements pendant la sauvegarde...
  - Niveaux incrémentaux de 0 à 9
  - `ufsdump` pour la sauvegarde d'une liste de fichiers ou 1 système de fichier. Il faut préciser la taille de la bande (même si sauvegarde dans un fichier...) car possibilité d'utiliser plusieurs bandes
    - ▶ `ufsdump 0uf /dev/rmt/0 / : niveau 0`
    - ▶ `ufsdump 1uf deauville:/dev/rmt/0 /export/home : niveau 1 sur bande distante`
- « u » a pour effet de mettre à jour `/etc/dumpdate` qui note la date des sauvegardes pour chaque niveau d'incrément. Ne sont sauvegardés pour un niveau donné que les fichiers modifiés



après la date des sauvegardes de niveau inférieur

- ufsrestore pour la restauration de fichiers
  - ▶ `ufsrestore rf /dev/rmt/0` : récupère le contenu de la partition sur la bande
  - ▶ `ufsrestore if moret.c0t1d0s7.19981201.1` : récupération en mode interactif

```
deauville-keryell > ufsrestore if moret.c0t1d0s7.19981201.1
```

```
ufsrestore > ls
```

```
.:
```

```
WWW/      users/
```

```
ufsrestore > cd WWW
```

```
ufsrestore > cd conf
```

```
ufsrestore > ls
```

```
./WWW/conf:
```

```
    srm.conf
```

```
ufsrestore > add srm.conf
```

```
ufsrestore > extract
```

```
You have not read any volumes yet.
```

```
Unless you know which volume your file(s) are on you should start  
with the last volume and work towards the first.
```



```
Specify next volume #: 1
set owner/mode for '.'? [yn] n
ufsrestore > quit
```

En général ne pas changer les droits/modes de « . » si restauration ailleurs que dans le répertoire d'origine

- Lors d'une restauration complète, il y a ré-allocation des inodes et il ne faut pas commencer la sauvegarde suivante par une incrémentale mais par une de niveau 0



- Problème : des morceaux du noyau risquent d'être écrasés lors de la restauration...
- Démarrer sans utiliser ces disques : depuis le CD-ROM, depuis un serveur d'installation (toujours en avoir un sous le coude), en tant que *diskless* depuis le réseau ou depuis un autre disque le cas échéant
- Monter le disque à restaurer dans /mnt et faire la restauration dedans puis démonter
- Comme la sauvegarde ne gère pas le boot, réinstaller la zone de démarrage sur le disque /

```
installboot /usr/platform/`uname -i'/lib/fs/ufs/bootblk /dev/rdsk/\emph{dev}
```



- `ufsdump/ufsrestore` efficace pour UFS Solaris mais non portable
- Outils portables pour transporter une arborescence
- `tar + GNU`
- `cpio` plus portable et multivolume + GNU
- `pax` gère les formats POSIX `cpio` et `ustar`
- Utile pour recopier des arborescences sous Unix
-  si sauvegarde avec des noms absous, restauration avec des noms absolu depuis /. Pour rattraper la sauce : changer la racine du processus de restauration via `chroot...`



- Capture de toute l'information au niveau de /dev/rdsk
- Peut être utile pour une restauration de / avec la zone de démarrage
- Duplication rapide de disquette

volcheck

```
dd if=/vol/dev/aliases/floppy0 of=devcfg-2.7-disquette bs=1440k  
eject
```

puis avec une nouvelle disquette

volcheck

```
fdformat -d -U
```

```
dd if=devcfg-2.7-disquette of=/vol/dev/aliases/floppy0 bs=1440k  
eject
```

- Clonage d'une installation de Windows NT multi-OS via Solaris

```
cd /export/verre4  
rsh roosevelt dd if=/dev/rdsk/c0d0p1 bs=512 | /usr/local/bin/gzip -9 > nt-4.0.server.roosevelt.gz
```



puis installation depuis une autre machine lors de la procédure  
JumpStart

```
mkdir /tmp/verre4
mount palo-alto:/export/verre4 /tmp/verre4
/usr/local/bin/gunzip -c /tmp/verre4/nt-4.0.server.roosevelt.gz | dd bs=512 of=/dev/rdsk/c0d0p1
```



- Fixe des limites à l'usage de chaque disque par utilisateur
  - ▶ Limite *soft* : déclenche un avertissement et laisse un délai de correction
  - ▶ Puis limite *hard* : empêche toute écriture
- Mettre en place les quotas
  - ▶ Au niveau du répertoire racine du système de fichier à contrôler avec les droits de root

```
touch quotas
chmod 600 quotas
```
  - ▶ Rajouter `rq` dans l'option du système de fichier dans `/etc/vfstab` pour la demande au démarrage
  - ▶ `quotaon disque` : démarre sans attendre le prochain reboot
- `edquota utilisateur` modifie les quotas



- quota -v [*utilisateur*] affiche l'état de ses quotas
- repquota Affiche tous les quotas
- Arrêt des quotas avec quotaoff



- Partage de fichiers en réseau par machines hétérogènes de fichiers ou de hiérarchies
- Aide à rendre les machines uniformes par centralisation de l'information
- Protocole portable (Unix, Windows, VMS,...)
- Devices non partageables
- Exportation contrôlée des ressources sur la base d'une hiérarchie et d'un groupe de machine
- Les hiérarchies exportées ne peuvent pas se recouvrir
- Sous Unix exportation d'un système de fichier ou d'une partie
- Version
  - 2 : Version la plus commune



- 3 : À partir de Solaris 2.5, nécessite clients et serveurs V3
  - ▶ Autorise les écritures asynchrones sur disque (ne bloque pas le client)
  - ▶ Macro-requêtes pour diminuer le traffic
  - ▶ Vérification des droits améliorés
  - ▶ Dépasse 8 Ko/paquets
  - ▶ Support des ACL
  - ▶ NFS au dessus de TCP en plus d'UDP (pas spécifique version 3)
  - ▶ Gestion des gros fichiers (plus de 2 Go)
  - ▶ Utilisation dynamique de plusieurs serveurs en cas de panne sur des systèmes de fichier en lecture seule (/usr/local,...)
  - ▶ WebNFS RFC 2054. RFC 2225 : meilleur débit que HTTP, pas de sur-coût à la FTP



- ▶ Sécurité Kerberos V5 et RPCSEC\_GSS (API)



- À la main avec

```
share [-F nfs] [-o options] [-d description] chemin
```

Quelques *options*

- ▶ **index=*file*** Renvoie *file* au lieu de la liste des fichiers du répertoire lors d'un accès via URL `nfs://...`
- ▶ **nosuid** Empêche la création de programme setuid et setgid
- ▶ **public** Définit ce système de fichier comme étant celui racine consultable par un butineur avec `nfs://`
- ▶ **ro[=*liste-machines*]** Exporte en lecture seulement à une liste de machine ou à  tout l'univers
- ▶ **root=*liste-machines*** Les machines spécifiées ont les droits de  root
- ▶ **rw[=*liste-machines*]** Exporte en lecture et écriture à une liste de machine ou à  tout l'univers



- ▶ `sec=liste-modes` système d'authentification utilisé
- ▶ `liste-machines` consiste en des noms de machine, des netgroup, des suffixe DNS (à condition d'avoir dns en tête de hosts dans /etc/nsswitch.conf, des (sous-)réseaux. - est utilisé comme privatif dans les accès
- Automatiquement au démarrage en mettant les lignes précédentes dans le fichier /etc/dfs/dfstab. Prise en compte des modifications avec shareall
- Vérifier que le serveur NFS est lancé avec  
`/etc/init.d/nfs.server start`  
Normalement lancé au démarrage si existence de  
`etc/dfs/dfstab`



- Avec `mount -F nfs`, `/etc/vfstab` ou l'auto-monteur
- Syntaxe `mount [-F nfs] [options_génériques] [-o options_spécifiques] [-O ] ressource mount_point`
  - *ressource* à monter
    - ▶ *host :pathname*
    - ▶ *nfs://host [:port]/pathname*
    - ▶ Une liste de ressources séparées par des virgules : système de tolérance aux pannes (mais en lecture seulement)
  - Quelques options
    - ▶ `fg` Bloque jusqu'à ce que le montage soit fait (défaut)
    - ▶ `bg` Réessaye en tâche de fond si le montage échoue
    - ▶ `hard` Réessaye l'entrée-sortie jusqu'à ce que le serveur réponde
    - ▶ `soft` Renvoie un code d'erreur si le serveur ne répond pas.





Si un programme ne teste pas les retours d'erreurs...

- ▶ `intr` Accepte d'interrompre les un accès hard bloquant en tapant `^C` au clavier
  - ▶ `nointr` Le contraire
  - ▶ `public` Passe par le répertoire exporté comme étant public
  - ▶ `ro` Monte en lecture seule
  - ▶ `rw` Monte en lecture/écriture
  - ▶ `suid` Autorise l'exécution de programmes en setuid.  
Comportement par défaut...
  - ▶ `nosuid` L'inverse
- 
- `nfssstat` permet d'avoir diverses statistiques sur le fonctionnement de NFS en fonction des options
  - `showmount` permet d'avoir des informations sur les clients ou les partitions exportées d'un serveur donné



- Trop de montages saturent le système
- ↗ Montage et démontage à la demande à partir de tables centralisées
- Tout ne peut pas être connu dès le départ
- Uniformisation des machines
- Éviter de devoir être `root` pour monter des répertoires (distants)
- Généralisation de l'espace de nommage  
`/net/machine/fichier`
- Changement de tous les montages de toutes les machines de manière centralisée
- ↗ AutoFS



- Une *carte* peut être un fichier /etc/carte ou une *carte* NIS, NIS+, FNS en fonction de /etc/nsswitch.conf
- Master map : /etc/auto\_master associe des répertoires avec des cartes

```
keryell@voltaire ~: more /etc/auto_master
# Master map for automounter
#
+auto.master
keryell@voltaire ~: ypcat -k auto.master
/home auto.home -hard,bg,intr
/xfn -xfn
/net -hosts           -nosuid,nobrowse
/- auto.direct
```

- ▶ -hosts signifie de monter les fichiers (exportés !) de n'importe quelle machine demandée dans le répertoire



- ▶ -xfn signifie de monter les ressources spécifiées par leur nom FNS
- ▶ /- évite d'associer une carte avec un répertoire
- Direct map : dirige AutoFS directement vers des systèmes de fichiers à partir d'un nom de répertoire

```
keryell@voltaire ~: ypcat -k auto.direct  
/var/mail -rw,hard,intr,actimeo=0      palo-alto:/export/verre2,  
/users -rw,hard,intr      palo-alto:/export/verre1
```

- Indirect map : monte des systèmes de fichier à partir d'une clé

```
keryell@voltaire ~: ypcat -k auto.home  
palo-alto / &:/export    /verre1 &:/export/verre1 \  
/verre2 &:/export/verre2 /verre3 &:/export/verre3 \  
/verre4 &:/export/verre4
```

- ▶ & rappelle le nom de la clé



- ▶ \* accepte n'importe quelle clé
  - \* & : /export
- Possibilité d'utiliser des variables pour changer localement des montages : \$ARCH, \$CPU, \$HOST, \$OSNAME, \$OSREL, \$OSVERS
- Possibilité de mettre des poids dans des montages redondants
-  Ne pas mettre au même endroit exportation et montage local mais plutôt par exemple /export/ ... et /home/ .... AutoFS utilise directement un LOFS
- automount contrôle le comportement d'automountd (relecture d'auto\_master, changement du temps de raffraîchissement)



- Trop d'informations spécifiques à 1 site pour être dupliquées sur chaque machine
- Besoin de pouvoir remettre à jour les informations instantanément sur toutes les machines
- Différents systèmes possibles sur Solaris sélectionnés par type de ressources selon /etc/nsswitch.conf
  - Fichiers : à dupliquer sur toutes les machines...
  - NIS
    - ▶ Distribue des tableaux associatifs
    - ▶ Pas très sécurisé en local car les mots de passe cryptés sont accessibles pour essais de craquage
    - ▶ Système très répandu
  - NIS+
    - ▶ Organisation hiérarchique de l'espace de nommage



- ▶ Distribue tout type d'information (binaires...)
  - ▶ Serveurs secondaires pour répartition de la charge et secours
  - ▶ Mécanisme d'authentification et d'autorisation
  - ▶ Système plutôt Sun
- DNS : système de nommage des machines sur Internet.  
Plutôt réservé à la ressource hosts
  - Federated Naming Service (FNS) : méthode de nommage fédérant NIS+, NIS, fichiers, DNS, et X.500/LDAP



- Choisir un nom de domaine NIS unique sur son réseau. Par exemple nom de domaine Internet
- Vérifier que `/etc/nodename` contient bien le nom de la machine
- Mettre dans `/etc/defaultdomain` le nom du domaine
- Choisir le serveur NIS maître et les serveurs esclaves
- Faire `/usr/sbin/ypinit -m` sur le serveur maître qui demande la liste de tous les serveurs
- Si `/etc/resolv.conf` existe sur le serveur les NIS font suivre les requêtes hosts au DNS
- Faire `/usr/sbin/ypinit -c` sur le serveur esclave qui demande la liste de tous les serveurs. Donner son propre nom en premier, puis le serveur maître puis les autres
- Sur les clients faire un `ypinit -c` et donner la liste des serveurs,



les plus proches en premier

- Remplacer les `/etc/nsswitch.conf` avec le contenu de `/etc/nsswitch.nis`
- `/var/yp/Makefile` contrôle les tables NIS exportées. Éditer pour rajouter ses propres ressources. Faire `make` dans `/var/yp` pour propager les ressources lorsque les fichiers de référence sont modifiés
- `/var/yp/securenets` restreint l'accès des NIS
- `ypcat [-k] table` permet d'afficher le contenu d'une table NIS



- Système de *spool* permettant d'entasser les impressions en différé
- Imprimantes locales : impressions sont mises en attente de libération de l'imprimante dans une zone de spool  
`/var/spool/lp`
- Imprimantes distantes : mise en attente dans `/var/spool/lp` et envoi par protocole BSD ou TCP *pass-through* sur le serveur de l'imprimante distante, voire l'imprimante elle-même (mais implémentation parfois fantaisiste des protocoles réseau...)
- ↵ Penser à avoir de gros `/var/spool/lp...`
- Installation par outils graphiques (`solstice AdminSuite` ou `admintool`) ou textuels (`lpadmin`)



Base de donnée dans

- `$HOME/.printers`
- `/etc/printers.conf` (modifiable par outils graphiques et `lpdadmin`)
- Ressource NIS `printers.confbyname`
- Ressource NIS+ et FNS
- Accès direct via  
`lpr -P serveur : imprimante fichier`

Choix de l'imprimante par défaut

- `$PRINTER` et `$LPDEST`
- Recherche d'une imprimante `_default` dans les fichiers précédents



- `lpstat -p imprimante` donne des informations sur l'*imprimante*
- `accept imprimante` accepte les demandes d'impression
- `reject -r une-raison imprimante` n'accepte plus les demandes d'impression
- `enable imprimante` permet l'impression
- `disable -r une-raison imprimante` arrête l'impression
- `cancel -u user | request-id-list | imprimante` annule des travaux d'impression
- `lpmove impr-src impr-dst` transfère les impressions depuis une imprimante vers une autre en faisant un `reject impr-src`
- `lpadmin -A` contrôle la marche à suivre en cas d'alerte



- `lpadmin -u allow:user-list` autorise des utilisateurs
- `lpadmin -u deny:user-list` interdit à des utilisateurs
- `/usr/lib/lp/lpshut` arrête le système d'impression
- `/usr/lib/lp/lpsched` démarre le système d'impression
- Messages de log dans `/var/lp/logs/`



- Besoin de lancer des tâches à un instant précis : sauvegardes, nettoyage nocturne, miroir de site WWW,...
- Géré par le système cron
- Le résultat de la tâche (stdout et stderr) est renvoyé par mail
- Répétitions : crontab

► Format fichier

*minutes secondes jour-du-mois mois*

*jour-de-la-semaine commande*

« \* » pour ignorer un champ

Commande exécutée par un shell (sh par défaut).

« % » représente un caractère fin de ligne

► **crontab -l [utilisateur]** Affiche la liste des travaux

```
22 03 * * * crontab -l > $HOME/bib/crontab.`hostname`
```

```
25 01 * * * rsh deauville $HOME/bin/gen_calendar
```

```
33 03 * * * cp /var/spool/mail/$USER $HOME/lettres/arrivee/mail.`/usr/bin/date +\%w`
```



- ▶ crontab -e pour éditer le fichier
- ▶ Contrôle des autorisations par utilisateur via /etc/cron.d/cron.allow et /etc/cron.d/cron.deny
- Une seule fois : at
  - ▶ at -m *heure-date* demande un script et envoie un mail après exécution
  - ▶ at -l affiche ses travaux en attente
  - ▶ atq affiche les travaux en attente
  - ▶ at -r efface une demande de tâche
  - ▶ Variante batch : empile la tâche sur la pile des travaux à exécuter
  - ▶ Contrôle des autorisations par utilisateur via /etc/cron.d/at.allow et /etc/cron.d/at.deny



- Administrer un domaine : faire des actions à distance
- `rlogin machine` permet de se connecter sur une machine distante. Aucun mot de passe demandé si autorisé par fichier (distant...) `$HOME/.rhost` ou `/etc/hosts.equiv`
- `rsh machine commande` exécute une commande à distance dans son `$HOME`. Autorisation par les mêmes fichiers que `rlogin`
- `rcp [-rp] [machine-src:]fichiers [machine-dst:]rep` utilise le protocole `rsh` pour faire un `cp` de fichiers.  
Méta-caractères acceptés mais les protéger pour action à distance
-  Ne pas laisser des `/etc/hosts.equiv` OU `.rhost` trop permissifs. Vérifier leur contenu
- `ping machine` teste si les paquets réseau font bien l'aller-retour



- `rusers [-l]` demande et affiche la liste des utilisateurs connectés aux machines du réseau
- `rup` demande et affiche la charge des machines du réseau
- Récupérer une copie d'écran à distance (`xwd -root si X11`)



- *File Transfer Protocole*
- `ftp [-i] machine` ouvre une connexion
- Le nom de login `anonymous` signifie une connexion anonyme (on donne son adresse de mail comme mot de passe par convention) si un compte anonyme existe
- Commandes classiques `ls`, `cd`, `mkdir`
- `lcd` change de répertoire localement
- `get/put` pour transférer un fichier
- `mget/mput` pour transférer une liste de fichier (sans demander confirmation si `ftp -i`)
- `bin` demande ) des transferts en binaire (pas de traduction de format de fin de ligne dans fichiers textuels, etc)



- quit ou bye pour arrêter
- Emballé maintenant dans les URL `ftp://machine /` (anonymous)



- Démarrage d'un Unix simplifié depuis
  - ▶ CD-ROM
  - ▶ Réseau
  - ▶ Disque
  - ▶ Bande

Besoin d'une disquette de démarrage sur PC pour récupérer un minimum de drivers

- Installation de type
  - ▶ WebStart : à partir d'un butineur WWW pour installation assez standard
  - ▶ Installation interactive : permet de faire une installation plus précise
  - ▶ Installation personnalisée JumpStart : automatique !



- Possibilité d'installation depuis rien ou mise à niveau version Solaris
- Peut préserver le contenu des disques utilisateurs. Ne dispense pas des sauvegardes...
- Intérêt installation JumpStart par réseau : tout automatique sans avoir même à se déplacer.Sur Sun :  
*rsh machine 'reboot net - install'*



- Contiendra le système et de quoi faire démarrer une machine via le réseau
- Création dans /export/Solaris7\_CD-ROM après insertion du CD-ROM Solaris 7

```
cd /cdrom/sol_7_x86/s2/Solaris_2.7/Tools  
. ./setup_install_server /export/Solaris7_CD-ROM  
mkdir /usr/local/jumpstart  
cp -p /cdrom/sol_7_x86/s2/Solaris_2.7/Misc/jumpstart_sample/* \  
/usr/local/jumpstart
```



- /usr/local/jumpstart/rules décrit les profils d'installation à utiliser : installation à faire en fonction des disques, du type de machine, du nom,...

```
# Les stations du maste're :  
any - commence_installation stations termine_installation
```

- Un fichier de profil décrit l'installation :

```
install_type      initial_install  
system_type       standalone  
# Il n'y a pas de support OEM pour x86 :  
cluster           SUNWCall  
fdisk all solaris delete  
fdisk all solaris 2000  
partitioning     explicit  
filesys          c0d0s0 200 /  
filesys          c0d0s6 auto /usr
```



```
# Pour avoir plein de swap :  
filesys      c0d0s1 free swap  
filesys      palo-alto:/opt - /opt ro  
filesys      palo-alto:/usr/local - /usr/local ro
```



- Nécessité de configuration de paramètres des machines
- Utilisation d'un système de nommage
- Fichier /usr/local/jumpstart/sysidcfg

root\_password=u5nhT.xHroW5U

system\_locale=fr.ISO8859-15

timezone=MET

timeserver=ntp-server.cri.ensmp.fr



- Chaque OS a des contraintes sur les tailles de partition `fdisk` ↗  
laisser les OS se débrouiller avec une installation indépendante  
sur une machine de référence
- Installer dans les 4 partitions
  - ▶ Windows NT
  - ▶ Windows 95 ou 98
  - ▶ Linux (non testé : laissé comme projet élève...)
  - ▶ Solaris 7
- Ensuite capturer les paramètres de partitionnement `fdisk`  
depuis Solaris 7

```
fdisk -W /usr/local/jumpstart/fdisk_multiple /dev/rdsk/c0d0p0
```

et copier une image des partitions autres que Solaris depuis le  
serveur palo-alto

```
cd /export/verre4
```



```
rsh miromesnil dd if=/dev/rdsck/c0d0p1 bs=512 | \
/usr/local/bin/gzip -9 > nt-4.0.miromesnil.gz
rsh miromesnil dd if=/dev/rdsck/c0d0p2 bs=512 | \
/usr/local/bin/gzip -9 > windows-95.miromesnil.gz
```

- On fera confiance au système multi-boot altruiste de Solaris



- Script de démarrage avant le début de l'installation

/usr/local/jumpstart/commence\_installation :

```
# /usr/local/jumpstart est monté dans /tmp/install_config  
df  
pwd  
# Partitionne le disque pour NT & 98 en plus :  
/sbin/fdisk -F /tmp/install_config/fdisk_multiple /dev/rdsck0d0p0
```

- Script de fin après l'installation

/usr/local/jumpstart/termine\_installation :

```
# À la fin de l'installation de Sun,  
# le futur / réside en fait dans /a,  
# / n'étant que temporaire pour l'installation.  
set -vx  
# Comme automount ne tourne pas, ni NIS ou DNS :  
/usr/sbin/mount -F nfs 10.2.16.200:/usr/local /a/usr/local  
# Fait croire à cfengine qu'il est dans /  
# alors qu'il est dans /a :  
/usr/sbin/chroot /a /usr/local/sbin/cfengine \
```



```
-v -f /usr/local/share/cfengine/cfengine.conf  
# Met le boot par défaut sur le disque Solaris  
# pour un démarrage automatique :  
eeprom bootpath=/isa/ata@1,1f0/cmdk@0,0:a  
## A French Keyboard :  
eeprom kbd-type=French  
mkdir /tmp/verre4  
mount palo-alto:/export/verre4 /tmp/verre4  
  
echo Installation of Windows NT at `date`...  
/a/usr/local/bin/gunzip -c /tmp/verre4/nt-4.0.miromesnil.gz  
| dd bs=512 of=/dev/rdsck/c0d0p1  
echo Installation of Windows NT ended at `date`.  
  
echo Installation of Windows 95 at `date`...  
/a/usr/local/bin/gunzip -c /tmp/verre4/windows-95.miromesnil.gz  
| dd bs=512 of=/dev/rdsck/c0d0p2  
echo Installation of Windows 95 ended at `date`.
```



- La variable \$lstations contient la liste de toutes les machines clientes du mastère
- Vérifier que /etc/hosts ou le DNS contient le nom et les numéros IP des machines à installer et /etc/ethers contient leur numéros Ethernet (pour RARP)
- Création des configuration de démarrage des installations sur le serveur palo-alto

```
cd /export/Solaris7_CD-ROM/Solaris_2.7/Tools  
for i in $lstations; do ./add_install_client  
    -c palo-alto:/export/local/jumpstart  
    -p palo-alto:/export/local/jumpstart  
    -s palo-alto:/export/Solaris7_CD-ROM $i i86pc;  
done
```

- Lancer les installations JumpStart sur les machines avec la disquette Solaris 7 sur PC ou boot net - intall sur Sun



- <http://www.iu.hioslo.no/~mark/cfengine.html>
- Automatise des tâches d'administration à partir de fichiers déclaratifs
- /usr/local/share/cfengine contient les fichiers de configuration
- /usr/local/share/conf contient les fichiers de référence
- Extrait de /usr/local/share/cfengine/cf.main

control:

```
# Where the reference files are found:  
shared_conf = ( /usr/local/share/conf )
```

shellcommands:

```
RelanceSyslog:::  
    /bin/kill -1 `cat /etc/syslog.pid`
```

```
solaris2_7.DemarreAccounting::
```

```
# Démarre l'accounting :  
/etc/init.d/acct start
```

links:



```
any::  
  # Utilise nofile=force car certaines cibles  
  # n'existent pas lors de  
  # l'installation du système :  
    /projects ->! /users/projects  
    nofile=force  
  
  # Met l'accounting en marche :  
solaris2_7::  
  /etc/rc2.d/S22acct ->! /etc/init.d/acct  
  /etc/rc0.d/K22acct ->! /etc/init.d/acct  
  
InstallSendmail::  
  /usr/lib/sendmail ->! /usr/local/lib/sendmail  
  
InstallSendmail.solaris2_7.palo_alto::  
  /etc/mail/sendmail.cf ->!  
  /usr/local/share/mail/iar2m.ensmp.fr.cf  
  
copy:  
  any::  
    # Déclare la route par défaut :  
    /usr/local/share/conf/defaultrouter  
      dest=/etc/defaultrouter
```



```
type=byte

# Déclare les netgroups :
/usr/local/share/conf/netgroup
dest=/etc/netgroup
type=byte

editfiles:
# Les exportations :
viroflay|rhune:: {
    # Exporte /export :
/etc/dfs/dfstab
AutoCreate
AppendIfNoSuchLine \
share -F nfs -o rw=caii -d $(dblquote)Le disque interne$(dblquote) /export
DefineClasses LanceExportfs
}
```



- Unix est un système robuste et éprouvé : vieux mais a évolué
  - Couvre tous les types d'ordinateur
  - Ouvert : accepte les standards
  - Énormément de possibilités
  - Extensible : rajout de modules et de matériel à chaud
  - Très configurable
  - Administration à plusieurs niveaux en fonction des besoins : basique avec interface graphique jusqu'aux systèmes très spécialisés avec les fichiers textuels
  - Installation automatique avec installation de plusieurs OS sur la même machine
- ~~> Travaux pratiques



# List of Slides

- 1 Introduction
- 2 Fragments d'archéologie
- 9 Sources d'information
- 11 Plan
- 12 Le démarrage
- 16 Niveaux de fonctionnement
- 20 Démarrage en mode mono-utilisateur
- 21 Changer un niveau de fonctionnement
- 22 Lancer des choses au démarrage
- 25 La hiérarchie de fichier dans Solaris 7
- 27 La hiérarchie de /
- 37 La hiérarchie de /usr
- 43 La hiérarchie de /export
- 45 Outils d'administration
- 46 Admintool
- 47 Solstice
- 48 Compte utilisateur
- 51 Le fichier /etc/passwd
- 53 Fichier /etc/shadow
- 55 Le fichier /etc/group
- 57 Administration des comptes
- 59 Répertoires utilisateurs
- 60 Shells et fichiers de login
- 63 Variables d'environnement
- 65 Gestion des média amovibles
- 66 Insertion médium amovible
- 68 Éjection médium amovible
- 69 Formattage disquettes
- 71 Paquets de logiciels
- 73 Automatisation de l'installation de paquets
- 75 Patches
- 76 Ajouter du matériel
- 77 Rajout de périphérique
- 78 Information sur la configuration matérielle



- |   |   |
|---|---|
| 79 ( Disques magnétiques )                    | 121 Organisation de sauvegardes                         |
| 81 Partitionnement des disques                | 125 Dérouleurs de bande                                 |
| 84 Nommage des disques logiques               | 126 <code>ufsdump/ufsrestore</code>                     |
| 87 Utilitaire <code>format</code>             | 129 Restauration de <code>/</code> et <code>/usr</code> |
| 88 <code>format</code> à l'œuvre              | 130 Transport portable de système de fichiers           |
| 91 Étape <code>fdisk</code> sur PC            | 131 Clonage de système de fichiers                      |
| 93 Systèmes de fichiers sous Solaris 7        | 133 Quotas : contrôle de l'usage des disques            |
| 97 Montage/démontage d'un système de fichiers | 135 Network File System — NFS                           |
| 98 Table des fichiers montés                  | 138 Exportation de fichiers                             |
| 100 Montage de systèmes de fichiers           | 140 Montage de fichiers via NFS                         |
| 102 Démontage                                 | 142 Auto-monteur  |
| 103 UFS journalisé                            | 143 Cartes de montage                                   |
| 108 Vérification d'un système de fichiers     | 146 Systèmes de nommage                                 |
| 111 Système de fichiers cache (CacheFS)       | 148 NIS   |
| 112 Mise en œuvre CacheFS                     | 150 Système d'impression                                |
| 115 Rajouter de la mémoire virtuelle          | 151 Rajouter une imprimante — client                    |
| 118 Fichiers temporaires en mémoire — TMPFS   | 152 Contrôle imprimante                                 |
| 119 Nécessité des sauvegardes                 | 154 Tâches planifiées                                   |

## IAR2M — Administration Unix —Conclusion—



156	Interaction avec d'autres systèmes	166	Clonage autres OS sur PC
158	ftp	168	Scripts d'installation
160	Procédures d'installation	170	Installation du boot JumpStart des machines
162	Création d'un serveur d'installation	171	Outil de configuration GNU cfengine
163	Créer un profil de configuration	174	Conclusion
165	Fichier d'identification sysidcfg	175	Table des matières

