

Dernière mise-à-jour : 2020/01/30 03:28

# SO213 - Gestion du matériel et des Performances

## Gestion des Paramètres du Matériel

### Fichiers Spéciaux

Dans l'ordinateur les périphériques sont reliés à un **contrôleur** qui communique avec le processeur à l'aide d'un **bus**. Le contrôleur ainsi que les périphériques nécessitent des pilotes. Sous Solaris, les pilotes sont généralement fournis sous la forme d'un **module**. Chaque périphérique est représenté par un **fichier spécial** dans le répertoire **/dev** et c'est dans ce fichier que le système trouve les informations nécessaires pour s'adresser au pilote.

Consultez le contenu du répertoire /dev :

```
# ls -l /dev
total 494
lrwxrwxrwx 1 root      root          26 janv  3 2010 agpgart -> ../devices/agpgart:agpgart
lrwxrwxrwx 1 root      other         30 janv  3 2010 allkmem -> ../devices/pseudo/mm@0:allkmem
lrwxrwxrwx 1 root      other         27 janv  3 2010 arp -> ../devices/pseudo/arp@0:arp
lrwxrwxrwx 1 root      root          25 janv  3 2010 bl -> ../devices/pseudo/bl@0:bl
drwxr-xr-x 3 root      root          512 janv  3 2010 brand
lrwxrwxrwx 1 root      root          28 janv  3 2010 ce -> ../devices/pseudo/clone@0:ce
drwxr-xr-x 4 root      root          512 janv  3 2010 cfg
lrwxrwxrwx 1 root      other         31 janv  3 2010 conslog -> ../devices/pseudo/log@0:conslog
lrwxrwxrwx 1 root      other         30 janv  3 2010 console -> ../devices/pseudo/cn@0:console
drwxr-xr-x 3 root      root          512 janv  3 2010 cpu
lrwxrwxrwx 1 root      root          33 janv  3 2010 crypto -> ../devices/pseudo/crypto@0:crypto
lrwxrwxrwx 1 root      root          39 janv  3 2010 cryptoadm -> ../devices/pseudo/cryptoadm@0:cryptoadm
lrwxrwxrwx 1 root      root          33 janv  3 2010 diskette -> ../devices/isa/fdc@1,3f0/fd@0,0:c
```

```

lrwxrwxrwx 1 root root          33 janv 3 2010 diskette0 -> ../devices/isa/fdc@1,3f0/fd@0,0:c
drwxr-xr-x 2 root sys           1024 janv 3 2010 dsk
drwxr-xr-x 3 root root          512 janv 3 2010 dtrace
lrwxrwxrwx 1 root root          29 janv 3 2010 dump -> ../devices/pseudo/dump@0:dump
lrwxrwxrwx 1 root root          32 janv 3 2010 e1000g -> ../devices/pseudo/clone@0:e1000g
lrwxrwxrwx 1 root root          39 janv 3 2010 e1000g0 -> ../devices/pci@0,0/pci8086,1e@3:e1000g0
lrwxrwxrwx 1 root root          29 janv 3 2010 ecpp0 -> ../devices/isa/lp@1,378:ecpp0
lrwxrwxrwx 1 root root          33 août 29 09:55 fb -> /devices/pci@0,0/display@2:text-0
lrwxrwxrwx 1 root root          10 janv 3 2010 fb0 -> fbs/text-0
drwxr-xr-x 2 root root          512 janv 3 2010 fbs
dr-xr-xr-x 2 root root          2064 août 30 15:49 fd
lrwxrwxrwx 1 root root          4 janv 3 2010 fd0 -> fd0c
lrwxrwxrwx 1 root root          33 janv 3 2010 fd0a -> ../devices/isa/fdc@1,3f0/fd@0,0:a
lrwxrwxrwx 1 root root          33 janv 3 2010 fd0b -> ../devices/isa/fdc@1,3f0/fd@0,0:b
lrwxrwxrwx 1 root root          9 janv 3 2010 fd0c -> diskette0
lrwxrwxrwx 1 root root          25 janv 3 2010 fm -> ../devices/pseudo/fm@0:fm
lrwxrwxrwx 1 root root          30 janv 3 2010 fssnapctl -> ../devices/pseudo/fssnap@0:ctl
lrwxrwxrwx 1 root other         29 janv 3 2010 ibd -> ../devices/pseudo/clone@0:ibd
lrwxrwxrwx 1 root other         29 janv 3 2010 icmp -> ../devices/pseudo/icmp@0:icmp
lrwxrwxrwx 1 root other         31 janv 3 2010 icmp6 -> ../devices/pseudo/icmp6@0:icmp6
lrwxrwxrwx 1 root other         25 janv 3 2010 ip -> ../devices/pseudo/ip@0:ip
lrwxrwxrwx 1 root other         27 janv 3 2010 ip6 -> ../devices/pseudo/ip6@0:ip6
lrwxrwxrwx 1 root root          30 janv 3 2010 ipauth -> ../devices/pseudo/ipf@0:ipauth
lrwxrwxrwx 1 root root          27 janv 3 2010 ipf -> ../devices/pseudo/ipf@0:ipf
lrwxrwxrwx 1 root root          27 janv 3 2010 ipl -> ../devices/pseudo/ipf@0:ipf
lrwxrwxrwx 1 root root          32 janv 3 2010 iplookup -> ../devices/pseudo/ipf@0:iplookup
lrwxrwxrwx 1 root root          29 janv 3 2010 ipnat -> ../devices/pseudo/ipf@0:ipnat
lrwxrwxrwx 1 root root          30 janv 3 2010 ipscan -> ../devices/pseudo/ipf@0:ipscan
lrwxrwxrwx 1 root other         35 janv 3 2010 ipsecah -> ../devices/pseudo/ipsecah@0:ipsecah
lrwxrwxrwx 1 root other         37 janv 3 2010 ipsecesp -> ../devices/pseudo/ipsecesp@0:ipsecesp
lrwxrwxrwx 1 root root          31 janv 3 2010 ipstate -> ../devices/pseudo/ipf@0:ipstate
lrwxrwxrwx 1 root root          30 janv 3 2010 ipsync -> ../devices/pseudo/ipf@0:ipsync
lrwxrwxrwx 1 root other         31 janv 3 2010 kbd -> ../devices/pseudo/conskbd@0:kbd
lrwxrwxrwx 1 root root          35 janv 3 2010 kdmouse -> ../devices/isa/i8042@1,60/mouse@1:l

```

```

lrwxrwxrwx 1 root other 35 janv 3 2010 keysock -> ../devices/pseudo/keysock@0:keysock
lrwxrwxrwx 1 root root 29 janv 3 2010 kmdb -> ../devices/pseudo/kmdb@0:kmdb
lrwxrwxrwx 1 root other 27 janv 3 2010 kmem -> ../devices/pseudo/mm@0:kmem
lrwxrwxrwx 1 root 29 janv 3 2010 kssl -> ../devices/pseudo/kssl@0:kssl
lrwxrwxrwx 1 root root 31 janv 3 2010 kstat -> ../devices/pseudo/kstat@0:kstat
lrwxrwxrwx 1 root other 31 janv 3 2010 ksyms -> ../devices/pseudo/ksyms@0:ksyms
lrwxrwxrwx 1 root root 30 janv 3 2010 llc1 -> ../devices/pseudo/clone@0:llc1
lrwxrwxrwx 1 root root 30 janv 3 2010 llc2 -> ../devices/pseudo/clone@0:llc2
lrwxrwxrwx 1 root root 37 janv 3 2010 lockstat -> ../devices/pseudo/lockstat@0:lockstat
lrwxrwxrwx 1 root root 28 janv 3 2010 lofictl -> ../devices/pseudo/lofi@0:ctl
lrwxrwxrwx 1 root other 27 janv 3 2010 log -> ../devices/pseudo/log@0:log
lrwxrwxrwx 1 root root 35 janv 3 2010 logindmux -> ../devices/pseudo/clone@0:logindmux
lrwxrwxrwx 1 root root 29 janv 3 2010 lp1 -> ../devices/isa/lp@1,378:ecpp0
drwxr-xr-x 2 root root 512 janv 3 2010 md
lrwxrwxrwx 1 root other 26 janv 3 2010 mem -> ../devices/pseudo/mm@0:mem
lrwxrwxrwx 1 root root 32 janv 3 2010 mouse -> ../devices/pseudo/consms@0:mouse
lrwxrwxrwx 1 root other 33 janv 3 2010 msglog -> ../devices/pseudo/sysmsg@0:msglog
lrwxrwxrwx 1 root root 27 janv 3 2010 nca -> ../devices/pseudo/nca@0:nca
lrwxrwxrwx 1 root other 27 janv 3 2010 null -> ../devices/pseudo/mm@0:null
lrwxrwxrwx 1 root 11 janv 3 2010 nvidia0 -> fbs/nvidia0
lrwxrwxrwx 1 root other 11 janv 3 2010 nvidia1 -> fbs/nvidia1
lrwxrwxrwx 1 root other 12 janv 3 2010 nvidia10 -> fbs/nvidia10
lrwxrwxrwx 1 root other 12 janv 3 2010 nvidia11 -> fbs/nvidia11
lrwxrwxrwx 1 root other 12 janv 3 2010 nvidia12 -> fbs/nvidia12
lrwxrwxrwx 1 root other 12 janv 3 2010 nvidia13 -> fbs/nvidia13
lrwxrwxrwx 1 root other 12 janv 3 2010 nvidia14 -> fbs/nvidia14
lrwxrwxrwx 1 root other 12 janv 3 2010 nvidia15 -> fbs/nvidia15
lrwxrwxrwx 1 root other 11 janv 3 2010 nvidia2 -> fbs/nvidia2
lrwxrwxrwx 1 root other 11 janv 3 2010 nvidia3 -> fbs/nvidia3
lrwxrwxrwx 1 root other 11 janv 3 2010 nvidia4 -> fbs/nvidia4
lrwxrwxrwx 1 root other 11 janv 3 2010 nvidia5 -> fbs/nvidia5
lrwxrwxrwx 1 root other 11 janv 3 2010 nvidia6 -> fbs/nvidia6
lrwxrwxrwx 1 root other 11 janv 3 2010 nvidia7 -> fbs/nvidia7
lrwxrwxrwx 1 root other 11 janv 3 2010 nvidia8 -> fbs/nvidia8

```

```

lrwxrwxrwx 1 root other 11 janv 3 2010 nvidia9 -> fbs/nvidia9
lrwxrwxrwx 1 root other 38 janv 3 2010 nvidiactl -> ../devices/pseudo/nvidia@255:nvidiactl
lrwxrwxrwx 1 root other 37 janv 3 2010 openprom -> ../devices/pseudo/openeeprom@0:openprom
lrwxrwxrwx 1 root root 35 janv 3 2010 physmem -> ../devices/pseudo/physmem@0:physmem
lrwxrwxrwx 1 root root 25 janv 3 2010 pm -> ../devices/pseudo/pm@0:pm
lrwxrwxrwx 1 root root 29 janv 3 2010 poll -> ../devices/pseudo/poll@0:poll
lrwxrwxrwx 1 root root 29 janv 3 2010 pool -> ../devices/pseudo/pool@0:pool
lrwxrwxrwx 1 root root 32 janv 3 2010 poolctl -> ../devices/pseudo/pool@0:poolctl
lrwxrwxrwx 1 root root 38 janv 3 2010 power_button -> ../devices/pseudo/power@0:power_button
lrwxrwxrwx 1 root root 27 janv 3 2010 ppm -> ../devices/pseudo/ppm@0:ppm
drwxr-xr-x 2 root root 512 janv 3 2010 printers
lrwxrwxrwx 1 root root 31 janv 3 2010 ptmajor -> ../devices/pseudo/ptm@0:ptmajor
lrwxrwxrwx 1 root root 29 janv 3 2010 ptmx -> ../devices/pseudo/clone@0:ptm
drwxr-xr-x 2 root sys 512 janv 3 2010 pts
lrwxrwxrwx 1 root root 29 janv 3 2010 ptyp0 -> ../devices/pseudo/ptc@0:ptyp0
lrwxrwxrwx 1 root root 29 janv 3 2010 ptyp1 -> ../devices/pseudo/ptc@0:ptyp1
lrwxrwxrwx 1 root root 29 janv 3 2010 ptyp2 -> ../devices/pseudo/ptc@0:ptyp2
lrwxrwxrwx 1 root root 29 janv 3 2010 ptyp3 -> ../devices/pseudo/ptc@0:ptyp3
lrwxrwxrwx 1 root root 29 janv 3 2010 ptyp4 -> ../devices/pseudo/ptc@0:ptyp4
lrwxrwxrwx 1 root root 29 janv 3 2010 ptyp5 -> ../devices/pseudo/ptc@0:ptyp5
lrwxrwxrwx 1 root root 29 janv 3 2010 ptyp6 -> ../devices/pseudo/ptc@0:ptyp6
lrwxrwxrwx 1 root root 29 janv 3 2010 ptyp7 -> ../devices/pseudo/ptc@0:ptyp7
lrwxrwxrwx 1 root root 29 janv 3 2010 ptyp8 -> ../devices/pseudo/ptc@0:ptyp8
lrwxrwxrwx 1 root root 29 janv 3 2010 ptyp9 -> ../devices/pseudo/ptc@0:ptyp9
lrwxrwxrwx 1 root root 29 janv 3 2010 ptypa -> ../devices/pseudo/ptc@0:ptypa
lrwxrwxrwx 1 root root 29 janv 3 2010 ptypb -> ../devices/pseudo/ptc@0:ptypb
lrwxrwxrwx 1 root root 29 janv 3 2010 ptypc -> ../devices/pseudo/ptc@0:ptypc
lrwxrwxrwx 1 root root 29 janv 3 2010 ptypd -> ../devices/pseudo/ptc@0:ptypd
lrwxrwxrwx 1 root root 29 janv 3 2010 ptype -> ../devices/pseudo/ptc@0:ptype
lrwxrwxrwx 1 root root 29 janv 3 2010 ptypf -> ../devices/pseudo/ptc@0:ptypf
lrwxrwxrwx 1 root root 29 janv 3 2010 ptyq0 -> ../devices/pseudo/ptc@0:ptyq0
lrwxrwxrwx 1 root root 29 janv 3 2010 ptyq1 -> ../devices/pseudo/ptc@0:ptyq1
lrwxrwxrwx 1 root root 29 janv 3 2010 ptyq2 -> ../devices/pseudo/ptc@0:ptyq2
lrwxrwxrwx 1 root root 29 janv 3 2010 ptyq3 -> ../devices/pseudo/ptc@0:ptyq3

```

```

lrwxrwxrwx 1 root root 29 janv 3 2010 ptyq4 -> ./devices/pseudo/ptc@0:ptyq4
lrwxrwxrwx 1 root root 29 janv 3 2010 ptyq5 -> ./devices/pseudo/ptc@0:ptyq5
lrwxrwxrwx 1 root root 29 janv 3 2010 ptyq6 -> ./devices/pseudo/ptc@0:ptyq6
lrwxrwxrwx 1 root root 29 janv 3 2010 ptyq7 -> ./devices/pseudo/ptc@0:ptyq7
lrwxrwxrwx 1 root root 29 janv 3 2010 ptyq8 -> ./devices/pseudo/ptc@0:ptyq8
lrwxrwxrwx 1 root root 29 janv 3 2010 ptyq9 -> ./devices/pseudo/ptc@0:ptyq9
lrwxrwxrwx 1 root root 29 janv 3 2010 ptyqa -> ./devices/pseudo/ptc@0:ptyqa
lrwxrwxrwx 1 root root 29 janv 3 2010 ptyqb -> ./devices/pseudo/ptc@0:ptyqb
lrwxrwxrwx 1 root root 29 janv 3 2010 ptyqc -> ./devices/pseudo/ptc@0:ptyqc
lrwxrwxrwx 1 root root 29 janv 3 2010 ptyqd -> ./devices/pseudo/ptc@0:ptyqd
lrwxrwxrwx 1 root root 29 janv 3 2010 ptyqe -> ./devices/pseudo/ptc@0:ptyqe
lrwxrwxrwx 1 root root 29 janv 3 2010 ptyqf -> ./devices/pseudo/ptc@0:ptyqf
lrwxrwxrwx 1 root root 29 janv 3 2010 ptyr0 -> ./devices/pseudo/ptc@0:ptyr0
lrwxrwxrwx 1 root root 29 janv 3 2010 ptyr1 -> ./devices/pseudo/ptc@0:ptyr1
lrwxrwxrwx 1 root root 29 janv 3 2010 ptyr2 -> ./devices/pseudo/ptc@0:ptyr2
lrwxrwxrwx 1 root root 29 janv 3 2010 ptyr3 -> ./devices/pseudo/ptc@0:ptyr3
lrwxrwxrwx 1 root root 29 janv 3 2010 ptyr4 -> ./devices/pseudo/ptc@0:ptyr4
lrwxrwxrwx 1 root root 29 janv 3 2010 ptyr5 -> ./devices/pseudo/ptc@0:ptyr5
lrwxrwxrwx 1 root root 29 janv 3 2010 ptyr6 -> ./devices/pseudo/ptc@0:ptyr6
lrwxrwxrwx 1 root root 29 janv 3 2010 ptyr7 -> ./devices/pseudo/ptc@0:ptyr7
lrwxrwxrwx 1 root root 29 janv 3 2010 ptyr8 -> ./devices/pseudo/ptc@0:ptyr8
lrwxrwxrwx 1 root root 29 janv 3 2010 ptyr9 -> ./devices/pseudo/ptc@0:ptyr9
lrwxrwxrwx 1 root root 29 janv 3 2010 ptyra -> ./devices/pseudo/ptc@0:ptyra
lrwxrwxrwx 1 root root 29 janv 3 2010 ptyrb -> ./devices/pseudo/ptc@0:ptyrb
lrwxrwxrwx 1 root root 29 janv 3 2010 ptyrc -> ./devices/pseudo/ptc@0:ptyrc
lrwxrwxrwx 1 root root 29 janv 3 2010 ptyrd -> ./devices/pseudo/ptc@0:ptyrd
lrwxrwxrwx 1 root root 29 janv 3 2010 ptyre -> ./devices/pseudo/ptc@0:ptyre
lrwxrwxrwx 1 root root 29 janv 3 2010 ptyrf -> ./devices/pseudo/ptc@0:ptyrf
lrwxrwxrwx 1 root root 34 janv 3 2010 ramdiskctl -> ./devices/pseudo/ramdisk@1024:ctl
lrwxrwxrwx 1 root root 33 janv 3 2010 random -> ./devices/pseudo/random@0:random
lrwxrwxrwx 1 root other 29 janv 3 2010 rawip -> ./devices/pseudo/icmp@0:icmp
lrwxrwxrwx 1 root other 31 janv 3 2010 rawip6 -> ./devices/pseudo/icmp6@0:icmp6
lrwxrwxrwx 1 root root 37 janv 3 2010 rdiskette -> ./devices/isa/fdc@1,3f0/fd@0,0:c,raw
lrwxrwxrwx 1 root root 37 janv 3 2010 rdiskette0 -> ./devices/isa/fdc@1,3f0/fd@0,0:c,raw

```

```

lrwxrwxrwx 1 root root          27 janv 3 2010 rds -> ../devices/pseudo/rds@0:rds
drwxr-xr-x 2 root sys           1024 janv 3 2010 rdsk
lrwxrwxrwx 1 root root          5 janv 3 2010 rfd0 -> rfd0c
lrwxrwxrwx 1 root root          37 janv 3 2010 rfd0a -> ../devices/isa/fdc@1,3f0/fd@0,0:a,raw
lrwxrwxrwx 1 root root          37 janv 3 2010 rfd0b -> ../devices/isa/fdc@1,3f0/fd@0,0:b,raw
lrwxrwxrwx 1 root root          10 janv 3 2010 rfd0c -> rdiskette0
drwxr-xr-x 2 root sys           512 janv 3 2010 rmt
lrwxrwxrwx 1 root root          27 janv 3 2010 rsm -> ../devices/pseudo/rsm@0:rsm
lrwxrwxrwx 1 root root          13 janv 3 2010 rsr0 -> rdsk/c1t0d0s2
lrwxrwxrwx 1 root other         27 janv 3 2010 rts -> ../devices/pseudo/rts@0:rts
drwxr-xr-x 2 root sys           512 janv 3 2010 sad
lrwxrwxrwx 1 root root          29 janv 3 2010 sctp -> ../devices/pseudo/sctp@0:sctp
lrwxrwxrwx 1 root root          31 janv 3 2010 sctp6 -> ../devices/pseudo/sctp6@0:sctp6
lrwxrwxrwx 1 root root          27 janv 3 2010 sdp -> ../devices/pseudo/sdp@0:sdp
lrwxrwxrwx 1 root root          33 janv 3 2010 smbios -> ../devices/pseudo/smbios@0:smbios
lrwxrwxrwx 1 root root          35 janv 3 2010 spdsock -> ../devices/pseudo/spdsock@0:spdsock
lrwxrwxrwx 1 root root          30 janv 3 2010 sppp -> ../devices/pseudo/clone@0:sppp
lrwxrwxrwx 1 root root          33 janv 3 2010 sppptun -> ../devices/pseudo/clone@0:sppptun
lrwxrwxrwx 1 root root          12 janv 3 2010 sr0 -> dsk/c1t0d0s2
lrwxrwxrwx 1 root root          6 janv 3 2010 stderr -> ./fd/2
lrwxrwxrwx 1 root root          6 janv 3 2010 stdin -> ./fd/0
lrwxrwxrwx 1 root root          6 janv 3 2010 stdout -> ./fd/1
drwxr-xr-x 2 root sys           512 janv 3 2010 swap
lrwxrwxrwx 1 root other         29 janv 3 2010 syscon -> ../devices/pseudo/cn@0:syscon
lrwxrwxrwx 1 root other         37 janv 3 2010 sysevent -> ../devices/pseudo/sysevent@0:sysevent
lrwxrwxrwx 1 root other         33 janv 3 2010 sysmsg -> ../devices/pseudo/sysmsg@0:sysmsg
lrwxrwxrwx 1 root other         29 janv 3 2010 systty -> ../devices/pseudo/cn@0:systty
lrwxrwxrwx 1 root other         27 janv 3 2010 tcp -> ../devices/pseudo/tcp@0:tcp
lrwxrwxrwx 1 root other         29 janv 3 2010 tcp6 -> ../devices/pseudo/tcp6@0:tcp6
drwxr-xr-x 2 root root          512 janv 3 2010 term
lrwxrwxrwx 1 root root          15 janv 3 2010 text-0 -> /dev/fbs/text-0
lrwxrwxrwx 1 root other         29 janv 3 2010 ticlts -> ../devices/pseudo/tl@0:ticlts
lrwxrwxrwx 1 root other         29 janv 3 2010 ticots -> ../devices/pseudo/tl@0:ticots
lrwxrwxrwx 1 root other         32 janv 3 2010 ticotsord -> ../devices/pseudo/tl@0:ticotsord

```

```

lrwxrwxrwx 1 root root 30 janv 3 2010 tnfctl -> ./devices/pseudo/tnf@0:tnfctl
lrwxrwxrwx 1 root root 30 janv 3 2010 tnfmap -> ./devices/pseudo/tnf@0:tnfmap
lrwxrwxrwx 1 root other 26 janv 3 2010 tty -> ./devices/pseudo/sy@0:tty
lrwxrwxrwx 1 root root 30 janv 3 2010 ttyp0 -> ./devices/pseudo/ptsl@0:ttyp0
lrwxrwxrwx 1 root root 30 janv 3 2010 ttyp1 -> ./devices/pseudo/ptsl@0:ttyp1
lrwxrwxrwx 1 root root 30 janv 3 2010 ttyp2 -> ./devices/pseudo/ptsl@0:ttyp2
lrwxrwxrwx 1 root root 30 janv 3 2010 ttyp3 -> ./devices/pseudo/ptsl@0:ttyp3
lrwxrwxrwx 1 root root 30 janv 3 2010 ttyp4 -> ./devices/pseudo/ptsl@0:ttyp4
lrwxrwxrwx 1 root root 30 janv 3 2010 ttyp5 -> ./devices/pseudo/ptsl@0:ttyp5
lrwxrwxrwx 1 root root 30 janv 3 2010 ttyp6 -> ./devices/pseudo/ptsl@0:ttyp6
lrwxrwxrwx 1 root root 30 janv 3 2010 ttyp7 -> ./devices/pseudo/ptsl@0:ttyp7
lrwxrwxrwx 1 root root 30 janv 3 2010 ttyp8 -> ./devices/pseudo/ptsl@0:ttyp8
lrwxrwxrwx 1 root root 30 janv 3 2010 ttyp9 -> ./devices/pseudo/ptsl@0:ttyp9
lrwxrwxrwx 1 root root 30 janv 3 2010 ttyna -> ./devices/pseudo/ptsl@0:ttyna
lrwxrwxrwx 1 root root 30 janv 3 2010 ttynb -> ./devices/pseudo/ptsl@0:ttynb
lrwxrwxrwx 1 root root 30 janv 3 2010 ttync -> ./devices/pseudo/ptsl@0:ttync
lrwxrwxrwx 1 root root 30 janv 3 2010 ttYPD -> ./devices/pseudo/ptsl@0:ttYPD
lrwxrwxrwx 1 root root 30 janv 3 2010 ttYPE -> ./devices/pseudo/ptsl@0:ttYPE
lrwxrwxrwx 1 root root 30 janv 3 2010 ttYpF -> ./devices/pseudo/ptsl@0:ttYpF
lrwxrwxrwx 1 root root 30 janv 3 2010 ttYq0 -> ./devices/pseudo/ptsl@0:ttYq0
lrwxrwxrwx 1 root root 30 janv 3 2010 ttYq1 -> ./devices/pseudo/ptsl@0:ttYq1
lrwxrwxrwx 1 root root 30 janv 3 2010 ttYq2 -> ./devices/pseudo/ptsl@0:ttYq2
lrwxrwxrwx 1 root root 30 janv 3 2010 ttYq3 -> ./devices/pseudo/ptsl@0:ttYq3
lrwxrwxrwx 1 root root 30 janv 3 2010 ttYq4 -> ./devices/pseudo/ptsl@0:ttYq4
lrwxrwxrwx 1 root root 30 janv 3 2010 ttYq5 -> ./devices/pseudo/ptsl@0:ttYq5
lrwxrwxrwx 1 root root 30 janv 3 2010 ttYq6 -> ./devices/pseudo/ptsl@0:ttYq6
lrwxrwxrwx 1 root root 30 janv 3 2010 ttYq7 -> ./devices/pseudo/ptsl@0:ttYq7
lrwxrwxrwx 1 root root 30 janv 3 2010 ttYq8 -> ./devices/pseudo/ptsl@0:ttYq8
lrwxrwxrwx 1 root root 30 janv 3 2010 ttYq9 -> ./devices/pseudo/ptsl@0:ttYq9
lrwxrwxrwx 1 root root 30 janv 3 2010 ttYqa -> ./devices/pseudo/ptsl@0:ttYqa
lrwxrwxrwx 1 root root 30 janv 3 2010 ttYqb -> ./devices/pseudo/ptsl@0:ttYqb
lrwxrwxrwx 1 root root 30 janv 3 2010 ttYqc -> ./devices/pseudo/ptsl@0:ttYqc
lrwxrwxrwx 1 root root 30 janv 3 2010 ttYqd -> ./devices/pseudo/ptsl@0:ttYqd
lrwxrwxrwx 1 root root 30 janv 3 2010 ttYqe -> ./devices/pseudo/ptsl@0:ttYqe

```

lrwxrwxrwx	1	root	root	30	janv	3	2010	ttyqf	-> ./devices/pseudo/ptsl@0:ttyqf
lrwxrwxrwx	1	root	root	30	janv	3	2010	ttyr0	-> ./devices/pseudo/ptsl@0:ttyr0
lrwxrwxrwx	1	root	root	30	janv	3	2010	ttyr1	-> ./devices/pseudo/ptsl@0:ttyr1
lrwxrwxrwx	1	root	root	30	janv	3	2010	ttyr2	-> ./devices/pseudo/ptsl@0:ttyr2
lrwxrwxrwx	1	root	root	30	janv	3	2010	ttyr3	-> ./devices/pseudo/ptsl@0:ttyr3
lrwxrwxrwx	1	root	root	30	janv	3	2010	ttyr4	-> ./devices/pseudo/ptsl@0:ttyr4
lrwxrwxrwx	1	root	root	30	janv	3	2010	ttyr5	-> ./devices/pseudo/ptsl@0:ttyr5
lrwxrwxrwx	1	root	root	30	janv	3	2010	ttyr6	-> ./devices/pseudo/ptsl@0:ttyr6
lrwxrwxrwx	1	root	root	30	janv	3	2010	ttyr7	-> ./devices/pseudo/ptsl@0:ttyr7
lrwxrwxrwx	1	root	root	30	janv	3	2010	ttyr8	-> ./devices/pseudo/ptsl@0:ttyr8
lrwxrwxrwx	1	root	root	30	janv	3	2010	ttyr9	-> ./devices/pseudo/ptsl@0:ttyr9
lrwxrwxrwx	1	root	root	30	janv	3	2010	ttyra	-> ./devices/pseudo/ptsl@0:ttyra
lrwxrwxrwx	1	root	root	30	janv	3	2010	ttyrb	-> ./devices/pseudo/ptsl@0:ttyrb
lrwxrwxrwx	1	root	root	30	janv	3	2010	ttyrc	-> ./devices/pseudo/ptsl@0:ttyrc
lrwxrwxrwx	1	root	root	30	janv	3	2010	ttyrd	-> ./devices/pseudo/ptsl@0:ttyrd
lrwxrwxrwx	1	root	root	30	janv	3	2010	ttyre	-> ./devices/pseudo/ptsl@0:ttyre
lrwxrwxrwx	1	root	root	30	janv	3	2010	ttyrf	-> ./devices/pseudo/ptsl@0:ttyrf
lrwxrwxrwx	1	root	root	31	janv	3	2010	ucode	-> ./devices/pseudo/ucode@0:ucode
lrwxrwxrwx	1	root	other	27	janv	3	2010	udp	-> ./devices/pseudo/udp@0:udp
lrwxrwxrwx	1	root	other	29	janv	3	2010	udp6	-> ./devices/pseudo/udp6@0:udp6
lrwxrwxrwx	1	root	root	34	janv	3	2010	urandom	-> ./devices/pseudo/random@0:urandom
drwxr-xr-x	2	root	root	512	janv	3	2010	usb	
lrwxrwxrwx	1	root	other	43	mars	13	2011	vboxguest	-> ./devices/pci@0,0/pci80ee,cafe@4:vboxguest
lrwxrwxrwx	1	root	root	29	janv	3	2010	vni	-> ./devices/pseudo/clone@0:vni
lrwxrwxrwx	1	root	root	30	janv	3	2010	volctl	-> ./devices/pseudo/vol@0:volctl
lrwxrwxrwx	1	root	root	2	janv	3	2010	vt00	-> fb
lrwxrwxrwx	1	root	root	35	janv	3	2010	winlock	-> ./devices/pseudo/winlock@0:winlock
lrwxrwxrwx	1	root	other	29	janv	3	2010	wscons	-> ./devices/pseudo/wc@0:wscons
lrwxrwxrwx	1	root	root	20	janv	3	2010	xsvc	-> ./devices/xsvc:xsvc
lrwxrwxrwx	1	root	other	27	janv	3	2010	zero	-> ./devices/pseudo/mm@0:zerolrwxrwxrwx
other	27	janv	3	2010	zfs				1 root

Chaque objet dans le répertoire **/dev** est un lien symbolique vers un fichier dans l'arborescence du répertoire **/devices**. Par exemple, la deuxième tranche du premier disque :

```
# ls -l /dev/dsk/c0d0s1
lrwxrwxrwx 1 root      root          50 janv  3  2010 /dev/dsk/c0d0s1 -> ../../devices/pci@0,0/pci-
ide@1,1/ide@0/cmdk@0,0:b
# ls -l /devices/pci@0,0/pci-ide@1,1/ide@0/cmdk@0,0:b*
brw-r----- 1 root      sys        102,  1 août 29 09:55 /devices/pci@0,0/pci-ide@1,1/ide@0/cmdk@0,0:b
crw-r----- 1 root      sys        102,  1 août 30 15:58 /devices/pci@0,0/pci-ide@1,1/ide@0/cmdk@0,0:b, raw
```

On peut noter dans la sortie de la commande que certains fichiers sont de type **bloc (b)**, tandis que d'autre sont de type **caractère (c)**.

```
...
brw-r----- 1 root      sys        102,  1 août 29 09:55 /devices/pci@0,0/pci-ide@1,1/ide@0/cmdk@0,0:b
crw-r----- 1 root      sys        102,  1 août 30 15:58 /devices/pci@0,0/pci-ide@1,1/ide@0/cmdk@0,0:b, raw
...
```

La différence entre les deux repose sur le type de communication entre le système et le module. Dans le premier cas le système accède au périphérique par des coordonnées du bloc de données sur le support tandis que dans le deuxième cas la communication d'échange de données se fait octet par octet sans utiliser des tampons.

Les deux informations clefs du fichier spécial sont situées à la place de la taille d'un fichier normal et se nomment le **majeur** et le **mineur** :

- le **majeur** identifie le pilote du périphérique et donc son contrôleur,
- le **mineur** identifie le périphérique ou une particularité du périphérique.

La correspondance entre les entrées du répertoire **/dev** et **/devices** est gérée par la commande **devfsadm** qui est lancée au démarrage du système. Cette même commande lance le démon **devfsadmd**. Ce démon sert aux re-configurations dynamiques des périphériques. Les commandes **disks**, **tapes**, **ports** et **devlinks** créées les entrées dans **/dev** :

Commande	Description
drvconfig	Configure le répertoire <b>/devices</b>
disks	Crée les entrées <b>/dev/</b> à partir de <b>/devices</b> pour les disques
tapes	Crée les entrées <b>/dev/</b> à partir de <b>/devices</b> pour les bandes
ports	Crée les entrées <b>/dev/</b> à partir de <b>/devices</b> pour les ports série
devlinks	Crée les entrées <b>/dev/</b> à partir de <b>/devices</b> pour les autres pilotes

## Commandes

### **prtconf**

Pour consulter la liste des périphériques , il convient d'utiliser la commande **prtconf** :

```
# prtconf -D
System Configuration: Sun Microsystems i86pc
Memory size: 1400 Megabytes
System Peripherals (Software Nodes):

i86pc (driver name: rootnex)
    scsi_vhci, instance #0 (driver name: scsi_vhci)
    isa, instance #0 (driver name: isa)
        i8042, instance #0 (driver name: i8042)
            keyboard, instance #0 (driver name: kb8042)
            mouse, instance #0 (driver name: mouse8042)
        fdc, instance #0 (driver name: fdc)
            fd, instance #0 (driver name: fd)
        lp, instance #0 (driver name: ecpp)
    pci, instance #0 (driver name: pci)
        pci8086,1237
        pci8086,7000
        pci-ide, instance #0 (driver name: pci-ide)
            ide, instance #0 (driver name: ata)
                cmdk, instance #0 (driver name: cmdk)
            ide, instance #1 (driver name: ata)
                sd, instance #0 (driver name: sd)
        display, instance #0 (driver name: vgatext)
        pci8086,1e, instance #0 (driver name: e1000g)
        pci80ee,cafe
        pci8086,7113
    iscsi, instance #0 (driver name: iscsi)
```

```
pseudo, instance #0 (driver name: pseudo)
options, instance #0 (driver name: options)
agpgart, instance #0 (driver name: agpgart)
xsvc, instance #0 (driver name: xsvc)
objmgr, instance #0 (driver name: objmgr)
acpi
used-resources
cpus, instance #0 (driver name: cpunex)
    cpu (driver name: cpudrv)
```

L'option **-D** précise que l'on souhaite les noms des drivers. L'option **-v** peut également être utilisée

## **prtdiag**

La commande `prtdiag` offre la même information sous une forme plus lisible :

```
# prtdiag
System Configuration: innotek GmbH VirtualBox
BIOS Configuration: innotek GmbH VirtualBox 12/01/2006
```

## ==== Processor Sockets =====

Version	Location Tag
-----	-----

## ==== Memory Device Sockets ====

Type Status Set Device Locator Bank Locator

## ==== On-Board Devices =====

#### ----- Upgradable Slots -----

ID Status Type Description

Veuillez noter que la commande ci-dessus ne peut produire de l'information que dans le cas où la carte mère supporte l'image System Management BIOS. Dans le cas contraire, utilisez la commande **smbios**.

**smbios**

```
# smbios
ID      SIZE TYPE
0       54   SMB_TYPE_BIOS (BIOS information)

Vendor: innotek GmbH
Version String: VirtualBox
Release Date: 12/01/2006
Address Segment: 0xe000
ROM Size: 131072 bytes
Image Size: 131072 bytes
Characteristics: 0x48018090
    SMB_BIOSFL_ISA (ISA is supported)
    SMB_BIOSFL_PCI (PCI is supported)
    SMB_BIOSFL_CDBOOT (Boot from CD is supported)
    SMB_BIOSFL_SELBOOT (Selectable Boot supported)
    SMB_BIOSFL_I9_KBD (int 0x9 8042 keyboard svcs)
    SMB_BIOSFL_I10_CGA (int 0x10 CGA svcs)
Characteristics Extension Byte 1: 0x1
    SMB_BIOSXB1_ACPI (ACPI is supported)
Characteristics Extension Byte 2: 0x0

ID      SIZE TYPE
1       72   SMB_TYPE_SYSTEM (system information)

Manufacturer: innotek GmbH
Product: VirtualBox
Version: 1.2
Serial Number: 0

UUID: 52688f3d-628e-4905-8ecf-0c7306bae00f
Wake-Up Event: 0x6 (power switch)
```

SKU Number:

Family: Virtual Machine

ID SIZE TYPE

8 50 SMB\_TYPE\_BASEBOARD (base board)

Manufacturer: Oracle Corporation

Product: VirtualBox

Version: 1.2

Serial Number: 0

Chassis: 3

Flags: 0x1

SMB\_BBFL\_MOTHERBOARD (board is a motherboard)

Board Type: 0xa (motherboard)

ID SIZE TYPE

3 31 SMB\_TYPE\_CHASSIS (system enclosure or chassis)

Manufacturer: Oracle Corporation

OEM Data: 0x0

Lock Present: N

Chassis Type: 0x1 (other)

Boot-Up State: 0x3 (safe)

Power Supply State: 0x3 (safe)

Thermal State: 0x3 (safe)

Chassis Height: 0u

Power Cords: 0

Element Records: 0

ID SIZE TYPE

2 34 SMB\_TYPE\_OEMSTR (OEM string table)

```
vboxVer_4.2.6  
vboxRev_82870
```

## psrinfo

La commande **psrinfo** est utilisée pour fournir des informations détaillées sur les processeurs du système :

```
# psrinfo -pv  
Le processeur physique a 2 virtuel processeurs (0 1)  
x86 (chipid 0x0 GenuineIntel family 6 model 58 step 9 clock 2300 MHz)  
    Intel(r) Core(tm) i7-3610QM CPU @ 2.30GHz
```

## isainfo

Cette commande est utilisée pour fournir les informations sur l'Instruction Set Architecture :

```
# isainfo -v  
64-bit amd64 applications  
    ssse3 tscp sse3 sse2 sse fxsr mmx cmov amd_sysc cx8 tsc fpu  
32-bit i386 applications  
    ssse3 tscp sse3 sse2 sse fxsr mmx cmov sep cx8 tsc fpu
```

## modinfo

Cette commande fournit une liste des modules chargés :

```
# modinfo | more  
Id      Loadaddr  Size Info Rev Module Name  
0  ffffffb800000 18d2f0  -   0  unix ()  
1  ffffffb904b70 3eeda  -   0  krtld ()
```

2	fffffffffb939860	27c700	-	0	genunix ()
4	fffffffffb5f000	5618	1	1	specfs (filesystem for specfs)
5	fffffffffb64520	4488	3	1	fifofs (filesystem for fifo)
6	fffffffff80e5000	18c50	155	1	dtrace (Dynamic Tracing)
7	fffffffffb688b8	5158	16	1	devfs (devices filesystem 1.16)
8	fffffffffb6d748	1f350	5	1	procfs (filesystem for proc)
10	fffffffffb8bfa8	3b40	1	1	TS (time sharing sched class)
11	fffffffffb8f2a8	9e0	-	1	TS_DPTBL (Time sharing dispatch table)
13	fffffffffb8f420	7c08	-	1	pci_autoconfig (PCI BIOS interface 1.44)
14	fffffffffb96e68	591f8	-	1	acpica (ACPI interpreter 1.5)
15	ffffffffff7800000	18310	-	1	pcie (PCI Express Framework Module)
16	fffffffffbbeea28	2610	-	1	busra (Bus Resource Allocator (BUSRA) )
17	fffffffffbff0fb0	30a0	1	1	uppc (UniProcessor PC)
19	ffffffff782b000	10268	2	1	pcplusmp (pcplusmp v1.4 compatible 1.136)
20	fffffffffbff3de8	4450	-	1	cpu.generic (Generic x86 CPU Module)
21	fffffffffbff7a10	1458	-	1	cpu_ms.GenuineIntel (Generic Intel model-sp)
22	ffffffff783a000	144b0	1	1	rootnex (i86pc root nexus 1.137)
23	fffffffffbff8db8	2798	-	1	iommulib (IOMMU library module)
24	ffffffff784dce0	298	57	1	options (options driver)
25	ffffffff78392f0	d18	2	1	pseudo (nexus driver for 'pseudo' 1.28)
26	ffffffff782a8b0	748	11	1	clone (Clone Pseudodriver 'clone')
27	ffffffff784e000	17c30	174	1	scsi_vhci (SCSI VHCI Driver 1.72)
28	ffffffff7865000	23238	-	1	scsi (SCSI Bus Utility Routines)
29	fffffffffbfb498	2730	120	1	isa (isa nexus driver for 'ISA' 1.40)
31	fffffffffbfd680	17e0	12	1	sad (STREAMS Administrative Driver ')
33	ffffffff7885000	3c130	2	1	ufs (filesystem for ufs)
34	ffffffff7864cf8	3b8	-	1	fssnap_if (File System Snapshot Interface)
35	ffffffff7819000	5bd0	246	1	ppm (platform pm driver v1.18)
36	ffffffff781e718	6198	86	1	pci (host to PCI nexus driver 1.71)

--A suivre--

## sysdef

La commande **sysdef** permet aussi de visualiser les périphériques mais cette fois-ci avec pseudo-devices, la mémoire virtuelle et les paramètres noyau :

```
# sysdef | more
*
* Hostid
*
1e022911
*
* i86pc Configuration
*
*
* Devices
*
scsi_vhci, instance #0
isa, instance #0
    i8042, instance #0
        keyboard, instance #0
        mouse, instance #0
    fdc, instance #0
        fd, instance #0
    lp, instance #0
pci, instance #0
    pci8086,1237 (driver not attached)
    pci8086,7000 (driver not attached)
    pci-ide, instance #0
--A suivre--
```

## dmesg

La commande **dmesg** permet de visualiser les messages du démarrage du système. Il est donc possible de retrouver des informations concernant le démarrage et la prise en compte des périphériques :

```
# dmesg | grep -i e1000g
Mar 17 09:23:23 unknown mac: [ID 469746 kern.info] NOTICE: e1000g0 registered
Mar 17 09:23:23 unknown e1000g: [ID 766679 kern.info] Intel(R) PRO/1000 Network Connection, Driver Ver. 5.1.11
Mar 17 09:23:28 unknown e1000g: [ID 801725 kern.info] NOTICE: pci8086,100e - e1000g[0] : Adapter 1000Mbps full
duplex copper link is up.
Mar 17 09:23:33 unknown /sbin/dhcpcagent[52]: [ID 778557 daemon.warning] configure_v4_lease: no IP broadcast
specified for e1000g0, making best guess
```

## uname

La commande **uname** peut afficher plusieurs informations importantes :

```
# uname -p
i386
# uname -m
i86pc
# uname -r
5.10
# uname -a
SunOS unknown 5.10 Generic_127128-11 i86pc i386 i86pc
```

## showrev

La commande **showrev** donne les caractéristiques de la machine :

```
# showrev
Hostname: unknown
Hostid: 1e022911
Release: 5.10
Kernel architecture: i86pc
Application architecture: i386
```

Hardware provider:

Domain:

Kernel version: SunOS 5.10 Generic\_127128-11

## devattr

La commande **devattr** permet de consulter les attributs d'un périphériques :

```
# devattr -v /dev/dsk/c0d0s0
alias='dpart100'
bdevice='/dev/dsk/c0d0s0'
capacity='11406150'
cdevice='/dev/rdsck/c0d0s0'
desc='Disk Partition'
dparttype='fs'
fstype='ufs'
mountpt='/'
removable='false'
type='dpart'
```

## vold

Le service vold permet le montage et démontage automatique des périphériques amovibles :

```
# ps -ef | grep vold
root 543 1 0 09:24:39 ? 0:00 /usr/sbin/vold -f /etc/vold.conf
root 1894 1578 0 22:02:48 pts/4 0:00 grep vold
```

Vold se configure par l'édition du fichier **/etc/vold.conf** et utilise le répertoire **/vol** lors de son travail :

```
# Copyright 2006 Sun Microsystems, Inc. All rights reserved.
```

```
# Use is subject to license terms.  
#  
# ident "@(#)vold.conf 1.27      06/01/20 SMI"  
#  
# Volume Daemon Configuration file  
  
# Database to use (must be first)  
db db_mem.so  
  
# Volfs support of removable media devices  
support nomedia  
  
# Labels supported  
label cdrom label_cdrom.so cdrom  
label dos label_dos.so floppy rmdisk pcmem  
label sun label_sun.so floppy rmdisk pcmem  
  
# Devices to use  
use cdrom drive /dev/rdsck/c*s2 dev_cdrom.so cdrom%d  
use floppy drive /dev/rdiskette[0-9] dev_floppy.so floppy%d  
use pcmem drive /dev/rdsck/c*s2 dev_pcmem.so pcmem%d forceload=true  
use rmdisk drive /dev/rdsck/c*s2 dev_rmdisk.so rmdisk%d  
  
# Actions  
eject dev/diskette[0-9]/* user=root /usr/sbin/rmmount  
eject dev/dsk/* user=root /usr/sbin/rmmount  
insert dev/diskette[0-9]/* user=root /usr/sbin/rmmount  
insert dev/dsk/* user=root /usr/sbin/rmmount  
notify rdsck/* group=tty user=root /usr/lib/vold/volmissing -p  
remount dev/diskette[0-9]/* user=root /usr/sbin/rmmount  
remount dev/dsk/* user=root /usr/sbin/rmmount  
  
# List of file system types unsafe to eject
```

```
unsafe ufs hsfs pcfs udfs
```

Le script **/etc/init.d/volmgt** est utilisé pour démarrer et arrêter le service vold :

```
#!/sbin/sh
#
# Copyright 2006 Sun Microsystems, Inc. All rights reserved.
# Use is subject to license terms.
#
# ident "@(#)volmgt      1.9      06/01/20 SMI"

case "$1" in
'start')
    if [ -f /etc/vold.conf -a -f /usr/sbin/vold -a \
        "${_INIT_ZONENAME:=`sbin/zonename`} = "global" ]; then
        echo 'volume management starting.'
        svcadm enable svc:/system/filesystem/volfs:default
    fi
    ;;

'stop')
    svcadm disable svc:/system/filesystem/volfs:default
    ;;

*)
    echo "Usage: $0 { start | stop }"
    exit 1
    ;;

esac
exit 0
```

Insérez le CD des **Additions invité** de VirtualBox en utilisant les menus Périphériques > Installer les Additions invité ... de la fenêtre de votre machine virtuelle. Saisissez maintenant la commande suivante :

```
# mount | grep cdrom
/cdrom/vboxadditions_4.1.14_77440 on /vol/dev/dsk/c1t0d0/vboxadditions_4.1.14_77440 read
only/nosetuid/nodevices/noglobal/maplcase/rr/traildot/dev=1740001 on jeu. août 30 17:03:50 2012
```

Vous pouvez avoir accès au CDROM via /cdrom :

```
# cd /cdrom
# ls
cdrom0           vboxadditions_4.1.14_77440
vboxadditions_4.0.4_70112
```

en utilisant le lien symbolique cdrom0 :

```
# cd cdrom0
# ls
32Bit           VBoxLinuxAdditions.run
64Bit           VBoxSolarisAdditions.pkg
AUTORUN.INF     VBoxWindowsAdditions-amd64.exe
autorun.sh      VBoxWindowsAdditions-x86.exe
runasroot.sh    VBoxWindowsAdditions.exe
```

Pour éjecter le CDROM, il convient d'utiliser la commande **eject** :

```
# eject
/vol/dev/rdsk/c1t0d0/vboxadditions_4.1.14_77440: Périphérique en cours d'utilisation
# cd ..
# eject
```

La commande **cdrw** permet d'identifier les graveurs CD et DVD sur le système :

# cdrw -l		
Recherche de périphériques de CD...		
Noeud	Périphérique connecté	Type de périphérique

cdrom0	VBOX	CD-ROM	1.0	Lecteur de CD
--------	------	--------	-----	---------------

## cfgadm

La commande **cfgadm** permet de lister, ajouter et retirer des périphériques de type hotplug.

# cfgadm	Ap_Id	Type	Receptacle	Occupant	Condition
	pcic0.pccard0	unknown	empty	unconfigured	unknown
	pcie2	unknown	empty	unconfigured	unknown
	usb0/1	unknown	empty	unconfigured	ok
	usb0/2	usb-mouse	connected	configured	ok
	usb1/1	unknown	empty	unconfigured	ok
	usb1/2	unknown	empty	unconfigured	ok
	usb1/3	unknown	empty	unconfigured	ok
	usb1/4	unknown	empty	unconfigured	ok
	usb1/5	unknown	empty	unconfigured	ok
	usb1/6	unknown	empty	unconfigured	ok
	usb1/7	unknown	empty	unconfigured	ok
	usb1/8	unknown	empty	unconfigured	ok

## stty -a

Les terminaux asynchrones sont gérés par les processus ttymon. Le premier port série sous Solaris est le /dev/term/a. La commande stty -a permet de visualiser l'état de la liaison :

```
# tty
/dev/pts/3
# stty -a
speed 38400 baud;
```

```
rangées = 24 ; colonnes = 94 ; pixels_y = 0 ; pixels_x = 0 ;
csdata ?
eucw 1:0:0:0, scrw 1:0:0:0
intr = ^c; quit = ^\; erase = ^?; kill = ^u;
eof = ^d; eol = \ÿ; eol2 = \ÿ; swtch = <undef>;
start = ^q; stop = ^s; susp = ^z; dsusp = ^y;
rprnt = ^r; flush = ^o; werase = ^w; lnext = ^v;
-parenb -parodd cs8 -cstopb hupcl cread -clocl -loblk -crtscs -crtsxoff -parext
-ignbrk brkint -ignpar -parmrk -inpck -istrip -inlcr -igncr icrnl -iuclc
ixon ixany -ixoff imaxbel
isig icanon -xcase echo echoe echok -echonl -noflsh
-tostop echoctl -echoprt echoke -defecho -flusho -pendin iexten
opost -olcuc onlcr -ocrnl -onocr -onlret -ofill -ofdel
```

Les émulations de terminaux sont gérées par des fichiers de description dans le répertoire /usr/share/lib/terminfo :

```
# file /usr/share/lib/terminfo/v/vt100
/usr/share/lib/terminfo/v/vt100:           Entrée Terminfo compilée
```

## Gestion des performances

### Commandes

#### iostat

La commande **iostat** affiche des statistiques sur l'utilisation des disques, des terminaux et des lecteurs de cartouche :

```
# iostat
  tty      cmdk0        sd0        nfs1      cpu
tin tout kps tps serv  kps tps serv  kps tps serv   us sy wt id
```

0	10	194	22	4	0	0	0	0	0	0	11	36	0	53
---	----	-----	----	---	---	---	---	---	---	---	----	----	---	----

ou

# iostat -xtc 5 2

device	extended device statistics								tty		cpu				
	r/s	w/s	kr/s	kw/s	wait	actv	svc_t	%w	%b	tin	tout	us	sy	wt	id
md5	0.0	0.0	0.0	0.0	0.0	0.0	152.7	0	0	0	10	1	2	0	96
sd0	9.5	6.4	65.6	27.9	0.5	0.1	35.5	0	2						
sd1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0	0						
sd2	0.0	0.6	0.3	12.5	0.0	0.0	34.4	0	0						
sd3	0.0	0.6	0.2	12.5	0.0	0.0	33.9	0	0						
sd4	0.0	0.6	0.3	12.5	0.0	0.0	38.3	0	0						
sd5	0.0	0.6	0.3	12.4	0.0	0.0	36.1	0	0						
sd6	0.0	0.0	0.0	0.0	0.0	0.0	3.0	0	0						
sd7	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0	0						
nfs1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0	0						
device	extended device statistics								tty		cpu				
	r/s	w/s	kr/s	kw/s	wait	actv	svc_t	%w	%b	tin	tout	us	sy	wt	id
md5	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0	0	0	161	1	2	0	97
sd0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0	0						
sd1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0	0						
sd2	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0	0						
sd3	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0	0						
sd4	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0	0						
sd5	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0	0						
sd6	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0	0						
sd7	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0	0						
nfs1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0	0						

The fields have the following meanings:

disk name of the disk

r/s reads per second

```
w/s      writes per second  
Kr/s     kilobytes read per second  
Kw/s     kilobytes written per second  
wait     average number of transactions waiting for service (Q length)  
actv     average number of transactions actively being serviced  
(removed from the queue but not yet completed)  
%w       percent of time there are transactions waiting  
          for service (queue non-empty)  
%b       percent of time the disk is busy (transactions  
          in progress)
```

Les options de cette commande sont :

```
Usage: iostat [-cCdDeEiImMnpPrstxXYz] [-l n] [-T d|u] [disk ...] [interval [count]]  
-c:      report percentage of time system has spent  
          in user/system/wait/idle mode  
-C:      report disk statistics by controller  
-d:      display disk Kb/sec, transfers/sec, avg.  
          service time in milliseconds  
-D:      display disk reads/sec, writes/sec,  
          percentage disk utilization  
-e:      report device error summary statistics  
-E:      report extended device error statistics  
-i:      show device IDs for -E output  
-I:      report the counts in each interval,  
          instead of rates, where applicable  
-l n:    Limit the number of disks to n  
-m:    Display mount points (most useful with -p)  
-M:    Display data throughput in MB/sec instead of Kb/sec  
-n:    convert device names to cXdYtZ format  
-p:    report per-partition disk statistics  
-P:    report per-partition disk statistics only,  
          no per-device disk statistics  
-r:    Display data in comma separated format
```

```
-s: Suppress state change messages
-T d|u Display a timestamp in date (d) or unix time_t (u)
-t: display chars read/written to terminals
-x: display extended disk statistics
-X: display I/O path statistics
-Y: display I/O path (I/T/L) statistics
-z: Suppress entries with all zero values
```

Consultez le manuel pour comprendre les options et expérimenez avec celles-ci.

## vmstat

La commande **vmstat** affiche des statistiques sur la pagination :

```
# vmstat
kthr      memory          page          disk          faults        cpu
r b w    swap   free   re   mf pi po fr de sr cd f0 s0 --  in   sy   cs us sy id
0 0 0 1414468 1067720 87 278 166 1 1 0 6 21 -0 -0 0 418 1909 756 11 36 53
```

Les options de cette commande sont :

```
# vmstat -h
Usage: vmstat [-cipqsS] [disk ...] [interval [count]]
```

Par défaut la commande vmstat affiche des informations depuis le démarrage du système sinon dans les **n** dernières secondes , **m** fois :

```
# vmstat 10 2
kthr      memory          page          disk          faults        cpu
r b w    swap   free   re   mf pi po fr de sr cd f0 s0 --  in   sy   cs us sy id
```

```
0 0 0 1409432 1061712 83 266 158 1 1 0 6 20 -0 -0 0 422 1875 746 12 35 53
0 0 0 1304184 936392 1 3 0 0 0 0 0 3 0 0 0 0 367 372 250 4 34 63
```

Consultez le manuel pour comprendre les options et expérimenez avec celles-ci.

## prstat

La commande **prstat** affiche des statistiques sur les processus :

```
# prstat
  PID USERNAME SIZE  RSS STATE PRI NICE   TIME CPU PROCESS/NLWP
  668 root      31M 26M sleep  59   0 0:00:47 9,3% Xorg/1
 1115 root      80M 16M sleep  59   0 0:00:09 4,3% gnome-terminal/2
 1033 root      94M 25M sleep  59   0 0:00:26 4,1% java/14
 1091 root     136M 69M sleep  49   0 0:01:12 3,6% firefox-bin/7
  802 noaccess 164M 85M sleep  59   0 0:00:34 1,4% java/23
 1023 root      67M 10M sleep  59   0 0:00:07 1,2% mixer_applet2/1
 1143 root     4760K 2840K cpu0  55   0 0:00:00 0,8% prstat/1
  960 root      67M 13M sleep  59   0 0:00:03 0,6% metacity/1
  967 root      73M 17M sleep  59   0 0:00:06 0,6% gnome-panel/1
  936 root     12M 9812K sleep  59   0 0:00:05 0,5% gconfd-2/1
 1013 root     69M 13M sleep  59   0 0:00:03 0,5% gnome-netstatus/1
  359 root    6144K 2816K sleep  59   0 0:00:01 0,2% nscd/29
  714 root    6136K 4336K sleep  59   0 0:00:01 0,2% httpd/1
  969 root     77M 20M sleep  59   0 0:00:03 0,1% nautilus/3
 1003 root     70M 13M sleep  59   0 0:00:02 0,1% wnck-applet/1
  941 root    5692K 3472K sleep  59   0 0:00:01 0,1% xscreensaver/1
  573 root    5824K 3016K sleep  59   0 0:00:01 0,1% httpd/1
  948 root     67M 9200K sleep  59   0 0:00:01 0,1% gnome-settings-/1
  946 root    9084K 3796K sleep  59   0 0:00:01 0,1% gnome-smproxy/1
```

```
974 root      7824K 4416K sleep    59   0  0:00:00 0,1% gnome-vfs-daemon/2
978 root      2232K 1268K sleep    59   0  0:00:00 0,1% mapping-daemon/1
Total: 85 processes, 259 lwps, load averages: 0,93, 0,87, 0,66
```

Consultez le manuel pour comprendre les options et expérimenter avec celles-ci.

## **sar**

La commande **sar** permet de surveiller toutes les ressources du système selon l'option qui est passée en argument à la commande. Quelques options importantes sont :

Option	Description
-u	Pourcentage d'utilisation du CPU
-q	Nombre de processus en attente
-r	Utilisation de la mémoire centrale
-w	Surveillance du swapping
-p	Surveillance de la pagination
-b	Utilisation des tampons
-d	Utilisation des disques

La commande **/usr/lib/sa/sadc** permet de collecter les informations.

Le script **/usr/lib/sa/sa1** exécute la commande **/usr/lib/sa/sadc**. Ce script prend deux options :

Option	Description
-t	L'intervalle entre les collectes
-n	Nombre de collectes

Le script **/usr/lib/sa/sa2** exécute la commande **sar** et consigne les informations dans un fichier au format **/var/adm/sa/sar<jj>**, par exemple :

```
# ls /var/adm/sa  
# ls  
sa21    sar21
```

Pour pouvoir fonctionner correctement, ces scripts doivent être appelés par **cron**.

Pour constater ce fonctionnement manuellement saisissez la commande suivante :

```
# /usr/lib/sa/sa1 5 5
```

Saisissez ensuite les commandes suivantes :

```
# sar  
  
SunOS unknown 5.10 Generic_127128-11 i86pc      03/22/2009  
  
10:50:22      %usr      %sys      %wio      %idle  
10:50:27          4         32          0         64  
10:50:32          4         33          0         63  
10:50:37          5         35          0         61  
10:50:42          4         32          0         64  
  
Average          4         33          0         63  
# sar -u 5 3  
  
SunOS unknown 5.10 Generic_127128-11 i86pc      03/22/2009  
  
10:51:04      %usr      %sys      %wio      %idle  
10:51:09          5         33          0         62  
10:51:14         12         34          0         53  
10:51:20         16         35          0         49  
  
Average          11         34          0         55  
# sar -r 5 3
```

```
SunOS unknown 5.10 Generic_127128-11 i86pc      03/22/2009
```

```
10:51:23 freemem freeswap
```

```
10:51:28 232663 2596960
```

```
10:51:33 232663 2596960
```

```
10:51:39 232663 2596960
```

```
Average 232663 2596960
```

```
# sar -w 5 3
```

```
SunOS unknown 5.10 Generic_127128-11 i86pc      03/22/2009
```

```
10:52:34 swpin/s bswin/s swpot/s bswot/s pswch/s
```

```
10:52:39 0.00 0.0 0.00 0.0 616
```

```
10:52:44 0.00 0.0 0.00 0.0 603
```

```
10:52:49 0.00 0.0 0.00 0.0 353
```

```
Average 0.00 0.0 0.00 0.0 524
```

```
# sar -b 5 3
```

```
SunOS unknown 5.10 Generic_127128-11 i86pc      03/22/2009
```

```
10:52:52 bread/s lread/s %rcache bwrit/s lwrit/s %wcache pread/s pwrit/s
```

```
10:52:58 0 0 100 0 0 100 0 0
```

```
10:53:03 0 0 100 0 0 0 0 0
```

```
10:53:08 0 0 100 0 0 100 0 0
```

```
Average 0 0 100 0 0 0 0 0
```

```
# sar -d 5 3
```

```
SunOS unknown 5.10 Generic_127128-11 i86pc      03/22/2009
```

```
10:53:10 device      %busy    avque   r+w/s   blks/s  avwait  avserv
```

10:53:15	cmdk0	0	0.0	0	0	0.0	0.0
	fd0	0	0.0	0	0	0.0	0.0
	nfs1	0	0.0	0	0	0.0	0.0
	sd0	0	0.0	0	0	0.0	0.0
10:53:20	cmdk0	0	0.0	0	0	0.0	0.0
	fd0	0	0.0	0	0	0.0	0.0
	nfs1	0	0.0	0	0	0.0	0.0
	sd0	0	0.0	0	0	0.0	0.0
10:53:26	cmdk0	0	0.0	0	0	0.0	1.8
	fd0	0	0.0	0	0	0.0	0.0
	nfs1	0	0.0	0	0	0.0	0.0
	sd0	0	0.0	0	0	0.0	0.0
Average	cmdk0	0	0.0	0	0	0.0	1.8
	fd0	0	0.0	0	0	0.0	0.0
	nfs1	0	0.0	0	0	0.0	0.0
	sd0	0	0.0	0	0	0.0	0.0

# sar -c 5 3

SunOS unknown 5.10 Generic\_127128-11 i86pc 03/22/2009

	scall/s	sread/s	swrit/s	fork/s	exec/s	rchar/s	wchar/s
10:53:31							
10:53:36	1269	306	162	0.00	0.00	71497	70800
10:53:41	1509	314	174	0.00	0.00	21162	20231
10:53:46	1190	317	169	0.00	0.00	56065	55719
Average	1322	312	168	0.00	0.00	49597	48940

# sar -g 5 3

SunOS unknown 5.10 Generic\_127128-11 i86pc 03/22/2009

```
10:53:49 pgout/s ppgout/s pgfree/s pgscan/s %ufs_ipf
10:53:54 0.00 0.00 0.00 0.00 0.00
10:53:59 0.00 0.00 0.00 0.00 0.00
10:54:05 0.00 0.00 0.00 0.00 0.00

Average 0.00 0.00 0.00 0.00 0.00
# sar -p 5 3
```

SunOS unknown 5.10 Generic\_127128-11 i86pc 03/22/2009

```
10:54:07 atch/s pgin/s ppgin/s pflt/s vflt/s slock/s
10:54:12 0.00 0.00 0.00 0.99 4.75 0.00
10:54:17 0.00 0.00 0.00 0.00 0.59 0.00
10:54:22 0.00 0.00 0.00 0.00 0.59 0.00

Average 0.00 0.00 0.00 0.33 1.97 0.00
# sar -k 5 3
```

SunOS unknown 5.10 Generic\_127128-11 i86pc 03/22/2009

```
10:54:25 sml_mem alloc fail lg_mem alloc fail ovsz_alloc fail
10:54:30 17293568 15564685 0 118423552 114753904 0 14131200 0
10:54:35 17293568 15565001 0 118423552 114753904 0 14131200 0
10:54:40 17293568 15564081 0 118423552 114748784 0 14131200 0

Average 17293568 15564589 0 118423552 114752200 0 14131200 0
# sar -m 5 3
```

SunOS unknown 5.10 Generic\_127128-11 i86pc 03/22/2009

```
10:54:42 msg/s sema/s
10:54:48 0.00 0.00
10:54:53 0.00 0.00
10:54:58 0.00 0.00
```

```
Average      0.00      0.00
# sar -v 5 3
```

```
SunOS unknown 5.10 Generic_127128-11 i86pc      03/22/2009
```

```
10:55:01  proc-sz    ov  inod-sz    ov  file-sz    ov  lock-sz
10:55:06  89/8218    0  6981/35596   0  881/881    0  0/0
10:55:11  89/8218    0  6981/35596   0  881/881    0  0/0
10:55:16  89/8218    0  6981/35596   0  881/881    0  0/0
```

```
# sar -y 5 3
```

```
SunOS unknown 5.10 Generic_127128-11 i86pc      03/22/2009
```

```
10:55:19  rawch/s canch/s outch/s rcvin/s xmtin/s mdmin/s
10:55:24      0      0     24      0      0      0
10:55:29      0      0     11      1      0      0
10:55:34      0      0     11      2      0      0
Average       0      0     16      1      0      0
```

Trouvez la signification de chaque option.

## Comptabilisation (Accounting)

La **Comptabilisation** sous Solaris 10 consiste en la création d'une historique de l'utilisation des ressources de la machine.

## Comptabilisation Basique

Pour consulter les données créées par la Comptabilisation Basique, il convient d'utiliser la commande **lastcomm** :

```
# lastcomm  
/var/adm/pacct: No such file or directory
```

Ce message indique que la Comptabilisation Basique n'est pas fonctionnelle.

Pour activer la Comptabilisation Basique, il convient d'exécuter la commande **/usr/lib/acct/startup** :

```
# /usr/lib/acct/startup
```

Exécutez ensuite quelques commandes :

```
# who  
root      console    juin  5 11:47  (:0)  
root      pts/3       juin  5 12:05  (:0.0)  
# pwd  
/
```

Puis consultez les données de la Comptabilisation Basique :

```
# lastcomm  
who      root      pts/3        0.00 secs Thu Jun  6 11:17  
startup  root      pts/3        0.00 secs Thu Jun  6 11:17  
rm      root      pts/3        0.00 secs Thu Jun  6 11:17  
rm      root      pts/3        0.00 secs Thu Jun  6 11:17  
rm      root      pts/3        0.00 secs Thu Jun  6 11:17  
turnacct  root      pts/3        0.00 secs Thu Jun  6 11:17
```

```
accton  S  root      pts/3          0.00 secs Thu Jun  6 11:17
```

Le système de Comptabilisation Basique n'a pas de service SMF associé :

```
# svcs -a | grep acct
# svcs -a | grep accounting
```

Pour cette raison, créez les scripts nécessaires dans **/etc/rc2.d** et **/etc/rc3.d** :

```
# ls -l /etc/init.d | grep acct
-rwxr--r-- 1 root      sys          625 juin 28 2011 acct
-rwxr--r-- 1 root      sys          397 juin 28 2011 acctadm
# cd /etc/rc2.d
# pwd
/etc/rc2.d
# ln -s ../init.d/acct S15acct
# cd ../rc3.d
# ln -s ../init.d/acct S15acct
```

## Comptabilisation Étendue

La **Comptabilisation Étendue** permet de surveiller les activités au niveau des :

- utilisateurs,
- groupes,
- tâches,
- zones.

La commande **acctadm** est utilisée pour gérer le système de Comptabilisation Étendue. L'option **-r** de la commande permet de visualiser la liste des ressources qui peuvent être surveillées :

```
# acctadm -r
process:
```

```
extended pid,uid,gid,cpu,time,command,tty,projid,taskid,ancpid,wait-status,zone,flag,memory,mstate
basic    pid,uid,gid,cpu,time,command,tty,flag
task:
extended taskid,projid,cpu,time,host,mstate,anctaskid,zone
basic    taskid,projid,cpu,time
flow:
extended saddr,daddr,sport,dport,proto,dsfield,nbytes,npkts,action,ctime,lseen,projid,uid
basic    saddr,daddr,sport,dport,proto,nbytes,npkts,action
```

L'utilisation de la commande **acctadm** sans options montre la configuration actuelle de la Comptabilisation Étendue :

```
# acctadm
          Comptabilisation des tâches : inactive
  Fichier de comptabilisation des tâches : none
          Ressources de tâche suivies : none
          Ressources de tâche non suivies : extended
          Comptabilisation des processus : inactive
  Fichier de comptabilisation des processus : none
          Ressources de processus suivies : none
          Ressources de processus non suivies : extended,host
          Comptabilisation du flux : inactive
  Fichier de comptabilisation du flux : none
          Ressources de flux suivies : none
          Ressources de flux non suivies : extended
```

La Comptabilisation Étendue peut être activée par groupe *basic* ou *extended* :

```
# acctadm -e extended -f /var/adm/exacct/task task
# acctadm
          Comptabilisation des tâches : actif
  Fichier de comptabilisation des tâches : /var/adm/exacct/task
          Ressources de tâche suivies : extended
          Ressources de tâche non suivies : none
          Comptabilisation des processus : inactive
```

```
Fichier de comptabilisation des processus : none
    Ressources de processus suivies : none
    Ressources de processus non suivies : extended,host
        Comptabilisation du flux : inactive
Fichier de comptabilisation du flux : none
    Ressources de flux suivies : none
    Ressources de flux non suivies : extended
```

ou en spécifiant les ressources :

```
# acctadm -e pid,uid,time,command,mstate -f /var/adm/exacct/proc process
# acctadm
    Comptabilisation des tâches : actif
Fichier de comptabilisation des tâches : /var/adm/exacct/task
    Ressources de tâche suivies : extended
    Ressources de tâche non suivies : none
    Comptabilisation des processus : actif
Fichier de comptabilisation des processus : /var/adm/exacct/proc
    Ressources de processus suivies : pid,uid,time,command,mstate
    Ressources de processus non suivies : gid,projid,taskid,cpu,tty,host,flag,ancpid,wait-status,zone,memory
        Comptabilisation du flux : inactive
Fichier de comptabilisation du flux : none
    Ressources de flux suivies : none
    Ressources de flux non suivies : extended
```

Afin de lire les données contenues dans les répertoires **/var/adm/exacct/proc** et **/var/adm/exacct/task**, il est nécessaire de compiler le logiciel **exdump** fourni dans le répertoire **/usr/demo/libexacct** :

```
# cd /usr/demo/libexacct
# ls
exdump.c  Makefile  README
```

L'utilisation de cet outil, une fois compilé est simple :

```
# ./exdump /var/adm/exacct/task
```

## La Commande DTrace

La commande **DTrace** est un outil de surveillance et de débogage. Elle comprend un langage de programmation simple appelé **D**. Voici un exemple de ce langage :

```
syscall:::  
/execname != "dtrace"/  
{  
    printf("%d %s %s\n", pid, probefunc, execname);  
}
```

Ce programme D peut surveiller tous les appels système et les imprimer à l'écran. La première ligne active le *probe syscall:::*. La deuxième ligne exclue de la sortie tout appel système effectué par DTrace elle-même. La troisième ligne imprimera à l'écran trois informations chaque fois qu'il y a un appel système, à savoir le PID du processus générant l'appel système, le nom de l'appel système et le nom du processus.

L'utilisation de ce programme se fait ainsi :

```
bash-3.2# /usr/sbin/dtrace -q -s exemple1.d | more  
1770 nanosleep VBoxClient  
1770 ioctl VBoxClient  
1770 ioctl VBoxClient  
1770 nanosleep VBoxClient  
423 nanosleep VBoxService  
2284 lwp_cond_wait java  
9979 lwp_cond_wait java  
1770 nanosleep VBoxClient  
1770 ioctl VBoxClient  
1770 ioctl VBoxClient  
1770 nanosleep VBoxClient  
1770 nanosleep VBoxClient
```

```
1770 ioctl VBoxClient  
1770 ioctl VBoxClient  
1770 nanosleep VBoxClient  
1770 nanosleep VBoxClient  
1770 ioctl VBoxClient  
1770 ioctl VBoxClient  
1770 nanosleep VBoxClient  
2158 lwp_cond_wait java  
2158 lwp_cond_wait java  
9979 lwp_cond_wait java  
9979 lwp_cond_wait java  
1770 nanosleep VBoxClient  
1770 ioctl VBoxClient  
1770 ioctl VBoxClient  
1770 nanosleep VBoxClient  
1770 nanosleep VBoxClient  
1770 ioctl VBoxClient  
1770 ioctl VBoxClient  
1770 nanosleep VBoxClient  
2158 lwp_cond_wait java
```

## Probes

Nous avons vu que la première ligne de notre programme **exemple1.d** est un **probe**. Le format de ce probe est :

```
provider:module:function:name
```

Un probe est quelque chose à surveiller. Il en existe des milliers pour le système Solaris 10 et ils sont mis à disposition de DTrace par un fournisseur de probes, le **provider**.

Pour visualiser la liste des probes, utilisez la commande **dtrace** avec l'option **-l** :

#	DTRACE -L	MODULE	FUNCTION NAME
1	dtrace		BEGIN
2	dtrace		END
3	dtrace		ERROR
4	nfsmapid437	nfsmapid	cb_update_domain daemon-domain
5	Xserver1500	Xorg	AddResource resource-alloc
6	Xserver1500	Xorg	WriteEventsToClient send-event
7	Xserver1500	Xorg	Dispatch request-start
8	Xserver1500	Xorg	AuthAudit client-auth
9	Xserver1500	Xorg	FreeClientResources resource-free
10	Xserver1500	Xorg	FreeClientNeverRetainResources resource-free
11	Xserver1500	Xorg	FreeResourceByType resource-free
12	Xserver1500	Xorg	FreeResource resource-free
13	Xserver1500	Xorg	Dispatch request-done
14	Xserver1500	Xorg	AllocNewConnection client-connect
15	Xserver1500	Xorg	CloseDownClient client-disconnect
16	fbt	bl	bl_attach entry
17	fbt	bl	bl_attach return
18	fbt	bl	bl_detach entry
19	fbt	bl	bl_detach return
20	fbt	bl	bl_getinfo entry
21	fbt	bl	bl_getinfo return
22	fbt	bl	bl_open entry
23	fbt	bl	bl_open return
24	fbt	bl	bl_ioctl entry
25	fbt	bl	bl_ioctl return
26	fbt	bl	_info entry
27	fbt	bl	_info return
28	fbt	kmdb	kdrv_open entry
29	fbt	kmdb	kdrv_open return
30	fbt	kmdb	kdrv_activate entry

```
31      fbt          kmdb          kdrv_activate return
--A suivre--
```

Pour consulter la liste des **probes** fournis par un **provider** spécifique, utilisez la commande suivante :

```
# dtrace -l -P fbt | more
ID  PROVIDER        MODULE           FUNCTION NAME
16  fbt            bl               bl_attach entry
17  fbt            bl               bl_attach return
18  fbt            bl               bl_detach entry
19  fbt            bl               bl_detach return
20  fbt            bl               bl_getinfo entry
21  fbt            bl               bl_getinfo return
22  fbt            bl               bl_open entry
23  fbt            bl               bl_open return
24  fbt            bl               bl_ioctl entry
25  fbt            bl               bl_ioctl return
26  fbt            bl               _info entry
27  fbt            bl               _info return
28  fbt            kmdb            kdrv_open entry
29  fbt            kmdb            kdrv_open return
30  fbt            kmdb            kdrv_activate entry
31  fbt            kmdb            kdrv_activate return
32  fbt            kmdb            kdrv_deactivate entry
33  fbt            kmdb            kdrv_deactivate return
34  fbt            kmdb            kdrv_ioctl entry
35  fbt            kmdb            kdrv_ioctl return
36  fbt            kmdb            kdrv_attach entry
37  fbt            kmdb            kdrv_attach return
38  fbt            kmdb            kdrv_detach entry
39  fbt            kmdb            kdrv_detach return
40  fbt            kmdb            _info entry
41  fbt            kmdb            _info return
42  fbt            vni              vniopen entry
```

```
43      fbt          vni          vniopen  return
44      fbt          vni          vniclose entry
45      fbt          vni          vniclose return
46      fbt          vni          vniwput entry
--A suivre--
```

**Module** spécifie quel module du noyau contient le probe. **Function** spécifie la fonction dans le module qui contient le probe.

## Predicates

Un **Predicate** est utilisé pour filtrer l'action du probe et se place juste après celui-ci. Prenons notre programme exemple1.d :

```
syscall:::
/execname != "dtrace"/
{
    printf("%d %s %s\n", pid, probefunc, execname);
}
```

Dans cet exemple, le probe est **syscall:::** et le predicate est **execname != “dtrace”**.

## Actions

Une **action** définit le comportement quand le probe est déclenché. Prenons notre programme exemple1.d :

```
syscall:::
/execname != "dtrace"/
{
    printf("%d %s %s\n", pid, probefunc, execname);
}
```

Ce programme peut être maintenant représenté d'une manière générique par :

```
PROBE  
/PREDICATE/  
{  
    ACTION;  
}
```

## Références

- [The Oracle Technology Network](#)
- 

<html> <center> Copyright © 2011-2018 I2TCH LIMITED.<br><br> </center> </html>