

Često korištene naredbe

```
ls
stat
chmod
chown
...
```

```
pvcreate
pvdisplay
vgcreate
vgdisplay
lvcreate
lvdisplay
dmsetup
```

```
fdisk
cfdisk
gdisk
cgdisk
mkfs.*
file
fsck
debugfs
mount
lsblk
lsscsi
lspci
blkid
df
du
resize2fs
dd
mkfifo
-----
chmod
chown
lsattr
chattr
-----
sar
iostat
pidstat
vmstat
blktrace
perf
```

```
hpacucli
ssaccli
perccli
megaccli
```

Ovisi o RAID kontroleru !

CPU 0
CPU 1
CPU 2

Block Device Layer

Programi/aplikacije

VFS (Virtual File system)

Direct I/O

Page cache

Opcionalni Virtual Block Layer

LVM DRBD MDADM ...

Datotečni sustav (File system)

XFS ext3/4 ZFS Btrfs NFS cifs ceph procfs sysfs tmpfs ramfs

Block File systems Network File systems Pseudo File systems "Special" File systems

Generic Block Layer

Blk-mq

Software queue

IO scheduler(s)

(noop/anticipatory/deadline/cfq)

IO scheduler(s)

(noop/anticipatory/deadline/cfq)

SCSI mid layer Block devices

/dev/sd* /dev/sr* /dev/st*

Mid layer Block devices

/dev/vd* /dev/nvme*

SCSI low layer Block device driver

sd_mod (sd) / sg (sg)

hpsa megaraid_sas libata virtio_scsi ata_piix

Low Level Device Drivers

nvme

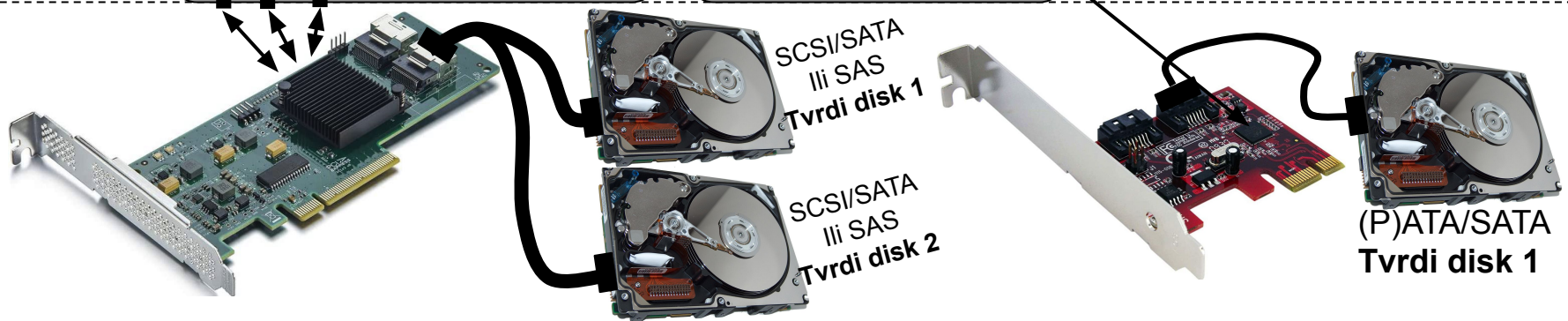
Device driver

Disk kontroler Hardware queues

Device driver

Disk Kontroler Hardware Queue

Device queue



Sysctl ili /proc ili /sys parametri

Postotak RAM memorije koja može biti korištena kao **page cache** memorija:
`vm.dirty_ratio=XX`
Nakon dostizanja zauzeća postotka definiranog ovdje, automatski se kreće sa čišćenjem cache memorije:
`vm.dirty_background_ratio=YY`
Brisanje cijele **page cache** memorije:
`echo 3 > /proc/sys/vm/drop_caches`
Promjena **cache pressure** mehanizma:
`vm.vfs_cache_pressure=ZZ`

Optimizacije na Filesystem sloju se rade na datotekama unutar vršnog direktorija: `/proc/sys/fs/` tj. u **sysctl** varijablama koje počinju sa: **fs.***
Maksimalan broj file deskriptora za cijeli sustav:
`fs.file-max = XX`
Maksimalan broj file deskriptora koje može otvoriti bilo koji (pojedini) program (proces):
`fs.nr_open=XX`
Vrijeme u sekundama, koliko će se čekati da program koji je zaključao datoteku zbog zapisivanja, da ju otključa, kako bi ju potom mogao otvoriti za zapisivanje neki drugi program:
`fs.lease-break-time=XX`

Optimizacije na **Generic Block** sloju te **I/O scheduleru** rade se na datotekama unutar vršnog direktorija koji ovisi o disku; za pr. `/dev/sda` je to: `/sys/block/sda/`
I/O CPU afinitet (**1**=CPU jezgra iz iste grupe, **2**=isključivo ista CPU jezgra koja je obradila konkretan I/O zadatak):
`/sys/block/sda/queue/rq_affinity`
Maksimalan broj I/O operacija na blok sloju:
`/sys/block/sda/queue/nr_requests`

Odabrani I/O scheduler (za `/dev/sda`) se zapisuje u:
`/sys/block/sda/queue/scheduler`

Za primjerice **HP RAID Smart Array** kontrolere, moguće je:
Uključiti **BBU RAM** memoriju na kontroleru:
`ssaccli ctrl slot=0 modify dwc=enable`
Za ostale parametre BBU, provjerite:
`ssaccli ctrl all show config detail | grep -i -e cache -e battery -e slot`
Mijenjati broj **I/O** kanala.
Provjerimo koliko ih ima:
`ssaccli ctrl all show detail | grep -i que`
Dodatno je moguće vidjeti koliko SCSI naredbi se može pohraniti u nîz za naredbe, vidljivo kao **queue_depth**, sa:
`lsscsi -l`
Moguće je balansirati upotrebu signala prekida (**IRQ**) [poglavlje:10.5.1.1]. Provjera koje sve **IRQ** koristi navedeni RAID kontroler:
`grep -i hpsa /proc/interrupts`