

# Blok uređaji i administracija fajl sistema

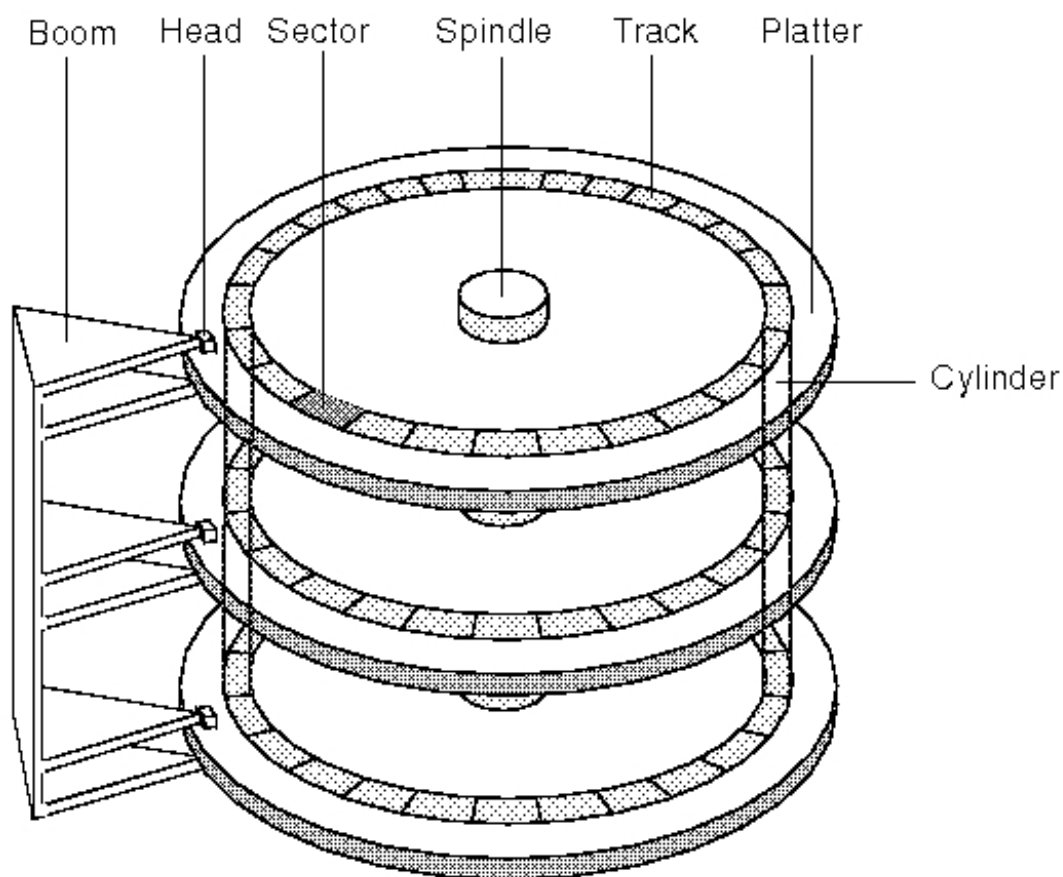
---

## Osnovni zadaci

---

Četiri su osnovna tipa zadataka u administraciji disk fajl sistema:

- **formatiranje diska na niskom nivou** - većina diskova koji se danas proizvode fabrički su preformatirani
- **podela diska na particije**
- **kreiranje fajl sistema** na particijama diska
- **aktiviranje fajl sistema** - montiranjem (mounting) na odgovarajuće direktorijume, čime se formira struktura aktivnog direktorijumskog stabla. Ovaj postupak se obavlja ili automatski, prilikom podizanja sistema (definisano u fajlu /etc/fstab), ili ručno, komandom mount.



## Anatomija diska

---

Diskovi se sastoje od particija koji pomažu u organizaciji podataka. Na disku mogu da postoje više particija i one ne mogu da se preklapaju. Prostor, na disku, koji nije dodeljen particijama se zove slobodan prostor (free space). Tip particije zavisi od particione tabele. Particionisanje diska je konvencija koje se pridržava većina operativnih sistema

uključujući i UNIX i MS Windows. Unutar particija se nalazi fajl sistem ili se particiji dodeljuju posebna zaduženja (npr swap).

## Particiona tabela

Svaki disk poseduje particionu tabelu, tabelu koja govori sistemu kako je disk podeljen. U ovoj tabeli se nalaze informacije o tome gde particija počinje, gde se završava, koja je particija boot-abilna, koji sektori diska su dodeljeni kojoj particiji isl. Postoje dva osnovna tipa particione tabele, *Master Boot Record (MBR)* i *GUID Partition table (GPT)*.

### MBR

- Informacije o svim particijama diska čuvaju se u **prvom logičkom sektoru**, tj. u prvom sektoru prve staze sa prve površine diska. Ovaj sektor je poznat pod imenom Master Boot Record (MBR) i njemu BIOS pristupa prilikom boot procedure.
- **MBR sadrži mali program** (Windows boot loader, GRUB boot loader) koji očitava particionu tabelu, proverava koja je particija aktivna, i očitava prvi sektor aktivne particije (boot sektor).
- U boot sektoru se nalazi program čijim pokretanjem započinje **boot-strap**, odnosno punjenje RAM memorije operativnim sistemom.
- **Informacije o particionoj tabeli** mogu se dobiti pomoću komande *fdisk -l*, recimo:

```
$ sudo fdisk -l /dev/sda
```

```
Disk /dev/sda: 465,8 GiB, 500107862016 bytes, 976773168 sectors
```

```
Units: sectors of 1 * 512 = 512 bytes
```

```
Sector size (logical/physical): 512 bytes / 512 bytes
```

```
I/O size (minimum/optimal): 512 bytes / 512 bytes
```

```
Disklabel type: dos
```

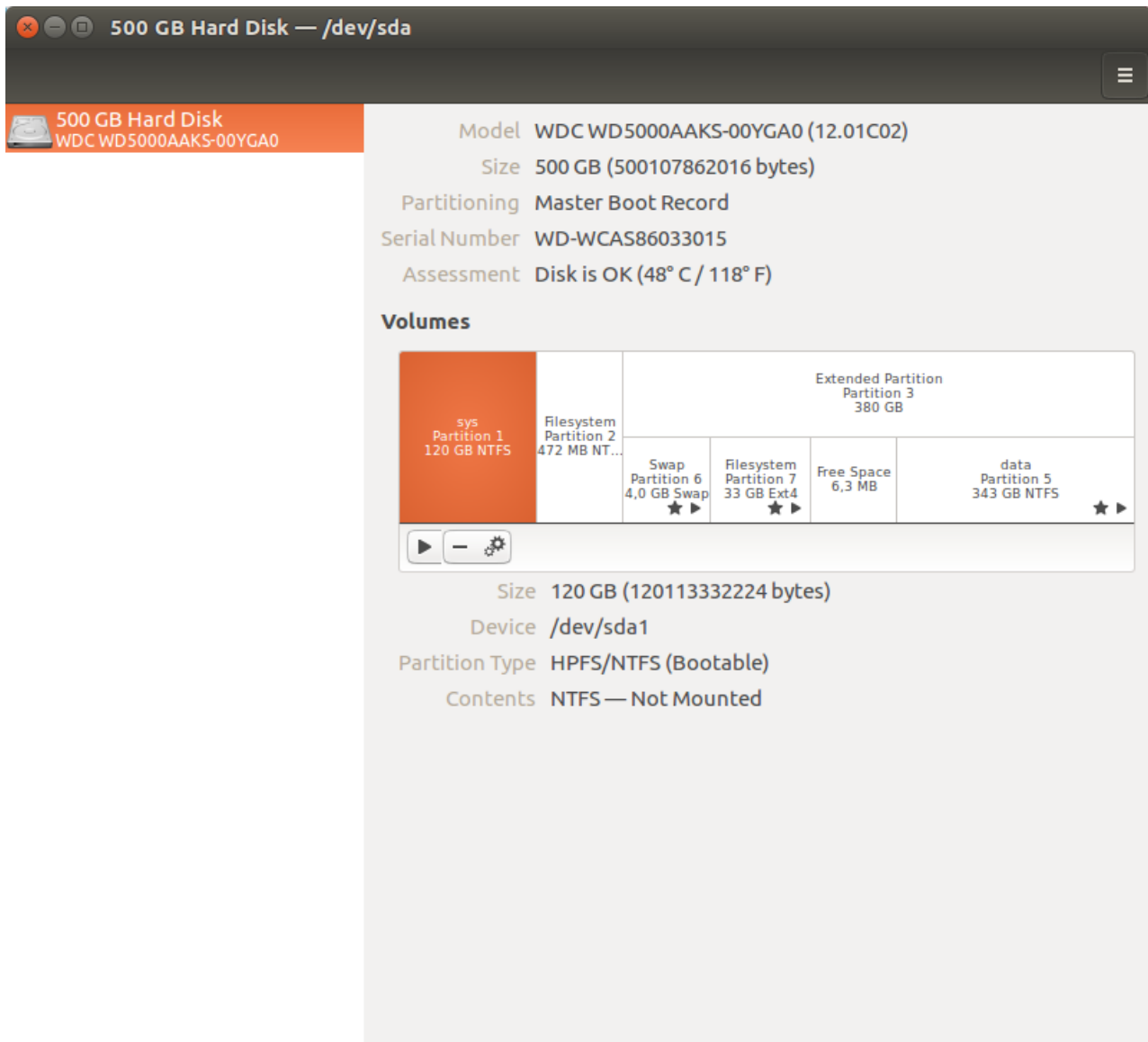
```
Disk identifier: 0x0c210c20
```

Device	Boot	Start	End	Sectors	Size	Id	Type
/dev/sda1	*	2048	234598399	234596352	111,9G	7	HPFS/NTFS/exFAT
/dev/sda2		234598400	235519999	921600	450M	27	Hidden NTFS WinRE
/dev/sda3		235522046	976773119	741251074	353,5G	5	Extended
/dev/sda5		307208192	976773119	669564928	319,3G	7	HPFS/NTFS/exFAT
/dev/sda6		235522048	243333119	7811072	3,7G	82	Linux swap / Solaris
/dev/sda7		243335168	307195903	63860736	30,5G	83	Linux

Partition table entries are not in disk order.

- **Originalan koncept particionisanja (MBR)** diskova na PC računarima dozvoljavao je najviše 4 particije po jednom disku.
- Međutim, teško je na samo 4 particije instalirati više operativnih sistema (naročito ako neki od njih zahtevaju dodatne particije, kao što su swap i boot particije) i boot manager (npr. GRUB), i odvojiti particiju za korisničke podatke.

- **Problem je rešen uvođenjem extended particije** koja služi kao okvir u kome se mogu kreirati nekoliko logičkih particija. Logičke particije se ponašaju kao primarne, a informacije o njima čuvaju se u boot sektoru extended particije, koji se još naziva i extended partition table. Na disku može postojati najviše jedna extended particija!
- Disk na primeru sa slike (iz programa *gnome-disk*) se ponaša kao da na njemu postoji pet primarnih particija, pri čemu osnovni koncept particionisanja nije narušen - u particionoj tabeli se vodi evidencija o samo tri particije.



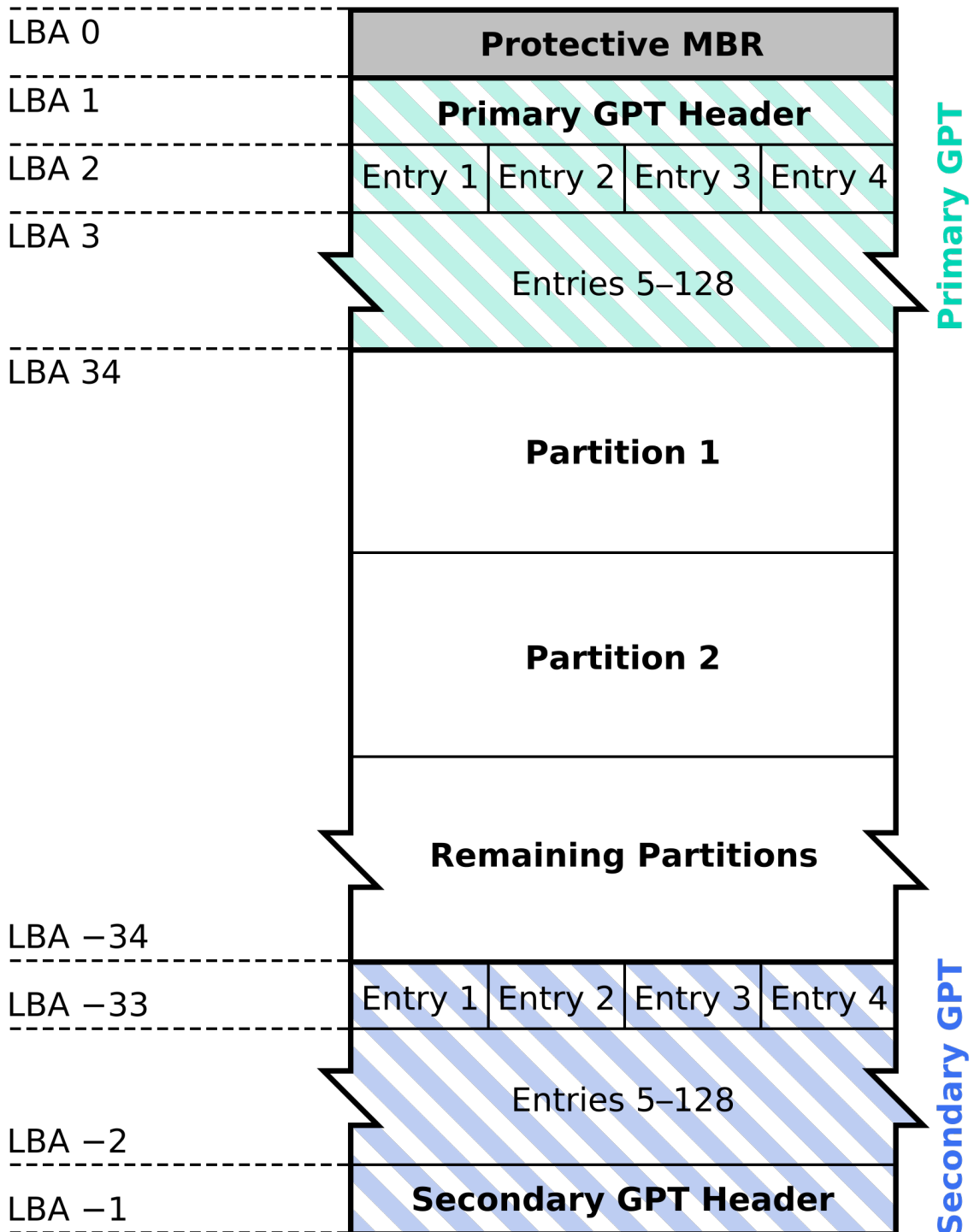
Jedno od najvećih ograničenja MBR-a je korišćenje 32 bita za skladištenje adresa blokova. Ako hard disk ima sektore od 512 bajtova MBR može da adresira samo 2TB ( $2^{32} \times 512$  B).

## GUID Partition Table (GPT)

- **GPT** je deo UEFI standarda (standard koji menja BIOS) i koristi 64 bita za skladištenje adresa blokova, što znači da je za disk sa 512-bajtnim sektorima maksimalna veličina 9.4ZB.

- Da bi se obezbedila kompatibilnost sa MBR-om i BIOS-om prvi sektor diska je rezervisan za "protective MBR".
- Zove se GUID Partiton Table zato što svaka particija na disku ima *globalni jedinstven identifikator* (GUID).
- GPT za adresiranje koristi **LBA** (Logical Block Addressing), pa se protective MBR nalazi na LBA0.

# GUID Partition Table Scheme



- **GPT Header** - lako se prepozna jer uvek počinje sa **EFI PART**. Definiše broj i veličinu particija koje se nalaze u tabeli. Takodje, sadrži svoju veličinu i lokaciju (uvek LBA1) i veličinu i lokaciju drugog GPT headera koji se uvek nalazi u zadnjem sektoru diska. Važno je napomenuti da sadrži i CRC32 sumu koja uključuje header i tabelu i koja se proverava od strane operativnog sistema prilikom startovanja. Ako se desi da suma nije tačna primarni header će biti zamenjen sekundarnim.
- GPT veoma jednostavno opisuje particije. Na početku se nalazi GUID tipa particije, zatim jedinstven GUID particije, nakon toga slede prvi i zadnji LBA particije i naziv particije.

## Struktura particija

---

Svaka particija na disku sadrži sledeće blokove:

- **Boot blok** - Lociran je na prvim nekoliko sektora particije. Sadrži informacije potrebne za boot-ovanje operativnog sistema koji se nalazi na toj particiji. Ako postoji jedan operativni sistem, a više particija, samo jedan boot block će da sadrži informacije potrebne za boot-ovanje sistema, a ostali će biti neiskorišćeni.
- **Super blok** - Jedan blok nakon boot block-a. Sadrži informacije o fajl sistemu kao što su veličina inode tabele, veličina logičkih blokova i veličina fajl sistema.
- **Inode table** - Skup podataka koji upravljaju fajlovima.
- **Data blokovi** - fajlovi i direktorijumi.

## Tipovi particija

---

Super blok sadrži jedan bajt po particiji koji identifikuje tu particiju. Na taj način se identifikuje koji operativni sistem koristi particiju i u koje svrhe (npr. kao fajl sistem ili swap prostor). Najčešće korišćene vrednosti su:

0	prazna particija, tj. neiskorišćen prostor
---	--

5	Extended
---	----------

80	Old MINIX
----	-----------

81	Linux / Minix
----	---------------

82	Linux swap
----	------------

83	Linux native
----	--------------

85	Linux extended
----	----------------

fd	Linux raid auto
----	-----------------

a5	FreeBSD
----	---------

a6	OpenBSD
a9	NetBSD
1	DOS 12bit FAT
4	DOS 16bit FAT (za sisteme datoteka manje od 32MB)
6	DOS 16bit FAT (za sisteme datoteka veće od 32MB)
7	HPFS / NTFS
64	Novell
a	OS/2 boot manager

## fdisk alat

**fdisk** je interaktivni alat (radi pomoću tekstualnih menija). Nešto napredniji alat s korisničke tačke gledišta, dostupan u većini distribucija je **cdisk**. Standardni **fdisk** omogućava:

- **p** - print the partition table - prikazivanje particione tabele,
- **l** - list known partition types - pregled podržanih tipova particija,
- **n** - add a new partition - kreiranje primarnih, extended i logičkih particija,
- **d** - delete a partition - brisanje particija,
- **t** - change a partition's system id - promena tipa particija,
- **a** - toggle a bootable flag - postavljanje flegla aktivne particije.

Promene se ne upisuju na disk dok korisnik ne napusti program opcijom **w** - write table to disk and exit. Napuštanje programa opcijom **q** - quit without saving changes ne povlači upisivanje promena na disk.

**Promena veličine particije** pomoću fdisk alata može se izvršiti, ali ne na potpuno trivijalan način. Koraci koje je potrebno redom učiniti su sledeći:

- kreiranje *backup-a* svih podataka sa particije,
- brisanje particije,
- kreiranje nove particije,
- povratak podataka na novu particiju.

Ako je pri tome potrebno da se neka druga particija smanji, postupak je još složeniji. Za pomeranje granica i promenu struktura u fajl sistemima se koriste drugi, napredniji alati, npr. pomenuti *gparted* (isporučuje se u okviru GNOME grafičkog okruženja).

## Specijalni fajlovi i particije diska

- **Svaka particija predstavljena je jednim specijalnim fajlom u direktorijumu /dev.**
- **Konvencija o imenima nodova** (fajlova u /dev direktorijumu) za particije kaže: na ime diska treba dodati broj particije. Brojevima od 1-4 označavaju se primarne i extended particije, a brojevima većim od 5 logičke. Npr. /dev/hda1 predstavlja prvu particiju na primary master disku, /dev/sdb7 treću logičku particiju na drugom SCSI disku.

Uređaj	Primarne particije	Logičke particije
IDE Primary Master	/dev/hda[1-4]	/dev/hda[5-16]
IDE Primary Slave	/dev/hdb[1-4]	/dev/hdb[5-16]
IDE Secondary Master	/dev/hdc[1-4]	/dev/hdc[5-16]
IDE Secondary Slave	/dev/hdd[1-4]	/dev/hdd[5-16]
Prvi SCSI disk	/dev/sda[1-4]	/dev/sda[5-16]
Drugi SCSI disk	/dev/sdb[1-4]	/dev/sdb[5-16]
Treći SCSI disk	/dev/sdc[1-4]	/dev/sdc[5-16]
Četvrti SCSI disk	/dev/sdd[1-4]	/dev/sdd[5-16]

## Fajl sistemi

Fajl sistem predstavlja način organizacije datoteka na sekundarnim memorijskim medijumima (njime je određen skup metoda i struktura podataka koje operativni sistem koristi za čuvanje fajlova). Sadrži:

- **zaglavlje** - podaci neophodni za funkcionisanje sistema datoteka,
- **strukture za organizaciju podataka** na medijumu (*metadata area*)
- **same podatke**, odnosno datoteke i direktorijumi.

Zahvaljujući VFS-u (*Virtual File System*), sve fajl sisteme OS posmatra na isti način, bez obzira da kog su tipa, da li se nalaze na lokalnom disku računara ili na mreži. Osnovni fajl sistem posmatra se kao nezavisna hijerarhijska struktura objekata (direktorijuma i fajlova) na čijem se vrhu nalazi root direktorijum (/). U objekte spadaju:

- **regularni fajlovi**,
- **direktorijumi**,
- **hard linkovi** (alternativna imena fajlova)

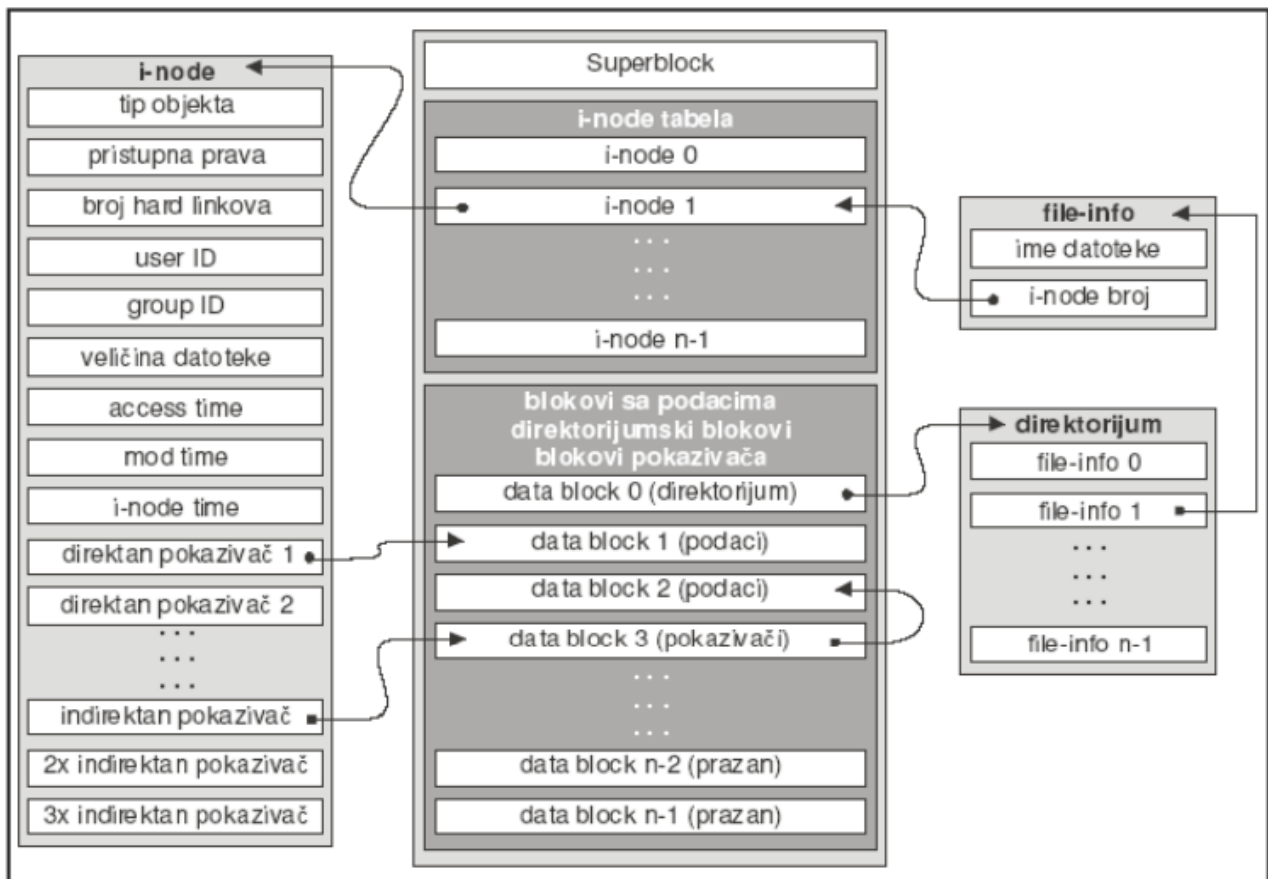


- **simbolički linkovi** (prečice, fajlovi koje sadrže putanje i imena objekata na koje upućuju)
- **blok i karakter specijalne datoteke** (opisuju uređaje, odnosno drajvere u kernelu). Korišćenjem ovih fajlova mogu se vršiti ulazno-izlazne operacije na uređajima koje opisuju. UNIX drajver pomoću specijalne datoteke korisniku predstavlja uređaj kao tok bajtova (stream), odnosno datoteku),
- **imenovani pipeline.**

## Inode tabela (tabela indeksnih čvorova)

---

- **Inode je osnovna struktura UNIX sistema datoteka koja opisuje jedan objekat** - indeksni čvor je drugi deo FCB-a. Sadrži sve informacije o objektu koji opisuje osim imena, i to sledeće:
  - **tip objekta** (npr. regularna datoteka, direktorijum ili simbolički link) i pristupna prava,
  - **broj hard linkova** na dati objekat,
  - **user ID**, odnosno ID korisnika koji je vlasnik objekta,
  - **group ID**, odnosno ID grupe korisnika kojoj objekat pripada,
  - **veličinu objekta** izraženu u bajtovima,
  - **vreme zadnjeg pristupa objektu** (access time) u UNIX vremenskom formatu,
  - **vreme zadnje modifikacije objekta** (mod time) u UNIX vremenskom formatu,
  - **vreme zadnje modifikacije indeksnog čvora objekta** (i-node time) u UNIX vremenskom formatu,
  - **listu direktnih pokazivača na blokove sa podacima**, koja je dovoljna da se adresiraju prvih 10-12 blokova podataka koji čine početak fajla (broj zavisi od tipa fajl sistema),
  - **listu indirektnih pokazivača** (lista pokazivača na jednostruke, dvostruke i trostruke indirektne blokove).
- **blokovi sa podacima** (data blocks). Svaki objekat koji se nalazi u direktorijumu (*directory-entry*) predstavljen je jednom *file-info* strukturom. Svaka *file-info* struktura sadrži ime objekta koji predstavlja i broj indeksnog čvora kojim je taj objekat u potpunosti opisan.
- **direktorijumski blokovi** (directory blocks),
- **blokovi indirektnih pokazivača** (indirection block).



## Tipovi fajl sistema

### Domaći (native) fajl sistemi

**Domaći fajl sistemi (native)** koji mogu se aktivirati na većini UNIX sistema su:

- **minix** - najstariji
- **xia** - modifikovana varijanta minixa
- **ext2** - Linux second extended
- **ext3** - "ext2 + journaling"
- **ext4** - najnoviji, ugrađen u Linux kernel 2.6.30
- **ReiserFS** - journaling sistem datoteka
- **XFS** - 64-bitni journaling sistem
- **BTRFS** - B-tree file system (Copy on Write)

**Journaling** predstavlja vođenje dnevnika transakcija. **Dnevnik transakcija** prati aktivnosti vezane za promenu meta-data oblasti, odnosno i-node tabele i objekata fajl sistema. Dnevnik prati relativne promene u fajl sistemu u odnosu na poslednje stabilno stanje. Transakcija se zatvara po obavljenom upisu i može biti ili u potpunosti prihvaćena ili odbijena.

### Strani (foreign) fajl sistemi

Ugrađena je podrška za nekoliko tipova stranih sistema datoteka čime je omogućena relativno laka razmena fajlova sa drugim operativnim sistemima; ponašaju se slično domaćim ali ne moraju imati sve funkcije domaćih fajl sistema (npr. hard linkove) i mogu imati ograničenja kojih nema na domaćim sistemima:

- **msdos** - razmena datoteka sa DOS i OS/2 FAT sistemom datoteka, read-write
- **umsdos** - proširenje msdos, dodata je podrška za duga imena datoteka, vlasništvo, pristupna prava, linkove i specijalne fajlove. Može se koristiti kao Linux native sistem datoteka, a neke distribucije Linux sistema dozvoljavaju instalaciju operativnog sistema na njemu
- **vfat** - FAT32
- **iso9660** - standard za CD-ROM sisteme datoteka
- **hpfs** - OS/2 High Performance File System.
- **ntfs** - NTFS
- **nfs** - UNIX mrežni fajl sistem koji omogućava deljenje lokalnog fajl sistema između većeg broja umreženih računara i brz pristup udaljenim fajlovima
- **smbfs** - mrežni sistem datoteka koji omogućava deljenje lokalnog fajl sistema sa umreženim računarima koji rade pod *MS Windows* operativnim sistemom. Koristi *Windows* protokol za deljenje fajlova.

## Kreiranje fajl sistema

---

**mkfs - kreiranje, inicijalizacija fajl sistema;** mkfs je *front-end* koji poziva odgovarajuće programe za kreiranje traženog sistema.

```
mkfs [-t fstype] [-c | -l bblst] device
```

gde je:

- **device** specijalni fajl koja predstavlja particiju na kojoj se kreira fajl sistem. Tim fajlom se kasnije predstavlja i fajl sistem
- **-t fstype** - tip sistema datoteka koji je potrebno kreirati fstype može biti ext2, ext3, reiser, msdos ili bilo koji drugi tip za koji u operativnom sistemu postoji podrška
- **-c** - pre kreiranja sistema datoteka se ispituje površina diska i inicijalizuje lista neispravnih blokova
- **-l bblst** opcija kojom se specificira fajl sa inicijalnom listom neispravnih blokova.

```
$ fdformat -v /dev/fdOH1440
```

```
Double-sided, 80 tracks, 18 sec/track. Total capacity 1440 kB.  
Formatting ... done
```

```
$ mkfs -t ext2 -c /dev/fdOH1440
```

mke2fs 0.5a, 5-Apr-94 for EXT2 FS 0.5, 94/03/10  
360 inodes, 1440 blocks  
72 blocks (5.00%) reserved for the super user  
First data block=1  
Block size=1024 (log=0)  
Fragment size=1024 (log=0)  
1 block group  
8192 blocks per group, 8192 fragments per group  
360 inodes per group  
Checking for bad blocks (read-only test): done  
Writing inode tables: done  
Writing superblocks and filesystem accounting information: done

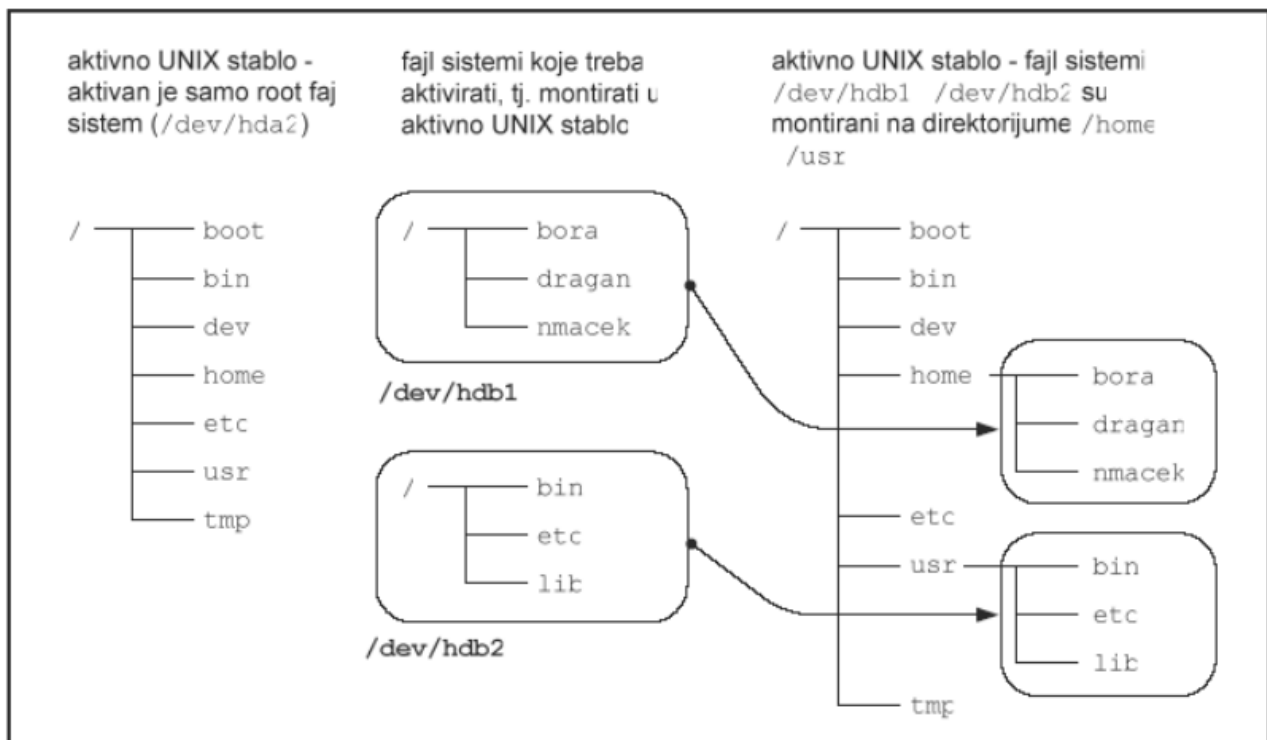
```
$ badblocks /dev/hda2 > /tmp/bad-block-list1  
$ mkfs -t ext2 -l /tmp/bad-block-list1 /dev/hda2
```

## Montiranje fajl sistema na aktivno UNIX stablo

**Pre korišćenja fajl sistem treba aktivirati.** Fajl sistemi se aktiviraju montiranjem na *mount-point* direktorijume. Montiranjem fajl sistema na *mount-point* direktorijume stvara se aktivno UNIX stablo koje čine svi aktivirani fajl sistemi.

Primer tri fajl sistema (npr. fajl sistemi na particijama /dev/hda2, /dev/hdb1 i /dev/hdb2).

```
$ mount /dev/hdb1 /home  
$ mount /dev/hdb2 /usr
```



```
$ mount [-r] [-t fstype] devicenode mountpoint
```

Nakon izvršenja komande kernel će montirati fajl sistem koji se nalazi na uređaju ili particiji čiji je nod *devicenode* na direktorijum *mountpoint*. Direktorijum *mountpoint* mora biti kreiran u aktivnom UNIX stablu pre montiranja fajl sistema i preporučljivo je da bude

prazan. Ukoliko *mountpoint* nije prazan, prethodni sadržaj, vlasnik i pristupna prava biće nevidljivi za korisnike sve dok je montirani sistem datoteka aktivan. Putanja direktorijuma *mountpoint* će posle montiranja postati putanja ka root direktorijumu montiranog sistema datoteka.

Montiranja diskete sa *msdos* fajl sistemom u mountpoint */media/floppy*, postiže se sa:

```
$ mount -t msdos /dev/fd0 /media/floppy
```

Bez parametra *-r* komanda *mount* će pokušati da aktivira fajl sistem za čitanje i pisanje, ukoliko za to postoji podrška u kernelu. Ako se fajl sistem nalazi na medijumu kao što je CD-ROM ili postoji potreba da se zabrani pisanje na sistem datoteka, sistem datoteka se pomoću opcije *-r* može aktivirati samo u režimu čitanja (*readonly*). Kernel će tada zaustaviti sve pokušaje pisanja na taj fajl sistem.

## Root i user fajl sistem

---

- **Root fajl sistem (/)** nastaje prilikom instalacije operativnog sistema.
  - To je prvi fajl sistem koji se aktivira prilikom podizanja sistema - montira se na root direktorijum aktivnog UNIX stabla, i u toku rada se ne može deaktivirati
  - Da bi kernel znao gde se root fajl sistem nalazi, nod root fajl sistema treba specificirati u boot manageru (kao što je GRUB)
  - Na root fajl sistemu nalaze se sistemski podaci i njegova struktura je strogo određena
  - Pun pristup ima samo superuser root
- **User sistem datoteka**
  - se po potrebi može aktivirati i deaktivirati u toku rada, ukoliko je korisnik za to dobio dozvole od superusera, npr. particija na lokalnom disku, eksterni HDD ili flash drive.
  - Pristupna prava u okviru user sistema datoteka određuje root.

## /etc/fstab i auto-mount

---

U fajlu */etc/fstab* opisani su fajl sistemi koji će se automatski aktivirati prilikom boot procedure. Aktiviranje svih fajl sistema opisanih u */etc/fstab*, u toku rada, se vrši komandom `mount -a`.

Fajl **/etc/fstab** sadrži sledeće informacije:

- device** - uređaj ili particija koja sadrži fajl sistem. Ubuntu koristi UUID da identifikuje particije. Alternativno uređaj može da se identifikuje preko:
- Labele: LABEL=labela,
  - Mrežnog identifikatora: Samba (*//server/share*), NFS (*server:/share*) ili SSHFS (*sshfs#user@server:/share*)
  - Uređaja (*/dev/sdxy*) - nije preporučljivo.
  - Lista uređaja i njihovih UUID se dobija alatom **blkid**.

\$ sudo blkid

- **mount point** - direktorijum na root fajl sistemu (mount point) preko kog će moći da se pristupi sadržaju uređaja. Za swap u ovo polje treba upisati none.
- **type** tip sistema datoteka (za izmenljive medijume koji podržavaju rad sa nekoliko tipova sistema datoteka (kao što su flopi diskovi) ovo polje je auto - kada se uređaj aktivira detektuje se koji je tip.
- **options** - opcije, neke od njih su
  - **noauto** zabranjuje aktiviranje prilikom podizanja OS-a i aktiviranje fajl sistema komandom *mount -a*. Sistem datoteka sa opcijom noauto može se aktivirati samo ručno.
  - **user** dozvoliće svim korisnicima da aktiviraju taj fajl sistem, što inače može da obavi samo superuser root
  - **usrquota** - aktiviranje sa limitom prostora na disku
  - **ro** - dozvoliće aktiviranje sistema datoteka isključivo u režimu čitanja.
- **dump** - da li će sistem datoteka biti uključen u listu za back-up, odnosno dump (vrednost polja je 1) ili ne (vrednost polja je 0)
- **pass num** - red kojim će fsck proveriti integritet sistema datoteka pri podizanju operativnog sistema: root sistem - 1, ostali - 2. Ako je fs\_passno 0, integritet sistema datoteka neće biti proveren.

```
# /etc/fstab: static file system information.
#
# Use 'blkid' to print the universally unique identifier for a
# device; this may be used with UUID= as a more robust way to name devices
# that works even if disks are added and removed. See fstab(5).
#
# <file system> <mount point> <type> <options>    <dump> <pass>
# / was on /dev/sda7 during installation
UUID=da246863-0c36-4d2f-aa1d-869ea3c97603 /          ext4    errors=remount-ro 0    1
# /mnt/data was on /dev/sda5 during installation
UUID=5D96DC3B67B9E6BC /mnt/data    ntfs    defaults,umask=007,gid=46 0    0
# swap was on /dev/sda6 during installation
UUID=9f238d52-1e39-4661-9864-bbcc62613843 none        swap    sw          0    0
```

Takođe, komanda **mount -l** daje listu aktiviranih file sistema sa labelama. Deaktiviranje fajl sistema vrši se komandom **umount**, koja zahteva da se navede jedan argument: nod za fajl sistem ili *mount-point* direktorijum.

\$ umount /dev/hdb3 ili \$ umount /inst\_packages

## Dozvole za aktiviranje fajl sistema

Procedure aktiviranja i deaktiviranja sistema datoteka zahtevaju privilegije superusera. Regularnim korisnicima može se omogućiti aktiviranje sistema datoteka:

- **davanjem lozinke superusera root**
- **sudo** (/etc/sudoers)

- davanjem dozvola svim korisnicima za montiranje sistema datoteka u fajlu `/etc/fstab`
- polju **fs\_mntops** navedena opcija **user**.

## Još neke komande

---

Slobodan prostor na particijama:

```
$ df -h
```

Filesystem	Size	Used	Avail	Use%	Mounted on
udev	2,0G	0	2,0G	0%	/dev
tmpfs	396M	6,5M	389M	2%	/run
/dev/sda7	30G	12G	17G	42%	/
tmpfs	2,0G	2,5M	2,0G	1%	/dev/shm
tmpfs	5,0M	4,0K	5,0M	1%	/run/lock
tmpfs	2,0G	0	2,0G	0%	/sys/fs/cgroup
/dev/sda5	320G	244G	76G	77%	/mnt/data
tmpfs	396M	72K	396M	1%	/run/user/1000
/dev/sdb1	7,4G	129M	7,2G	2%	/media/marko/MARKO

Iskorišćen prostor u direktorijumu:

```
$ du -h /
```

Filesystem	Size	Used	Avail	Use%	Mounted on
/dev/sda7	30G	12G	17G	42%	/

## Virtuelna memorija (swap)

---

Virtuelna memorija (swap) je deo diska ili sistema datoteka koji služi za privremeno skladištenje neaktivnih procesa, čime se prividno povećava količina operativne memorije. Swap se može realizovati na dva načina:

- u formi fajlova u postojećem fajl sistemu - veličina swap fajla može se lako povećati, što u slučaju swap particije nije tako jednostavno
- kao kvazi-fajl sistem na posebnoj particiji - funkcioniše brže nego swap fajl, jer se zaobilaze rutine za pristup objektima fajl sistema

**Swap particija** se naknadno može kreirati fdisk-om ili nekog drugog programa za rad sa particijama (cfdisk, sfdisk), pri čemu kreirana particija treba da bude odgovarajućeg tipa (82 - Linux swap). Nakon toga je potrebno pokrenuti **mkswap** koji će kreirati logičku strukturu swap prostora, a zatim aktivirati swap prostor komandom **swapon**.

```
$ mkswap /dev/sda2
$ swapon /dev/sda2
```

Swap se po potrebi može isključiti:

```
$ swapoff /dev/sda2
```

**free** - informacije o količini i zauzetosti operativne memorije i swap prostora:

```
$ free
```

	total	used	free	shared	buff/cache	available
Mem:	4045940	1976512	602004	27524	1467424	1729040
Swap:	3905532	0	3905532			

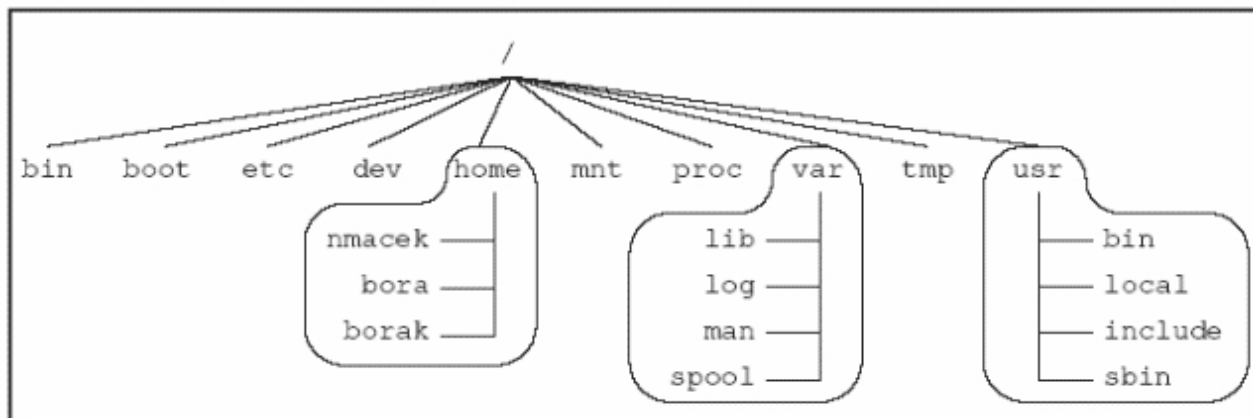
## Aktivno UNIX stablo

**Filesystem Hierarchy Standard v2.1.** FHS standard definiše organizaciju aktivnog UNIX stabla i podelu stabla na nekoliko sistema datoteka specifične namene koje treba kreirati na odvojenim particijama ili diskovima.

Po FHS standardu aktivno stablo čine sledeći sistemi datoteka:

- **/ (root)**, čiji se koreni direktorijum poklapa sa korenim direktorijumom aktivnog UNIX stabla. Root sistem datoteka sadrži osnovnu direktorijumsku strukturu stabla i sve fajlove neophodne za podizanje sistema i dovođenje sistema u stanje u kom ostali fajl sistemi mogu biti aktivirani,
- **/usr** - većina korisničkih komandi, biblioteke, dokumentacija (man pages) i ostale relativno nepromenljive datoteke neophodne za normalno funkcionisanje sistema,
- **/var** - promenljive datoteke, poput spool direktorijuma (npr. red za štampu), log datoteka i nekih privremenih datoteka,
- **/home** - korisnički podaci odnosno lični direktorijumi svih korisnika sistema. Realizacijom ovog direktorijuma u vidu posebnog sistema datoteka olakšavaju se postupci arhiviranja podataka,
- **/bin** - najčešće korišćene komande koje mogu koristiti regularni korisnici,
- **/sbin** - komande namenjene superuseru, ali ih po potrebi mogu koristiti i obični korisnici ukoliko im se za to daju dozvole. /sbin se ne nalazi u putanji regularnih korisnika, ali se nalazi u putanji superusera
- **/etc** - konfiguracione datoteke
- **/root** - lični direktorijum korisnika root (superuser)
- **/lib** - deljene biblioteke neophodne za rad programa iz root fajl sistema
- **/lib/modules** - kernel moduli
- **/dev** - specijalni fajlovi (nodes).
- **/tmp** - privremeni fajlovi
- **/boot** - fajlovi koje koristi boot loader (npr. GRUB), uključujući i slike kernela. Poželjno je ovaj direktorijum realizovati u formi odvojenog fajl sistema koji će se nalaziti u okvirima prvih 1024 cilindara diska
- **/mnt** - direktorijum u kome se nalaze *mount-point* direktorijumi (npr. /mnt/windowsC, /mnt/floppy, /mnt/cdrom). U nekim distribucijama Linux sistema ovaj direktorijum je preimenovan u /media.





## /etc direktorijum - neki zanimljivi fajlovi

/etc sadrži većinu konfiguracionih datoteka, uključujući i konfiguracione datoteke mrežnog okruženja.

- **/etc/fstab** - tabela auto-mount sistema datoteka (filesystem table).
- **/etc/group** - datoteka u kojoj su opisane sve korisničke grupe.
- **/etc/init** - konfiguraciona datoteka init procesa.
- **/etc/issue** - tekst koji proces getty prikazuje pre linije za unošenje korisničkog imena prilikom prijavljivanja na sistem (login prompt).
- **/etc/motd** - poruke od kojih se jedna prikazuje nakon uspešnog prijavljivanja na sistem (message of the day). Sadržaj ove datoteke određuje administrator sistema.
- **/etc/mtab** - tabela aktiviranih fajl sistema (mount table). Koriste je razni programi kojima trebaju informacije o aktiviranim fajl sistemima (npr. df)
- **/etc/rc, /etc/rc.d ili /etc/rc?.d** - skript fajlovi koji se pokreću prilikom podizanja sistema ili promene nivoa izvršenja, ili direktorijumi u kojima se te skript datoteke nalaze
- **/etc/passwd** - datoteka u kojoj su opisani svi korisnici sistema
- **/etc/profile** - fajl koji se izvršava po pokretanju bash shella. Ovo je globalni fajl, tj. izvršavaće se bez obzira na to koji se korisnik prijavljuje na sistem
- **/etc/securetty** - identifikacija "sigurnih" terminala, odnosno terminala s kojih root sme da se prijavi na sistem (secure getty).
- **/etc/shadow** - šifrovane lozinke korisnika sistema.
- **/etc/shells** - spisak komandnih interpretera kojima se veruje, tj. koje korisnici mogu navesti kao podrazumevane terminale komandom chsh.
- **/etc/termcap** - baza podataka o mogućnostima terminala (terminal capabilities).

## /usr fajl sistem

/usr fajl sistem je relativno veliki, s obzirom da je većina programa koji pripadaju distribuciji Linux sistema tu instalirana. Lokalno instaliran softver smešta se u **/usr/local**, tako da se olakšava nadogradnja (upgrade) Linux distribucije novijom verzijom. Ovaj fajl sistem može se nalaziti na mreži i računari ga mogu aktivirati u režimu čitanja kao mrežni sistem datoteka, čime se štedi na utrošenom prostoru na lokalnim diskovima i olakšava administracija. Tada je dovoljno izvršiti promene na jednom mestu i one će biti vidljive na

svim računarima.

- **/usr/X11R6** - X Windows sistem
- **/usr/bin** - većina korisničkih programa (user binaries).
- **/usr/sbin** - komande za administraciju raznih serverskih programa (superuser binaries).
- **/usr/share/man, /usr/share/info, /usr/share/doc** - on-line dokumentacija (man pages, GNU info, ...).
- **/usr/include** - zaglavlja za programe pisane u C programskom jeziku.
- **/usr/lib** - relativno nepromenljive prateće datoteke raznih programa, uključujući i neke konfiguracione datoteke.
- **/usr/local** - mesto za instalaciju softvera koji ne spada u Linux distribuciju.

## /var fajl sistem

U /var fajl sistemu se nalaze datoteke koje se menjaju prilikom regularnog funkcionisanja sistema, poput spool direktorijuma i log datoteka. Ovaj direktorijum je specifičan za svaki sistem (radnu stanicu ili server) i mora se realizovati lokalno, a ne kao mrežni fajl sistem.

- **/var/cache/man** - keš u kome se čuvaju formatirane man stranice.
- **/var/lib** - promenljive prateće datoteke raznih programa, uključujući i neke konfiguracione datoteke.
- **/var/local** - Promenljivi podaci koji pripadaju softveru instaliranom u /usr/local/bin direktorijumu (u ovakav softver spadaju npr. programi koje nije instalirao superuser). Lokalno instaliran softver koristi i ostale poddirektorijume /var sistema datoteka, npr. /var/lock.
- **/var/lock** - lock datoteke, odnosno indikatori korišćenja resursa, koje kreiraju razni programi koji koriste neke resurse sistema. Programi koji prate ovu konvenciju mogu pre korišćenja resursa da dobiju informaciju o njegovom zauzeću.
- **/var/log** - log datoteke koje kreiraju razni programi. Primeri ovih datoteka su /var/log/wtmp, u kojoj su zabeležena prijavljivanja (login) i odjavljivanja (logout) sa sistema i /var/log/messages, u koju syslog upisuje poruke kernela i sistemskih programa.
- **/var/mail** - datoteke koje predstavljaju mailbox-ove. U zavisnosti od stepena odstupanja od FHS standarda, neke distribucije čuvaju ove datoteke u direktorijumu /var/spool/mail.
- **/var/run** - datoteke u kojima se čuvaju informacije o sistemu koje su validne do sledećeg podizanja sistema (reboot). Na primer, /var/run/utmp sadrži informacije o korisnicima koji su trenutno prijavljeni na sistem.
- **/var/spool** - spool direktorijumi za poštu (/var/spool/mail) i redovi za štampače (/var/spool/lpd, /var/spool/cups).
- **/var/tmp** - privremene datoteke koje su prevelike da bi bile smeštene u /tmp ili treba da postoje na disku duže nego što bi to bilo moguće u direktorijumu /tmp.

## /proc fajl sistem

/proc fajl sistem omogućava lak pristup strukturama podataka kernela, na osnovu čega se mogu dobiti informacije o sistemu (npr. o procesima, odakle potiče i naziv). /proc nije sistem datoteka u pravom smislu te reči, već jedan kvazi-sistemom datoteka koji sadrži samo simboličku predstavu ovih struktura. Nijedna datoteka sa direktorijuma proc ne zauzima mesto na disku. Sve ove datoteke, koje se mogu videti pomoću standardnih alata za rad sa datotekama, kreira kernel u operativnoj memoriji računara. **Za svaki proces na /proc sistemu datoteka postoji direktorijum sa imenom rednog broja procesa u kom je taj proces opisan.**

- **/proc/1** - direktorijum u kome se nalaze fajlovi sa informacijama o prvom procesu (init).
- **/proc/cpuinfo** - informacije o procesoru
- **/proc/devices** - spisak blok i karakter uređaja koje podržava aktivni kernel
- **/proc/filesystems** - sistemi datoteka za čije korišćenje je kernel konfigurisan
- **/proc/kcore** - slika operativne memorije sistema. Može se iskopirati na drugo mesto u aktivnom stablu, čime se na realnom sistemu datoteka kreira slika operativne memorije (*memory dump*)
- **/proc/kmsg** - poruke koje kernel generiše a koje se dalje prosleđuju *syslog* procesu
- **/proc/meminfo** - informacije o korišćenju operativne i swap memorije
- **/proc/modules** - informacije o aktivnim modulima kernela
- **/proc/net** - statusne informacije mrežnih protokola
- **/proc/uptime** - vreme rada sistema (od poslednjeg podizanja operativnog sistema)
- **/proc/version** - verzija kernela.

## Korisni linkovi

---