



Operativni sistemi 1

Sistemi datoteka

Nemanja Maček

- Pojam datoteke i direktorijuma
- Linkovi (reference)
- Zaštita i deljenje datoteka
- Sistemi datoteka
- Dodela prostora za datoteke
- Upravljanje slobodnim prostorom
- Pouzdanost sistema datoteka
- Značajniji sistemi datoteka (Linux)

- **Datoteka** (engl. *file*) je:
 - za korisnika kolekcija povezanih informacija (logička celina sa značenjem),
 - za OS objekat koji se čuva na sekundarnoj memoriji.
- **Formalno**, datoteka je
 - organizovani skup podataka
 - koji ima svoje ime i
 - koji se prema određenom prostornom rasporedu čuva na sekundarnoj memoriji.
- Datoteku osim sadržaja i imena opisuju i dodatni atributi poput:
 - **tipa datoteke** (pooću tipa OS preliminarno određuje vrstu datoteke povezuje je sa nekom aplikacijom),
 - informacije o **vlasništvu** i **pravima pristupa**, odnosno atributi koji regulišu kontrolu pristupa datoteci (kao što su kontrola čitanja i modifikacije datoteke),
 - **vreme i datum** kreiranja, zadnje izmene, zadnjeg pristupa, itd.

Operacije koje se mogu obaviti nad datotekama

- U operacije koje se mogu obaviti nad datotekama spadaju:
 - otvaranje datoteke,
 - kreiranje datoteke,
 - čitanje podataka iz datoteke u memorijski bafer,
 - upis podataka u datoteku
 - pozicioniranje unutar datoteke (engl. *file seek*),
 - brisanje datoteke (ova operacija uklanja kontrolni blok datoteke iz direktorijuma, a sadržaj datoteke se najčešće fizički ne briše, tako da je moguć povratak sadržaja – engl. *undelete*),
 - odsecanje datoteke (engl. *truncate*), i
 - zatvaranje datoteke.

- OS po pravilu čuva:
 - **globalnu sistemsku tabelu otvorenih datoteka** (engl. *system wide open-file table*), i
 - **posebnu tabelu otvorenih datoteka za svaki proces** (engl. *per process open-file table*).
- Prilikom otvaranja datoteke:
 - u direktorijumskoj strukturi se traži kontrolni blok datoteke,
 - kontrolni blok se upisuje u glavnu sistemsku tabelu,
 - dalji pristup datotekama se realizuje pomoću ovih tabela.
- Kontrolni blok datoteke se ažurira na disku po zatvaranju datoteke.

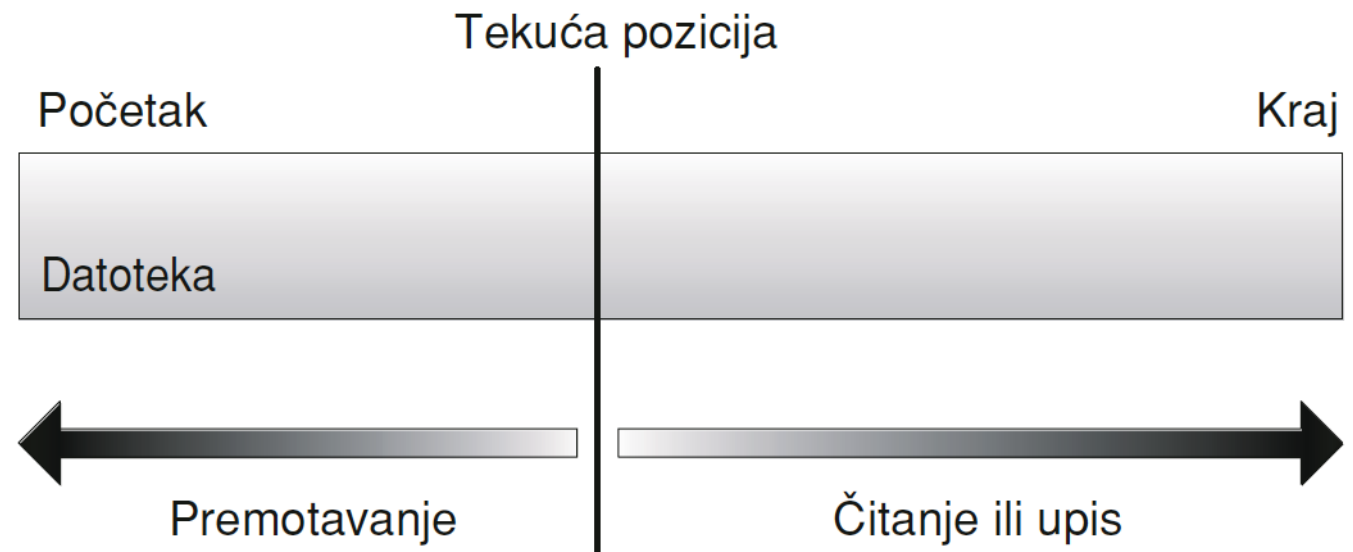
- Datoteke se grubo mogu podeliti na:
 - izvršne datoteke (kompajlirane programe),
 - datoteke sa podacima, koje mogu biti
 - tekstualne ili
 - binarne.
- Na većini OS posle kreiranja datoteka osim imena dobija i **tip**.
 - Pomoću tipa OS može da odredi vrstu datoteke i da je poveže sa nekom aplikacijom.
 - Tip datoteke se može realizovati **ekstenzijom imena datoteke**.
 - Tipično za Windows.
 - Nije tipično za Linux.
- **Ekstenzije:** exe, lib, dll, cpp, py, txt, docx, mp3, zip, 7z, jpeg, png, avi, mp4, ...

Metode pristupa datotekama

- Postoji nekoliko metoda pristupa datotekama:
 - sekvencijalni metod pristupa,
 - direktni metod pristupa,
 - metod pristupa pomoću indeksnih tabela.

Sekvencijalni metod pristupa

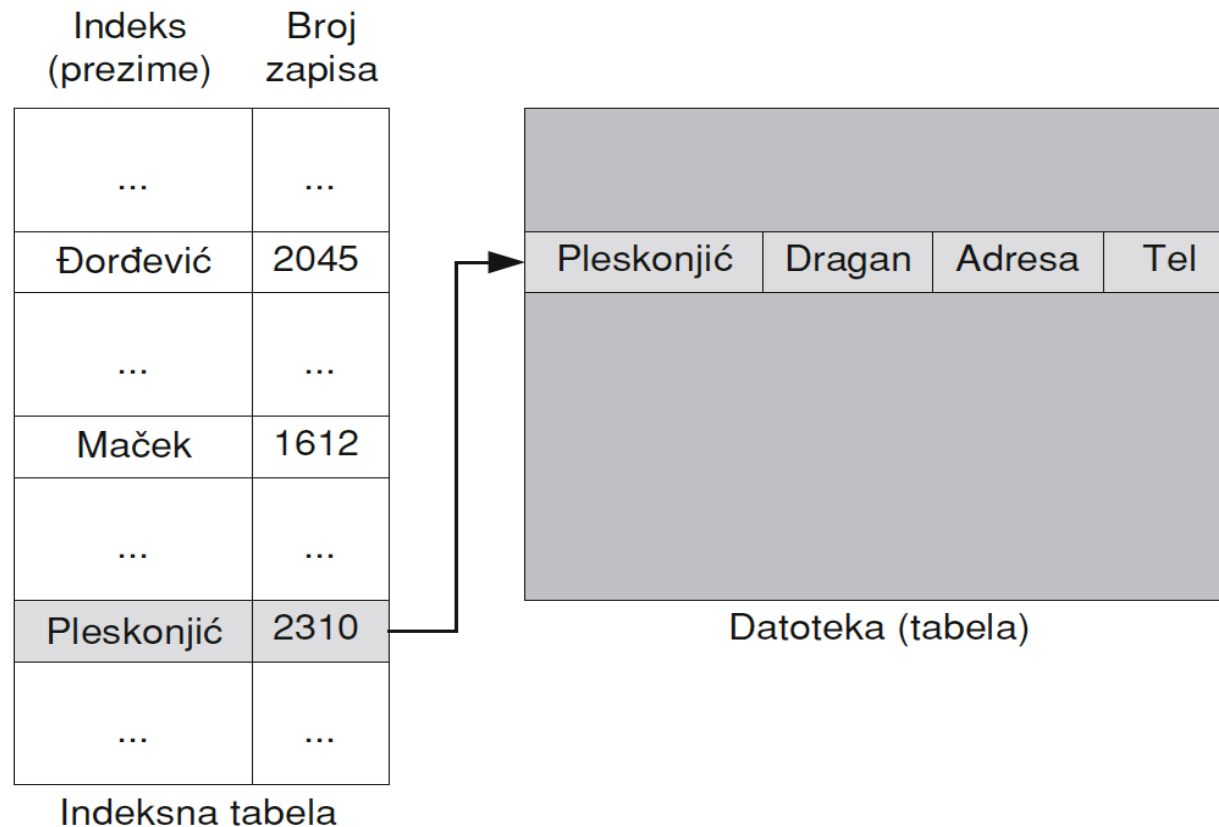
- Informacije se prosleđuju u tačnom redosledu **jedna iza druge**.
- Posle svakog pristupa datoteci **vrednost tekućeg ukazivača se ažurira**.
- Sekvencijalni pristup zahteva da postoji mogućnost da se datoteka **premoti na početak**.
- Moguće operacije:
 - čitanje sledećeg bloka,
 - upis u sledeći blok,
 - pozicioniranje na početak.
- Primer: magnetna traka



- Omogućava pristup bilo kom delu datoteke, tako što se:
 - najpre se odredi njegova pozicija na disku,
 - zatim se pristupi podacima.
- Direktnan pristup omogućava korisniku da pristupi krajnjem bloku datoteke bez čitanja prethodnog sadržaja!
- Moguće operacije:
 - čitanje n-tog bloka,
 - upis u n-ti blok,
 - pozicioniranje na n-ti blok,
 - čitanje sledećeg bloka,
 - čitanje prethodnog bloka.
- Primer: disk

Metod pristupa pomoću indeksnih tabela

- Koristi se npr. za pristup odgovarajućem slogu u bazi podataka.
- Svakoj datoteci pridružena je indeksna datoteka, uređena po nekom kriterijumu.



- Na jednom disku se može smestiti veliki broj datoteka.
- Radi preglednosti i kontrole pristupa grupišu se u **direktorijume**.
- Direktorijum je struktura koja sadrži **kontrolne blokove** svih datoteka koje su u njemu logički smeštene.
- Svi direktorijumi sa svojim datotekama obrazuju sistem datoteka.
- Direktorijumi se moraju implementirati tako da obezbede **efikasno pronalaženje**:
 - postojećih datoteka,
 - prostora za nove datoteke.
 - Tehnike za obezbeđivanje efikasnosti: heš, B+ stabla.

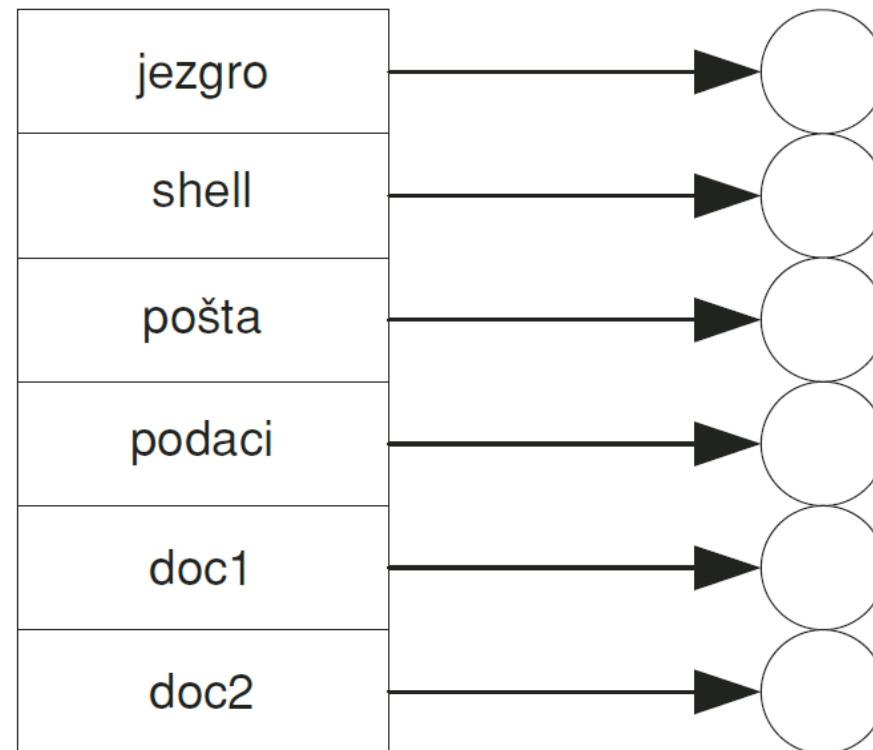
Operacije koje se mogu obaviti nad direktorijumima

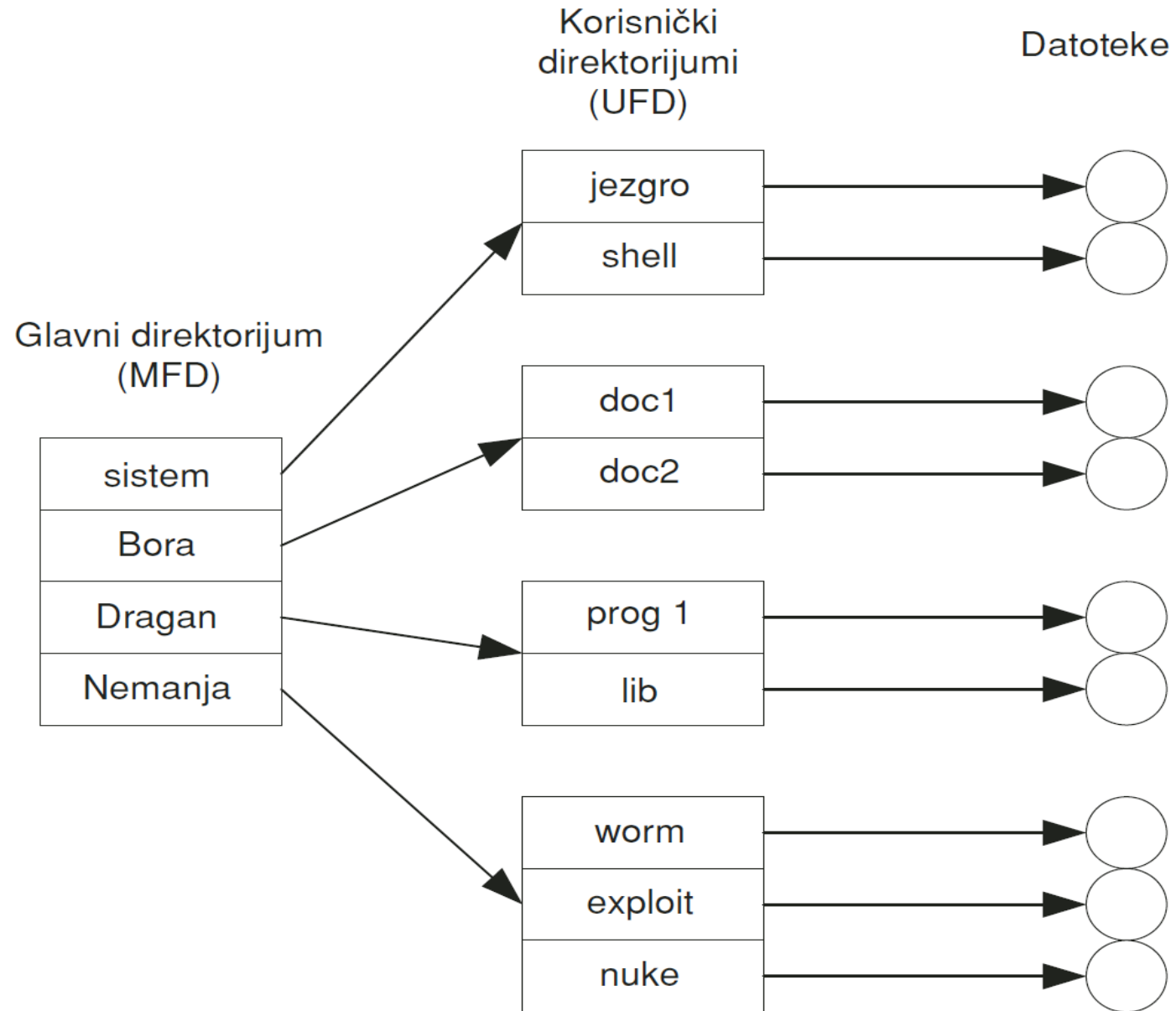
- U operacije koje se mogu obaviti nad direktorijumima spadaju:
 - prikazivanje sadržaja direktorijuma (takozvano “listanje” direktorijuma),
 - pretraživanje direktorijuma (datoteka se u okviru tekućeg direktorijuma pronalazi tako što se upoređuju kontrolni blokovi datoteka sa zadatim imenom),
 - promena imena datoteke,
 - promena imena poddirektorijuma,
 - kreiranje i brisanje datoteka u okviru tekućeg direktorijuma,
 - kreiranje i brisanje poddirektorijuma u okviru tekućeg direktorijuma.

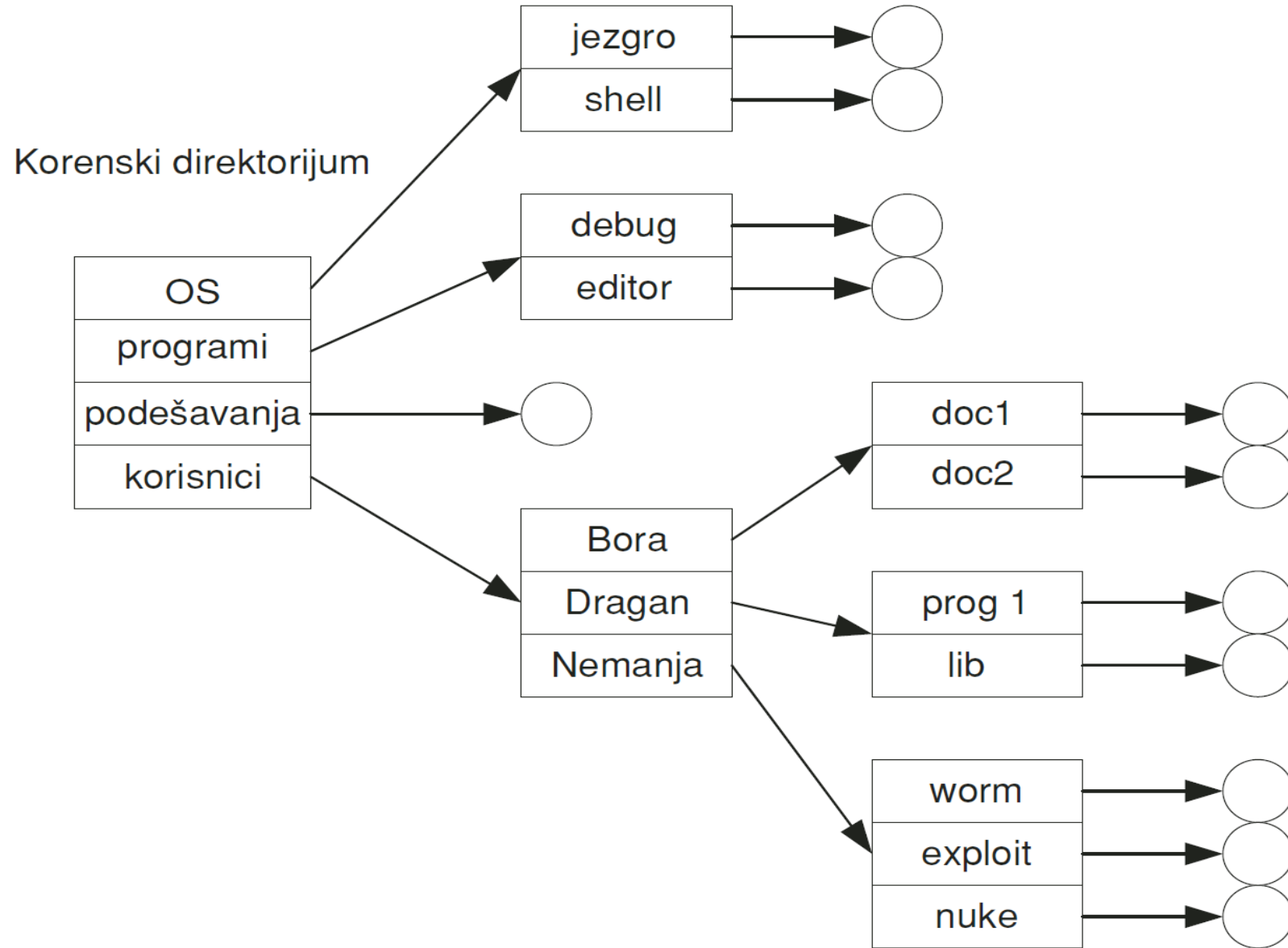
- **Jednonivovska** (v. str. 14).
 - Na sistemu datoteka postoji jedan direktorijum u kome se nalaze sve datoteke.
 - Nepreglednost i problem imenovanja datoteka koje pripadaju različitim korisnicima.
 - Sve datoteke u direktorijumu moraju da imaju različita imena.
- **Dvonivovska** (v. str. 15).
 - Na prvom nivou se nalazi **glavni direktorijum** (engl. *master file directory*, MFD)
 - Na drugom poseban **direktorijum za svakog korisnika** (engl. *user file directory*, UFD).
 - Ne rešava problem logičke organizacije datoteka za pojedinačne korisnike.
- **Višenivovska** (direktorijumsko stablo, v. str. 16).
 - Direktorijum najvišeg nivoa je početni ili **korenski direktorijum** (engl. *root*).
 - U svakom direktorijumu se mogu kreirati datoteke i **poddirektorijumi** (engl. *subdirectory*).
 - Stablo rešava sve probleme **logičke organizacije sa aspekta korisnika**.
 - Svaka datoteka ima svoje **ime i putanju**.
 - Putanja može biti apsolutna ili relativna.

Direktorijum

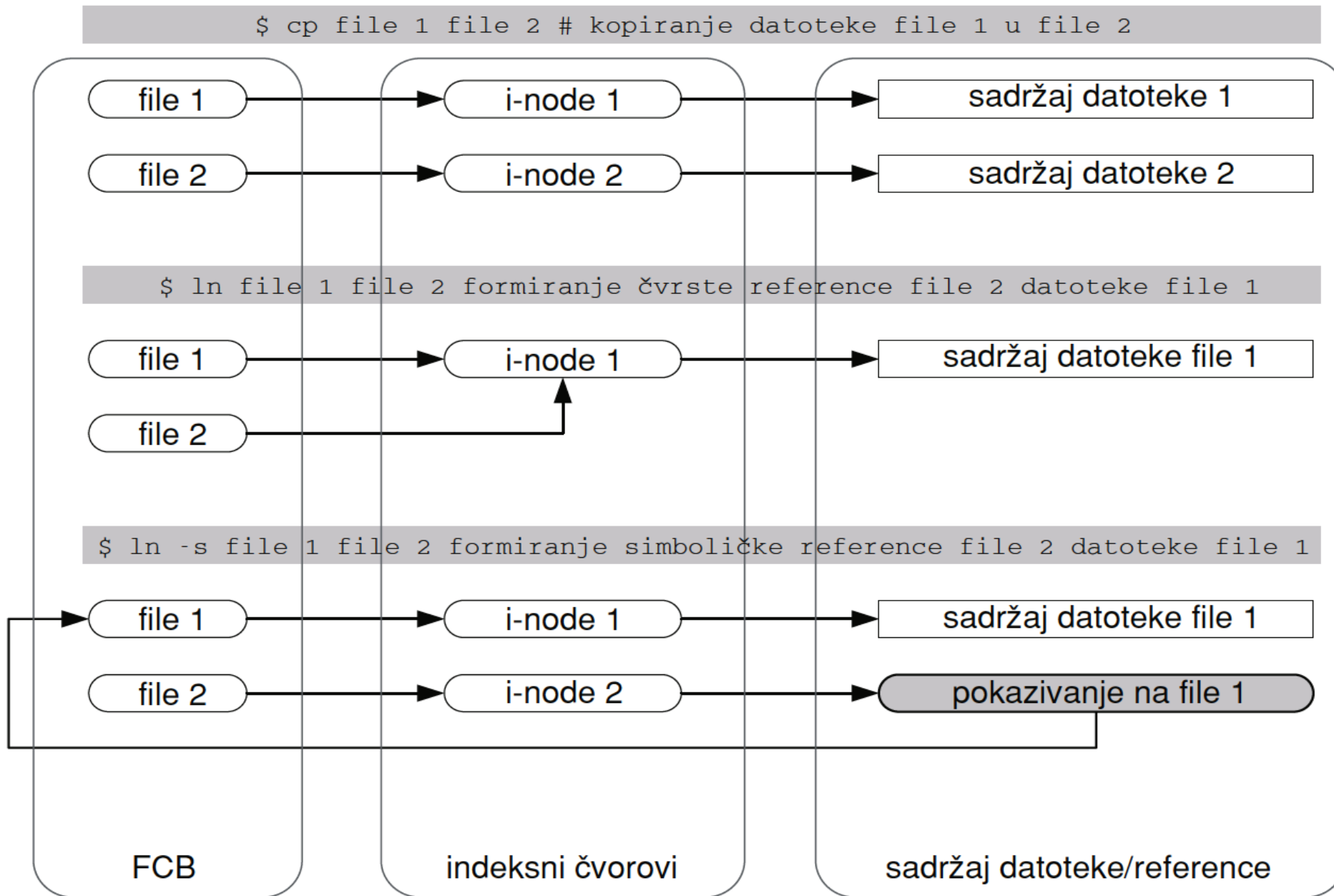
Datoteke







- Datoteke se na većini sistema datoteka mogu “**linkovati**” (referencirati drugim imenima).
- Razlozi za korišćenje linkova:
 - lakše pretraživanje i korišćenje datoteka,
 - ušteda prostora na disku.
- Neki sistemi datoteka podržavaju više vrsta linkova.
 - Linux: **čvrsta referenca** (engl. *hard link*) i **simbolička referenca** (engl. *symbolic link*).
 - Windows: **prečica** (engl. *shortcut*) – uslovno rečeno, ekvivalent simboličkog linka.
- Napomene u vezi hard linkova:
 - Link se mora nalaziti na fizički istom sistemu datoteka.
 - Ne mogu se linkovati datoteke sa mrežnog sistema datoteka (NFS).
 - Ne mogu se linkovati direktorijumi ili nepostojeće datoteke.



- Tri **vlasničke kategorije**:
 - vlasnik (engl. *owner*),
 - grupa (engl. *group*),
 - ostali (engl. *others, public*),
- Prava pristupa za svaku vlasničku kategoriju se eksplicitno dodeljuju svakom objektu prilikom kreiranja (uz redukciju preko promenljive **umask**) a kasnije se mogu promeniti.
- **Potpuni skup prava** za svaku vlasničku kategoriju čine:
 - pravo čitanja (engl. *Read*),
 - pravo upisa (engl. *Write*),
 - pravo izvršavanja (engl. *eXecute*).
- Primer:

```
$ ls -l dat1  
-rw-r--r--  1  u1  g1  509  Mar 10  17:21  dat1
```
- **Liste za kontrolu pristupa (ACLs).**

Kontrola pristupa (Windows NTFS)

- Administrator sistema kreira korisničke grupe i u njih ućlanjuje korisnike i druge grupe.
- Svakom objektu na NTFS sistemu datoteka se prilikom kreiranja dodeljuje grupa informacija koja se tiće kontrole pristupa (engl. **Security Descriptor**).
- Security descriptor sadrži informacije o:
 - dozvolama za pristup resursu (NTFS dozvole),
 - vlasniku resursa.
- NTFS dozvole se dodeljuju ili oduzimaju korisnicima i grupama.
 - Svaki korisnik ili grupa predstavlja jedan **Access Control Entry** (ACE).
 - Skup svih dozvola predstavlja tabelu **Access Control List** (ACL).
 - Grupe i korisnici su u tabeli predstavljeni numeričkim vrednostima **SID** (engl. *Security Identifiers*) koje ih jednoznačno identifikuju.
 - Prava pristupa data grupi prenose se na grupe i korisnike koji su u tu grupu ućlanjeni.

Kontrola pristupa (Windows NTFS)

- Postoje dve vrste dozvola koje se mogu dodeliti: **standardne** i **specijalne**.
- Svaka **standardna dozvola** je sistemski predefinisani skup atomarnih dozvola.
- **Atomarne dozvole** omogućavaju korisniku da nad objektima izvrši određenu akciju.
- U standardne dozvole spadaju:
 - Read,
 - Write,
 - Read and Execute,
 - Modify (predstavlja uniju standardnih dozvola Read and Execute i Write i atomarne dozvole za brisanje objekta),
 - Full Control,
 - List Folder Contents (dozvola koja se može dati samo nad direktorijumom, koja omogućava korisniku da vidi sadržaj direktorijuma).

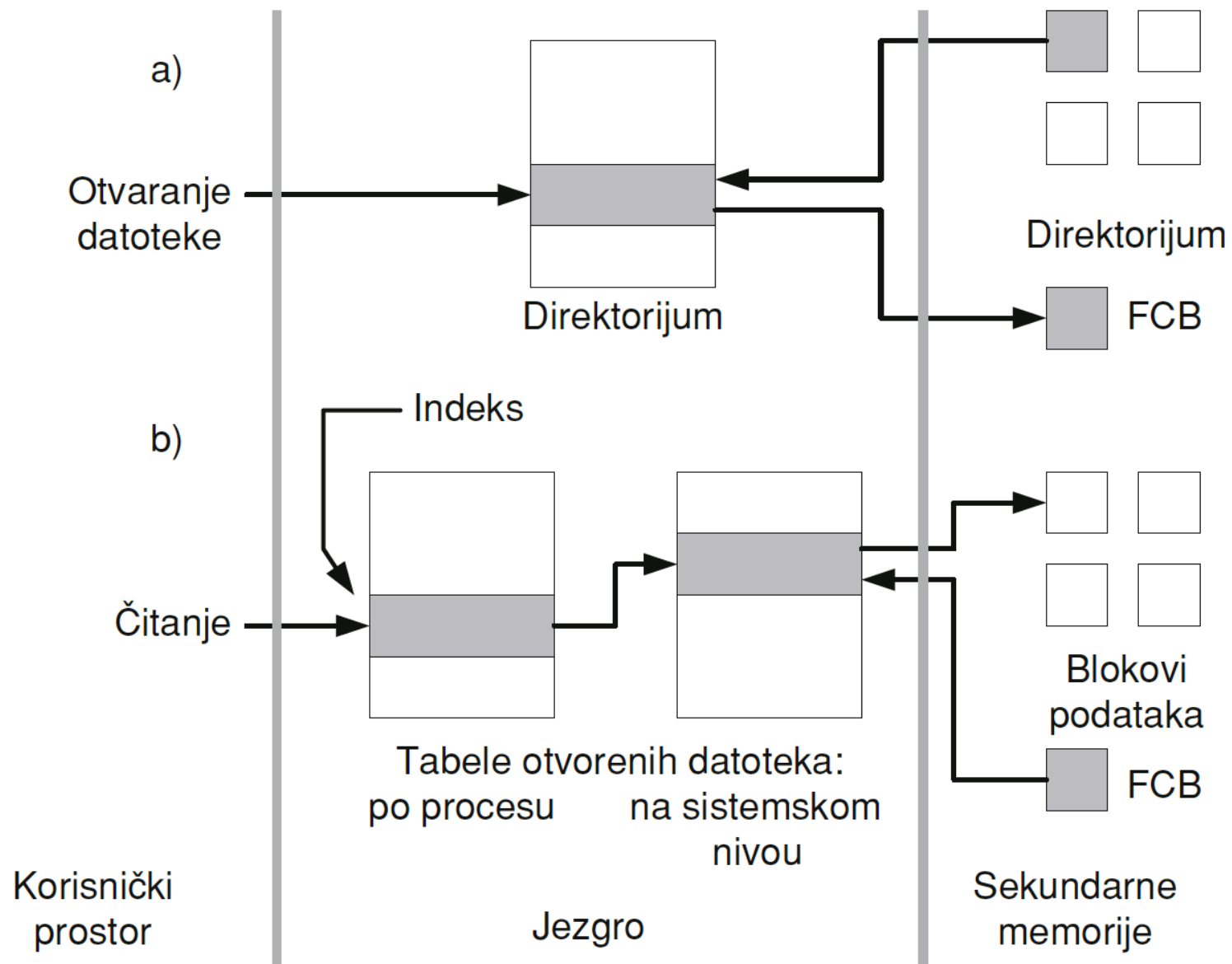
- Diskovi se mogu podeliti na više delova.
 - Delovi diska se nazivaju **particije** (engl. *partitions*) ili **volumeni** (engl. *volumes*).
 - Da bi deo diska mogao da se iskoristi za skladištenje podataka, potrebno je na njemu kreirati sistem datoteka.
 - Svaki deo diska predstavlja granice u čijim se okvirima može kreirati sistem datoteka.
- **Sistem datoteka** je skup metoda i struktura podataka koje operativni sistem koristi za čuvanje datoteka.
- Sistem datoteka čine:
 - **zaglavlje** (podaci neophodni za funkcionisanje sistema datoteka),
 - **meta podaci** (strukture za organizaciju podataka na medijumu),
 - **podaci** (datoteke i direktorijumi).
- Zaglavlje i meta-podaci čine premašenje sistema, ali bez njih sistem datoteka ne može da funkcioniše.

- Prilikom realizacije sistema datoteka, definiše se:
 - **Logička struktura sistema datoteka.**
 - Način predstavljanja sistema datoteka korisniku.
 - Definicija datoteka, direktorijuma, njihovih atributa i operacija dozvoljenim nad njima, odnosno sve što predstavlja logičku sliku sistema datoteka.
 - **Fizička struktura sistema datoteka.**
 - Strukture podataka na disku koje služe za skladištenje podataka.
 - Primer: **blokovi** (na Linux-u) i **klasteri** (engl. *clusters*) na FAT sistemu datoteka.
 - **Preslikavanje logičke strukture sistema datoteka u fizičku.**
 - Uspostavljanje veze između sadržaja konkretne datoteke i struktura na disku.
 - Primer: sadržaj datoteke mydoc.txt koja se nalazi na direktorijumu c:\1 nalazi u blokovima 15,16 i 25.

Strukture podataka neophodne za realizaciju sistema datoteka

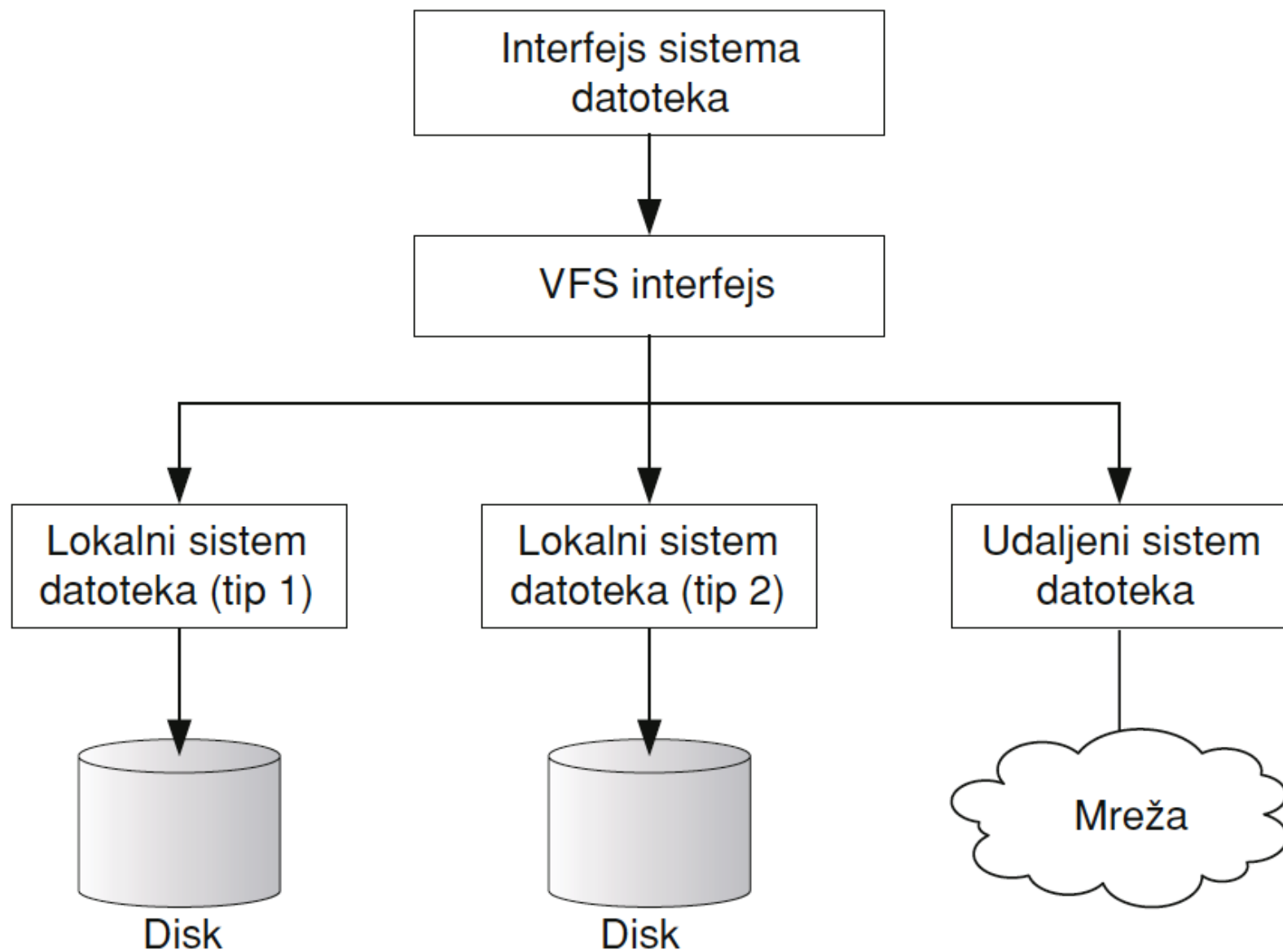
- **BCB** (engl. *boot control block*).
 - Sadrži informacije koje se potrebne za otpočinjanje procesa podizanja OS.
 - Obično prvi blok sistema datoteka.
- **Kontroni blok partcije** (engl. *partition control block*, PCB).
 - Sadrži informacije o sistemu datoteka.
 - Primer: veličina bloka, ukupni broj blokova, broj slobodnih blokova, itd.
- **Kontrolne strukture za alokaciju datoteka**.
 - Određuju konkretan sadržaj datoteke (blokove u kojima je smešten sadržaj).
 - Primer: i-node tabela (UNIX), FAT tabela (FAT) i MFT (NTFS).
- **Direktorijumske strukture** koje sadrže kontrolne blokove datoteka
- **Kontrolni blokovi datoteka** (engl. *file control block*, FCB).
 - Sadrži attribute datoteka i ukazivače na alokaciju datoteke.
 - Primer (Linux): fileinfo strukture u direktorijumu + i-node.

- U operativnoj memoriji se definišu strukture koje ubrzavaju rad sistema datoteka. U memorijske strukture podataka spadaju:
 - tabela otvorenih datoteka **na sistemskom nivou** (engl. *system-wide open-file table*), koja sadži kontrolne blokove za svaku otvorenu datoteku.
 - tabela otvorenih datoteka **po procesu** (engl. *per-proces open-file table*), koja sadži ukazivač na glavnu tabelu i druge informacije, kao što je tekući ukazivač na datoteku.
- Prilikom otvaranja datoteke kontrolni blok se iz direktorijuma upisuje u glavnu memorijsku tabelu.
- Pri svakom pristupu datoteci iz tabele procesa se preko glavne tabele otvorenih blokova pronalaze blokovi datoteke na disku, a zatim se njima pristupa.



- Svaki sistem datoteka pre korišćenja mora da se **aktivira**.
 - Kod nekih OS ovaj postupak je transparentan i ne zahteva učestvovanje korisnika.
 - Drugi OS zahtevaju da korisnik sa odgovarajućim privilegijama:
 - eksplicitno odredi koji će sistemi datoteka biti **automatski aktivirani** a koji ne,
 - koje sisteme datoteka **ostali korisnici naknadno mogu aktivirati**.
- Dve osnovne metode aktiviranja sistema datoteka su:
 - **mapiranje na logičke diskove** (koreni direktorijum sistema datoteka poklapa se sa korenim direktorijumom logičkog diska) i
 - **montiranje na drugi direktorijum** (koreni direktorijum sistema datoteka poklapa se sa direktorijumom na koji je montiran).

- Većina operativnih sistem podržava rad sa različitim sistemima datoteka.
- Zato se uvodi novi logički koncept – **virtuelni sistem datoteka** (engl. *virtual filesystem*, VFS).
- VFS omogućava korisniku da na isti način pristupa svim datotekama, bez obzira kom sistemu datoteka pripadaju.
 - Korisnik se putem sistemskih poziva obraća virtuelnom sistemu datoteka.
 - VFS obrađuje zahrev i upućuje poziv za rutine odgovarajućeg sistema datoteka.

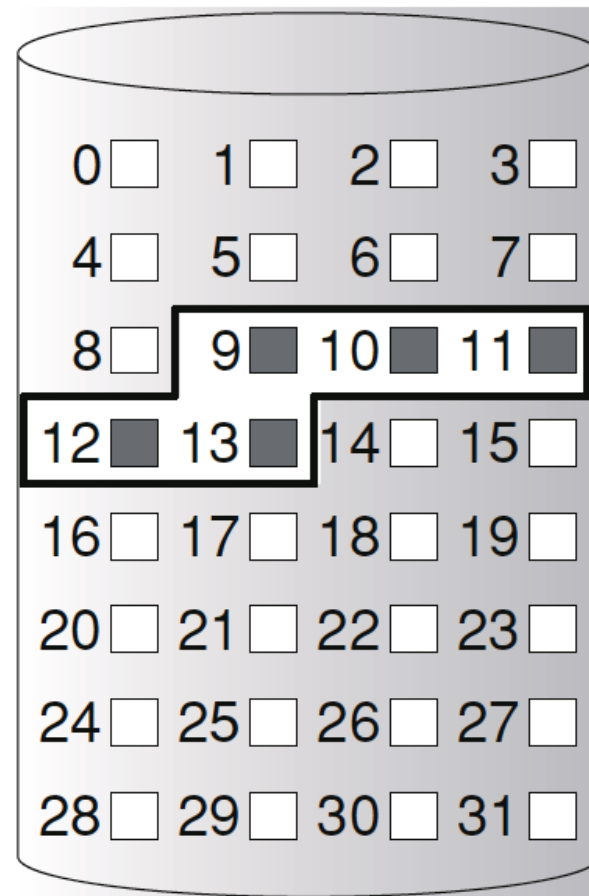


- Blokovi diska se datotekama mogu dodeliti korišćenjem sledećih metoda alokacije:
 - dodela kontinualnog prostora (v. str. 31),
 - vezivanje blokova,
 - mapa datoteka,
 - metoda indeksnih blokova.

Direktorijum

Datoteka	Početak	Dužina
doc1	2	4
doc2	9	5
image	17	2
worm	25	4

Datoteka doc2 smeštena je u
blokove 9, 10, 11, 12 i 13.

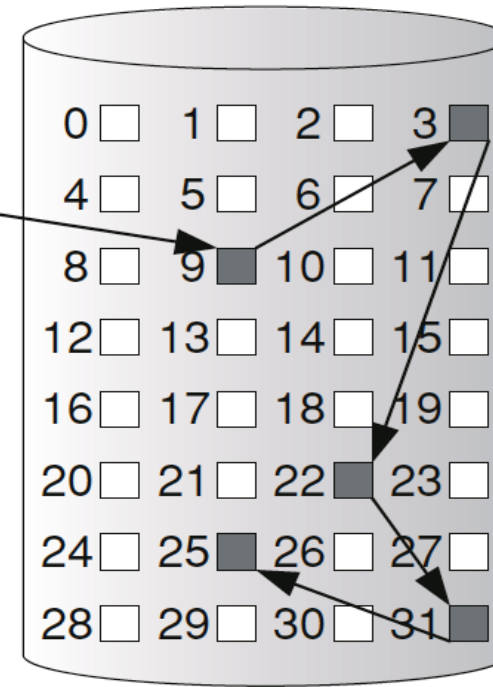


- Blokovi dodeljeni datoteci mogu biti proizvoljno razbacani po celom disku.
- Nekoliko bajtova svakog bloka odvoji se za smeštanje pokazivača na sledeći blok
- Poslednji blok sadrži NULL vrednost.
- U kontrolni blok datoteke smešta se samo informacija o početku i kraju datoteke, odnosno broj prvog i poslednjeg bloka datoteke.
- Nedostaci:
 - **sekvencijalan pristup datotekama** (pre čitanja nekog bloka datoteke moraju pročitati svi prethodni blokovi),
 - **mala pouzdanost** (defekt jednog bloka učiniće nepristupačnim ostatak datoteke).

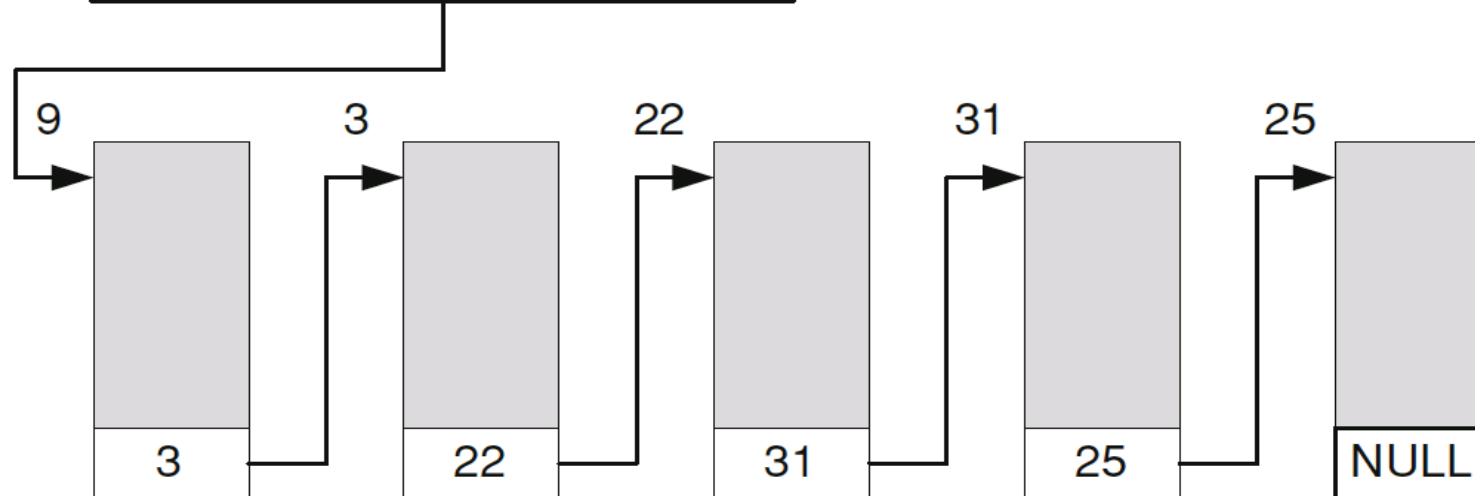
Direktorijum

Datoteka	Početak	Kraj
doc1	2	4
doc2	9	25
image	17	2
worm	24	30

Datoteka doc2 smeštena je u blokove 9, 3, 22, 31 i 25.



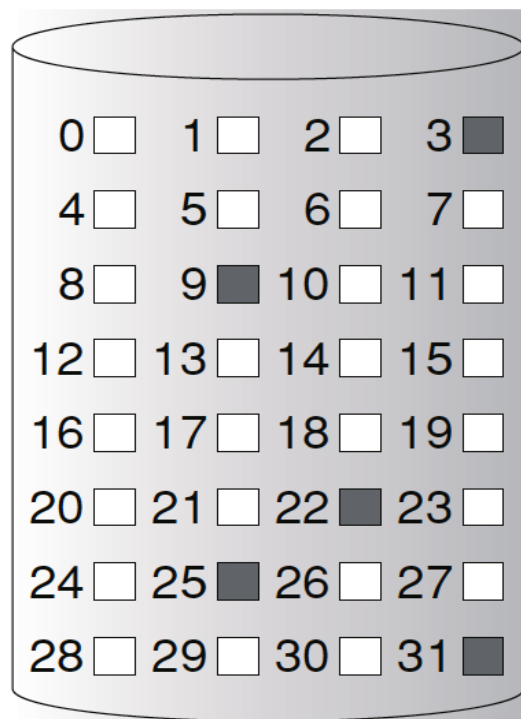
doc2	9	25
------	---	----



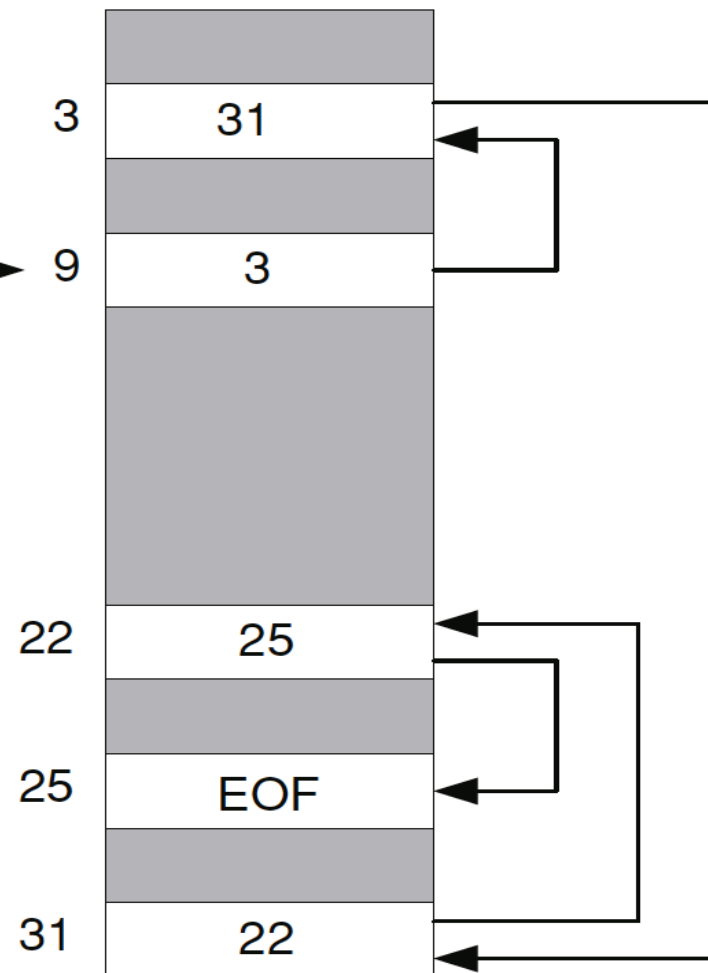
- Uvodi se **centralizovana tabela** u kojoj će biti opisan prostorni raspored svih datoteka.
- Primer: **FAT**.
 - Sistem datoteka se prilikom kreiranja deli na **klaster** (engl. *cluster*).
 - U okvirima jednog klastera datoteci je dodeljen **kontinualan prostor**.
 - Datoteci se može dodeliti ceo broj klastera u proizvoljnom rasporedu.
 - Jedan klaster ne mogu koristiti dve datoteke.
 - Neiskorišćen deo klastera zove se *slack* i predstavlja gubitak.
 - Svi klasteri jedne datoteke čine njenu **povezanu listu**.
 - Povezane liste svih datoteka čuvaju se u FAT tabeli na sledeći način.
 - Svaki klaster diska opisan je jednim zapisom u FAT tabeli u kome je naveden broj sledećeg klastera dodeljenog datoteci.
 - Kraj datoteke je klaster u čijem slogu u FAT tabeli stoji oznaka za kraj datoteke EOF.
 - Kontrolni blok datoteke sadrži samo broj početnog klastera
 - Svi ostali klasteri određuju na osnovu FAT tabele.

Direktorijum

Datoteka	Početak
doc1	2
doc2	9
image	17
worm	24



Mapa svih datoteka

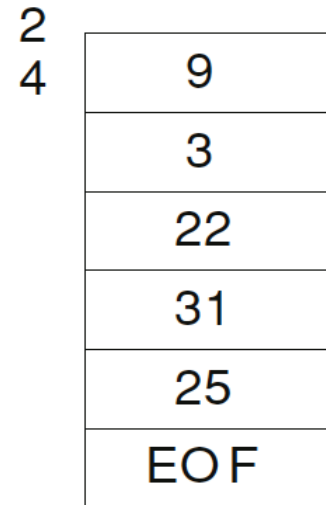
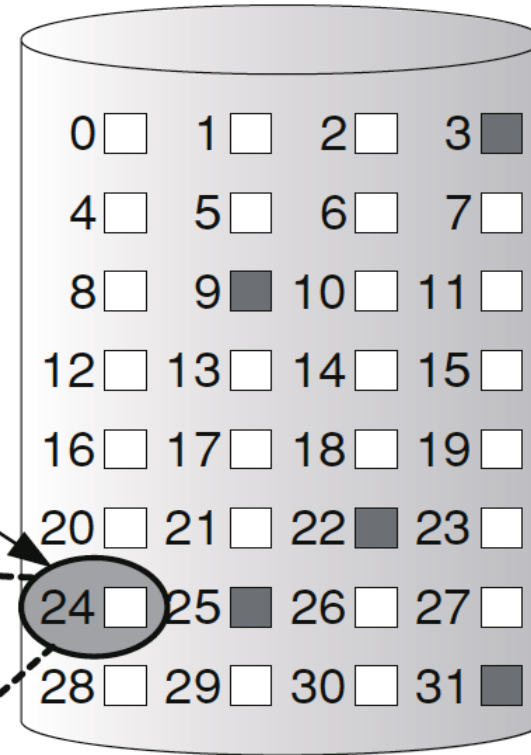


Datoteka doc2 smeštena je u blokove 9, 3, 31, 22 i 25.

- Svakoj datoteci se dodeljuje **indeksni blok**.
- Indeksni blok sadrži informacije o blokovima koji su dodeljeni datoteci.
- Svi ukazivači na blokove podataka koji pripadaju datoteci smešteni na jednom mestu.
- Kraj datoteke označen je slogom EOF u indeksnom bloku.
- U kontrolnom bloku datoteke se nalazi adresa odgovarajućeg indeksnog bloka.
- Prednosti:
 - nema eksterne fragmentacije,
 - pristup datoteci je direktan.

Direktorijum

Datoteka	Indeksni blok
doc1	4
doc2	24
image	12
worm	30

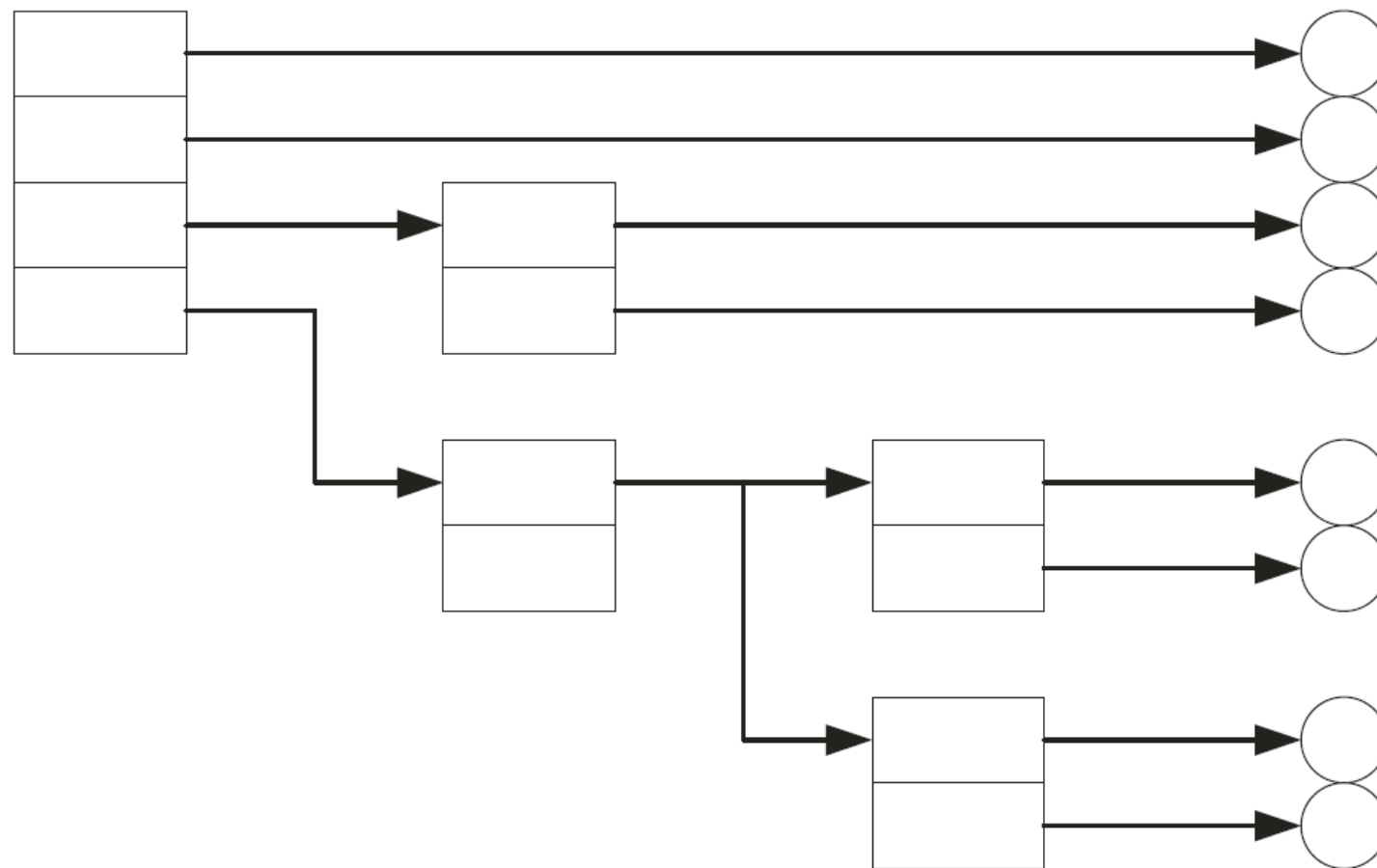


Datoteka doc2 smeštena je u
blokove 9, 3, 22, 31 i 25.

Blok 24 je indeksni blok
datoteke doc2

- Potencijalan problem je dodela prostora velikim datotekama čiji se prostorni raspored ne može opisati jednim indeksnim blokom.
- Ovaj problem se može rešiti na dva načina: vezivanjem indeksnih blokova i višenivovskim adresiranjem.
- **Vezivanje indeksnih blokova.**
 - Za opis prostornog rasporeda datoteke koristi se veći broj indeksnih blokova.
 - Poslednji zapis u indeksnom bloku predstavlja ukazivač na sledeći indeksni blok.
- **Višenivovsko adresiranje** (v. str. 39).
 - Kontrolni blok datoteke ukazuje na jedan indeksni blok prvog nivoa.
 - Indeksni blok prvog nivoa koji ukazuje na n indeksnih blokova drugog nivoa.
 - Indeksni blokovi drugog nivoa adresiraju blokove datoteke.
 - Broj nivoa se po potrebi može povećati.

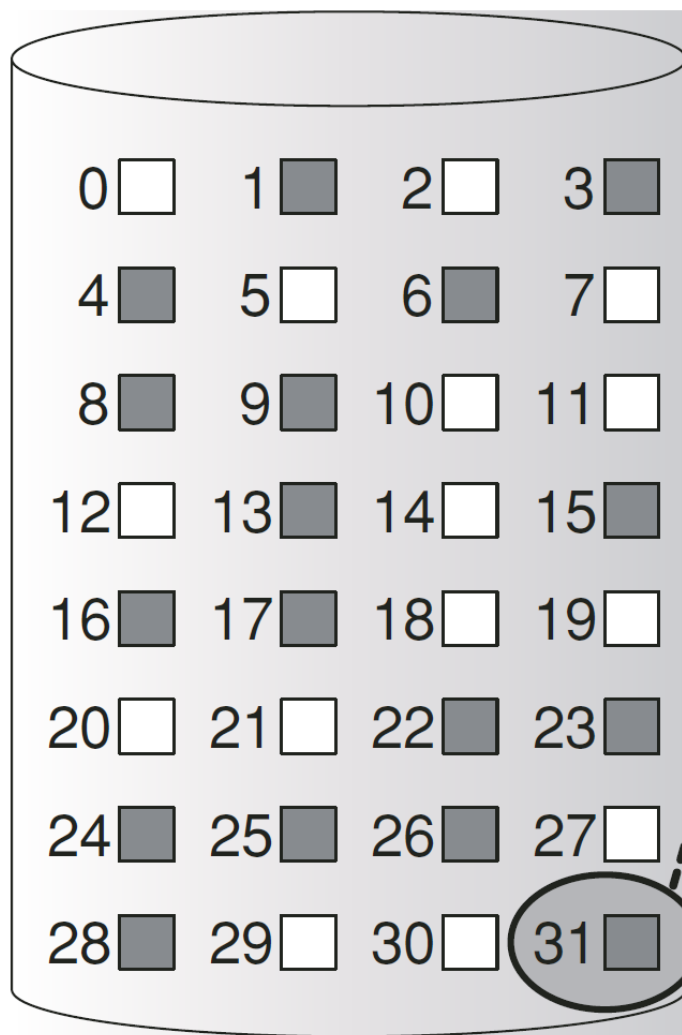
Indeksni blok
prvog nivoa Indeksni blokovi
drugog nivoa Indeksni blokovi
trećeg nivoa Blokovi
podataka



Upravljanje slobodnim prostorom

- Slobodni prostor na disku se dodeljuje datotekama prilikom kreiranja i povećavnja datoteka.
- Upravljanje slobodnim prostorom značajno utiče na performanse sistema datoteka.
- Postoje dve osnovne metode za upravljanje slobodnim prostorom:
 - **mape bitova,**
 - **povezane liste.**

- Za svi blok diska uvodi se bit koji vrednostima 0 i 1 opisuje da li je taj blok **slobodan** ili **zauzet**.
- Mapa bitova se formira kao **struktura podataka na disku**.
- Mapa je **niz svih prethodno pomenutih bitova** i kao takva opisuje koji su blokovi slobodni u sistemu datoteka.
- Mape bitova:
 - zahtevaju formiranje posebne meta strukture na disku,
 - uz pomoć memorijskih struktura za keširanje relativno brzo pretražuju,
 - zbog keširanja mora se voditi računa o koherenciji između mape na disku i mape u memoriji.

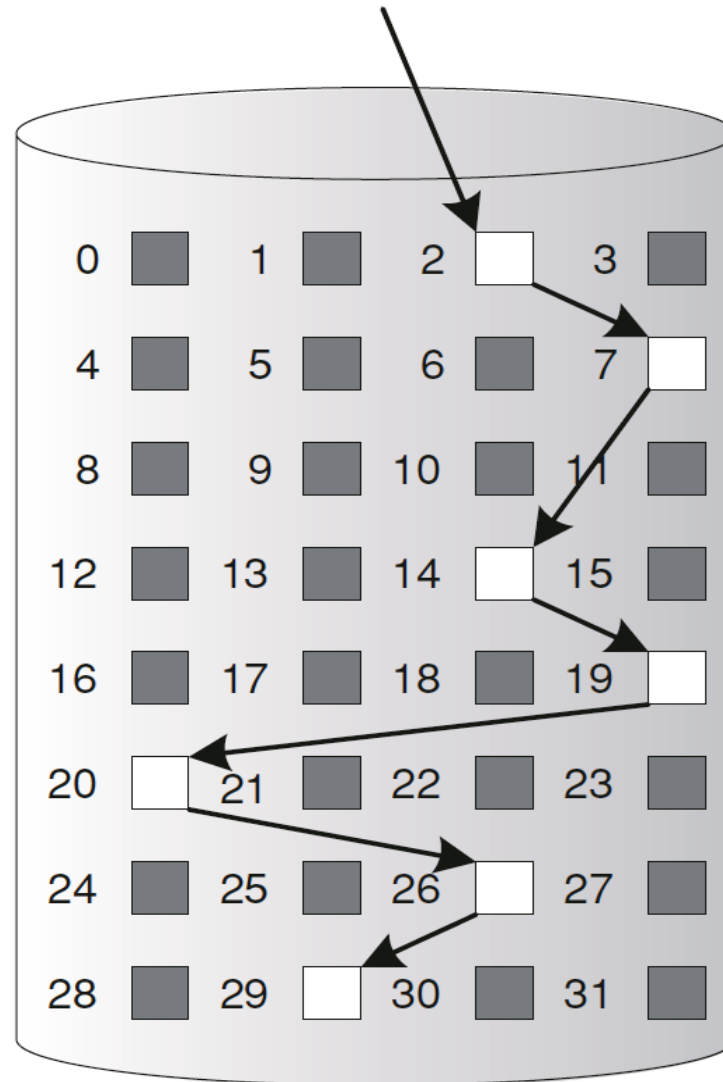


Mapa bitova:

0	1	0	1	1	0	1	0
1	1	0	0	0	1	0	1
1	1	0	0	0	0	1	1
1	1	1	0	1	0	0	1

- Liste se formiraju na sledeći način:
 - početak liste ukazuje na prvi slobodni blok,
 - svaki slobodni blok ukazuje na sledeći.
- Metoda ne koristi dopunske strukture za vođenje evidencije (osim pokazivača).
- Performanse su znatno manje u odnosu na mape bitova ali se mogu uvećati:
 - **grupisanjem ukazivača** na slobodne blokove,
 - uvođenjem informacije o **broju uzastopnih slobodnih blokova**.

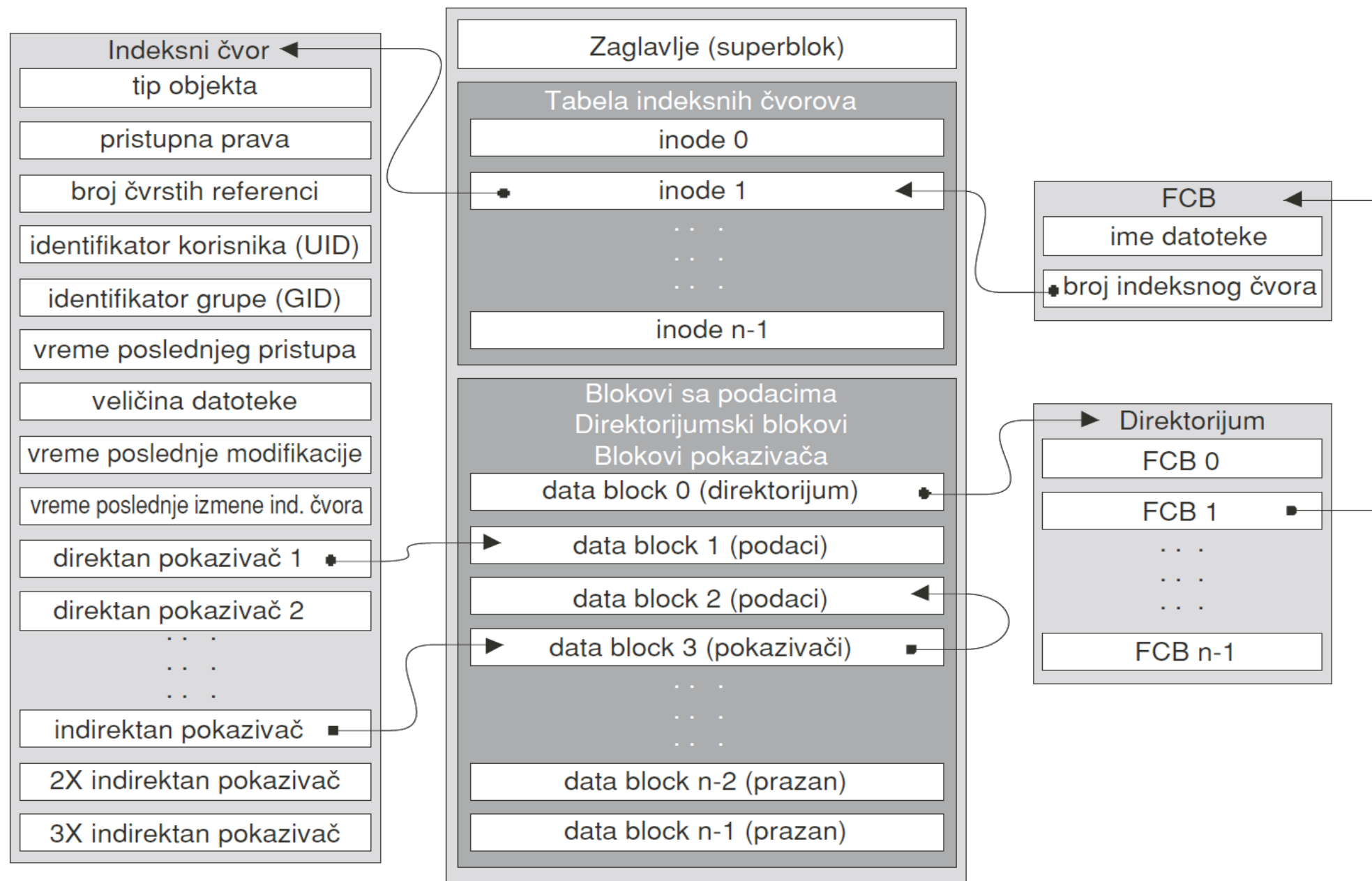
Početak liste
slobodnih blokova



- Oštećenja sistema datoteka mogu biti fizička i logička.
- Rešenja za povećanje pouzdanosti su:
 - vođenje dnevnika transakcija (praćenje svih promena u sistemu datoteka i praćenje promena u meta strukturama), i
 - arhiviranje i rezervne kopije podataka.

- U objekte Linux sistema datoteka spadaju:
 - **regularne datoteke**,
 - **direktorijumi** (mogu se posmatrati kao specijalne datoteke koje sadrže objekte sistema datoteka, uključujući i poddirektorijume),
 - **hard linkovi** (alternativna imena datoteka),
 - **simbolički linkovi** (prečice, odnosno datoteke čiji su sadržaji putanje i imena objekata na koji upućuju),
 - **blok i karakter specijalne datoteke** (opisuju uređaje, odnosno drajvere u kernelu),
 - imenovani *pipeline*.

- Osnovna struktura svih domaćih Linux sistema datoteka je slična.
- Linux sistem datoteka čine:
 - **zaglavlje** (engl. *superblock*),
 - **tabela indeksnih čvorova** (engl. *i-node table*),
 - **blokovi sa podacima** (engl. *data blocks*),
 - **direktorijumski blokovi** (engl. *directory blocks*),
 - **blokovi indirektnih pokazivača** (*indirection block*).



1. B. Đorđević, D. Pleskonjić, N. Maček (2005): Operativni sistemi: teorija, praksa i rešeni zadaci. Mikro knjiga, Beograd.
2. R. Popović, I. Branović, M. Šarac (2011): Operativni sistemi. Univerzitet Singidunum, Beograd.

Pitanja su dobrodošla.