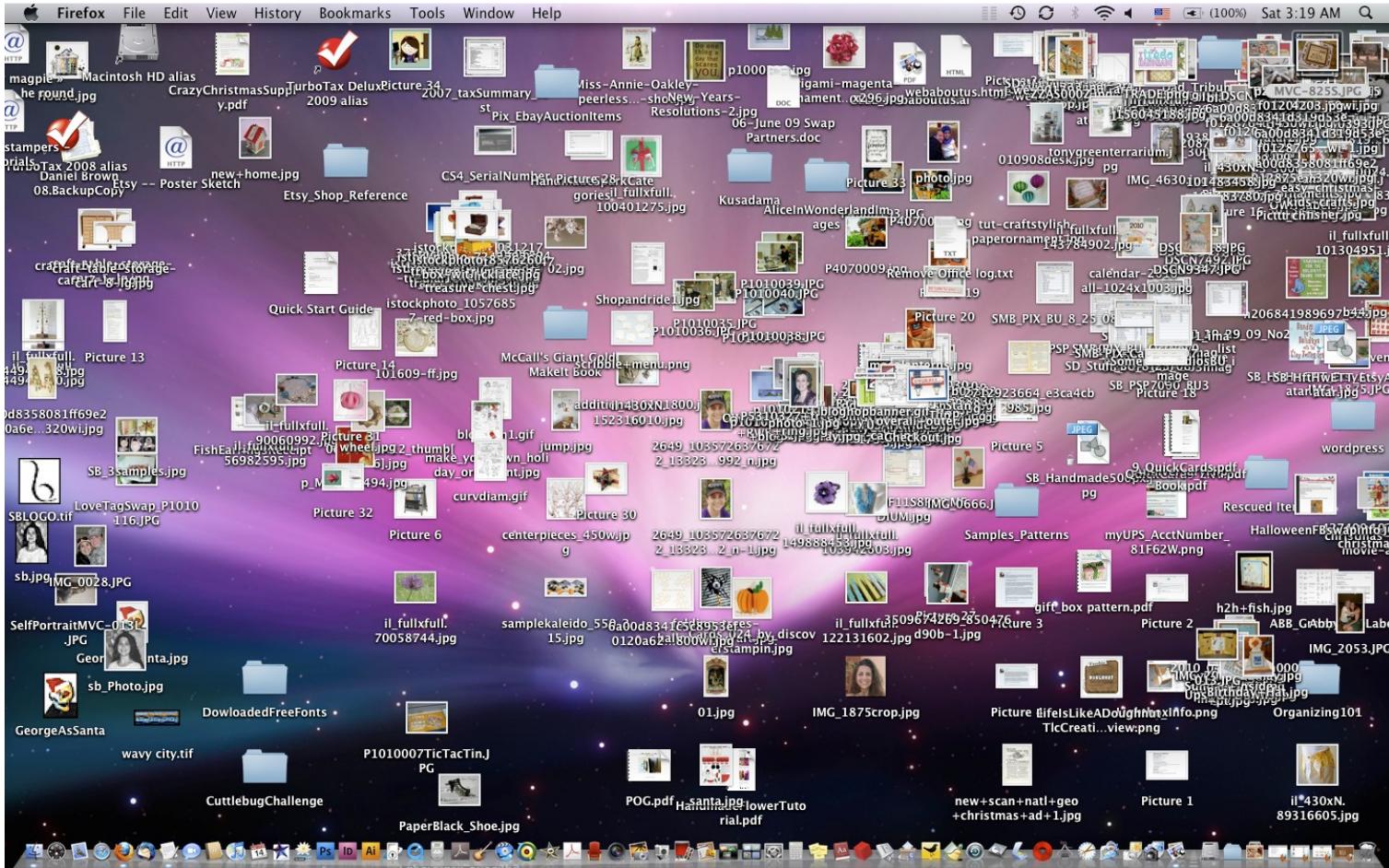


# OPERATIVNI SISTEMI

Slajdovi su kreirani na osnovu knjige "Operativni sistemi, principi unutrašnje organizacije i dizajna, 7. izdanje", William Stallings, CET, Beograd, 2013.

Upravljanje fajlovima

# Upravljanje fajlovima



# Fajlovi

- Resursi koji sadrže grupe podataka
  - Programski kod aplikacije
  - Ulazni/izlazni podaci koji se koriste pri izvršavanju
- Sistem za upravljanje fajlovima
  - Sistemska softver zadužen za rad sa fajlovima
  - Obezbeđuje apstrakciju resursa uskladištenih u sekundarnoj memoriji
  - Može se smatrati delom OS

# Fajl sistem

- Skup metoda i struktura podataka koje OS koristi za upravljanje fajlovima
- Definiše kako su fajlovi na disku organizovani
- Postoje različiti tipovi fajl sistema
  - ext2, ext3, ext4
  - NFS
  - FAT
  - NTFS
  - ...

# Poželjne osobine fajlova

- Dugoročno postojanje
  - Fajlovi se skladište u sekundarnoj memoriji i ne nestaju kada se aplikacija završi ili korisnik odjavi
- Deljivost između procesa
  - Različiti procesi (ako imaju prava) mogu da koriste iste fajlove
- Struktura
  - Interna struktura fajla je takva da aplikacijama bude pogodna za korišćenje
  - Međusobno se fajlovi organizuju u složene strukture
    - Npr. hijerarhijska struktura

# Operacije nad fajlovima

- Kreiranje
- Brisanje
- Otvaranje
- Zatvaranje
- Čitanje
- Upisivanje

# Organizacija podataka

## □ Polje

- Osnovni element podataka
- Sadrži pojedinačnu vrednost
- Definisani dužinom i vrstom podatka

## □ Zapis

- Skup logički povezanih polja
- Aplikacija može sa zapisom da postupa kao sa jedinicom

# Organizacija podataka

## □ Fajl

- Skup logički povezanih zapisa
- Tretira se kao jedan entitet
- Adresira se po imenu
- Kontrola pristupa je najčešće na nivou fajla

## □ Baza podataka

- Strukturiran opis podataka aplikacije
- Skladišti se kao jedan ili više povezanih fajlova
- Obično upravljane posebnim softverom koji nije deo OS
  - Sistem za upravljanje bazom podataka

# Sistem za upravljanje fajlovima

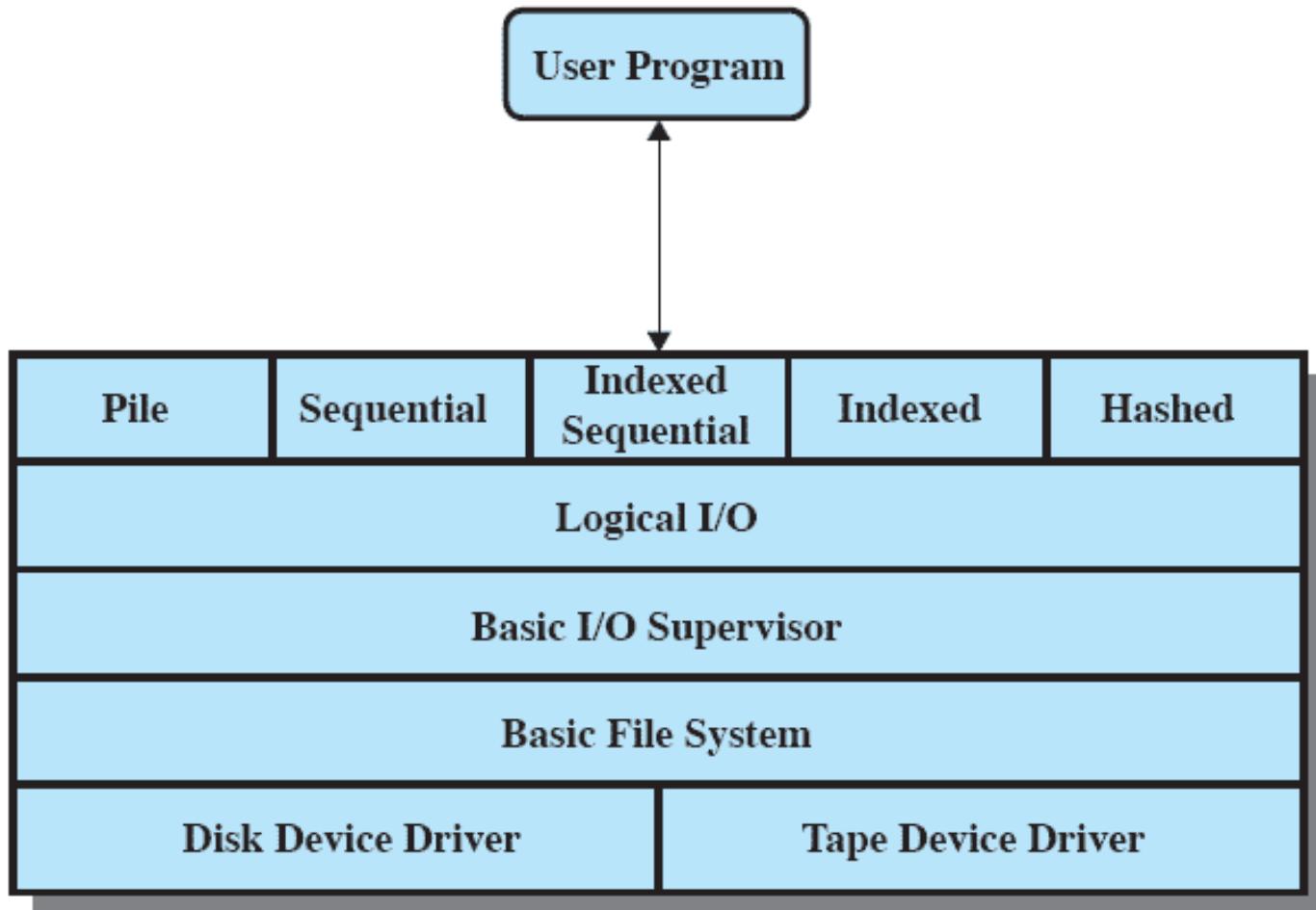
- Obezbeđuje korisnicima i aplikacijama usluge vezane sa upotrebom fajlova



# Sistem za upravljanje fajlovima

- Jedini način da se pristupi fajlovima je kroz sistem za upravljanje fajlovima
- Programer ne mora da razmišlja o načinu upravljanja fajlovima na niskom nivou
- Ovim je upravljanje fajlovima uniformno za sve aplikacije

# Softverske komponente za upravljanje fajlovima



# Drajveri uređaja

- Direktno upravljanje uređajem na najnižem nivou
- Započinju U/I operaciju na uređaju
- Obrađuju rezultat kada uređaj završi operaciju
- Deo OS

# Osnovni fajl sistem

- Još uvek je to nivo fizičkog U/I
- Bavi se blokovima podataka koji se razmenjuju sa uređajem
  - Smeštanje blokova na uređaju
  - Baferovanje blokova u glavnoj memoriji
- Deo OS

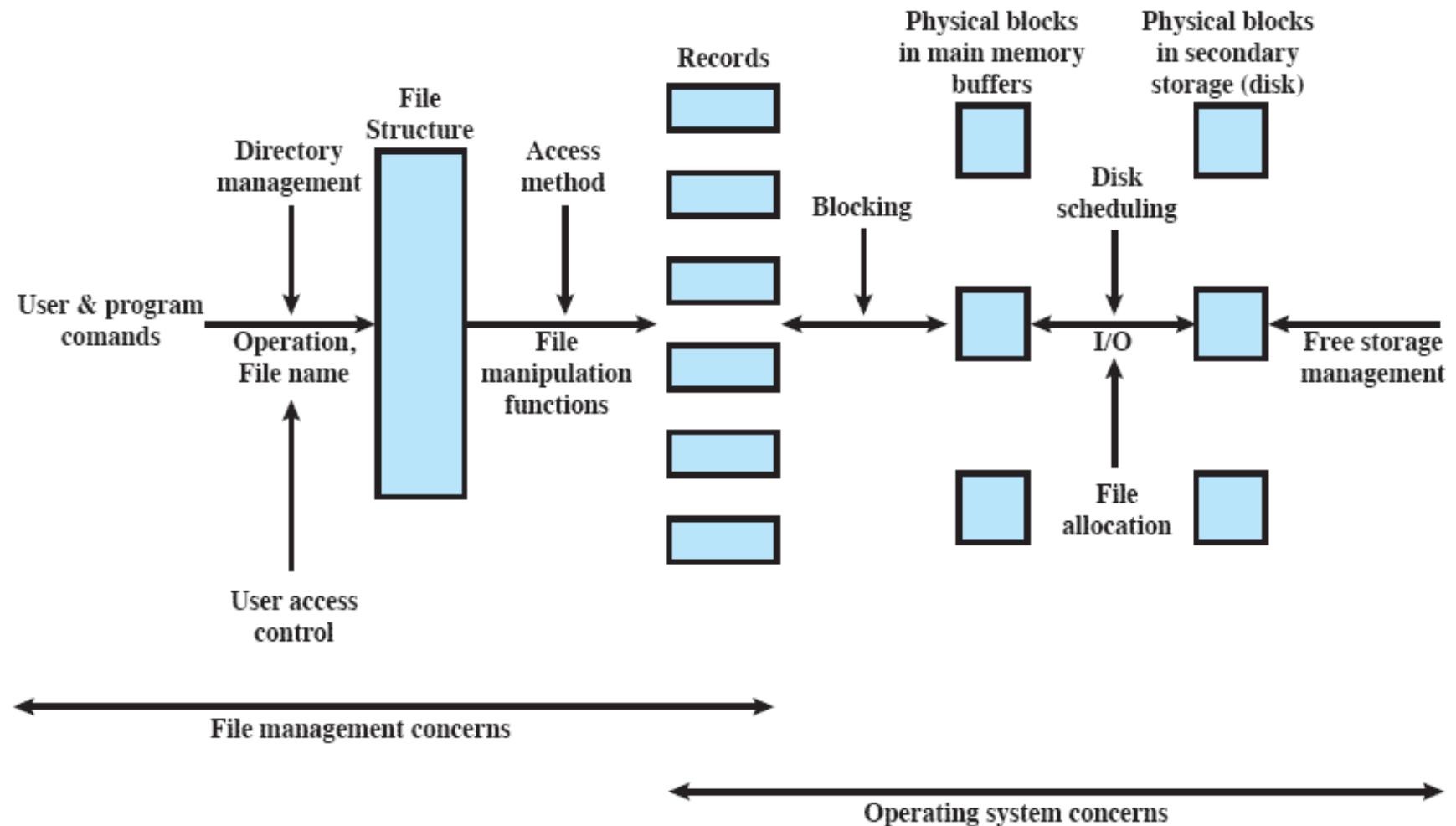
# Osnovni U/I supervizor

- Generalna organizacija U/I zahteva za sve uređaje
  - Inicira i završava U/I zahteve
  - Bira uređaj kojem se pristupa
  - Vrši raspoređivanje zahteva za uređaje
- Deo OS

# Logički U/I

- Čini ga sistem za upravljanje fajlovima kao poseban sistemski uslužni program
- Apstrakcija upravljanja fizičkim blokovima u sekundarnom skladištu
  - Omogućuje korisnicima rad sa fajlovima
- Pruža standardni interfejs između aplikacija preko fajl sistema do uređaja

# Elementi upravljanja fajlovima



# Organizacija fajlova

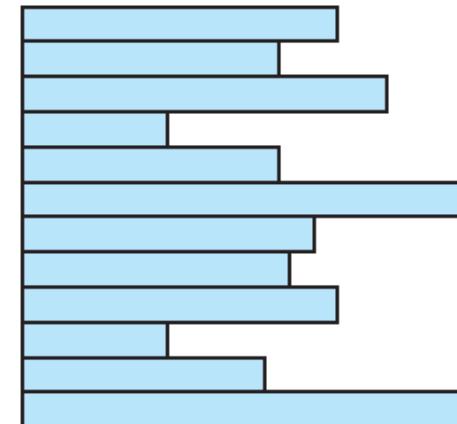
- Logičko strukturiranje zapisa koji određuje način na koji se zapisima pristupa
- Kriterijumi izbora organizacije
  - Vreme pristupa
  - Jednostavnost ažuriranja
  - Ekonomičnost skladišta
  - Jednostavnost održavanja
  - Pouzdanost

# Tipovi organizacije fajlova

- Gomila
- Sekvencijalni fajl
- Indeksni sekvencijalni fajl
- Indeksni fajl
- Direktni ili heširani fajl

# Gomila

- Podaci se prikupljaju u redosledu u kojem dolaze
  - Nema strukture
- Zapisi mogu da imaju različita polja i dužinu
- Komplikovana pretraga
  - Moraju se proći svi zapisi i sva polja



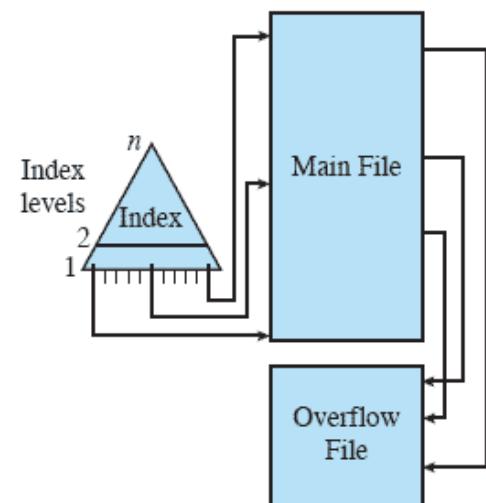
Variable-length records  
Variable set of fields  
Chronological order

# Sekvencijalni fajl

- Fiksni format zapisa
  - Svi zapisi iste dužine
    - Ista polja u istom redosledu
  - Zapis ima polje koje je ključ
  - Zapisi najčešće sortirani po ključu
  - Spora pretraga
    - Potrebno je sekvencijalnim prolaskom pronaći zapis sa odgovarajućim ključem
  - Komplikovano ažuriranje
    - Problem je ubaciti novi zapis jer su postojeći sortirani

# Indeksni sekvencijalni faj

- Zapisi isti kao kod sekvencijalnog
- Dva proširenja
- Indeksi koji omogućuju brži pristup
  - Mogu biti u više nivoa
  - Indeks je sekvencijalni fajl
  - Svaki slog sadrži ključ (jednak ključu u glavnom fajlu) i pokazivač na glavni fajl
  - U indeksnom fajlu je manji broj slogova
  - Preko ključa u indeksu se dolazi do odgovarajućeg dela glavnog fajla

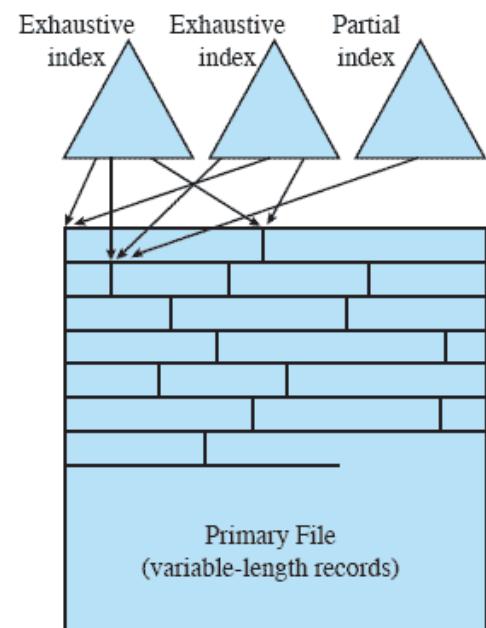


# Indeksni sekvencijalni fajl

- Fajl prekoračenja
  - Novi zapis ide u poseban fajl
  - U zapis koji prethodi novom zapisu se upiše pokazivač na novi zapis u fajlu prekoračenja
  - Periodično se vrši spajanje fajla prekoračenja sa glavnim fajлом

# Indeksni fajl

- Različiti indeksi za različita polja
- Omogućuju brzo pronalaženje ne samo zapisa, nego i polja unutar zapisa
- Zapisi mogu biti različitih dužina

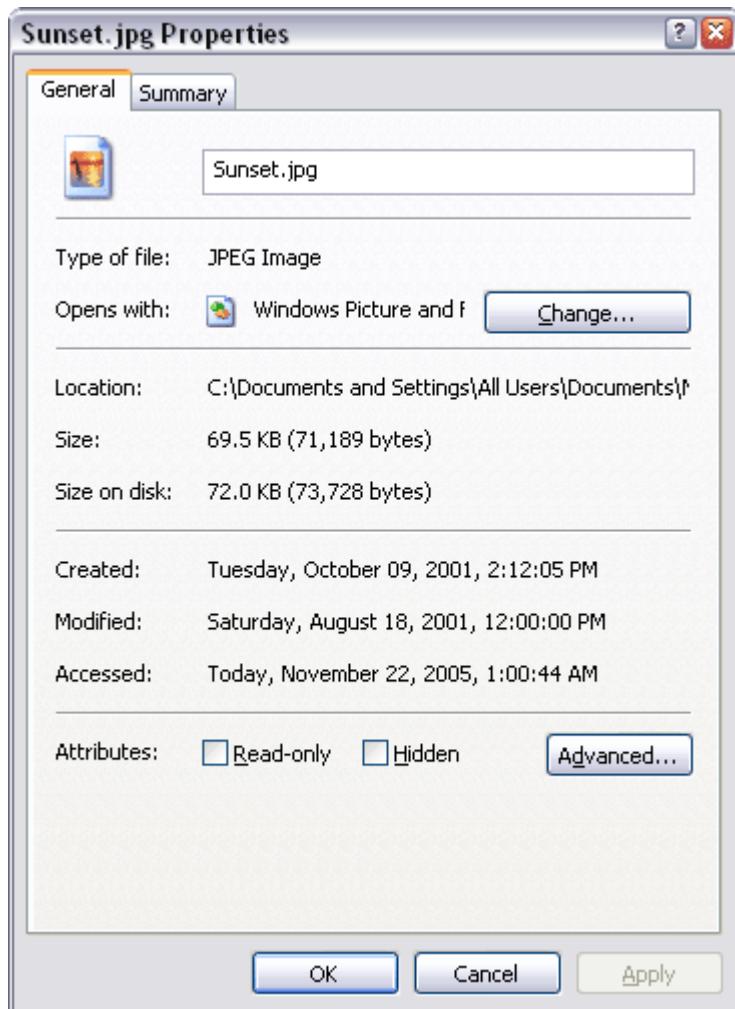


# Direktni ili heširani fajl

- Zapisi imaju ključ
- Izračunava se heš kod na osnovu ključa
- Pristupa se lokaciji na koju pokazuje izračunati heš
- Na toj lokaciji može biti više zapisa koji imaju isti heš kod
  - Prolazi se sekvencijalno kroz sve takve zapise da bi se pronašao željeni zapis

# Informacije o fajlu

## Upravljački blok fajla



[file permissions](#)

[file dates \(create, access, write\)](#)

[file owner, group, ACL](#)

[file size](#)

[file data blocks or pointers to file data blocks](#)

# Informacije o fajlu

## Osnovne informacije

- Ime fajla
  - Jedinstveno unutar direktorijuma
- Tip fajla
  - Korisniku predstavljeno ekstenzijom

# Informacije o fajlu

## Adresne informacije

- Volumen
  - Uredaj na kojem je fajl uskladišten
- Početna adresa
  - Adresa u sekundarnom skladištu u kojoj je smešten prvi blok fajla
- Veličina
  - Trenutna veličina fajla

# Informacije o fajlu

## Upravljanje pristupom

- **Vlasnik**
  - Korisnik koji ima specijalna prava u upravljanju fajlom
- **Prava pristupa**
  - Za različite korisnike definiše operacije koje mogu da vrše nad fajlom

# Informacije o fajlu

## Informacije o upotrebi

- Vreme kreiranja
- Korisnik koji je kreirao fajl
- Vreme poslednjeg čitanja
- Korisnik koji je poslednji izvršio čitanje
- Vreme poslednje modifikacije
- Korisnik koji je poslednji izvršio modifikaciju
- Informacije o trenutnoj upotrebi
  - Koji procesi koriste fajl
  - Da li je fajl zaključan
  - Da li je izmenjen u memoriji

# Direktorijumi fajlova

- Fajlovi se evidentiraju u direktorijumima
- Direktorijum obezbeđuje mapiranje imena fajlova na fajl resurse na disku
- I direktorijum je fajl

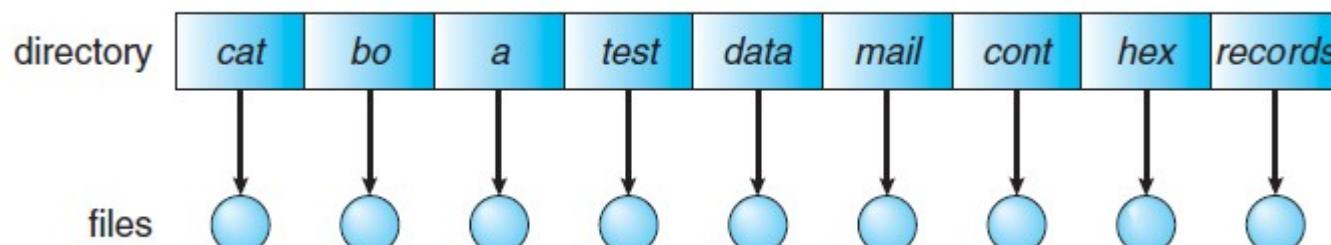


# Operacije nad direktorijumom

- Pretraživanje
  - Pronalaženje fajla u direktorijumu
- Kreiranje fajla
  - Potrebno je dodati novu stavku u direktorijum
- Brisanje fajla
  - Potrebno je obrisati stavku u direktorijumu
- Listanje direktorijuma
  - Spisak fajlova u direktorijumu sa dodatnim informacijama o fajlovima
- Ažuriranje direktorijuma
  - Promena atributa fajla koji se skladište u direktorijumu

# Struktura direktorijuma

- Pravi se tako da efikasno podrži opisane operacije
- Najjednostavnija varijanta
  - Lista stavki
  - Svaka stavka opisuje jedan fajl
  - Lista se definiše u sekvencijalnom fajlu
  - Fajlovi nisu nikako organizovani
  - Ime fajla mora biti jedinstveno na nivou čitavog sistema

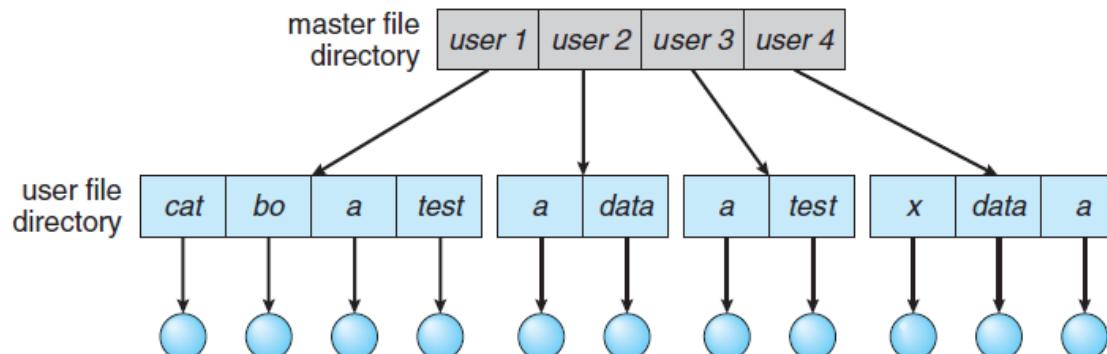


\*Slika preuzeta iz „Operating System Concepts 9th Edition“, Silbershatz,  
Galvin i Gagne.

# Struktura direktorijuma

## □ Organizacija u dva nivoa

- Pokušaj da se reše neki od problema sekvenčne organizacije direktorijuma
- Postoji jedan glavni direktorijum i
- Po jedan direktorijum za svakog korisnika
- Korisnički direktorijum je sekvenčni spisak korisnikovih fajlova
- Na nivou jednog korisnika, fajlovi moraju imati jedinstveno ime

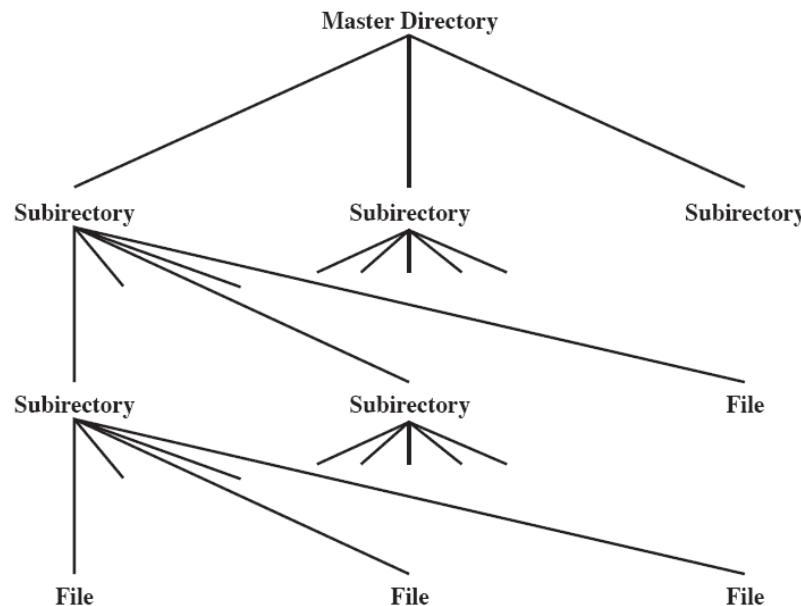


\*Slika preuzeta iz „Operating System Concepts 9th Edition“, Silberschatz,  
Galvin i Gagne.

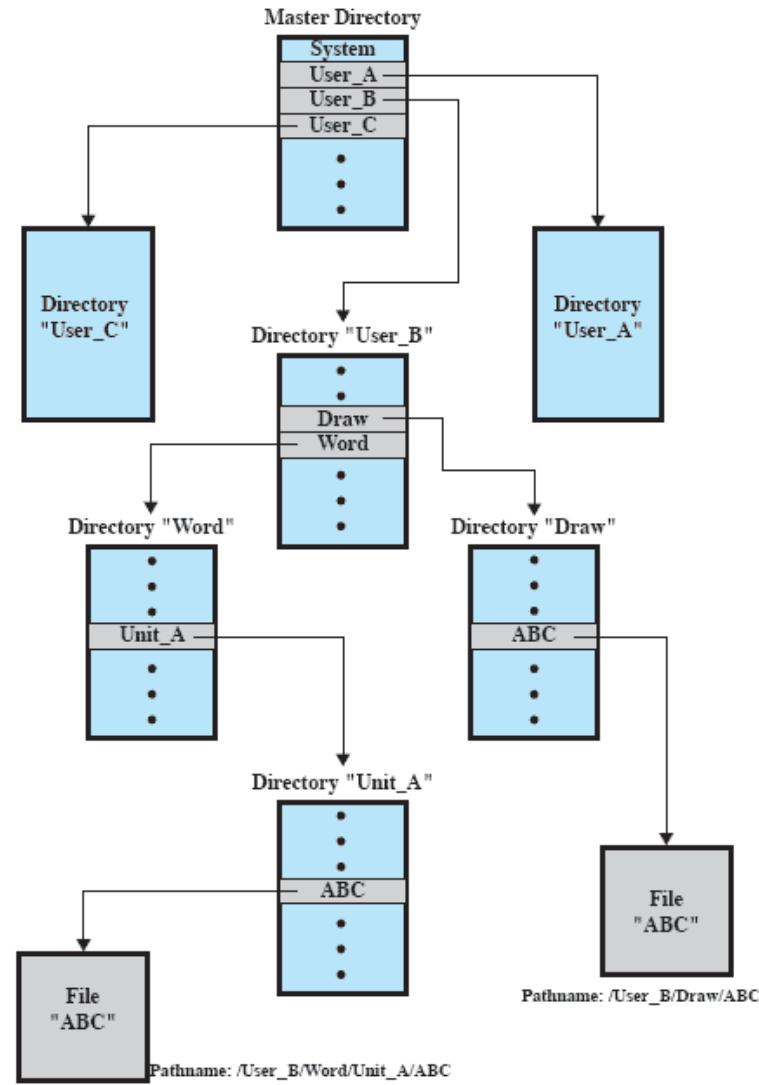
# Struktura direktorijuma

## □ Hjerarhijska struktura

- Univerzalno prihvaćena varijanta
- Svaki direktorijum može da sadrži poddirektoriume i failove



# Primer hijerarhijske strukture



# Imenovanje

- Korisnik treba da ima mogućnost referenciranja fajla po imenu
  - Nije realno očekivati da se svim fajlovima na disku daju različita imena
- Fajlovi se referenciraju preko putanje
  - Npr. /home/rs12345/downloads/documents
  - Putanja se dobija idući od korenskog direktorijuma, preko njegovih poddirektorijuma dok se dođe do direktorijuma u kojem je traženi fajli
  - Ista imena fajlova su dozvoljena ako se putanje razlikuju

# Radni direktorijum

- Stalno navođenje pune putanje je naporno za korisnika
- U svakom trenutku postoji **radni direktorijum**
  - Direktorijum u kojem je korisnik trenutno pozicioniran
  - Sve putanje do fajlova su relativne u odnosu na radni direktorijum
  - Osim ako se eksplicitno navede puna putanja

# Deljenje fajlova

- Standardni zahtev u multiprogramiranom sistemu je da iste fajlove koriste različiti procesi i korisnici
- Potrebno je upravljati
  - Pravima pristupa fajlu
  - Istovremenim pristupom fajlu



# Prava pristupa fajlu

- Nikakvo
  - Korisnik nije ni svestan postojanja fajla
- Znanje
  - Korisnik može da utvrdi da fajl postoji i ko je vlasnik
- Izvršavanje
  - Korisnik može da učita i izvrši program
  - Ne može da kopira fajl

# Prava pristupa fajlu

- Čitanje
  - Korisnik može da čita fajl u bilo koju svrhu
  - Najčešće uključuje i pravo kopiranja
- Dodavanje podataka
  - Korisnik ima pravo samo da dodaje nove podatke u fajl
  - Ne može da menja ili briše postojeći sadržaj

# Prava pristupa fajlu

- Ažuriranje
  - Korisnik može da menja, briše i dodaje podatke u fajl
- Promena prava pristupa
  - Korisnik može da promeni prava pristupa nad fajlom
  - Obično ovo pravo ima samo vlasnik
- Brisanje
  - Korisnik ima pravo da briše fajl

# Prava pristupa fajlu

- Od svih pomenutih prava najčešće je podržana kontrola samo za
  - Pravo čitanja
  - Pravo upisa
  - Pravo izvršavanja

# Organizacija korisnika

- Prava se dodeljuju
  - Specifičnom korisniku
    - Identifikovan na osnovu korisničkog identifikatora
  - Korisničkoj grupi
    - Korisnici se organizuju u grupe
    - Korisnik može biti član više grupa
  - Svima
    - Prava se odnose na sve korisnike bez obzira na njihov identitet ili pripadnost grupama

# Istovremeni pristup fajlu

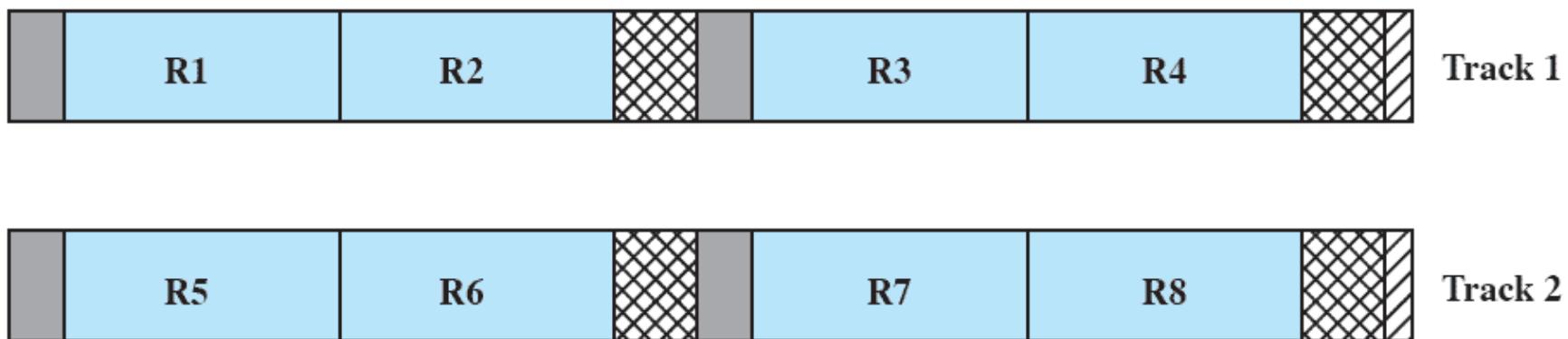
- Različiti procesi mogu istovremeno da pristupaju fajlu
- U principu je to dozvoljeno ponašanje
- Moguće je zaključati fajl da bi se zabranila istovremena izmena

# Blokovi i zapisi

- Zapisi su logičke jedinice pristupa strukturiranom fajlu
- Blokovi su jedinice nad kojima se vrše U/I operacije sa sekundarnim skladištem
- Tri metoda mapiranja zapisa na blokove
  - Fiksno blokiranje
  - Blokiranje promenljive dužine sa premošćenjem
  - Blokiranje promenljive dužine bez premošćenja

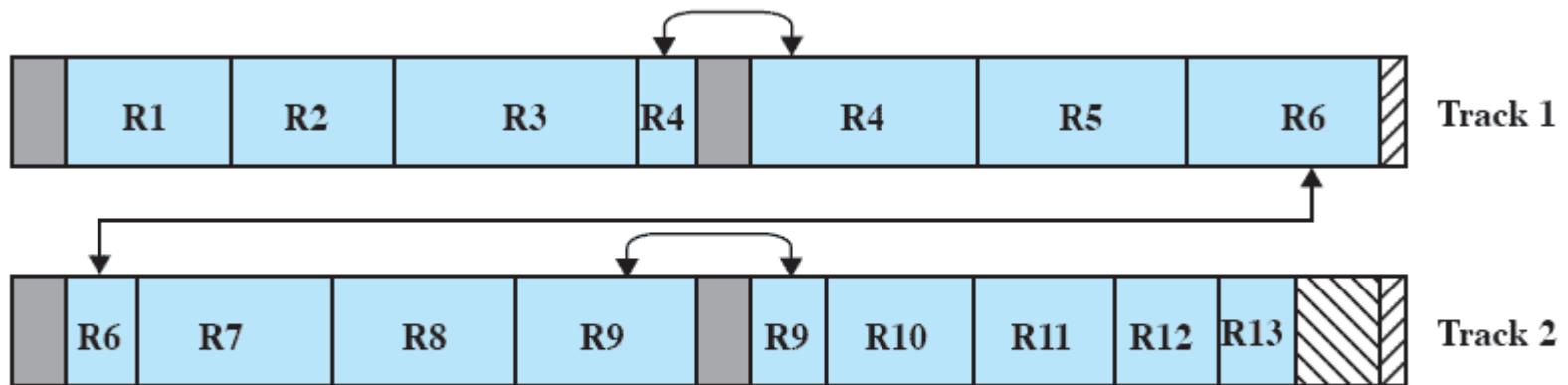
# Fiksno blokiranje

- Uobičajena varijanta
- Blokovi fiksne dužine
- Ceo broj zapisa se smešta u blok
- Na kraju svakog bloka može biti neiskorišćenog prostora
  - **Internaz fragmentacija**



# Blokiranje promenljive dužine sa premošćenjem

- Zapisi promenljive dužine se pakuju u blokove promenljive dužine bez neiskorišćenog prostora
- Neki zapisi mogu da se protežu na više blokova
  - Pokazivač ukazuje na sledeći blok gde se zapis nastavlja
- Teško za implementaciju



# Blokiranje promenljive dužine bez premošćenja

- Blokovi su promenljive dužine, ali nema premošćavanja
- Neiskorišćen prostor
  - Ostatak bloka se ne koristi ako naredni zapis ne može da stane u preostali neiskorišćeni prostor



# Upravljanje sekundarnim skladištem

- OS je odgovoran za dodeljivanje blokova diska fajlovima
- Dva pitanja upravljanja
  - Prostor u sekundarnom skladištu treba dodeljivati fajlovima
  - Evidencija prostora raspoloživog za dodeljivanje



# Dodeljivanje fajlova

1. Kada se stvori novi fajl, da li se maksimalan prostor koji se traži dodeljuje odmah?
2. Prostor se dodeljuje fajlu u „delovima“
  - Deo je susedni skup dodeljenih blokova
  - Koja treba da bude veličina dela?
3. Kako voditi evidenciju o dodeljenim delovima?

# Dodeljivanje unapred naspram dinamičkog dodeljivanja

## □ Dodeljivanje unapred

- Potrebno je u trenutku kreiranja znati maksimalni prostor koji će fajlu trebati
- Teško je tu vrednost pouzdano proceniti
- Često se preceni potrebna veličina

## □ Dinamičko dodeljivanje

- Delovi se dodeljuju prema potrebi

# Veličina dela

- Dve krajnosti
  - Deo je dovoljno velik da se u njega smesti ceo fajl
  - Jedan deo ima veličinu jednog bloka
- Veliki delovi
  - Podaci su smešteni u susednim lokacijama, što povećava performansu pri pristupu
  - Teško je preraspoređivati prostor pri dodeljivanju novih fajlova
  - Ako su delovi promenljive veličine, nema interne fragmentacije
- Mali delovi
  - Komplikovanija evidencija dodeljivanja (tabele dodeljivanja)
  - Jednostavnije preraspoređivanje prostora
  - Rasipanje prostora minimalno

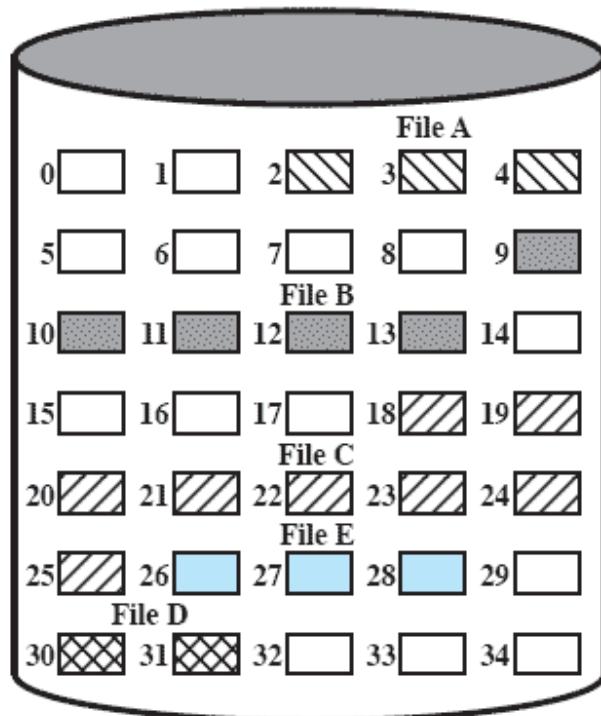
# Metode dodeljivanja fajlova

- Tri najčešće korišćene metode
  - Susedno dodeljivanje
  - Ulančano dodeljivanje
  - Indeksno dodeljivanje

# Susedno dodeljivanje

- Jedan skup susednih blokova dodeljuje se fajlu u trenutku stvaranja
  - Dodeljivanje unapred
  - Delovi promenljive veličine
- Samo jedna stavka po fajlu u tabeli dodeljivanja
  - Početni blok i dužina fajla
- Jednostavno preuzimanje podataka i pristup i-tom bloku
  - Početni blok + i
- Eksterna fragmentacija
  - Potrebno sažimanje

# Susedno dodeljivanje



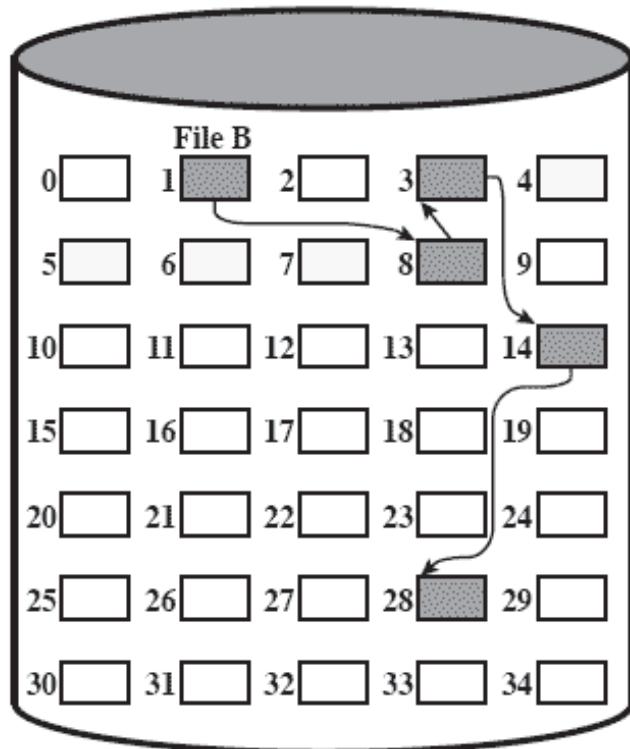
File Allocation Table

File Name	Start Block	Length
File A	2	3
File B	9	5
File C	18	8
File D	30	2
File E	26	3

# Ulančano dodeljivanje

- Dodeljivanje na bazi pojedinačnog bloka
  - Delovi fiksne veličine
  - Mogu se dodeljivati unapred, ali češće po potrebi
- Svaki blok sadrži pokazivač na sledeći blok u lancu
- Samo jedna stavka po fajlu u tabeli dodeljivanja
  - Početni blok i dužina fajla
- Nema eksterne fragmentacije
- Nije u saglasnosti sa principom lokalnosti
  - Pristupa se različitim delovima diska
  - Može da se vrši povremeno konsolidacija da se premeste podaci u susedne blokove
- Ne omogućuje direktni pristup bloku

# Ulančano dodeljivanje



File Allocation Table

File Name	Start Block	Length
...	...	...
File B	1	5
...	...	...

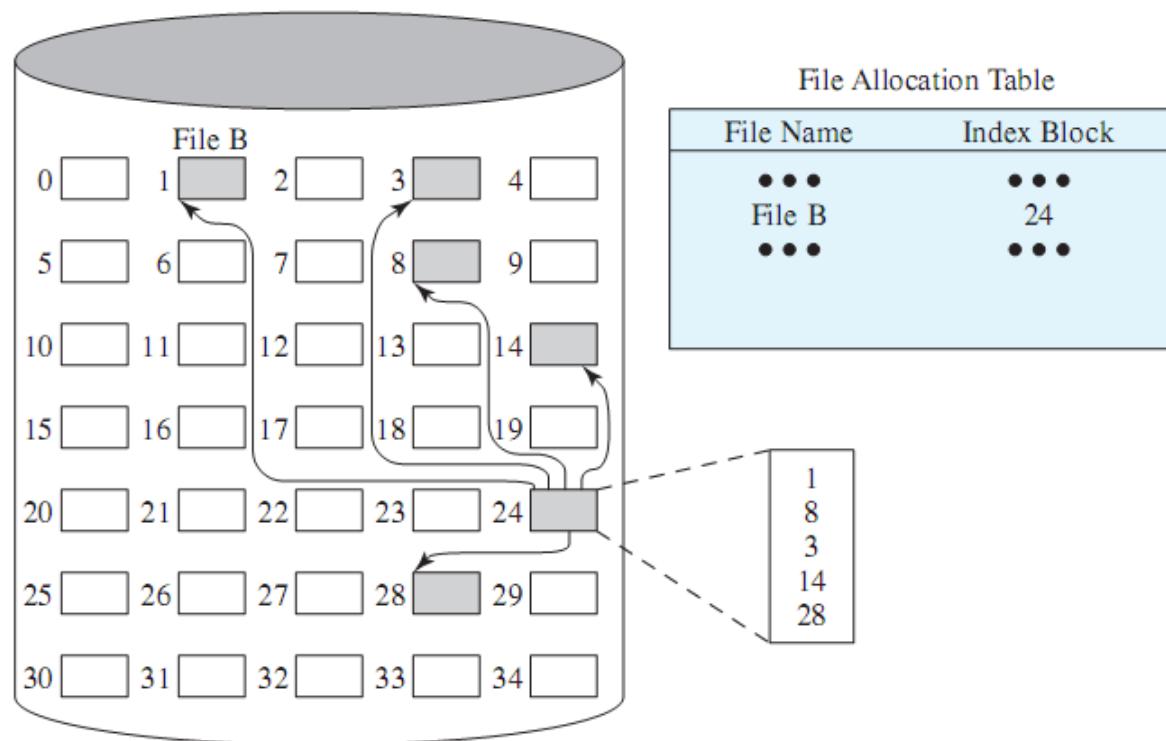
# Indeksno dodeljivanje

- Dodeljeni delovi ne moraju biti susedni
- Delovi ne pokazuju na sledeći deo
- Tabela dodeljivanja za svaki fajl sadrži evidenciju dodeljenih delova
  - Evidencija se zove indeks delova
  - Indeks sadrži jednu stavku za svaki dodeljeni deo
  - Indeks se čuva u posebnom bloku
  - Stavka u tabeli dodeljivanja sadrži adresu bloka u kojem je indeks
- Najpopularniji oblik dodeljivanja fajlova
  - Omogućuje i sekvencijalan i direktni pristup

# Indeksno dodeljivanje

## □ Prva varijanta

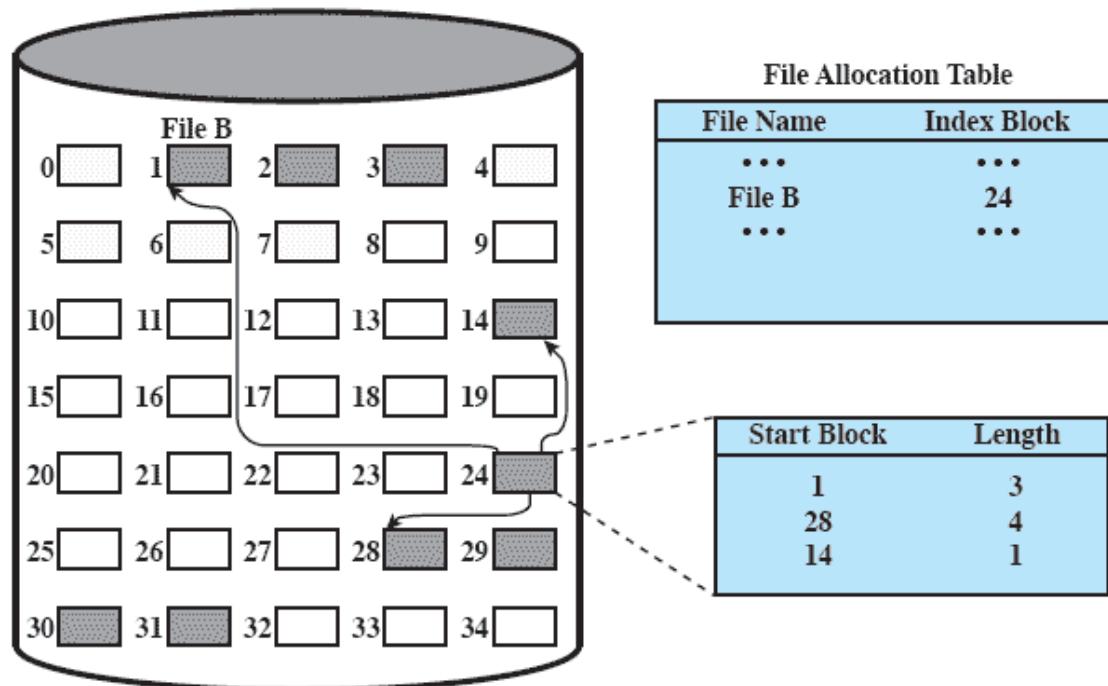
- Dodeljivanje na nivou blokova fiksne veličine
- Nema spoljašnje fragmentacije



# Indeksno dodeljivanje

## □ Druga varijanta

- Dodeljivanje na nivou delova promenljive veličine
- Poboljšava lokalnost



# Upravljanje slobodnim prostorom

- Kao što se vrši upravljanje dodeljivanjem, mora se vršiti upravljanje slobodnim prostorom
  - Sistem mora da zna koji prostor je slobodan da bi bio dodeljen fajlovima
- Osim tabele dodeljivanja fajlova, potrebna je i **tabela dodeljivanja diska**
  - Ima više varijanti da se realizuje

# Tabele bitova

- Vektor koji sadrži jedan bit za svaki blok
  - 0 ako je blok slobodan
  - 1 ako je blok zauzet
- Problemi
  - Velika tabela za velike diskove
  - Sporo pronalaženje slobodnih blokova

# Ulančani slobodni delovi

- Slobodni delovi se ulančavaju
  - korišćenjem pokazivača na sledeći i
  - Dužine slobodnog dela
- Evidentira se samo adresa i dužina prvog slobodnog dela
  - Nema potrebe za tabelom dodeljivanja diska
- Eksterna fragmentacija

# Indeksiranje

- Indeks slobodnih delova, kao kod indeksnog dodeljivanja fajla
  - Evidentiraju se slobodni delovi promenljive veličine
  - Po jedna stavka u tabeli za svaki slobodan deo
- Efikasna podrška za sve metode dodeljivanja diska

# Lista slobodnih blokova

- Svaki blok ima svoj broj
- Održava se lista brojeva slobodnih blokova
- Lista je velika pa se skladišti na disku
- Deo liste se skladišti u glavnoj memoriji zbog bržeg pristupa

# Bezbednost fajl sistema

- Svaki korisnik ima svoj identitet
- OS sprovodi pravila korišćenja sistema
  - Kontroliše pristup resursima sistema
- OS je potreban skup pravila za sprovođenje kontrole pristupa

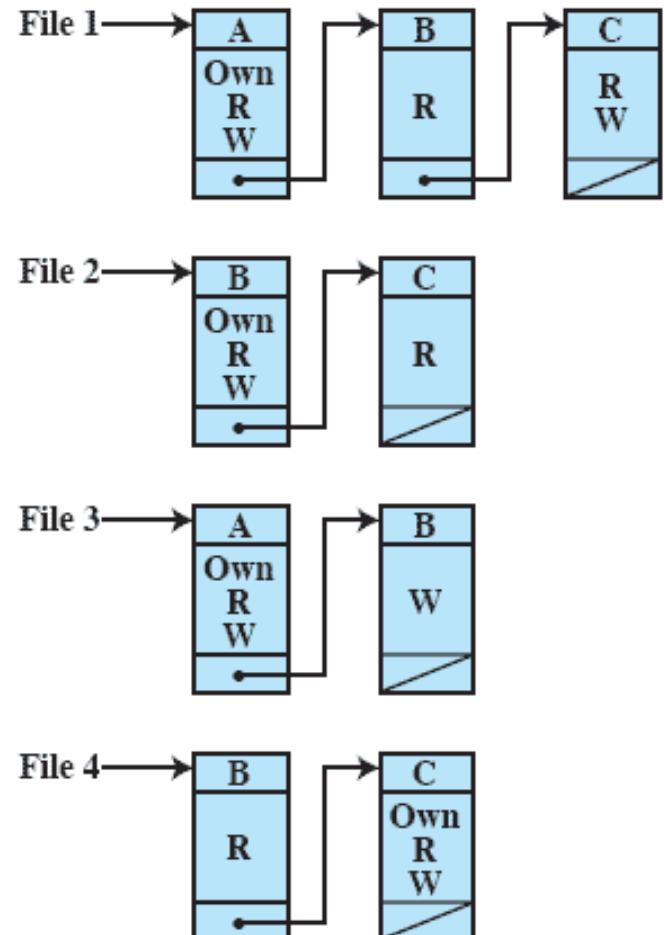
# Matrica pristupa

- Jedna varijanta evidencije prava pristupa
- Za svakog subjekta definiše se skup operacija koje može da sprovodi nad određenim objektom

	File 1	File 2	File 3	File 4	Account 1	Account 2
User A	Own R W		Own R W		Inquiry Credit	
User B	R	Own R W	W	R	Inquiry Debit	Inquiry Credit
User C	R W	R		Own R W		Inquiry Debit

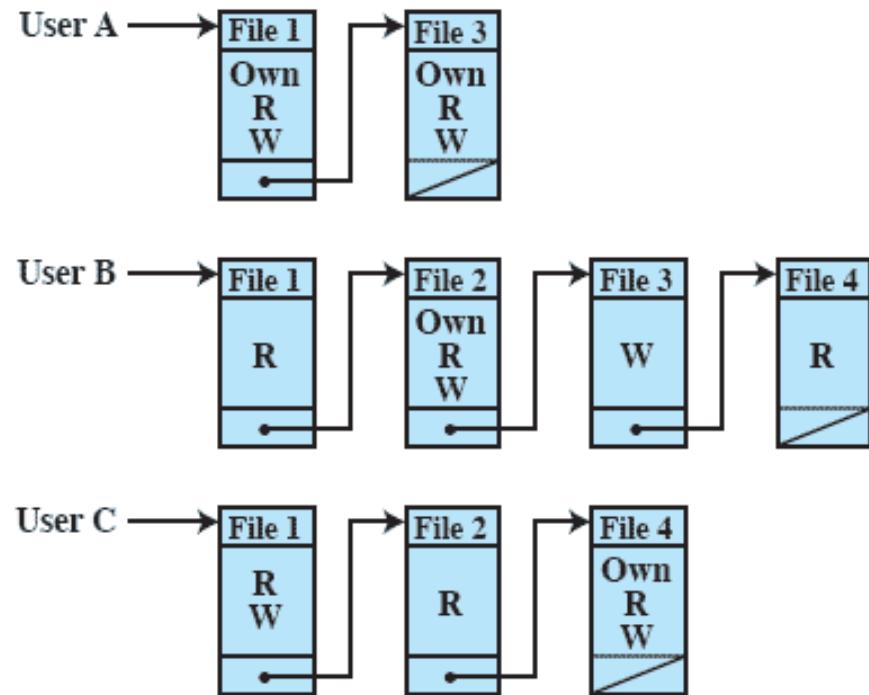
# Liste kontrole pristupa

- Dekompozicija matrice pristupa po kolonama
- Za svaki fajl se navode korisnici i njihova prava pristupa



# Liste sposobnosti

- Dekompozicija matrice pristupa po redovima
- Za svakog korisnika se navode prava pristupa nad fajlovima

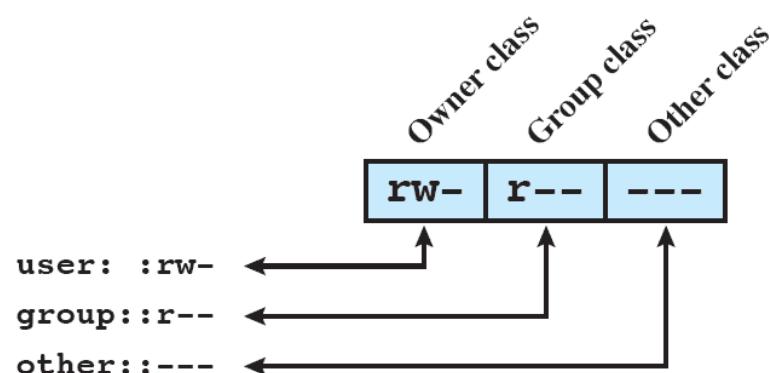


# Kontrola pristupa fajlovima u UNIX-baziranim sistemima

- Prava pristupa se definišu za tri subjekta:

- Vlasnik fajla
- Korisnici koji pripadaju grupi kojoj fajl pripada
- Ostali korisnici

- Moguća prava
  - Čitanje
  - Pisanje
  - Izvršavanje



# Implementacija fajl sistema



# Strukture za upravljanje fajlovima

## □ Na disku

- *Boot control block*
  - Za svaki volumen po jedan (ili nijedan)
  - Informacije potrebne za startovanje operativnog sistema
  - Volumen ih skladišti najčešće u prvom bloku (ukoliko se OS nalazi na volumenu)
- *Volume control block*
  - Za svaki volumen po jedan
  - Generalne informacije o volumenu
    - Broj i veličina blokova
    - Lista slobodnih blokova

# Strukture za upravljanje fajlovima

## □ Na disku

- Struktura direktorijuma
  - Po jedna za svaki fajl sistem
  - Definiše spisak i organizaciju fajlova
- Upravljački blok fajla
  - Po jedan za svaki fajl
  - Informacije o fajlu (naziv, vlasnik, veličina, ...)

# Strukture za upravljanje fajlovima

## □ U memoriji

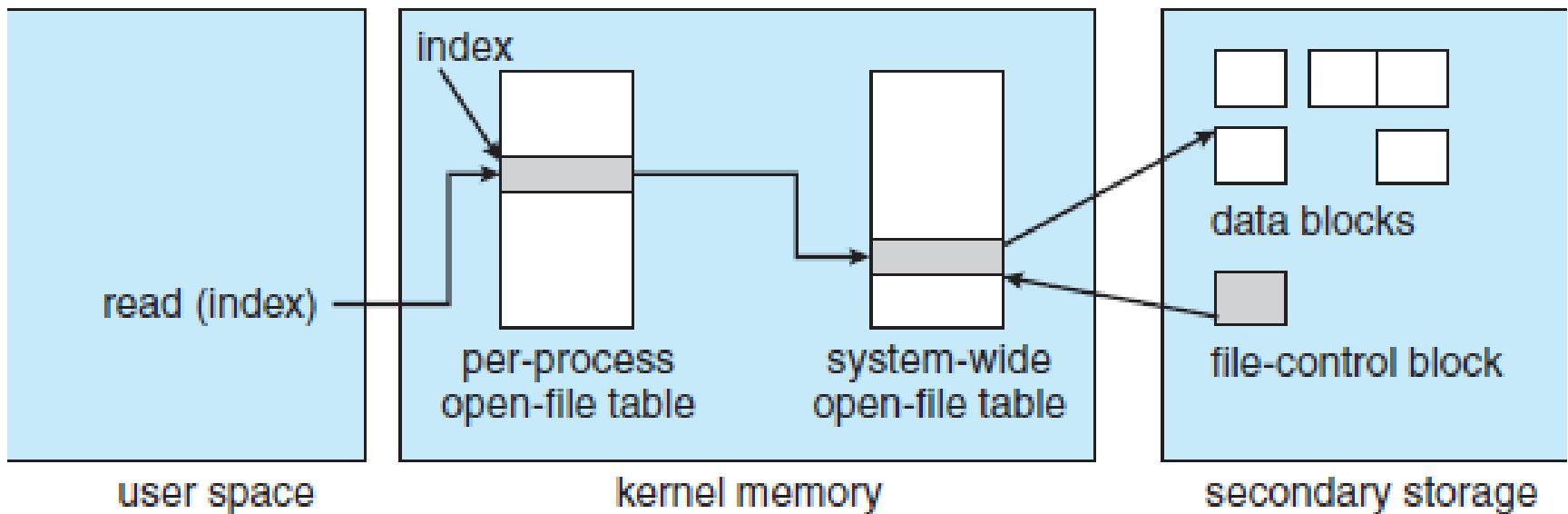
- Informacije o svakom volumenu u fajl sistemu
- Globalna tabela otvorenih fajlova
  - Jedna za ceo sistem
  - Sadrži kopije upravljačkih blokova otvorenih fajlova (onih kojima se trenutno pristupa)
- Lokalne tabele otvorenih fajlova
  - Po jedna za svaki proces
  - Trenutna pozicija unutar fajla pri čitanju/pisanju
  - Mod pristupa u kojem je fajl otvoren (samo čitanje, upis, ...)

# Strukture za upravljanje fajlovima

## □ U memoriji

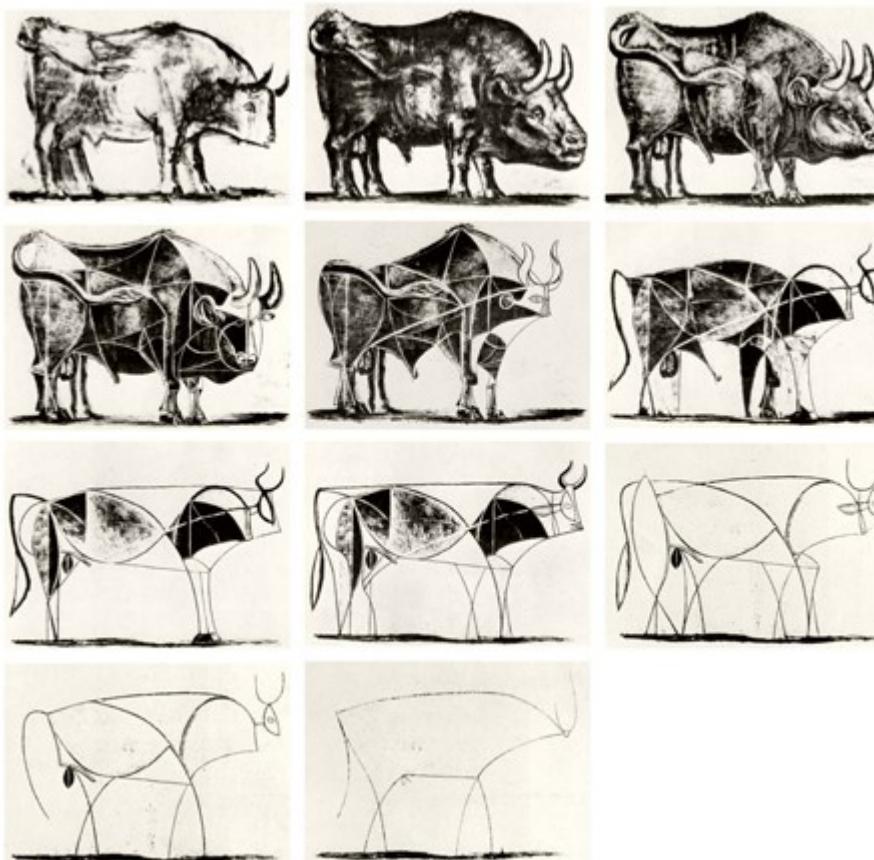
- Baferi
  - Sadrže blokove fajla kada se čitaju/upisuju na disk
- Keš direktorijuma
  - Informacije o direktorijumima kojima je nedavno pristupano

# Strukture za upravljanje fajlovima



# Implementacija fajl sistema

- Kako da OS podrži različite fajl sisteme?
- Rešenje je apstrakcija



# Implementacija fajl sistema

- Umesto da OS implementira funkcije za svaki specifičan fajl sistem
  - OS sadrži generičke funkcije za rad sa fajl sistemom
  - Konkretan fajl sistem specijalizuje generičku funkcionalnost za svoju strukturu i ponašanje
  - Kao polimorfizam u objektno orijentisanom programiranju

# Implementacija fajl sistema

Generički sistemski pozivi -  
open(), read(), write(), ...

- Interfejs koji svaki konkretan fajl sistem mora da podrži
- Specifikacija funkcija za konkretne fajl sisteme

Konkretan fajl  
sistem

