Operativni sistemi: interna struktura i principi dizajna

Poglavlje 12 Upravljanje datotekama

Sedmo izdanje

William Stallings

Datoteke

- Kolekcije podataka koje kreiraju korisnici
- Za korisnika je sistem datoteka jedan od najvažnijih delova OS-a
- Poželjne osobine datoteka:

Dugotrajno postojanje

- datoteke se čuvaju na disku ili drugoj sekundarnoj memoriji i ne nestaju kada se korisnik odjavi

Deljivost među procesima

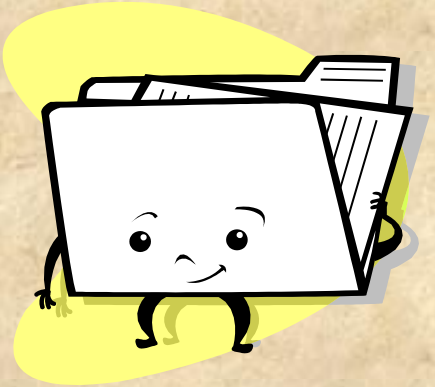
- datoteke imaju imena i mogu imati dozvole pristupa koje omogućavaju kontrolisano deljenje

Struktura

- datoteke se mogu organizovati u hijerarhijsku ili složeniju strukturu koja odražava odnose među datotekama

Sistemi datoteka

- Sistem datoteka pruža korisnicima apstrakciju diska
- Omogućava čuvanje podataka organizovanih u datoteke, kao i skup funkcija koje se mogu izvršavati nad datotekama
- Održava skup atributa pridruženih datoteci
- Tipične operacije su:
  - Kreiranje/brisanje
  - Otvaranje/zatvaranje
  - Čitanje/upis



Ciljevi sistema za upravljanje datotekama

- Zadovoljiti potrebe korisnika za upravljanjem podacima
- Garantovati da su podaci u datoteci ispravni
- Optimizovati performanse
- Obezbediti U/I podršku za različite tipove uređaja za skladištenje
- Minimizovati mogućnost gubitka ili uništenja podataka
- Obezbediti korisničkim procesima standardizovan skup rutina U/I interfejsa
- Obezbediti U/I podršku za više korisnika kod višekorisničkih sistema

Minimalni zahtevi korisnika

- Svaki korisnik:

1. treba da može da kreira, briše, čita, upisuje i menja datoteke
2. može imati kontrolisan pristup datotekama drugih korisnika
3. može kontrolisati koje su vrste pristupa njegovim datotekama dozvoljene
4. treba da može da restrukturira datoteke u oblik pogodan za dati problem
5. treba da može da premešta podatke između datoteka
6. treba da može da pravi rezervne kopije datoteka i da ih oporavi u slučaju oštećenja
7. treba da može da pristupa svojim datotekama po imenu, a ne po numeričkom identifikatoru

Informacije u direktorijumu datoteka

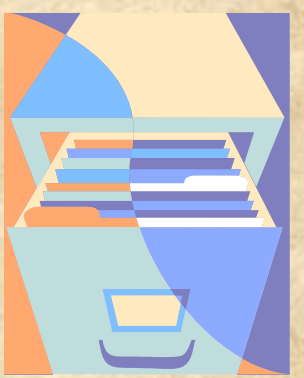


Tabela 12.2 Elementi informacija u direktorijumu datoteka

	Osnovne informacije
Ime datoteke	Ime koje je izabrao tvorac (korisnik ili program). Mora biti jedinstveno u okviru datog direktorijuma.
Tip datoteke	Na primer: tekstualna, binarna, modul za učitavanje itd.
Organizacija datoteke	Za sisteme koji podržavaju različite organizacije
	Informacije o adresi
Volumen	Označava uređaj na kome je datoteka smeštena
Početna adresa	Početna fizička adresa u sekundarnoj memoriji (npr. cilindar, staza i broj bloka na disku)
Zauzeta veličina	Trenutna veličina datoteke u bajtovima, rečima ili blokovima
Alocirana veličina	Maksimalna veličina datoteke
	Informacije o kontroli pristupa
Vlasnik	Korisnik kome je dodeljena kontrola nad datotekom. Vlasnik može da odobrava/uskraćuje pristup drugim korisnicima i da menja te privilegije.
Informacije o pristupu	U najjednostavnijem obliku, ovaj element sadrži ime i lozinku svakog ovlašćenog korisnika.
Dozvoljene akcije	Kontroliše čitanje, upis, izvršavanje i prenos preko mreže
	Informacije o korišćenju
Datum kreiranja	Kada je datoteka prvi put smeštena u direktorijum
Identitet tvorca	Obično, ali ne obavezno, trenutni vlasnik
Datum poslednjeg čitanja	Datum kada je poslednji put pročitan neki zapis
Identitet poslednjeg čitaoca	Korisnik koji je izvršio čitanje
Datum poslednje izmene	Datum poslednjeg ažuriranja, umetanja ili brisanja
Autor poslednje izmene	Korisnik koji je izvršio izmenu
Datum rezervne kopije	Datum kada je poslednji put napravljena rezervna kopija datoteke na drugom medijumu
Trenutno korišćenje	Informacije o trenutnoj aktivnosti nad datotekom, npr. koji proces ili procesi drže datoteku otvorenom, da li ju je neki proces zaključao i da li je datoteka ažurirana u glavnoj memoriji, ali još ne i na disku

Operacije koje se izvode nad direktorijumom

- Da bismo razumeli zahteve za strukturu datoteka, korisno je razmotriti vrste operacija koje se mogu izvoditi nad direktorijumom:

Pretraga

Kreiranje datoteka

Brisanje datoteka

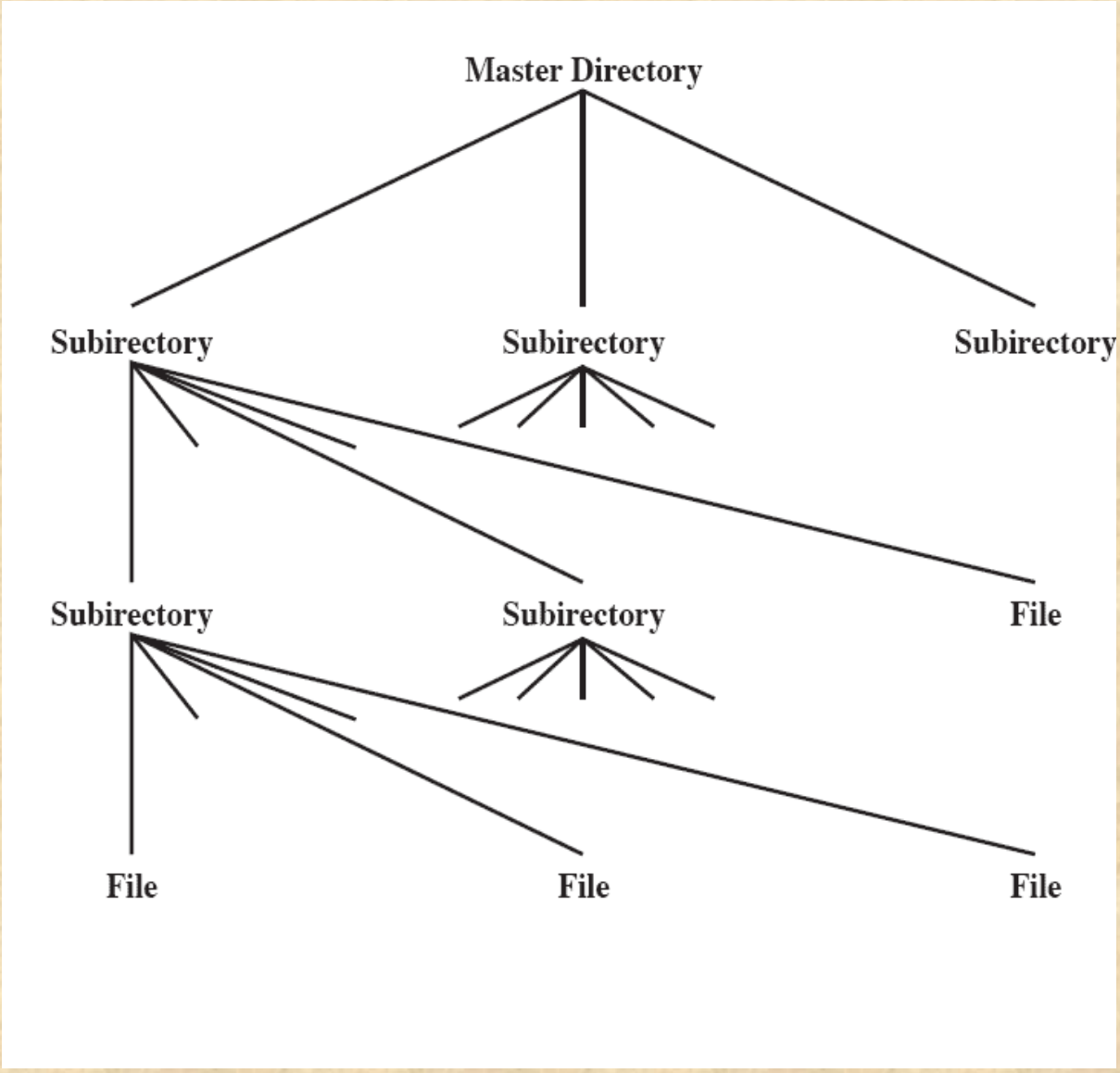
Listanje sadržaja

Izmena stavki



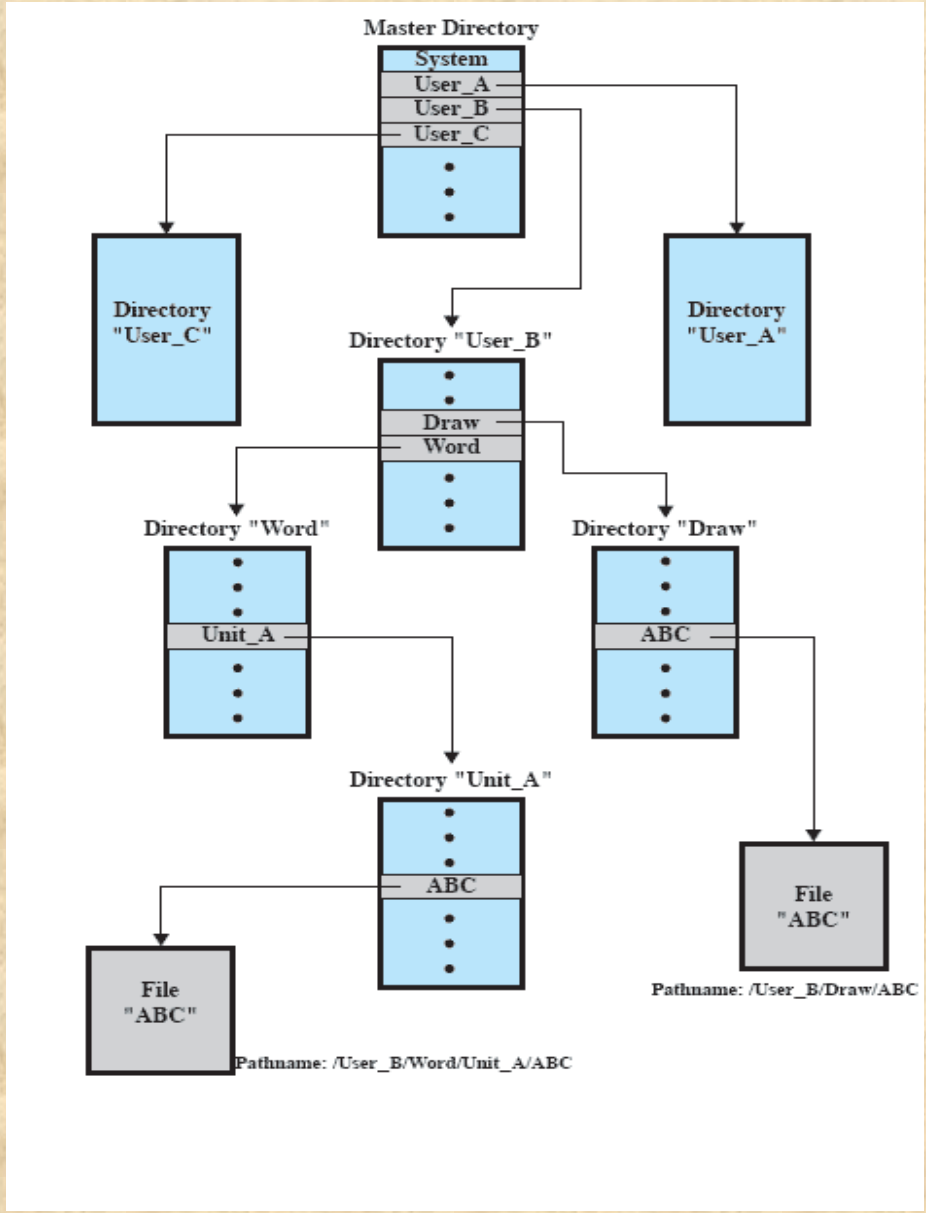
Sl. 12.4: Direktorijum sa strukturom stabla

- Glavni direktorijum sa korisničkim direktorijumima
- Korisnički direktorijum sadrži poddirektorijume i datoteke
- Pojednostavljuje jedinstveno imenovanje datoteka više korisnika.



Slika 12.7
Primer stabla direktorijuma





Deljenje datoteka



Kada se datoteke dele među više korisnika, javljaju se dva pitanja:

- prava pristupa
- upravljanje istovremenim pristupom

Prava pristupa



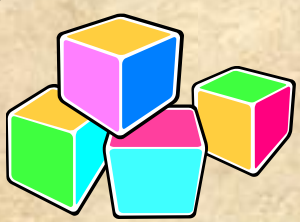
- *Bez prava*
  - korisniku nije dozvoljeno ni da čita korisnički direktorijum u kome je datoteka
- *Uvid*
  - korisnik može da utvrdi da datoteka postoji i ko je njen vlasnik, pa od vlasnika može da zatraži dodatna prava pristupa
- *Izvršavanje*
  - korisnik može da učita i izvrši program, ali ne i da ga kopira
- *Čitanje*
  - korisnik može da čita datoteku u bilo koje svrhe, uključujući kopiranje i izvršavanje
- *Dodavanje*
  - korisnik može da dodaje podatke u datoteku, ali ne može da menja niti briše njen postojeći sadržaj
- *Ažuriranje*
  - korisnik može da menja, briše i dodaje podatke datoteke
- *Promena zaštite*
  - korisnik može da menja prava pristupa dodeljena drugim korisnicima
- *Brisanje*
  - korisnik može da obriše datoteku iz sistema datoteka

Prava pristupa korisnika



Blokiranje zapisa

- Blokovi su jedinica U/I sa sekundarnom memorijom
  - da bi se obavio U/I, zapisi moraju biti organizovani u blokove



- Za datu veličinu bloka mogu se koristiti tri metoda blokiranja:

1) **Fiksno blokiranje** – koriste se zapisi fiksne dužine, a u blok se smešta ceo broj zapisa (ili bajtova)
   Unutrašnja fragmentacija – neiskorišćen prostor na kraju svakog bloka kod zapisa, ali ne i kod bajtova
   pogodno za datoteke organizovane kao tok bajtova.

2) **Blokiranje promenljive dužine sa prelaskom** – zapisi se pakuju u blokove bez praznog prostora

3) **Blokiranje promenljive dužine bez prelaska** – zapisi promenljive dužine ne prelaze granicu bloka

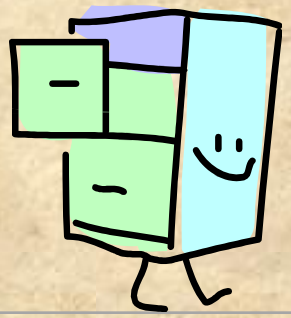


Alokacija datoteka

- Diskovi su podeljeni na *fizičke* blokove (sektore na stazi)
- Datoteke su podeljene na *logičke* blokove (delove datoteke)
- Veličina logičkog bloka = umnožak veličine fizičkog bloka
- Operativni sistem ili sistem za upravljanje datotekama zadužen je za alokaciju blokova datotekama
- Prostor se datoteci dodeljuje kao jedan ili više ***delova*** (portions) – jedan ili više uzastopnih blokova diska. Deo je veličine logičkog bloka.
- ***Tabela alokacije datoteka (FAT):***
  - Opšti naziv za strukturu podataka kojom se prate delovi diska dodeljeni datoteci

Prealokacija naspram dinamičke alokacije

- Politika prealokacije zahteva da se maksimalna veličina datoteke deklariše u trenutku zahteva za kreiranje datoteke
- Za mnoge aplikacije teško je pouzdano proceniti maksimalnu moguću veličinu datoteke
  - često dovodi do rasipanja prostora jer korisnici i programeri aplikacija obično precenjuju veličinu
- Dinamička alokacija dodeljuje prostor datoteci u delovima, po potrebi





Veličina dela

- Pri izboru veličine dela pravi se kompromis između efikasnosti sa stanovišta pojedinačne datoteke i ukupne efikasnosti sistema
- Treba uzeti u obzir:
  1) kontinualnost prostora poboljšava performanse, posebno kod operacija `Retrieve_Next` (sekvencijalni pristup).
  2) veliki broj malih delova povećava veličinu tabela potrebnih za upravljanje informacijama o alokaciji
  3) delovi fiksne veličine pojednostavljuju ponovnu alokaciju prostora
  4) delovi promenljive veličine ili mali delovi fiksne veličine minimizuju gubitak neiskorišćenog prostora usled prekomerne alokacije

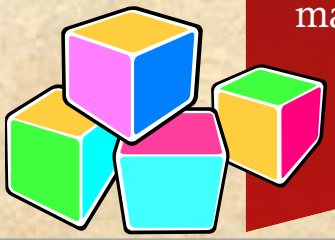
Ukratko o alternativama

- Dve glavne alternative:



Veliki kontinualni delovi promenljive veličine

- daju bolje performanse, naročito kod sekvencijalnog pristupa
- promenljiva veličina sprečava rasipanje
- tabele alokacije datoteka su male



Blokovi

- mali fiksni delovi pružaju veću fleksibilnost
- mogu zahtevati velike tabele ili složene strukture za svoju alokaciju
- kontinualnost više nije primarni cilj
- blokovi se alociraju po potrebi

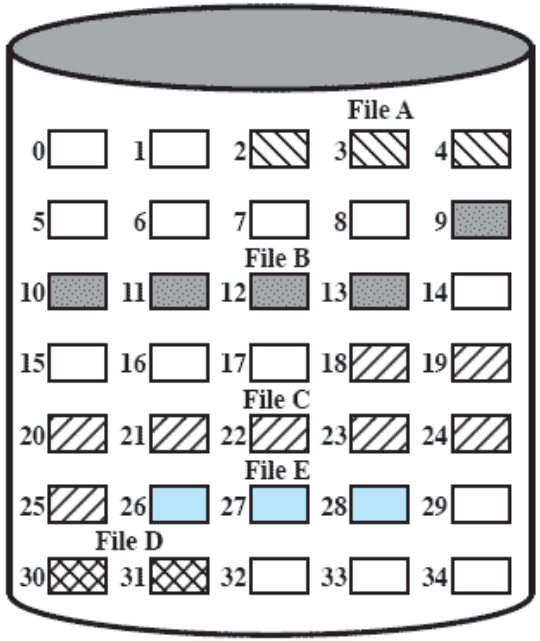
Tabela 12.3

Metodi alokacije datoteka

	Kontinualna	Ulančana	Indeksna	
Prealokacija?	Neophodna	Moguća	Moguća	
Delovi fiksni ili promenljivi?	Promenljivi	Fiksni blokovi	Fiksni blokovi	Promenljivi
Veličina dela	Velika	Mala	Mala	Srednja
Učestanost alokacije	Jednom	Od niske do visoke	Visoka	Niska
Dužina trajanja alokacije	Srednja	Duga	Kratka	Srednja
Veličina tabele alokacije	Jedna stavka	Jedna stavka	Velika	Srednja

Kontinualna alokacija datoteka

- Datoteci se pri kreiranju dodeljuje jedan kontinualan skup blokova
- Strategija prealokacije sa delovima promenljive veličine
- Najbolja je sa stanovišta pojedinačne sekvencijalne datoteke

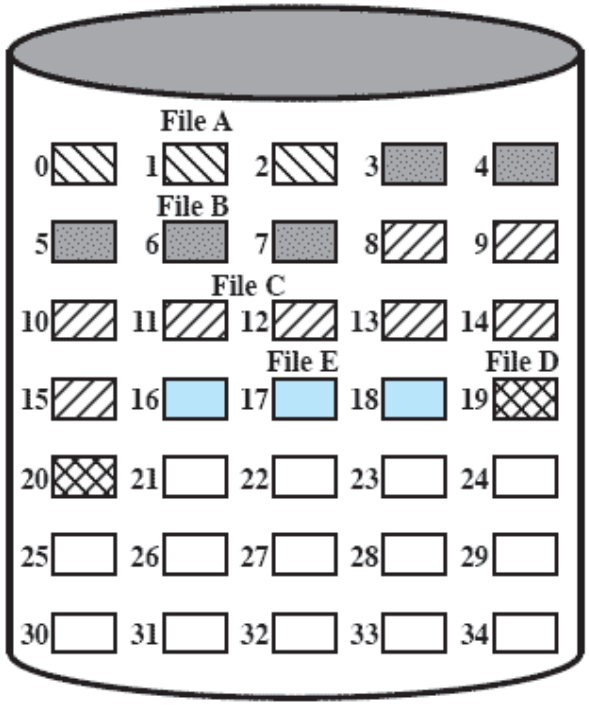


File Allocation Table

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

Figure 12.9 Contiguous File Allocation

Posle sažimanja



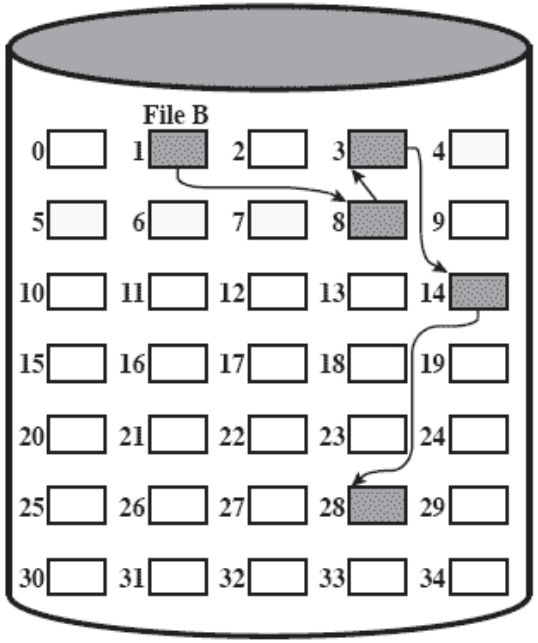
File Allocation Table

File Name	Start Block	Length
File A	0	3
File B	3	5
File C	8	8
File D	19	2
File E	16	3

Slika 12.10 Kontinualna alokacija datoteka (posle sažimanja)

Ulančana alokacija

- Alokacija se vrši na nivou pojedinačnih blokova
- Svaki blok sadrži pokazivač na sledeći blok u lancu
- Tabeli alokacije datoteka potrebna je samo jedna stavka po datoteci
- Nema spoljašnje fragmentacije
- Pogodnija za sekvencijalne datoteke

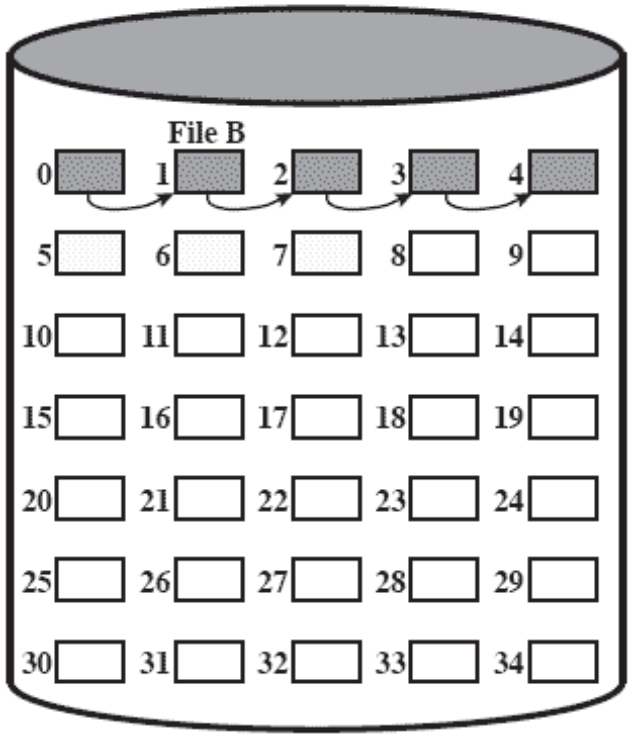


File Allocation Table

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

Figure 12.11 Chained Allocation

Ulančana alokacija posle konsolidacije



File Allocation Table

File Name	Start Block	Length
• • •	• • •	• • •
File B	0	5
• • •	• • •	• • •

Figure 12.12 Chained Allocation (After Consolidation)

Indeksna alokacija sa delovima veličine bloka

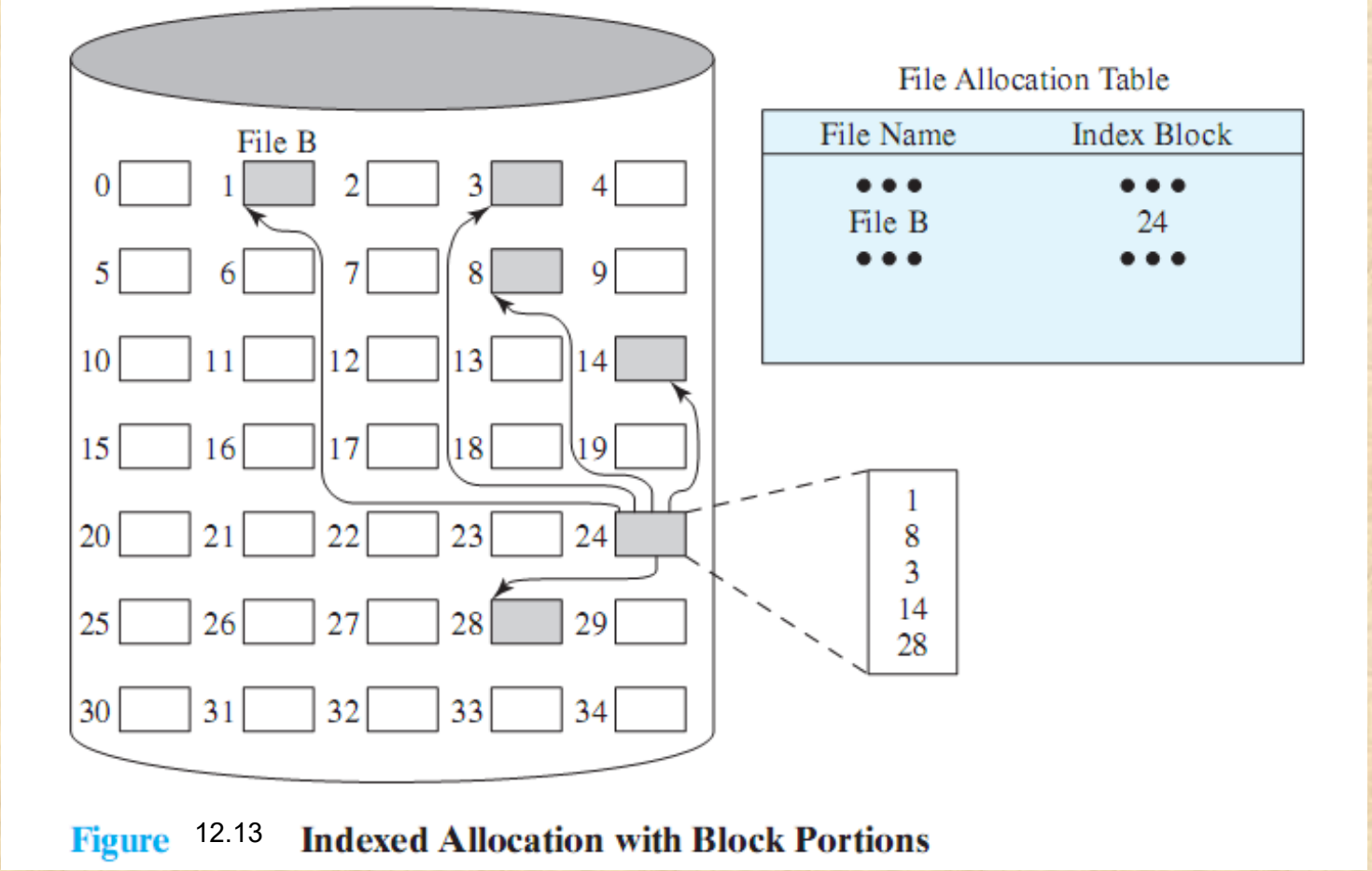


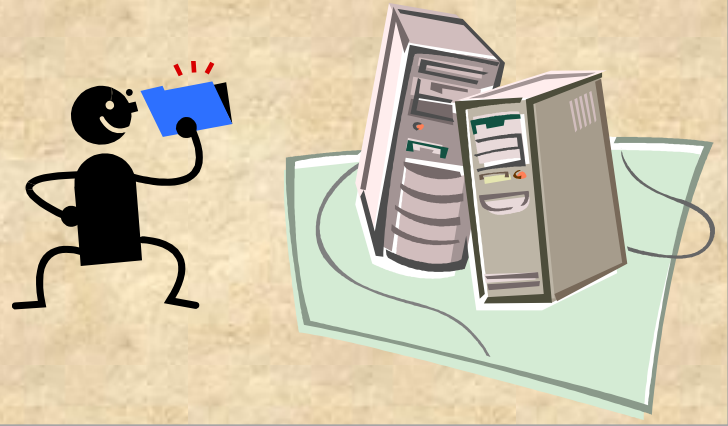
Figure 12.13 **Indexed Allocation with Block Portions**

Pregled

- Sistemi datoteka mogu podržavati datoteke organizovane kao niz bajtova ili kao niz zapisa
- Metodi pristupa zavise od organizacije datoteke
- Smeštanje datoteka na disku može biti kontinualno, ulančano ili indeksno
- Logički blokovi datoteke preslikavaju se na jedan ili više sektora diska, čime se dobijaju fizički blokovi (delovi).
- Direktorijumi preslikavaju korisnička imena datoteka u interna imena
- Tabele alokacije datoteka preslikavaju datoteke na lokacije na disku
- Liste slobodnih blokova vode evidenciju o nealociranom prostoru.

Upravljanje slobodnim prostorom

- Kao što se mora upravljati alociranim prostorom, mora se upravljati i nealociranim
- Za alokaciju datoteka potrebno je znati koji su blokovi slobodni
- Pored tabele alokacije datoteka potrebna je i ***tabela alokacije diska***
  - Bit vektori
  - Ulančani slobodni delovi
  - Indeksiranje.
  - Lista slobodnih blokova



Bit tabele (bit vektori)

- Ovaj metod koristi vektor koji sadrži po jedan bit za svaki blok na disku
- Svaka 0 odgovara slobodnom bloku, a svaka 1 bloku koji je u upotrebi

Prednosti:

- dobro radi sa svim metodama alokacije
- minimalne je veličine

Ulančani slobodni delovi

- Slobodni delovi mogu se ulančati pomoću pokazivača i vrednosti dužine u svakom slobodnom delu
- Zanemarljiv dodatni utrošak prostora jer nije potrebna tabela alokacije diska
- Pogodan za sve metode alokacije datoteka

Nedostaci:

- dovodi do fragmentacije
- pri svakoj alokaciji bloka blok se mora prvo pročitati da bi se dobio pokazivač na novi prvi slobodni blok, pre upisa podataka u taj blok

Indeksiranje

- Slobodan prostor tretira se kao datoteka i koristi se indeksna tabela, kao kod alokacije datoteka
- Radi efikasnosti, indeks slobodnog prostora treba da se zasniva na delovima promenljive veličine, a ne na blokovima
- Ovaj pristup pruža efikasnu podršku za sve metode alokacije datoteka



Lista slobodnih blokova

Svakom bloku se redom dodeljuje broj

- lista brojeva svih slobodnih blokova čuva se u rezervisanom delu diska

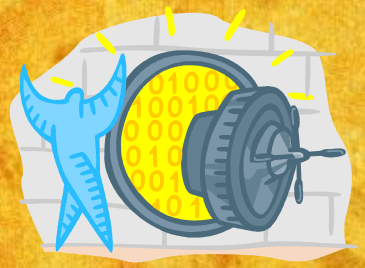
U zavisnosti od veličine diska, za čuvanje jednog broja bloka potrebna su 24 ili 32 bita

- lista slobodnih blokova je 24 ili 32 puta veća od odgovarajuće bit tabele i mora se čuvati na disku

Postoje dve efikasne tehnike za čuvanje malog dela liste slobodnih blokova u glavnoj memoriji:

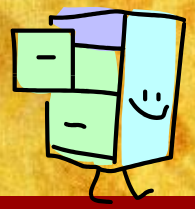
- lista se može tretirati kao stek, pri čemu se prvih nekoliko hiljada elemenata steka drži u glavnoj memoriji
- lista se može tretirati kao FIFO red, pri čemu se po nekoliko hiljada stavki sa početka i sa kraja reda drži u glavnoj memoriji

Volumeni



- Volumen je u suštini logički disk
- Skup adresibilnih sektora u sekundarnoj memoriji koje OS ili aplikacija može da koristi za skladištenje podataka
- Sektori volumena ne moraju biti uzastopni na fizičkom uređaju za skladištenje
  - dovoljno je da tako izgledaju OS-u ili aplikaciji
- Volumen može nastati sklapanjem i spajanjem manjih volumena





Rezime

- **Sistem za upravljanje datotekama:**
  - je skup sistemskog softvera koji korisnicima i aplikacijama pruža usluge za rad sa datotekama
  - obično se posmatra kao sistemski servis koji pruža operativni sistem
- **Datoteke:**
  - sastoje se od skupa zapisa
  - ako se datoteka obrađuje prvenstveno kao celina, sekvencijalna organizacija je najjednostavnija i najprikladnija
  - ako je potreban sekvencijalni pristup, ali je poželjan i proizvoljan pristup pojedinačnim zapisima, indeksno-sekvencijalna datoteka može dati najbolje performanse
  - ako se datoteci pristupa uglavnom proizvoljnim redosledom, najprikladnija može biti indeksirana ili heširana datoteka
  - servis direktorijuma omogućava hijerarhijsku organizaciju datoteka
- **Potrebna je neka strategija blokiranja**
- **Ključna funkcija upravljanja datotekama je upravljanje prostorom na disku**
  - strategija alokacije blokova diska datoteci
  - održavanje tabele alokacije diska koja pokazuje koji su blokovi slobodni