

## Dosya Sistemi

1

## Bilgilerin Uzun Vadeli Saklanması

- saklanacak veriler çok fazla olabilir
- veriler proses sonlandıktan sonra da kaybolmamalı
- bilgiye prosesler ortak olarak ulaşabilmeli

2

## Dosya Sistemi Görevleri

- dosya isimlendirme
- dosyalara erişim
- dosyaların kullanımı
- koruma ve paylaşım
- gerçekleştirme

3

## Dosya Sistemi Özellikleri

- kullanıcı açısından
  - dosyaların içerikleri
  - dosya isimleri
  - dosya koruma ve paylaşma
  - dosya işlemleri
  - ...
- tasarımcı açısından
  - dosyaların gerçekleştirilmesi
  - boş alanların tutulması
  - mantıksal blok boyu
  - ....

⇒ Dosya sistemi  
gerçekleşmesi

⇒ Kullanıcı arayüzü

4

## Dosya Tipleri

- Dosyalar
  - ASCII dosyalar
  - ikili dosyalar
- Kataloglar
  - çoğu işletim sisteminde katalog = dosya

5

## Dosya İçi Erişim

- sıralı erişim
- rasgele erişim

6

## Dosyaların Özellikleri (Attribute)

- |                             |                         |
|-----------------------------|-------------------------|
| • erişim hakları            | • kilit bayrakları      |
| • parola                    | • kayıt uzunluğu        |
| • yaratıcı                  | • anahtar konumu        |
| • sahibi                    | • anahtar uzunluğu      |
| • salt oku bayrağı          | • yaratılma zamanı      |
| • saklı bayrağı             | • son erişim zamanı     |
| • sistem bayrağı            | • son değişiklik zamanı |
| • arşiv bayrağı             | • dosya boyu            |
| • ASCII/ikili dosya bayrağı | • maksimum dosya boyu   |
| • rasgele erişim bayrağı    |                         |
| • geçici bayrağı            |                         |

7

## Dosya İşlemleri

- yaratma / silme
- isim değiştirme
- açma / kapama
- yazma / okuma / ekleme
- dosya işaretçisi konumlandırma
- özellik sorgulama / değiştirme

⇒ sistem çağrıları ile (*open, creat, read, write, close, .....*)

8

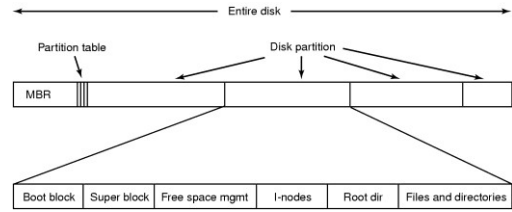


## Katalog İşlemleri

- yaratma / silme
- kataloğu açma / kapama
- kataloğu okuma
  - örneğin dosyaların listelenmesi
  - okumadan önce açılması lazım
- isim değiştirme
- bağla / kopar (link/unlink)
  - UNIX'te dosya silmeye özdeş

13

## Dosya Sistemi Gerçeklemesi



Örnek Dosya Sistemi Yapısı

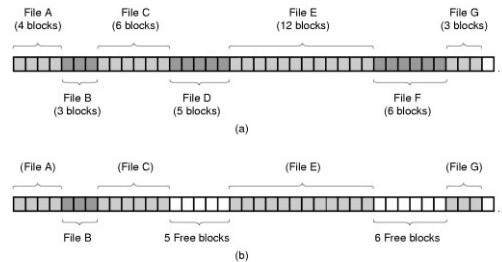
14

## Dosya Sistemi Gerçeklemesi (1)

- sürekli yer ayırma ile gerçekleştirme
  - dosyanın ilk bloğunun adresi ve blok sayısı tutulur
- avantajları
  - basit gerçekleştirme
  - daha etkin *okuma* işlemi
- sorunları
  - diskte parçalanma (fragmentation)
  - sıkıştırma maliyeti yüksek
  - boşluk listesi tutulmalı
    - dosya boyu en baştan bilinmeli ve sonradan değişemez
    - dosyaların maksimum boyları kısıtlı
- CD-ROM dosya sistemlerine uygun (tek yazımlık)

15

## Dosya Sistemi Gerçeklemesi (1)



(a) Diskte sürekli yer ayırma: örnek 7 dosya

(b) D ve F dosyaları silindikten sonra diskin durumu

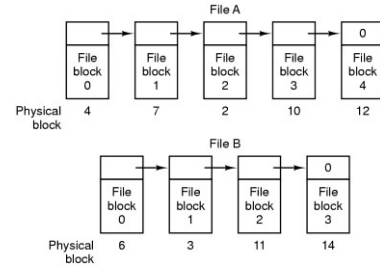
16

## Dosya Sistemi Gerçeklemesi (2)

- **bağlantılı listeler kullanarak gerçekleştirme**
  - her bloğun ilk sözcüğü bir sonraki bloğa işaretçi
  - parçalanma yok (yalnız son blokta iç parçalanma)
  - yalnız dosyanın ilk bloğunun adresi tutulur
  - dosyadaki verilere erişim
    - sıralı erişim kolay
    - rasgele erişim zor-zaman alır
  - bloktaki veri boyu 2'nin kuvveti olmayabilir
    - okumada bloklar genelde 2'nin kuvveti boyunda

17

## Dosya Sistemi Gerçeklemesi (2)



Dosya bloklarının bağlantılı liste yapısında tutulması

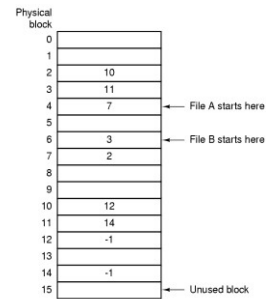
18

## Dosya Sistemi Gerçeklemesi (3)

- **bellekte dosya tabloları ile gerçekleştirme**
  - işaretçiler diskteki bloklarda değil bellekte tabloda tutulur
  - FAT (File Allocation Table)
  - rasgele erişim daha kolay
    - tablo bellekte-erişim hızlı
  - başlangıç bloğunun bilinmesi yeterli
  - tüm tablo bellekte yer almalı-bellekte alan kaplar
  - tablo boyu disk boyuna bağlı
    - örnek: 20 GB disk ve blok boyu 1K olsun: tabloda 20 milyon en az 3 sekizli boyunda kayıt gerekli (20MB)

19

## Dosya Sistemi Gerçeklemesi (3)



Bellekte dosya tablosu tutularak gerçekleştirme

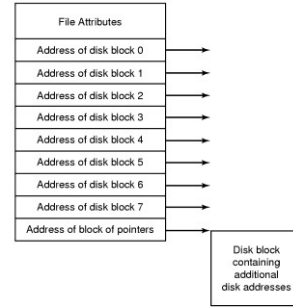
20

### Dosya Sistemi Gerçeklemesi (4)

- her dosyaya ilişkin bir i-node (index-node)
  - dosyanın özellikleri
  - dosyanın bloklarının disk adresleri
- sadece açık dosyaların i-node yapıları bellekte tutulur
  - toplam bellek alanı bir anda açık olmasına izin verilen maksimum dosya sayısı ile orantılı
- basit bir yapıda dosyanın maksimum blok sayısı üzerinde kısıtlama söz konusu
  - çözüm: son gözü blok işaretçileri içeren ek tabloya işaretçi

21

### Dosya Sistemi Gerçeklemesi (4)



Örnek i-node yapısı

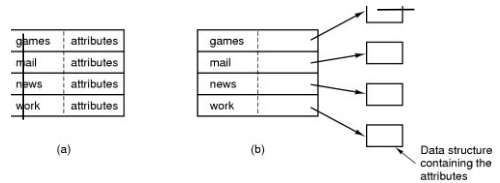
22

### Katalogların Gerçeklenmesi (1)

- dosya adı ile diskteki bloklar arasında ilişki kurulması
- dosya özellikleri nerede tutulmalı ?
  - katalog yapısı içinde ?
    - (a) katalog yapısı: sabit boyda dosya adı, dosya özellikleri, disk blokları adresleri
      - MS-DOS / Windows
    - (b) katalog yapısı: dosya adı, i-node numarası
      - dosya özellikleri i-node içinde
      - UNIX
- güncel işletim sistemlerinde dosya isimleri uzun olabilir
  - dosya ismi için sabit bir üst sınır tanımla
    - gereksiz yer kaybı

23

### Katalogların Gerçeklenmesi (1)



#### (a) Basit katalog yapısı

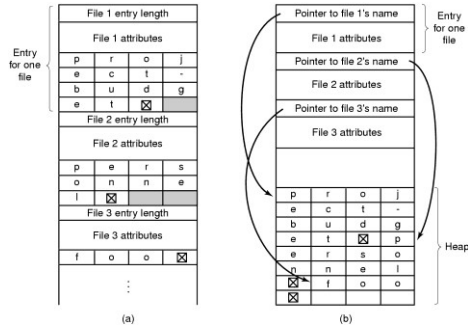
sabit uzunluklu dosya adı

disk adres ve dosya özellik bilgileri

#### (b) Her kayıt bir i-node yapısını gösterir

24

## Katalogların Gerçeklenmesi (2)



Uzun dosya isimlerinin tutulması:

- (a) sıralı
- (b) en sonda

25

## Katalogların Gerçeklenmesi (2)

- dosya isimlerinin sıralı tutulmasının sorunları
  - dosya isimleri sözcük başlarında başlamalı
    - sona boşluk ekle
  - dosya silinince değişken boylu bölgeler boşalır
    - sıkıştırmak kolay
  - bir kayıt birden fazla sayfa boyunda olabilir
    - dosya adını okurken olası sayfa hatası
- dosya isimlerinin en sonda tutulması
  - tüm dosyalar için eş boylu kayıt alanı
  - yine sayfa hatası olabilir
  - dosya isimleri sözcük başında başlamak zorunda değil

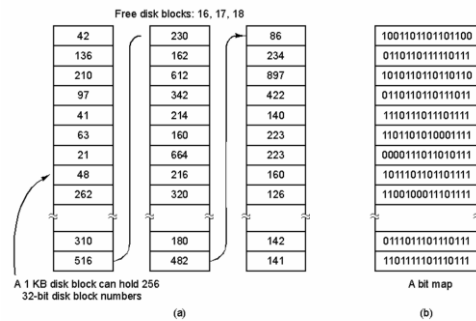
26

## Disk Alanı Yönetimi

- dosyalar sabit boylu bloklara bölünür
- blok boyu ne olmalı?
  - sektör, iz, silindir boyu?
  - aygıtla bağlı
- boy seçimi önemli
  - başarımlı ve etkin yer kullanımı çelişen hedefler
  - ortalama dosya boyuna göre seçmek iyi
  - çoğu sistemde çok önceden belirlenmiş
    - UNIX: çoğunlukla 1K

27

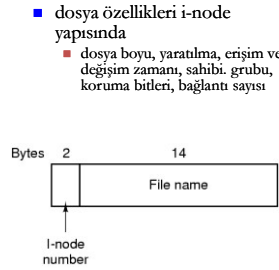
## Boş Blokların Yönetimi



28

## UNIX V7 Dosya Sistemi (1)

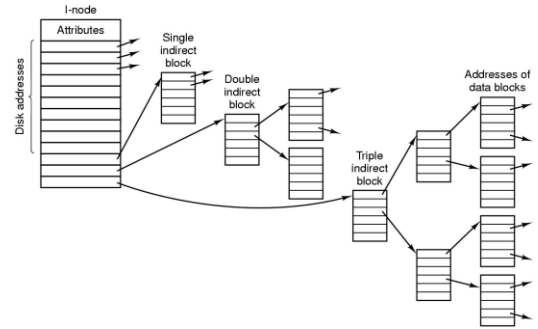
- kök katalogundan başlayan ağaç yapısı
- dosya adı max. 14 karakter
  - / ve NUL olamaz
  - NUL = 0 (isimleri 14 karaktere tamamlamak için)
- katalog yapısında dosya başına bir kayıt
  - dosya adı (14 karakter)
  - i-node numarası (2 sekizli)



UNIX V7 katalog kaydı

29

## UNIX V7 Dosya Sistemi (2)



UNIX i-node yapısı

30