

# Operációs rendszerek

ELTE IK.

© Dr. Illés Zoltán

# Mi történt a múlt héten...

- Operációs rendszerek kialakulása
  - Sz.gép – Op.rendszer generációk
- Op. Rendszer fogalma
- Fogalmak:
  - Fájlok, könyvtárak, processzek
- Rendszerhívások, rendszer struktúrák
- Rendszer struktúrák
- Háttértárak
- RAID

# Mi következik ma...

- **Fájlok**
  - **Fájltípusok**
- **Könyvtárak**
  - **Könyvtárszerkezetek**
- **Fájlrendszerek**
- **Fájlrendszer kérés ütemezések**
- **Biztonsági kérdések**
- ...

# Fájlrendszer

- Fájl: adatok egy logikai csoportja, névvel egyéb paraméterekkel ellátva.
- Könyvtár: fájlok (könyvtárak) logikai csoportosítása.
- Fájlrendszer: módszer, a fizikai lemezünkön, kötetünkön a fájlok és könyvtárak elhelyezés rendszerének kialakítására.

# Fájlok

- A fájl az információtárolás egysége.
- Névvel hivatkozunk rá.
- Jellemzően egy lemezen helyezkedik el.
  - De általánosan az adathalmaz, adatfolyam akár képernyőhöz, billentyűzethez is köthető.
- A lemezen általában 3 féle fájl, állomány található:
  - Rendes, normál felhasználói (alkalmazói) állomány.
  - Ideiglenes állomány
  - Adminisztratív állomány. Ez a működéshez szükséges, általában rejtett.

# Fájl jellemzők

- Fájlnev: Karaktersorozat
  - Operációs rendszer függvénye, hogy milyen a szerkezete(hossza, megengedett karakterek, kis-nagybetű különbözőség)
- Egyéb attribútumok (információ)
  - Mérete, tulajdonosa, utolsó módosítás ideje, rejtett (hidden) fájl-e, rendszer fájl-e, hozzáférési jogosítványok, tulajdonos,...
- Fizikai elhelyezkedés
  - Valódi fájl, link (hard), link (soft)

# Könyvtárak

- Valójában egy speciális bejegyzésű állomány, tartalma a fájlok nevét tartalmazó rekordok listája.
- Könyvtár szerkezetek
  - Katalógus nélküli rendszer, szalagos egység
  - Egyszintű, kétszintű katalógus rendszer
    - Nem igazán használt
  - Többszintű, hierarchikus katalógus rendszer
    - Fa struktúra
    - Hatékony keresés
    - Ma ez a tipikusan használt.
- Abszolút, relatív hivatkozás
  - PATH környezeti változó

# Hozzáférési jogok

- Nincs általános jogosítvány rendszer
- Jellemző jogosítványok:
  - Olvasás
  - Írás, létrehozás, törlés
  - Végrehajtás
  - Módosítás
  - Full control
- Jogok nyilvántartása
  - Attribútumként
  - ACL
    - NFS-AFS különbözőség, hasonló elv, különböző implementáció

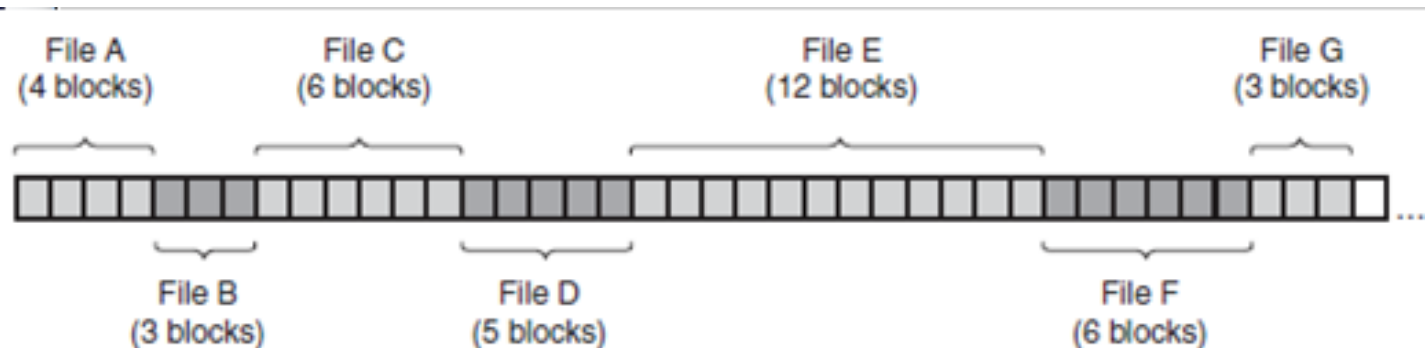


# Fájlok elhelyezése

- A partíció elején, az ún. Szuperblokk (pl. FAT esetén a 0. blokk ) leírja a rendszer jellemzőit.
- Általában következik a helynyilvántartás (FAT, láncolt listás nyilvántartás)
- Ezután a könyvtárszerkezet (inode), a könyvtár bejegyzésekkel, fájl adatokkal. (FAT16-nál a könyvtár előbb van, majd utána a fájl adatok.)
- Hova kerüljön az új fájl?
- Milyen módszert válasszunk?

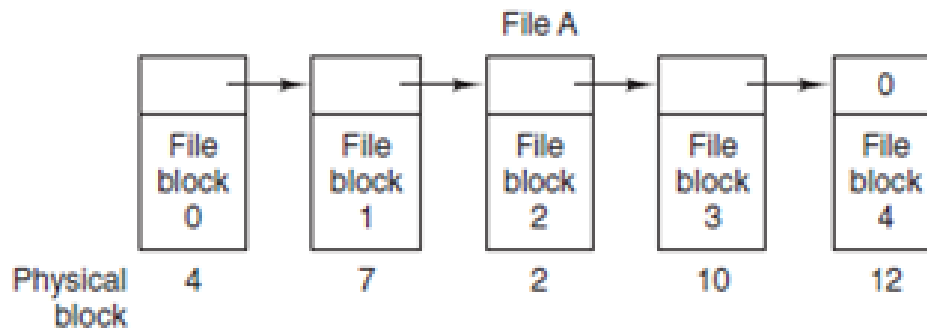
# Fájl elhelyezési stratégiák I.

- Folyamatos elhelyezésű
  - First Fit (első szabad hely, ahová befér)
  - Best Fit (arra a helyre, ahol a legkevesebb szabad hely marad)
  - Worst Fit (Arra a helyre illesztjük, ahol a legtöbb szabad hely marad)
  - Mindegyik veszteséges.



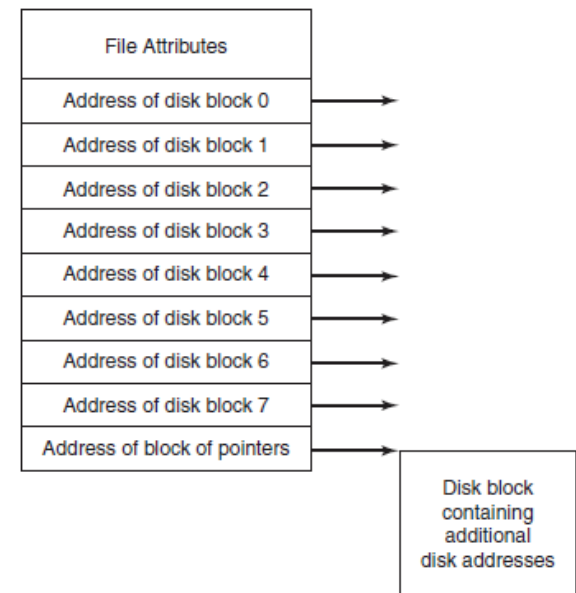
# Fájl elhelyezési stratégiák II.

- Láncolt tárolás
  - Nincs veszteség (csak a blokk méret).
  - A fájl adatai egy láncolt blokk listában vannak.
    - Az utolsó blokk elérése lassú.
  - Szabad-foglalt blokkok: File Allocation Table, FAT
    - Nagy méretű lehet és a FAT mindig a memóriában van!



# Fájl elhelyezési stratégiák III.

- Indextáblás elhelyezés
  - A könyvtár katalógus a file node-ok címét tartalmazza
  - Az inode cím mutat a fájl adatokra.



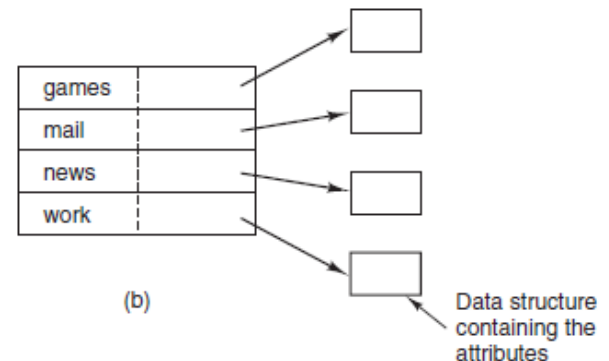
An example i-node.

# Könyvtárak megvalósítása

- A fő funkciója, hogy a névből meghatározza az adatok helyét. (keresés: lineáris, hash táblás, cache-el)
- A könyvtárbejegyzés tartalmazhatja a:
  - A címét a teljes fájlnak (folyamatos elhelyezésnél)
  - Az első blokk címét (láncolt listánál)
  - Az i-node számot.
- Hogyan tárolja az attributumokat?
  - Bejegyzések hossza azonos, rögzített
  - I-node-okban

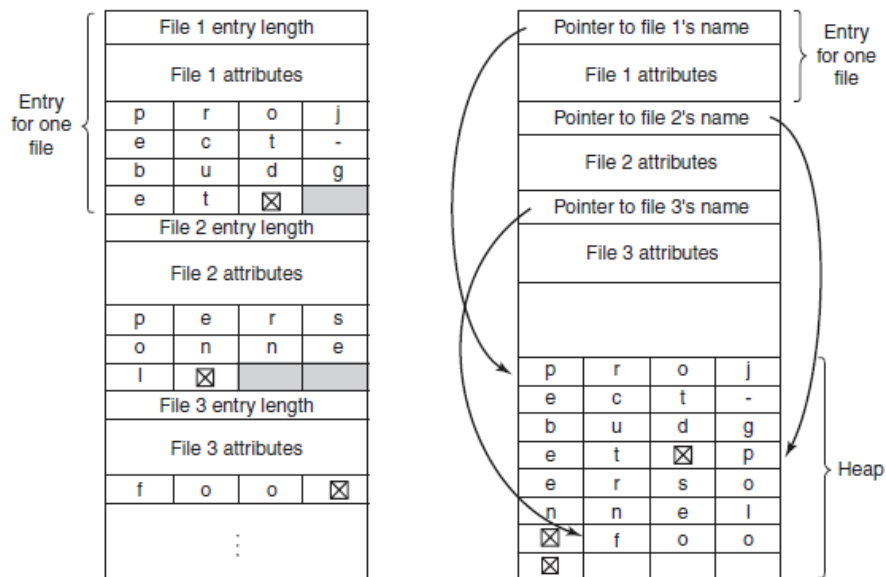
games	attributes
mail	attributes
news	attributes
work	attributes

(a)



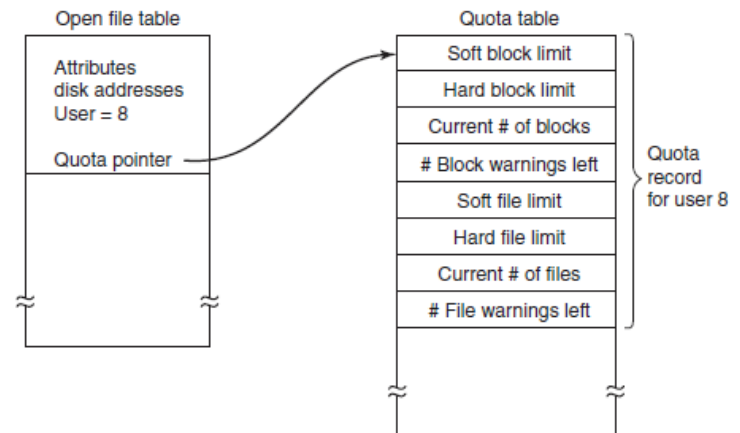
# Fájlnév tárolás

- Korábban – fix hosszúságú (8+3)
- Ma – általában max 255 karakter hosszú
  - Helytakarékos megoldások
    - Különböző hosszúságú bejegyzések.  
Az első helyen a bejegyzés hossza, majd az attribútumok, majd a név.
    - Egyforma hosszú bejegyzések, mutató a fájl névre.



# Diszk kvóták

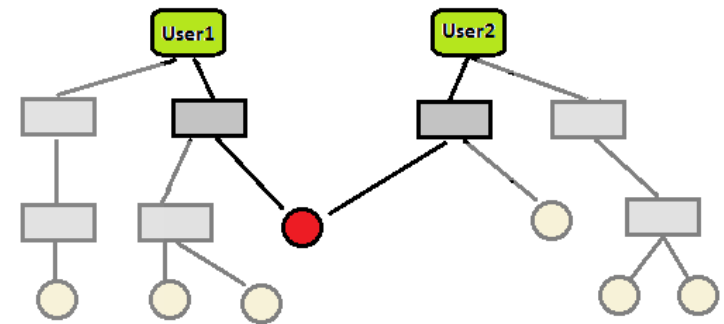
- A felhasználói felnyitáskor
  - Megnyitja a fájl táblát a memóriában
  - A kvóta táblát a memóriában



- Ha új blokkot foglal -> változik a kvóta tábla
- Bejelenkezéskor ellenőrzés
  - Szoft limit átlépésekor bejelentkezhet (kivéve túllépte a lehetséges figyelmeztetések számát)
  - Hard limit átlépésnél – be sem jelentkezhet

# Megosztott fájlok

- Megoldandó, hogy mindenki számára minden változás látszódjon!
- Két módszer használatos:
  - A fájl blokkjai egy struktúrában találhatóak és erre mutat a könyvtárbejegyzés (nem a blokkok maguk vannak felsorolva a bejegyzésben) (pl. UNIX, i-node) A user1 és user2 ugyanarra a struktúrára mutat. (Mi történik törlésnél?)
  - Új link fájl létrehozásával. (symbolic link) (lassú elérés.)





# Titkosított fájlrendszer

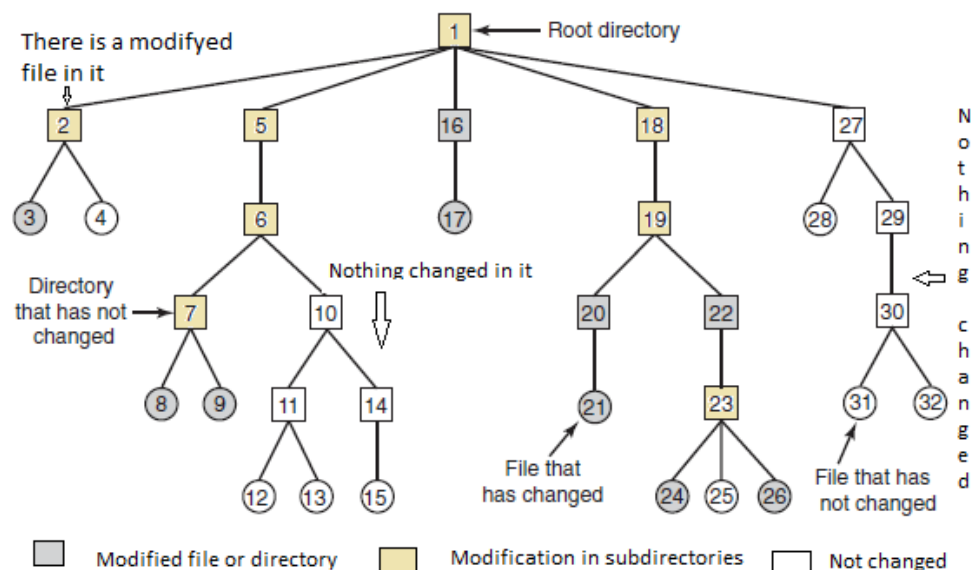
- CPU támogatás
  - AES-NI (New Instructions) utasításkészlet
  - Szimmetrikus kulcs
- Windows 2000 óta létezik NTFS-ben (EFS)
- Windows Jelenleg: Bitlocker
  - Beépített win10 prof.
- Checkpoint Pointsec- független titkosító
- Linux – dm-crypt
  - 2.6.4 kernel óta, alapértelmezett titkosítás

# Mentések

- Fizikai mentés(Full backup) (mindent lemásol)
  - Előny (simple, quick)
  - Hátrány (felesleges mentés, pl. szabad blokkok..)
- Logikai mentés (csak a módosítottat menti)
  - Előny (Igény esetén egy adott fájl vagy könyvtár is visszaállítható nemcsak az egész)
  - Hátrány (bonyolult algoritmus)
- Vegyes használat
  - Időnként fizikai mentés
  - Sűrűbben logikai mentés
- Visszaállítás (fizikai mentés, első logikai mentés, ...., utolsó logikai mentés)

# Logikai mentés - algoritmus

- A UNIX rendszerekben gyakran használt algoritmus
  - Minden módosított fájl és minden könyvtár megjelölése
  - Eltávolítjuk azokat a könyvtár jelöléseket, amelyekben (vagy amelyek alkönyvtáraiban) nem volt módosítás.
  - Mentjük a megjelölt könyvtárakat attribútumokkal).
  - Mentjük a megjelölt fájlokat attribútumokkal.



# A logikai mentések kérdései

- A szabad blokk lista nem fájl (nincs mentve).  
(Visszaállítható, hiszen a komplementere a foglaltaknak.)
- Linkek – minden könyvtárban, ahol a módosított állományra volt link, azt vissza kell állítani.
- A fájlok lyukakat tartalmazhatnak (pl. seek+write) – Visszaállításkor a nem használt helyeket nem kell lefoglalni!
- Nem valódi fájlokat pl. nevesített csővezetékeket nem kell menteni.

# Fájl, könyvtár műveletek

- Fájl
  - Megnyitás
  - Műveletek: Írás, olvasás, hozzáfűzés
  - Lezárás
- Adatok
  - Bináris- bájt sorozat
  - Szöveges- karakter sorozat
- Elérés módja, szekvenciális, random
- Könyvtár műveletek
  - Létrehozás, tartalom listázása, állomány törlés

# Fájrendszer típusok

- Merevlemezen alkalmazott fájlrendszer
  - FAT, NTFS, EXT2FS, XFS, stb.
- Szalagos rendszereken (elsősorban backup) alkalmazott fájlrendszer
  - Tartalomjegyzék, majd a tartalom szekvenciálisan
- CD, DVD, Magneto-opto Disc fájlrendszere
  - CDFS, UDF (Universal Disc Format), kompatibilitás
- RAM lemezek (ma már kevésbé használtak)
- FLASH memória meghajtó (FAT32)
- Hálózati meghajtó
  - NFS, AFS
- Egyéb pszeudó fájlrendszerek
  - Zip, tar.gz, ISO

# Naplózott fájlrendszerek

- Fájlrendszer sérülés, áramszünet stb. esetén inkonzisztens állapotba kerülhet.
- Gyakran nevezik: LFS-nek (Log-structured File System) vagy JFS-nek(Journaled)
- Adatbázis kezelők mintájára: művelet + log
  - Tranzakciós alap
  - Leállítás, hiba esetén a log alapján helyre lehet állítani.
  - Célszerűen a log másik lemez (másik partíció)
- Nagyobb erőforrás igény- nagyobb megbízhatóság
- Adatmódú – Metaadat alapú naplózás

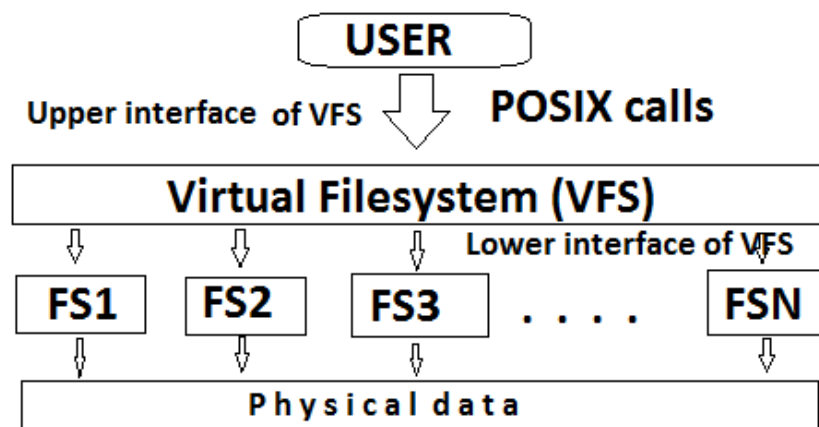
# Fájlrendszer támogatás

- Mai operációs rendszerek „rengeteg” típust támogatnak
  - PL: Linux 2.6 kernel több mint 50-et.
- Fájlrendszer csatolása
  - Mount, eredményeképpen a fájlrendszer állományok elérhetők lesznek.
  - Automatikus csatolás (pl. USB drive)
  - Kézi csatolás (Linux, mount parancs)
- Külön névtérben való elérhetőség (Windows)
  - A,B,C,...
- Egységes névtér (UNIX)



# Különböző fájlrendszerek együttes használata egy gépen

- Lehetséges több különböző fájlrendszert használni ugyanazon a gépen.
  - Pl. Windows: NTFS, FAT-32, UDF (DVD,CD) stb.
    - A rendszer egymástól függetlenül használja őket.  
Az hogy éppen melyiket kell használni, a drive betű határozza meg. (c:, a:)
  - Pl.:UNIX: ext2,ext3, UDF, stb.
    - A modern UNIX rendszerek egységesen kezelik létrehozva egy közbülső réteget a virtuális fájlrendszert! (VFS)



# Alkalmazás- Diszk kapcsolat

- Réteges felépítés
  - Alkalmazói szint
    - Az alkalmazás, fejlesztői könyvtárak segítségével megoldja a lemezen tárolt adatok írását-olvasását.
    - Szöveges, bináris fájlműveletek
  - Operációs rendszer szint
    - Fájrendszer megvalósítás
      - Elérhetőség, jogosultságok
    - Kötetkezelő (Volume manager)
    - Eszközmeghajtó (device driver)
      - BIOS-ra alapozva
  - Hardver eszköz szintje
    - I/O meghajtó, IDE, SATA stb.

# FAT

- File Allocation Table
  - Talán a legrégebbi, ma is élő fájlrendszer!
- A FAT tábla a lemez foglaltsági térképe, annyi eleme van, ahány blokk a lemezen
  - Pl: Fat12, FDD, Cluster méret 12 bites. Ha értéke 0, szabad, ha nem foglalt.
  - Biztonság kedvéért 2 tábla van.
- Láncolt elhelyezés
  - A katalógusban a file adatok (név stb.) mellett csak az első fájl blokk sorszáma van megadva.
  - A FAT blokk azonosító mutatja a következő blokk címét.
  - Ha nincs tovább, FFF az érték.
- Rögzített bejegyzés méret, 32 bájt (max. 8.3 név)
- System,Hidden,Archive,Read only, könyvtár attribútumok
- A fájl utolsó módosítás ideje is tárolva van.

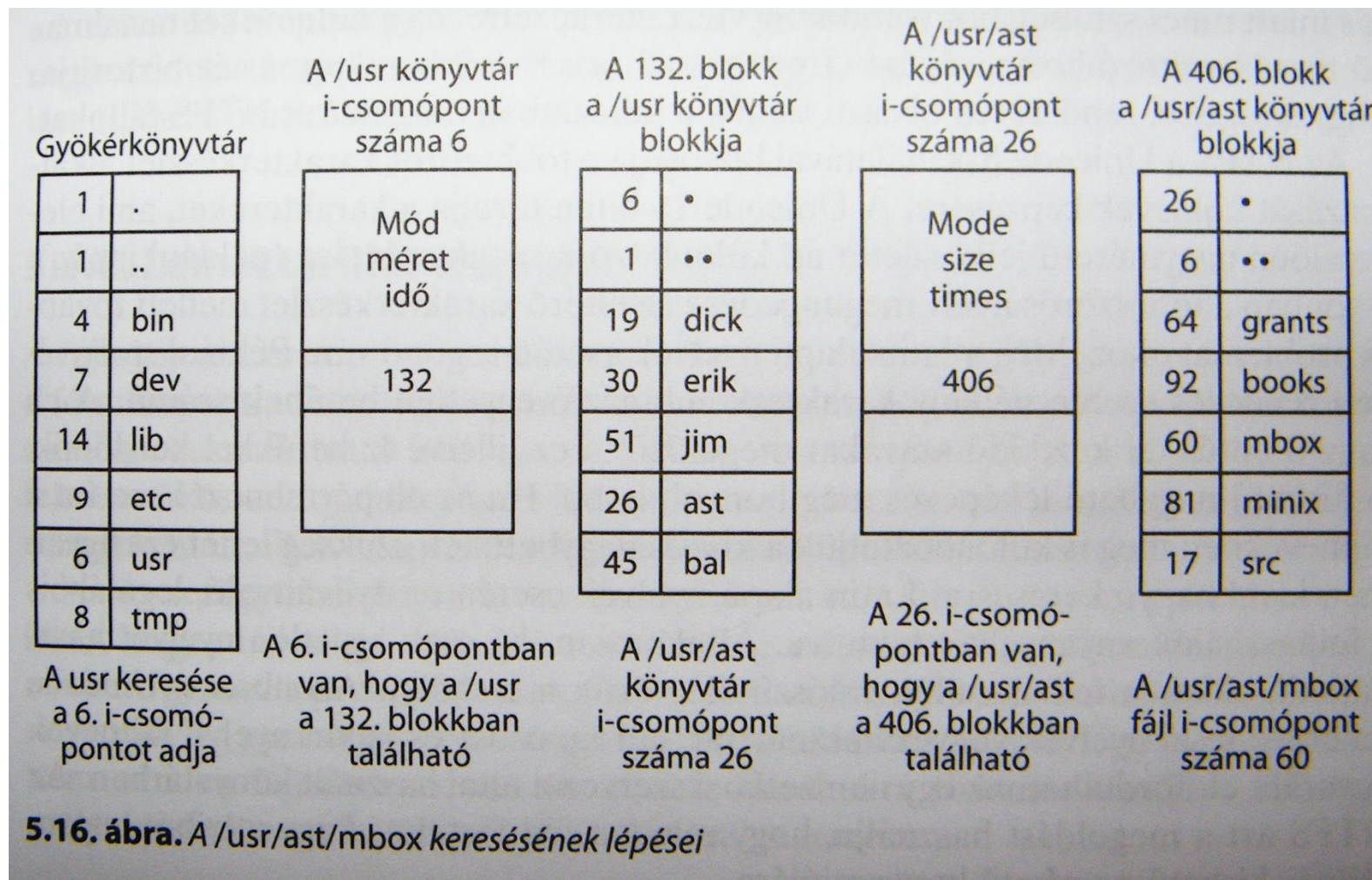
# FAT jellemzők

- FAT16, 16 bites cluster leíró, 4 bájt (2x2) írja le a fájl kezdőblokkját
  - Max. 4 GB partíciós méret (64kb blokk méretnél), jellemzően 2 GB.
  - Fájl méret maximum is a 4 (2) GB.
  - Külön könyvtári terület (FDD-n ez a 0. sáv)
    - FDD-n 512 könyvtári bejegyzés
    - HDD-n 32736 könyvtári bejegyzés (16 bit előjelesen)
- FAT32 (1996-tól elérhető)
  - 28 bites cluster leíró
  - 2 TB partíciós méret (alap szektor mérettel)
- 32MB-ig, 1 cluster = 1 blokk(512bájt)
  - 64 MB, 1 cluster=1KB (2 blokk), 128MB, 1 cluster=2KB
  - 1 cluster max. 64 KB lehet.
- Támogatták már a hosszú fájl neveket is
  - Többszörös 8.3 részre fenntartott bejegyzésekkel.
- Töredezettség mentesítés szükséges.

# UNIX könyvtárszerkezet

- Indextáblás megoldás
- Boot blokk után a partíció superblokkja (fájlrendszer paraméterek)
- Ezt követi a szabad terület leíró rész.
- i-node tábla, majd gyökérkönyvtár bejegyzéssel)
- Moduláris elhelyezés, gyorsan elérhető az információ, sok kicsi táblázat, ez alkotja a katalógust.
- Egy fájl egy i-node ír le!
  - 15 rekeszből áll, első 12 a fájl blokkokra mutat.
  - Ha kevés, a 13. rekesz újabb i-node-ra, ami +15 rekesz.
  - Ha ez is kevés, a 14. rekesz újabb i-node-ra ami az első mintáját ismétli.

# i-node láncolás példa

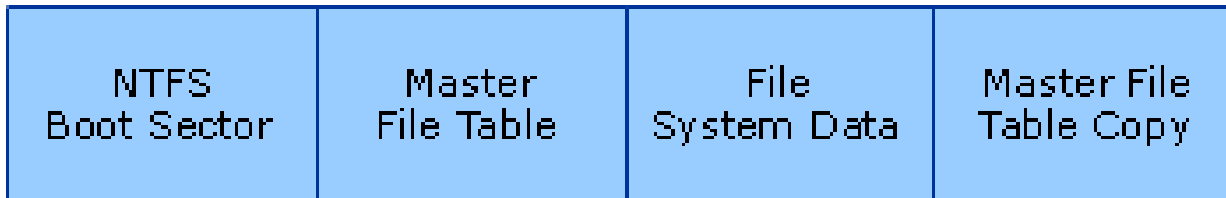


# NTFS

- New Technology File System
  - FAT-NTFS hatékonysági határ: kb. 400 MB.
- 255 karakteres fájl név, 8+3 másodlagos név
- Kifinomult biztonsági beállítások
- Ahogy a FAT esetén, itt is szükséges a töredezettség mentesítés.
- Titkosított fájlrendszer támogatása, naplózás
- POSIX támogatás
  - Hard link (fsutil hardlink parancs), időbélyegek, kis-nagybetűk különböznek
- Tömörített fájl, mappa, felhasználói kvóta kezelés
- Az NTFS csak klasztereket tart nyilván, szektort (512bájtt) nem

# NTFS partíció felépítése

- A Master File Table egy táblázat.
- A Files System Data szintén





# NTFS partíció Boot szektor

- Boot sector
  - JMP +0x52 (EB 52)
  - OEMID (8 byte, MSWINx.y)
  - BPB (Bios Paraméter Blokk)
    - Bytes per sector (512)
    - Sectors per cluster (8)
  - Extended BPB
    - Total Sector number (8 byte-on tárolva)
    - LCN – Logical Cluster Number for MFT
    - Volume serial number
  - Betöltő kód (betölti az ntldr.dll-t, majd tovább az ntfs.sys,ntoskrnl.exe)
  - Sector end(0xAA55)

# MFT

- NTFS partíció az MFT (Master File Table) táblázattal kezdődik
  - 16 attribútum ad egy fájl bejegyzést.
  - Minden attribútum 1kb. Ha ez nem elég akkor egy attribútum mutat a folytatásra.
  - Az adat is egyfajta attribútum, így egy bejegyzés több adatsort tartalmazhat. (PL: Betekintő kép)
  - Elvi fájl méret  $2^{64}$  bájt lehet
  - Ha a fájl  $< 1\text{kb}$ , belefér az attribútumba, közvetlen fájl.
  - Nincs fájl méret maximum.

# Az NTFS partíció felépítése

0	\$Mft – Master File Table
1	\$MftMirr – MFT Mirror
2	\$LogFile – Naplófájl
3	\$Volume – Kötetfájl
4	\$AttrDef – Attribútum definíciók
5	\ – Gyökérkönyvtár
6	\$BitMap – Cluster foglaltság
7	\$Boot – Bootszektor
8	\$BadClus – Hibás clusterek
9	\$Secure – Biztonsági leírók
10	\$UpCase – Unicode karaktertábla
11	\$Extend – Egyéb metadata
12	Nem használt
...	...
15	Nem használt
16	Felhasználói fájlok és mappák

**Az NTFS metadata számára fenntartva**

Köszönöm a figyelmet!

