

# UNIX: fájlrendszerek

kiegészítő fóliák az előadásokhoz

*Mészáros Tamás*

<http://home.mit.bme.hu/~meszaros/>

Budapesti Műszaki Egyetem  
Méréstechnika és Információs Rendszerek Tanszék

# Alapfogalmak

- Fájl (állomány, nem „file”)
  - adattárolási hely
- Fájlrendszer (állományrendszer)
  - fájlok tárolásának szervezése, hozzáférés biztosítása
- Fájlrendszerek felhasználói felülete
  - programozói (API, rendszerhívások)
  - parancssori (grafikus)
- Fájlrendszerek szervezési felülete
  - diszk szervezés

# Történeti áttekintés

- System V első fájlrendszer *s5fs*
  - 80-as évek, alap implementáció, egyszerű szervezés
- 4.2 BSD Fast File Systems *FFS (Linux ext2)*
  - megnövelt teljesítmény
  - új szolgáltatások
  - akkori diszk hardver felépítéshez optimalizált rendszer
- Virtuális fájlrendszerek *vnode/vfs*
  - moduláris, objektum-orientált
  - cserélhető szervezési modulok, akár hálózati is
- Elosztott fájlrendszerek
  - NFS: transzparens hálózati fájlrendszer RPC megvalósítással
- Modern fájlrendszerek
  - ext3, ext4, xfs, ReiserFS, Solaris ZFS
  - gnome-vfs (felhasználói címterű) vfs: ftp, smb, dav, stb. protokollokkal
  - Klaszter fájlrendszerek, pl. Red Hat GFS

# A fájlrendszer felhasználói szemmel

- Operációs rendszer felhasználó (végfelhasználó)
  - parancssori és grafikus felület
  - könyvtárszervezés, speciális könyvtárak
  - fájlok és könyvtárak kezelése, attribútumaik
  - fájlrendszerek menedzselése (rendszergazda)
- Programozó (alkalmazás fejlesztő)
  - programozói interfészek (rendszerhívások, rendszerkönyvtárak)
  - fájlleírók, nyitott fájl objektumok és kezelésük
  - zárolási módszerek: kötelező, ajánlott

# Felhasználói interfész

- Diszkek, partíciók, fájlrendszerek
- Tipikus UNIX könyvtárszerkezet
- Fájlrendszerek csatlakoztatása a könyvtárszerkezethez
- Fájl attribútumok
  - típus (- d p l b c s)
  - linkek (hard, szoft)
  - eszköz, inode, méret, stb.
  - időbélyegek
  - azonosítási és hozzáférés-szabályozási adatok
  - listázási parancs: `ls -la`

```

-rw-r--r--  2 root root      189 sze  8  2006 /etc/hosts
-rwxr-xr-x  1 root root 616920 nov 17  01:29 /bin/bash
srwxr-xr-x  1 clamilt clamilt 0 ápr 23  10:16 clamav.sock
crw-rw----  1 root tty      4, 0 ápr 20  2007 /dev/tty0
-r-s--x---  1 root apache 10760 jan 14  14:22 suexec
          
```

# Gyakorlatok

- Csatolt fájlrendszerek: `df`, `mount`, `umount`
- A UNIX könyvtárstruktúra felépítése: `cd`, `pwd`, `ls`
- Fájlok attribútumai: `ls -la`, `ls -lZ`

# Programozói interfész

- Fájlok megnyitása (létrehozása)
  - *open()* rendszerhívás és paraméterei
  - a fájlleíró és a nyitott fájl objektum
  - fájlok többszörös megnyitása, a *fork()* hatása
- Írás és olvasás: *read()*, *write()*
- Fájlok zárolása
  - kötelező (mandatory): *fcntl()*, *lockf()*
  - ajánlott (advisory): *flock()*
- Fájlok lezárása: *close()*

# Fájlrendszerek szervezése

- Csatlakoztatás az elő fájlrendszerhez
  - csatlakoztatási pont
  - elfedés
- Szervezés a háttértáron
  - blokkos tárolás
  - fájlok leírói (diszk inode)
  - szabad helyek kezelése
- Szervezés a memóriában
  - csatlakoztatás nyilvántartása
  - fájlok leírói (memória inode)
  - kapcsolat a nyitott fájl objektumokhoz



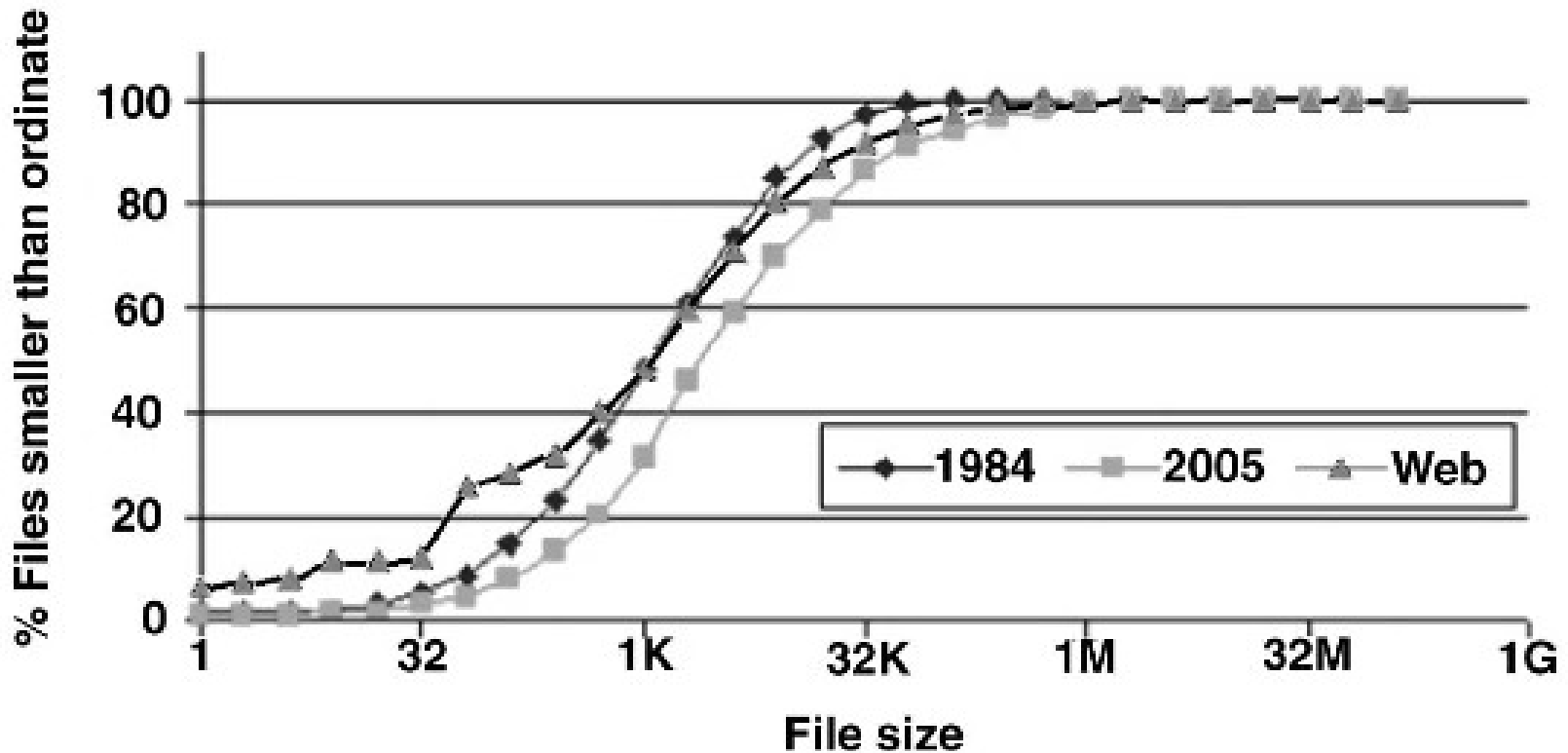
# A tárolás megvalósítása

- A diszken elhelyezett fájlrendszer részei
  - szuperblokk (fájlrendszer metaadatok)
  - inode lista (fájl metaadatok)
  - tárolt adatok
- Szuperblokk
  - a fájlrendszer mérete
  - szabad blokkok jegyzéke
  - zárolási információk
  - módosítás jelzőbit

## Az index node (inode)

- hitelesítési információk (UID, GID)
- típus
- hozzáférési jogosultságok
- időbélyegek
- méret
- adatblokkok elhelyezkedése (címtábla)
  - 10-15 db direkt blokkcím
  - 1x, 2x és 3x indirekt blokkcímek(számítási példa)

# Címzés és tipikus fájlméretek



Andrew S. Tanenbaum, Jorrit N. Herder, Herbert Bos: File size distribution on UNIX systems: then and now. Operating Systems Review 40(1): 100-104 (2006)

## Az inode a memóriában

- a nyitott fájl objektumhoz kapcsolódik
- diszk inode tartalma bekerül a memóriába
- az aktív használat információival bővül
  - státusz (zárolt, módosított, stb.)
  - háttértár eszköz (fájlrendszer) azonosítója
  - hivatkozás számláló (fájlleírók)
  - csatlakoztatási pont adminisztrációja
  - ...

## Allokáció a diszken

- Szempontok: teljesítmény, megbízhatóság
- Cilinder (blokk) csoport (FFS, ext2, ...)
- Allokációs elvek
  - szuperblokk másolása minden csoportba
  - inode lista és szabad blokkok csoportonként kezelve
  - egy könyvtár – egy csoport
  - kis fájlok egy csoportba
  - nagy fájlok „szétkenve” több csoportba
  - új könyvtárnak egy új, kevésbé foglalt csoportot keres

# A virtuális fájlrendszer

- Implementáció-független fájlrendszer absztrakció
- Célok:
  - többféle fájlrendszer egységes egyidejű támogatása
  - egységes kezelés a csatlakoztatás után (programozó IF)
  - speciális fájlrendszerek (hálózati, processz, stb.)
  - modulárisan bővíthető rendszer
- Absztrakció
  - inode       $\longrightarrow$       vnode
  - fs          $\longrightarrow$       vfs

## A vnode absztrakció

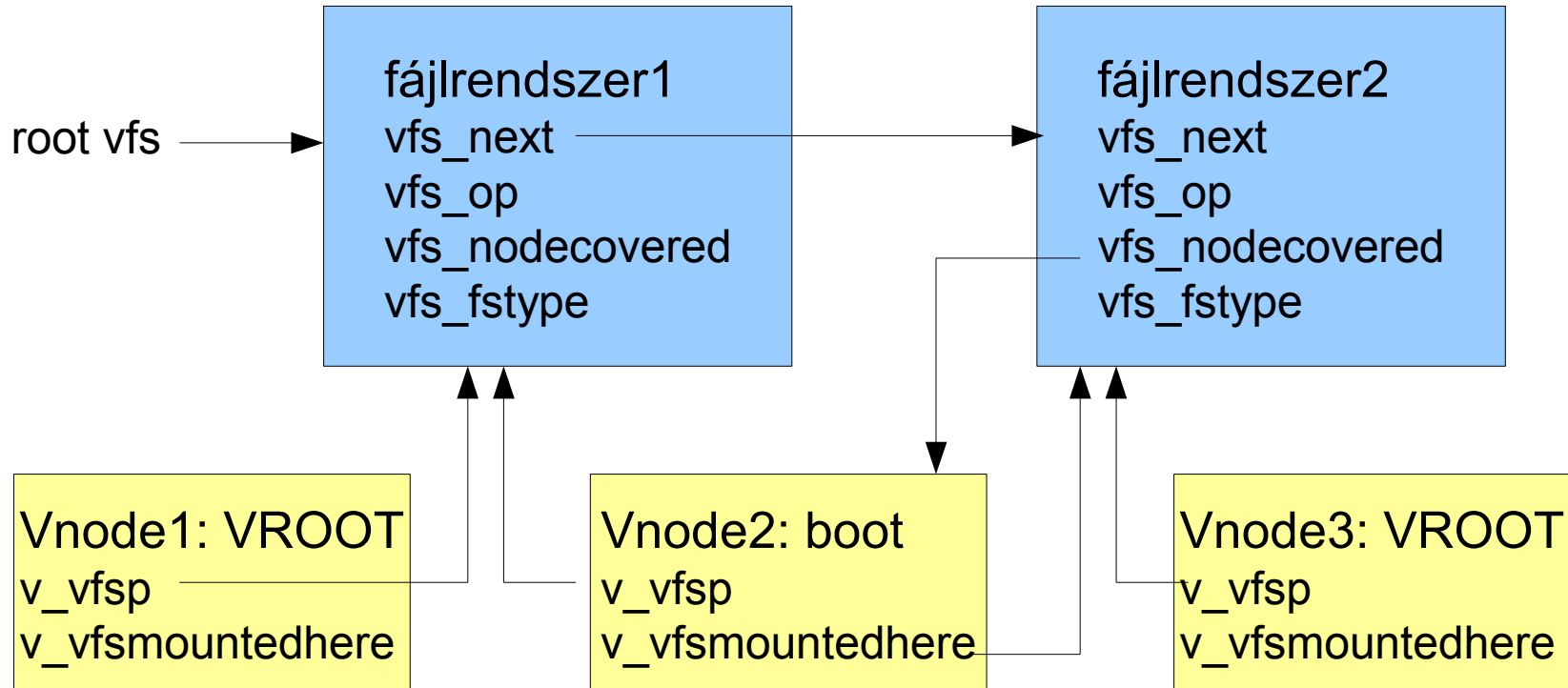
- adatmezők
  - közös adatok (típus, csatlakoztatás, hivatkozás száml.)
  - `v_data`: állományrendszertől függő adatok (inode)
  - `v_op`: az állományrendszer metódusainak táblája
- virtuális függvények
  - állományrendszertől független: `vop_open`, `vop_read`,...
  - a tényleges metódusokra helyettesítődnek be
- segédrutinok, makrók

## A vfs absztrakció

- adatmezők
  - közös adatok (fájlrendszer típus, csatlakoztatás, hivatkozás, `vfs_next`)
  - `vfs_data`: állományrendszertől függő adatok
  - `vfs_op`: az állományrendszer metódusainak táblája
- virtuális függvények
  - állományrendszertől független: `vfs_mount`, `vfs_umount`, `vfs_sync`,...
  - a tényleges metódusokra helyettesítődnek be
- segédrutinok, makrók

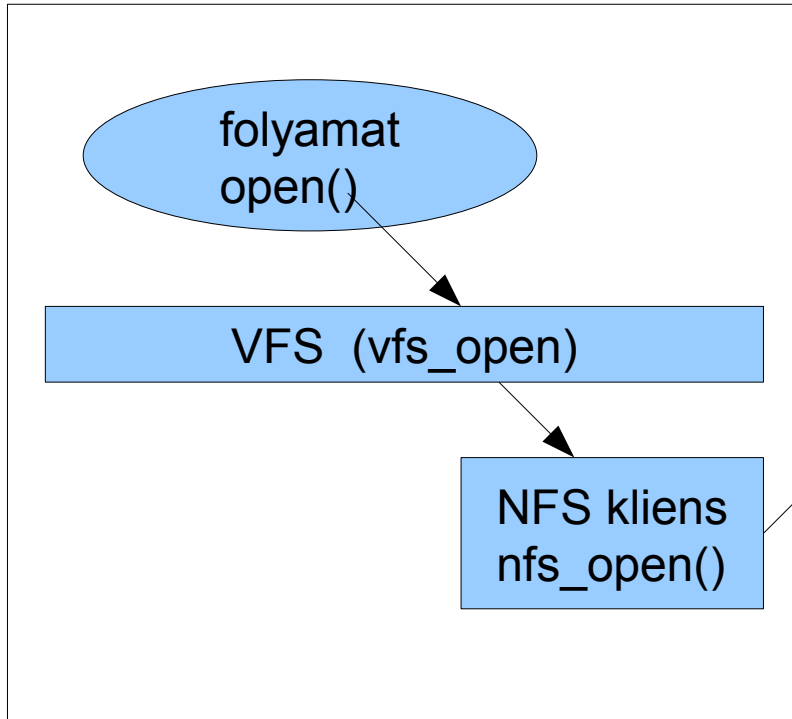


# A vfs és a vnode kapcsolata

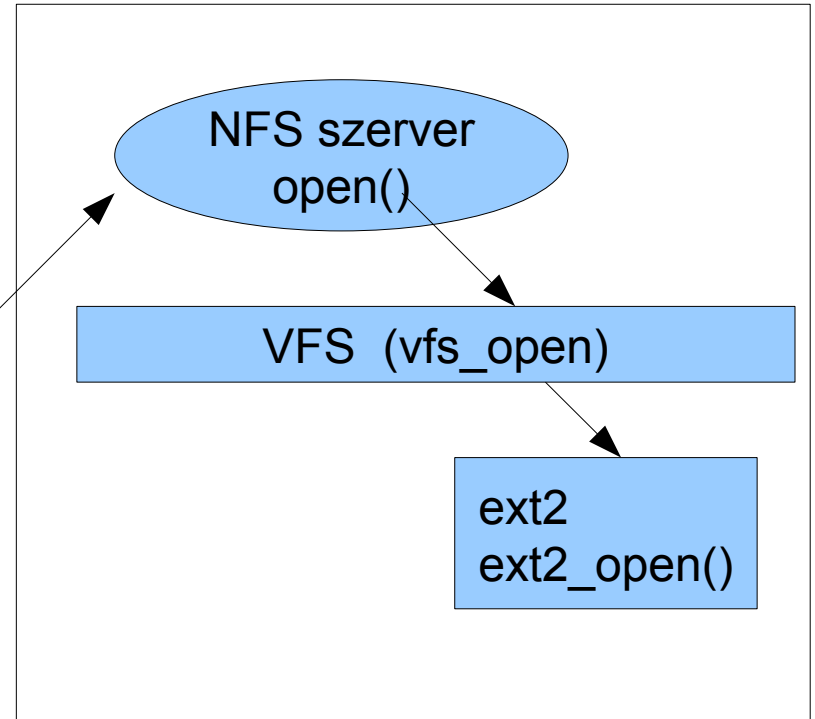


# Alkalmazási példa: NFS egyszerűsített felépítés

szgép 1



szgép 2



RPC