Operációs Rendszerek MSc

Operációs Rendszerek
MSc

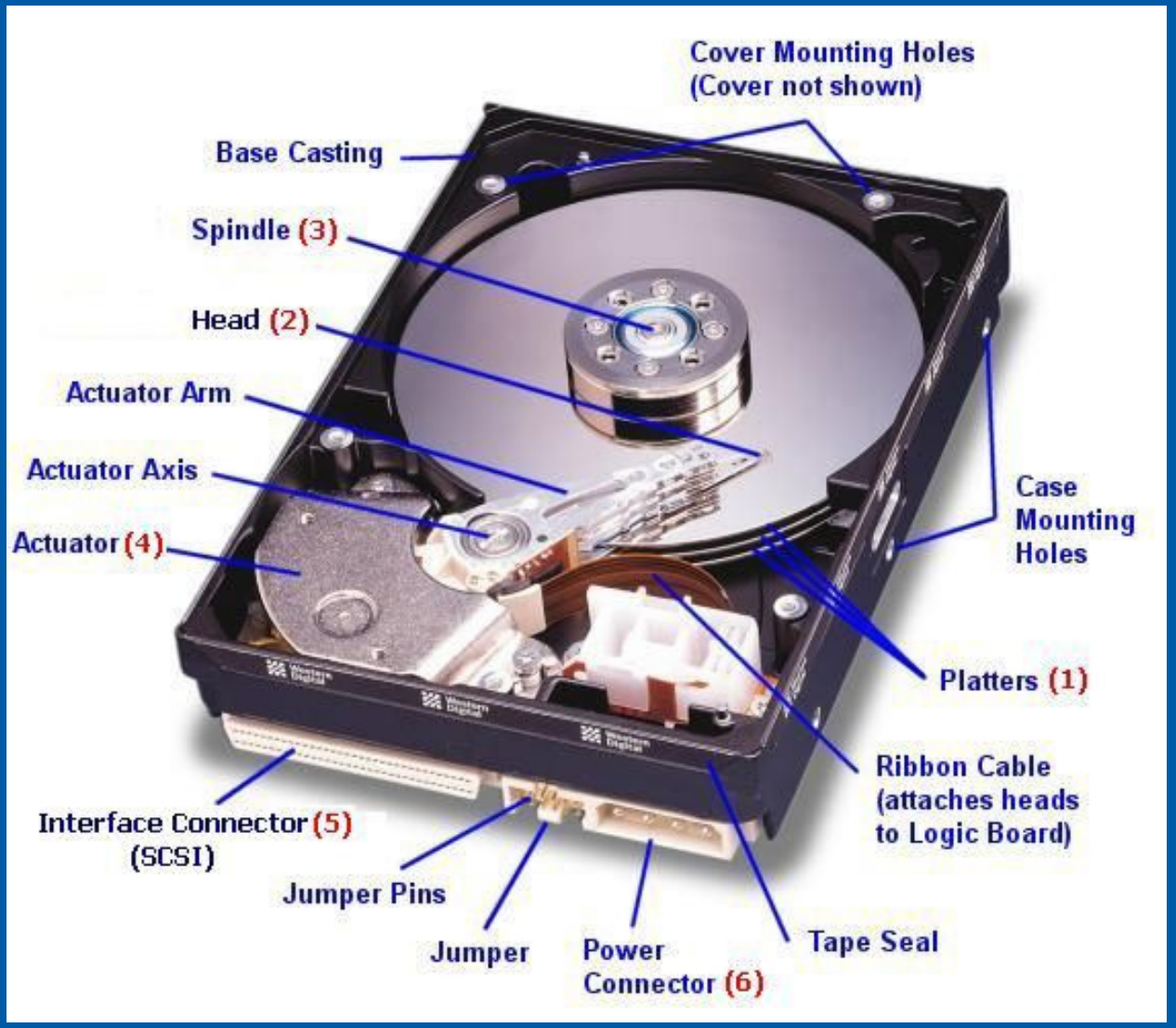
Modern fájlrendszerek

2019/2020/I.

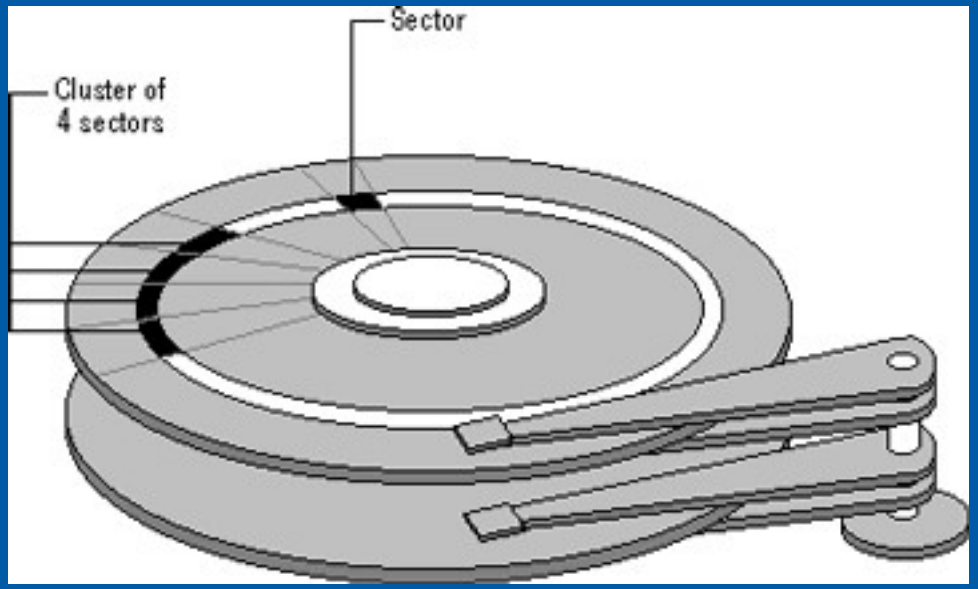
Dr. Vincze Dávid
Miskolci Egyetem, IIT
vincze.david@iit.uni-miskolc.hu

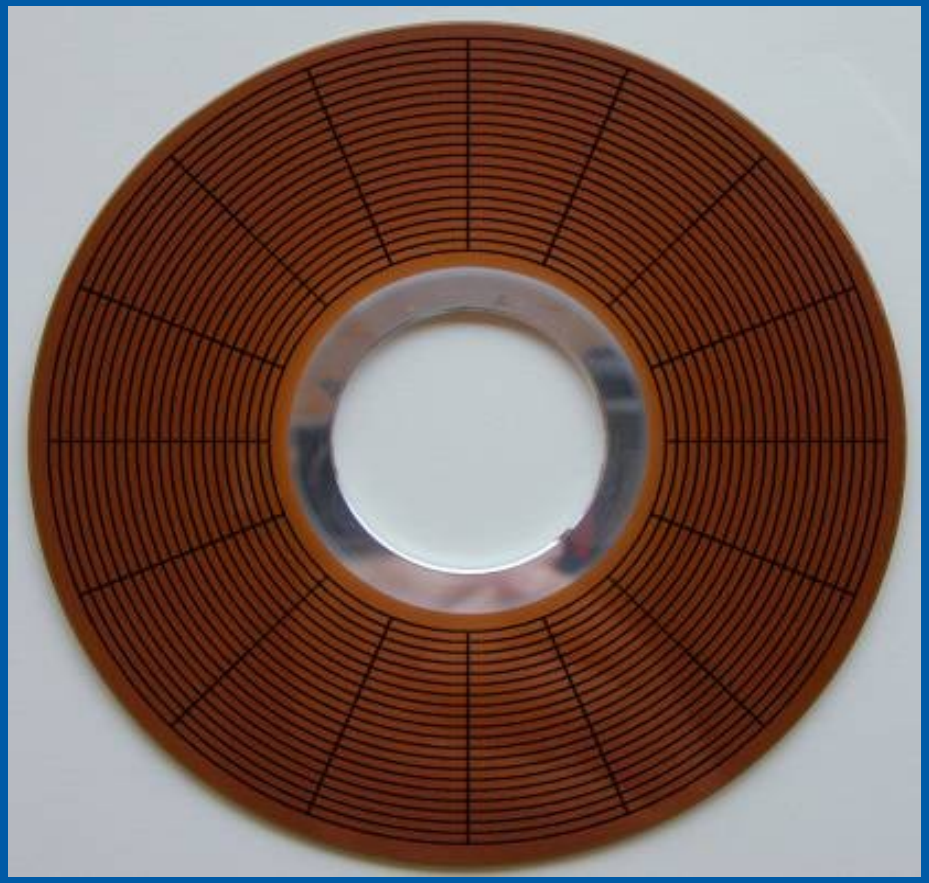
Operációs Rendszerek MSc

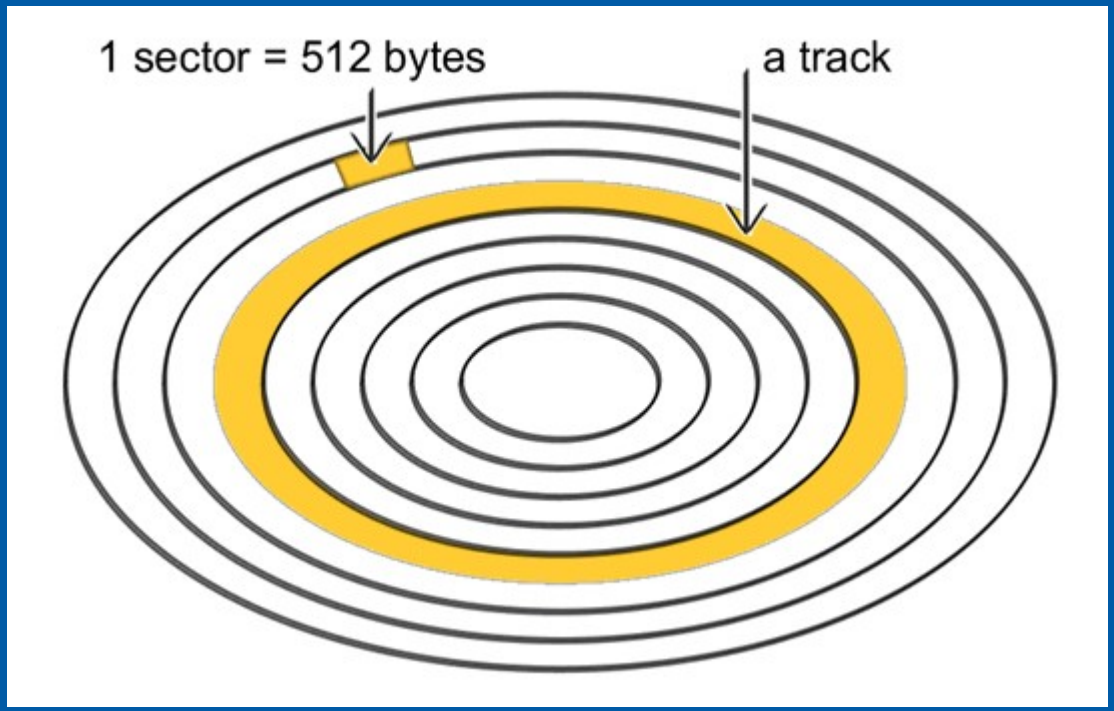
- Fájlrendszerek
  - Miért?



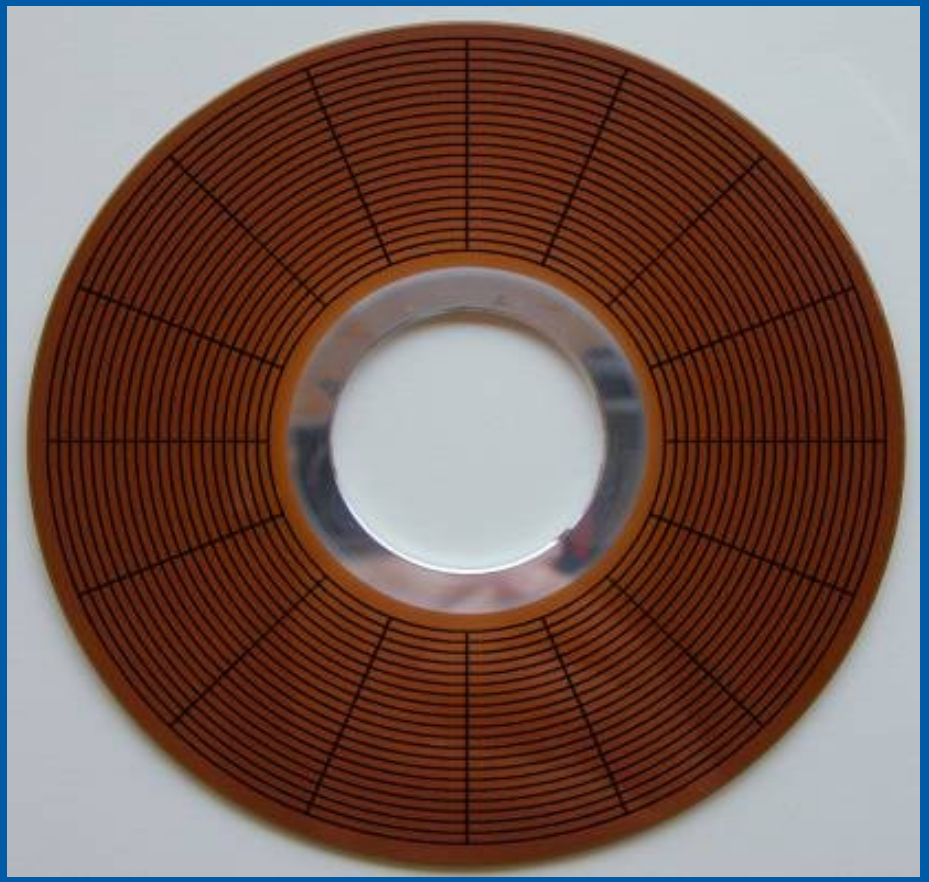
https://coub.com/view/zbc2v

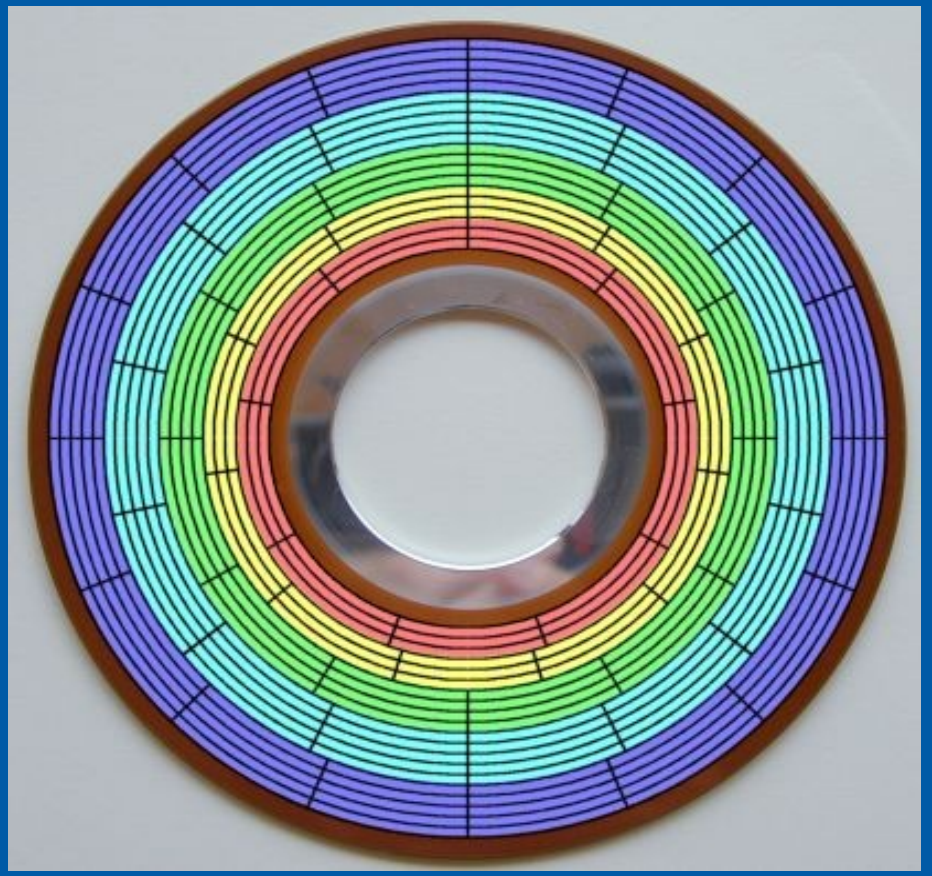

Sector
Cluster of
4 sectors

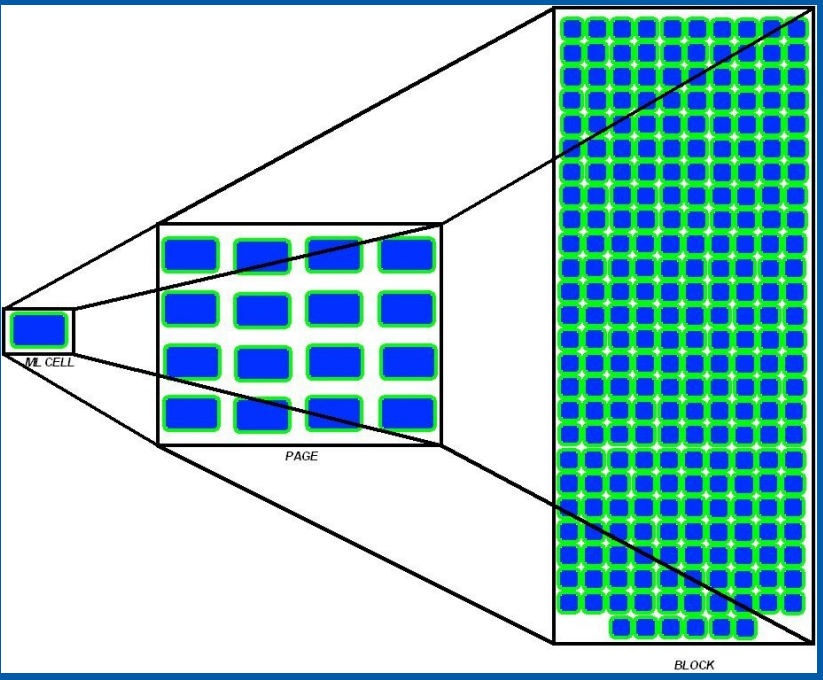


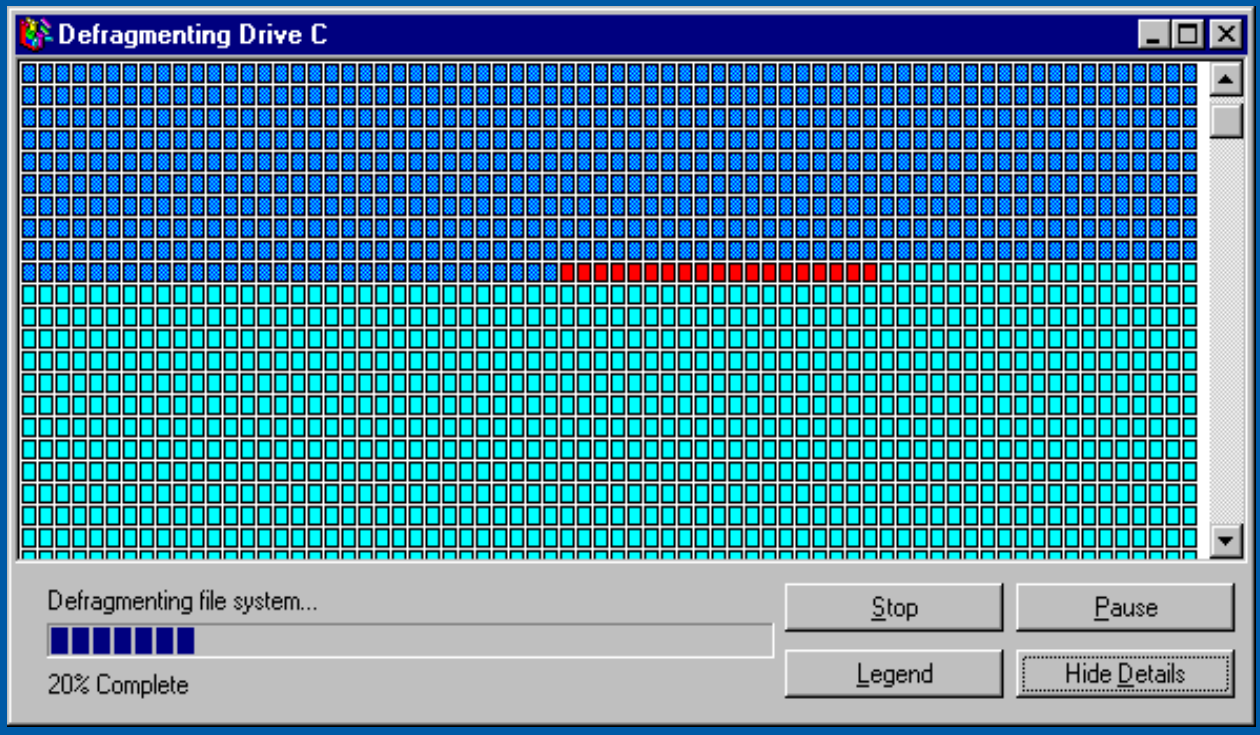

1 sector = 512 bytes
a track

Operációs Rendszerek MSc

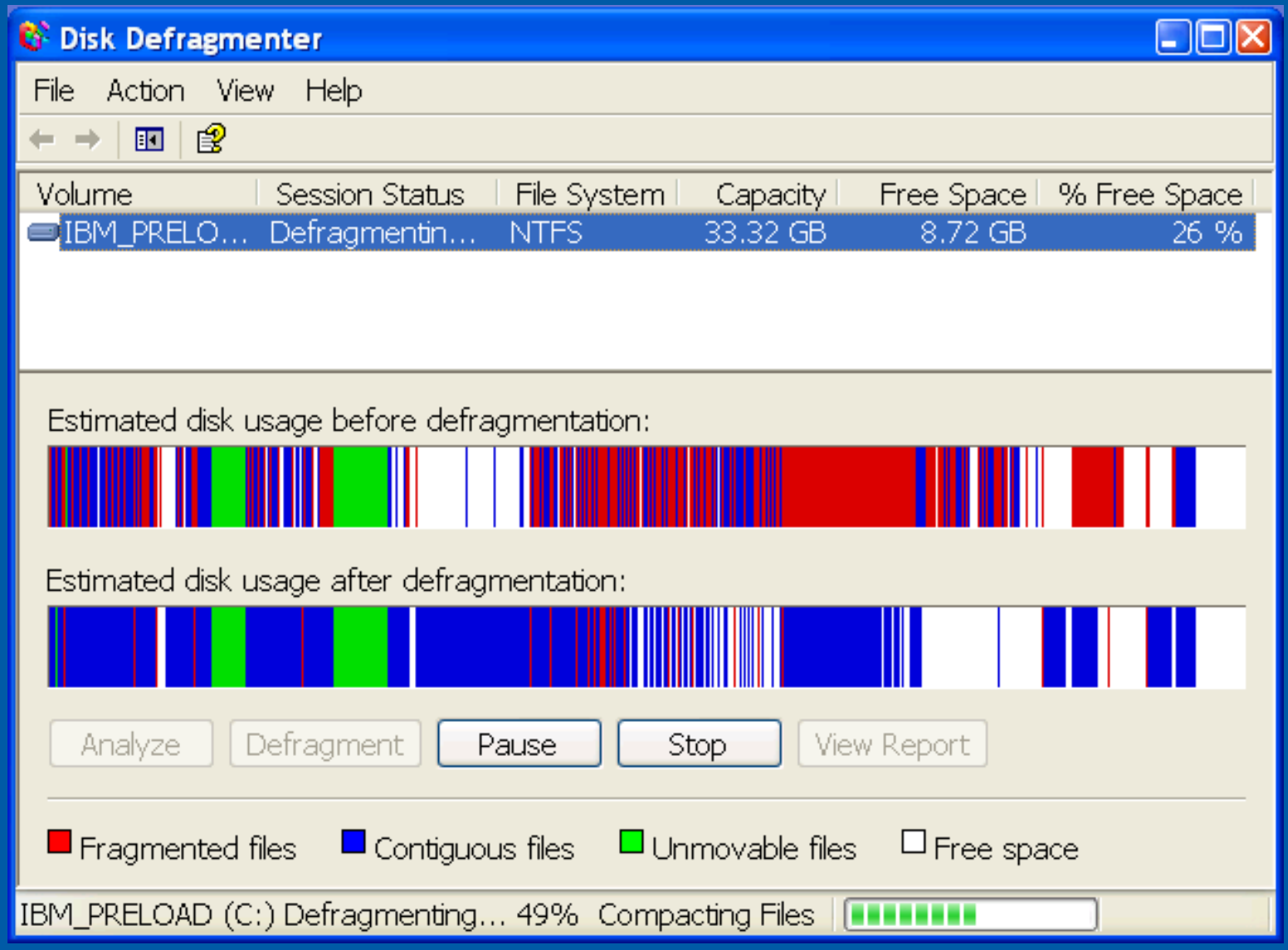


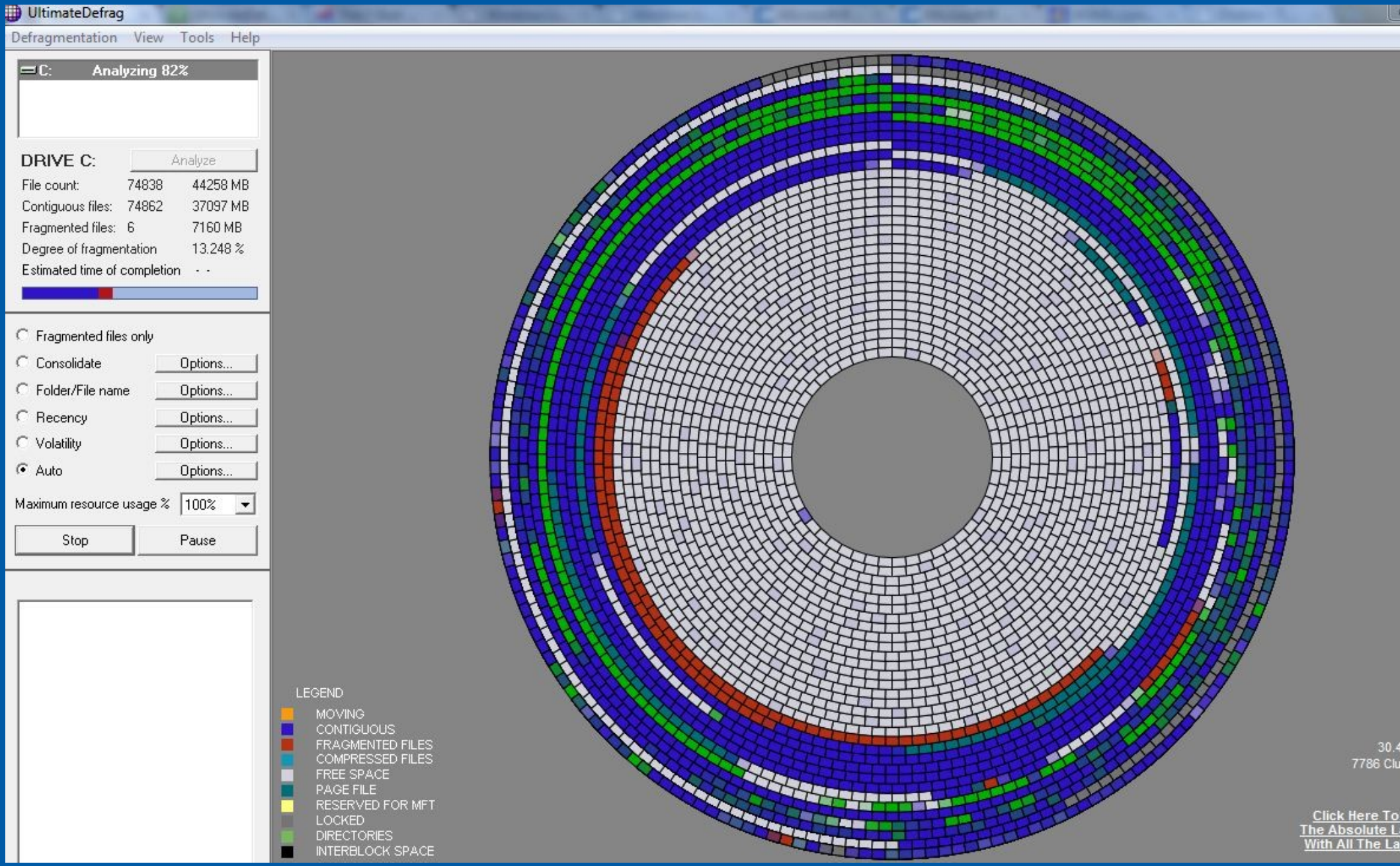



ML CELL
PAGE
BLOCK


Defragmenting Drive C
Defragmenting file system...
20% Complete
Stop
Pause
Legend
Hide Details

Operációs Rendszerek MSc




UltimateDefrag
Defragmentation View Tools Help
C: Analyzing 82%
DRIVE C:
Analyze
File count: 74838 44258 MB
Contiguous files: 74862 37097 MB
Fragmented files: 6 7160 MB
Degree of fragmentation 13.248 %
Estimated time of completion - -
Fragmented files only
Consolidate Options...
Folder/File name Options...
Recency Options...
Volatility Options...
Auto Options...
Maximum resource usage % 100%
Stop
Pause
LEGEND
MOVING
CONTIGUOUS
FRAGMENTED FILES
COMPRESSED FILES
FREE SPACE
PAGE FILE
RESERVED FOR MFT
LOCKED
DIRECTORIES
INTERBLOCK SPACE
Click Here To

Operációs Rendszerek MSc

- Fájlrendszerek
  - Lokális
    - FAT, ext2/3/4, XFS, JFS, NTFS, ISO9660, stb.
  - Ál/virtuális
    - procfs, sysfs, tmpfs, devfs, stb.
  - Hálózati
    - NFS, OpenAFS, SMB/CIFS, stb.
  - Klaszter
    - GPFS, CXFS, GFS2, OCFS2, PVFS, stb.

Operációs Rendszerek MSc

- Fájlrendszerek
  - FAT - File Allocation Table
    - Manapság elterjedt:
      memóriakártyán, pendriveon, floppy lemezen, stb.
    - Egyszerű felépítés, sok OS támogatja
    - Táblázatban tárolja, hogy hogyan vannak kiosztva a diszk területei (duplán)
    - Az alap foglalási egység a cluster
      - A cluster egymást követő szektorokból áll és fix a mérete
    - A táblázatban tárolva van, hogy foglalt-e a cluster (nem 0 értékű), illetve, ha foglalt akkor melyik clusteren folytatódik, vagy, hogy éppen fájl vége van-e azon a clusteren, netán hibás az a cluster.

    - Verziói: FAT12, FAT16, FAT32
      - A verziószám a tábla méretére utal. (Címezhető clusterek száma)

Operációs Rendszerek MSc

- Fájlrendszerek - FAT
  - Alapból 8.3 fájlnév formátum, később kapott hosszú fájlnév támogatást
  - Fragmentáció ellen nincs beépített „védelem”, az első szabad cluster-t foglalja le
  - Az allokációs táblából két példányt tart karban
    - Ha az egyik sérül még ott a másik
  - FAT32 max FS méret: 2 TB
  - FAT32 max fájlméret: 4GB
  - Directory table: spec. fájl
    - Ez egy directory...
    - Ebben van a benne lévő fájlok:
      - neve, kiterjesztése, mérete, attribútumai, első clustere

Operációs Rendszerek MSc

XXXXXXXX	XXXXXXXX	00000009	00000004
00000005	00000007	00000000	00000008
FFFFFFFF	0000000A	0000000B	00000011
0000000D	0000000E	FFFFFFFF	00000010
00000012	FFFFFFFF	00000013	00000014
00000015	00000016	FFFFFFFF	00000000
00000000	00000000	00000000	00000000
00000000	00000000	00000000	00000000
00000000	00000000	00000000	00000000
00000000	00000000	00000000	00000000
00000000	00000000	00000000	00000000
00000000	00000000	00000000	00000000
00000000	00000000	00000000	00000000
00000000	00000000	00000000	00000000
00000000	00000000	00000000	00000000
00000000	00000000	00000000	00000000

Root Directory:
2, 9, A, B, 11

File #1:
3, 4, 5, 7, 8

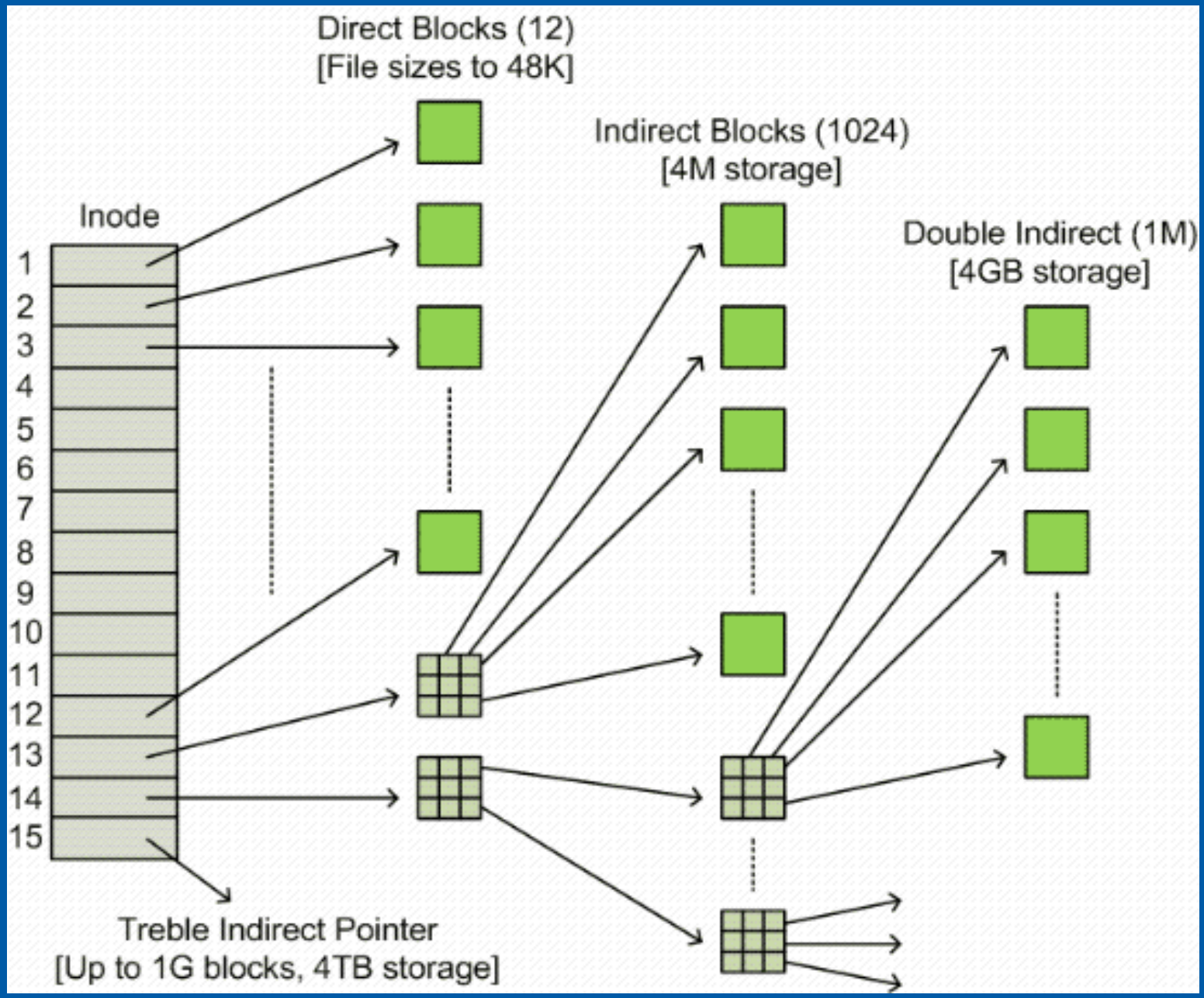
File #2:
C, D, E

File #3:
F, 10, 12, 13, 14, 15, 16

Operációs Rendszerek MSc

- Fájlrendszerek - UFS
  - UNIX File System
  - UNIX-ok elterjedt fájlrendszere (volt)
  - Superblock
  - inode: (i-bög)
    - Ez tartalmaz minden metaadatot egy adott fileról
    - UID, GID
    - Méret
    - Jogosultságok
    - Időbélyegek
    - Link számláló
    - Mutatók az adatra (12+1+1+1)
  - Directory listákban csak a fájlok nevei vannak, illetve a hozzá tartozó inode

Operációs Rendszerek MSc



Operációs Rendszerek MSc

- Fájlrendszerek - Ext2
  - Extended FS 2
    - Volt ext1 is (extended fs), max 2GB méret, 255 kar. név
      - UFS-en alapult
      - Kihalt, a kernelben sincs már hozzá nagyon régóta kód
  - Linux alá fejlesztették
  - Mára már kevésbé használt, boot fs-nek még látni manapság is

  - ACL támogatás
  - Quota támogatás
  - Foglaltságot bitmap-el nyilvántartja
  - A metaadatokat pedig táblában
  - Foglaltság kijelzés: rendszergazdának fenntartva

Operációs Rendszerek MSc

- Fájlrendszerek – Ext2
  - Blokkokat használ (1k, 2k, 4k, 8k*)
  - Ebből alakít ki block group-okat
    - Amennyi belefér az aktuális blokkméretbe (*8bit)
  - Minden blokk csoportban van:
    - Másolat a szuperblokkról
    - Block bitmap
    - Inode bitmap
    - Inode tábla
    - Adat terület
  - Blokk csoportok leírója is egy táblában

Operációs Rendszerek MSc

- Fájlrendszerek - Ext2
  - Szuperblokk
    - Több is van belőle
    - Magic (0xEF53)
    - Mount count
    - Blokk csoport száma
    - Blokkméret (1k, 2k, 4k)
    - Blokkok száma egy csoportban
    - Szabad blokkok / inodeok száma az FS-en
    - Az első inode (2.)
    - Fájl rendszer állapot (clean, dirty)
    - Egyebek...
      - A kernel forrásában megtalálhatóak a leíró struktúrák

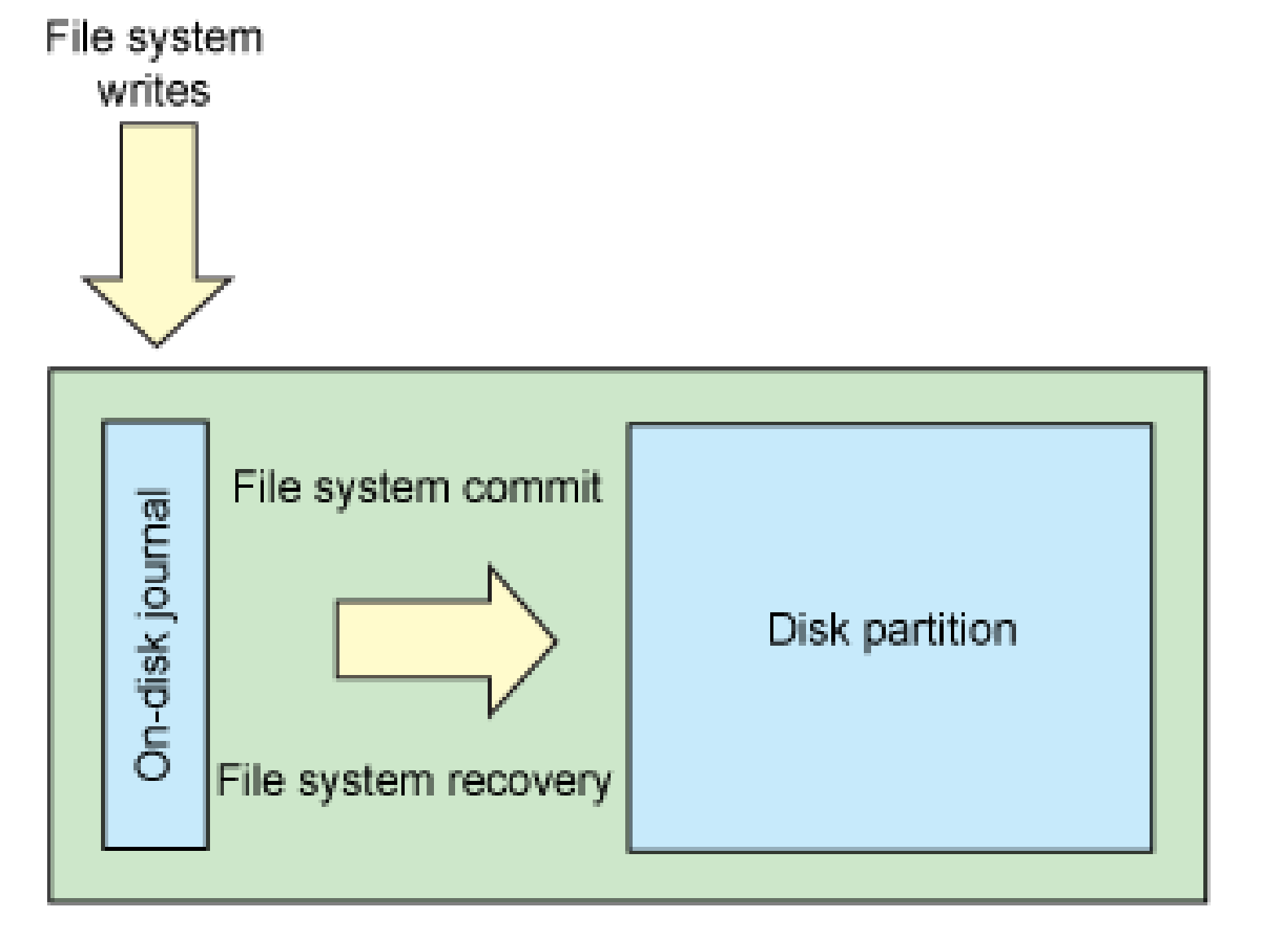
Operációs Rendszerek MSc

- Fájlrendszerek - Ext2
  - Directory
    - Fájl bejegyzéseket tartalmazza
    - Fájl nevének hosszát, fájl nevét, inode számát
    - A . és .. automatikusan létrejönnek és nem törölhetőek
      - Hozzájuk rendelődik a saját és a szülő inode-ja
    - Nincs indexelés
  - Inode
    - Hasonló mint UFS-nél
      - blokk foglaltság, időbélyegek, UID/GID, stb.
    - A 2. inode-on mindig a root directory van
      - (chroot!)
  - *Miért nem az inode-on van tárolva a fájlnév?*
  - *Számoljuk ki a max fájlméretet!*

Operációs Rendszerek MSc

- Naplózó fájlrendszerek (Journaling FSs)
  - Napló? Mit naplóz?
    - A műveleteket még mielőtt végrehajtódnának
    - (ebben a kontextusban metaadat: létrehozás, eltávolítás, fájlméret növelés, csökkenés, stb.)
    - Ideális esetben ezek a műveletek azonnal íródnak a háttértárra
  - Physical Log
    - Mindent naplóz
      - konkrét adatokat, blokkokat, egymás után
    - Nagy az overhead, de adatvesztés, fs korrupció nem lesz
  - Logical Log
    - Ez az elterjedtebb
    - Magát a logikai műveletet (metaadat) naplózza

Operációs Rendszerek MSc



Operációs Rendszerek MSc

- Fájlrendszerek – Naplózó FS
  - Writeback mód
    - Metaadatot naplózza
    - Utána írja az adatblokkokat a helyükre
  - Ordered mód
    - Először írja az adatblokkokat a helyükre
    - Ezután naplóz
  - Data mód
    - Két helyre ír, mindent, naplóba is, helyükre is
    - Kétszer ír → teljesítményromlás
    - Nagyobb megbízhatóság

Operációs Rendszerek MSc

- Fájlrendszerek – Naplózó FS
  - IBM JFS (JFS2)
    - Ez volt az első kereskedelmi forgalomban elérhető naplózó fájlrendszer
    - AIX-on jelent meg, később open source Linux port
    - 64-bites
    - Ordered naplózás
    - Allocation Group-okat használ
    - Extent alapú helyfoglalás
    - B+ fák használata
      - Directory tartalmak
      - Extent nyilvántartás
    - Nincs beépített journal commit policy-je, egy külső program triggereli a commitet (*kupdate*)
    - Dinamikus i-node allokáció

Operációs Rendszerek MSc

- Fájlrendszerek – Naplózó FS
  - IBM JFS (JFS2)
    - ACL támogatás
    - Quota támogatás
    - Szuperblokk tartalma:
      - Az egész FS mérete
      - Az elérhető adatblokkok száma
      - Az AG-k mérete
      - FS blokkmérete
    - Menet közben átméretezhető
    - Defragmentálódást próbája minimalizálni (extent)
    - AIX JFS1...

Operációs Rendszerek MSc

- Fájlrendszerek – Naplózó FS
  - ext3 – (ext4 később)
    - A szerkezete megegyezik az ext2 szerkezetével
    - Csak kapott hozzá egy naplót
    - Ezért könnyedén konvertálható ext2 <> ext3
    - Támogatott naplózás típusok:
      - Writeback, Ordered, Data (lásd előrébb)
    - Konfigurálható a commit policy
      - Default: 1/4 napló telítettség, illetve periodikusan
    - Hátránya, hogy csak egy kiegészítés az ext2-höz, nem az alapoktól fogva naplózó
    - Quota támogatás
    - Online FS növelés
    - HTree directory indexelés

Operációs Rendszerek MSc

- Fájlrendszerek – Naplózó FS
  - ReiserFS (v3) – Hans Reiser/Namesys
    - Az első jfs ami bekerült a Linux kernelbe
    - Naplózás típus: ordered, de képes block-ra is
    - Tail packing: (block suballocation)
      - Blokknál kisebb fájl esetén veszteség
        - 1 byte fájl foglaltság is 1 egész block foglaltság
        - Több fájlt tud egy logikai blokkon eltárolni
        - Hatékonyabb helykihasználtság
        - Teljesítmény veszteség
    - Egy nagy B+ fa
      - Mindennek Object ID (directory, inode lista, fájl, metadata)
    - Block foglaltság bitmap alapján

Operációs Rendszerek MSc

- Fájlrendszerek – Naplózó FS
  - ReiserFS (v3 v4) – Hans Reiser/Namesys
    - Commit policy:
      - Függ a napló méretétől
      - Mennyi a kiírandó blokkok száma
    - Quota támogatás (külön patch)
    - Menetközbeni növelés (grow) csökkentés (shrink)
    - Fejlesztés bizonytalan...

    - Reiser4
      - Még hatékonyabb... stb.
      - Delayed allocation
      - Kis fájlokkal nagyon jó teljesítmény
      - B* fa
      - Kernelben nincs benne alapból egyelőre (?)

Operációs Rendszerek MSc

- Fájlrendszerek – XFS

  - SGI XFS
    - IRIX, 64-bit FS
    - Nagy fájlok, nagy fájlrendszer (8EB fájl, 16EB FS)
    - Realtime szolgáltatás is van
    - Quota támogatás
    - ACL támogatás
    - Menet közbeni növelés (grow)
    - Csökkentés (shrink) sem offline sem online módon sincs

Operációs Rendszerek MSc

- Fájlrendszerek – XFS
  - SGI XFS

    - Allocation Group (AG)
      - Több egyenlő részre osztja a területet (max 1TB), és ezekben egymástól függetlenül lehet foglalni
      - Szuperblokk (primary + másolatok)
      - Saját maguk kezelik a szabad helyet, inodeokat, stb.
      - Értelmezhető alfájlrendszernek is
      - Egyszerre így több is címezhető (SMP rendszereken ez előnyös)

Operációs Rendszerek MSc

- Fájlrendszerek – XFS
  - SGI XFS
    - Delayed allocation
      - Számszerűen lefoglalja a területet, de a diszkeken nem foglal le konkrét blokkokat
      - Eközben a memóriában van a tényleges tartalom
      - Előnyös ha pl. rövid életű a fájl
      - Vagy ha közben szabadul fel egy köztes terület, és nagyobb összefüggő terület foglalható le
      - Hátrány ha rendszerösszeomlás van → adatvesztés
    - Pre-allocation
      - Többet lefoglal, mint amennyire pillanatnyilag szükség van (kisebb fragmentáció, de „pazarlás”)

Operációs Rendszerek MSc

- Fájlrendszerek – XFS
  - SGI XFS

    - A napló különválasztható, akár másik eszközre is
    - Aszinkron módon commit-eli a naplót
      - Delayed logging, log checksum
    - Extent-eket használ
    - Az inodeok és a szabad terület B+ fákon van tárolva
      - szabad terület: 2 B+ fa, egyik a szabad extentek hossza szerint, a másik az extent kezdőblokkja szerint indexelve

Operációs Rendszerek MSc

- Fájlrendszerek – XFS
  - SGI XFS
    - Online defragmentáció
      - Habár törekszik az allokáció a töredezettség minimalizálására, előfordulhat, hogy mégis nagy lesz a mértéke
      - Menet közben (felcsatolt, használatban lévő) képes átrendezni a blokkokat
    - Garantált I/O sebesség („realtime”)
      - XFS egyedi tulajdonsága
      - Fájlrendszerenként tud garantálni I/O sávszélességet
      - Alapvetően media-streaming-hez lett kifejlesztve
    - Napló

Operációs Rendszerek MSc

- Fájlrendszerek – XFS
  - Szuperblokk
    - Magic (”XFSB”)
    - Blokkméret (alaphelyzet 4K, 0.5-64K)
    - Összesen mennyi blokk van
    - Hol kezdődik a napló (belső naplózás esetén)
    - Mennyi blokk foglalt a naplóban
    - AG méret blokkokban
    - Root inode száma (4K blokkméret esetén 128)
    - FS verziója
    - Inode méret
    - Egy blokkban mennyi inode van
    - Százalékosan a blokkokból mennyin lehet inode
    - FS név
    - tovább

Operációs Rendszerek MSc

- Fájlrendszerek – XFS
  - Szuperblokk
    - Real-time paraméterek
    - Quota paraméterek
    - Egyebek...
    - Csak az elsődleges szuperblokkban:
      - Összes inode száma
      - Összes szabad inode száma
      - Összes szabad adatblokk száma
      - Összes szabad extent száma

Operációs Rendszerek MSc

- Fájlrendszerek – XFS
  - Inode
    - Magic (”IN”)
    - Mode (jogok, rwxrwxrwx)
    - UID/GID
    - Linkek száma
    - Időbélyegek
    - Méret byteokban
    - Stb.

Operációs Rendszerek MSc

- Fájlrendszerek – XFS


```
gold:~# xfs_growfs /var/lib/mysql
meta-data=/var/lib/mysql         isize=256    agcount=8, agsize=98304 blks
         =                       sectsz=512
data     =                       bsize=4096   blocks=786432, imaxpct=25
         =                       sunit=0      swidth=0 blks, unwritten=1
naming   =version 2              bsize=4096
log      =internal               bsize=4096   blocks=2560, version=1
         =                       sectsz=512   sunit=0 blks
realtime =none                   extsz=65536  blocks=0, rtextents=0
data blocks changed from 786432 to 1048576
```

Operációs Rendszerek MSc

- Fájlrendszerek – Naplózó FS
  - ext4
    - Nagyobb (ext2/3) max fs és fájl méret:
      - 48-bites címzés (1EB fs, 16TB fájl)
    - Multiblock allokáció
    - Extent-eket használ
    - Journal checksum – naplóbejegyzések ellenőrző összeggel vannak ellátva
    - Kikapcsolható journal (ha csak a teljesítmény számít)
    - Inode-okban nanosec pontosságú timestamp-ek
    - Online defragmentálás
    - Quota támogatás
    - ACL támogatás
    - Online növelés (grow), csökkentés (shrink)

Operációs Rendszerek MSc

- Fájlrendszerek – Naplózó FS
  - ext4
    - Visszafelé kompatibilis ext3-al és ext2-vel
    - Ext3 limit: 32 000 aldirectory, ext4-ben unlimited
    - Gyorsabb fájlrendszer (teljes) ellenőrzés:
      - A nem használt block csoportok és inodeok meg vannak jelölve, így azok kihagyhatóak
    - Delayed allocation – késleltetett foglalás
    - Apró fájlok tartalmát magában az inode-ban el tudja tárolni

Operációs Rendszerek MSc

- Fájlrendszerek – Naplózó FS
  - NTFS
    - Minden fájl → A metadatokat is fájlként tárolja
      - Az egész fs egy nagy adatterület
    - Naplózó FS (kikapcsolható)
    - Max 256TB FS, max 16TB fájl
    - Master File Table
      - Ebben vannak a fájlok leírói, inode
      - Az első 16 bejegyzés metaadat fájloknak fenntartva
      - Biztonsági másolat
    - Titkosítás támogatás
    - Indexelve (B fa) tárolja a fájlneveket (v3-tól egyebeket is)
    - Terület foglaltság bitmap alapján
      - best-fit, cluster az alap
    - Alternate Data Stream

Operációs Rendszerek MSc

- Fájlrendszerek – Naplózó FS
  - btrfs (Oracle)
    - B-tree filesystem
    - Ez majd mindennél jobb lesz...
    - Több device-ra kiterjed
    - Online defragmentáció
    - Online grow és shrink
    - Online device hozzáadás, kivétel
    - Fájlrendszer szintű tükrözés, stripe-olás (lásd RAID)
    - Transzparens adattömörítés
    - File-klónozás (CoW), pillanatkép (snapshot)
    - Ellenőrző összegek metaadatokra és adatokra egyaránt

Operációs Rendszerek MSc

- Fájlrendszerek – Naplózó FS
  - ZFS
    - Sun (Oracle)
    - 128-bites *lol*
    - Solaris, Linux, BSD
    - Az elsődleges szempont az adatintegritás
    - **Deduplikáció**
    - Titkosításra lehetőség
    - Változó blokkméret
    - Zpool
      - Több fizikai diszkre is kiterjedhet
    - FS szintű RAID-0, RAID-1
    - RAID-Z, RAID-Z2 (RAID5-6)

Operációs Rendszerek MSc

- Fájlrendszerek – Naplózó FS
  - **ZFS Dedup**
    - Minden blokkról checksum-ot készít
    - Ezt eltárolja (mem)
    - Új blokk kiírásakor ha van egyezés egy korábbival, akkor csak hivatkozik rá
    - Ez elég drága művelet:
      - Pl. 1TB esetén, 64k blokkokkal, kb. 5GB memória szükséges!
      - Nincs ingyen a nyilvántartás karbantartása és a keresés benne
    - Mérlegeljünk:
      teljesítménycsökkenés, memória fogyasztás vs. megspórolt diszk hely

Operációs Rendszerek MSc

- Fájlrendszerek
  - ISO9660 (CDFS)
    - Read-only, előre masterelt
    - Nincs fragmentáció
    - Szabad hely nyilvántartásra nincs szükség
    - Első 32k-t nem használja semmire (felhasználható)
    - Utána volume descriptor (kötet leíró) következik
      - Info az egész FS-ről
      - Root dir helye
      - Mekkora az egész FS
      - Azonosító, stb.
    - Ezután directory tree
    - Logikai blokkokat használ, nem közvetlen a szektorokat
    - Multi-session
      - Utolsó session-be beíródik az egész directory tree + az új
    - Bootolás: El Torito, floppynak emulálja a BIOS

Operációs Rendszerek MSc

- Flash / SSD
  - Törölt (tiszta) állapotban 1-es állapotú minden bit
  - Íráskor vagy marad, vagy 0-ra változik
  - 0-ról 1-re csak teljes törléssel állítható vissza
  - Csak blokkban törölhető
  - Véges számú írást visel el
  - Ha kevés az írás, akkor jó lehet az 1:1 megfeleltetés is (logikai-fizikai)
  - Ha azonban számottevő az írások száma más megoldás kell:
  - Wear-leveling:
    - Szétosztani az egész eszközön a blokkokat, még ha az logikailag ugyanaz a blokk

Operációs Rendszerek MSc

- Flash / SSD
  - Eszköz szinten wear-leveling + hagyományos FS
  - Fájlrendszer szinten megvalósítani a wear-leveling-et
  - Ne csak a felhasználói adatokra gondoljunk!
    - pl. FAT tábla
    - szuperblokkok
    - napló / journal
    - inode nyilvántartás, stb.
    - directory bejegyzései
    - utolsó módosítás/hozzáférés dátuma
  - Igazítás blokk határra (partíció, fs adatblokk)

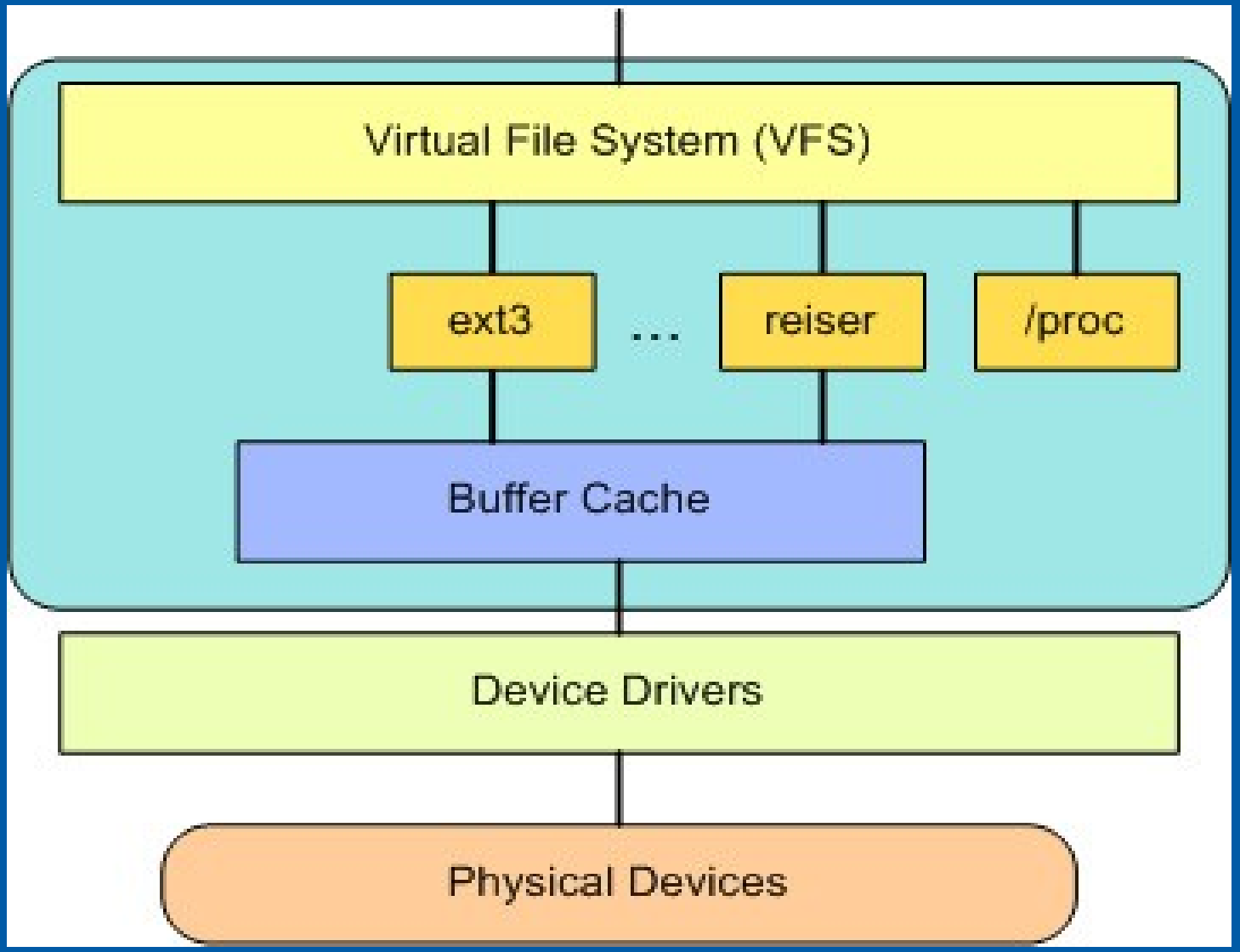
Operációs Rendszerek MSc

- Flash / SSD
  - Flash fájlrendszer megfontolások:
    - Az adatok nem írhatóak ki egyből, előtte komplett blokk törlés szükséges
    - Blokk allokálásnál nem fontos a folytonosság, hiszen az összes blokk elérési ideje megegyezik
    - A blokkok csak véges számú törlést viselnek el
  - Log-structured FS
    - Folyamatosan ír, mindig „előre”, naplóként
    - Körkörös buffer szerűen
    - Régebbi verziójú adatok nem kerülnek törlésre
      - Helyreállítás egyszerűbb
    - Az eszköz végére érve elölről kezdi
    - pl. JFFS, JFFS2, YAFFS, UBIFS, F2FS, stb.

Operációs Rendszerek MSc

- Fájlrendszerek
  - Linux VFS (Virtual File System)
    - Közös interfész a fájlrendszer eléréshez
    - Ez az ami közvetlenül elérhető rendszerhívásokkal
    - Ez alá épülnek be az igazi FS implementációk
    - „Fordít” a rendszerhívások és az konkrét FS hívásai között
      - open(),close(),read(),write(), stb.

Linux VFS

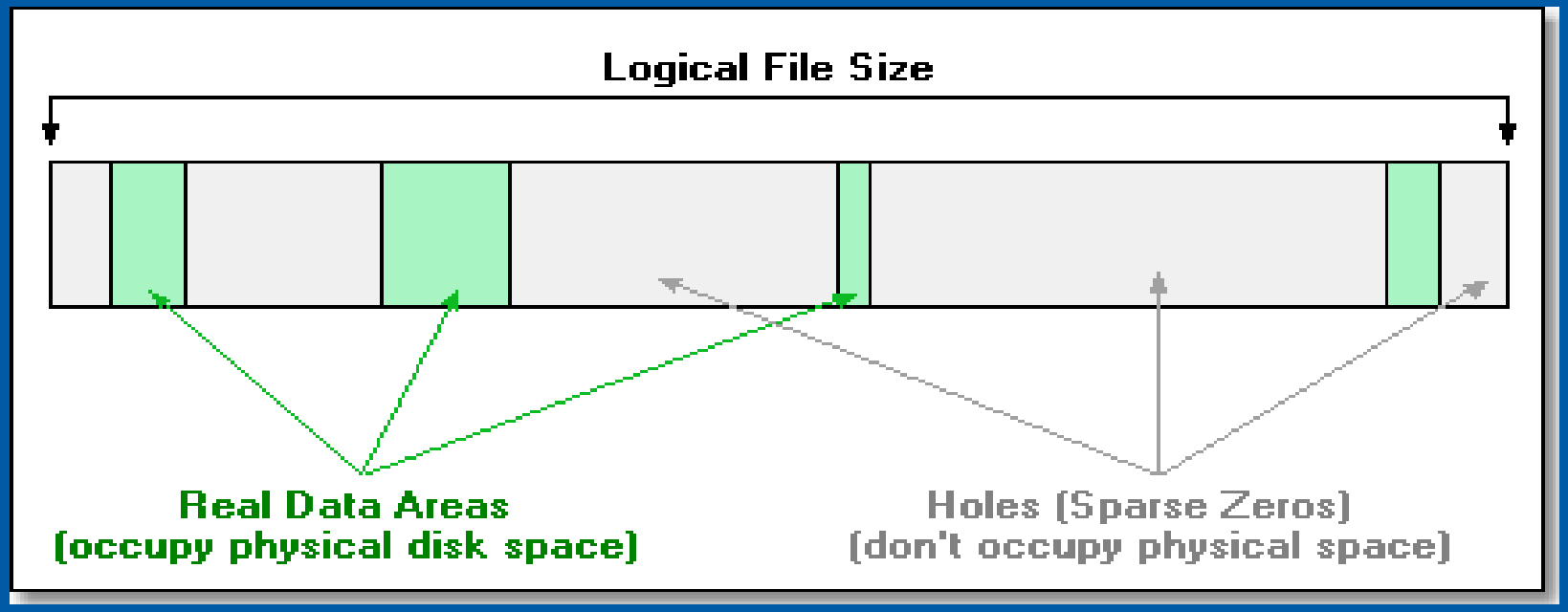

Virtual File System (VFS)
ext3
...
reiser
/proc
Buffer Cache
Device Drivers
Physical Devices

Operációs Rendszerek MSc

- Fájlrendszerek
  - Fragmentáció (töredezettség)
    - Internal fragmentation (belső)
      - Kevesebb adatot tárolunk, mint lehetne
      - pl. 1 byte egy blokkban
      - Elvész a kapacitás
    - External fragmentation (külső)
      - Maguk a blokkok/cluster-ek/extent-ek nem folytatólagosak
      - „Lyukak” vannak a diszken
      - Sok létrehozás + törlés eredménye
      - HDD-nél teljesítményromlást okoz, mert sokat „ugrál” a fej
      - SSD-nél nem feltétlen jelent gondot, mert egyenrangú minden rész elérése (esetleg a több kis címzés egy nagy bulk helyett jelenthet elenyésző teljesítmény igény többletet – blokk vs. extent)

Operációs Rendszerek MSc

- Fájlrendszerek
  - Sparse file („ritka file”)



  - Helytakarékosság
  - Fájl méret vs. valós foglaltság
  - Töredezettség nagyobb lehet ha később használják

Operációs Rendszerek MSc

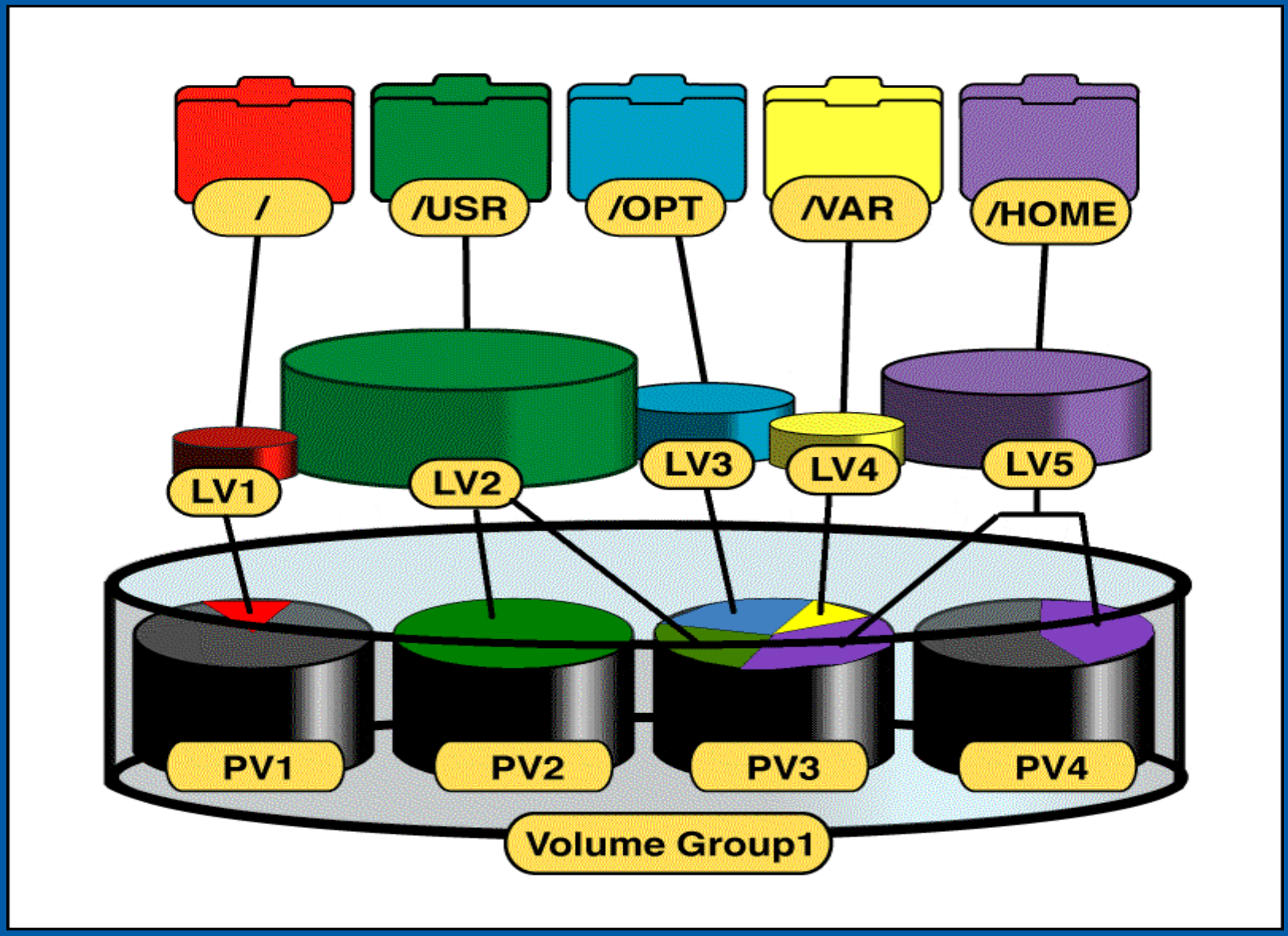
- Fájlrendszerek
  - Linux mount opciók
    - ro/rw
    - dev/nodev
    - exec/noexec
    - suid/nosuid
    - atime/noatime
    - barrier/nobarrier
    - tail/notail (reiser)
    - defaults

    - *man mount :-)*

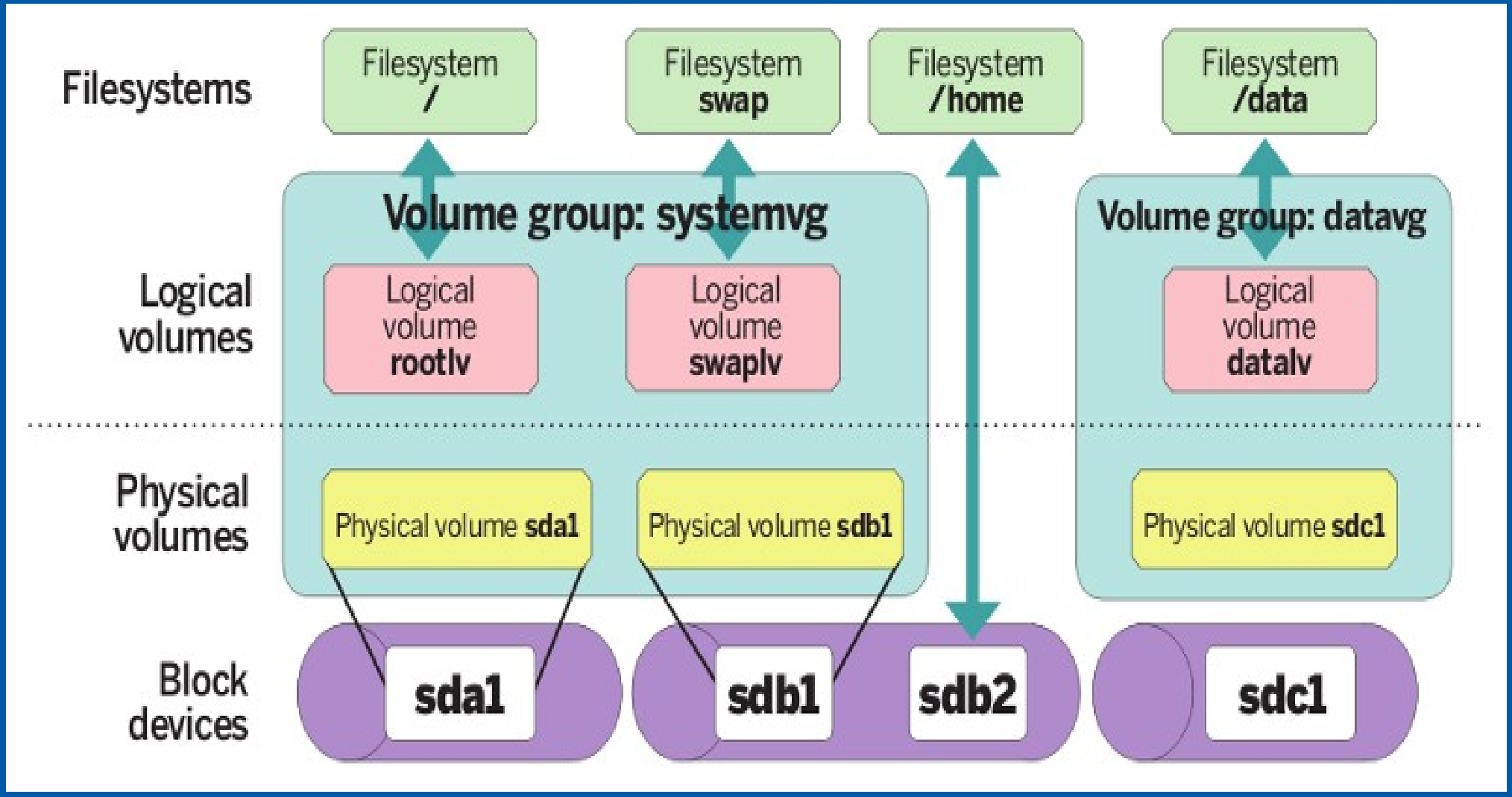
Operációs Rendszerek MSc

- Logikai kötet kezelés
  - Több rendszeren is van, vagy hasonló van (pl. AIX, HP-UX, stb.)
  - LVM – Logical Volume Management
  - Linuxon LVM
  - PV – Physical Volume
    - Magára a fizikai eszközre hozzuk létre, pl. /dev/sda, sdb2, md0, stb.
  - VG – Volume Group
    - A PV-k szervezhetőek össze egy csoportba
  - LV – Logical Volume
    - A VG-n alakítható ki, ez lesz maga az elérhető blokkos eszköz
  - Összefűzni több eszközt
  - Feldarabolni kisebbekre (több más FS, más opciókkal)
  - Menet közbeni átméretezés, VG bővítés, PV bővítés, áthelyezés, stb.
  - Stripe, mirror…

Operációs Rendszerek MSc



Operációs Rendszerek MSc



Operációs Rendszerek MSc

- Logikai kötet kezelés
  - PV-ken UUID (Universally Unique ID)
  - PV-ket Physical Extent (PE)-kre bontja fel
  - LV-ket Logical Extent (LE)-kre bontja fel
  - A LE-ek PE-khez vannak rendelve
  - A kialakított LV-k nem kell, hogy folytonosan egymást követő LE-kből álljanak össze
  - A VG-k menet közben kaphatnak új PV-ket
  - El is lehet tőlük venni, feltéve, ha nincsenek rajta PE-k használatban

  - External Fragmentation

  - Snapshot (lásd SAN-oknál FlashCopy, stb.)

Operációs Rendszerek MSc

- ÁI-fájlrendszerek
  - tmpfs
    - Az operatív memóriában (RAM) tárolja az adatokat, nincs mögötte háttértár (másodlagos tároló)
    - = ramdisk
    - Tehát pl. újraindításkor, rendszerösszeomláskor elvész
    - A kernel belső cache-t használja tárolásra
    - Szükség esetén a swap-et is tudja használni
    - A mérete dinamikusan változik, nő, csökken a valós memóriahasználat
    - Célszerű pl. /tmp alá ezt tenni
      - Ha SSD-n lenne alapból a rendszer, akkor főleg...
      - Esetleg más átmeneti dolgok alá, aminél az adatbiztonság nem fontos (pl. browser cache)

Operációs Rendszerek MSc

- ÁI-fájlrendszerek
  - procfs
    - Ehhez sem tartozik háttértár
    - Látszólag rendes fájlokat tartalmaz
    - Elsődlegesen a rendszeren futó processzekről ad információkat
    - PID-enként létrejön egy directory, azon belül az adott PID-ű processzről információk
    - Egyéb fájlok
      - információszolgáltatás
      - konfigurálás (sysctl)
    - A fájlok hozzáférésekor, külön erre a célra írt kernel függvények hívódnak meg
      - Olvasáskor ezek generálnak tartalmat
      - Íráskor ezek értelmezik a tartalmat

Operációs Rendszerek MSc

- Ál-fájlrendszerek
  - devfs
    - Ehhez sem tartozik háttértár
    - Kizárólag speciális eszközfájlokat tartalmazhat
    - Manapság már nem használt, lásd devtmpfs és udev
    - A létrehozás és törlés „automatikus”, maguk a driverek „kérik”
    - /dev alá felcsatolva

Operációs Rendszerek MSc

- ÁI-fájlrendszerek
  - sysfs
    - Mint a procfs
    - Mondván, hogy a procfs teli van nem is processzekre vonatkozó dolgokkal
    - Legyen egy külön fs, amin keresztül csak a rendszerinfók, konfigurációk érhetőek el
    - v2.6 Linux kerneltől
    - Jelenleg kardinális részét képezi a Linux kernelnek

Operációs Rendszerek MSc

- Hálózati fájlrendszerek
  - NFS – Network FileSystem
  - UDP vagy TCP felett
  - Megvalósítás RPC-k segítségével
  - Kliens-szerver
  - A szerveren fut egy NFS kiszolgáló (daemon)
  - Linuxon van kernelben, illetve user-space-ben megvalósított kiszolgáló
  - A szerveren helyi FS-eket ajánlunk ki
  - A kliens felcsatoláskor RPC-n kereszül lekérdezi, hogy hol érhető el (melyik porton)
  - Ezután magához az NFSd-hez fordul
  - Ha sikeres a felcsatolás, akkor a felhasználó számára helyi FS-ként fog látszódni

Operációs Rendszerek MSc

- Adatbázis fájlrendszerek
  - Komolyabb adatbáziskezelők képesek a nyers diszket használni
  - Nem OS által biztosított FS réteg
  - Saját maga a DBMS alakít ki rajta egy struktúrát
  - Optimális adattárolás
    - Egyszerre több diszket kezelhet
    - Adatok határokra (szektor, blokk) igazítása
    - REDO log, stb. külön-külön diszkekre
  - pl. Oracle

Operációs Rendszerek MSc

- Klaszter fájlrendszerek
  - CLVM - Cluster LVM
  - CXFS - Cluster XFS
  - GPFS/VSD
    - IBM General Parallel FS
    - Virtual Shared Device (AIX)
  - GFS2
    - RedHat Global File System
  - OCFS2
    - Oracle Cluster FS
  - DLM – Distributed Locking Manager
  - PVFS
    - Parallel Virtual FS

Operációs Rendszerek MSc

- Klaszter fájlrendszerek
  - CLVM - Cluster LVM
  - Ugyanazt tudja, mint az LVM
  - Olvasáskor nem lép fel probléma
    - (nem adat olvasás, hanem az PV/VG/LV leírók olvasása)
  - Írásnál annál inkább
    - Zárolás
    - A módosítások propagálása az adott node-ok között

Operációs Rendszerek MSc

- Klaszter fájlrendszerek
  - CXFS/GPFS/OCFS2/GFS2
  - Shared Storage (pl. SAN)
  - Fizikai szinten elérhető minden node számára
  - A diszket nem érdekli ki írja/olvassa
  - Konkurens hozzáférés
  - Elosztott metaadat (GPFS)
  - Elosztott zárolás (DLM)
  - GPFS: Hálózati hiba esetén képes detektálni, melyik a legnagyobb elérhető csoport, és ez életben tud maradni
  - GPFS: Online FS ellenőrzés és javítás

Operációs Rendszerek MSc

- Klaszter fájlrendszerek
  - PVFS
  - Több tároló eszköz egy nagy virtuális eszközként
  - Minden node számára
  - Metaadat szerver, I/O node, Computing node
  - Egy felettes réteg
    - Használható „alá” bármilyen FS
  - Objektum azonosítók (handle)
    - Mindegyik tartalmazza, melyik kiszolgálón található
  - Az adatokat stripe-olja a node-ok között