



Virtualizációs Technológiák

SAN/NAS/DAS

RAID szintek

Storage virtualizáció

Kovács Ákos

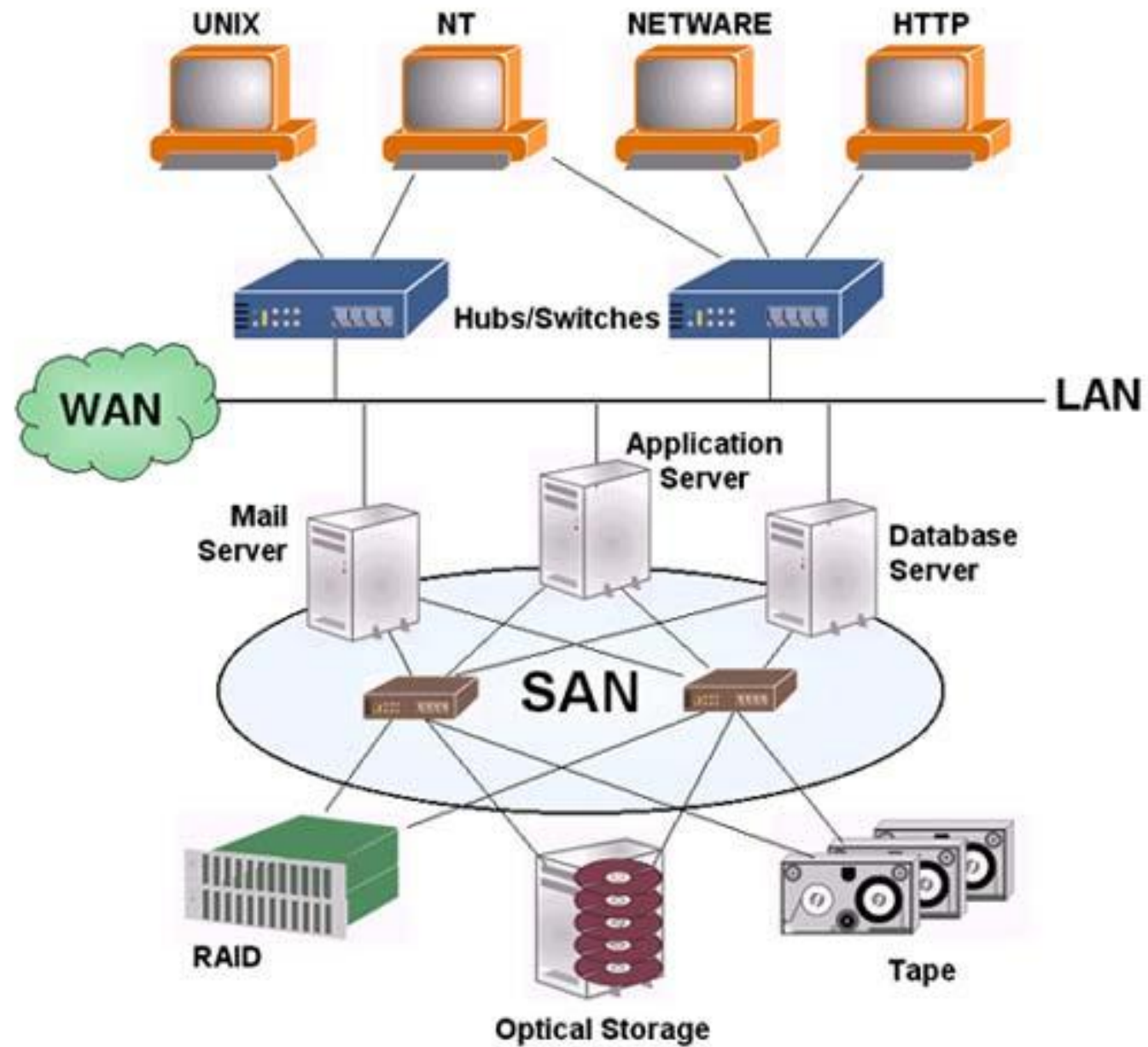
Technológiák

- SAN
- NAS
- RAID
- VSAN

SAN

- Storage Area Network
- Dedikált hálózat mely hozzáférést biztosít valamilyen tárhely szolgáltatáshoz blokk szinten
- Pl.: ISCSI, FC, Infiniband

Storage Area Networks

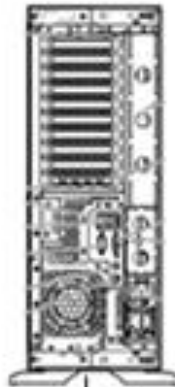


NAS

- Network Attached Storage
- Valamilyen már meglévő tárhely kiajánlása a hálózatra fájlrendszer szinten
 - Pl.: NFS (Network File System)
 - SMB (Server Message Block)
 - CIFS (Common Internet File System)

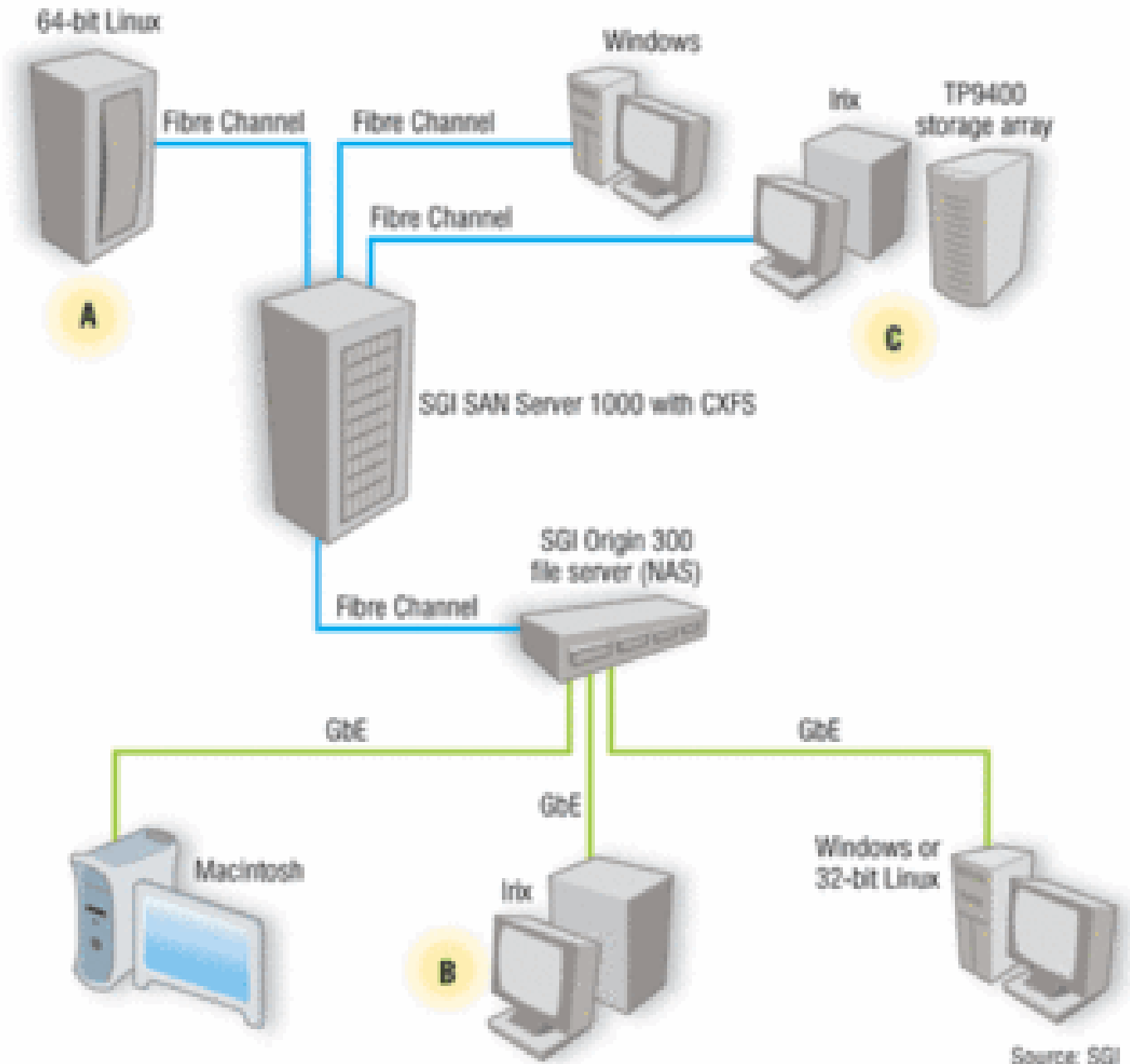
NAS Server

Clients



Ethernet

Combining NAS and SAN



ISCSI

- Internet Small Computer System Interface
- Kliens (Initiator)
- ISCSI kiszolgáló (target)
- Helyi tárhely illúziója (virtuális merevlemez)
- A már meglévő hálózatba integrálható

ISCSI

- Azonosítók, címek: IQN (**iSCSI Qualified Name**)
- iqn.2014-02.hu.sze.tilb.storage2:storage4T-vmware

Típus	dátum	FQDN az azonosításhoz	A kiajánlott blokkeszköz neve
-------	-------	--------------------------	----------------------------------

ISCSI

- iSNS – Internet Storage Name Services
 - DNS szolgáltatás storagekhez
- Autentikáció CHAP
 - User:pass
 - Sérülékeny, legjobb fizikailag/logikailag elszeparálni

NFS

- **Network File System** (*NFS*) állományok és könyvtárak megosztását teszi lehetővé a hálózaton keresztül.
- Az *NFS* a fogyasztói réteg szolgáltatása, ami bármiféle kapcsolatrendszerben funkcionál, ami *TCP* és *UDP* protokollokat használ.
- Az *NFS* elsődleges célpontja a helyi és a távoli fájlok használati módja közti határ elmosása, a távoli fájlok elérésének egyszerűsítése volt.

NFS

- 1. verzió
 - SUN Microsystems, soha nem volt mindenki számára elérhető verzió
- 2. verzió
 - UDP alapú fájllelérés (halott ügy), a zárolást a protokollon kívül oldották meg, az úgynevezett *NLM* (*network lock manager*, hálózati zárolás manager) segítségével.
- 3. verzió
 - A 3-as verzió leginkább az akkori technológia által megkövetelt újdonságokkal való kiegészítése volt a korábbi protokollnak. Támogatta a 64 bites offszet és méret értékeket, így 4 GB-nál nagyobb fájlokat is megvalósíthattunk.
 - Ebben a verzióban vált hivatalosan is elérhetővé, a *TCP* feletti működés, melyet több gyártó már a 2-es verzióhoz is implementált.
 - Ezen verzió ráadásul a *portmapper* szolgáltatást is használta, ami nagyon nehézkessé tette a tűzfalakon való áthaladást. A *portmapper* gyakorlatilag egy olyan adatbázist tartott nyilván, hogy mely szolgáltatások, milyen porton figyelnek. A kliens először a *portmapper*hez fordult, hogy megkérdezze a portszámot, majd a válasz alapján csatlakozott a távoli géphez.

NFS

- 4. verzió
 - A 4-es verzió szakított az állapot nélküiség gondolatával, így tartalmaz egy csatolással (*mount*) és egy zárolással (*locking*) kapcsolatos alrendszert is. A fejlesztők kölcsönöztek elképzeléseket mind az *AFS*, mint a *CIFS* fájlrendszerektől, például a sebesség optimalizációkat, erős biztonság megkövetelését. Ez a protokoll elvileg képes a több szerveren replikált fájlok hibatűrő elérésére is.
 - Az NFS a számítógép rendszermagja által nyújtott fájlrendszereket teszi hálózaton elérhetővé, nem használ különleges fájlrendszer absztrakciós réteget.

Jumbo Frame

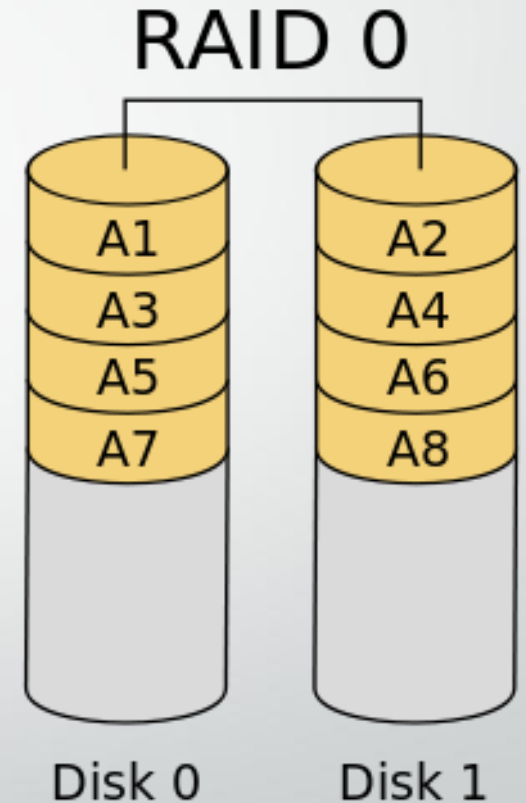
- Hardveres támogatás kell hozzá
- Az ethernet keret 1500 bájtos max payloadját kitolja 9000-ig
- Az eszközökön a jumbo frame engedélyezése, ha elfelejtjük el fogják dobni az eszközök mivel nem az ethernet szabványának megfelelően van formázva a csomag
- Operációs rendszerekben MTU átállítása (Max Transmission Unit)

RAID tömbök

- RAID – Redundant Array of Independent Disks
- RAID-ben eredetileg 5 szintet definiáltak (RAID 1-től RAID 5-ig). Az egyes szintek nem a fejlődési, illetve minőségi sorrendet tükrözik, hanem egyszerűen a különböző megoldásokat. A kezdeti 5 szinthez később hozzávették a RAID 6-ot. RAID 0-ként szokták említeni azt a változatot, ahol a lemezeket összefűzzük, azaz redundancia nélkül kapcsoljuk össze. Ezeken kívül használják még több RAID tömb egymásra építését is, a legelterjedtebb a RAID 10 (vagy RAID 1+0), RAID 01 (vagy RAID 0+1), illetve a RAID 50 (vagy RAID 5+0). A RAID alapötlete a lemezegységek csíkokra (stripes) bontása. Ezek a csíkok azonban nem azonosak a lemez fizikai sávjaival (tracks), amit az angol és magyar elnevezés különbözősége is jelez.

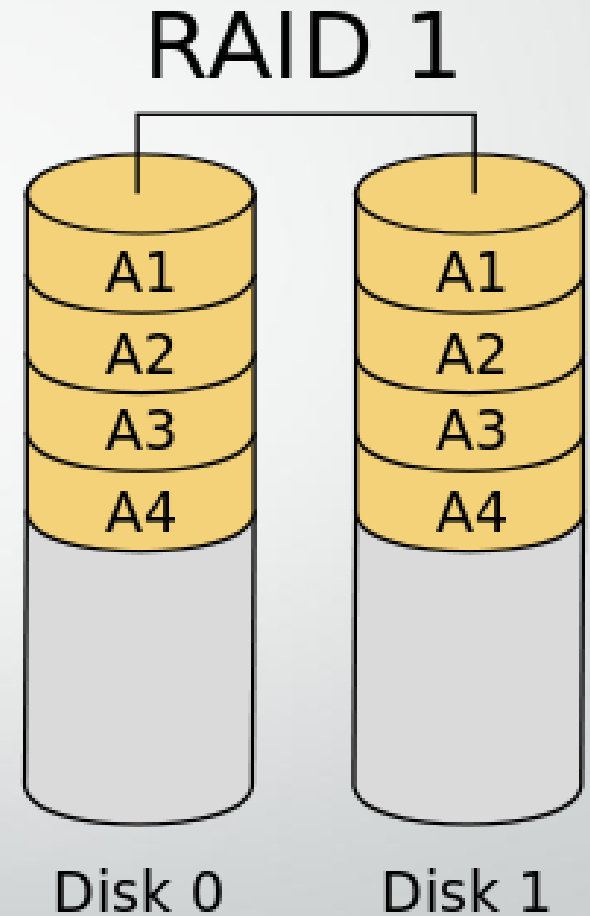
RAID 0

- A RAID 0 az egyes lemezek egyszerű összefűzését jelenti, viszont semmilyen **redundanciát nem ad**, így nem biztosít hibatűrést, azaz egyetlen meghajtó meghibásodása az egész tömb hibáját okozza.
- Mind az **írási**, mind az **olvasási műveletek párhuzamosítva** történnek, ideális esetben a sebesség az **egyes lemezek sebességének összege lesz**, így a módszer a RAID szintek közül a **legjobb teljesítményt nyújtja** (a többi módszernél a redundancia kezelése lassítja a rendszert).
- A megoldás lehetővé teszi különböző kapacitású lemezek összekapcsolását is, viszont a nagyobb kapacitású lemezekben is **csak a tömb legkisebb kapacitású lemezének méretét lehet**.



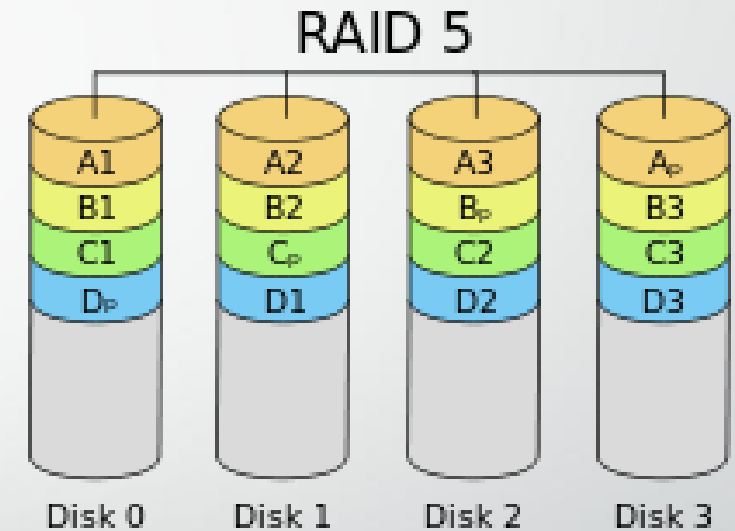
RAID 1

- A RAID 1 eljárás alapja az **adatok tükrözése** (disk mirroring), azaz az információk egyidejű tárolása a tömb minden elemén. A kapott logikai lemez a **tömb legkisebb elemével lesz egyenlő méretű**. Az adatok olvasása párhuzamosan történik a diszkekről, **felgyorsítván az olvasás sebességét**; az írás **normál sebességgel, párhuzamosan** történik a meghajtókon.
- Az eljárás **igen jó hibavédelmet biztosít**, bármely meghajtó meghibásodása esetén folytatódhat a működés. **A RAID 1 önmagában nem használja a csíkokra bontás módszerét.**



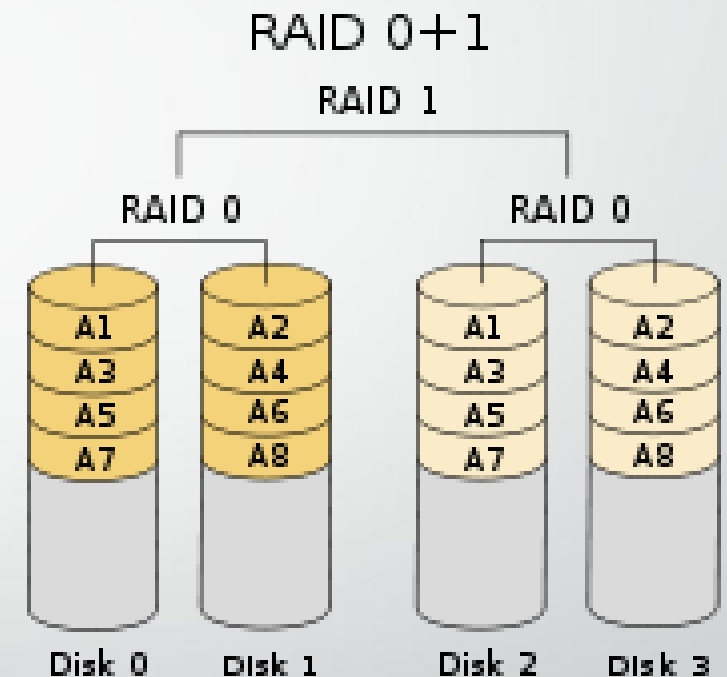
RAID 5

- A RAID 5 a paritás információt nem egy kitüntetett meghajtón, hanem **„körbeforgó paritás” (rotating parity)** használatával, egyenletesen az összes meghajtón elosztva tárolja, kiküszöbölve a paritás-meghajtó jelentette szűk keresztmetszetet.
- **Minimális meghajtószám: 3.** Mind az írási, mind az olvasási műveletek párhuzamosan végezhetőek. Egy meghajtó meghibásodása esetén az adatok sértetlenül visszaolvashatóak, a hibás meghajtó adatait a vezérlő a többi meghajtóról ki tudja számolni. A hibás meghajtót ajánlott azonnal cserélni, mert két meghajtó meghibásodása esetén az adatok elvesznek!
- A tárolható adatmennyiség:
 $xmin * (y-1)$ ahol:
x: lemezméret
y: lemezek száma



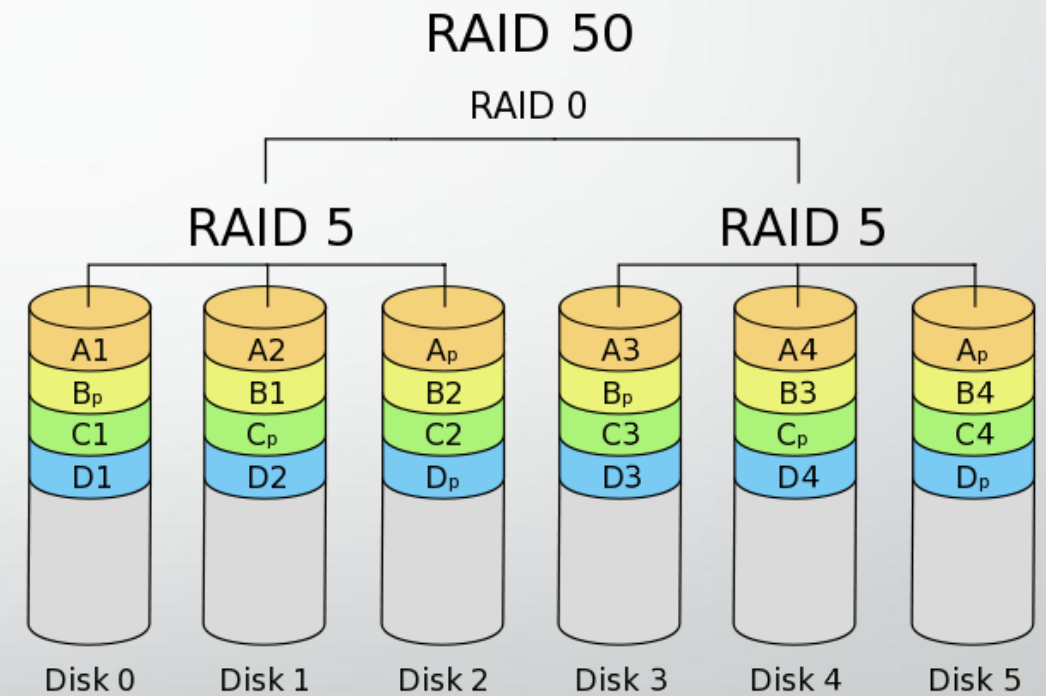
RAID 10 vagy 01 (nem ugyanaz a kettő!)

- Ez egy olyan hibrid megoldás, amelyben a RAID 0 által hordozott sebességet a RAID 1-et jellemző biztonsággal ötvözhetjük. Hátránya, hogy **minimálisan 4 eszközre van szükségünk**, melyekből 1-1-et összefűzve, majd páronként tükrözve építhetjük fel a tömbünket, ezért **a teljes kinyerhető kapacitásnak mindössze a felét tudjuk használni**. Mivel a tükrözés (RAID 1) a két összefűzött (RAID 0) tömbre épül, ezért egy lemez meghibásodása esetén **az egyik összefűzött tömb mindenképp kiesik, így a tükrözés is megszűnik**.



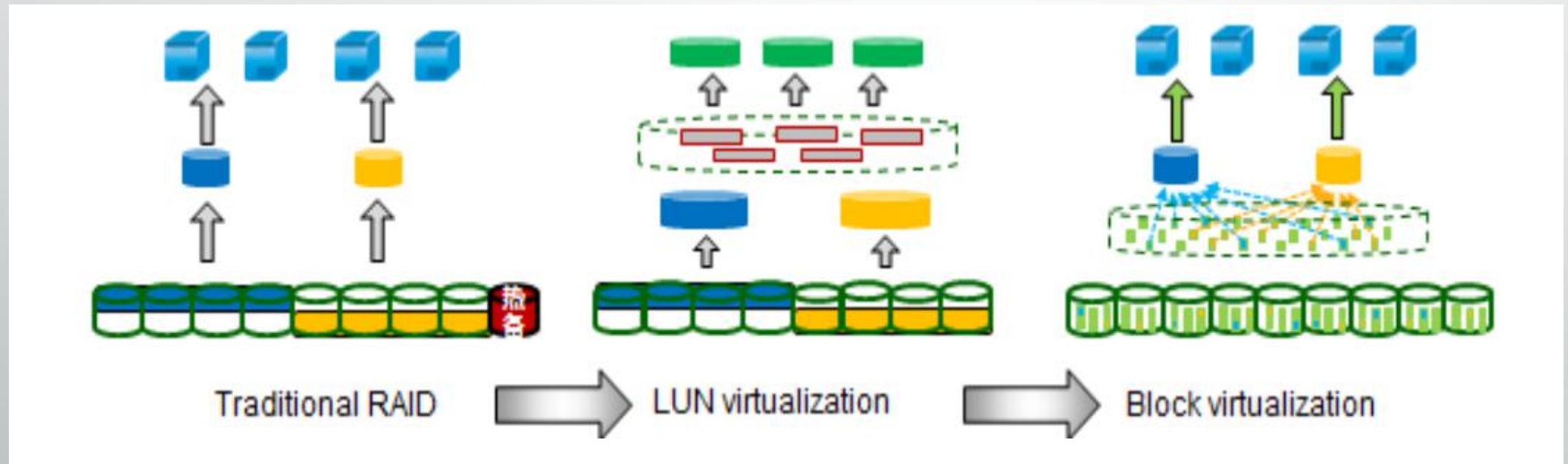
RAID 50

- A RAID 10-hez hasonló megoldás
- Két különböző RAID szint használata 1 tömbhöz

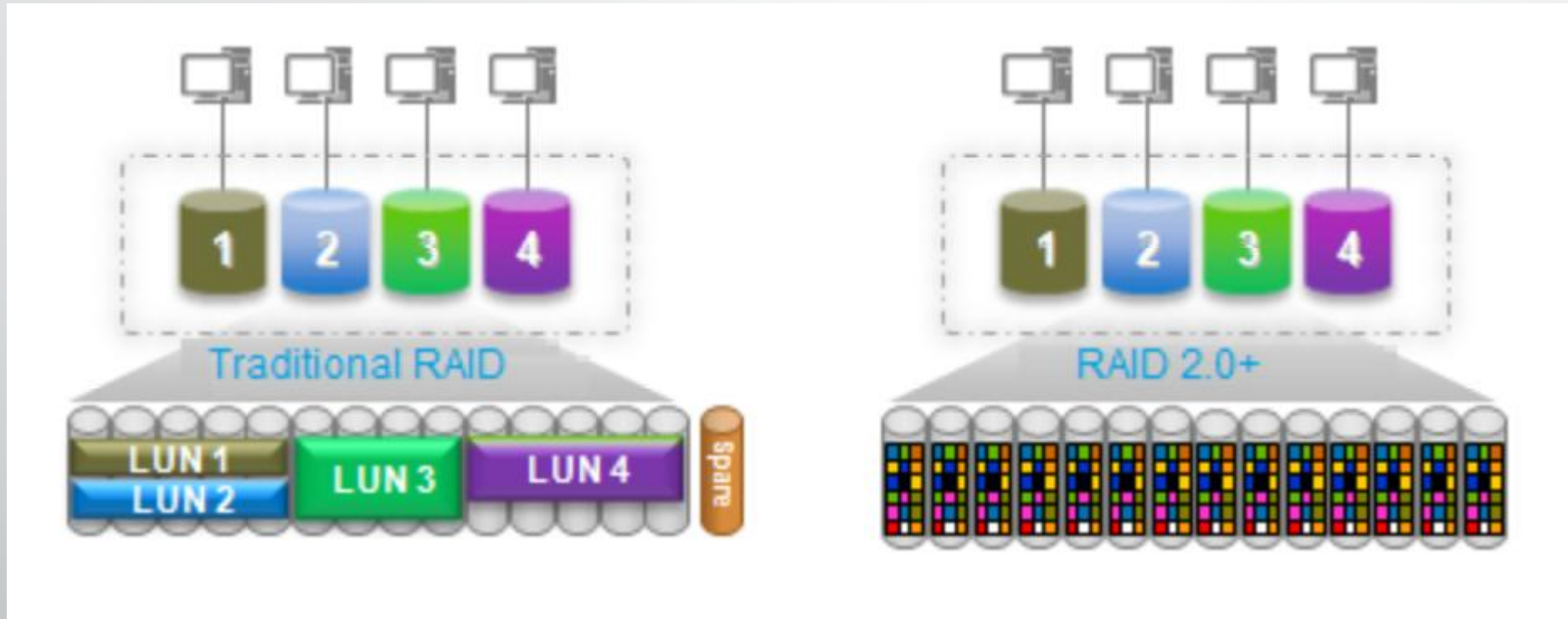


RAID 2.0+

- A RAID új megközelítése, nem a DISK az adategység
- Chunk-ok definiálása (alapértelmezetten 64 MB)

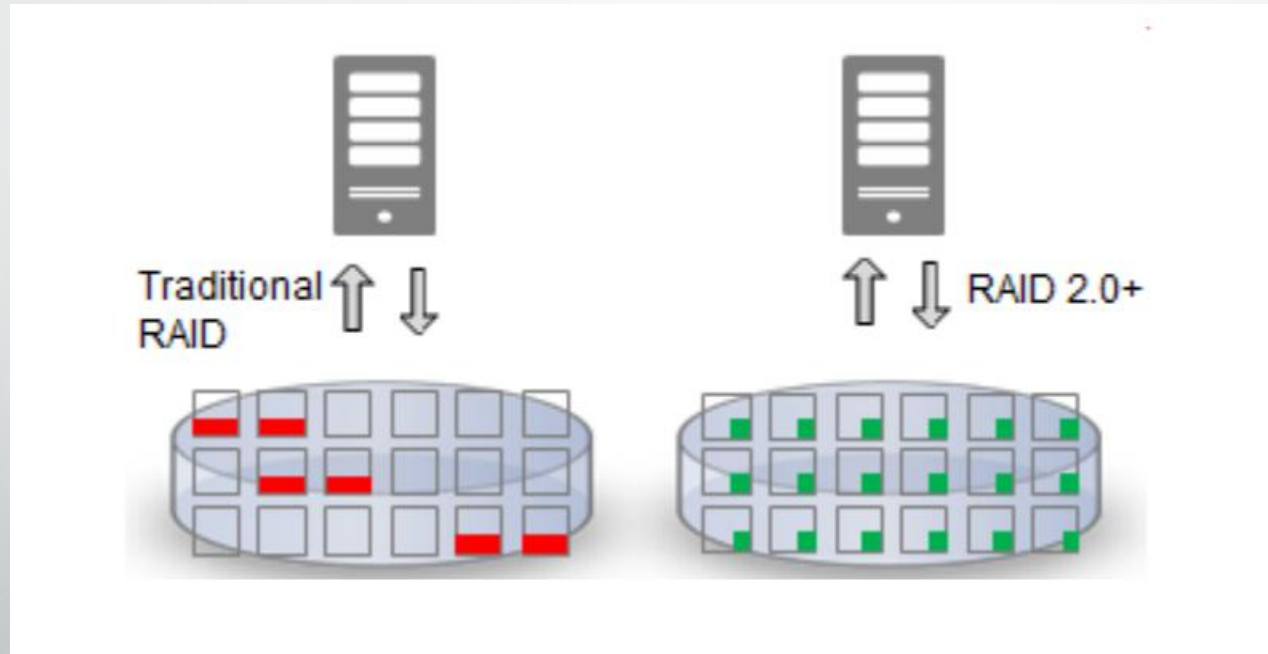


RAID 2.0+



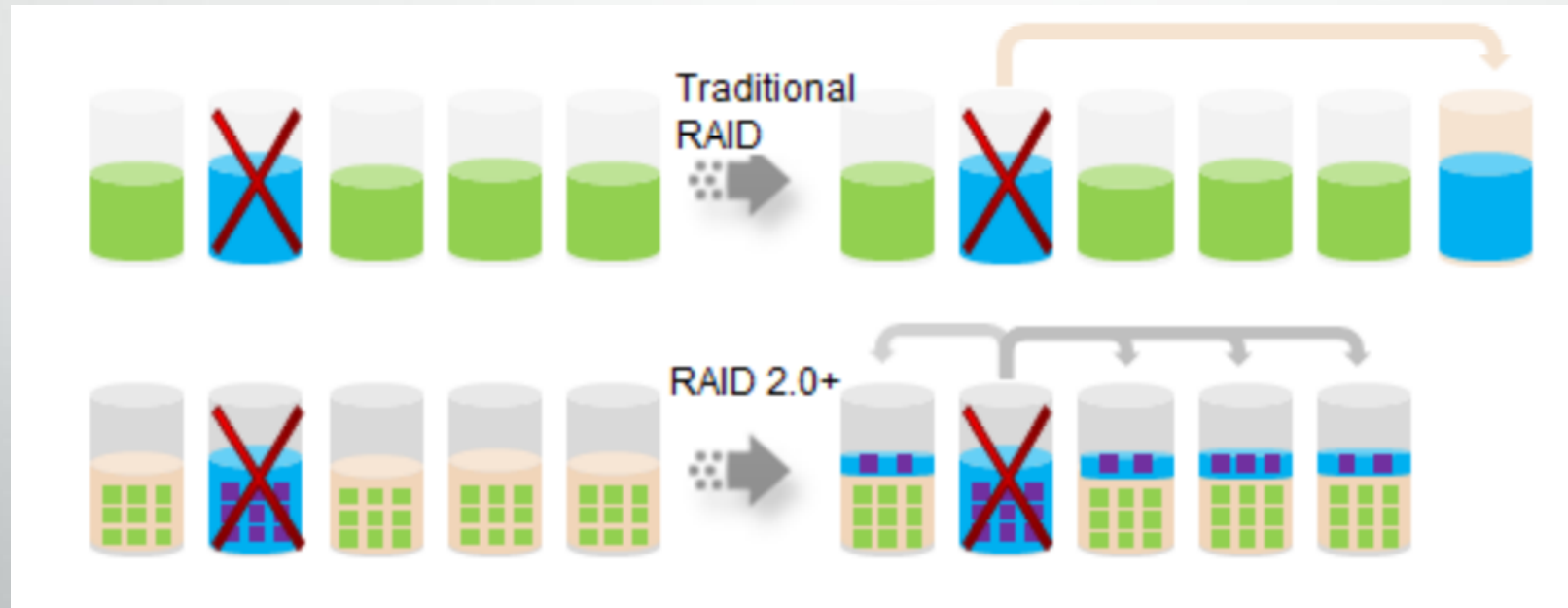
RAID 2.0+

- 2 rétegű virtualizáció
 - 1. a diskek managementje
 - 2. felsőbb rétegű LUN-ok managementje
- Disk groupok definiálása
- Azon belül CHUNK-ok definiálása



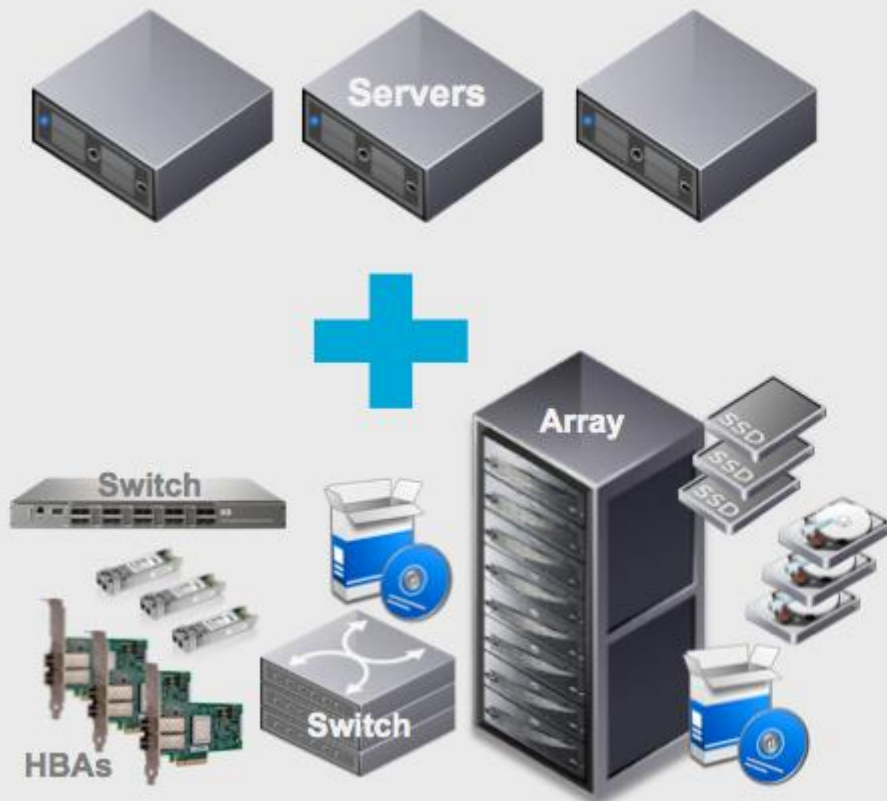
RAID 2.0+

- Hozzáférések TIER-be rendezése
- TIER₀ – SSD – nagy sebességű, de drága (smartcache)
- TIER₁ – SAS – Közepes sebesség – kompromisszum
- TIER₂ – NL-SAS – Kisebb sebesség nagy tárolókapacitás
- Újraépítés: Many-to-one vs Many-to-many



Új megközelítés (HCI)

Traditional Infrastructure



Hyper-Converged Infrastructure



VSAN

- Reqs: (Cons)
 - **\$2500** list per CPU socket and is sold separately from vSphere
 - Minimum 3 hosts
 - Minimum ESXi host licensing is **Essentials Plus** and above (Essentials Plus kits come with 3 licensed hosts and vSphere)
 - Dedicated 1 Gb NICs required (10GB recommended)
 - **Multicast** must be enabled on the VSAN VLAN
 - Minimum 1 HDD and 1 SSD on the first 3 hosts
- Azért ez nem olcsó!

VSAN

- De! (pros)
 - Szoftveres megoldás, könnyen frissíthető (új feature-ök)
 - Ugyanazon felületet alatt kezelhető
 - Könnyen bővíthető
 - Kevesebb eszköz -> kevesebb költség
 - Különböző policy-k (védelem (protection, security), performance)