Párhuzamos és Grid rendszerek (4. ea)

Elosztott fájlrendszerek

Szeberényi Imre
BME IIT
<szebi@iit.bme.hu>

Elosztott fájlrendszerek

- Nagyméretű klaszterekhez
- Földrajzilag is elosztott rendszerekhez
  - NFS
  - AFS, CODA, InterMezzo
  - Lustre, SFS
  - GFS
  - GlusterFS
  - OCFS
  - Hadoop
  - Gfarm file system
  - Google file system
  - GPFS
  - Parallel Virtual FS
  - QFS
  - CernVMFS
  - Nimbus,
  - S3

http://en.wikipedia.org/wiki/List_of_file_systems

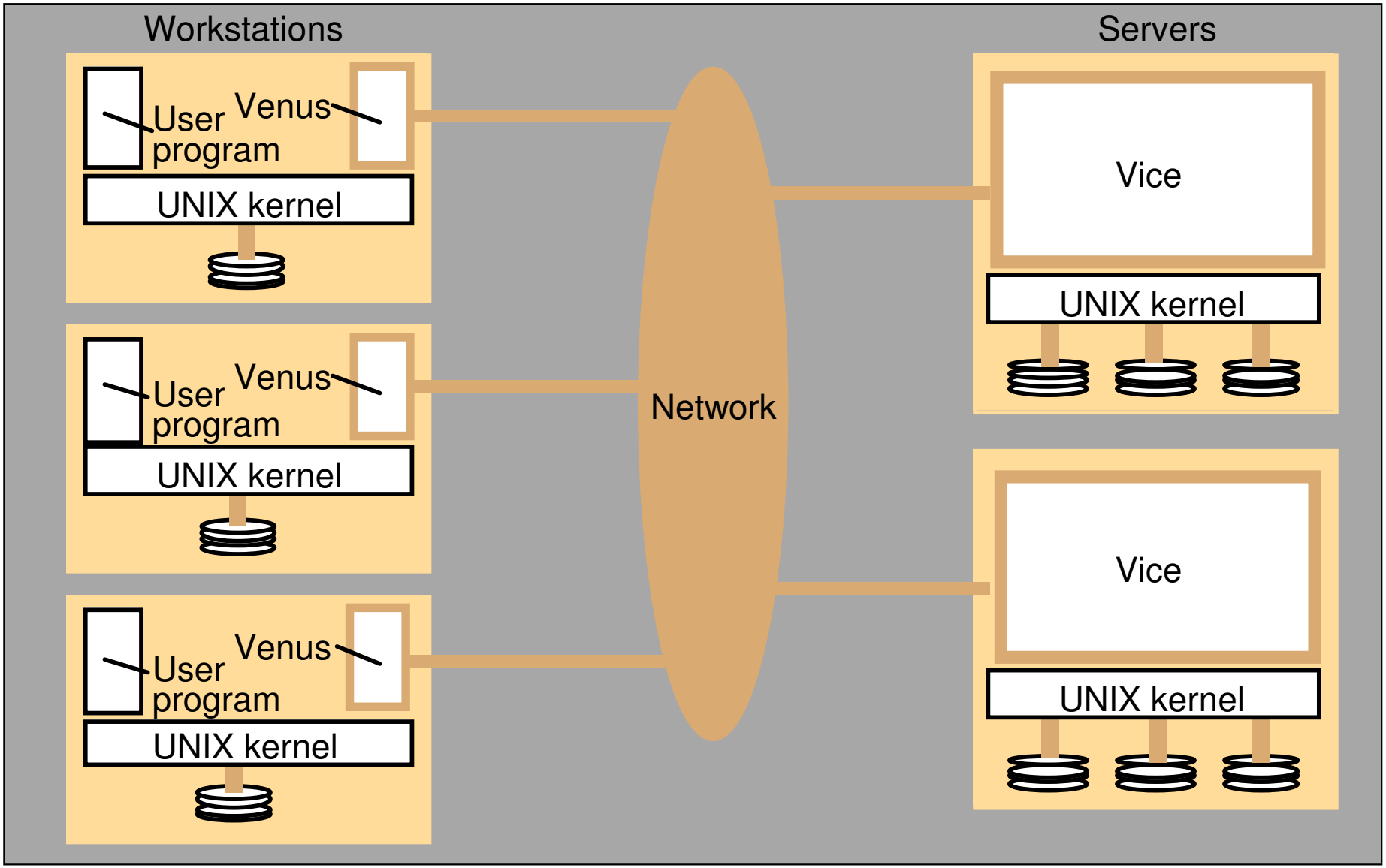
AFS (Andrew File System)

- Elosztott fájlrendszer, ami fájlok megosztására alkalmas lokális és távolsági hálózaton.
- Transzparens fájlhozzáférést biztosít.
- Az NFS-hez hasonló, annak alternatívájaként jött létre.
- Ma az OpenAFS számos UNIX, LINUX, WinX platformon elérhető.

AFS történelem

- Carnegie Mellon Egyetemen 1984-ben fejlesztették ki UNIX környezetben. Ma azonban nem csak UNIX változat létezik.
- A fő cél az volt, hogy az egyetemi korlátozott sávszélességű hálózaton hatékony fájlelérést tegyenek lehetővé.

AFS processzek



AFS alapfogalmai

- Cellák
- Kötetek
- Tokenek
- Cache menedzser
- Fájl védelem
- Fájl névtér

AFS cella

- Egy AFS cella alá azok a szerverek tartoznak, melyek adminisztrációja közös, és az AFS felé egyetlen közös fájlrendszert alkotnak.
- Tipikusan az egy domain név alá tartozó gépek egy AFS cellát alkotnak.
- Általában a domain név valamilyen változata a cellanév.
- A munkaállomások a felhasználókról a cella szervertől kérnek információkat.

Kötetek

- A diszkterületet az AFS további részekre, osztja ezek az AFS kötetek.
- Az AFS kötet egy tárolóegység ami a fájlok és katalógusok adatait tárolja.
- Az AFS kötettek fájlok formájában jelennek meg a befogadó operációs rendszerben, így azok könnyen átmozgathatók, akár másik gépre is.

Tokenek

- **Az AFS** nem használja a UNIX felhasználói azonosítóját (UID). Ha ezt tenné, akkor minden UNIX gépen azonos UID kiosztásnak kellene lennie, mint az NFS-nél.
- Az azonosításhoz AFS tokent alkalmaznak, ami egy egyedi azonosítást tesz lehetővé.
- Egy token adott ideig (24 óra) érvényes.

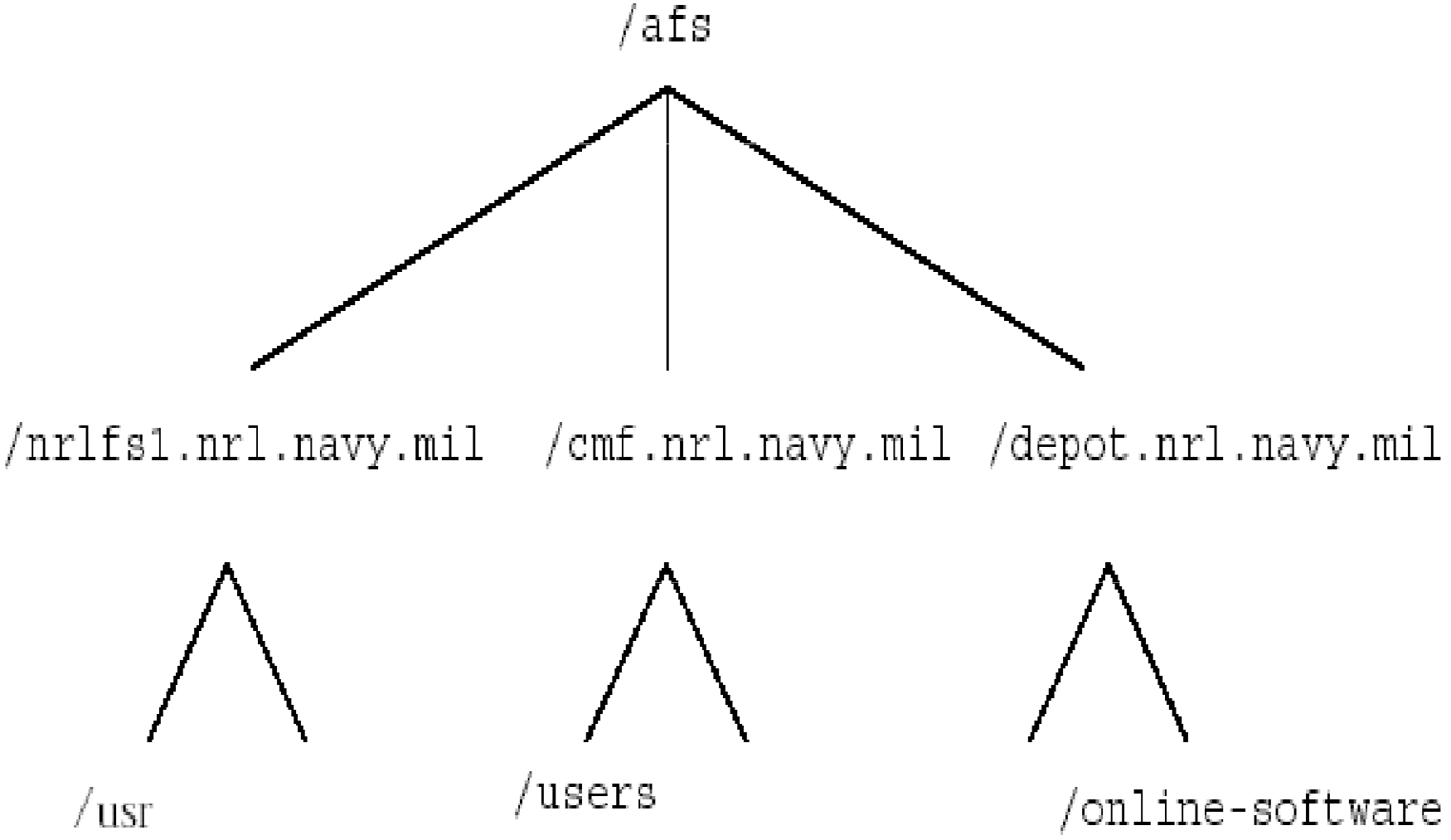
Cache menedzser

- A korlátozott sávszélesség miatt a működés központi eleme a cache, ahova az éppen használt fájlok letöltődnek.
- A cache menedzser feladata a cache-ben tárolt információk frissítése, karbantartása.
- Amennyiben a cache-ben tárolt fájlrészlet változik, úgy azt vissza kel tölteni a szervere.
- Ha a szerveren változik meg a fájl, akkor arról CallBack technikával értesít minden cache-t.

Védelem

- A védelmi mechanizmus némileg eltér az alap UNIX védelmi rendszertől.
- A UNIX 3x3-as védelmétől pontosabban szabályozható ACL (Access Control List) segítségével.
  - Lookup (l)
  - Insert (i)
  - Delete (d)
  - Administer (a)
  - Read (r)
  - Write (w)
  - Lock (k)

Névtér


/afs
/nrlfs1.nrl.navy.mil
/cmf.nrl.navy.mil
/depot.nrl.navy.mil
/usr
/users
/online-software

Névtér /2

- UNIX-hoz hasonló hierarchikus struktúra
- Az AFS gyökér névtér rendszerint a /afs. Az alatta levő szinteket a cellák képviselik.
  - adminisztratív domain
    - AFS szerverek halmaza egy cégnél, egyetemen, laborban stb.
  - Lokális cella
    - alapértelmezett cella, amihez az adott munkaállomás csatlakozik.
  - idegen cella
    - más cella az AFS névtérben

Venus és Vice

- Venus
  - AFS kliens által futtatott processz.
- Vice
  - AFS szerver által futtatott processz.

Fájl műveletek

- A kliens munkaállomás a szerverrel csak az open/close műveletek kiszolgálásakor kommunikál.
- A fájl megnyitásakor a Venus a teljes fájlt a cache-be tölti, és a fájl lezárásakor írja azt vissza.
- Az adatok olvasását/írását a lokális másolaton a kernel végzi.
- A Venus a katalógusokat és a szimbólikus linkeket is a lokális gyorsítótárban tárolja.
- A fenti gyorsítótárazási mechanizmus alól a katalógusok módosítása a kivétel, aminek a végrehajtásáért a közvetlenül szerver a felelős.

Fájl megosztás

- Lokális fájlokhoz hasonlóan.
  - nincs külön mount
  - nem kell belépni a mási gépre
  - csak jogosultság kell
- A /afs katalógus alatt tetszőleges cella fájljai elérhetők.
  - Természetesen megfelelő jogosultsággal.
  - Csak a megfelelő útnevet kell hozzá tudni.
- A fájlmegosztást nem korlátozza a földrajzi távolság, vagy az adott operációrendszer típusa.

Login és authentikáció

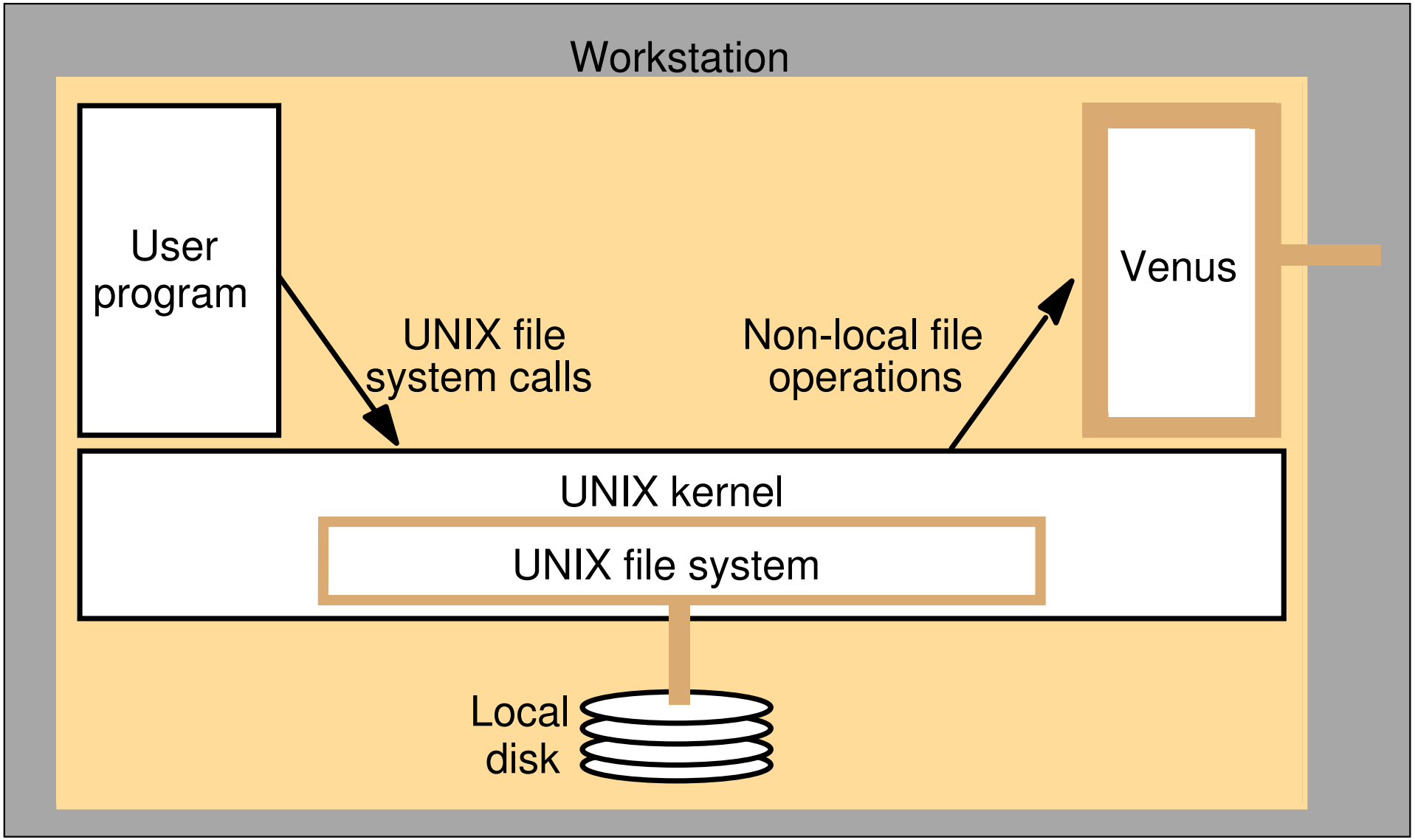
1. Bejelentkezéssel együtt token is generálódik
2. Külön kell tokent generálni.
   - klog,

```
Tokens held by the Cache Manager:
   --End of list--
szebi:$ /usr/afs/bin/klog
Password:
szebi:$ /usr/afs/bin/tokens
Tokens held by the Cache Manager:
User's tokens for afs@bme.hu [Expires Apr  7 00:47]
   --End of list–
........
User's tokens for afs@cern.ch [Expires Apr  7 00:53]
User's tokens for afs@bme.hu [Expires Apr  7 00:47]
```

Megvalósítás

- A kliens oldali programok a szokásos módon, rendszerhívással kezelik az állományokat.
- A távoli fájlok megnyitásakor Venus processzhez jut a kérés, amit az lebont az útnév alapján.
- Az alacsonyszintű I/O kezelését a befogadó operációs rendszer végzi. A gyorsítótár a lokális gép diszkjén jön létre.

Rendszerhívás szint



AFS parancsok

Az AFS parancsok 3 csoportba oszthatók:

- Fájlszerver parancsok (fs)
  - AFS szerver információk listázása
- Védelmi parancsok(pts)
  - ACL listák létrehozása
- Authentikációs parancsok
  - klog, unlog, kpasswd, tokens

AFS előnyei

- Gyorsítótárazásból fakadó előnyök:
  - Lényegesen csökkenti a hálózati forgalmat.
  - Alacsonyabb sávszélességnél is jól használható.
- Helyfüggetlenség:
  - Az AFS a földrajzi helyet a szerver oldalon rendeli fájlnévhez. Így a névtér helyfüggetlen.
- Skálázhatóság:
  - A rendszer tervezési fázisában igen nagyra (~10000 kliens) tervezték. A kliens/szerver arányt pedig 200:1-re. Mindkét értéket túlteljesíti.

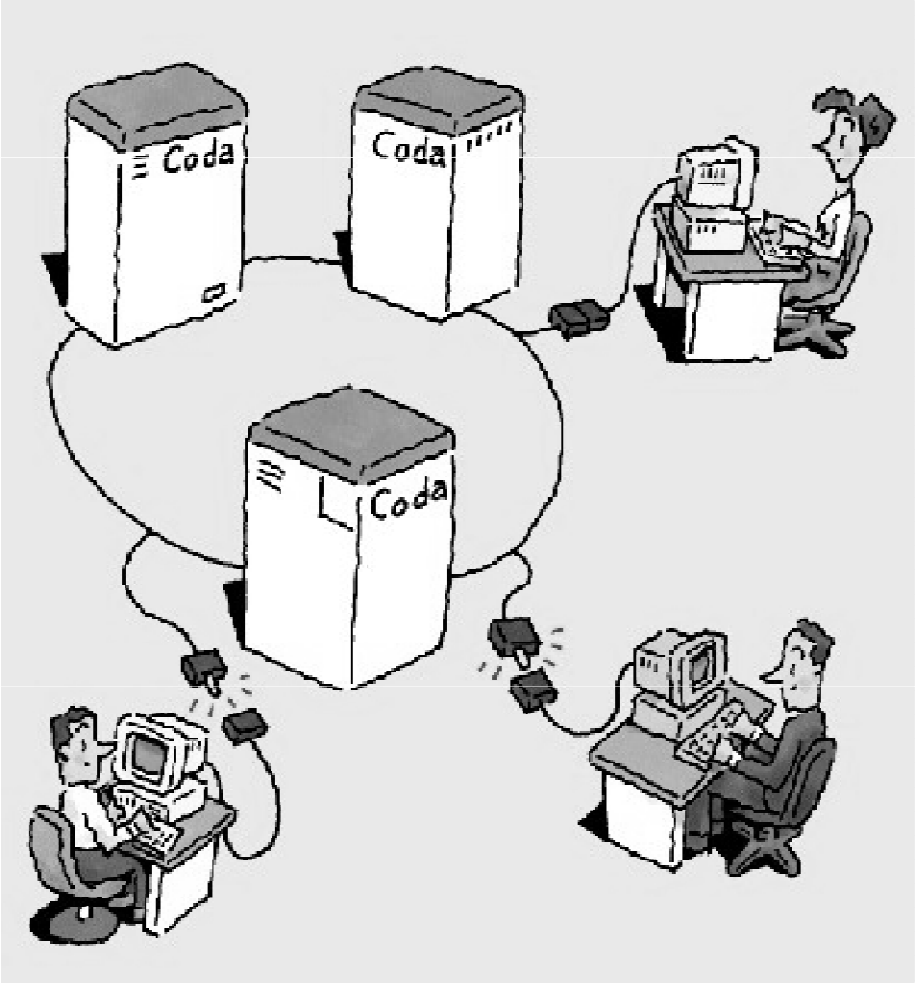
AFS előnyei /2

- Single systems image (SSI):
  - Egy fájlszerver kialakítása lényegesen egyszerűbb, mint NFS-sel.
- Fokozott biztonság:
  - Kerberos használata
  - ACL használata
- Fájlok egyszerű megosztása
- Egyszerű rendszer menedzsment
- Robosztus
- Replika lehetőség.

AFS hátrányai

- Minden munkaállomásra installálni kell.
- Háttérszerver komplexitása.
- Tokenek érvényességének lejártából fakadó gondok.

CODA



- AFS-2 leszármazott
- disconnected
- replica
- Kerberos-like
- 87 óta fejlesztik.
  2009 óta csend.

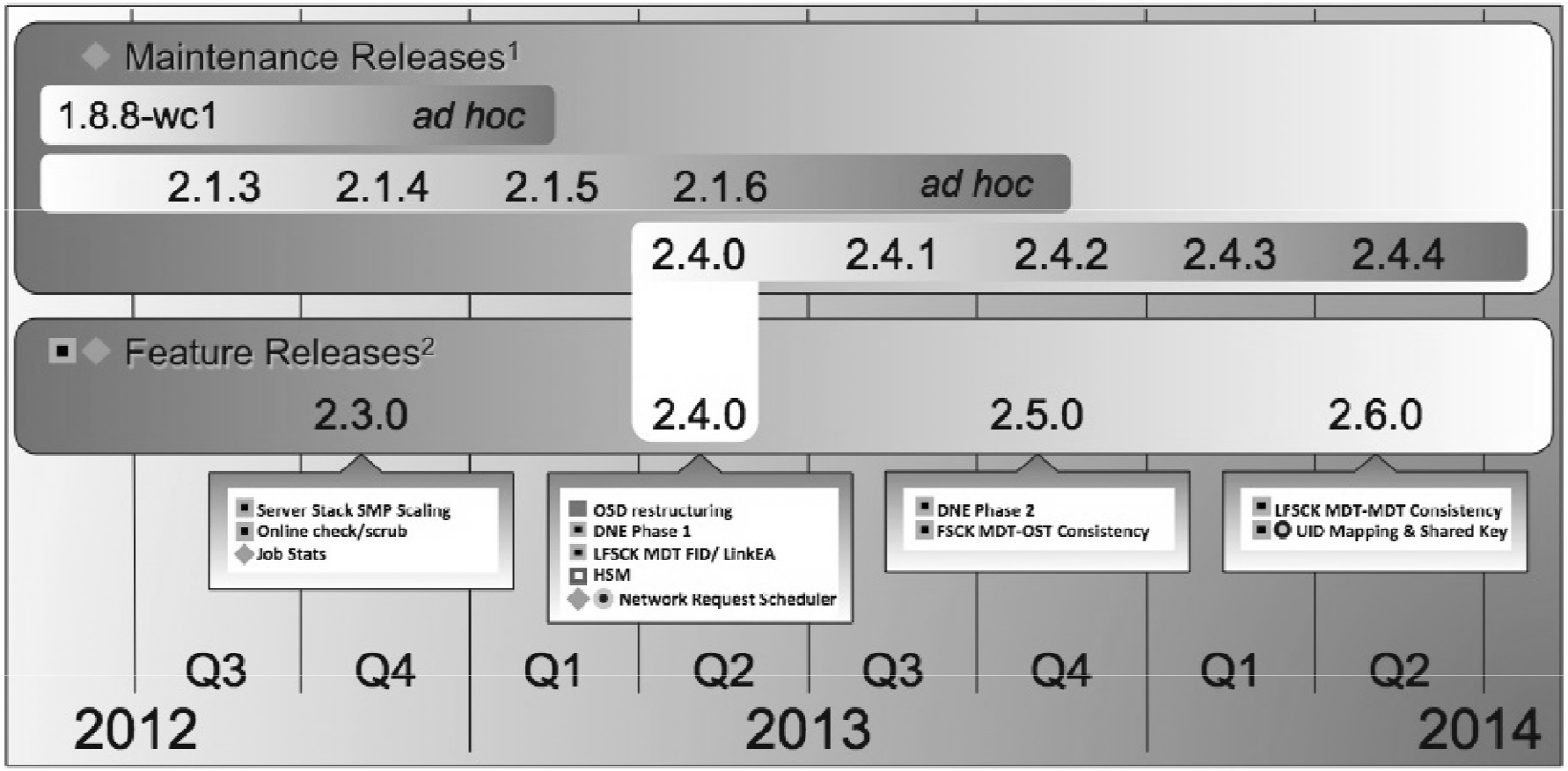
Lustre

- Objektum-orientált elosztott fájlrendszer.
- Jól skálázható.
- Nagyméretű klaszterekhez, és nagy fájlokhoz tervezték.
- Lustre 2007-től GPL.
- SUN ZFs
- 15 a top 30 szupergépből Lustre-t hazsnál

Lustre történelem

- 1999 by Carnegie Mellon University
- Lustre 1.0 2003-ban (Cluster File Systems)
- 2007-ben SUN felvásárolta a CFS-t.
  - Open source software (RedHat, SUSE, …)
- 2010-ben Oracle felvásárolta az SUN-t
  - 2011-ben 1.8 supportot megszüntette (számos szervezet folytatta)
  - Whamcloud, OpenSFS, EOFS,
- 2012-ben Whamloud-ot megvette az INTEL

Community Lustre Roadmap



Sponsor for Intel Development and Releases: ORNL, OpenSFS, LLNL, Intel

Third Party Development: CEA, Xyratex, Indiana University

[1] Maintenance releases focus on bug fixes and stability. Updates to the current version are made at 3 month intervals. Updates to past versions will be made on an ad hoc basis.
[2] Feature releases focus on introducing new features. New release versions are expected at 6 month intervals. New maintenance versions from the feature release stream are anticipated at 18 month intervals.

Lustre architektúra

- Három fő funkcionális egysége van:
- Metadata szerver (MDS), ami a fájl neveket, katalógusokat, védelmi kódokat és egyéb metaadatot tárol.
- Object storage szerverek (OSS), melyek az adatokat tárolják.
- Kliens ami az adatokat felhasználja, létrehozza.

Lustre architektúra /2

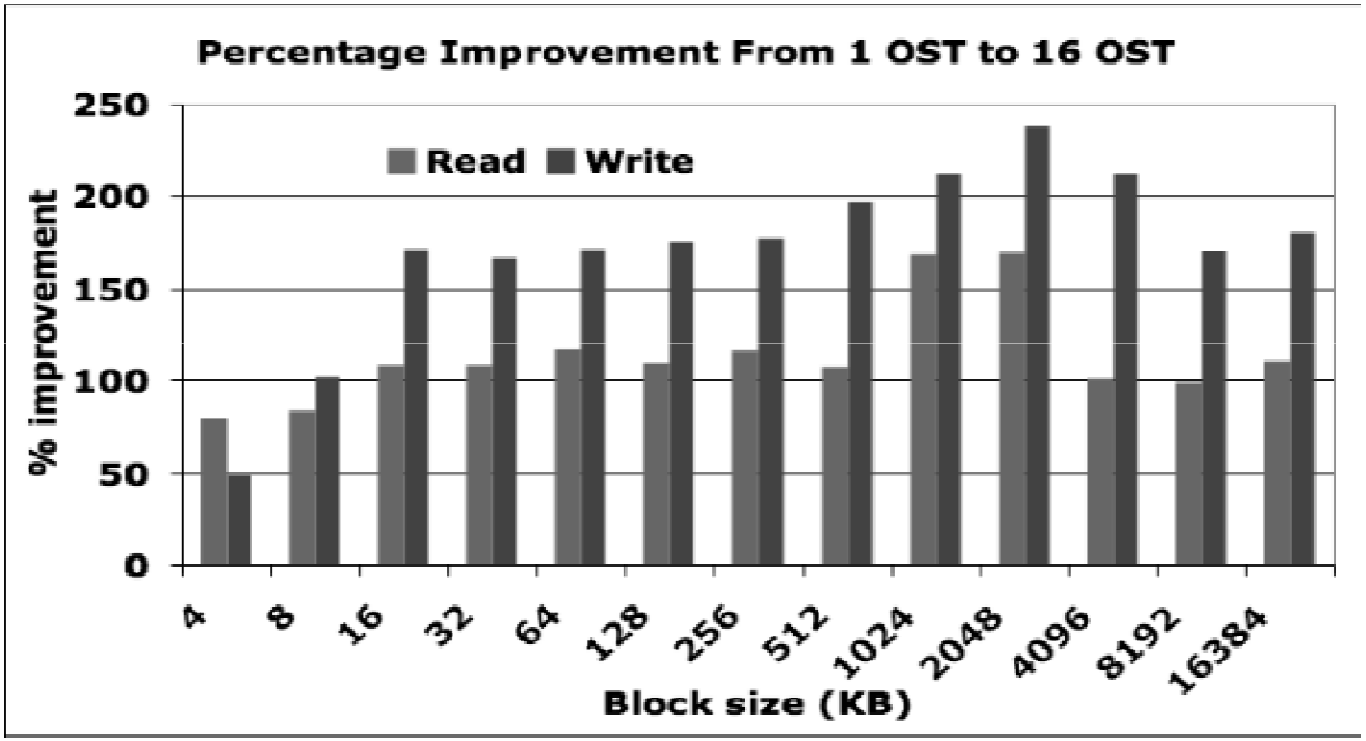
- Az adatok logikai kötetmenedzsmenttel ellátott RAID tárolókban tárolódnak, amit az OSS és az MDS dedikált módon használ.
- Jelenleg egy módosított ext4 fájlrendszer a logikai tároló. ZFS support (béta)
- Amikor egy kliens fájlt akar elérni, először az MDS-ben meg kell keresnie.

Lustre architektúra /2

- A fájl egyes darabjai több OSS-en tárolódhatnak, ami a kliens és az OSS között szűk keresztmetszet kialakulását gátolja.
- A kliensek nem módosítják közvetlenül az OSS-ben tárolt adatokat, hanem ezt a OSS-re bízzák, szemben a GFS megoldásával.
- Ez a módszer növeli a megbízhatóságot és a hibatűrést.

Skálázhatóság teljesítmény

- TOP 500-as lista tetején (Titan is)
- Skálázhatóság, nagy rendelkezésre állás
- Üzleti szupport (Oaracle-n kívül mindenki)



S. Saini, J. Rappleye, J. Chang, D. Barker, P. Mehrotra, R. Biswas: I/O Performance Characterization of Lustre and NASA Applications on Pleiades

ZFS

- Sun: 2001-2004, 2005-től Solaris része
- Zettabyte File System
- 128 Bit - extra nagy kapacitás
- Pool elvű tárolók – elosztott sávszélesség és kapacitás
- Tranzakció kezelés – Copy on Write
- Snapshots (ro) és klónozás
- Adat integritás – ellenőrző összeg (külön)

ZFS kapacitások

- 1 ZB = 10^{21} 1 ZiB (zebi B) = 2^{70}
- 2^{64} shnapshot
- 2^{48} fájl / dir
- 2^{64} byte / fájl
- 2^{78} byte / pool
- 2^{64} device / pool
- 2^{64} pool / system

Hogyan kapunk diszk címet

Hagyományos FS esetén:

- FS(1): filename → object (inode)
- FS(2): object → volume LBA
- VM: volume LBA → array LBA
- RAID: array LBA → disk LBA

Sok réteg, szigorú szeparáció, eltérő gyártók

Hogyan kapunk diszk címet (2)

ZFS esetén:

- ZPL: filename → object
- DMU: object → DVA
- SPA: DVA → LBA

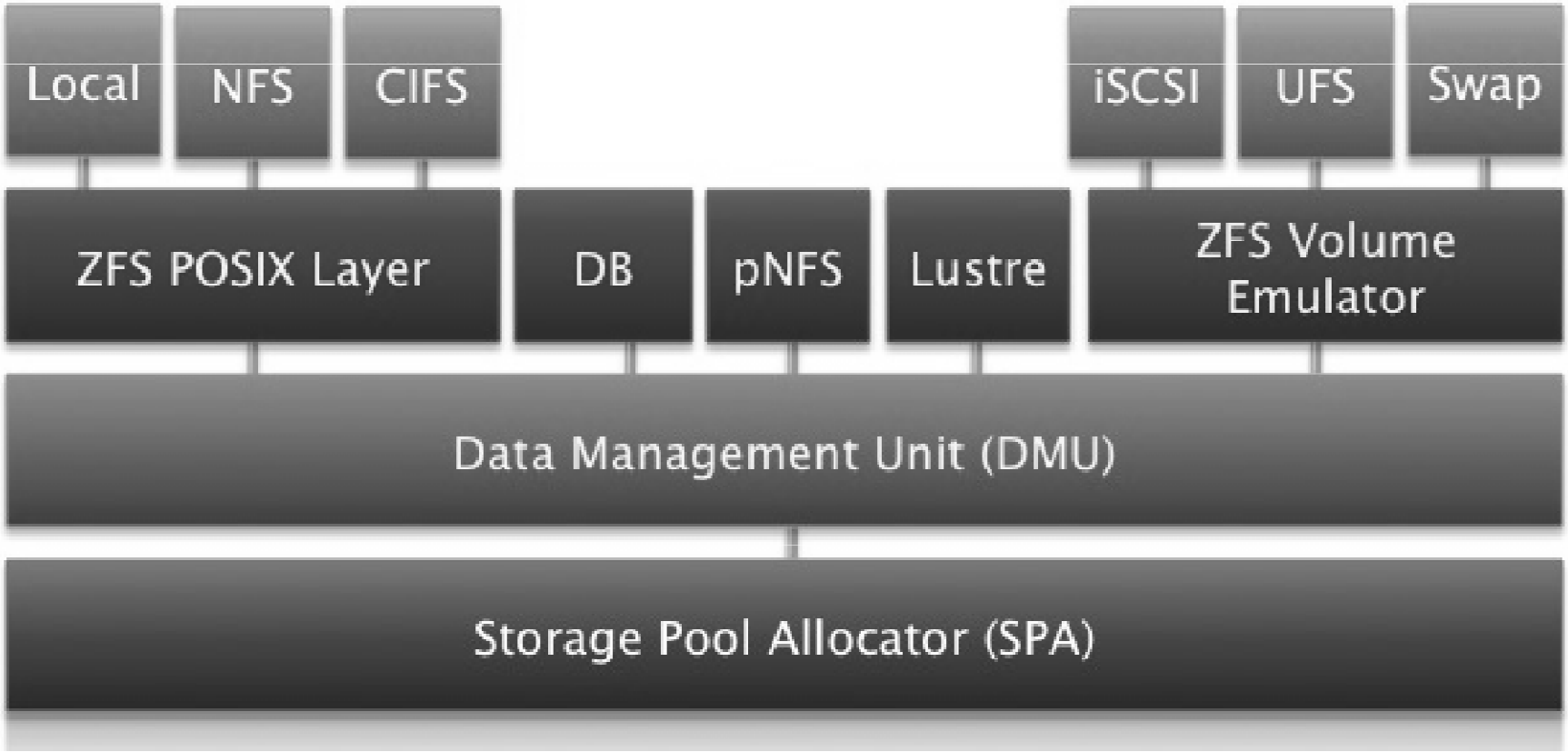
ZPL: ZFS POSIX layer (standard syscall)

DMU: Data Management Unit (transactional object store)

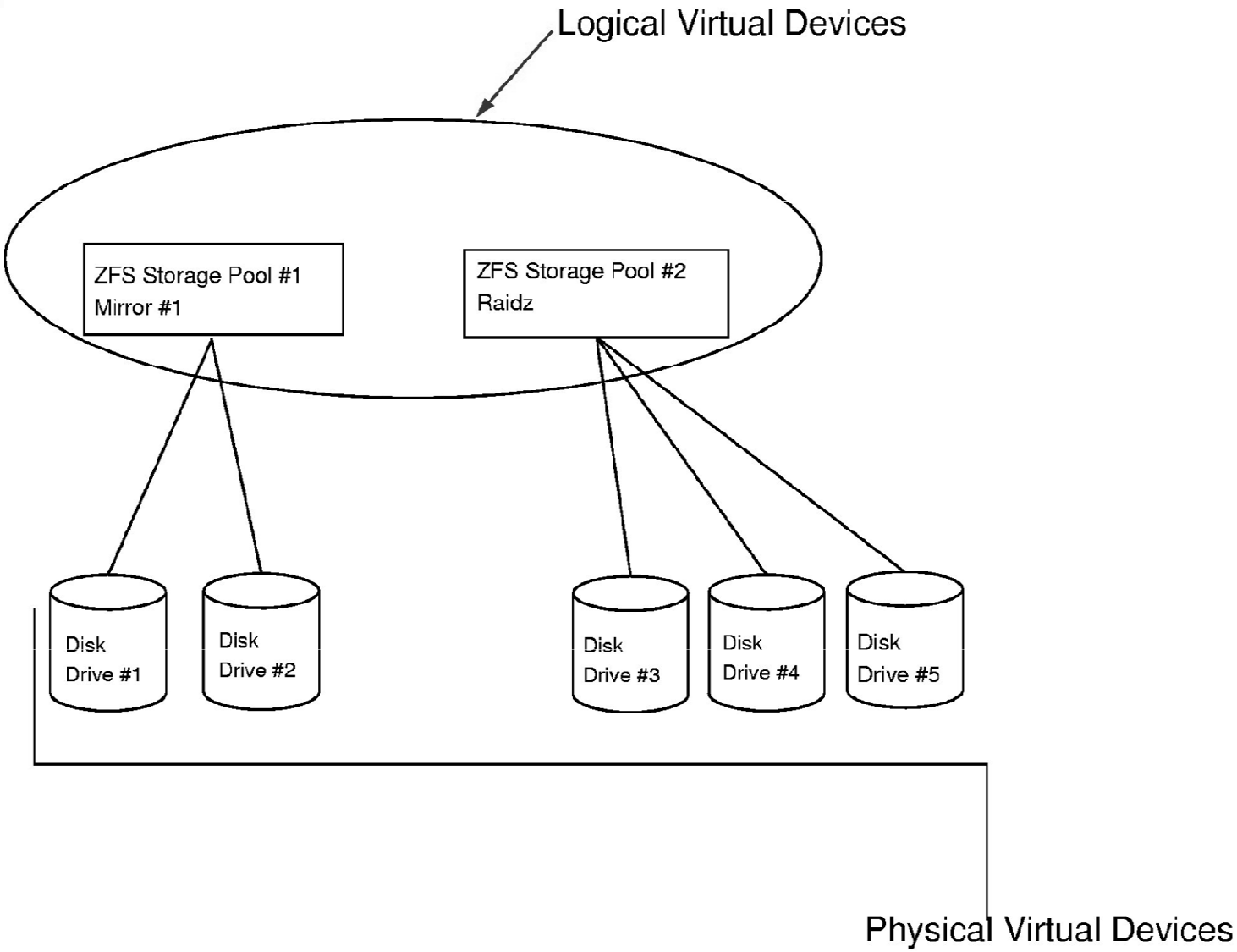
DVA: Data Virtual Address (vdev + offset)

SPA: Storage Pool Allocator (blokk alloc, data transform)

Architektúra

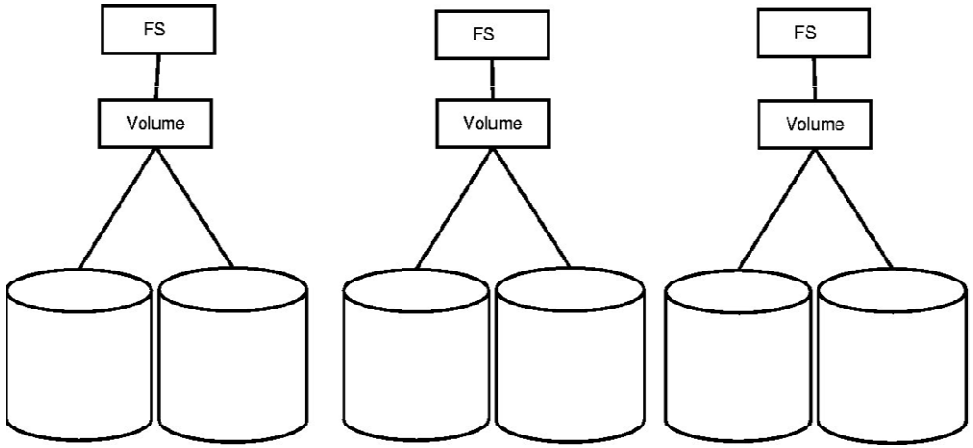


ZFS – VM hasonlóság

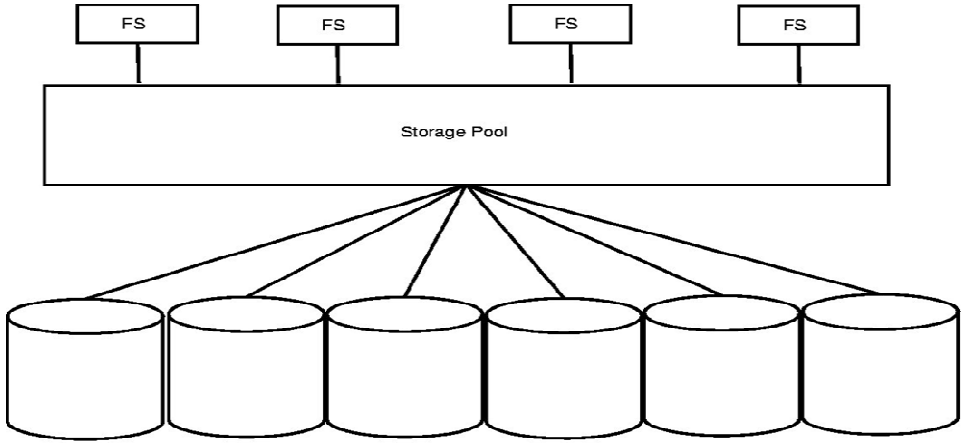


Kötet és Pool

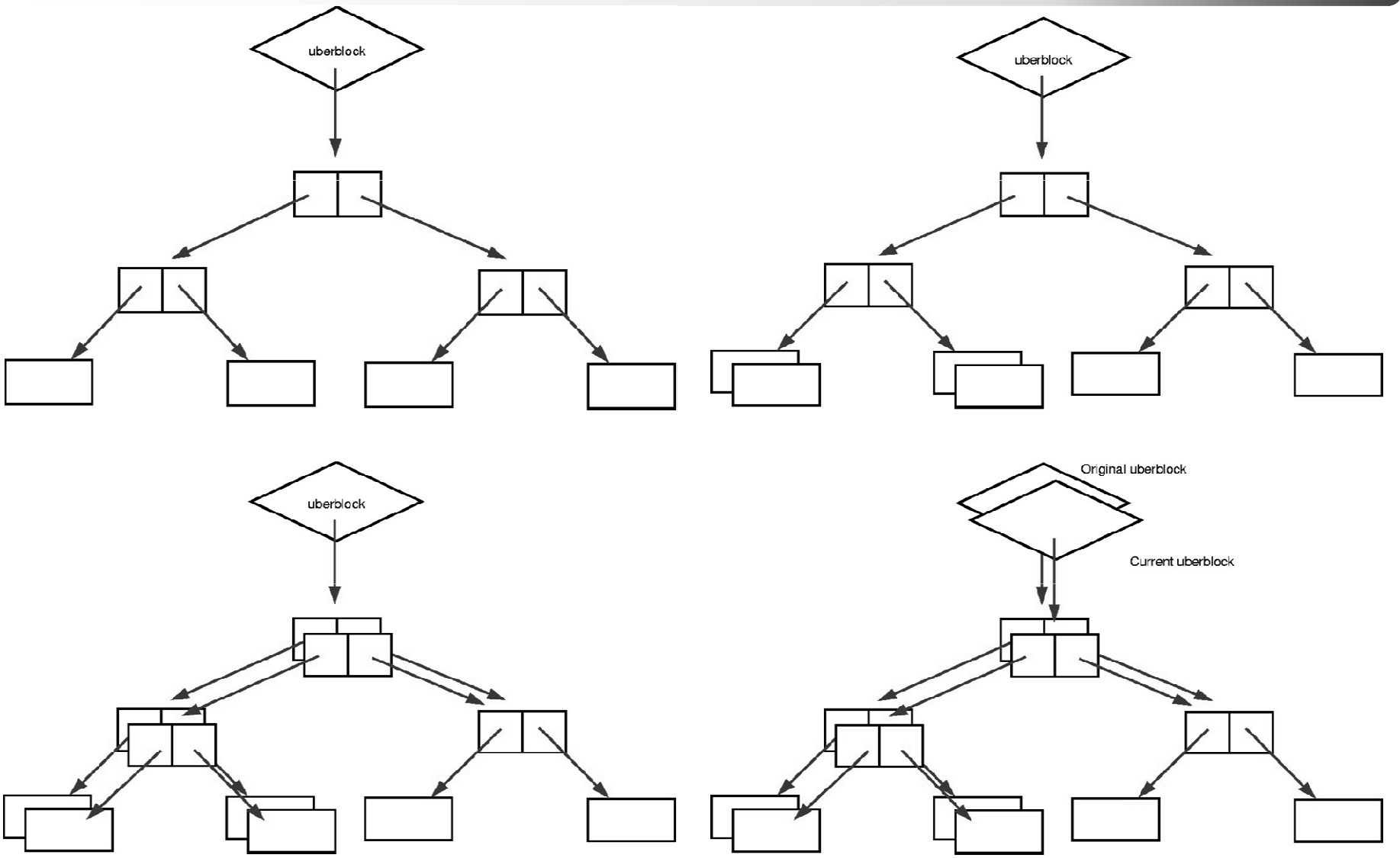
- Hagyományos kötet kezelés



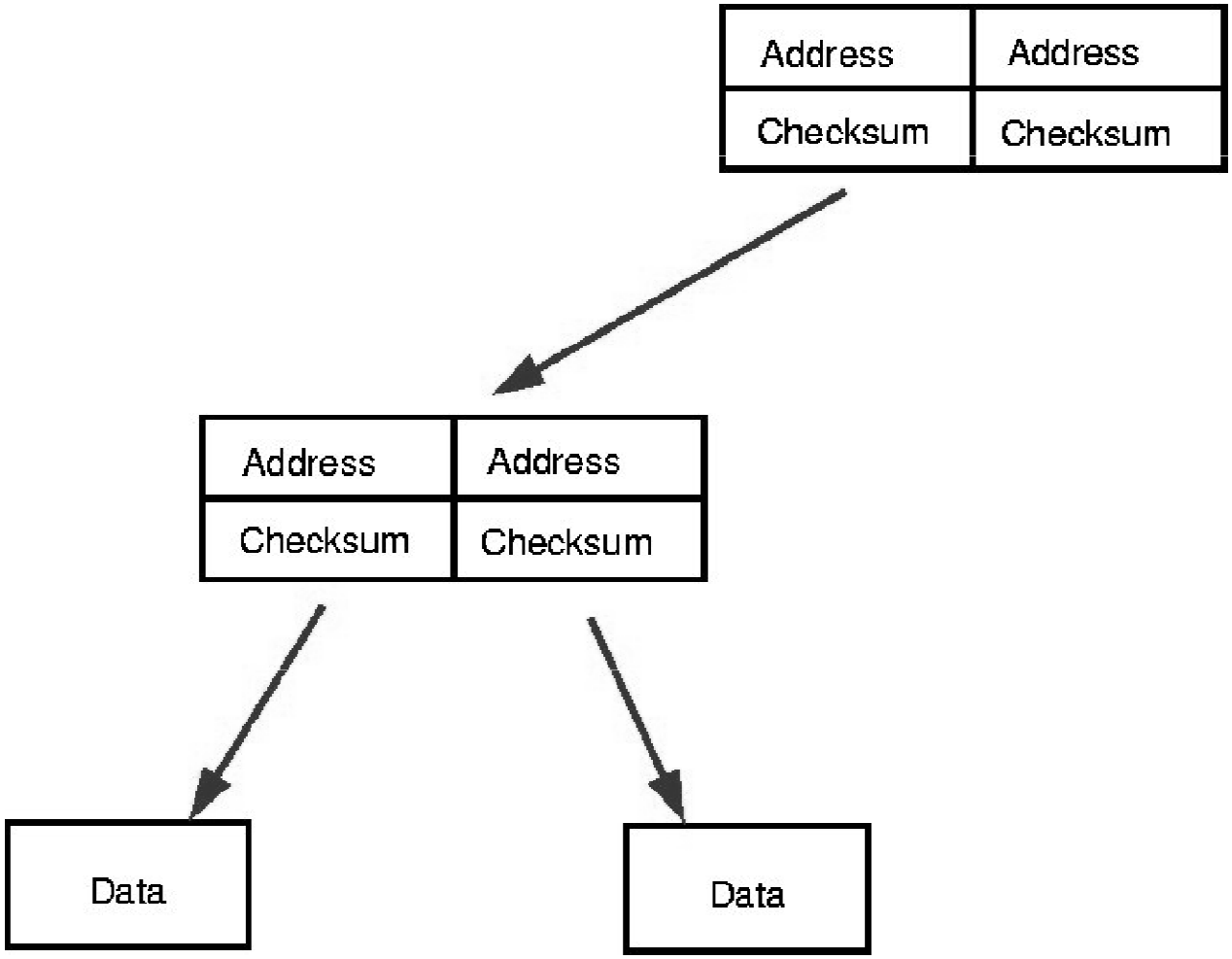
- Pool:
  - Automatikus méretezés
  - osztott sávszélesség



ZFS - Copy on Write (COW)



ZFS – ellenőrző összeg



ZFS elérhetősége

- OpenSolaris, OpenIndiana
- BSD, OSX
- Linux: CCDL és a GPL üti egymást
- Linux FUSE
- Native ZFS (Gentoo, Ubuntu)

http://en.wikipedia.org/wiki/ZFS

GlusterFS

- Célkitűzés FUSE alapokon megvalósítani elosztott fájlrendszert.
- A céget 2011-ben megvette a RedHat.
- Azóta a közösség láthatóan halódik

CernVMFS

- HTTP
- http cache
- alapvetően SL, de kliens több Linux változatra