Operációs rendszerek

Elosztott rendszerek

Elosztott rendszerek

- Az elosztott rendszer:
  - autonóm műveletvégző egységek,
  - összeköttetés: kommunikációs csatornán keresztül,
  - komponensek összehangolásával feladatok közös megoldása,
  - a megosztott erőforrások növelik a hatékonyságot.

Elosztott rendszerek

- Problémák:
  - a szoftver (párhuzamossági probléma, közös rendszer képzése, OPR feladata),
  - a hálózat (sávszélesség, túlterhelés),
  - a biztonság (titkos adatok, károkozás).

Elosztott rendszerek

- Alapvetően kétféle cél:
  - sok felhasználó; ”egymás melletti” munka végzése és kapcsolattartása (hálózati vagy multikomputeres rendszerek, lazán csatoltak),
  - részfeladatok párhuzamos, maximális sebességgel való megoldása (multiprocesszoros vagy párhuzamos rendszerek, szorosan csatoltak).

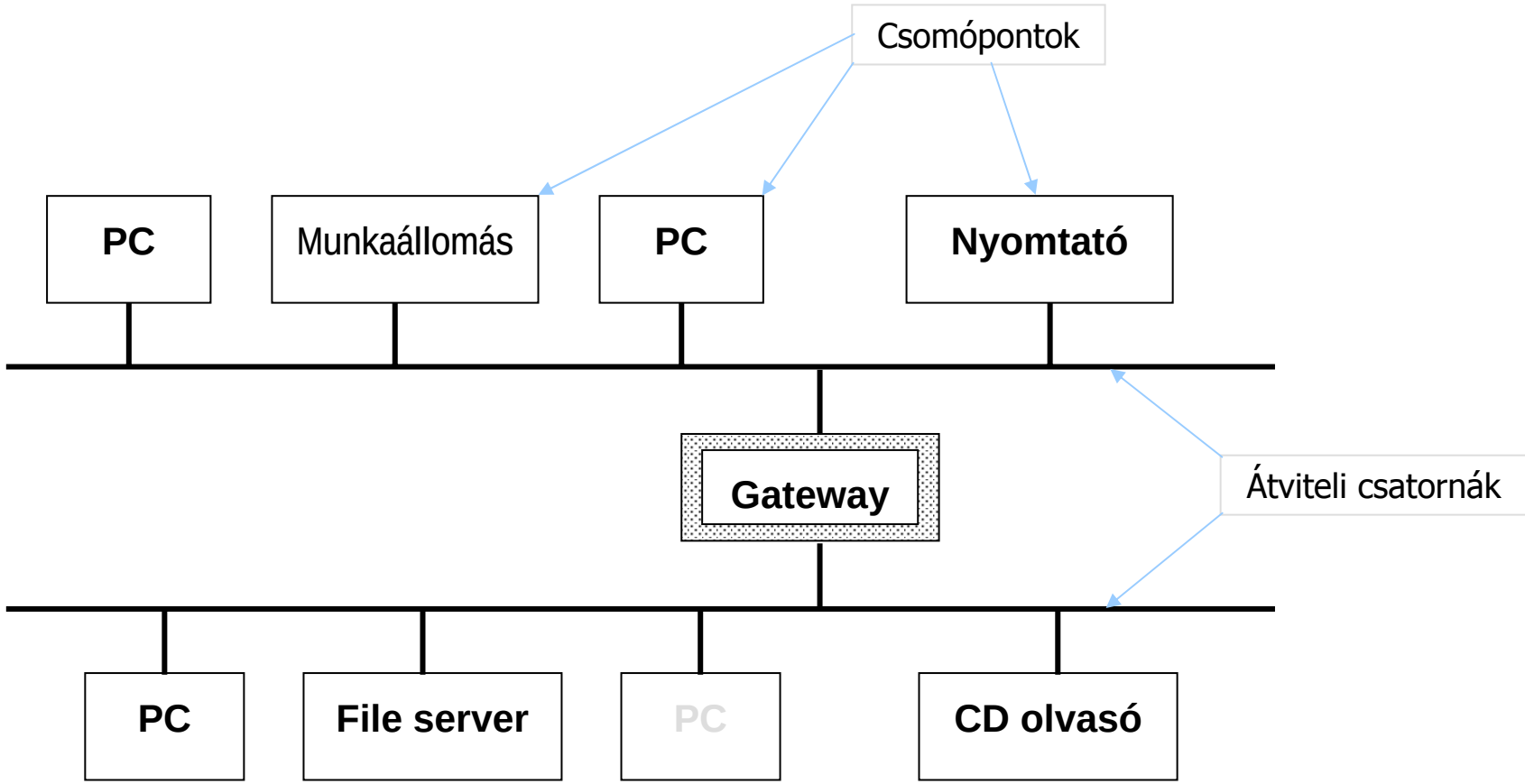
Elosztott rendszerek

- Lazán csatolt rendszerek (a terminológia szerinti elosztott rendszerek):
  - a processzoroknak független környezet,
  - önálló órajel és memória,
  - a kapcsolattartás hálózaton.
- Szorosan csatolt rendszerek:
  - közösen használt órajel a processzoroknak,
  - osztatlan memória, ahol a kapcsolattartás is történik.

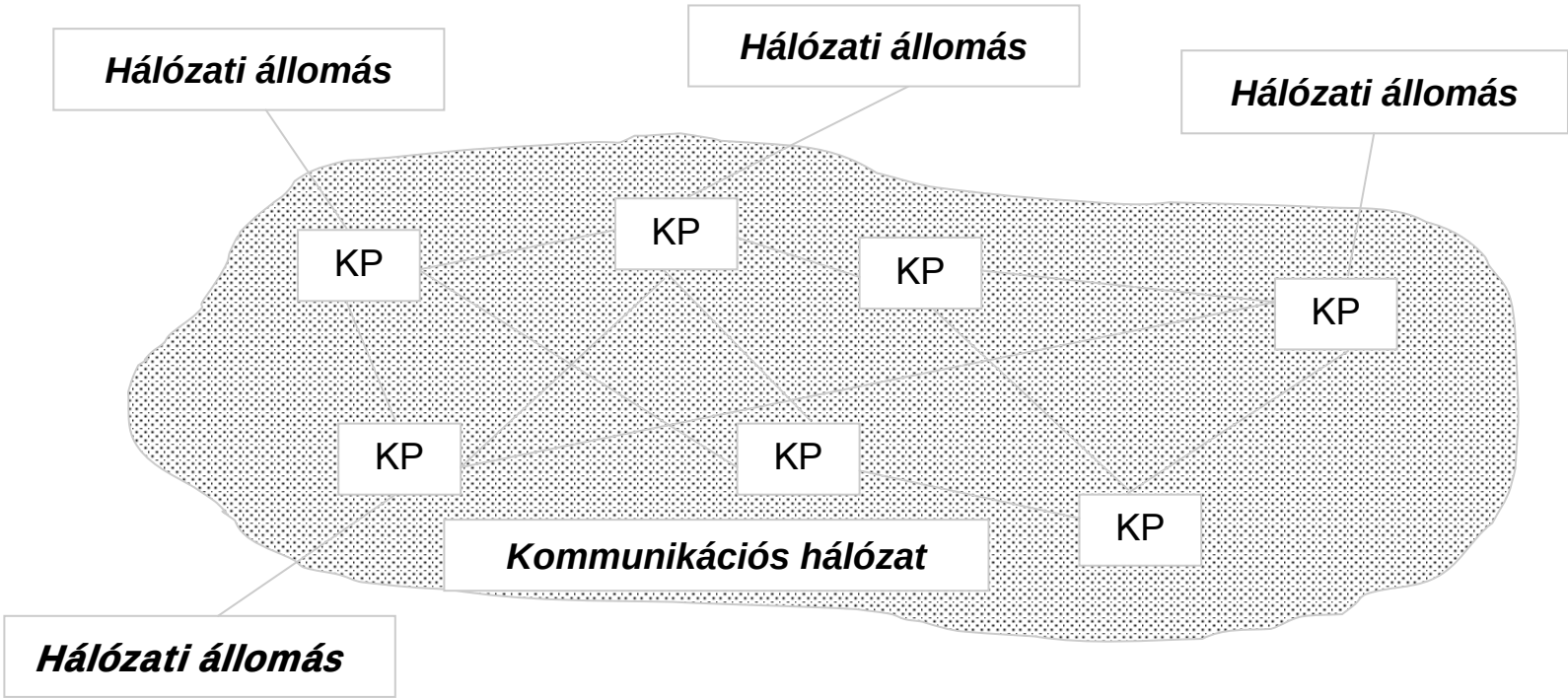
Elosztott rendszerek típusai

- Lokális (helyi) hálózat (*LAN*):
  - sebesség: 10Mbit/s - 1Gbit/s,
  - topológia: sín, gyűrű, csillag (kábel, rádió jel),
  - összekapcsolhatóság (gateway).
- Távoli csomópontok közötti (nagy területű) hálózat (*WAN*):
  - kommunikációs processzorok használata,
  - sebesség: 100Mbit/s fölött, de lassabb mint a LAN,
  - ”topológia”: modem, optikai kábel.

LAN


Csomópontok
PC
Munkaállomás
PC
Nyomtató
Gateway
Átviteli csatornák
PC
File server
PC
CD olvasó

WAN


Hálózati állomás
Hálózati állomás
Hálózati állomás
KP
KP
KP
KP
KP
KP
KP
Kommunikációs hálózat
Hálózati állomás

Protokoll-ok

A csomópontok egymással történő kommunikációjának szabályrendszere. Így pl.:

- címzés, egymás azonosítására,
- átviteli csatorna elérése,
- küldött adatformátum,
- redundáns infók; hiba felderítés és javítás,
- aktív vagy passzív nullás rendszer.

Az Ethernet protokoll

A LAN-ok esetén a leggyakoribb az Ethernet.

Tulajdonságai:

- alacsony szintű, az adatkapcsolat és a fizikai közeg elérését szabályozza,
- csomópontok busz topológiában,
- az átviteli közeg protokoll-ja az ütközés-detektáláson alapul,
- legsérülékenyebb az átviteli közeg: a kábel, ha sérül, minden kapcsolat megszűnik.

Az Ethernet ütközés-detektálása

- Címzett adatcsomagok (frame-k) az adatbuszon.
- A csomópontok figyelik az adatbusz forgalmát:
  - adás csak akkor, ha nincs adatforgalom,
  - ha mégis, akkor van az ún. ütközés,
  - ”saját” adattól való eltérés esetén ”ütközés jel” szétküldése, hogy az utolsó frame érvénytelen,
  - véletlen ideig való várakozás, az újabb frame előtt.

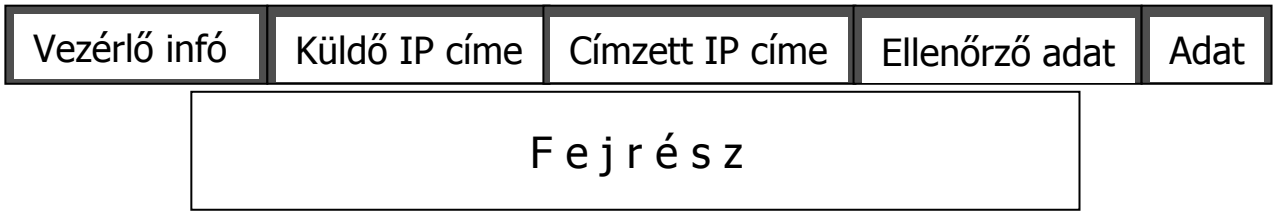
A TCP/IP protokoll

A WAN-ok esetén használatos.

Tulajdonságai:

- a fizikai átvitel fölé épül (pl. egy Ethernet-re épül rá),
- a csomópontok címzési módját, ill. az adatcsomagok méretét és formátumát rögzítik,
- a legalacsonyabb szintű az IP (*Internet Protocol*),
- erre épül rá a TCP (*Transport Control Protocol*),
- de az UDP (*User Datagram Protocol*) is.

Az IP protokoll



- az adatátvitele nem túl megbízható,
- az adatcsomag formátuma bonyolult:
  - fejrész (csak ez redundáns),
  - adatrész.

Az IP-re épülő protokoll-ok

A TCP biztosítja a megbízható hálózati átvitelt.

Az UDP csak az üzeneteket továbbítja.

Mindkettő lehetőséget ad:

- a csomópontokon futó folyamatok közvetlen elérésére,
- az egyes alkalmazások számára, hogy közvetlenül elérjék őket (pl.: FTP, telnet, e-mail).

Elosztott rendszerek előnyei

- Erőforrás-megosztás.
- Nyílt rendszer.
- Konkurens működés.
- Méretezhető rendszer.
- Hibatűrés, megbízhatóság.
- Átlátszóság.

Erőforrás-megosztás.

Azon eszközök (HW és SW) gyűjteménye amelyeket a CPU használ.

HW elemeknél a költségtakarékosság, míg a SW elemeknél a hatékonyság növelés a döntő. Az egyik legfontosabb a file-rendszer.

Vannak olyan elemek amelyek szorosan a procihoz tartoznak ezeket nem célszerű osztottan kezelni (pl. memória).

Megosztás esetén az egyes erőforrások fizikailag kötődnek egy csomóponthoz. Ezt az erőforrás-kezelő biztosítja.

Erőforrás-kezelő.

Az erőforrás-kezelő általános feladatai:

- az erőforrások megnevezése, és értelmezése,
- a kommunikációs *interface* biztosítása,
- párhuzamos elérés szabályozása.

Erőforrás-kezelő.

A leggyakoribb modell a kliens-szerver.

Lényege, hogy a szolgáltató (a szerver) egy adott szolgáltatást nyújt (kérésre) az ügyfeleknek (kliensek). Azok erről egy választ kapnak vissza.

A szerver-kliens kapcsolat mindig egy feladatra vonatkozik, így egy szerver is lehet kliens!

Nyílt rendszer

Akkor, ha bővíthető új elemekkel (HW és SW) különösebb nehézségek nélkül!

Megvalósítása:

- komponensek közötti interface-k alapos meghatározása, kidolgozása, publikálása,
- így lehetséges az egységes kommunikáció a folyamatok között,
- eltérő HW/SW, de a publikus IF szabványokhoz igazodik.

Ilyen rendszer pl. a UNIX.

Konkurens működés

- Folyamatok párhuzamos futtatása (több szerver és több kliens!),
- Adott feladat esetén, részben független részfeladatok párhuzamos végrehajtása.
- Időbeni szabályozás (szinkronizáció), főleg, ha közös erőforrást használnak, vagy kommunikálnak egymással.

Méretezhető rendszer

Kapacitás és méret növelés úgy, hogy a működő rendszer lényegében változatlan marad.

(Példa ill. ellenpélda, a telefonszámok használata)

Hibatűrés, megbízhatóság

Hibatűrő, ha képes a hibákat felismerni és kezelni úgy, hogy a funkcionalitása ne változzon (ill. csak korlátozott mértékben).

Megoldás a redundancia (HW és SW).

Megvalósítási példák:

- az ún. szavazó rendszer (HW):
  - páratlan számú (min. 3) párhuzamosan működő egység,
  - a végső kimenetet a többségi elven működő szavazógép állítja elő.

Hibatűrés, megbízhatóság

- az ún. javító blokkok (SW):
  - az SW modulokra bontott,
  - egy adott modul eltérő megvalósítású, de azonos funkcionalitású,
  - ezek a modul-verziók alkotnak egy blokkot,
  - mindegyikhez tartozik egy ún. elfogadási teszt,
  - ha egy modul ”megbukik”, akkor jön a következő a blokkon belül és így tovább,
  - az új blokk futtatása előtt menteni kell az addigi adatokat egy független tárba.

Átlátszóság

A rendszer elfedi a felhasználó elöl, az egyes szolgáltatások, erőforrások fizikai elhelyezkedését, ill. elosztott természetét. Így azok határvonala átlátszó lesz.

Átlátszóság formák:

- **Hozzáférés**, helyi és távoli erőforrások azonos eljárásokkal történő kezelése.
- **Hely**, adatobjektumok kezelése azok helyének ismerete nélkül.
- **Hálózati**, az előző kettő.

Átlátszóság

- **Konkurencia**, párhuzamosan futó folyamatok, osztott adathasználata.
- **Másolat**, adatobjektumok többszörözése anélkül, hogy a felhasználók ill. alkalmazások ezt észrevennék.
- **Hiba**, HW és SW hibák elrejtése úgy, hogy a felhasználók ill. alkalmazások a hibák mellett is elvégezhessék a feladataikat.
- **Vándorlási**, adatobjektumok szabad mozgása anélkül, hogy a felhasználók ill. alkalmazások működését zavarnák.

Átlátszóság

- **Teljesítmény**, a terhelés változásával lehetőség van a rendszer átkonfigurálására.
- **Méretezés**, rendszer bővíthetőség, a redszer-struktúra és az alkalmazások algoritmusainak megváltoztatása nélkül.

Elosztott rendszerek OPR-ei

Hálózati operációs rendszer:

- nem biztosítják a hálózati átlátszóságot,
- bejelentkezés távoli gépekre (távoli csomópontok elérése pl. telnet-tel),
- programok futtathatósága távoli csomópontokon,
- adatmozgás a lokális és a távoli csomópontok között (pl. FTP-vel).

Elosztott rendszerek OPR-ei

Elosztott operációs rendszer:

- elosztott hardveren futó és azt elrejtő,
- adatvándorlás, adott csomópontba másolja az adatot, majd vissza a helyére,
- számítás-vándorlás, a tevékenység végrehajtását telepítjük át (pl. nagytömegű adaton egyszerű műveletek),
- folyamat-vándorlás, az előző kiterjesztett változata.

A folyamat-vándorlás okai

A terhelés arányos elosztása.

Független részfolyamatok esetén, a számítás felgyorsítása.

Speciális HW és/vagy SW ellátottság az adott csomópontban.

Az adatok gyorsabb elérése az adott csomópontban.

Elosztott file-rendszerek

A távoli csomópontokban elhelyezkedő file-ok elérése, a helyi file-okkal azonos *interface*-n keresztül, az OPR-rel együttműködve.

Követelmények

Hozzáférés átlátszósága:

- lokális és távoli file-ok azonos eljárásokkal történő kezelése.

Elhelyezkedés átlátszósága:

- a file-ok neve nem hivatkozik fizikai elhelyezkedésükre.

Vándorlás átlátszósága:

- a file-ok a rendszerben mozgathatók anélkül, hogy megváltozna az elnevezésük.

Követelmények

Skálázhatóság (méretezés átlátszósága):

- a terhelés növekedésével, új komponensekkel lehet bővíteni.

Hibatűrés:

- egyes komponensek hibája esetén képes tovább működni.

Felhasználók mobilitása:

- az összes file a rendszer minden belépési pontjáról elérhető.

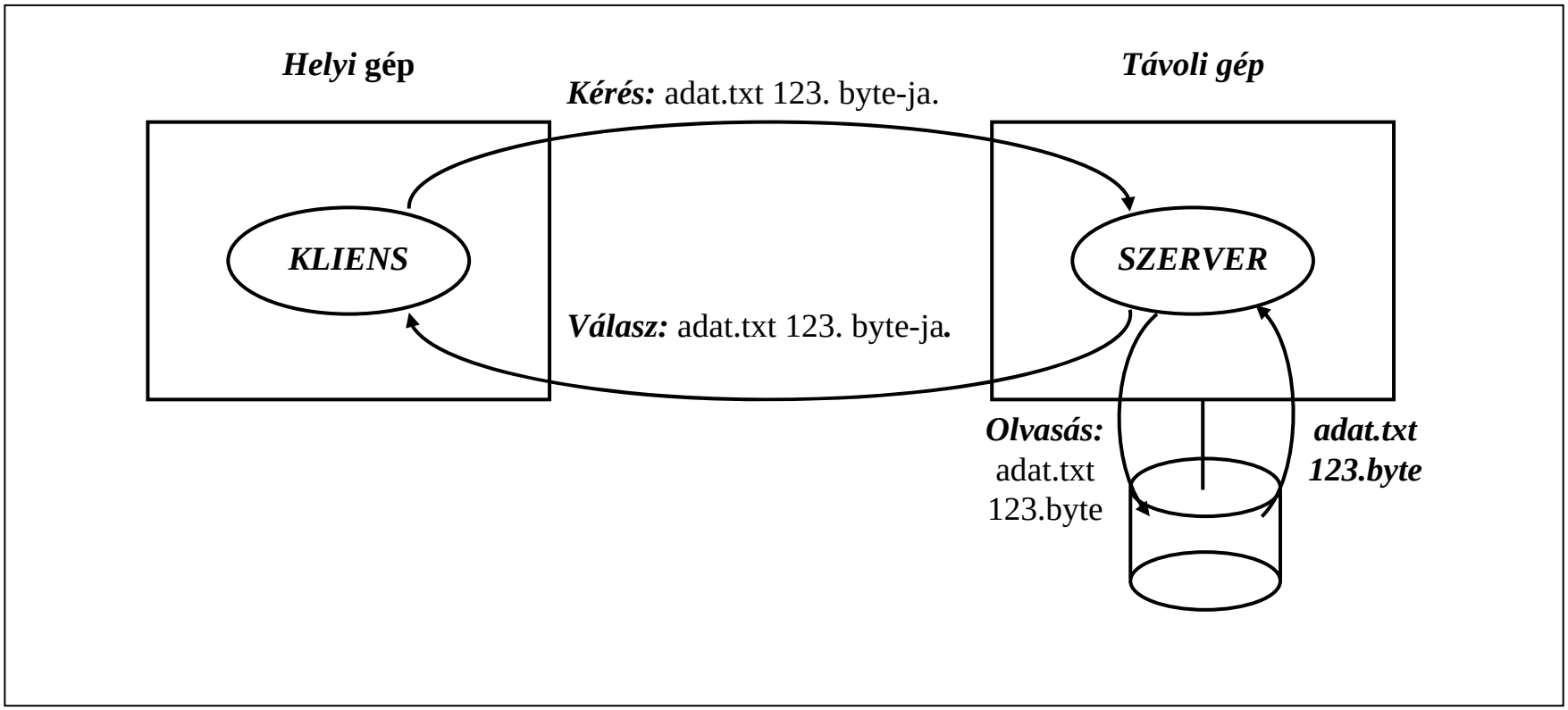
Követelmények

File-ok mobilitása:

- a file-ok áthelyezhetők az egyik helyről a másikra a rendszer futása közben is.

Elosztott file-rendszerek megvalósítása

Kliens-szerver modell:



Kliens állapotának tárolása

Állapottárolós:

- a szerver információt tárol a kliensekről (a file-okról kapcsolat-leíró készül),
- gyors kommunikáció (előre olvasás, egymás utáni hozzáférések),
- szerver leálláskor ”meghal minden” (hiszen elveszik az állapot leíró), bonyolult az újraindulás.

Állapotmentes megvalósítás:

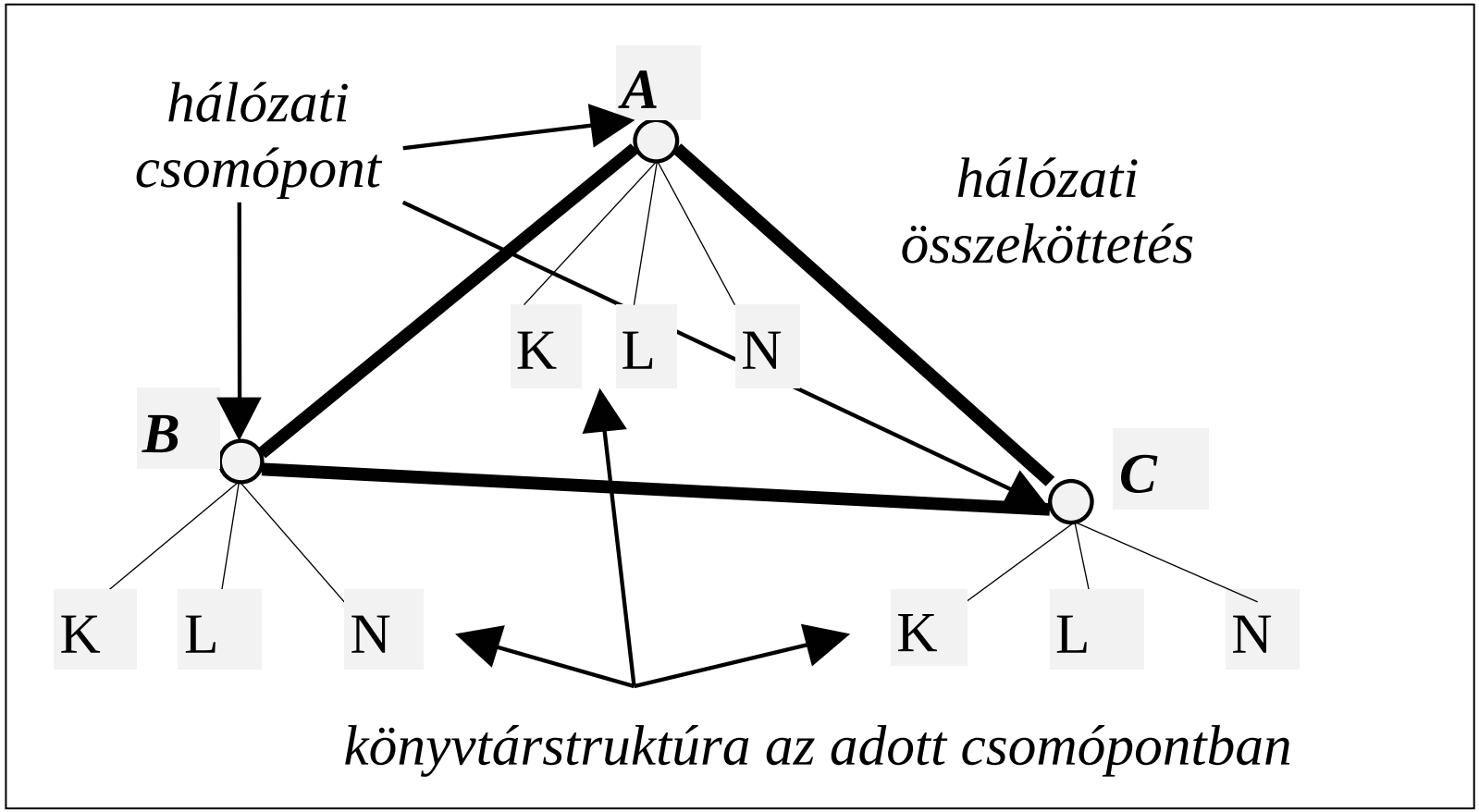
- a szerver nem tárol információt a kliensekről,
- redundáns a kommunikáció, így lassabb,
- egyszerű újraindítás.

File-ok elnevezése elosztott file-rendszerben

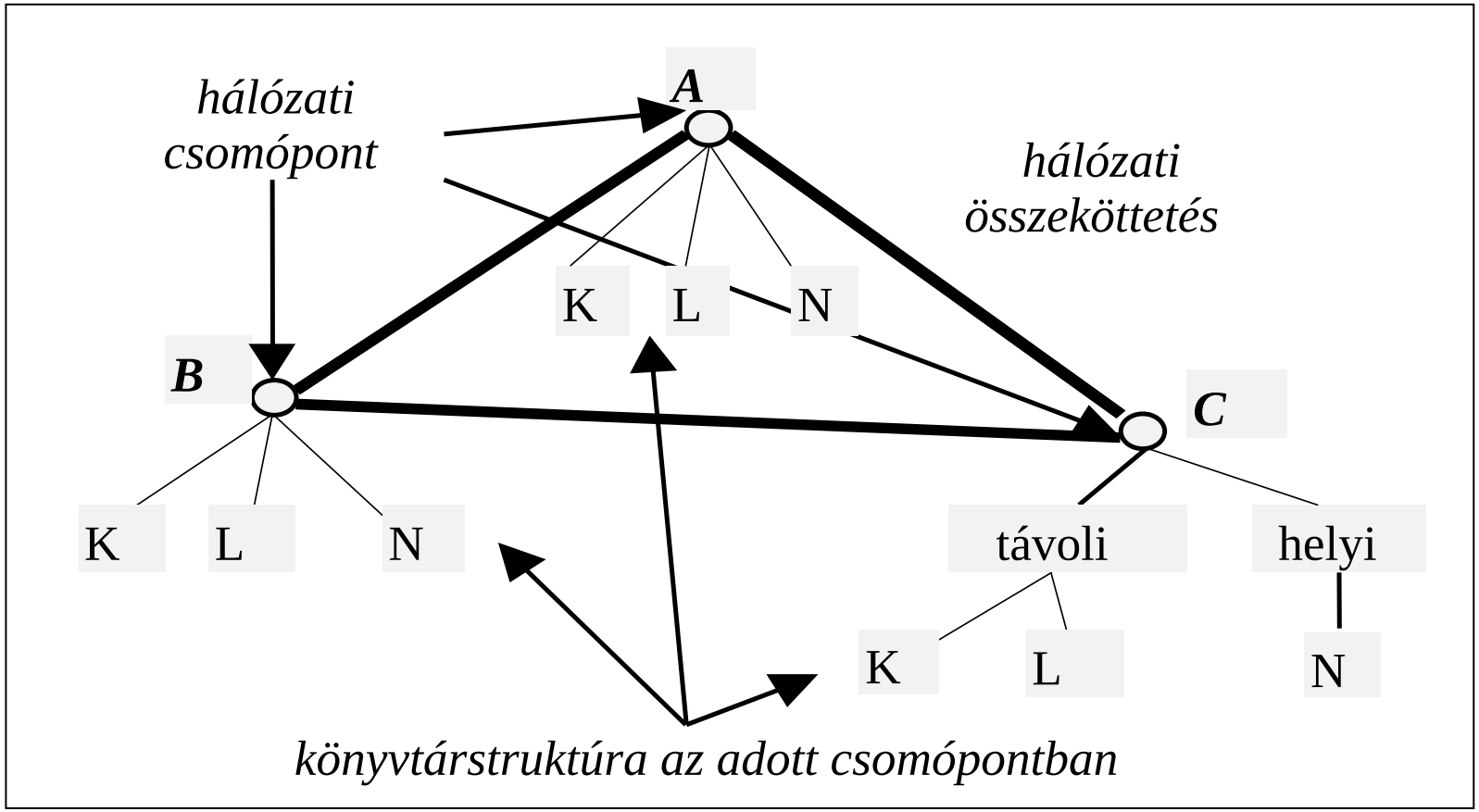
Az egyes csomópontokban az állománynevek ”tere” lehet:

- *uniform* (egységes, különböző csomópontokban ugyanazzal a névvel),
- nem *uniform* (nem egységes, eltérő név {és útvonal!} lehetőséggel).

Uniform névtér



Nem uniform névtér



Védelem és biztonság

Védelem és biztonság

Védelem:

- eljárások és módszerek azon rendszere, amelyek biztosítják az erőforrások elérésének szabályozását. Továbbá megakadályozzák az illetéktelen erőforrás használatot. Tipikusan ”belső” probléma.

Biztonság:

- annak a mértéke, hogy a rendszerben tárolt adatok mennyire sérthetetlenek. Működési környezet figyelését igényli.

Védelem

Az ún. szabály rendszer meghatározza, hogy mit kell tenni a rendszer zökkenőmentes, biztonságos használatához.

Az ún. mechanizmus (módszer) pedig, lehetőséget teremt a szabályozás megvalósítására, azaz a rendszerobjektumok kezelésének mikéntjét határozza meg.

A számítógépes rendszer, objektumokat használó folyamatok halmaza, ahol a műveletek végrehajtását jogosítványokhoz kell kötni.

Védelmi tartományok

A rendszer abszolút biztonságosan működik, ha minden pillanatban az összes folyamat csak azokkal a jogosítványokkal rendelkezik, amelyek ”megilletik”.

Ez megvalósíthatatlan, ezért az objektumok elérésének a szabályozására a ~-okat használjuk.

Ezek, jogosítványok gyűjteménye az objektumokon végezhető műveletek végrehajtására.

Megvalósításuk ún. elérési mátrix-szal, amely lehet statikus ill. dinamikus.

Elérési mátrix statikus védelmi tartományokkal

Tartományok	Objektumok			
	adat.txt	doc.doc	help.dat	nyomtató
A	**olvasás**		**olvasás**	
B				**nyomtatás**
C		**olvasás**		
D	**olvasás, írás**		**olvasás, írás**	

Elérési mátrix dinamikus védelmi tartományokkal

- Védelmi tartomány váltása:

Tartományok	Objektumok				Tartományok			
	adat.txt	doc.doc	help.dat	nyomtató	A	B	C	D
A	**olvasás**		**olvasás**			**váltás**		
B				**nyomtatás**			**váltás**	**váltás**
C		**olvasás**						
D	**olvasás írás**		**olvasás írás**		**váltás**			

Elérési mátrix dinamikus védelmi tartományokkal

- Védelmi tartomány váltás: lsd az előző ábrát.
- Elérési jogosítványok másolása:
  - adott védelmi tartományban futó folyamat jogosult átadni, egy adott művelet elvégzésére szóló jogosítványt, más védelmi tartományoknak.
- Objektum tulajdonlása:
  - adott védelmi tartomány ún. tulajdonosi joggal rendelkezik egy adott objektum felett. Így ez adhat jogosítványt más védelmi tartománynak az objektumon elvégezhető művelet végzésére.

Elérési mátrix ábrázolása és kezelése

Egy-egy védelmi tartomány általában csak néhány objektum elérésére tartalmaz jogosítványokat, ezért az elérési mátrix igen nagy és ritka kitöltésű lesz!

Cél az optimális tárolás és kezelés.

Ezek:

Elérési mátrix ábrázolása és kezelése

- Globális tábla:
  - listába gyűjti a: <tartomány, objektum, művelet végzési jog> hármasokat,
  - igen hosszú lesz a lista, így a műveletek elvégzése is hosszadalmas
  - ritkán használatos.
- Objektum elérési lista:
  - minden objektumhoz tároljuk a: <tartomány, művelet végzési jog> párosokat, mezőnkénti tárolás,
  - gyorsítja a jogosultság ellenőrzését.

Elérési mátrix ábrázolása és kezelése

- Tartományok jogosítványainak listája:
  - minden védelmi tartományhoz tároljuk a: <objektum, művelet végzési jog> párosokat, rekordonkénti tárolás.
  - gyorsítja a tartományok szerinti elérést.
- Zár-kulcs módszer:
  - az előző kettő ötvözete, ez a leghatékonyabb,
  - bitminták kialakítása minden objektumhoz (zár) és minden védelmi tartományhoz (kulcs),
  - ha az adott tartomány kulcsa illeszkedik egy objektum zárjába, akkor a tartományt birtokló folyamat elvégezheti a kulcshoz tartozó műveletet.

Biztonság

Rosszindulatú támadások és ”véletlen” sérülések elleni védekezés:

A szándékos behatolások típusai:

- adatok illetéktelen olvasása,
- adatok illetéktelen módosítása,
- adatok tönkretétele.

Cél, hogy a behatolás költsége nagyobb legyen, mint a remélt haszon!

Biztonsági módszerek

A felhasználók azonosítása:

- személyes tulajdonságai alapján (pl.: ujjlenyomat, kézlenyomat, retinalenyomat, DNS kód, aláírás (haha)),
- birtokában lévő tárgyak (pl.: kulcs, azonosító, kártya),
- általa ismert infó (pl.: név, jelszó, algoritmus).

Legelterjedtebb a harmadik, de a jelszó miatt problémás lehet. Ezért a rendszer a következőkre ”kényszerítheti” a felhasználót:

- ”nehezen kitalálható” jelszó megadatása,
- gyakori jelszó csere (☹, aha).

Támadási stratégiák

Tipikus, személyre jellemző jelszavakkal.

Szisztematikus, szótár szavaival.

Személyes környezet beható ”tanulmányozása”.

Lehallgatás, leolvasás.

Általános biztonsági módszerek

Veszélyeztetett pontok figyelése, ”gyanús” aktivitás esetén ”intézkedés”. Ilyenek:

- sikertelen jelszavak utáni exponenciális idejű késleltetés,
- véges számú jelszó-kísérlet utáni letiltás,
- aktivitás-naplózás (csak utólagos felderítésre jó).

Kódolt üzenetek (rejtjelezés) a publikus csatornákon,

Partner hitelesítés (pl.: elektronikus aláírás).

Rosszindulatú programok

Tevékenységi köreik pl.:

- felhasználók bosszantása,
- file-ok törlése, átírása,
- adatok módosítása,
- HW elemek rongálása,
- az OPR működésének ”felborítása”,
- hálózati kapcsolatok lassítása, tönkretétele,
- ”szaporodás”.

Rosszindulatú programok

A hatásmechanizmusuk alapján osztályozzuk ezeket. Így vannak:

- vírusok,
- férgek,
- trójai falovak.

A vírusok

- A leggyakoriabbak.
- Kódszegmens, amely futtatható programokhoz csatolódik.
- A program futásakor aktivizálódik, valamilyen ”nevezetes” dátum hatására:
  - végrehajtja a ”feladatát” (file-ok {rendszer is!} törlése, módosítása, HW károsítás),
  - és ”szaporodik”, többnyire ”csendben”, kár okozás nélkül.
- A .DOC és .XLS file-ok programkódot tartalmaznak (!!!), így hajlamosak a fertőzésre!

A férgek

- Egy önálló program.
- Hálózati kapcsolatokon keresztül terjed.
- File-ok {rendszer is!} törlése, módosítása, ill. hálózati kapcsolatok rombolása.
- Hálózaton keresztül, védett infók visszaküldése a ”gazdinak”.
- ”Szaporodik”, és igyekszik minél több példányban futni, így terheli a rendszert.
- Nagy felkészültséget és rendszer ismeretet igényel, ezért ritkább, de komolyabb károkat okoz.

A trójai falovak

- Nyilvánosan árusított programokban vannak elrejtve.
- A tényleges funkció működése ”mögött” tevékenykedik (ál-programok, úgy csinál mintha, aztán mégse, pl.: banki beléptető képernyő).
- Károkozásuk a vírusokéhoz hasonlatos.
- Nem szaporodik.
- Rejtekajtó használata:
  - veszélyes, ha a fordító programokba, vagy a programkönyvtárakba épül be, mert így a rejtett tevékenység kódja bekerül a lefordított kódba, és szinte lehetetlen felderíteni.

Védekezési módok

- Vírusirtó programok használata.
- Tűzfal használata.
- Jogtiszta SW-ek.
- Újonnan felkerülő programok vírus ellenőrzése!