

# *Operációs rendszerek*

## A Windows NT memóriakezelése

# Az NT memóriakezelése

## Memóriakezelő feladatai:

- Logikai-fizikai címtranszformáció:
  - a folyamatok virtuális címterének címeit megfelelteti a fizikai címeknek.
- A virtuális memóriakezelés megvalósítása:
  - túlterheltség esetén: a régen nem használt memóriaterületek (lapok) háttértárra mentése.
  - hivatkozás a háttértárra mentett memórialapra:
    - a szálat várakoztatja addig,
    - amíg a kérdéses lapot be nem mozgatta a fizikai memóriába.

# Tulajdonságok

- 32-bites virtuális memóriakezelés:
  - 4 GB-os címtartomány, alapesetben:
    - a felső 2 GB az alkalmazásoké,
    - az alsó 2 GB az OPR-é.
  - nem alapesetben: boot.ini file-ban "/3G".
- A memóriát laponként (page) kezeli.

Többletszolgáltatások a rendszer hatékonyságának növelése érdekében, pl.:

- copy-on-write mechanizmus,
- a file-ok memóriaként történő elérése (memory mapped files), így az osztott file használat megvalósulhat.

# A boot.ini file

- [boot loader]
- timeout=30
- default=multi(0)disk(0)rdisk(0)partition(1)\WINDOWS  
S
- [operating systems]
- multi(0)disk(0)rdisk(0)partition(1)\WINDOWS=  
"Microsoft Windows XP Professional - magyar" /3G

# A memóriakezelő felhasználói interface-e

A memóriakezelő szolgáltatásai a Win32  
API-n keresztül érhetőek el.

# A memóriakezelő szolgáltatásai a felhasználóknak I.

- Virtuális memória allokáció, ill. felszabadítás.
- Osztott elérésű (shared) memória létrehozása.
- A file-ok osztott elérésű memóriához hasonló elérése.
- Virtuális memória kezelés (pl.: információkérés, adatok memóriába rögzítése, kiírása háttértárra)

# A memóriakezelő szolgáltatásai a felhasználóknak II.

- Memória védelmi funkciók.
- Kernel szintű funkciók (elsősorban a device driver-ek) támogatása (pl.: fix fizikai memóriaterület használata).

Fizikai memória optimális  
kihasználása,  
kétfázisú memóriafoglalás



# Memóriafooglalás az NT-ben

- A memóriafooglalás két lépésben történik a Windows NT-ben:
  - reserve  
virtuális címtartomány lefooglalása,
  - commit  
virtuális memória lefooglalása.
- Lehetőség a két lépés egy függvényhívásban történő végrehajtására.

# A reserve művelet

- Nem jelent tényleges fizikai memóriafoglalást.
- A folyamat csak deklarálja az operációs rendszer számára, hogy mennyi memóriára lesz, vagy lehet szüksége.
- Mindig lapok foglalása történik:
  - x86-os rendszerekben ez 4 KB,
  - esetleges belső tördelődés fellépése.
- A memórialapok mindig 64 KB memóriaegységek határán kezdődnek, az esetleges lapszám növelés miatt.

# A commit művelet

- Tényleges tárterület foglalás a rendszerben.
- Csak a korábban a reserve művelettel már lefoglalt memórián hajtható végre.
- Csak a commit művelet után tudja a folyamat a memóriát használni.
  - A csak reserve-elt memóriacímre történő hivatkozás hibát okoz.

# A két lépcsőben történő memória foglalás előnyei

- Hatékonyabb működés:
  - reserve csak címtartomány foglalást végez,
    - bejegyzés az OPR belső táblázatában,
  - tényleges erőforrás-használat (memória, ill. ún. backing store foglalás) csak a commit után,
  - a folyamatok futtatásához használt fizikai memóriaigény csökken,
  - lehetséges előre lefoglalni egybefüggő címtartományokat a rendszer terhelése nélkül.

# A szálak user stack-jének foglalása I.

- A stack-nek folyamatos címtartománynak kell lennie.
- Alapértelmezés:
  - 1 MB memóriát foglal reserve művelettel,
  - de csak 2 lapnyi (2 x 4 KB) memóriát foglal a commit művelettel.
  - Az első lap a stack teteje.
  - A második lap szerepe, hogy a rendszer érzékelje, ha a stack megtelt, és automatikusan foglaljon commit művelettel új oldalakat.

# A szálak user stack-jének foglalása II.

- A kétlépéses memóriafoglalás nélkül:
  - minden szál indulásakor a rendszermemóriából ténylegesen le kellene foglalni 1 MB-nyi területet, melynek valószínűleg jelentős részét a szálak többsége nem is használná.

# Osztott elérésű memória

# Osztott elérésű memória

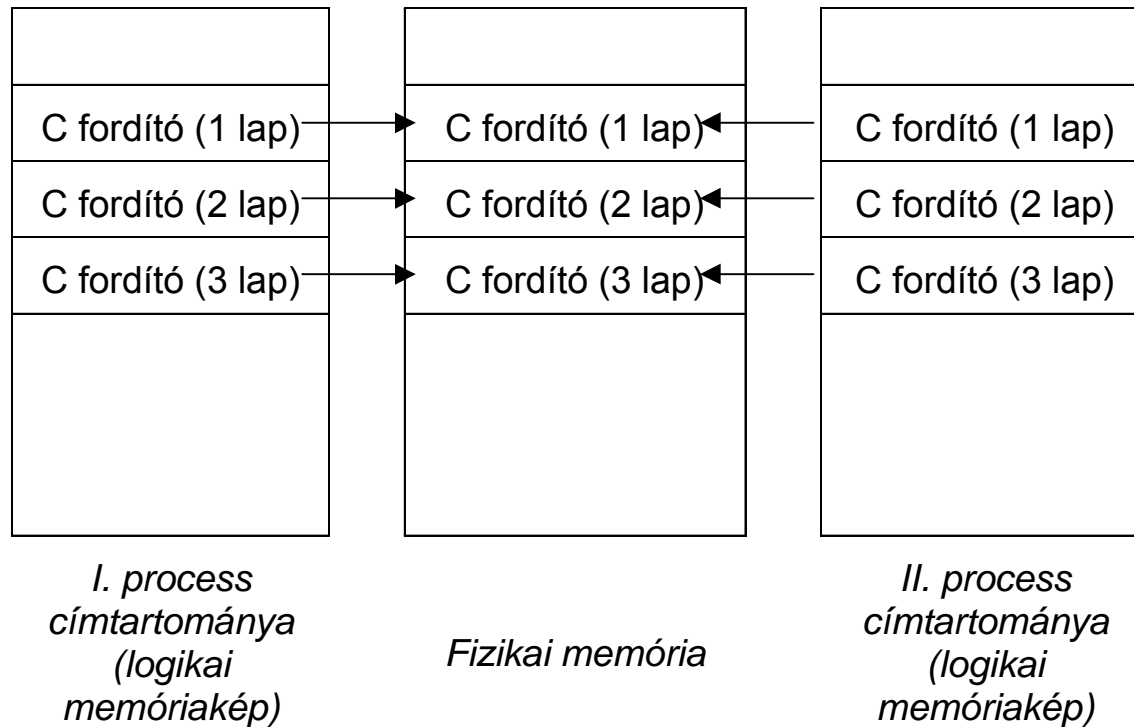
- Folyamatok közötti információ csere.
- Ugyanazon utasításhalmaz, program (pl. C fordító) használata.
- Két folyamat adott virtuális címtartományában azonos tartalom.
- A folyamatok adott virtuális címtartományához azonos fizikai memórialapokat rendel.
- Megvalósítása az NT-ben:
  - a section object-tel (szekció objektum).



# Section object

- NT objektum.
- Jellemzői:
  - a neve,
  - és az általa reprezentált file:
    - back storage (pagefile) → osztott elérésű memória,
    - adat file → memóriába leképzett (mapped) file, azaz file-részlet (hiszen a file mérete nagyobb lehet, mint az adott folyamat címtartománya. Pl.: DLL-ek kezelése),
  - ACL (Access Control List) – konfigurálható elérést szabályozó védelmi lista:
    - mely folyamatok érhetik el,
  - mely (virtuális) memóriatartományban látható,
  - standard output átirányítások kezelése,
  - a cache manager kiszolgálása file-okkal.

# Osztott elérésű memória

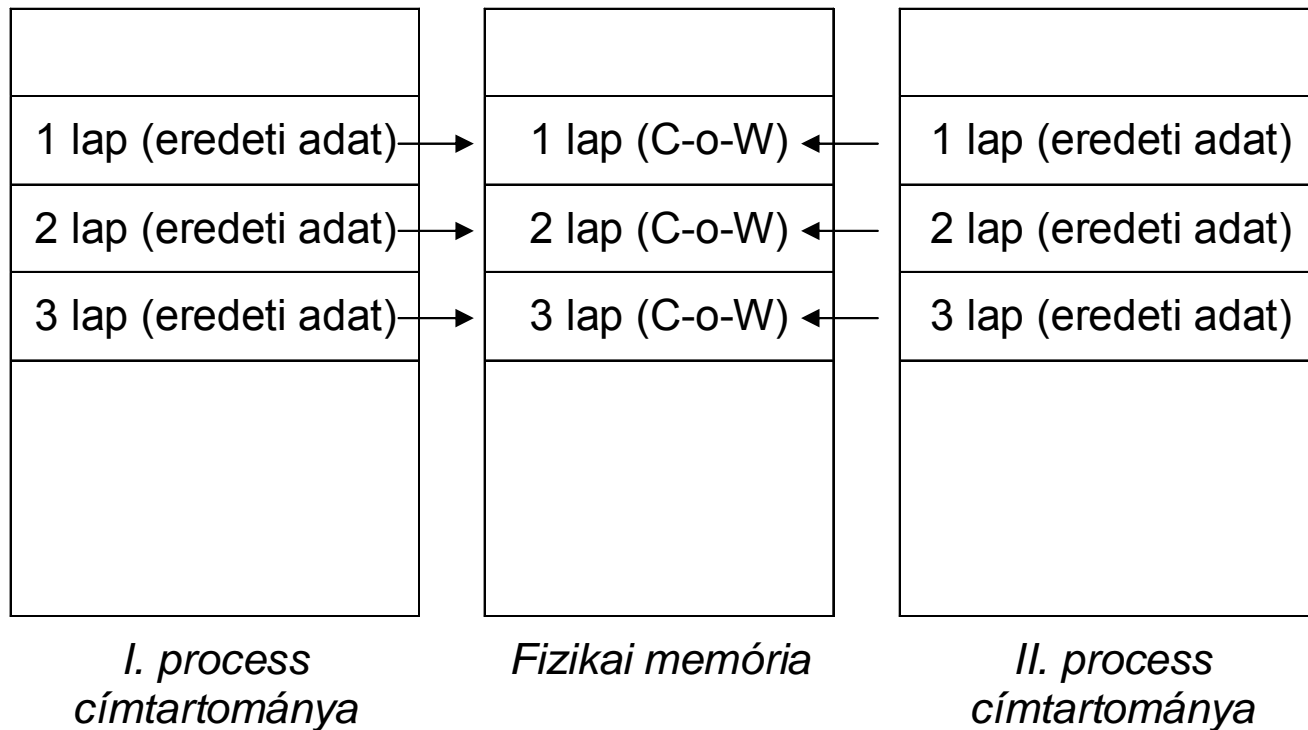


Fizikai memória optimális  
kihasználása,  
copy-on-write mechanizmus

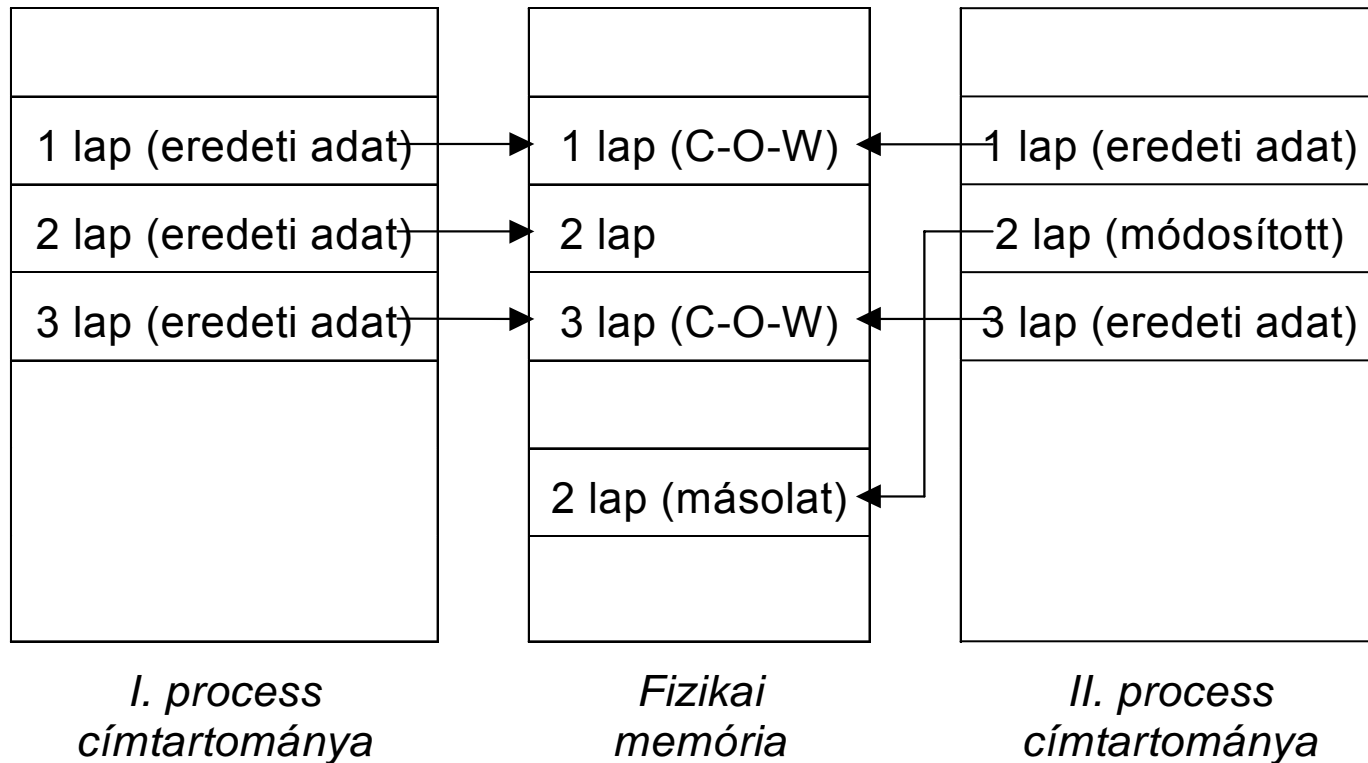
# A copy-on-write mechanizmus

- Memórialap takarékosági elv.
- "Másolás írás esetén".
- C-o-W jelzőbit bebillentése az adott memória lapon, ha egy folyamat egy olyan section object-et kezd használni amely már használatban van.
- Az új folyamat csak akkor kap önálló memórialap másolatot az eredetiről, ha memória írási műveletet kezdeményez.

# Copy-on-write memórialapok az írást megelőzően



# Copy-on-write memórialapok az írást követően



# Memóriavédelem

# Memóriavédelem

- A cél:
  - a megbízhatóan működő, robosztus operációs rendszer azaz,
  - hogy egyik folyamat se módosíthassa egy másik folyamat címtartományába tartozó memóriaterületeket, (kivéve az osztottan használt részeket).



# Memóriavédelem megvalósítás szintjei

- Kernel modulok adatstruktúrái a felhasználói módú folyamatok számára elérhetetlenek (HW megszakítás).
- Folyamatonkénti diszjunkt (elkülönült) logikai címtartományok (az osztott kivételével).  
Elérés memóriakezelőn keresztül:
  - HW-rel támogatott logikai-fizikai transzformáció,
  - adott memórialap hozzáférésének szabályozása.
- Processzorfüggő memória védelmi attribútumok:
  - r, w, x,
  - pl.: a kódszegmensek memóriaterületei "r"-t kapnak.

# Memóriavédelem megvalósítás szintjei

- Section object, ACL (Access Control List):
  - szekció objektum elérési lista,
  - csak az itt felsorolt folyamatok érhetik el,
  - az NT védelmi rendszeréhez tartozik.

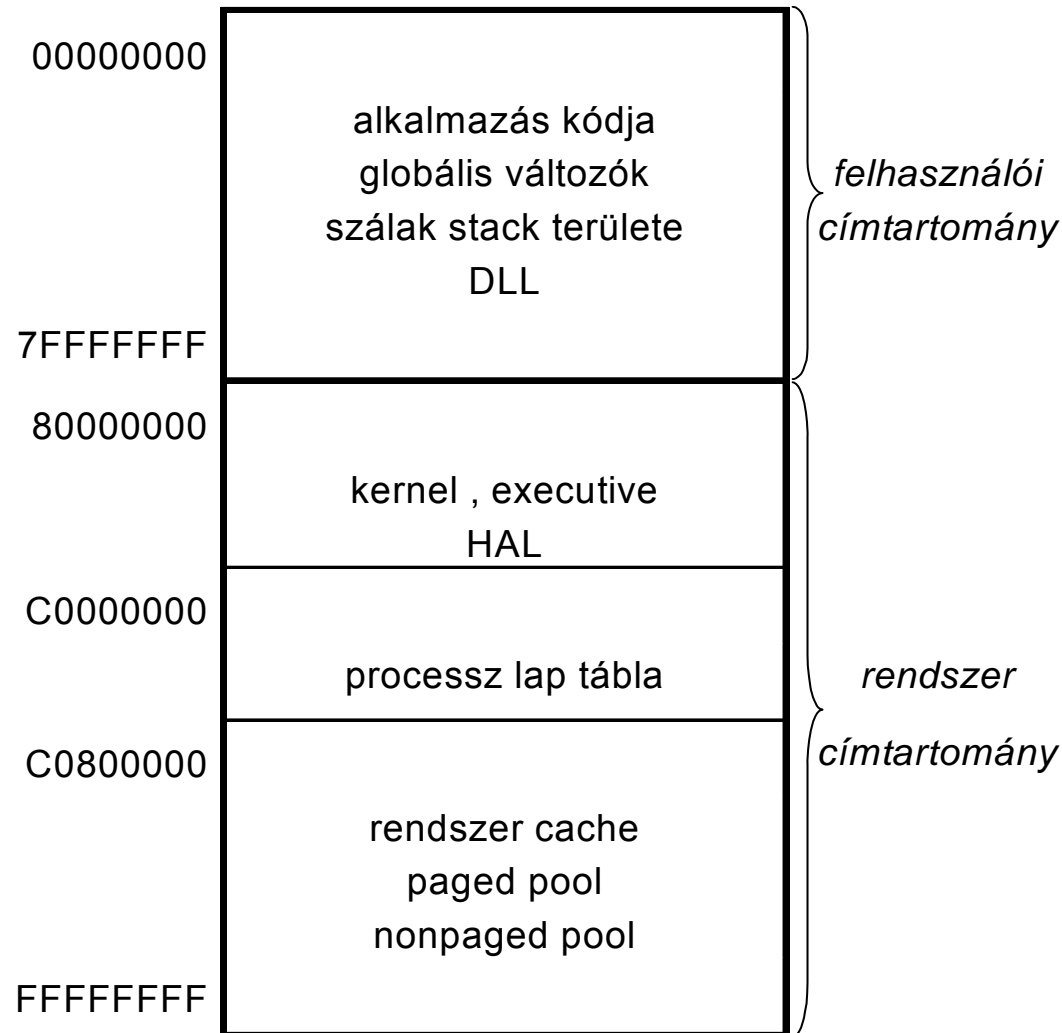
# Kernel komponensek memóriafooglalása

# Memória foglalás

- Kernel komponensek esetén nem mindig megengedett a lapozás, pl.:
  - kernel szinten megvalósított kölcsönös kizárás,
  - driver-ek által használt memóriaterületek.
- Így a kernel komponensek memória foglalása:
  - lapozott memória tárból (paged memory pool) vagy,
  - nem lapozott memória tárból (nonpaged memory pool) történhet.

# Címtranszformáció, virtuális címtartomány

# A logikai címtér felépítése



# A logikai címtér

- Alkalmazás kód:
  - legfelső címtartomány a rendszer dinamikusan foglalt elemei,
  - nem minden cím használható (legalsók nem).
- OPR:
  - kernel, HAL, executive,
  - processz lap tábla:
    - memória címtranszformációs táblázatok.

# Címtranszformáció

- Kétszintű indexelés.
- HW támogatással:
  - DEC Alpha,
  - Intel x86.

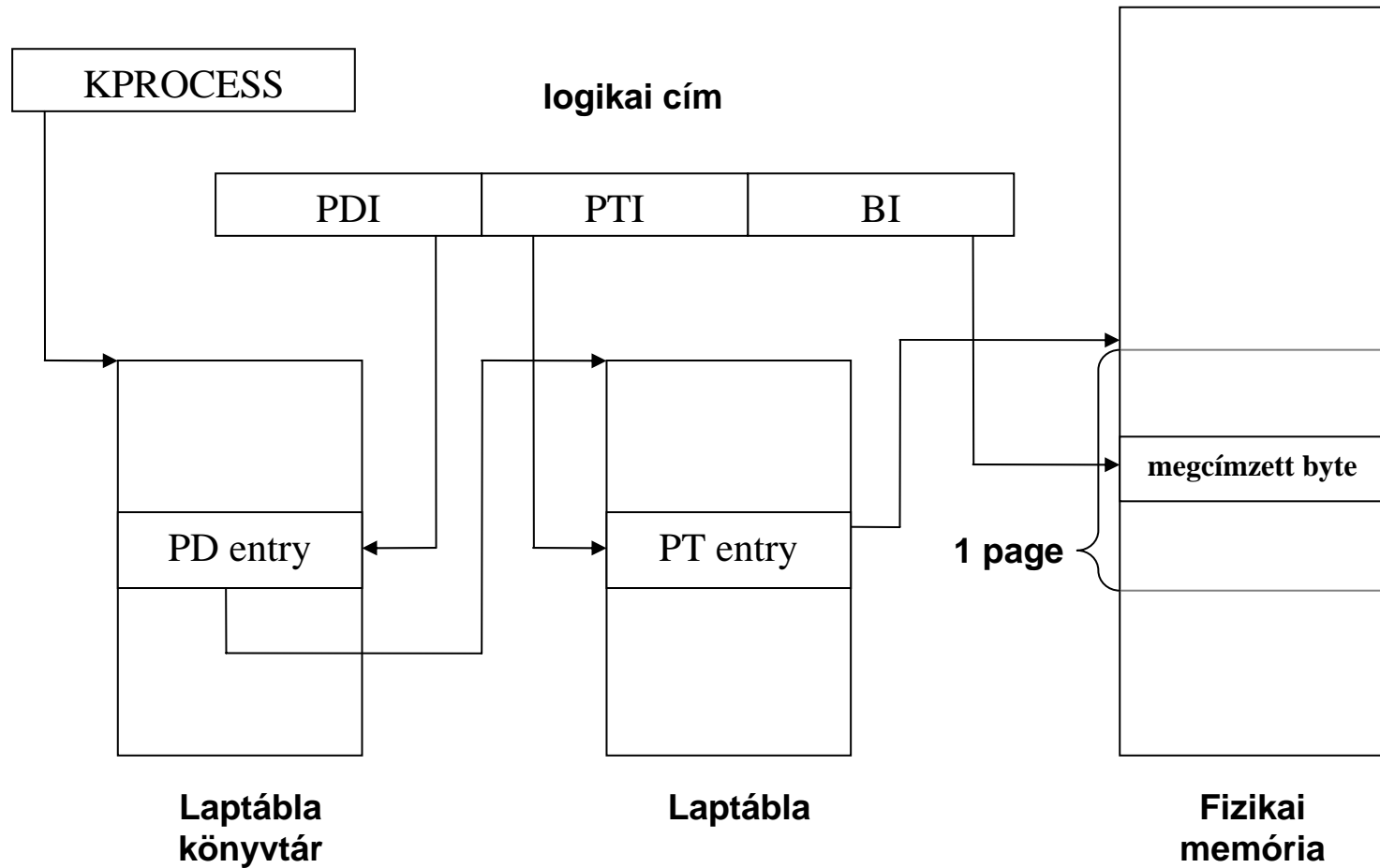


# A virtuális cím felépítése (x86)

- Processzorfüggő.
- A PDI (Page Directory Index) és PTI (Page Table Index) kijelöli az elérendő memórialapot.
- A BI (Byte Index) a lapon belüli eltolást határozza meg.

31		0 (LSB)
<b>laptábla könyvtár index (PDI)</b>	<b>laptábla index (PTI)</b>	<b>byte index (BI)</b>
x86: 10 bit Alpha: 8 bit	x86: 10 bit Alpha: 11 bit	x86: 12 bit Alpha: 13 bit
virtuális memórialap szám		eltolás

# A címtranszformáció menete x86-os processzorok esetén



# Adatszerkezetek címtanszformációhoz

- Folyamatonkénti lap könyvtár (page directory):
  - maximum 1024 laptábla kezdőcím-bejegyzés,
  - kezdőcíme a KPROCESS-ben.
- Laptábla (page table):
  - a logikai-fizikai címtanszformáció helye,
  - maximum 1024 hozzárendelési bejegyzés,
  - maximum 512 db lehet egy processzben, illetve a rendszerben,

# Az NT méretezhető memóriakezelése, NT memória modelljei

# Az NT memória modelljei

- "Méretezhető" operációs rendszer.
- Három memória méretkategória:
  - kicsi (small),
  - közepes (medium),
  - nagy (large).

# A fizikai memória mérete különböző memóriamodellek esetén

méret kategória	<b>x86</b>	<b>Alpha</b>
<b>kicsi</b>	≤ 19 MB	≤ 31 MB
<b>közepes</b>	20-32 MB	nem használt
<b>nagy</b>	≥ 32 MB (NT Workstation) ≥ 64 MB (NT Server)	≥ 32 MB (NT Workstation) ≥ 64 MB (NT Server)

# Memória modellek használata

- Virtuális memóriakezelés méretezése:
  - laphiba esetén memóriába bemozgatandó lapok száma:
    - demand paging (megkívánt lapszám)
    - előretekintő lapozás,
    - kód és adat tároló lapok megkülönböztetése,
  - folyamatokhoz rendelt fizikai memórialapok (working set) számának meghatározása.

# Előrettekintő lapozás

Laphiba esetén a hibát okozó lapon felül  
betöltendő lapok száma:

	Adat szegmens	Kód szegmens
kicsi	0	1
közepes	1	3
nagy	3	7



# A "working set" mérete

Folyamatokhoz rendelt fizikai memórialapok száma (working set mérete):

	minimális	maximális
kicsi	20	45
közepes	30	145
nagy	50	345