

Operációs rendszerek

Folyamatok kezelése a UNIX-ban



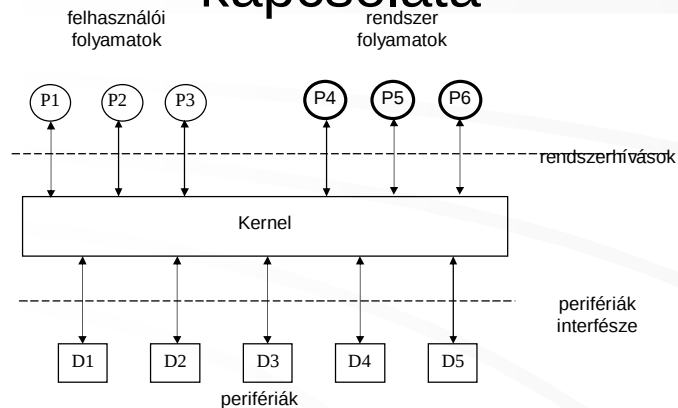
Folyamatok a UNIX-ban

- Folyamat: multiprogramozott operációs rendszer alapfogalma - absztrakt fogalom.
- Gyakorlati kép: végrehajtás alatt álló program.
- Folyamat *virtuálisan* egyedül fut a rendelkezésre álló hardveren.
- Hardver erőforrások elérése:
 - operációs rendszeren keresztül.
- Műveletek:
 - erőforrás igénylés, várakozás.
- Operációs rendszer feladata:
 - *összehangolja az egyes folyamatok perifériás műveleteit,*
 - *elosztja a rendelkezésre álló hardver erőforrásokat.*

hétfő, 2006. február 27.

Dr. Benyó Balázs
Operációs rendszerek III.

Folyamatok és a rendszer kapcsolata



hétfő, 2006. február 27.

Dr. Benyó Balázs
Operációs rendszerek III.

Interfészek

OR interfészek:

- Programozói interfész (rendszerhívások):
 - Folyamat $\rightarrow$ OR
- Periféria interfész:
 - OR $\rightarrow$ I/O eszközök (perifériák)

hétfő, 2006. február 27.

Dr. Benyó Balázs
Operációs rendszerek III.

Folyamatok párhuzamos végrehajtása

- egy fizikai processzor
- virtuális párhuzamos végrehajtás
- CPU ütemezés:
gyors váltás, futó folyamatok számára szinte észrevétlen
- UNIX:
időszeletet (time slice vagy quantum),
prioritásos, preemptív (kiszorításos)

hétfő, 2006. február 27.

Dr. Benyó Balázs
Operációs rendszerek III.

Folyamatok környezete

- Követelmény:
 - folyamatok megszakítása, ill. újraindítása
- Folyamatok környezete:
a folyamat és az őt végrehajtó gép állapota (ill. annak leírása).
 - programszámláló (program counter)
 - regiszterek,
 - Memory Management Unit
 - használt erőforrások stb.

hétfő, 2006. február 27.

Dr. Benyó Balázs
Operációs rendszerek III.

Folyamatok memóriája a UNIX-ban

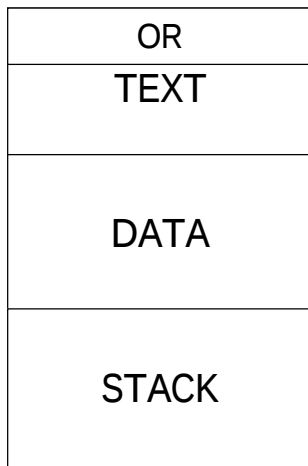
Folyamatok memóriája

- Egyszerre több folyamat a fizikai memóriában.
- Fogalmak:
 - logikai memóriakép
 - fizikai memóriakép
 - virtuális memóriakezelés/tárcsere
- *OR feladata:*
 - memória cím–memória tartalom összerendelés
 - memória tartalom fizikai memóriába mozgatása

hétfő, 2006. február 27.

Dr. Benyó Balázs
Operációs rendszerek III.

Memória használat

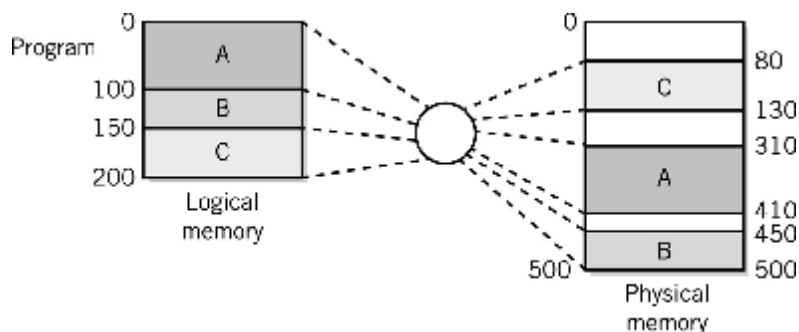


hétfő, 2006. február 27.

- program kód:
 - statikus
 - dinamikus (DLL)
- adatterületek:
 - inicializált (DATA)
 - nem inicializált (BSS)
 - heap (dinamikus adatok)
- veremtár:
 - dinamikusan növelhető terület

Dr. Benyó Balázs
Operációs rendszerek III.

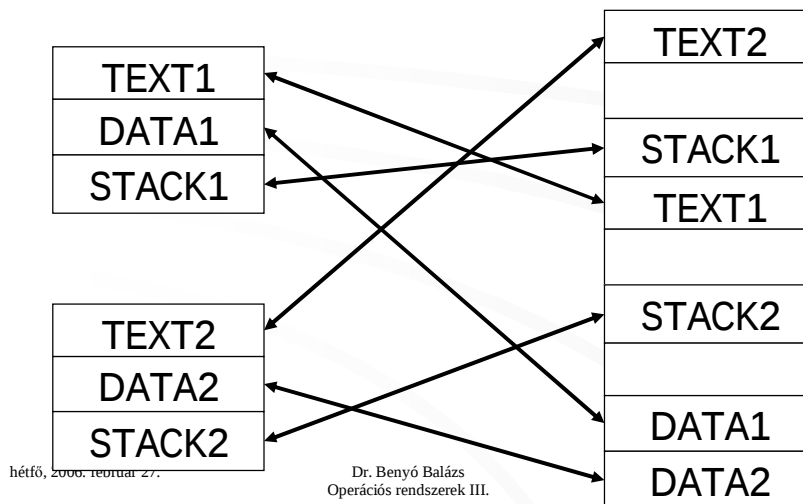
Logikai-fizikai címtranszformáció



hétfő, 2006. február 27.

Dr. Benyó Balázs
Operációs rendszerek III.

Logikai-fizikai címtranszformáció



UNIX memória kezelés

- Logikai – fizikai címtranszformáció:
 - folyamat leíró adatokban:
 - Region Table
 - Memory Management Unit

hétfő, 2006. február 27.

Dr. Benyó Balázs
Operációs rendszerek III.

A folyamatok környezetének adatelemei UNIX-ban

A folyamatok környezetének adatelemei

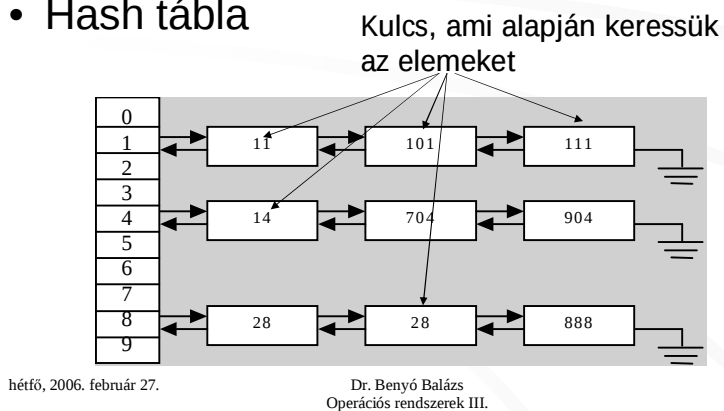
- Minden adat, ami szükséges a folyamat újraindításához.
- Két adatstruktúrában tárolódik:
 - *u area (user area, felhasználói terület)*
processz memóriájában
 - *proc (processz) struktúra*
rendszer memóriájában
 - *Tárolás: hash tábla*

hétfő, 2006. február 27.

Dr. Benyó Balázs
Operációs rendszerek III.

Dinamikus adatszerkezetek

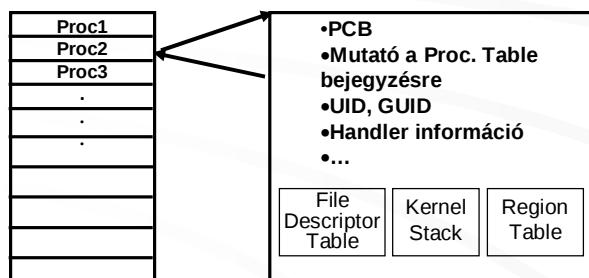
- Láncolt lista
- Hash tábla



Folyamat környezet leíró adatok

Processz-táblázat

Proc2-höz tartozó u area



hétfő, 2006. február 27.

Dr. Benyó Balázs
Operációs rendszerek III.

A proc struktúra elemei I.

- processz azonosító (PID)
- user area-ra mutató referencia
- processz állapota:
 - kernel vagy user módban fut, várakozó vagy futásra kész, stb. (l. állapot diagramm)
- ütemezési információ
 - *p_cpu, p_usrpri, ...* (l. ütemezés)
- ütemezésnél használt egyéb mutatók (pl. milyen prioritási kategóriában van)

hétfő, 2006. február 27.

Dr. Benyó Balázs
Operációs rendszerek III.

A proc struktúra elemei II.

- signalok kezelésére vonatkozó információ:
 - *ignored-blocked-posted-handled* (l. signalok)
- MMU információ
 - milyen MMU regisztereket kell frissíteni a folyamat futtatásakor)
- a processzek különböző listáihoz aktív, zombie, szabad) tartozó linkek (l. állapot diagram)
- jelzőbitek

hétfő, 2006. február 27.

Dr. Benyó Balázs
Operációs rendszerek III.

A proc struktúra elemei III.

- a PID alapján *hash táblában* történő tároláshoz szükséges mutatók
- a processzek hierarchiájában elfoglalt helyre vonatkozó információ:
 - processz szülője, processz gyermekei, stb.

hétfő, 2006. február 27.

Dr. Benyó Balázs
Operációs rendszerek III.

A u area elemei I.

Egyszerű adatalemek:

- *PCB (hardver környezet, regiszterek)*
- *mutató a folyamat proc struktúrájára*
- *jogosítványok (valós és hatásos UID, GID)*
- *signal handler (kezelő) rutinok címei*
- *CPU használati statisztika (ütemezéshez)*
- *környezeti változók (aktuális könyvtár, kontroll terminál, keresési út, stb.)*

hétfő, 2006. február 27.

Dr. Benyó Balázs
Operációs rendszerek III.

A u area elemei II.

Összetett adatelemek:

- *a folyamathoz tartozó memória (a felhasználói címtartomány) elérésére szolgáló táblázat (címtranszformációs tábla, per-process Region Table)*
- *fájl-leírók táblázata (per process File Descriptor Table)*
- *kernel módban használt veremtár (per process Kernel Stack)*

hétfő, 2006. február 27.

Dr. Benyó Balázs
Operációs rendszerek III.

Összetett adatelemek

- **Kernel Stack:**
 - verem tár
 - kernel (privilegizált) módban futáskor
 - (rendszerhívások, IT stb.)
- **Region Table:**
 - MMU címtranszformációs táblázat
- **File Descriptor Table:**
 - A folyamat által megnyitott fájlok eléréséhez szükséges információ tárolása.

hétfő, 2006. február 27.

Dr. Benyó Balázs
Operációs rendszerek III.

A hardver környezet (PCB)

- CPU regiszterei
- utasításszámláló (PC)
- stack pointer (SP)
- processzor-állapotleíró bitek (process status word), pl. carry (átvitel) bit
- memóriakezelő egység regiszterei (Memory Management Units, MMU)
- Lebegőpontos egység (Floating Point Units) regiszterei

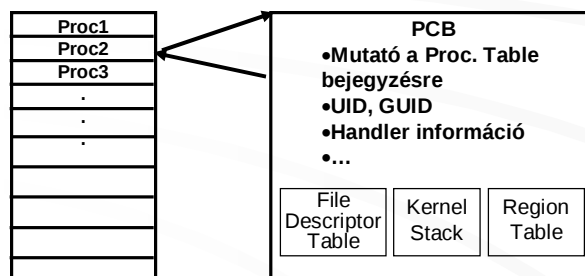
hétfő, 2006. február 27.

Dr. Benyó Balázs
Operációs rendszerek III.

Folyamat környezet leíró adatok

Processz-táblázat

Proc2-höz tartozó u area

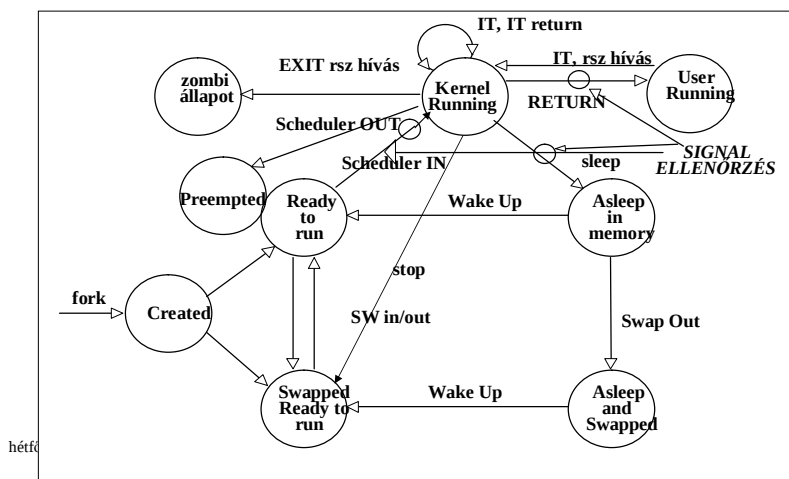


hétfő, 2006. február 27.

Dr. Benyó Balázs
Operációs rendszerek III.

Folyamatok állapot-átmeneti diagramja

Folyamatok állapot-átmeneti diagramja



Folyamatok állapot-átmeneti diagramja II.

Új folyamat keletkezése:

- *fork()* rendszerhívás
 - *gyermek folyamat*
 - *fork()* visszatérési értéke: 0
 - *szülő folyamatnak*
 - *fork()* visszatérési értéke: *gyerek PID-je*
- *Ősfolyamat: init (PID=0)*
- *scheduler (ütemező, PID=1)*

hétfő, 2006. február 27.

Dr. Benyó Balázs
Operációs rendszerek III.

Folyamat indulása

```
int result=fork();
if(result==0)/*visszatérési érték vizsgálata*/
{
    printf("GYERMEK VAGYOK");
    exec("gyermek_kód");
    /*a gyermek végrehajtása*/
}
else
{
    if(result<0)
    {
        printf("HIBA");
    }
    else
    {
        printf("Anya proc:gyermekem PID-je:%d",result);
        /*ha a visszatérési érték >0, kiírja */
    }
}
```

Folyamatok állapotai I.

- *created (létrehozott) állapot*
- *ready to run (futásra kész) állapot*
- *kernel running*
- *user running*
- *asleep in memory (memóriában alvó)*
 - *sleep rendszerhívás*
 - *wake up átmenet*

hétfő, 2006. február 27.

Dr. Benyó Balázs
Operációs rendszerek III.

Folyamatok állapotai II.

- *swapped out*
 - *swapped out and asleep*
 - *swapped out and ready to run*

Preemptív ütemezés:

- *kernel running à ready to run*
- *preemptív (erőszakosan megszakított, kiszorított) állapot*

Futás befejezése: zombie állapot

hétfő, 2006. február 27.

Dr. Benyó Balázs
Operációs rendszerek III.

A UNIX OR feladatainak megvalósítása

Az OR feladatai

- A UNIX, mint operációs rendszer a feladatait a következő négy formában látja el:
- Kernel:
 - *rendszerhívások (system call)*
 - *kivétel (exception)*
 - *hardver megszakítások (hardware interrupt)*
 - *(call-out függvények)*
- Önálló folyamat:
 - *daemon folyamatok*

Rendszerhívások

- UNIX programozói interfésze.
(*API, application programming interface*)
- Operációsrendszer szolgáltatásainak elérése:
 - erőforrás foglalás,
 - rendszerinformációk gyűjtése, elszámolás biztosítása,
 - védelmi és biztonsági mechanizmusok biztosítása.
- Privilegizált futási mód.
- Rendszer és folyamat erőforrásainak használata.

hétfő, 2006. február 27.

Dr. Benyó Balázs
Operációs rendszerek III.

Kivétel (exception)

- Pl. nullával történő osztás, memóriaterületről történő kicímzés, stb.
- Utasítás végrehajtással szinkron események.
- Kivételek kezelése kernel feladat.

hétfő, 2006. február 27.

Dr. Benyó Balázs
Operációs rendszerek III.

Hardver megszakítások (hardware interrupt)

- Perifériák (I/O egység) állapotának megváltozása.
- IT kezelés kernel feladat (csak rendszer erőforrásokkal!).
- Aszinkron módon fellépő eseményekre a rendszer reagálni tudjon
 - interrupt rutin lefuttatása,
 - nincs *környezetváltást (context switch)*.

hétfő, 2006. február 27.

Dr. Benyó Balázs
Operációs rendszerek III.

Az interrupt rutin által végzett tipikus műveletek

1. (végrehajtási mód váltása: user → kernel)
2. Elmenti az aktuális processzor környezetet.
3. IT kiváltó esemény azonosítása.
4. IT kiszolgáló rutin meghatározása.
5. IT rutin hívása.
6. Környezet visszaállítása.
7. (végrehajtási mód váltás: kernel → user)

hétfő, 2006. február 27.

Dr. Benyó Balázs
Operációs rendszerek III.

Óra-interrupt

- Az ütemező hardver-támogatása: az óra-interrupt, a hardver óra adott időközönként megszakítást okoz.
- Gyakorisága: *tikk* (óra-tikk, CPU-tikk): 10 ms (rendszerfüggő: HZ).

hétfő, 2006. február 27.

Dr. Benyó Balázs
Operációs rendszerek III.

Az óramegszakítás-kezelő fő feladatai I.

- A hardver óra újraindítása, amennyiben ez szükséges.
- Az aktuálisan futó folyamat CPU használati statisztikájának frissítése
prioritás újraszámítása,
időszlet lejártának kezelése stb.
- *SIGXCPU signal* küldése az aktuális folyamatnak, amennyiben az túllépte az idő kvótáját.

hétfő, 2006. február 27.

Dr. Benyó Balázs
Operációs rendszerek III.

Az óramegszakítás-kezelő fő feladatai II.

- A rendszer óráinak frissítése (napi óra).
- A *call-out* függvények ellenőrzése.
- Bizonyos rendszerfolyamatok kezelése (swapper, pagedaemon).
- Riasztások (*alarm*) kezelése.

hétfő, 2006. február 27.

Dr. Benyó Balázs
Operációs rendszerek III.

Daemon folyamatok

- Önálló folyamatokként megvalósított OR funkciók.
- Nem privilegizált módban futnak.
- Rendszerhívásokon keresztül érik el a kernel szolgáltatásait.

hétfő, 2006. február 27.

Dr. Benyó Balázs
Operációs rendszerek III.

Folyamatok futási környezete I.

- Folyamatok logikai memóriaképe:
 - operációs rendszer által használt memória,
 - folyamat által használt memória.
- Memória védelme:
 - rendszer környezet (*system context*) és
 - felhasználói környezet (*user context*) megkülönböztetése Java VM (hordozható!), PC emulátorok

hétfő, 2006. február 27.

Dr. Benyó Balázs
Operációs rendszerek III.

Folyamatok futási környezete II.

- Rendszer futási környezet:
 - operációs rendszer memóriaterületei,
 - kernel erőforrásai,
 - kernel stack
- Felhasználói futási környezet:
 - alkalmazás adatterületei,
 - user stack.

hétfő, 2006. február 27.

Dr. Benyó Balázs
Operációs rendszerek III.

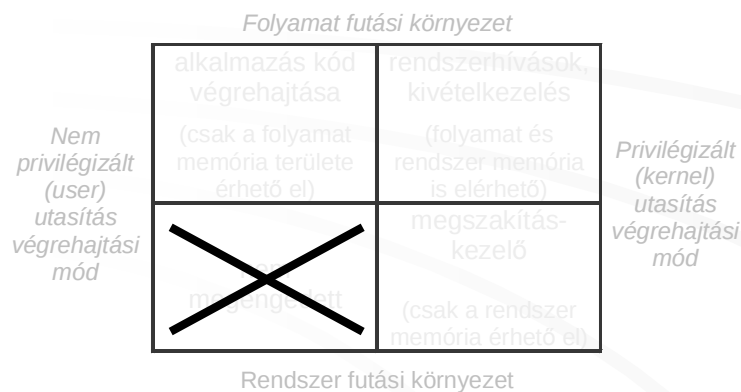
Folyamatok futási környezete III.

- A processzor különböző utasítás végrehajtási módjai:
 - felhasználói (nem privilegizált) mód (user mode)
 - Néhány speciális utasítás végrehajtása és a memória adott részének elérése tiltott.
 - privilegizált, kernel mód (kernel mode).

hétfő, 2006. február 27.

Dr. Benyó Balázs
Operációs rendszerek III.

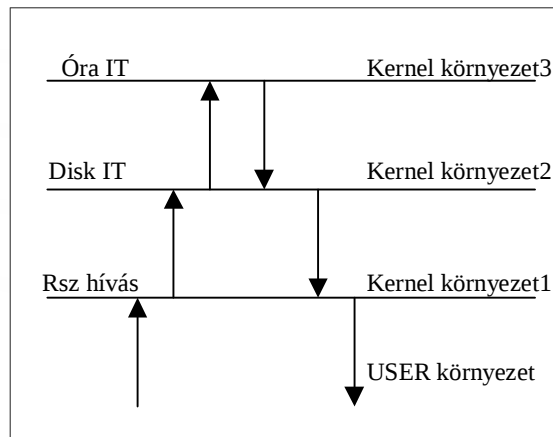
Folyamatok futási környezete IV.



hétfő, 2006. február 27.

Dr. Benyó Balázs
Operációs rendszerek III.

Környezetek érvényessége



hétfő, 2006. február 27.

Dr. Benyó Balázs
Operációs rendszerek III.