

## 6. Operační systém a plánování procesů

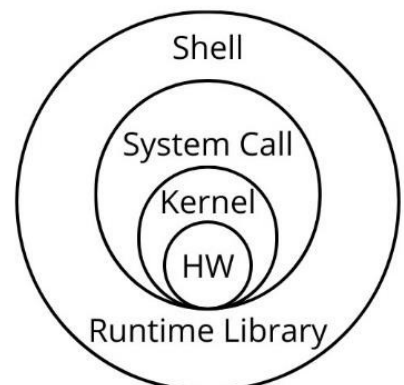
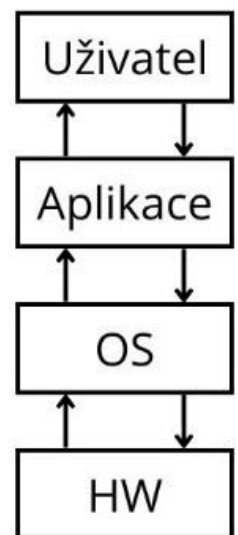
- Charakteristika OS
- Typy jader OS
- Proces vs. vlákno
  - Charakteristika
  - PCB vs. TCB
- Přepínání kontextu
- Plánovače OS
  - Preemptivní vs. nepreemptivní plánování
- Plánovací algoritmy
  - FCFS
  - SJF
  - SRTF
  - RR
  - OPS
  - MFQS

### Charakteristika OS

- Základní programové vybavení PC
  - Kolekce programů tvořící spojující vrstvu mezi HW a uživatelskými programy
- Při startu je zaveden do OP, kde zůstane až do vypnutí
- Zajišťuje abstrakci pro různorodý HW
- Zajišťuje správu
  - Procesů
  - Paměti
  - Souborového systému
  - Periferií
  - UI
- Shell=terminál
- Runtime library = knihovna s příkazy určené pro uživatele pro komunikaci s kernelem

### Jádro OS

- Říká se mu CORE nebo Kernel
- Je to nejnižší, nejdůležitější a nejzákladnější část OS
- Systémové volání
  - Využívá systém nebo uživatelské programy pro služby jádra
  - Speciální instrukce OS
  - Kontrolovaný přechod do režimu jádra
- Volání jádra
  - Kernel interface→přímé volání spec Instrukce
  - Library Interface→Využití systémových knihoven→fopen, fwrite

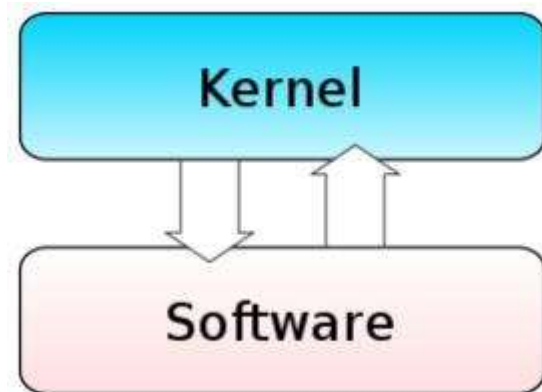


- Služby jádra
  - Open/close
  - Read/write
  - Kill[PID]
  - Fork – vytvoření potomka → duplikace
  - Exec
  - Exit

## Typy jader

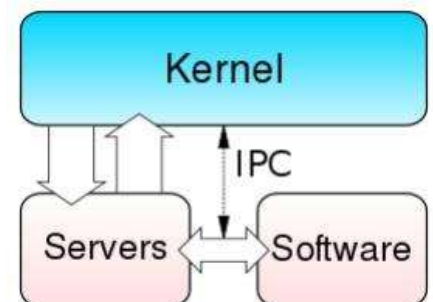
### Monolitické

- Modulární
  - MS-DOS
  - WIN95/98, MAC OS do verze 8.6
- Nemodulární
  - Linux, Net BSD, FreeBSD
- Komplexní jádro
  - Privilegovaný režim
    - Všechny subsystémy implementující služby jsou tak těsně provázány → vysoká efektivita
  - Dynamické nahrávání modulů
    - Vylepšení koncepce
    - Možnost nahrání za běhu bez nutnosti restartu
    - Zavedeny do adresního prostoru jádra
    - Jisté zpoždění



### Mikrojádru

- Obsahuje jen základní služby
  - Správa procesů a paměti
  - Meziprocesní komunikace
- Ostatní služby v podobě serverů běží v uživatelském režimu
  - Správa souborového systému
  - Ovladače ařízení
  - Rozšířená správa paměti
  - Síťové protokoly
- Jednodušší na programování a bezpečnější
  - Chyba v subsystému neznamena pád systému
- Vysoká rezie
  - Volání služeb jádra je nahrazeno IPC komunikací mezi aplikačními a systémovými procesy
    - Vyšší počet přechodů při změně kontextu
    - Přechod mezi privilegovaným a uživ. Režimem
  - Celkem 2 generace vývoje
  - Jako celek se neprosadilo

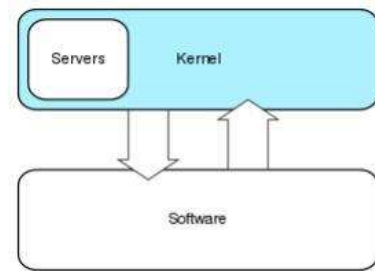


### Hybridní

- Kombinace vlastností/výhod monolitického a mikrojádru
- Mikrojádru doplněno o příslušné služby v podobě serverů

- Snížení režie
- Souborový systém, síťový protokol
- Dynamické přidávání modulů
- Většina dnešních OS

Micro jádro je rozšířeno o kód, který by mohlo běžet v podobě serveru v uživatelském režimu, ale za účelem zmenšení režie IPC je těsněji provázán a běží v privilegovaném režimu v podobě serveru



## Exo

- Experimentální, poskytující velmi nízké rozhraní, zaměřené hlavně na bezpečné sdílení prostředků
- Menší než mikro
- Využíváno v MIT

## Nano

- Menší než mikro
- Služby řešeny jako ovladače, umístěny mimo jádro

## Proces x vlákno

- Proces běžící program
- Vlákno → Odlehčený proces, nemůže běžet samostatně

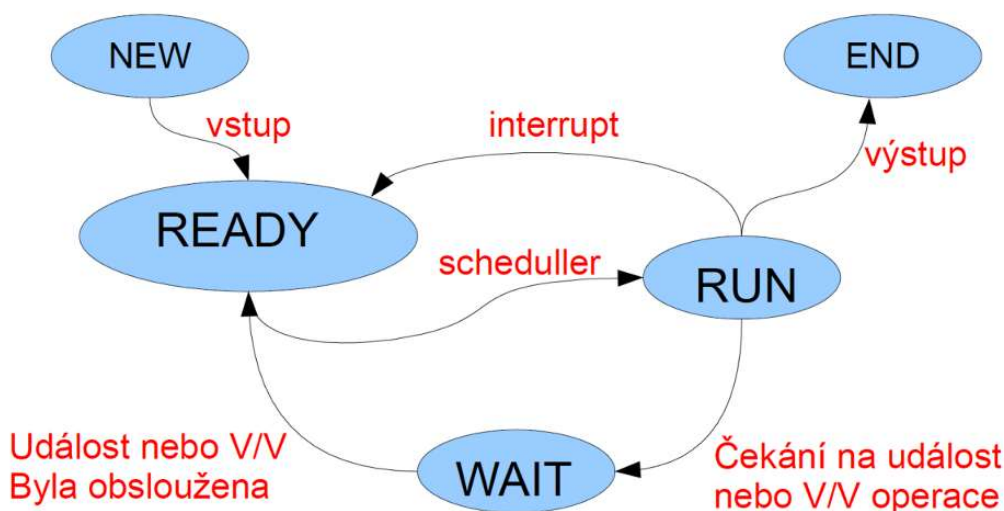
## Proces

- Po dobu jeho běhu je umístěn v OP
- V OS je definován (PCB)
  - PID
  - Programem, kterým je řízen
  - Obsahem registrů
    - EIP → Čítač registrů
    - ESP → Adresa zásobníku
    - Záloha registrů z CPU, protože ty budou přepsány jiným procesem
  - Daty (konstanty, proměnné)
    - Limit paměti
  - Použitím dalších zdrojů OS a vazbou na jeho objekty
    - UID, GID, signály, soubory
    - Seznam otevřených souborů
- Proces – správa
  - Process management
  - Context switch
    - Dispatcher
  - Memory management
  - IPC komunikace
    - Inter process Communication
- Plánování

- Přidělován strojový čas
  - Doba využití CPU než dojde k přepnutí kontextu (Interrupt ho pošle na WAIT nebo READY)
- Jeden proces může v OS běžet víckrát
  - Třeba chrome, souborový systém
  - Respektive jeden program má spuštěných více procesů(jede paralelně víckrát)

## Vlákno

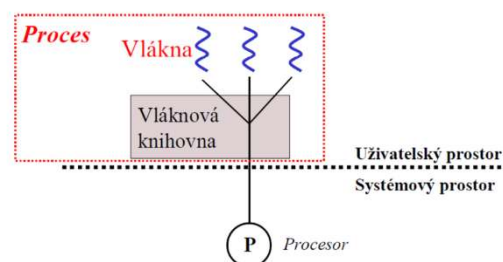
- Odlehčený proces
  - Thread
  - Sám nemůže existovat
- TCB
  - TID
  - Ukazatel na proces
  - Ukazatel na aktuální instrukci vlákna
  - Stav vlákna
  - registry
- Snížení režie
  - Méně časté přepínání kontextu
- Společný adresní prostor (paměťový)
  - Menší nároky na paměť
  - Stejná práva v rámci procesu
  - Jednodušší předávání informací
- Podléhají plánování obdobně jako procesy
- V OS bez podpory vláken 1 vlákno = 1 proces
- Podpora a použití vláken urychluje samotný běh procesu
- Vlákno je možné použít na uživatelské úrovni nebo na úrovni jádra OS
  - Některé OS podporují obě
    - Windows 2000/XP, Free BSD
- Výhody
  - Urychlení výpočtů a odezvy celkového běhu programu
  - Efektivní využití systému
    - Jedno vlákno pracuje s diskem zatímco druhé čeká na data z Cache
  - Přehlednější strukturalizace programu
- Nevýhody
  - Složitě sledování toku programu
  - Omezení počtu vytvoření vláken
  - Synchronizace



#### Vlákno na uživatelské úrovni

- OS o nich neví
  - Nezávislé na OS
  - Plně v režii programátora
- Pro přepínání není nutno volat Jádro OS
- Lze použít i v O, který neobsahuje žádnou podporu vláken
  - Nutno použít Thread library
- Výhody
  - Rychlé přepínání
  - Rychlá tvorba a jejich zánik
  - Uživatelský proces nad nimi má plnou kontrolu
- Nevýhody
  - Jádro OS o nich neví, přiděluje čas celému procesu
    - Dva vlákna stejného procesu nemohou běžet současně
  - Při volání služby blokuje všechny ostatní vlákna

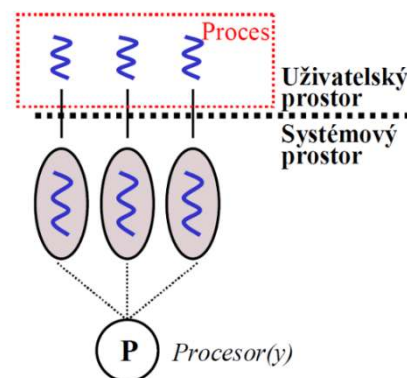
#### Vlákna na uživatelské úrovni



#### Vlákno na úrovni OS

- O vše se stará OS
  - Plánování, přepínání, rušení
- Jeden proces může využít více CPU
- Volání služby neblokuje ostatní vlákna procesu
- Náročnější na správu
- Často nespravedlivé plánování
  - Strojový čas je přidělován vláknům

#### Vlákna na úrovni jádra OS



#### Přepínání kontextu

Přepínání kontextu je operace, při které operační systém ukládá stav běžícího procesu nebo vlákna a načítá stav jiného procesu nebo vlákna, aby mohl pokračovat ve svém provádění.

## Plánovače OS → Preemptivní x Ne preemptivní

- Preemptivní → OS může přerušit úlohu a na CPU dát jinou
- Ne preemptivní → Běží dokud úloha neskončí, nebo se sama nepřeruší (Hrozí zaseknutí)

## Plánovací Algoritmy

- FCFS
  - Fronta, First Come First Served
- SJF
  - Shortest Job First
  - Ne preemptivní
  - Vybere se nejkratší
- SRTF
  - Shortest Remaining time first
  - Funguje jako SJF, ale je preemptivní
    - Přejde kratší úloha tak tam dá kratší
- RR
  - Round Robin
  - Dokolečka, všichni stejné časové kvantum
- PS
  - Priority Scheduling
  - Podle jejich priority
- MFQS
  - Multilevel Feedback Queue Scheduling
  - Procesy rozděleny do několika front podle priority, každá fronta má svůj RR s časovým kvantem
  - Procesy se můžou přesouvat mezi frontami na základě jejich chování
  - Nižší fronty mohou jet až jsou vyšší prázdné