# Správa procesů a paměti v operačních systémech (6)

## Typologie a charakteristika OS

**Co je to operační systém?**

Správce fyzických prostředků daného systému, který zpracovává pomocí logických prostředků úlohy zadané uživatelem. Softwarová platforma = operační systém

**Funkce OS:**

* Správa paměti
* Správa procesů
* Správa periferií
* Správa systému
* Správa souborů
* Správa uživatelů
* Správa úloh
* Uživatelské rozhraní
* Programové rozhraní (API)

**Rozdělení OS:**

1. Podle počtu ovládaných procesorů
   1. Jednoprocesorové
   2. Víceprocesorové
2. Podle složitosti správy uživatelů
   1. Jednouživatelské
   2. Víceuživatelské
3. Podle počtu spuštěných programů
   1. Jednoprogramové
   2. Víceprogramové
4. Podle míry specializace
   1. Speciální
   2. Univerzální

**Dále rozlišujeme podskupiny OS:**

* Realtimové OS

Pracují téměř v reálném čase, použití tam, kde jsou vysoké nároky na interaktivitu systému (např. systémy řízení letadel, elektráren, laboratoří), příklady OS: QNX, RT Linux

* Distribuované OS

Pracuje na více než jednom procesoru, má svůj program rozdělen na části

Distribuovanost = možnost co nejvíce rozložit výpočet systému na více míst, které pracují paralelně

**Cloud computing a OS**

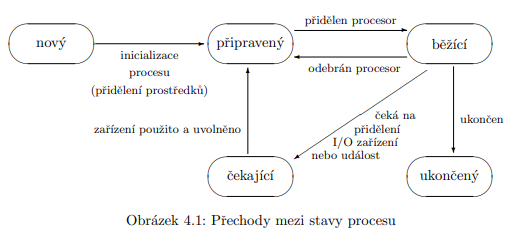
= provozování služeb, aplikací a úložišť na Internetu

OS, které běží v cloudu: ICloud, OOS, Glide OS

## Evidence procesů

Proces = instance programu určená nejen jeho kódem, ale i dalšími vlastnostmi (stav, priorita, ID, přidělené prostředky)

**Stavy procesu:**

* Nový – proces byl právě vytvořen
* Běžící – proces má právě přidělen procesor (je vykonáván)
* Připravený – čeká na přidělení

procesoru

* Čekající – čeká na přístup k I/O

prostředku (např. stisk klávesy)

* Ukončený – proces byl ukončen

Dále mohou být definovány stavy **pozastavený** (provádění programu je pozastaveno, např. debugging), **zombie** (program byl ukončen, ale jeho prostředky – včetně ID, nebyly ještě uvolněny), **uspaný** (proces čeká na splnění podmínky)

Úloha – bývá to obvykle jedno vlákno aplikace

**Evidence procesů ve Windows**

Používají se tyto datové struktury: EPROCESS, ETHREAD, PEB, TEB, KPROCESS

**Evidence procesů v Linuxu**

Používají se tyto datové struktury: task, task\_struct

## Správa procesů

Správce procesů vede tabulku procesů. Záznam v této tabulce se nazývá Process Controll Block (PCB). PCB obvykle obsahuje:

* PID (ID procesu)
* Stav procesu
* Programový čítač určující, která instrukce se právě provádí
* Hodnoty registrů
* Ukazatele do front
* Informace správce paměti
* Účtovací informace
* Další přidělené prostředky (zařízení, soubory)

**Priority procesů**

Každý proces má přiřazenu prioritu. Ta slouží k plánování přidělování procesoru.

Windows – priority 0 – 31

0 systémová úroveň, používá se pro vlákno nulové stránky

1-15 dynamické běžící procesy   
16-31 procesy reálného času

Linux – priority

1 – 40 pro běžné procesy

1 – 99 realtimové procesy

## Správce paměti

Paměť = vnitřní paměť (operační)

Funkce správce paměti:

* Udržuje informace o paměti
* Přiděluje procesům paměť
* Uvolněnou paměť přiřazuje do volné paměti
* Odebírá procesům paměť
* Může zajišťovat ochranu paměti

Každý proces má přidělen paměťový prostor v rozsahu určitých adres (adresový prostor):

* Fyzický adresový prostor (AP) – AP, který je fyzicky k dispozici ve výpočetním systému
* Logický AP – AP, který mají k dispozici procesy

## Reálné metody přidělování paměti

Používají se v případech, kdy log. AP nepřekračuje fyz. AP

1. Přidělení jedné souvislé oblasti paměti – spočívá v přidělení veškerého adresového prostoru kromě oblasti OS (Výhody (V): jednoduchost; Nevýhody (N): nemožnost míst spuštěno více programů)
2. Přidělování bloků pevné velikosti – V: jednoduchost, možnost multitaskingu; N: proces vyžadující více paměti, než je délka největšího volného bloku nelze spustit
3. Dynamické přidělování bloků paměti – V: částečně odstraňuje nevýhody předchozí metody; N: pravděpodobnost fragmentace, počet spustitelných procesů je limitován požadavky již spuštěných procesů
4. Segmentace – každému procesu je přiřazeno několik různě dlouhých bloků paměti (segmentů) V: velikost segmentů může být různá, segmenty lze prodlužovat a přesouvat; N: nutnost HW podpory, komplikovanější ochrana paměti
5. Jednoduché stránkování – pracuje s fyz. a log. adresou objektu v paměti, paměťový prostor je rozdělen na stejně dlouhé úseky a procesu je přiděleno tolik úseků, kolik potřebuje V: nejsou problémy s fragmentací; N: omezení daná velikostí fyz. adresového prostoru

## Virtuální paměť

= koncept, kdy vnitřní paměť rozšíříme o oblast na vnější paměti (obvykle HDD)

Důvodem může být nemožnost spustit proces, protože má příliš vysoké požadavky na vnitřní paměť

## Defragmentace

Paměť je fragmentovaná, pokud volné oblasti paměti netvoří souvislý blok

Řešení:

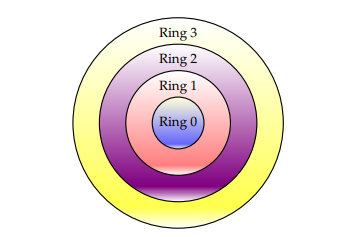
* Výběr vhodného bloku paměti (při spuštění procesu se hledá vhodně velký blok paměti)
  + Metoda *first fit* – správce paměti přidělí paměť z prvního volného bloku
  + Metoda *best fit* – správce paměti projde všechny bloky a přiřadí nejvhodnější
  + Metoda *last fit* – správce přidělí poslední vyhovující blok
* Setřásání paměti (přesouvání bloků paměti, které jsou neobsazené tak, aby po přesunutí bylo možné vytvořit větší volný blok spojením menších bloků)

## Stránkování

Fyz. vnitřní paměť je rozdělena na *rámce*, logická paměť je rozdělena na *stránky*

1. **Na žádost** – používají se různé metody při určování, který rámec má byt uvolněn, aby se vytvořilo místo v operační paměti (HW závislé řešení)
   1. FIFO (First In, First Out) – odložena je stránka, která má přiřazen rámec nejdéle
   2. LFU (Last Frequently Used) – odložena je nejméně používaná stránka
   3. LRU (Last Recently Used) – odložena je nejdéle nepoužívaná stránka
   4. NUR (Not Used Recently) – používá se tzv.  *used bit*
2. **Segmentace se stránkováním na žádost** – proces má přiděleno několik segmentů, každý segment se může rozkládat na několika stránkách

## Ring 0-3 (HW ochrana prostředků)

Každý proces běží v některém z Ringů, což určuje možnosti přístupu k chráněným prostředkům (paměť, I/O porty, HW).

Ring 0 = režim jádra,  
Ring 3 = uživatelský režim   
(většina OS používá jen Ringy 0 a 3)

## Swapování procesů

Je to jednoduchá metoda virtualizace, která spočívá v tom, že se neodkládají jednotlivé stránky paměti, ale vždy celý paměťový prostor odkládaného procesu.

Výhody: jednoduchost

Nevýhody: HW závislé řešení, přesouvají se zbytečně velké paměťové bloky

## Endianita

Určuje pořadí ukládání jednotlivých bajtů příslušného datového typu. Jedná se o způsob, který definuje, v jakém pořadí jsou v operační paměti ukládány jednotlivé řady čísel, které mají vice jak jeden bajt. V praxi se můžete setkat i s pojmem byte order (pořadí bajtů).