

M18 Virtuální paměť a paměť cache

- virtuální paměť
 - technika dovolující programům využívat více paměti než je jí fyzické
 - koncept vytváří iluzi velkého, spojitého adresového prostoru pro každou běžící aplikaci
 - paging
 - virt. pam. prostor je rozdělen na malé bloky nazývané stránky (pages)
 - RAM je rozdělena do bloků stejné velikosti jako v prostoru - rámce
 - když program potřebuje přístup k určité stránce, OS zjistí, zda je tato stránka již načtena do rámce; pokud není, dojde k výpadku stránky (page fault) a OS načte potřebnou stránku z diskového úložiště do volného rámce ve fyzické paměti
 - segmentace
 - virtuální paměť může být také rozdělena do segmentů různé velikosti odpovídající logickým částem programu (kód, data a zásobník)
 - segmenty mohou být umístěny kdekoli v adresním prostoru
 - segmenty mohou růst nebo se zmenšovat podle potřeby
 - page tables
 - OS používá tabulky stránek k mapování virtuálních adres na fyzické adresy
 - tabulka stránek obsahuje záznamy určující kde je každá stránka virtuální paměti uložena ve fyzické paměti
 - když procesor potřebuje přístup k určité virtuální adrese, nejprve zkontroluje tabulku stránek, aby zjistil odpovídající fyzickou adresu
 - swapování - pokud není dostatek fyzické paměti pro všechny aktuálně běžící procesy, operační systém může přenést některé stránky z fyzické paměti na disk do swapovacího prostoru
 - každý proces má svůj vlastní virtuální adresní prostor - vyšší bezpečnost a stabilita systému
 - ochrana paměti mezi procesy - jeden proces nemůže narušit paměť jiného procesu
- cache
 - rychlá paměť, která slouží k dočasnému ukládání často používaných dat nebo instrukcí
 - umístěna mezi procesorem a hlavní pamětí (RAM)
 - cílem je zrychlit výkon systému tím, že sníží dobu potřebnou k přístupu k datům
 - multilevel cache
 - v moderních procesorech
 - každá úroveň má různé charakteristiky (rychlost, velikost)
 - určena k optimalizaci přístupu k datům na různé úrovni hierarchie paměti
 - uspořádaná do úrovní (L1, L2, L3); L1 - nejmenší a nejrychlejší; L3 - největší a nejpomalejší

Logické souvislosti s cache

- logické souvislosti určují, jaká data by měla být uložena v cache a jak by měla být spravována
- princip prostorové lokalizace (Spatial Locality)
 - pokud byl přístup k určité adrese v paměti, je vysoká pravděpodobnost, že blízké adresy budou také přístupné brzy
 - cache často ukládá bloky paměti (například několik po sobě jdoucích adres) namísto jednotlivých adres
- princip časové lokalizace (Temporal Locality)
 - pokud byla nedávno přístupná určitá paměťová adresa, je vysoká pravděpodobnost, že k ní bude znovu přístup v blízké budoucnosti
 - cache uchovává nedávno přístupná data, aby mohla být rychleji dostupná při opakovaných přístupech
- hierarchie paměti
 - různé úrovně paměti mají různé rychlosti a velikosti
 - nejrychlejší a nejmenší paměť je umístěna nejbližší procesoru (L1 cache); pomalejší a větší paměť je dále (RAM, pevný disk)
- algoritmy pro správu cache
 - určují, která data budou uložena a která budou odstraněna, když je cache plná
 - algoritmy využívají logické strategie pro maximalizaci cache hit rate

- LRU (Least Recently Used) - nahrazuje nejméně nedávno použitá data
- FIFO (First In, First Out) - nahrazuje data v pořadí, v jakém byla do cache uložena
- LFU (Least Frequently Used) - nahrazuje data, která byla nejméně často přistupována; kombinace časové a prostorové lokalizace
- přednačítání (Prefetching) - cache může předem načíst data, která ještě nebyla požadována, ale jsou pravděpodobně potřebná v blízké budoucnosti

Konzistence dat v cache

- nesmí dojít k situacím, kdy jsou stará nebo neplatná data používána místo aktuálních dat
- write strategy
 - strategie zápisu určují, jakým způsobem jsou data zapisována
 - write-through
 - data jsou zapisována do cache a zároveň okamžitě do RAM
 - vysoký úroveň konzistence; pomalejší
 - write-back
 - data jsou zapisována pouze do cache a do RAM jsou zapsána až tehdy, když jsou z cache odstraněna
 - rychlejší; složitější správa paměti
- coherence protocols (Protokoly pro zajištění konzistence)
 - v systémech s více cache
 - MESI Protocol (Modified, Exclusive, Shared, Invalid)
 - každá cache line (řádek) může být ve čtyřech stavech: Modified (M), Exclusive (E), Shared (S), Invalid (I)
 - pokud je jedna cache line v modifikovaném stavu, žádná jiná cache nemá kopii této cache line ve stavu exclusive nebo shared
 - MOESI Protocol (Modified, Owner, Exclusive, Shared, Invalid) - rozšiřuje MESI protokol o stav Owner (O)
 - dragon protocol\
 - v multiprocessorových systémech
 - v situacích, kdy více procesorů sdílí stejné paměťové adresy
- synchronizace a invalidace
 - invalidate on write - při zápisu do cache jsou ostatní kopie dané cache line invalidovány; žádná jiná cache nemůže používat zastaralá data
 - update on write - při zápisu do cache jsou aktualizovány i ostatní kopie dané cache line; všechny cache mají aktuální data
- atomic operations
 - zejména v systémech s více procesory nebo jádry
 - zajišťují, že při čtení a zápisu dat nedochází k nekonzistentním stavům
- konzistenční protokoly v multiprocessorových systémech
 - snooping-based protocols - každá cache monitoruje (snoops) sběrnici pro zjištění operací, které mohou ovlivnit její vlastní kopie dat
 - directory-based protocols - používá centrální nebo distribuovaný adresář, který sleduje, která cache má kopii které paměťové adresy a spravuje aktualizace a invalidace dat

Vyřazování a aktualizace cache

- vyřazování
 - LRU (Least Recently Used) - nahrazuje nejméně nedávno použitá data
 - FIFO (First In, First Out) - nahrazuje data v pořadí, v jakém byla do cache uložena
 - LFU (Least Frequently Used) - nahrazuje data, která byla nejméně často přistupována; kombinace časové a prostorové lokalizace
 - random replacement - položka k vyřazení je vybrána náhodně; v některých případech efektivní, obvykle neposkytuje nejlepší výkon
- aktualizace
 - write-through
 - data jsou zapisována do cache a zároveň okamžitě do RAM

- vysoký úroveň konzistence; pomalejší
- write-back
 - data jsou zapisována pouze do cache a do RAM jsou zapsána až tehdy, když jsou z cache odstraněna
 - rychlejší; složitější správa paměti

Adresa na sběrnici

- každý proces běžící v systému používá virtuální adresy, které jsou nezávislé na fyzické paměti; každý proces má svůj vlastní adresní prostor
- tato virtuální adresa je předána správci paměti, který ji přeloží na fyzickou adresu pomocí tabulky stránek
- virtuální adresa je rozdělena na číslo stránky a offset; číslo stránky je využito k určení stránky; offset pak k určení buňky na stránce
- pokud má virtuální adresa 32 bitů a velikost stránky je 4 KB (12 bitů pro offset), pak zbývajících 20 bitů tvoří číslo stránky
- fyzická adresa je vytvořena kombinací čísla fyzického rámce a offsetu z původní virtuální adresy
- pokud je například číslo fyzického rámce 0x12345 a offset 0x678, výsledná fyzická adresa bude 0x12345678
- stránkovací mechanismy a optimalizace
 - TLB (Translation Lookaside Buffer)
 - malá vyrovnávací paměť v procesoru
 - uchovává nedávné překlady virtuálních adres na fyzické adresy
 - při TLB hit je překlad velmi rychlý - není potřeba přistupovat k tabulce stránek v paměti
 - při TLB miss musí procesor přistoupit k tabulce stránek - pomalejší
- postup převádění virtuální adresy na fyzickou
 1. procesor vygeneruje virtuální adresu
 2. virtuální adresa je rozdělena na číslo stránky a offset
 3. číslo stránky je použito k vyhledání záznamu v tabulce stránek
 4. pokud je záznam
 - nalezen v TLB - fyzická adresa je rychle získána
 - nenalezen v TLB - tabulka stránek je prohledána v paměti
 5. záznam z tabulky stránek poskytuje číslo fyzického rámce
 6. fyzická adresa je vytvořena kombinací fyzického rámce a offsetu
 7. fyzická adresa je použita k přístupu k datům v paměti