

# 18. Architektura IA32-Virtuální paměť

- a) účel, princip funkce, logická struktura
- b) metody přepočtu adres
- c) aplikace: operační systém a virtuální paměť

## 1 Úvod

**Paměť** je zařízení, které je schopné přijmout informace, uchovat je po danou dobu a na požádání je vydat.

**Polovodičová paměť** využívá vlastností polovodičových tranzistorů. Realizují se klopnými obvody (technologie TTL), nebo obnovováním elektrického náboje (CMOS).

**Ideální paměť** je dostatečně velká, rychlá, levná a udrží data i po vypnutí napájení.

## 2 Virtuální paměť

Virtuální paměť je způsob, který umožňuje sdílet operační paměť pro více procesů, aniž by nežádoucím způsobem ovlivňovali mezi sebou. Každý proces využívá vlastní adresový prostor a myslí si tedy, že v systému běží sám. Tento princip dovoluje využívat jinou velikost operační paměti, než je fyzicky v systému dostupná. Virtuální znamená proto, že je neskutečná- velikost skutečné operační paměti je jiná.

### 2.1 Účel

- řeší nedostatek operační paměti
- ochrana paměti (stránkovací mechanismus umožňuje ochranu na úrovni stránek)

### 2.2 Struktura

Pokud pracuje procesor v chráněném režimu, lze rozdělit správu paměti do dvou částí: na segmentaci a stránkování. Segmentace poskytuje mechanismus oddělení částí kódu, dat a zásobníku ve víceúlohovém systému tak, že se neovlivňují mezi sebou.

Stránkování poskytuje mechanismus pro realizaci virtuální paměti, která je založena na tom, že potřebné části běžícího programu jsou mapovány do fyzické operační paměti dle potřeby. Stránkování může být využito i pro odstínění mezi běžícími úlohami.

### 2.3 Princip funkce

Virtuální paměť je rozdělena na stránky, fyzická operační paměť na rámce. Stránky i rámce mají stejnou velikost. Pokud se stane, že není potřebná stránka v operační paměti, procesor vygeneruje přerušení č. 14 Page Fault (výpadek stránky) a OS se postará o natažení stránky do fyzické paměti. Pokud je v operační paměti ještě místo, je stránka nahrána na nějaké volné místo. Pokud není, musí být uvolněno- nějaká stránka musí být vytěsněna na externí médium. Nejčastěji se vytěsní stránka, která byla nejdéle nepoužita. Oblast disku, která se používá pro ukládání stránek se říká stránkovací soubor (odkládací soubor).

### 2.4 Ochrana paměti

Dalším efektem virtuální paměti je ochrana paměti, tedy stránky. Ta se kontroluje paralelně s překladem adresy.

Na úrovni stránek se kontrolují:

- Typ stránky (pouze pro čtení, nebo pro zápis i čtení)
- Přístupová práva (supervisor nebo uživatelský mód)

## 2.5 Stránkování vs. segmentace

Program pracuje s logickou adresou. Segmentací je logická adresa přepočítána na lineární adresu. Stránkování se na rozdíl od segmentace dá vypnout. Pokud je tedy vypnuté, je lineární adresa zároveň adresou fyzickou. Pokud je stránkování zapnuté, je lineární adresa přepočítána stránkovací jednotkou na fyzickou.

## 2.6 Logická struktura virtuální paměti

Stránkování je záležitost operačního systému. Procesor mu jen napomáhá svou stránkovací jednotkou v přepočtu lineární adresy na fyzickou. Protože přepočet adresy je relativně zdlouhavá činnost, je součástí stránkovací jednotky TLB. TLB (Translation Lookaside Buffer) je rychlá asociativní paměť, která je používána k uložení naposledy přepočítaných adres. Používá se jako vyrovnávací paměť tak, aby nemusel při každém přístupu do paměti probíhat výpočet umístění stránky. Asociativní paměť TLB prohledává všechny své záznamy zároveň. **Bez TLB by byl překlad adresy natolik pomalý, že by používání virtuální paměti ani nemělo smysl.**

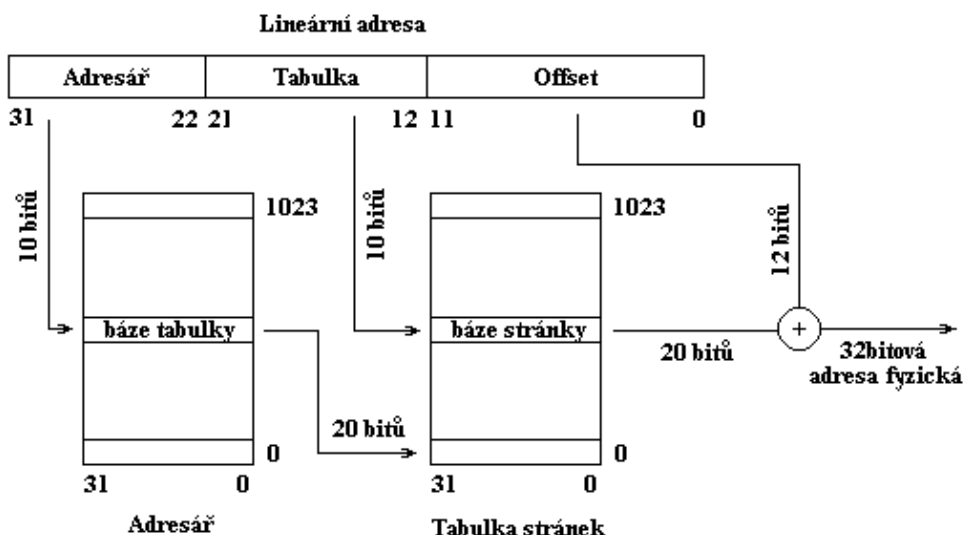
Z hlediska virtuální paměti je podstatné tyto řídicí registry:

- **CR0** – v tomto registru se zapíná stránkování nastavením bitu 31
- **CR2** – v tomto registru je v případě výpadku stránky její lineární adresa
- **CR3** – obsahuje fyzickou adresu adresáře stránek aktuálně běžící úlohy

## 2.7 Přepočet lineární adresy na fyzickou

Lineární adresa se na fyzickou se přepočítává při každém přístupu do paměti. To vyžaduje, aby přepočet netrval dlouho a proto taky adresu přepočítává procesor (hardware).

Zde se používá dvouúrovňový překlad, což má výhodu v tom, že **adresář stránek není tak veliký** (má velikost jako jedna stránka-1024 položek x 4bajty), a nevýhodu, že je nutné na jeden překlad 2x přistupovat do paměti.



Horních 10 bitů lineární adresy se použije jako ukazatel do adresáře stránek. Počáteční adresa adresáře stránek je uložena v registru CR3 (musí být vždy dělitelná 4k). Adresář stránek je struktura 1024 čtyřbajtových položek. Horních 20 bitů je ukazatel do tabulky stránek, spodních 12 pak atributy. Důležitý je především bit 0, který určuje platnost položky. Je-li nenastaven pak požadovaná stránka s tabulkou stránek není v paměti. Vznikne přerušení výpadek stránky a rutina obsluhy se postará o natažení stránky do paměti. Tabulka stránek má 1024 čtyřbajtových položek. Horních 20 bitů je finální adresa a spodních 12 atributy. Bit 0 opět určuje platnost položky. Neplatnost znamená, že stránka není v paměti a horních

31 bitů položky určuje místo uložení stránky na vnějším nosiči (většinou). Přepočtení adresy vyžaduje 2x číst z paměti – zrychluje TLB

V příloze 1 jsou vysvětleny příznaky položky adresáře stránek a položky tabulky stránek.

## 3 Operační systém Windows XP a virtuální paměť

O virtuální paměť se stará manažer virtuální paměti- *Virtual Memory Manager*

Je to jedna z komponent exekutivy pracující v režimu jádra. Je to oddělený proces, jehož účelem je správa fyzické a virtuální paměti. Kód této komponenty je umístěn v části paměti, která není nikdy vytěsněna na disk.

### 3.1 Velikost virtuální paměti

4GB v 32-bitového režimu

$10 + 10 + 12 = 4\text{GB}$

dvouúrovňový překlad

8TB (8192GB) v 64-bitovém režimu

$10 + 10 + 10 + 13 = 43$ , což poskytuje virtuální adresový prostor 8TB

velikost stránky (4kB, nebo 8kB).

tříúrovňový překlad

### 3.2 Hyperprostor

Je to 4MB prostor, obsahující adresář stránek a 1023 tabulek stránek (nultá se nepoužívá, slouží pro odhalení chyb). Každý proces má vlastní adresář stránek a z toho vyplývá, že při přepnutí kontextu se obsah změní.

### 3.3 Struktura virtuální paměti

Dle architektury IA32 má proces přidělený 4GB virtuální paměti:

- Horní 2GB jsou rezervovány pro použití systémem. Tento prostor je pro všechny procesy stejný. Nevytěšňuje se na vnější médium.
- Spodní 2GB je privátní adresový prostor pro daný proces. Obsahuje kód a data procesu.

### 3.4 Databáze rámců

Databáze rámců je **zřetězený seznam** položek tabulky stránek tříděný dle stavu stránky. Používá se pro udržování přehledu o stavu všech rámců. Obsahuje položku pro každý rámec v systému.

#### 3.4.1 Rámec může nabývat následujících stavů:

Valid (platná)	Stránka je používána aktivním procesem. V položce tabulky stránek je nastavena jako platná.
Modified (upravena)	Do této stránky bylo zapsáno, ale nebylo zapsáno na disk. V tabulce stránek je označena jako neplatná a v přechodném stavu
Standby (v pohotovosti)	Stránka byla odebrána ze seznamu pracovních stránek procesu. V tabulce stránek je označena jako neplatná a v přechodném stavu.
Free (volná)	Není spojena s žádnou položkou v tabulce stránek a je předpřipravena k použití. Předtím musí být vynulována, pokud to není stránka určená pouze pro čtení
Zeroed (vynulována)	Volná stránka, která byla již vynulována a je IHNED připravena k přidělení.
Bad (špatná)	Stránka, která vygenerovala nějakou hardwarovou chybu a nemůže být použita.

Z výše uvedeného vyplývá, že stránky se nachází v přechodném stavu, když jsou na *Modified* nebo *Standby* seznamu. Tyto stránky jsou ještě umístěny v paměti, ale jsou označeny jako neplatné. Můžou nastat dva následující stavy:

- Pokud k nějaké stránce na tomto seznamu bude proces přistupovat, bude vygenerováno přerušení výpadku stránky, stránka se pouze označí jako platná a proces ji může použít.
- Naopak, pokud k určité stránce proces dlouho nepřistupuje, OS se snaží označit daný rámec jako *Free*. Než to udělá, stránky se stavem *Modified* je nutné zapsat na disk, protože jejich obsah byl upraven. Stránky ze seznamu *Standby* je možné označit rovnou za *Free*.

### 3.5 WIN XP postup přidělování paměti

Samotný proces přidělování paměti je realizován ve dvou krocích. V prvním kroku se požadované množství paměti ve virtuální paměti rezervuje. Ve druhém kroku se přidělí fyzická paměť (Commit, mapování virtuální na fyzickou).

Když proces žádá o přidělení paměti, je mu paměť nejprve rezervována. Rezervace spočívá v tom, že se modifikují adresář stránek a tabulka stránek, ale paměť se nepřidělí (tedy v případě žádného volného místa není nutné jej uvolňovat). Když chce proces s danou stránkou pracovat, není stránka v paměti připravena a je vygenerováno přerušení výpadku stránky. V této chvíli se paměť teprve přidělí (tomuto principu se říká stránkování na žádost).

Z tohoto dvou-úrovňového přidělování paměti plyne výhoda- pokud by proces žádal o přidělení velkého prostoru, které by nevyužil, uvolňování paměti pro vyhovění požadavku by zabralo mnoho zbytečného času.

## 4 Zdroje:

Kniha MS Windows Internals:

[http://book.itzero.com/read/microsoft/0507/Microsoft.Press.Microsoft.Windows.Internals.Fourth.Edition.Dec.2004.internal.eBook-DDU\\_.html/](http://book.itzero.com/read/microsoft/0507/Microsoft.Press.Microsoft.Windows.Internals.Fourth.Edition.Dec.2004.internal.eBook-DDU_.html/)

<http://msdn2.microsoft.com/en-us/library/ms810616.aspx>

<http://www.cs.umd.edu/class/sum2003/cmcs311/Notes/Memory/virtual.html>

<http://www.windowsfordevices.com/articles/AT7967807599.html>

<http://lwn.net/Articles/253361/>

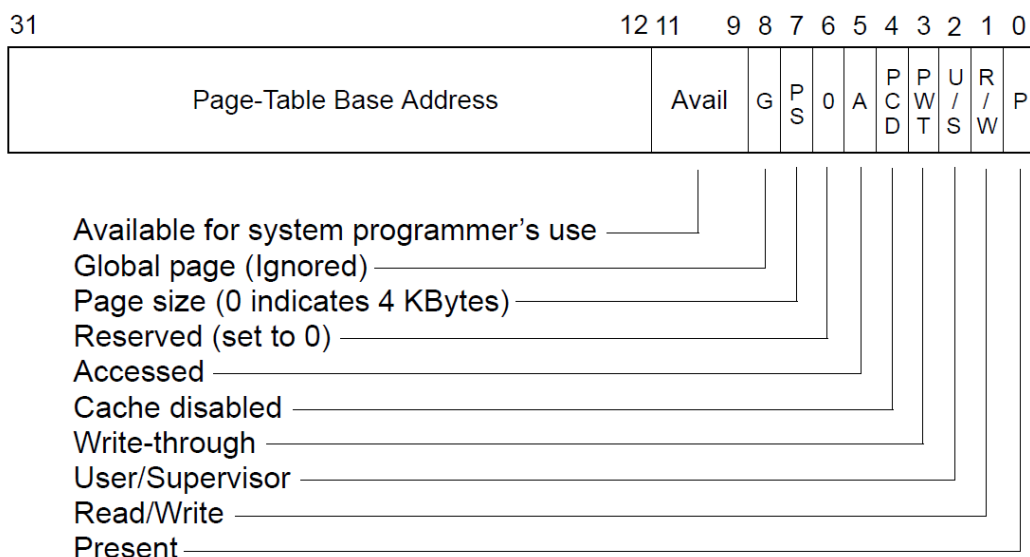
[http://www.student.cvut.cz/cwut/index.php/Virtu%C3%A1ln%C3%AD\\_pam%C4%9B%C5%A5\\_a\\_dynamick%C3%BD\\_p%C5%99eklad\\_adres](http://www.student.cvut.cz/cwut/index.php/Virtu%C3%A1ln%C3%AD_pam%C4%9B%C5%A5_a_dynamick%C3%BD_p%C5%99eklad_adres)

<http://www.cs.gmu.edu/cne/itcore/virtualmemory/vmhistory.html>

## 5 Přílohy (nemusíme umět, jen pro doplnění)

### 5.1 Položka adresáře stránek

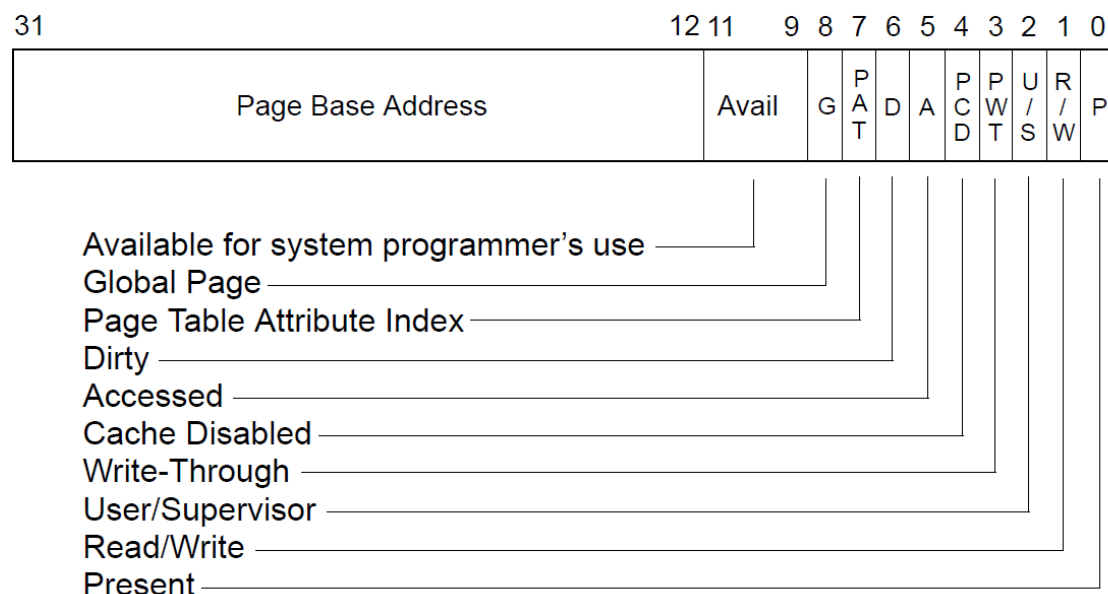
**Page-Directory Entry (4-KByte Page Table)**



- P** Signalizuje, zda je stránka přítomna v operační paměti
- R/W** Charakterizuje, zda je daná stránka pouze pro čtení, nebo i pro zápis. To platí i pro položku v adresáři stránek.
- U/S** Specifikuje přístupová práva- S (Supervisor), U (User)
- PWT** Nastavuje strategii cachování (Write-through nebo Write-Back) tabulky stránek, nebo stránky. Pokud je cachování vypnuto (bit CD v CR0), není na toto brán zřetel.
- PCD** Tento bit umožňuje vypnout cachování pro určitou tabulku stránek, či stránku. To se používá pro stránky namapovaných I/O portů
- A** Určuje, zda do tabulky stránek či stránky bylo přístupováno. Obvykle správa paměti tento bit nastaví na 0, když je stránka nahrána do fyzické operační paměti. Procesor nastaví bit do 1, když je na stránku poprvé přístupováno. Nulování bitu provádí pouze software.
- PS** Udává velikost stránek, jestli 4kB nebo 4MB.

## 5.2 Položka tabulky stránek

### Page-Table Entry (4-KByte Page)



Zde jsou vysvětleny příznaky, které nejsou vysvětleny u položky adresáře stránek

- D Tento příznak je vynulován (a to softwarově) když je stránka nahrána do paměti. Procesor (tedy hardware) jej nastaví při prvním zápisu do stránky.
- PAT Page Attribute Table...souvisí s cachováním
- G Nastavuje stránku jako globální. To umožňuje (pokud je to umožněno bitem PGE v CR4) často používaným stránkám, že nebudou vymazány z TLB, když je nově nahrán CR3 v momentě přepnutí úlohy. Tento bit nastavuje pouze software (ne sám procesor). Tento příznak má význam pouze v položkách tabulky stránek (kdy položka ukazuje na stránku), v adresáři stránek je stav tohoto bitu ignorován.

## 5.3 Položka databáze rámce

Ačkoli mají položky databáze rámce pevnou délku, význam některých jejích polí závisí na stavu stránky.

Working set index		
PTE address		
Share count		
Flags	Type	Reference count
Original PTE contents		
PFN of PTE		

**PFN for a page in a working set**

Forward link		
PTE address		
Backward link		
Flags	Type	Reference count
Original PTE contents		
PFN of PTE		

**PFN for a page on the standby or the modified list**

Forward link		
PTE address		
Color chain PFN number		
Flags	Type	Reference count
Original PTE contents		
PFN of PTE		

**PFN for a page on the zero or free list**

Event address		
PTE address		
Share count		
Flags	Type	Reference count
Original PTE contents		
PFN of PTE		

**PFN for a page with an I/O in progress**

working set- podmnnožina virtuálních stránek, které jsou v paměti

První tabulka představuje stránku, která je aktivní a je součástí pracovní sady. Pole Share Count představuje počet položek tabulky stránek, které ukazují na tuto stránku (stránky označeny jako *read-only*, *copy-on-write* nebo *shared read/write* mohou být sdíleny vícero procesy). Pro tabulky stránek- toto pole je počet platných položek v tabulce stránek. Dokud je toto pole větší než 0, stránka není určena pro odebrání z paměti.

Druhá část obrázku je pro stránku, která je součástí *standby* nebo *modified* seznamu (její stav je *standby* nebo *modified*). V tomto případě pole *Forward link* a *Backward link* spojuje prvky uvnitř seznamu. Toto spojování umožňuje, aby byly stránky snadno manipulovatelné při výpadku stránky. Když je stránka na jednom z výše uvedených seznamů v poli *Share Count* je nula (protože stránka není součástí žádné pracovní sady) a proto toto pole může mít význam *Backward link*. V poli *Reference Count* je také nula. A pokud je nenulová, je se stránkou manipulováno (týká se vstupu/výstupu- např. je zapisován na disk), toto je první odebrání ze seznamu.

Třetí obrázek je pro stránku která je součástí *free* nebo *zeroed* seznamu. Kromě toho jsou spojeny dohromady s dvěma seznamy, tato položka databáze rámců používá dodatečné pole k tomu, aby spojilo fyzické stránky by „color“ jejich umístění v cache paměti procesoru. Windows se pokouší minimalizovat zbytečnou práci s cache pamětí procesu používáním různých fyzických stránek v procesorové cache paměti. Tato optimalizace umožňuje vyhnout se používání stejné položky chacha pro dvě různé stránky kdykoliv je to možné. Pro systémy s přímo mapovanou cache pamětí, může mít toto za následek významného zvýšení výkonu.

Stav na čtvrtém obrázku je pro stránky, na kterých jsou prováděny operace vstupu/výstupu (například stránka je čtena). Pokud probíhá vstup/výstup, první pole ukazuje na objekt, který tento přesun provádí.

### 5.3.1 Význam polí

**PTE address-** virtuální adresa položky tabulky stránek, která ukazuje do této stránky

**REFERENCE COUNT**- počet odkazů do této stránky

**Type**- určuje typ stránky (valid, transition, Standby, modified, free, zeroed)

*working set index* je indexem do seznamu pracovní sady (nebo systém nebo sekce seznamu pracovní sady, nebo nula když není v žádné pracovní sadě) kde virtuální adresa mapuje fyzickou stránku. Jestliže stránka je privátní stránka (využívá ji pouze jeden proces), pole *working set index* odkazuje přímo na položku, protože stránka je mapována pouze jednou virtuální adresou.

### 5.3.2 Flags (příznaky)

Modified state	udává, jestli byla stránka modifikována. (pokud byla stránka modifikována, je nutné jí před odstraněním z paměti uložit na disk)
Prototype PTE	udává, jestli je na položku tabulky stránek odkazováno položkou databáze rámců (například jestli je stránka sdílitelná)
Parity error	udává, že fyzická stránka obsahuje chybu parity
Read in progress	udává, že tato stránka je právě čtena
Write in progress	udává, že do stránky je zapisováno
Start of nonpaged pool	pro stránky umístěné v oblasti, která se nevytěšňuje na disk, udává že toto je první položka nevytěšňované části.
End of nonpaged pool	také se týká oblasti, která se nevytěšňuje na disk. Udává, že tato stránka je poslední položkou nevytěšňované části.
In-page error	udává, že nastala chyba vstupu/výstupu během operace. V tomto případě první položka databáze rámce obsahuje kód chyby.