

Operační systémy 2

Implementace virtuální paměti

Petr Krajča

Katedra informatiky
Univerzita Palackého v Olomouci

11. říjen, 2011

Úvodem

- **x86/i386**

- budeme **uvažovat jen 32-bitové procesory** z rodiny i386 (kompatibilní a novější)
- podpora starších módů pro kompatibilitu (budeme zanedbávat)

- **AMD64**

- označení rodiny procesorů (Intel EM64-T)
- rozšíření i386 **na 64 bitů** (long mode)
- **registry rozšířeny na 64 bitů** (např. **EAX \Rightarrow RAX**); větší počet
- možnost **adresovat rozsáhlejší paměť**

- procesory se vyvíjely několik desetiletí (procesor 80386 uveden 1985)
 \Rightarrow relikty minulosti

- **kombinují segmentaci i stránkování**

Úrovně oprávnění

- čtyři úrovně oprávnění (rings)
- nelze používat některé instrukce
- zabraňují aplikacím poškodit systém
- ring 0 – nejvyšší oprávnění (jádro)
- ring 3 – nejmenší oprávnění (aplikace)
- nejčastěji se používá kombinace ring 0 + ring 3
- ostatní původně určeny pro ovladače, příp. knihovny
- mikrokernél; virtualizace (XEN)
- přechod mezi úrovněmi přes brány (gates)

Typy adres na procesorech i386

- **logická adresa** – vidí ji aplikace; 48 bitů (16 selektor segmentu; 32 offset); segment je často implicitní
- **lineární (virtuální) adresa** – v adresním prostoru procesu (32 bitů)
- **fyzická adresa** – „číslo bytu“ přímo v primární paměti (32 bitů, s PAE 36); sdílená dalšími HW zařízeními; (nevyužité adresy mohou mít další použití SWAP)

PAE: Physical Address Extension

- umožňuje rozšířit využitelnou paměť RAM z 4GB na 64GB (Pentium Pro a novější)
- přesměruje část adresního prostoru do jiné části fyzické paměti
- změna formátu segmentových deskriptorů
- stránky 4KB/2MB
- změna na úrovni OS (případně ovladačů);
- bez úprav jednotlivé procesy stále omezeny na 4GB (AWE)
- mimochodem přidává podporu pro NX bit

I386: Segmentace I.

- paměť je možné rozdělit na segmenty (kód, data, zásobník, etc.)
- pro každý segment lze nastavit oprávnění (ochrana paměti)
- segmenty jsou popsány pomocí deskriptorů 8 B záznam
 - báze
 - limit (velikost segmentu)
 - požadovaná úroveň oprávnění (ring 0-4)
- deskriptory segmentů uloženy v
 - Global Descriptor Table (GDT) – sdílená všemy procesy
 - Local Descriptor Table (LDT) – každý proces má vlastní
- každá může mít až 8192 záznamů
- přístupné přes registry GDTR, LDTR
- první záznam v GDT „null“ deskriptor
- granularita (stránky vs. byty)

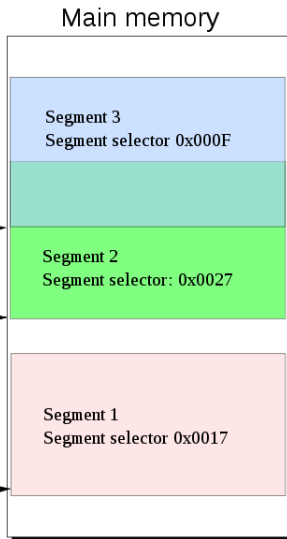
I386: Segmentace II.

Local Descriptor Table (LDT)

5			
4	0x21430	0xC000	•
3			
2	0x0CEF0	0xA300	•
1	0x28C00	0xFC00	•
0			

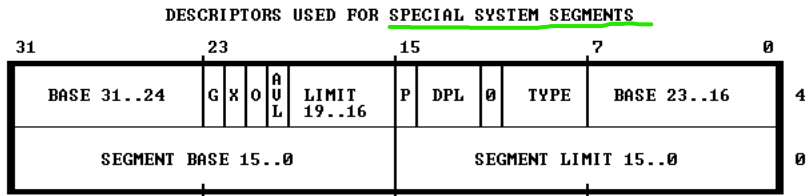
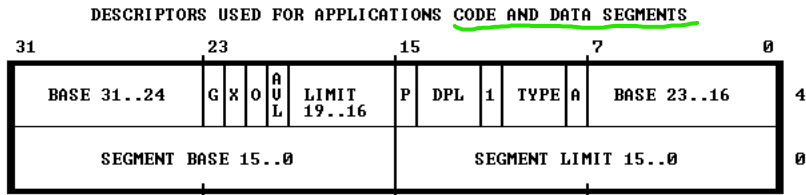
Linear base
address
(BASE)

Segment size
(LIMIT)



I386: Segmentace III.

Figure 5-3. General Segment-Descriptor Format



- ```

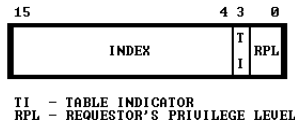
A - ACCESSED
AUL - AVAILABLE FOR USE BY SYSTEMS PROGRAMMERS
DPL - DESCRIPTOR PRIVILEGE LEVEL
G - GRANULARITY
P - SEGMENT PRESENT

```

## I386: Segmentace IV.

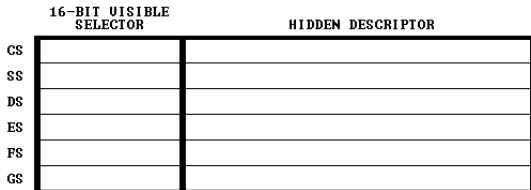
- do segmentových registrů (CS, SS, DS) se ukládá selector (16 bitů) – ukazatel do GDT nebo LDT

Figure 5-6. Format of a Selector



- při načtení se do seg. registru načte i deskriptor (ale nejde k němu explicitně přistupovat)

Figure 5-7. Segment Registers

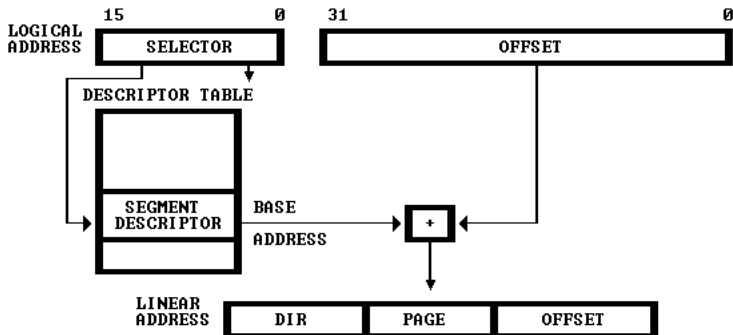




## I386: Překlad adres I. (segmentace)

- logická adresa  $\Rightarrow$  linární adresa (segmentace)
- ověří se oprávnění a limit (přístup za hranici segmentu)  $\Rightarrow$  neoprávněný přístup
- báze segmentu je sečtena s offsetem  $\Rightarrow$  lineární

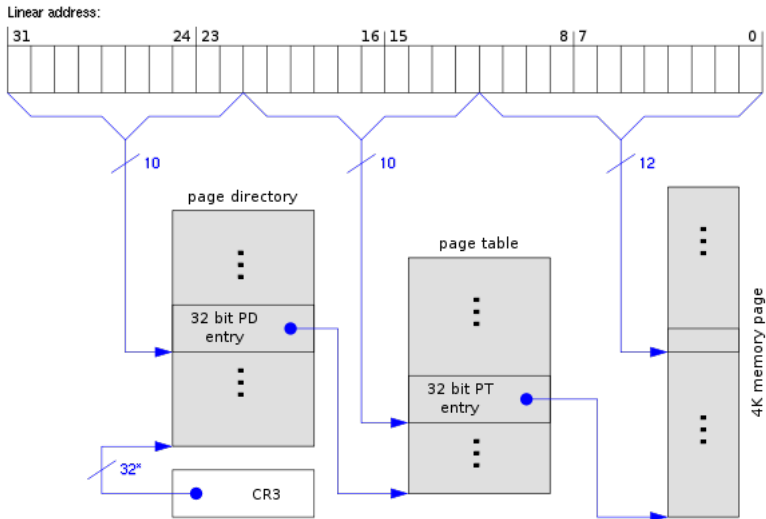
Figure 5-2. Segment Translation



## I386: Překlad adres II. (stránkování)

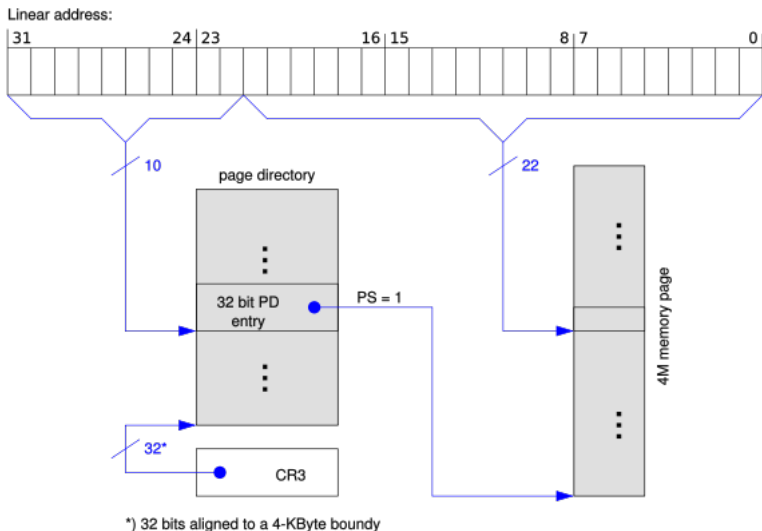
- lineární adresa  $\Rightarrow$  fyzická adresa
- standardní stránka/rámec: 4 KB
- hierarchická struktura
  - adresář stránkovacích tabulek (Page Directory)
  - adresář stránek (Page Tables)
  - offset
- adresáře mají velikost jedné stránky, každá položka 4B  $\Rightarrow$  1024 záznamů
- lineární adresa rozdělena na 10 + 10 + 12 bitů (PDI + PTI + offset)
- maximální kapacita 4 GB
- adresa PDT v CR3 (zarovnané na celé stránky)
- nastavením příznaku v PDT (pro adresu rámce se používá jen 20b), lze obejít přepočít přes PT a používat stránky velikosti 4MB (zbytek adresy je offset)
- velikost stránek lze kombinovat

## I386: Překlad adres III. (stránkování – 4 KB stránka)



\*) 32 bits aligned to a 4-KByte boundary

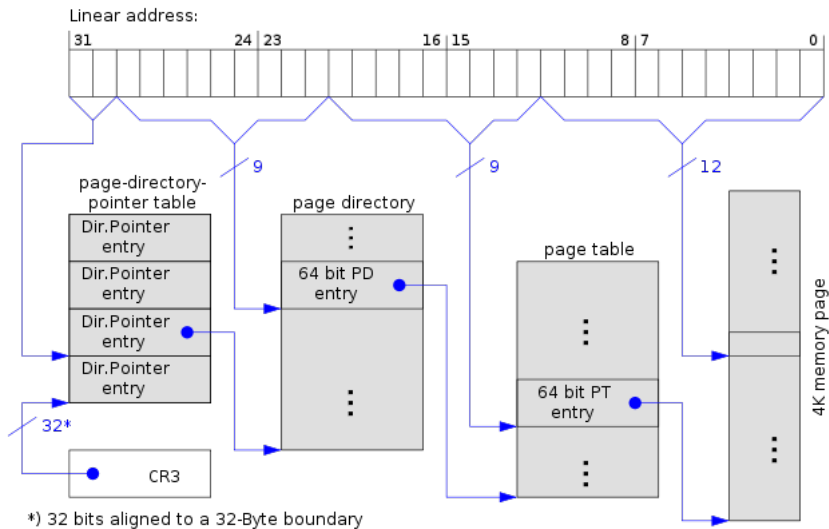
## I386: Překlad adres IV. (stránkování – 4 MB stránka)



## I386: Překlad adres IV. (PAE)

- od Pentium Pro
- každá tabulka 4KB, ale velikost záznamu 8B  $\Rightarrow$  512 záznamů
- stránkování trojúrovňové
- adresa rozdělena na  $2 + 9 + 9 + 12$  bitů
  - 2b – ukazatel na adresář tabulek stránek (Page Directory Pointer Index)
  - 9b – ukazatel v adresáři tabulek stránek
  - 9b – ukazatel v tabulce stránek
  - 12b – offset
- velké stránky 2MB  $\Rightarrow$  offset 21 bitů
- potenciální rozšíření

## I386: Překlad adres VI. (PAE – 4 KB stránka)



# AMD64: Typy adres

- segmenty existují, ale nepoužívají se k adresaci, pouze ke kontrole oprávnění
- deskriptor kódového segmentu se používá k přechodu mezi 32- a 64bitovým režimem
- současné procesory AMD64:
  - 40bitové fyzické adresy (max. 52; způsob stránkování)
  - 48bitové logické adresy (max. 64; velikost registrů)
- možnost rozšíření  $\implies$  kanonické adresy
- nejvyšší platný bit je okopírován do vyšších bitů
- dělí paměť na tři bloky (viz Keprt str. 119)

# AMD64: Stránkování

- používá se režim PAE
- zavedena čtyřúrovňová hierarchie
- záznamy v tabulkách stránek mají 8 B
- při 4KB stránkách  $\Rightarrow 4 \times 9 + 12 = 48$  adresovatelných bitů
- pro 2MB stránky vynechaná jednoúroveň, možnost použít 4 rezervované bity  $\Rightarrow 52$  bitů
- nepoužité bity: nejvyšší – NX bit, ostatní k dispozici OS



# Ochrana paměti I. (segmenty)

- ochrana paměti na úrovni segmentů je zaplá a nejde vypnout (ale jde nastavit, aby nebyla účinná)
- prováděné kontroly
  - kontrola typu segmentu (některé segmenty nebo segmentové registry mohou být použité jenom určitým způsobem)
  - kontrola velikosti segmentu (limitu), i.e., jestli program našahá za hranice segmentu
  - kontrola oprávnění
  - omezení adresovatelné domény (omezení přístupu jen k žádoucím segmentům)
  - omezení vstupních bodů procedur (brány)
  - omezení instrukční sady
- AMD64 v long mode neprovádí některé kontroly (báze a limit jsou ignorovány)

## Ochrana paměti II. (stránkování)

- stránkování funguje souběžně se segmentací
- bit pro systémové stránky (zákaz přístupu z ring 3)  $\Rightarrow$  volání funkce OS (přes bránu)
- bit pro zákaz zápisu
- AMD64 + PAE mají NX bit (zákaz spouštění)  $\Rightarrow$  viry
- možnost nastavit nastavit bity na jednotlivých stránkách i adresářích  $\Rightarrow$  efektivnější

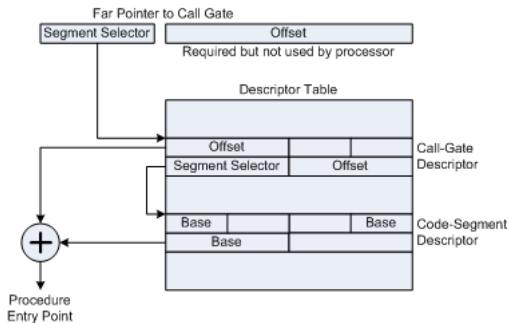
# Úrovně oprávnění

- **DPL** (**Descriptor Privilege Level**) – úroveň oprávnění **daného descriptoru**
- **CPL** (**Current Privilege Level**) – **aktuální úroveň oprávnění**; odpovídá DPL v CS
- **RPL** (**Requested Privilege Level**) – úroveň **oprávnění daného selektoru** (požadovaná úroveň)
- pokud  $\max(CPL, RPL) \leq DPL$  je oprávnění **v pořádku**
- jinak procesor vyvolá vyjímku
- nižší hodnota  $\implies$  vyšší oprávnění)

# Ochrana u CS

- přechod mezi kódovými segmenty možný přímo, přes segment TSS, přes bránu
- různé druhy bran: call-, trap-, interrupt-, task- gate (popsané v GDT)
- deskriptor volací brány obsahuje selektor a offset volaného kódu
- ověřuje se DPL brány i DPL kódového segmentu
- podle nastavení „bitu konformity“ se případně použije zásobník pro každou úroveň oprávnění (adresy uloženy v TSS)
- nekomformní  $\implies$  změna oprávnění
- možné použít i k přepnutí režimu procesoru (např. 16-, 32- bitů)
- ústup od používání  $\implies$  SYSENTER + SYSEXIT

# Brány pro CS



# Implementace v NT na i386

- paměť 2GB:2GB (systém + proces); lze změnit na 1:3
- je možné používat Address Windowing Extension (AWE) – zpřístupní víc než 2GB
- rozdělení stránek na volné, rezervované, komitované  $\implies$  demand paging (nulování stránek)
- množina procovních rámců (50–345); balance manager
- stránky se odswapovávají podle přístupového bitu
- načítá několik stránek současně (clustering)
- logical prefetcher umožňuje urychlit start systému (sleduje přístup na FS a vytváří log)
- používá se segmentace i stránkování
- procesy jsou navzájem oddělené (LDT)

# Implementace v Linuxu na i386

- paměť 1GB:3GB (systém + proces)
- paměť rozdělena na zóny (procesy, DMA, highmem)
- podpora NUMA
- stránky v několika frontách (active, inactive)
- rozdělení stránek na volné, rezervované, komitované  $\Rightarrow$  demand paging
- OOM killer