Algorytm tłumaczenia adresu (Intel)

20 grudnia 2000

Wstęp

Algorytm Tłumacznia adresu pamieciowego wiąże się ściśle z **abstrakcyjnym modelem PAMIĘCI WIRTUALNEJ**, wykorzystywanym przez Linux'a.

Ten model jest adaptacją Linux'owego **modelu pamieci niezależnego od architektury** do danej maszyny (w naszym przypadku - x86):

Model pamieci niezależny od architektury przewiduje **Segmentację z potrójnym stronicowaniem**. Oznacza to, że pamięć jest podzielona na segmenty i znalezienie właściwego segmentu jest pierwszym krokiem w tłumaczeniu adresu. Potrójne stronicowanie oznacza, że aby określić adres fizyczny musimy odczytać odpowiednie pozycje w dwóch katalogach i jednej tablicy stron.

Algorytm ma dwie zasadnicze fazy:

1. **Adres wirtualny(logiczny) -> Adres liniowy (Segmentacja)**
2. **Adres liniowy -> Adres fizyczny (Stronicowanie)**

Adaptacja do architertury Intela polega głównie na redukcji stopnia stronicowania z trójpoziomowego do dwupoziomowego. Tłumaczeniem adresów zajmuje sie Memory Management Unit (MMU) czyli sprzetowa jednostka zarządzania pamiecią.

1 Segmentacja

Pamięć wirtualna jest podzielona na segmenty. Jej wielkość jest ograniczona przez długość adresu liniowego do 4Gb. Wielkość wirtualnego segmentu jest ograniczona do 3 Gb (arbitralnie). Segment jest znacznie wiekszy niż cala pamięć fizyczna. Po to właśnie jest pamięCC wirtualna, aby dawaCC procesom iluzje dostępu do duZZej, jednorodnej i szybkiej pamięci.

Adres wirtualny , który dochodzi do MMU, ma 48 bitów. Oto jego format:

47-32	31-0
selektor segmentu	offset w segmencie

Linux przydziela wszystkim procesom obszary we wspólnych segmentach danych i kodu, marginalizując tym samym mechanizm segmentacji. Segment pamięci jest określony przez selektor w adresie wirtualnym. Selektor określa pozycje w tablicy z deskryptorami segmentow oraz prawa z jakimi proces korzysta z pamięci. Oto jego format:

15-3	2	1-0
indeks	TI	RPL

TI - **Table Indicator** - określa do której tablicy deskryptorów się odwołujemy

0 - globalna tablica deskryptorów

1 - lokalna tablica deskryptorów

RPL - określa poziom praw. W Linux'e wyróżniamy tylko dwa:

0 - poziom jądra

1 - poziom użytkownika

W Linux'e zawszaw jest: TI =0

Każdy segment w systemie opisany jest przez ośmiobajtowy deskryptor segmentu, który zawiera niezbędne informacje (adres fizyczny, wielkość, typ oraz prawa). Po znalezieniu deskryptora odczytujemy adres fizyczny danego segmentu i dodajemy go do przesunięcia w segmencie. Wynikiem jest **adres liniowy**.

2 Stronicowanie

Pamieć wirtualna podzielona jest na kawałki nazywane **stronami.** Pamieć fizyczna jest podzielona na kawałki (tej samej wielkości co strony) nazywane **ramkami**. Adres liniowy sklada sie z adresu strony i przesuniecia wewnątrz niej. Tłumaczenie adresu polega na tłumaczeniu adresu strony na adres ramki a przesunięcie wewnątrz strony/ramki pozostaje takie samo. Położenie strony w pamięci fizycznej jest określone przez indeksy w katalogach/tablicach stron. W abstrakcyjnym modelu pamięci wirtualnej są to : **Katalog stron(pgd)**, **pośredni katlog stron (pmd)** i **tablica stron(pte).** Dla procesorów x86 pośredni katalog stron jest pomijany co oznacza, że jego rozmiar zawsze jest równy 1 (pozycja w glównym katalogu jest traktowana jako pośredni katalog stron). Format adresu liniowego jest następujący:

31-22	21-12	11-0
katalog	tablica	offset

Sam **Algorytm** tłumaczenia adresu liniowego na Intelu dziala nastepująco:

- Z katalogu stron odczytujemy adres tablicy stron

- Z tablicy stron odczytujemy adres fizyczny szukanej strony i poniewaz offset na stronie (ostatnie 12 bitów) nie zmienia sie mamy już gotowy adres fizyczny.

Przy odczytywaniu adresu z katalogu (tabilcy) stron, odczytujemy także informacje o tablicy stron (stronie), które są zapisane na najmłodszych dwunastu bitach każdej pozycji w katalogu lub w tablicy stron. Oto jej format:

31-12	11-9	8	7	6	5	4	3	2	1	0
ADRES	OS	O	O	D	A	O	O	U/S	R/W	P

D - strona zmodyfikowana (dirty page)
R/W - 0 tylko do czytania przez użytkownika
U/S - 1 oznacza strone użytkownika (nie systemowa)
P - czy jest w pamieci (1-tak, 0-nie i reszta bitów to położenie w swap'ie)
A - 1 oznacza, że strona była czytana/zapisana (accessed)
OS - do wykorzystania przez system (LRU, etc.)

Wynikiem algorytmu jest adres fizyczny danego bajtu oraz ewentualne ściągnięcie strony na której się on znajduje.