

ZARZĄDZANIE PAMIĘCIĄ OPERACYJNĄ

CELE ZARZĄDZANIA PAMIĘCIĄ OPERACYJNĄ

- ✘ przemieszczanie procesów
- ✘ ochrona zawartości pamięci
- ✘ dostęp do obszarów dzielonych
- ✘ organizacja logiczna
- ✘ organizacja fizyczna pamięci

(C) IISI d.KIK PCz 2013

Systemy operacyjne

1

(C) IISI d.KIK PCz 2013

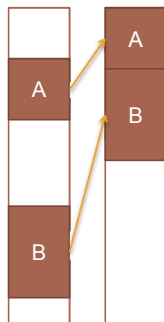
Systemy operacyjne

2

CELE ZARZĄDZANIA PAMIĘCIĄ OPERACYJNĄ

Przemieszczanie procesów

- ✘ procesy mogą być przemieszczane w pamięci w czasie działania, należy zapewnić właściwą zamianę adresów logicznych na fizyczne



(C) IISI d.KIK PCz 2013

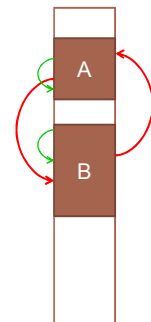
Systemy operacyjne

3

CELE ZARZĄDZANIA PAMIĘCIĄ OPERACYJNĄ

Ochrona zawartości pamięci

- ✘ sprawdzanie, czy generowane przez proces adresy są mu przydzielone



(C) IISI d.KIK PCz 2013

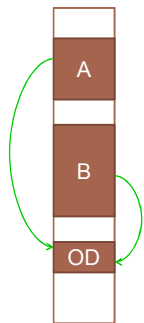
Systemy operacyjne

4

CELE ZARZĄDZANIA PAMIĘCIĄ OPERACYJNĄ

Dostęp do obszarów dzielonych

- ✘ część danych lub programów może być wykorzystywana przez kilka procesów, należy zapewnić kontrolę nad dostępem - ochronę przed jednoczesnym dostępem



(C) IISI d.KIK PCz 2013

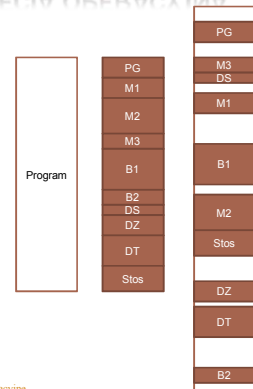
Systemy operacyjne

5

CELE ZARZĄDZANIA PAMIĘCIĄ OPERACYJNĄ

Organizacja logiczna pamięci

- ✘ segmenty programu i danych (kodowane niezależnie, różne poziomy ochrony, dzielenie segmentów między procesy)



(C) IISI d.KIK PCz 2013

Systemy operacyjne

6

CELE ZARZADZANIA PAMIĘCIĄ OPERACYJNA

Organizacja fizyczna pamięci

- ✘ pamięć główna
- ✘ pamięć pomocnicza

TWORZENIE ADRESÓW

- ✘ kompilacja
- ✘ ładowanie
- ✘ wykonanie

(C) IISI d.KIK PCz 2013

Systemy operacyjne

7

(C) IISI d.KIK PCz 2013

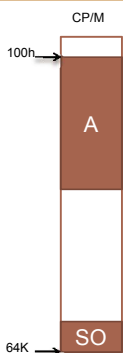
Systemy operacyjne

8

TWORZENIE ADRESÓW

Kompilacja

- ✘ adres umieszczenia kodu jest znany w trakcie kompilacji



(C) IISI d.KIK PCz 2013

Systemy operacyjne

9

TWORZENIE ADRESÓW

Ładowanie

- ✘ adres jest znany w trakcie ładowania



(C) IISI d.KIK PCz 2013

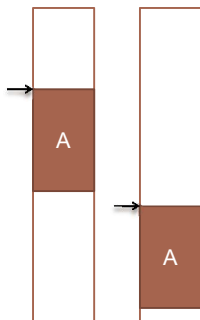
Systemy operacyjne

10

TWORZENIE ADRESÓW

Wykonanie

- ✘ adres może się zmieniać w trakcie wykonywania procesu



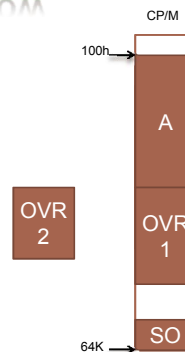
(C) IISI d.KIK PCz 2013

Systemy operacyjne

11

TWORZENIE ADRESÓW

Nakładki



(C) IISI d.KIK PCz 2013

Systemy operacyjne

12

TWORZENIE ADRESÓW

Biblioteki ładowane dynamicznie



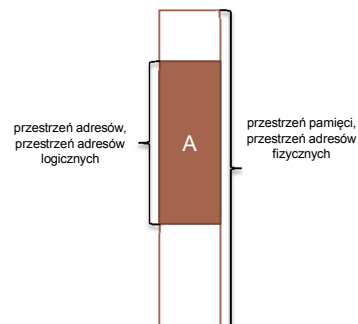
(C) IISI d.KIK PCz 2013

Systemy operacyjne

13

PAMIĘĆ WIRTUALNA

zapewnia odwzorowanie przestrzeni adresów na przestrzeń pamięci



(C) IISI d.KIK PCz 2013

Systemy operacyjne

14

IMPLEMENTACJA PAMIĘCI WIRTUALNEJ

Pamięć wirtualna może być implementowana w formie:

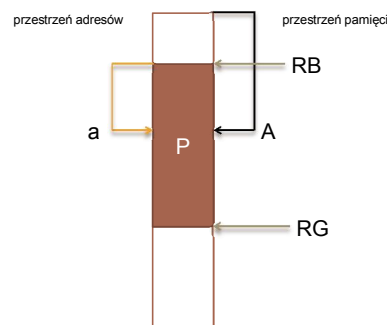
- ✘ Rejestru bazowego i granicznego
- ✘ Stronicowania
- ✘ Segmentacji
- ✘ Segmentacji ze stronicowaniem

(C) IISI d.KIK PCz 2013

Systemy operacyjne

15

REJESTR BAZOWY I GRANICZNY



(C) IISI d.KIK PCz 2013

Systemy operacyjne

16

REJESTR BAZOWY I GRANICZNY

- ✘ $A = f(a) = RB + a$
- ✘ Gdy proces ładowany jest do pamięci, wtedy najmniejszy adres zajęty przez proces jest umieszczany w rejestrze bazowym i wszystkie adresy w programie traktowane są jako adresy **względne** w odniesieniu do adresu bazowego.
- ✘ Odwzorowanie polega zatem na dodawaniu adresu użytego przez proces do adresu bazowego.
- ✘ Przemieszczanie można uzyskać przez przepisanie bloku pamięci i wpisanie nowego adresu do rejestru bazowego.

(C) IISI d.KIK PCz 2013

Systemy operacyjne

17

REJESTR BAZOWY I GRANICZNY

Ochronę pamięci zapewnia rejestr graniczny.

RG – koniec przydzielonego bloku

RG – długość przydzielonego bloku

$$A = f(a) = RB + a$$

$$A > RG$$

$$A = f(a) = RB + a$$

$$a > RG$$

(C) IISI d.KIK PCz 2013

Systemy operacyjne

18

FRAGMENTACJA

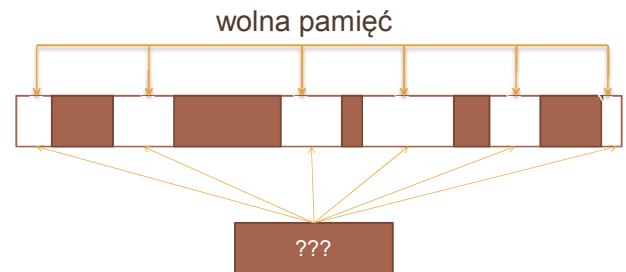
- ✘ W miarę ładowania i usuwania procesów z pamięci operacyjnej pamięć ulega „rozdrobnieniu”. Zjawisko to nazywane jest to **fragmentacją zewnętrzną**.
- ✘ W pamięci powstają „dziury”, po zwolnionych blokach, które nie zostały w pełni wypełnione przez bloki innych procesów.

(C) IISI d.KIK PCz 2013

Systemy operacyjne

19

FRAGMENTACJA ZEWNĘTRZNA



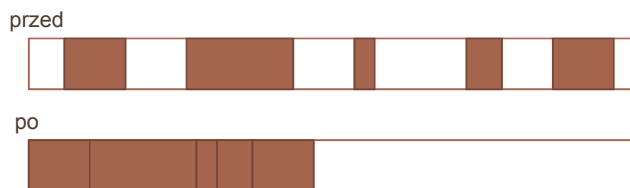
(C) IISI d.KIK PCz 2013

Systemy operacyjne

20

FRAGMENTACJA ZEWNĘTRZNA - UPAKOWANIE

- ✘ Upakowanie pamięci jest stosowane, gdy pojawia znaczna fragmentacja zewnętrzna pamięci.
- ✘ Bloki pamięci przenoszone są tak, aby pozostało w niej najmniej „dziur” lub nie było ich wcale.



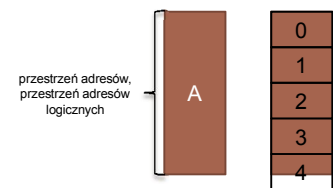
(C) IISI d.KIK PCz 2013

Systemy operacyjne

21

STRONICOWANIE

Przestrzeń adresów jest dzielona na równe strony (2^n ; najczęściej 4 KB)



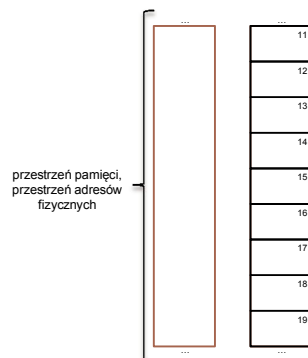
(C) IISI d.KIK PCz 2013

Systemy operacyjne

22

STRONICOWANIE

Przestrzeń pamięci dzielona jest na **ramki stron**.



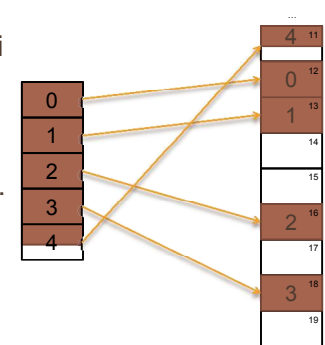
(C) IISI d.KIK PCz 2013

Systemy operacyjne

23

STRONICOWANIE

Strony z przestrzeni adresów umieszczane są w ramkach stron przestrzeni pamięci.



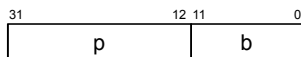
(C) IISI d.KIK PCz 2013

Systemy operacyjne

24

STRONICOWANIE

Dla adresu 32 bitowego i 4 KB strony, adres w przestrzeni adresów (logiczny) można przedstawić:



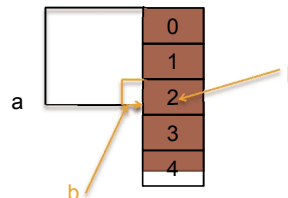
gdzie:

p – numer strony

b – przesunięcie na stronie

STRONICOWANIE

Graficznie adres można przedstawić:



STRONICOWANIE

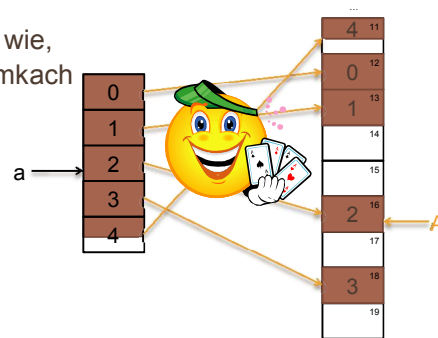
Adres w przestrzeni pamięci można wyznaczyć:

$$A = f(a) = f(p, b) = p' + b$$

p' – numer ramki zawierającej stronę p

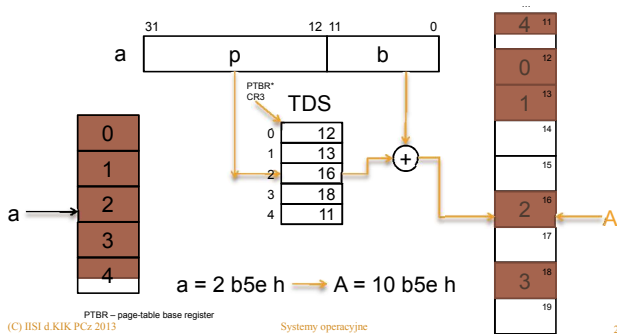
STRONICOWanie

Pytanie: skąd system wie, w których ramkach są strony?

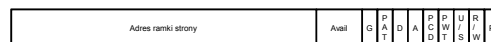


STRONICOWANIE

Tablica deskryptorów stron



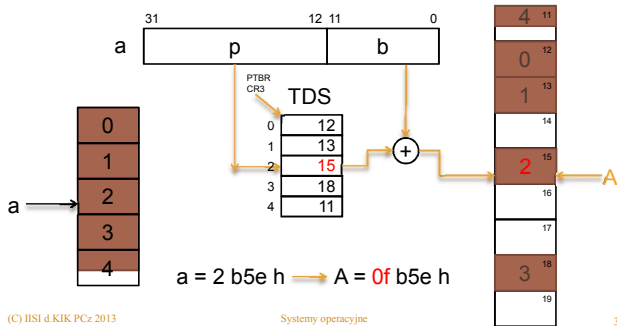
DESKRYPTOR STRONY



- * Available for system programmer's use – dostępne dla programistów systemowych
- * Global Page – strona globalna – nie jest usuwana z TLB przy przełączaniu procesów
- * Page Table Attribute Index – PAT pozwala przyporządkować typ pamięci do strony
- * Dirty – strona zmieniona
- * Accessed – strona użyta
- * Cache Disabled – wyłączenie zapamiętywania strony w pamięci podręcznej
- * Write-Through – jednoczesne zapisywanie strony w pamięci podręcznej i głównej
- * User/Supervisor – poziom ochrony użytkowy/systemowy
- * Read/Write - tylko odczyt/zezwoleńie na zapis
- * Present – strona obecna

STRONICOWANIE

Przemieszczenie strony



(C) IISI d.KIK PCz 2013

Systemy operacyjne

31

STRONICOWANIE

Pamięć wirtualna widziana przez system składa się z:

- × pamięci operacyjnej
- × pamięci pomocniczej (obszaru wymiany) w postaci:
 - + pliku wymiany
 - + partycji wymiany



(C) IISI d.KIK PCz 2013

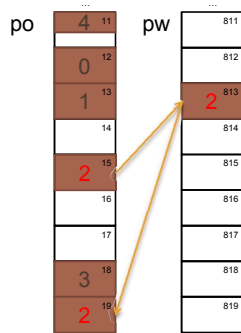
Systemy operacyjne

32

STRONICOWANIE

Korzystanie z pamięci wirtualnej:

- × odesłanie strony z pamięci operacyjnej do pomocniczej
- × i pobranie jej z powrotem z pamięci pomocniczej do operacyjnej w inne miejsce



(C) IISI d.KIK PCz 2013

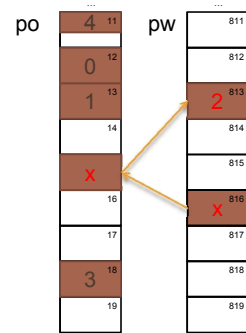
Systemy operacyjne

33

STRONICOWANIE

Korzystanie z pamięci wirtualnej - wymiana:

- × odesłanie strony z pamięci operacyjnej do pomocniczej
- × pobranie innej z pamięci pomocniczej do operacyjnej w jej miejsce



(C) IISI d.KIK PCz 2013

Systemy operacyjne

34

STRONICOWANIE

Zadania:

- × Wykonywanie operacji odwzorowywania adresów, czyli określania, do której ramki odnosi się adres generowany przez proces.
- × Przesyłanie stron z pamięci pomocniczej do pamięci głównej oraz odsyłanie nieużywanych już stron z powrotem do pamięci pomocniczej.

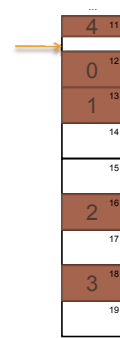
(C) IISI d.KIK PCz 2013

Systemy operacyjne

35

STRONICOWANIE

- × W przypadku stronicowania nie ma zjawiska fragmentacji zewnętrznej.
- × Występuje jednak fragmentacja wewnętrzna.
- × Średnio traci się pół ramki na jeden blok pamięci.



(C) IISI d.KIK PCz 2013

Systemy operacyjne

36

STRONICOWANIE

Efektywny czas dostępu do pamięci

$$EAT = 2 * MAT$$

- ✗ MAT – memory access time (np. 100 ns)
- ✗ EAT – effective access time (200 ns)

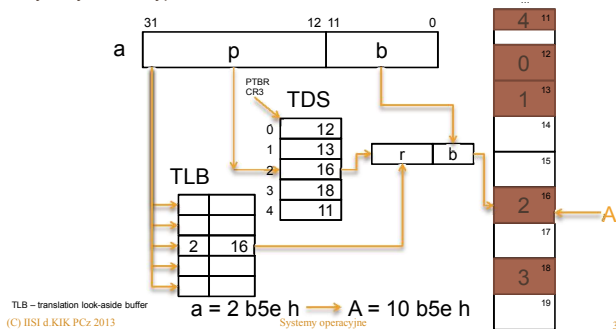
STRONICOWANIE

Pamięć asocjacyjna (skojarzeniowa, TLB-translation look-aside buffer) to:

- ✗ bardzo szybka pamięć wbudowana w procesor
- ✗ adresowana zawartością
- ✗ służąca do przechowywania ostatnio używanych deskryptorów stron (lub segmentów)

STRONICOWANIE

Zastosowanie pamięci asocjacyjnej do przechowywania ostatnio używanych deskryptorów stron



STRONICOWANIE

Efektywny czas dostępu do pamięci z użyciem TLB

$$EAT = HR * (MAT+TLBAT) + (1-HR)*(2* MAT+TLBAT)$$

- ✗ TLBAT – TLB access time (np. 20 ns)
- ✗ HR – hit ratio – współczynnik trafień (np. 80%,98%)
- ✗ $EAT_{80} = 0,8 * 120 + 0,2 * 220 = 140ns$
- ✗ $EAT_{98} = 0,98 * 120 + 0,02 * 220 = 122ns$

PAMIĘĆ PODRĘCZNA

- ✗ Nazywana również notatnikową lub cache
- ✗ Przechowuje ostatnio używane fragmenty pamięci operacyjnej (kilkadziesiąt bajtów) i ich adresy
- ✗ Procesor nie musi odwoływać się do pamięci operacyjnej, jeśli właściwy fragment jest w pamięci notatnikowej
- ✗ Bardzo szybka
- ✗ Najczęściej wbudowana do wnętrza procesora

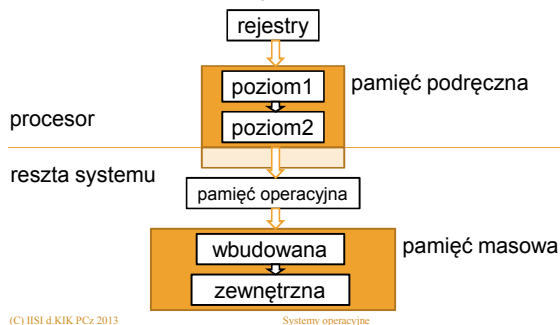
PAMIĘĆ PODRĘCZNA

Może składać się z kilku poziomów:

- ✗ poziom 1 – do kilkudziesięciu kilobajtów
 - ✗ może występować w architekturze Harvard
- ✗ poziom 2 – od kilkudziesięciu kilobajtów do kilkunastu megabajtów
- ✗ ...

PAMIĘĆ PODRĘCZNA

W strukturze pamięci:



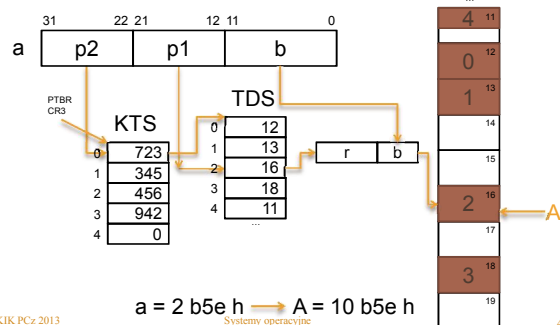
(C) IISI d.KIK PCz 2013

Systemy operacyjne

43

STRONICOWANIE

Stronicowanie dwupoziomowe



(C) IISI d.KIK PCz 2013

Systemy operacyjne

44

STRONICOWANIE

Efektywny czas dostępu do pamięci

$$EAT = 3 * MAT$$

Efektywny czas dostępu do pamięci z użyciem TLB

$$EAT = HR * (MAT + TLBAT) + (1 - HR) * (3 * MAT + TLBAT)$$

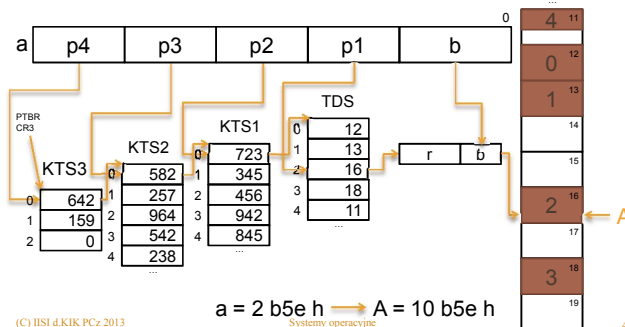
(C) IISI d.KIK PCz 2013

Systemy operacyjne

45

STRONICOWANIE

Stronicowanie wielopoziomowe

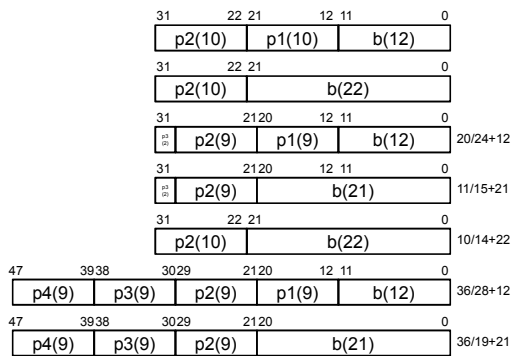


(C) IISI d.KIK PCz 2013

Systemy operacyjne

46

STRONICOWANIE W PROCESORACH INTEL



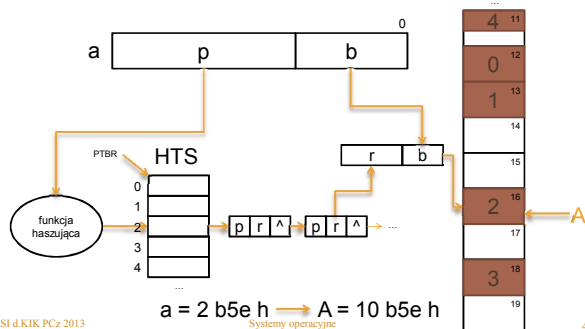
(C) IISI d.KIK PCz 2013

Systemy operacyjne

47

STRONICOWANIE

Haszowana tablica stron – najczęściej dla systemów 64-bitowych. Z powiązanej listy występującej w tablicy wybiera się element dopasowany do strony p.



(C) IISI d.KIK PCz 2013

Systemy operacyjne

48

STRONICOWANIE

- ✘ Odmianą haszowanej tablicy stron jest **tablica stron grupowanych**.
- ✘ Każdy wpis odnosi się do np. kilkunastu stron.

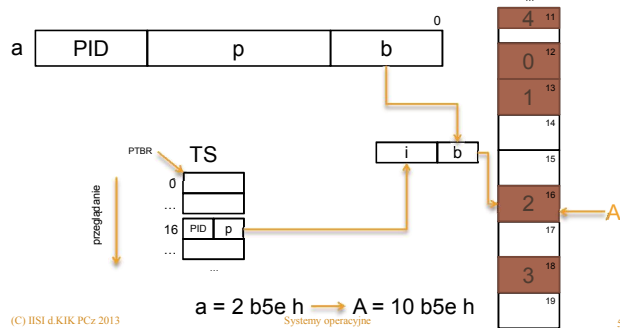
(C) IISI d.KIK PCz 2013

Systemy operacyjne

49

STRONICOWANIE

Odwrócona tablica stron – jedna tablica dla całego systemu - UltraSARC, PowerPC. Często jest haszowana.



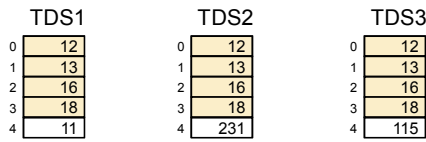
(C) IISI d.KIK PCz 2013

Systemy operacyjne

50

STRONICOWANIE

Strony dzielone – część stron może być dzielona przez kilka procesów np. strony zawierające kod (wznawialny - czysty) programu lub stałe.



(C) IISI d.KIK PCz 2013

Systemy operacyjne

51

STRONICOWANIE

Stronicowanie na żądanie – do pamięci sprowadzane są tylko niektóre strony procesu (*leniwa wymiana*), pozostałe są pobierane tylko wówczas, gdy są potrzebne.

Przebieg:

- ✘ błąd braku strony
- ✘ sprowadzenie brakującej strony do wolnej ramki
- ✘ uaktualnienie tablicy stron
- ✘ wznowienie procesu

(C) IISI d.KIK PCz 2013

Systemy operacyjne

52

STRONICOWANIE

Sprawność stronicowania:

$$EAT = (1-p) \cdot MAT + p \cdot PFST$$

- ✘ PFST – page fault service time – czas obsługi błędu braku strony
- ✘ p – prawdopodobieństwo wystąpienia błędu braku strony

(C) IISI d.KIK PCz 2013

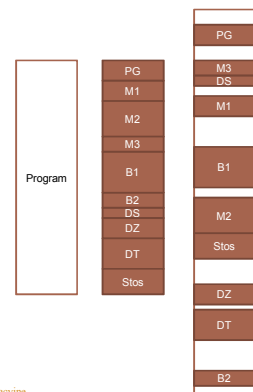
Systemy operacyjne

53

SEGMENTACJA

Segmentacja polega na podziale pamięci przydzielonej procesowi na niezależne fragmenty zwane **segmentami**.

Odzwierciedla ona organizację logiczną pamięci.



(C) IISI d.KIK PCz 2013

Systemy operacyjne

54

SEGMENTACJA

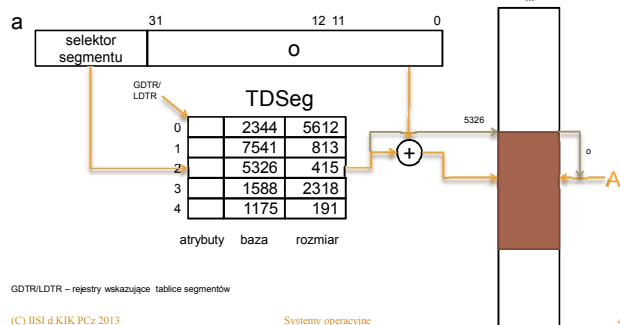
Adres w przestrzeni pamięci wyznacza się:

$$A = f(a) = B + o$$

B – adres bazowy segmentu
o – przesunięcie w segmencie

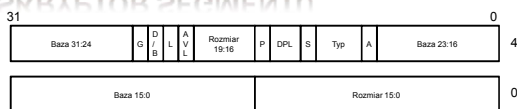
SEGMENTACJA

Adresowanie w segmentacji



GDTR/LDTR – rejestry wskazujące tablice segmentów

DESKRYPTOR SEGMENTU



- * G — Granularity – ziarnistość – rozmiar liczony w bajtach (0) lub 4KB (1)
- * D/B — Default operation size – rozmiar operacji (0 = 16-bit; 1 = 32-bit)
- * L — 64-bit code segment – segment zawiera kod 64-bitowy (IA-32e mode only)
- * AVL — Available for use by system software – dostępne dla programów systemowych
- * DPL — Descriptor privilege level – poziom ochrony
- * P — Segment present – segment obecny w pamięci
- * S — Descriptor type - rodzaj (0 = systemowy; 1 = kod/dane)
- * TYPE — Segment type - typ segmentu (dane, kod, odczyt, zapis, wykonywanie)
- * A — Accessed - segment użyty

SEGMENTACJA

Mogą występować dwie tablice deskryptorów segmentów:

- * globalna
 - + dla systemu operacyjnego
- * lokalna
 - + dla każdego procesu
 - + informacja o niej znajduje się w tablicy globalnej

SELEKTOR SEGMENTU



- * RPL — Requestor's privilege level – poziom ochrony procesu wywołującego
- * TI — Table indicator – wskaźnik rodzaju tablicy
- * CPL – bieżący poziom ochrony
- * EPL – efektywny poziom ochrony $EPL = \max(CPL, RPL)$
- * Dla danych musi być $EPL \leq DPL$
 - + można odwoływać się do danych o niższym/rownym poziomie ochrony
- * Dla kodu $CPL \geq DPL$
 - + można wywoływać programy o wyższym/rownym poziomie ochrony

SEGMENTACJA

Segmenty mogą występować w kilku tablicach deskryptorów segmentów, stają się wówczas segmentami dzielonymi.

TDS1		TDS2		TDS3	
0	1256	0	8172	0	1670
1	5279	1	9731	1	4724
2	16	2	16	2	16
3	18	3	18	3	18
4	11	4	231	4	115
...		

SEGMENTACJA A STRONICOWANIE

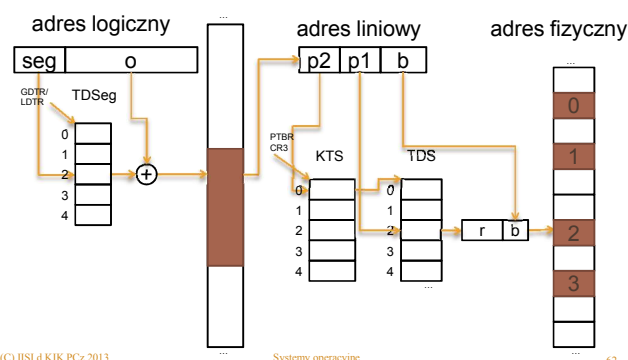
- ✘ Celem segmentacji jest logiczny podział pamięci, natomiast stronicowania fizyczny.
- ✘ Strony mają ustalony rozmiar, segmenty mogą być dowolne.
- ✘ Stronicowanie odbywa się w sposób sprzętowy i przekroczenie zakresu strony powoduje przejście do następnej, w segmentacji podział adresu jest logiczny i nie powoduje przejścia do następnego segmentu.
- ✘ Segmentacja zapewnia poziomą ochrony kodu i danych.

(C) IISI d.KIK PCz 2013

Systemy operacyjne

61

SEGMENTACJA ZE STRONICOWANIEM



(C) IISI d.KIK PCz 2013

Systemy operacyjne

62

SEGMENTACJA ZE STRONICOWANIEM

- ✘ Segmentacja ze stronicowaniem to połączenie cech dwóch metod implementacji pamięci wirtualnej (segmentacji i stronicowania).
- ✘ Polega ona na stronicowaniu segmentów. Pozwala to wyeliminować fragmentację zewnętrzną przy uproszczeniu przydziału:
 - + każda pusta ramka może być użyta na potrzebną stronę.
- ✘ W takiej organizacji pamięci każdy segment może posiadać własną tablicę stron. Przy czym nie musi ona być pełna, a zawiera tyle stron ile jest potrzebne segmentowi.

(C) IISI d.KIK PCz 2013

Systemy operacyjne

63

STRATEGIE PRZYDZIAŁU PAMIĘCI

Strategie przydziału pamięci dzieli się na:

- ✘ strategię rozmieszczenia
- ✘ strategię wymiany
- ✘ strategię pobierania

(C) IISI d.KIK PCz 2013

Systemy operacyjne

64

STRATEGIE PRZYDZIAŁU PAMIĘCI

- ✘ **Przydział ciągły** – podzielenie pamięci na obszary. Jeden obszar może zajmować dokładnie jeden proces.
- ✘ Stopień wieloprogramowości ograniczony przez liczbę obszarów.
- ✘ Przydział ciągły nie jest dobrym rozwiązaniem ze względu na fragmentację wynikającą z pozostałych po procesach dziur.

(C) IISI d.KIK PCz 2013

Systemy operacyjne

65

STRATEGIE PRZYDZIAŁU PAMIĘCI

Strategie rozmieszczania:

- ✘ służą do **określania miejsca** w pamięci głównej, do którego mają być załadowane dane lub kod, to znaczy do wybierania podzbioru niezajętych obszarów

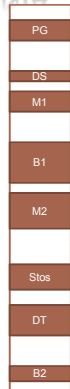
(C) IISI d.KIK PCz 2013

Systemy operacyjne

66

ROZMIESZCZENIE BEZ STRONICOWANIA

- ✘ System prowadzi ewidencję dziur, wiadomo ile jest dziur i jakie są duże
- ✘ Wyróżniamy następujące strategie rozmieszczenia:
 - + najlepsze dopasowanie
 - + najgorsze dopasowanie
 - + pierwsze dopasowanie
 - + algorytm bliźniaków



(C) IISI d.KIK PCz 2013

Systemy operacyjne

67

ROZMIESZCZENIE BEZ STRONICOWANIA

Najlepsze dopasowanie

- ✘ dziury są uporządkowane rosnąco pod względem rozmiaru
- ✘ system szuka bloku najmniejszego spełniającego warunek rozmiaru

(C) IISI d.KIK PCz 2013

Systemy operacyjne

68

ROZMIESZCZENIE BEZ STRONICOWANIA

Najgorsze dopasowanie

- ✘ dziury są uporządkowane malejąco pod względem rozmiaru
- ✘ system umieszcza proces w pierwszej dziurze na liście

(C) IISI d.KIK PCz 2013

Systemy operacyjne

69

ROZMIESZCZENIE BEZ STRONICOWANIA

Pierwsze dopasowanie

- ✘ dziury są uporządkowane rosnąco pod względem ich adresów bazowych
- ✘ system przydziela pierwszą z możliwych dziur

(C) IISI d.KIK PCz 2013

Systemy operacyjne

70

ROZMIESZCZENIE BEZ STRONICOWANIA

Algorytm bliźniaków

- ✘ Rozmiar dziur jest zawsze = 2^n
- ✘ Gdy potrzebna ilość pamięci jest mniejsza niż połowa dziury, system dzieli wybraną dziurę na dwie równe części do momentu aż dziura będzie odpowiednia. Blok umieszczany w powstałej dziurze.
- ✘ Jeśli nie ma wystarczająco dużej dziury system łączy dwie takie same sąsiadujące dziury w jedną dwa razy większą.

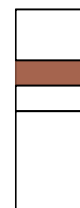
(C) IISI d.KIK PCz 2013

Systemy operacyjne

71

ROZMIESZCZENIE BEZ STRONICOWANIA

Algorytm bliźniaków



(C) IISI d.KIK PCz 2013

Systemy operacyjne

72

ROZMIESZCZENIE ZE STRONICOWANIEM

- ✘ Nie ma potrzeby przechowywania listy dziur
- ✘ Występuje lista wolnych stron
- ✘ Nie powoduje fragmentacji zewnętrznej
- ✘ Występuje fragmentacja wewnętrzna

(C) IISI d.KIK PCz 2013

Systemy operacyjne

73

STRATEGIE PRZYDZIAŁU PAMIĘCI

Strategie wymiany:

- ✘ służą do określenia bloków, które można usunąć z pamięci operacyjnej, aby powstały w niej wolne obszary

(C) IISI d.KIK PCz 2013

Systemy operacyjne

74

WYMIANA ZE STRONICOWANIEM

Proces żąda dostępu do strony w obszarze wymiany. W przypadku braku wolnych ramek na dysk jest odsyłana strona:

- ✘ najdawniej używana
- ✘ najmniej używana
- ✘ najdawniej załadowana

(C) IISI d.KIK PCz 2013

Systemy operacyjne

75

WYMIANA ZE STRONICOWANIEM

Wymiana strony **najdawniej używanej**:

- ✘ w deskrytorze strony musi być zapisany czas odwołania do strony
- ✘ stosowanie tego algorytmu jest obciążone kosztami zapisywania czasu odwołań do wszystkich stron

(C) IISI d.KIK PCz 2013

Systemy operacyjne

76

WYMIANA ZE STRONICOWANIEM

Wymiana **najmniej używanej** strony

- ✘ w tym algorytmie należy odnotowywać częstość użycia strony
- ✘ wadą tej metody jest fakt, że ostatnio załadowana strona może być nierozważnie wymieniona
- ✘ można zabronić wymiany stron załadowanych podczas określonego przedziału czasu

(C) IISI d.KIK PCz 2013

Systemy operacyjne

77

WYMIANA ZE STRONICOWANIEM

Wymiana **najdawniej załadowanej** strony:

- ✘ algorytm ten wymaga zapisu ciągu stron ładowanych do pamięci, co pozwala wskazać najdawniej załadowane strony
- ✘ pomija się w nim fakt, że najwięcej odniesień może być do strony, która najdłużej znajduje się w pamięci

(C) IISI d.KIK PCz 2013

Systemy operacyjne

78

WYMIANA BEZ STRONICOWANIA

Wymieniane segmenty posiadają różny rozmiar, zależy od tego sposób wymiany

- ✘ jeżeli do pamięci ma być załadowany mały segment wystarczy odesłać tylko mały segment
- ✘ jeśli segment jest duży, to trzeba odesłać duży segment lub kilka mniejszych
- ✘ może wystąpić konieczność upakowania pamięci

(C) IISI d.KIK PCz 2013

Systemy operacyjne

79

WYMIANA BEZ STRONICOWANIA

- ✘ Najprostszy algorytm polega na zastąpieniu jednego segmentu oraz sąsiedniej dziury, których łączny obszar wystarczy do pomieszczenia nowego segmentu
- ✘ Gdy takich segmentów jest kilka można zastosować algorytm najdawniej załadowanych, bądź najmniej używanych segmentów

(C) IISI d.KIK PCz 2013

Systemy operacyjne

80

STRATEGIE PRZYDZIAŁU PAMIĘCI

Strategie pobierania:

- ✘ służą do określania momentu, w którym bloki mają być załadowane do pamięci operacyjnej z obszaru wymiany

(C) IISI d.KIK PCz 2013

Systemy operacyjne

81

STRATEGIE POBIERANIA

Strategie pobierania z pamięci pomocniczej:

- ✘ pobieranie na żądanie
- ✘ pobieranie przewidyujące

(C) IISI d.KIK PCz 2013

Systemy operacyjne

82

STRATEGIE POBIERANIA

Pobieranie na żądanie:

- ✘ wystąpienie błędu braku bloku (segmentu lub strony) generuje żądanie pobrania bloku, a strategie rozmieszczania służą do przydziału pamięci dla nowego bloku
- ✘ w systemach bez stronicowania przesyła się bloki najczęściej na żądanie

(C) IISI d.KIK PCz 2013

Systemy operacyjne

83

STRATEGIE POBIERANIA

Pobieranie przewidyujące:

- ✘ system może pobierać nowe bloki w oparciu o:
 - + znajomość konstrukcji programu
 - + wnioskowanie z dotychczasowego działania procesu

(C) IISI d.KIK PCz 2013

Systemy operacyjne

84

ZASADA LOKALNOŚCI

- ✘ Wiele programów wykazuje cechę zwaną **działaniem w kontekście**.
- ✘ Oznacza to, że w dowolnie małych przedziałach czasu program działa wewnątrz konkretnego modułu logicznego.
- ✘ Wykonuje on rozkazy należące do jednej procedury lub pobiera dane z jednego obszaru.



(C) IISI d.KIK PCz 2013

Systemy operacyjne

85

ZASADA LOKALNOŚCI

- ✘ Odniesienia do pamięci wydają się grupować wewnątrz małych obszarów przestrzeni adresów
 - + lokalność odniesień jest jeszcze większa w przypadku częstego występowania pętli
- ✘ Do zmiany tych obszarów dochodzi tylko okresowo.
- ✘ **Procesy chwilowo wykonują się lokalnie.**

(C) IISI d.KIK PCz 2013

Systemy operacyjne

86

SZAMOTANIE

- ✘ Gdy proces nie ma wystarczającej ilości ramek, szybko może nastąpić brak strony.
- ✘ Stronę brakującą system sprowadza do pamięci operacyjnej, wymieniając z inną stroną.
- ✘ Może dojść do sytuacji, w której wymieniona strona (już nieobecna w pamięci) jest potrzebna dla aktywnego procesu, następuje ponowne jej sprowadzenie do pamięci.
- ✘ W rezultacie w systemie będą następowały po sobie kolejne braki stron i ich wymiany.
- ✘ Procesy będą wskazywać braki stron, system dokona wymiany stron, po czym sprowadzi je z powrotem.

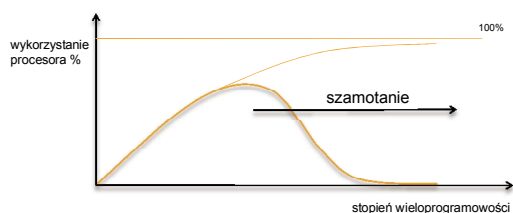
(C) IISI d.KIK PCz 2013

Systemy operacyjne

87

SZAMOTANIE

Szamotanie to duża aktywność stronicowania, system spędza więcej czasu na stronicowaniu (wymianie) niż na wykonaniu procesów.



(C) IISI d.KIK PCz 2013

Systemy operacyjne

88

SZAMOTANIE

- ✘ Efekt szamotania można ograniczyć za pomocą **lokalnego algorytmu zastępowania**
 - + Przy zastępowaniu lokalnym, gdy jakiś proces zaczyna się szamotać wówczas nie wolno mu zabierać ramek innego procesu i doprowadzać go także do szamotania
- ✘ Aby zapobiec szamotaniu należy dać procesowi tyle ramek, ile potrzebuje.

(C) IISI d.KIK PCz 2013

Systemy operacyjne

89

SZAMOTANIE

Metodą mierzenia szamotania jest mierzenie częstości braków stron:

- ✘ ustala się górną i dolną granicę pożądanego poziomu braków stron
- ✘ jeśli proces przekracza górną granicę, to dostaje dodatkową ramkę
- ✘ jeśli częstość występowania braków stron spada poniżej dolnej granicy to usuwa się ramkę procesu



(C) IISI d.KIK PCz 2013

Systemy operacyjne

90

SZAMOTANIE – MODEL STREFOWY

W trakcie wykonywania proces przechodzi z jednej strefy do innej.

- ✘ Strefa to zbiór stron pozostających we wspólnym użyciu.
- ✘ Program składa się z wielu stref, które mogą na siebie zachodzić.
- ✘ Procesowi przydziela tyle ramek, aby mógł w nich zmieścić swoją bieżącą strefę.
- ✘ Gdy przydzielone jest mniej ramek, niż wynosi rozmiar bieżącej strefy, wówczas proces zaczyna się szamotać.

(C) IISI d.KIK PCz 2013

Systemy operacyjne

91

MODEL ZBIORU ROBOCZEGO

- ✘ Opiera się na założeniu, że program ma charakterystykę strefową.
- ✘ Parametr Δ definiuje okno zbioru roboczego.
- ✘ **Za zbiór roboczy przyjmuje się zbiór stron, do których nastąpiło Δ ostatnich odwołań.**

(C) IISI d.KIK PCz 2013

Systemy operacyjne

92

MODEL ZBIORU ROBOCZEGO

- ✘ **Jeśli zbiór roboczy procesu znajduje się w pamięci, wówczas proces wykonuje się bez szamotania.**
- ✘ Strona aktywnie używana znajduje się w zbiorze roboczym.
- ✘ Gdy strona przestanie być używana, wypada ze zbioru roboczego po Δ jednostkach czasu liczonych od ostatniego odwołania do niej.

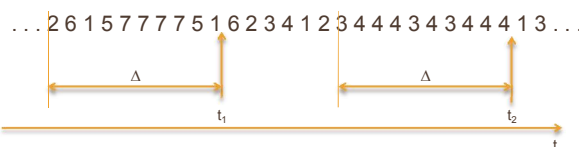
(C) IISI d.KIK PCz 2013

Systemy operacyjne

93

MODEL ZBIORU ROBOCZEGO

Ślad odniesień:



Dla $\Delta = 10$

odpowiednie zbiory robocze wynoszą:

$$ZR(t_1) = \{1, 2, 5, 6, 7\}$$

$$ZR(t_2) = \{3, 4\}$$

(C) IISI d.KIK PCz 2013

Systemy operacyjne

94

MODEL ZBIORU ROBOCZEGO

Dokładność zbioru roboczego zależy od wyboru parametru Δ :

- ✘ Jeśli jest za mały – nie obejmie całego zbioru
- ✘ Jeśli za duży, to może zachodzić na kilka stref programu
- ✘ W skrajnym przypadku, gdy Δ jest nieskończenie duże, zbiorem roboczym staje się zbiór stron, z którymi proces kontaktował się w trakcie działania.

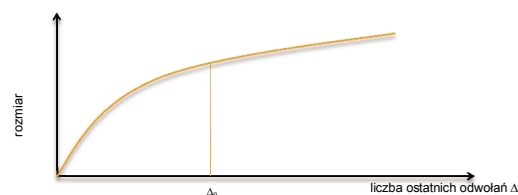
(C) IISI d.KIK PCz 2013

Systemy operacyjne

95

MODEL ZBIORU ROBOCZEGO

Oczekiwany rozmiar zbioru roboczego.



(C) IISI d.KIK PCz 2013

Systemy operacyjne

96