

## Pamięć wirtualna

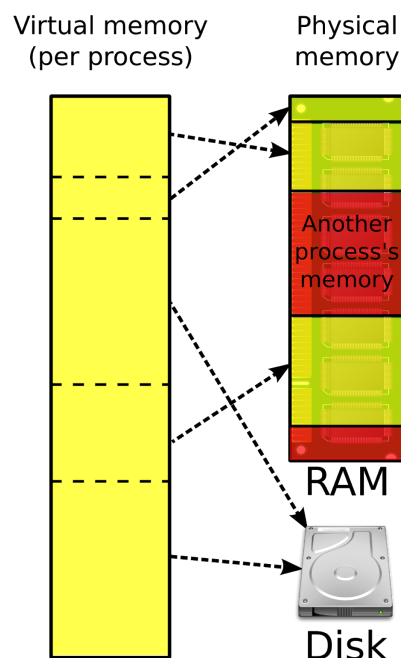
Odwieczny problem ograniczonej dostępnej ilości pamięci fizycznej, zmusił informatyków do poszukiwania technik na ominięcie braku wymaganej ilości pamięci w różnych sytuacjach. Jednym z mechanizmów był tzw. "overlay", kiedy to program był dzielony na ukryte fragmenty, z których tylko jeden w danym momencie rezydował w pamięci. Za każdym razem kiedy inny fragment był wymagany odczytywało się go i wpisywało do pamięci nadpisując wcześniejszą jej zawartość. Mechanizm pamięci wirtualnej dzieli całą pamięć fizyczną komputera na zbiór stron. Dzięki temu zamiast jednego nadpiswalnego rejonu, pamięć została podzielona na wiele takich nadpiswalnych części.

Pamięć wirtualna odseparowuje pamięć logiczną od jej fizycznej realizacji. Można ją zaimplementować jako:

- stronicowanie na żądanie
- segmentacje na żądanie - procesy przebywają w pamięci pomocniczej (na dysku). Dla wykonania sprowadza się proces do pamięci, tylko te strony, które są potrzebne. Typowanie potrzebnych stron odbywa się podczas poprzedniego pobytu procesu w pamięci.

Jeśli proces odwołuje się do strony, której nie ma w pamięci, to:

- System sprawdza, czy odwołanie do pamięci było dozwolone czy nie (bit poprawności)
- Jeśli było poprawne, sprowadza stronę do pamięci, modyfikuje tablicę stron i wznowia działanie procesu



**Zalety pamięci wirtualnej:**

- Program nie jest ograniczony wielkością pamięci fizycznej
- Każdy program zajmuje mniej miejsca niż program "kompletny". Można więc w tym samym czasie wykonywać więcej zadań, polepszając wykorzystanie procesora
- maleje liczba operacji wejścia- wyjścia koniecznych do załadowania programów do pamięci oraz do realizacji wymiany

**Algorytmy zastępowania stron**

Założenie, że tylko część stron każdego procesu jest potrzebna w pamięci może doprowadzić do nadprzydziału(nadmiar procesów w pamięci i absolutny brak wolnych ramek). Aby nie blokować procesu potrzebującego kolejnej ramki, stosuje się zastępowanie stron.

- Algorytmy optymalizowane pod kątem minimalizacji częstości braku stron.
- Algorytmy ocenia się na podstawie ich wykonania dla pewnego ciągu odniesień do pamięci i zsumowanie liczby braku stron w tym ciągu.

**Algorytm FIFO (first in, first out)**

O każdej ze stron zapamiętuje się informacje, kiedy ona została sprowadzona do pamięci. Zastępuje się najstarszą stroną.

**Algorytm optymalny**

Zastępuje tę stronę, która najdłużej nie będzie używana. Ten algorytm jest trudny w realizacji, ponieważ wymaga wiedzy o przyszłym ciągu odniesień. Jest używany głównie do teoretycznych studiów porównawczych ( o ile % dany algorytm jest gorszy od optymalnego)

**Algorytm LRU (latest recently used)**

Zastępowanie stron które były najdawniej używane. Typowanie najstarszych stron poprzez:

- licznik (w tablicy stron jest rejestr czasu ostatniego używania)
- stos (przy każdym odwołaniu do strony, jej numer jest wyjmowany i umieszczany na szczycie stosu)

### **Algorytm MFU(most frequently used)**

Wprowadzony licznik odwołań do strony. Ten algorytm wyrzuca stronę najczęściej używaną, ponieważ zakłada że najrzadziej używana jest niedawno wprowadzona do pamięci i będzie niedługo w użyciu. Przeciwnieństwem tego algorytmu jest algorytm **LFU(least frequently used)** który w pierwszej kolejności pozbywa się najrzadziej używanych stron.

### **Źródła:**

[http://students.mimuw.edu.pl/SO/Projekt02-03/temat1-g5/pj189398/ref\\_wstep.html](http://students.mimuw.edu.pl/SO/Projekt02-03/temat1-g5/pj189398/ref_wstep.html)

<http://www.metal.agh.edu.pl/~wilk/wyk-os-io/io-wyk8.pdf>