Teoria pamięci komputerowej

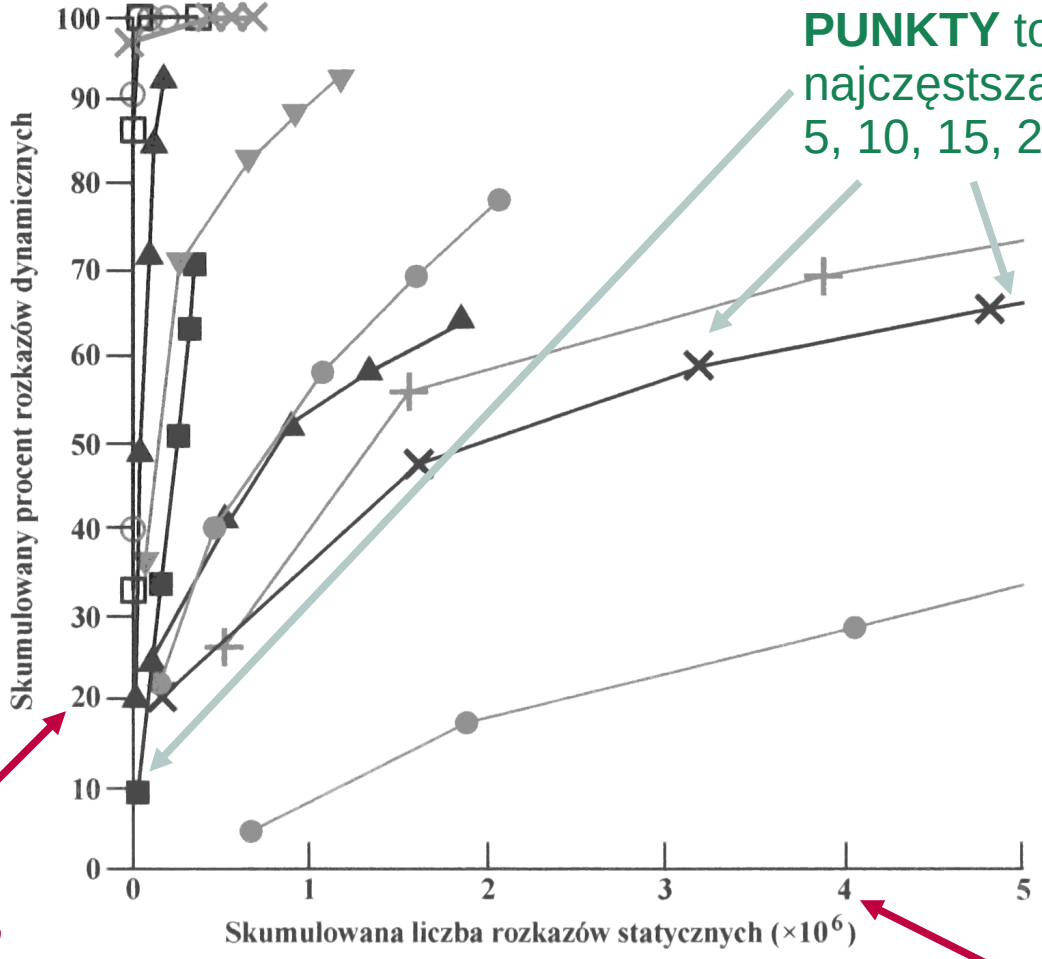
- Pamięć komputerowa to konglomerat wyników z wielu innych dziedzin nauki
- Różnorodność wymogów dla systemów komputerowych ma odzwierciedlenie w różnych rodzajach pamięci
- Pewne wspólne cechy programów komputerowych sugerują rozwiązania w zakresie organizacji pamięci

Natura programów

- Sekwencyjne wykonanie poleceń
  - `FETCH: adres A, adres A+1, adres A+2, …`
- Wywoływanie wąskiego zbioru procedur
  - `CALL/JMP: pA, pB, pC, pB …`
- Iteracje: wielokrotne, niewielka liczba instrukcji
  - ```
    for(i=0;i<10;i++)
    { c1 = getchar(stdin); c2 = code(c1); fputc(c, fOut); }
    ```
- Przetwarzane dane: w strukturach, rekordach
  - `int a[1000]; for(i=0;i<1000;) { a[i] = 15; }`
    ```

Wnioski

- Powtarzalne schematy dostępu do pamięci
- Niejednorodny rozkład odwołań do pamięci
- Prawdopodobieństwo odniesienia się do danej lokacji zmienne w czasie (wolno)
- Korelacja między sekwencją odwołań uprzednią a następną



PUNKTY to procedury:
najczęstsza
5, 10, 15, 20 najczęstszych

rozkazy
„w historii”

rozkazy w pamięci

Rysunek 4.5. Lokalność rozkazów na podstawie ponownego wykorzystania kodu w jedenastu programach testowych SPEC CPU2006

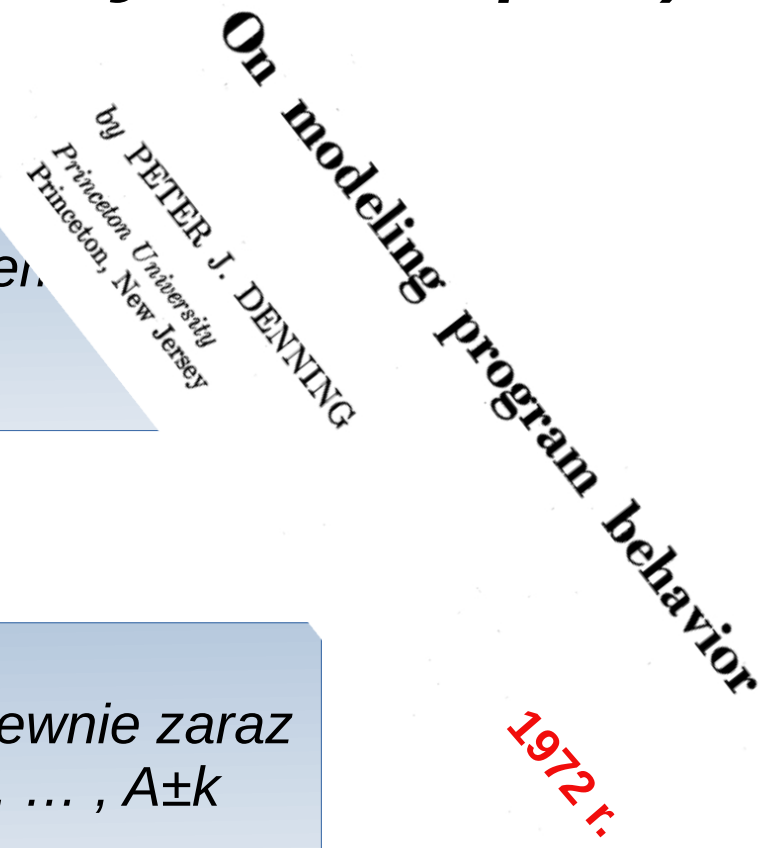
Zasada lokalności (*Locality Principle*)

- Lokalność czasowa

Niedługo wykonam to, co wykonałem chwilą

- Lokalność przestrzenna

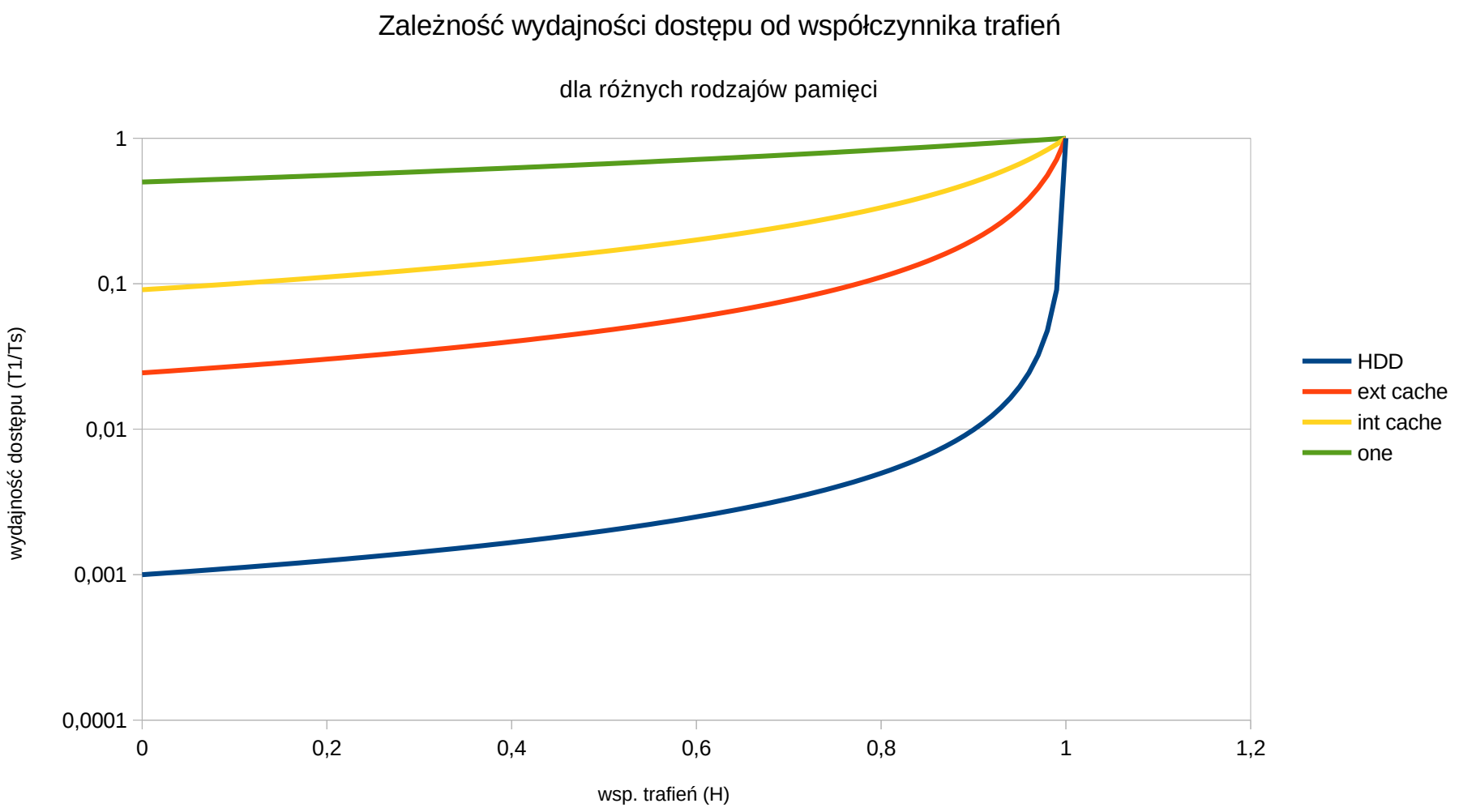
Skoro już zaglądam do adresu A, to pewnie zaraz będę zaglądał do adresu A±1, A±2, … , A±k


On modeling program behavior

by PETER J. DENNING

Princeton University
Princeton, New Jersey

1972 r.


Zależność wydajności dostępu od współczynnika trafień
dla różnych rodzajów pamięci
wydajność dostępu (T1/Ts)
1
0,1
0,01
0,001
0,0001
0
0,2
0,4
0,6
0,8
1
1,2
wsp. trafień (H)
HDD
ext cache
int cache
one

NOSZENIE:

- po jednej teczce
- po kilka teczek
- ile wejdzie na biurko

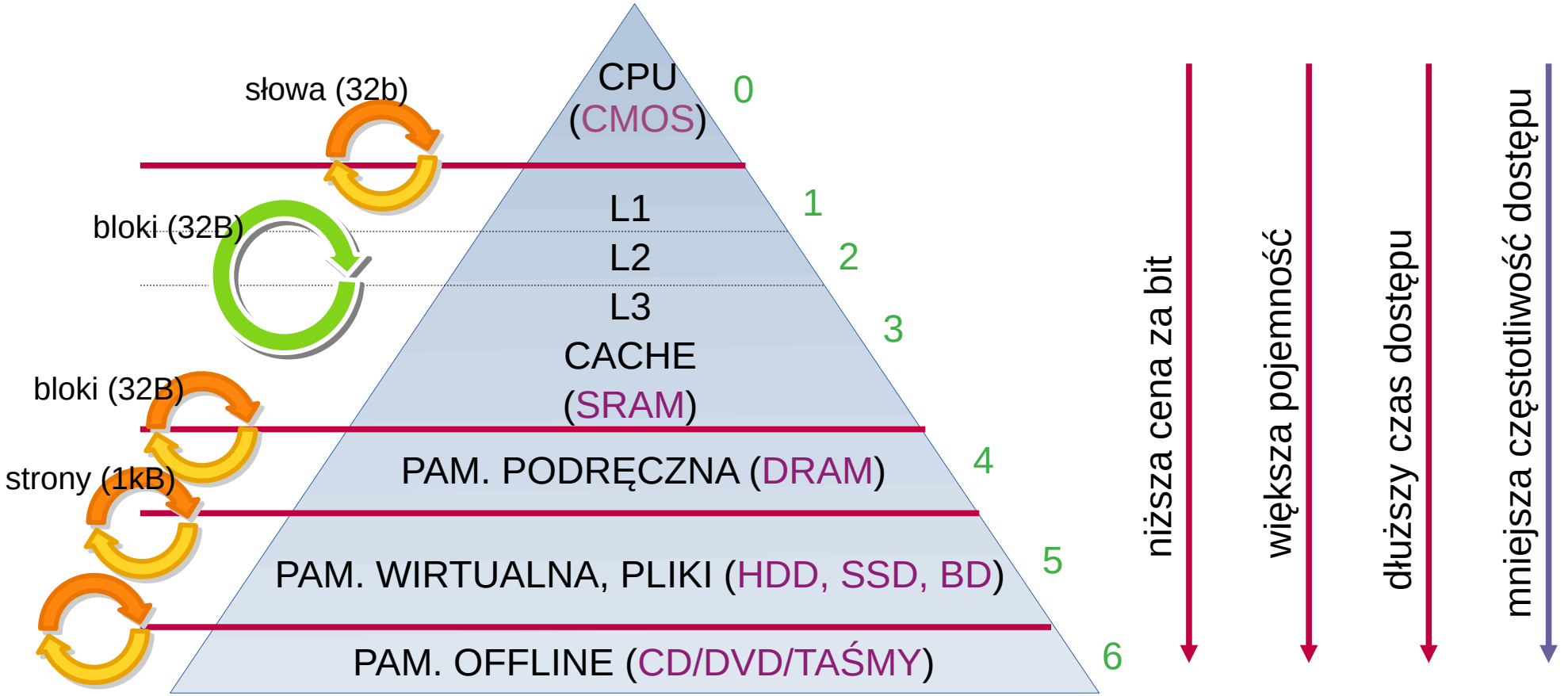
ODDAWANIE:

- najmniej używaną
- wszystkie na biurku
- najwcześniej przyniesioną

OPCJE:

- dostawić regał po drodze
- robić notatki

Uwaga: tutaj mówimy o przenoszeniu teczek, w pamięci będą się tworzyć **kopie**



- koszt/B: $C_i > C_{i+1}$
- czas dostępu: $T_i < T_{i+1}$
- prędkosć transferu: $R_i < R_{i+1}$
- rozmiar: $S_i < S_{i+1}$

Rodzaje dostępu do pamięci

- **Sekwencyjny**
  - pamięć w rekordach
  - dostęp po kolei, przepisywanie -> przesuwanie
- **Swobodny (*Random*)**
  - każda lokalizacja ma swój unikalny adres
- **Bezpośredni (*Direct*)**
  - adres odpowiada lokalizacji fizycznej
  - dostęp najpierw „z grubsza”, potem przeszukiwanie
- **Skojarzeniowy (*Associative*)**
  - dostęp swobodny, na podstawie zawartości słowa
  - przeszukiwana cała pamięć jednocześnie

- Pewna ilość szybkich, drogich pamięci…
- … uzupełniona większą ilością tańszej…
- … na różnych poziomach...
- … przy zagwarantowaniu że dostęp do wolniejszych pamięci będzie rzadszy

Przykład

Są dwa poziomy pamięci: M0, M1.

Dostęp do M0 to 0,01 μs, do M1 – 0,1 μs.

Jeśli słowo jest w M0 – procesor ma dostęp. Jeśli jest w M1 – najpierw słowo ładowane jest do M0. (Pomijamy czas sprawdzania gdzie jest słowo).

1) Jak wygląda średni czas dostępu względem współczynnika trafień (*hit rate*)?
2) Jeśli H=95%, jaki jest średni czas dostępu?

Zasady

- Zapewnić lokalność
- Zasada nakładania

Potrzebne zasoby, znajdujące się w M_n, są kopiowane do M_{n-1}, M_{n-2}, …, M_0, czyli:
$M_i \subseteq M_{i+1}$

- Spójność pamięci
  - pionowa
  - pozioma