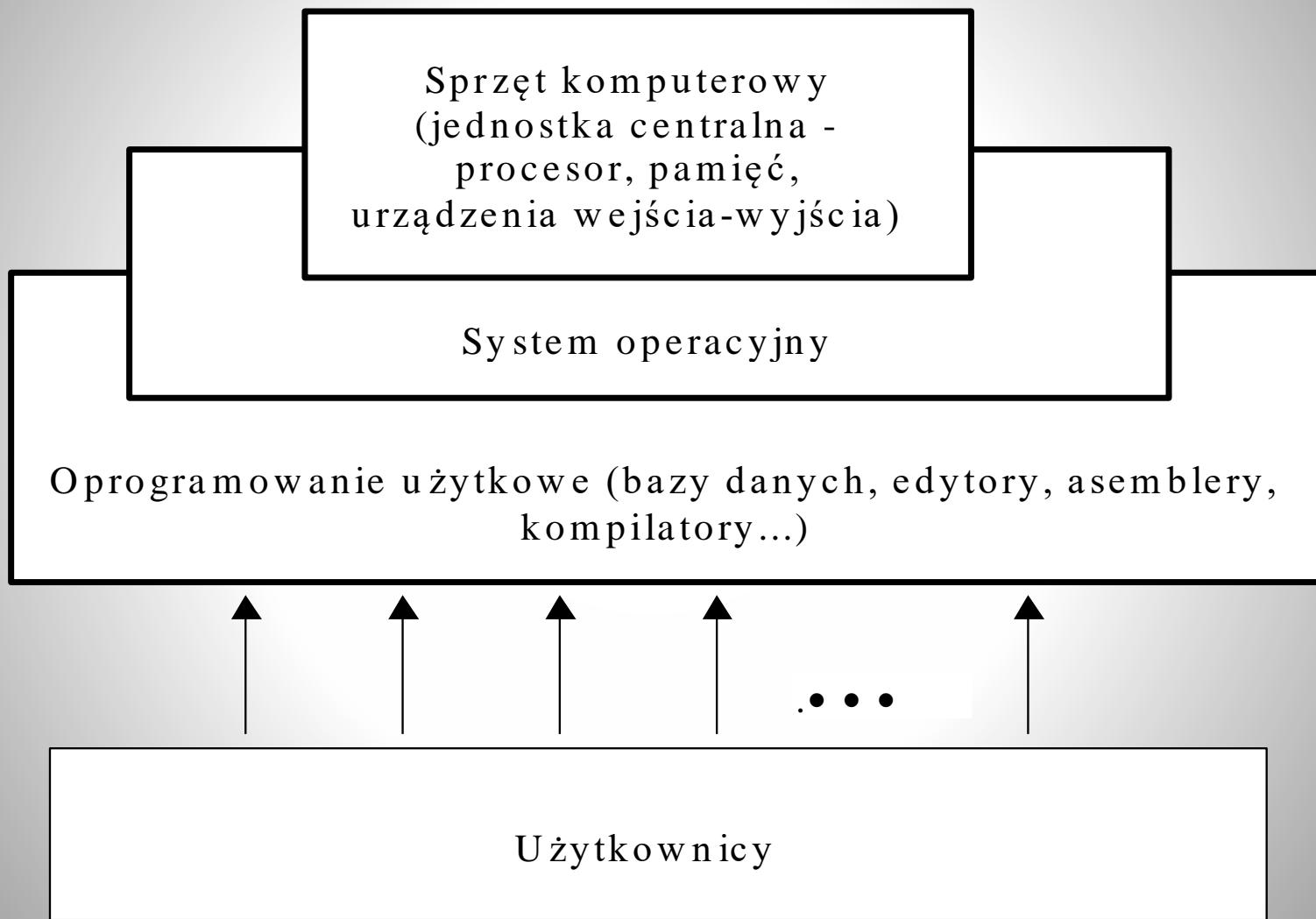
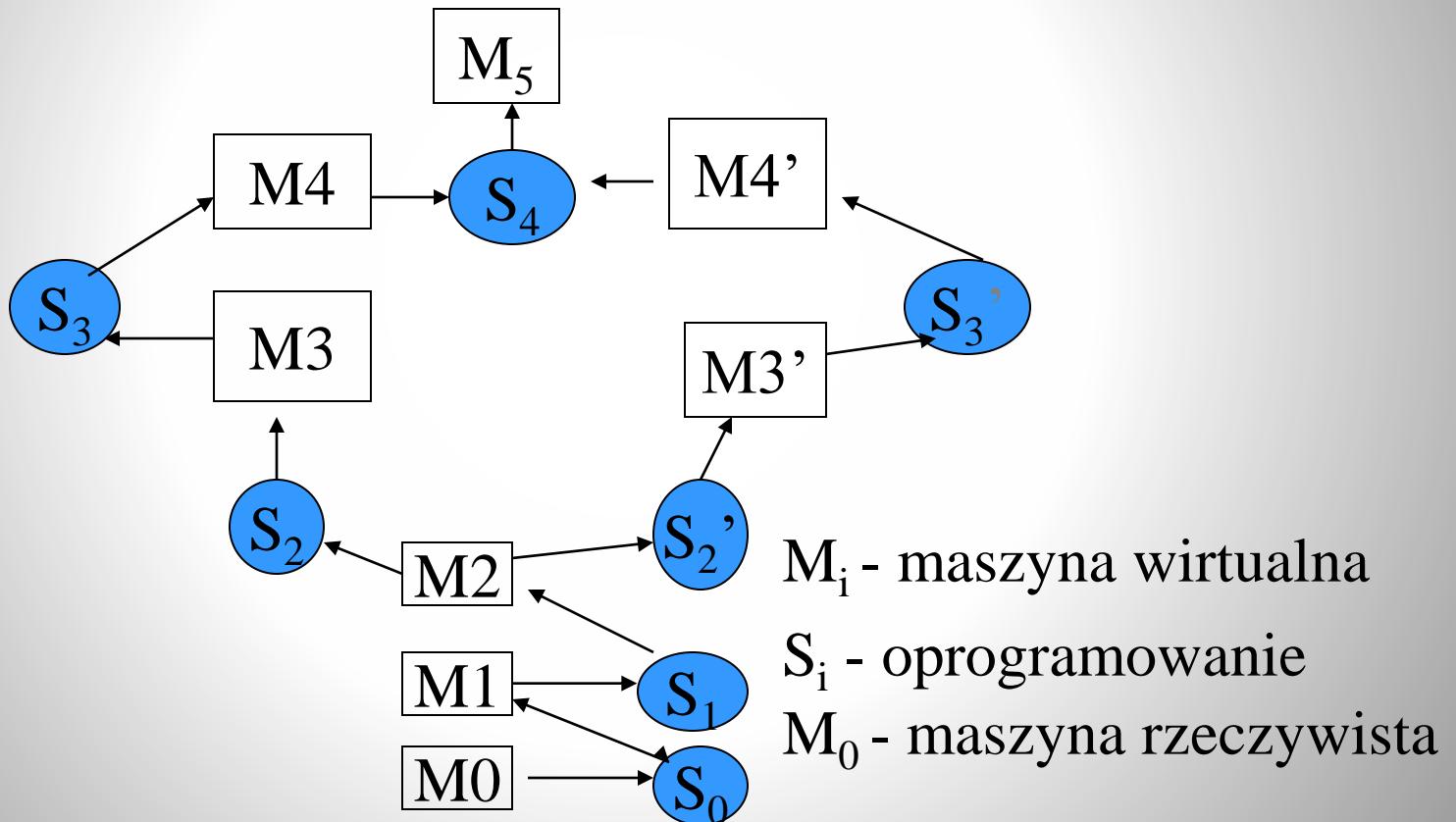


Schemat systemu komputerowego



Hierarchia maszyn wirtualnych wg Weidermana



System operacyjny jest to zbiór programów umożliwiających:

- efektywny podział urządzeń systemu (zasobów maszyny)
- dostarczenie użytkownikowi komputera wirtualnego dla jego potrzeb
- jednoczesną realizację procesów obliczeniowych dla potrzeb jednego lub wielu użytkowników

Kategorie „czystych” systemów operacyjnych (tryby pracy SO)

- dla przetwarzania wsadowego
(off-line, batch)
- z podziałem czasu - tryb bezpośredni, interakcyjny (on-line)
- dla działania w czasie rzeczywistym
(real-time)

Metody realizacji systemu operacyjnego:

- zastosowanie wieloprogramowości
- zastosowanie wymiany (swapping)
- wieloprocesorowość

Zadania systemu wieloprogramowego:

- zarządzanie procesami i zasobami
- zarządzanie pamięcią operacyjną
- organizacja systemu plików
- ochrona (zabezpieczenie przed zniszczeniem SO i innych plików, ochrona dostępu - tajność)

Rozwój systemów operacyjnych

- proste systemy operacyjne
- systemy wsadowe
- buforowanie i spooling
- wieloprogramowość
- wielozadaniowość
- systemy dla komputerów osobistych
- systemy równoległe
- systemy rozproszone
- systemy czasu rzeczywistego

Systemy o charakterze bezpośrednim

- dostęp do komputera oparty o harmonogram pracy maszyny
- programista = operator
- powstanie oprogramowania wspomagającego
 - biblioteki funkcji (głównie we/wy)
 - asemblery, później kompilatory języków (Cobol, Fortran)
 - programy łączące i ładujące
- brak oprogramowania rezydentnego
- nieefektywne wykorzystanie zasobów
 - niskie wykorzystanie CPU
 - pełna sekwencyjność pracy urządzeń
 - przestoje sprzętu związane z wykonywaniem czynności operatorskich

Systemy oparte o pracę wsadową

- operator (nie programista) gromadzi zadania we wsady,
 - zmniejsza się czas instalacji,
 - nadal jest to niewygodne dla użytkownika
- automatyczne porządkowanie zadań (sterowanie przekazywane z jednego do drugiego zadania)
- rezydentny monitor
- karty sterujące przetwarzaniem wsadu (*Job Control Language*)
- zwiększoa przepustowość w systemie kosztem średniego czasu obrotu zadania

Przetwarzanie pośrednie

- wsad kopiowany jest na nośnik magnetyczny (taśmę) *off-line*
- wynik także kopiowany jest na taśmę i drukowany *off-line*
- Korzyści
 - komputer główny nie jest ograniczony prędkością pracy czytników kart i drukarek, a jedynie prędkością szybszych stacji taśmowych
 - nie są potrzebne zmiany w programach użytkowych przy przejściu do trybu pracy pośredniej
 - możliwość używania wielu systemów czytnik-taśma i taśma drukarka dla jednego CPU

Zrównoleglenie we/wy w pojedynczej maszynie

buforowanie

we → bufor → jedn.centralna → bufor → wy

Metoda jednoczesnego wykonywania obliczeń i wejścia-wyjścia dla jednego zadania:

- nie eliminuje całkowicie przestojów CPU czy urządzeń we-wy
- wymaga przeznaczenia pamięci na systemowe bufory
- niweluje wahania w czasie przetwarzania danych

Zrównoleglenie we/wy w pojedynczej maszynie

spooling (simultaneous peripheral operation on-line)

przenoszenie danych do szybszej pamięci zewnętrznej



Metoda jednoczesnego wykonywania wejścia-wyjścia jednego zadania i obliczeń dla innego zadania

- Możliwe dzięki upowszechnieniu się systemów dyskowych
- Podczas wykonywania jednego zadania system operacyjny:
 - czyta następne zadanie z czytnika kart na dysk (kolejka zadań)
 - drukuje umieszczone na dysku wyniki poprzedniego zadania
- *Pula zadań* - możliwość wyboru kolejnego zadania do wykonania

Wieloprogramowość

- wprowadza się wiele zadań do wykonania
- program jest przetwarzany w procesorze centralnym dopóki nie musi być wykonane we/wy
- szeregowanie zadań
- efektywne wykorzystanie zasobów systemu
- przetwarzanie wsadowe

Wielozadaniowość

- zadania przetwarzane współbieżnie (podział czasu)
- dostęp bezpośredni zamiast przetwarzania wsadowego
- priorytety

Systemy operacyjne dla komputerów osobistych

- mikrokomputery – lata 70
- początkowo brak mechanizmów ochrony
- DOS, Windows, OS/2, Apple Macintosh
- adaptacja technik stosowanych w większych so (podział czasu, pam. wirtualna, ochrona)
- problem ochrony

Systemy równoległe

Systemy wieloprocesorowe z więcej niż jednym CPU

- Systemy ściśle powiązane - procesory dzielą pamięć i zegar; komunikacja poprzez pamięć dzieloną
- Korzyści: zwiększoną przepustowość, ekonomiczne, zwiększoną niezawodność
- Symetryczna wieloprocesorowość (każdy procesor wykonuje identyczną kopię systemu operacyjnego, wiele procesów może się wykonywać równocześnie, bez spadku wydajności)
- Asymetryczna wieloprocesorowość (każdy procesor wykonuje przydzielone mu zadanie; procesor główny szereguje i przydziela pracę procesorom podległym; częściej występuje w dużych systemach)

Systemy rozproszone

- luźno powiązane; różne procesory
- przetwarzanie bezpośrednie
- przyśpieszenie obliczeń
- podział zasobów na prywatne i współdzielone
- niezawodność (przejęcie zadań uszkodzonej jednostki przez inne)
- łączność między użytkownikami

Systemy czasu rzeczywistego

- ograniczenia czasowe (rygorystyczne, łagodne)
- stosowanie szybkich pamięci ROM; bez pam. wirtualnej

JĄDRO SYSTEMU

- główna część systemu operacyjnego
- zadaniem jądra systemu - stworzenie środowiska, w którym mogą działać procesy
- udogodnienia sprzętowe
 - mechanizm przerwań
 - ochrona pamięci operacyjnej
 - zbiór rozkazów uprzywilejowanych
 - zegar czasu rzeczywistego

Sterowanie przerwaniami (interrupt drive)

Algorytm układu sterowania JC

- pobierz następny rozkaz
- zwiększ licznik rozkazów
- wykonaj rozkaz
- jeśli jest przerwanie, realizuj to, które ma największy priorytet
- przejdź do punktu pierwszego

Mechanizm przerwań - przechowuje wartości licznika rozkazów przerywanego procesu i przekazuje sterowanie do ustalonego miejsca w pamięci operacyjnej - programu obsługi przerwań (wektor przerwań - tablica wskaźników do procedur obsługi)

Ochrona pamięci operacyjnej

- Ochrona wektora przerwań
- Ochrona procedur obsługi przerwań
- Oddzielenie obszaru pamięci programów
(rejestr bazowy; graniczny)

Zbiór rozkazów uprzywilejowanych

bit trybu pracy procesora: 0 – monitor
 1- użytkownik

tryb monitora; nadzorcy (*supervisor mode*) - część listy rozkazów maszynowych zarezerwowanych wyłącznie do użytku systemu operacyjnego

rozkazy uprzywilejowane:

- włączanie i wyłączanie przerwań
- przełączanie procesora od procesu do procesu
- obsługiwanie rejestrów używanych przez sprzętową ochronę pamięci operacyjnej
- wykonywanie operacji wejścia i wyjścia
- zatrzymywanie procesora centralnego i sterowanie jego operacjami

Zegar czasu rzeczywistego - zegar sprzętowy, który wysyła przerwania w ustalonych odstępach czasu rzeczywistego; jest podstawą realizacji polityki planowania przydziału i rozliczania zasobów, wykorzystanych przez różnych użytkowników.

czasomierz (timer)

STRUKTURY SYSTEMÓW KOMPUTEROWYCH

- Struktura jednolita
- Struktura warstwowa
- Struktura klient-serwer

Struktura jednolita

pierwsze systemy operacyjne - nie mają żadnej struktury

- ogromny program zawierający wiele procedur
- procedury mogą bez żadnych ograniczeń wywoływać i być wywoływanie przez inne procedury
- w niewielkim stopniu ograniczony rozmiar danych systemu
- utrzymywanie takiego systemu - wiele problemów: wprowadzenie zmian do jednej procedury może być przyczyną kłopotów w innych częściach systemu

Programy użytkowe

Rezydujące programy systemowe

Programy obsługi urządzeń z poziomu MS-DOS

Programy obsługi urządzeń z poziomu pamięci
ROM BIOS

Warstwowa struktura systemu MS-DOS

Użytkownicy

programy shell i polecenia
kompilatory i interpretery
biblioteki systemowe

Interfejs funkcji systemowych jądra

sygnały	system plików	planowanie przydziału procesora
obsługa terminali	wymiana	zastępowanie stron
system znakowego we/wy	system blokowego we/wy	stronicowanie na żądanie
programy obsługi terminali	programy obsługi dysków i taśm	pamięć wirtualna

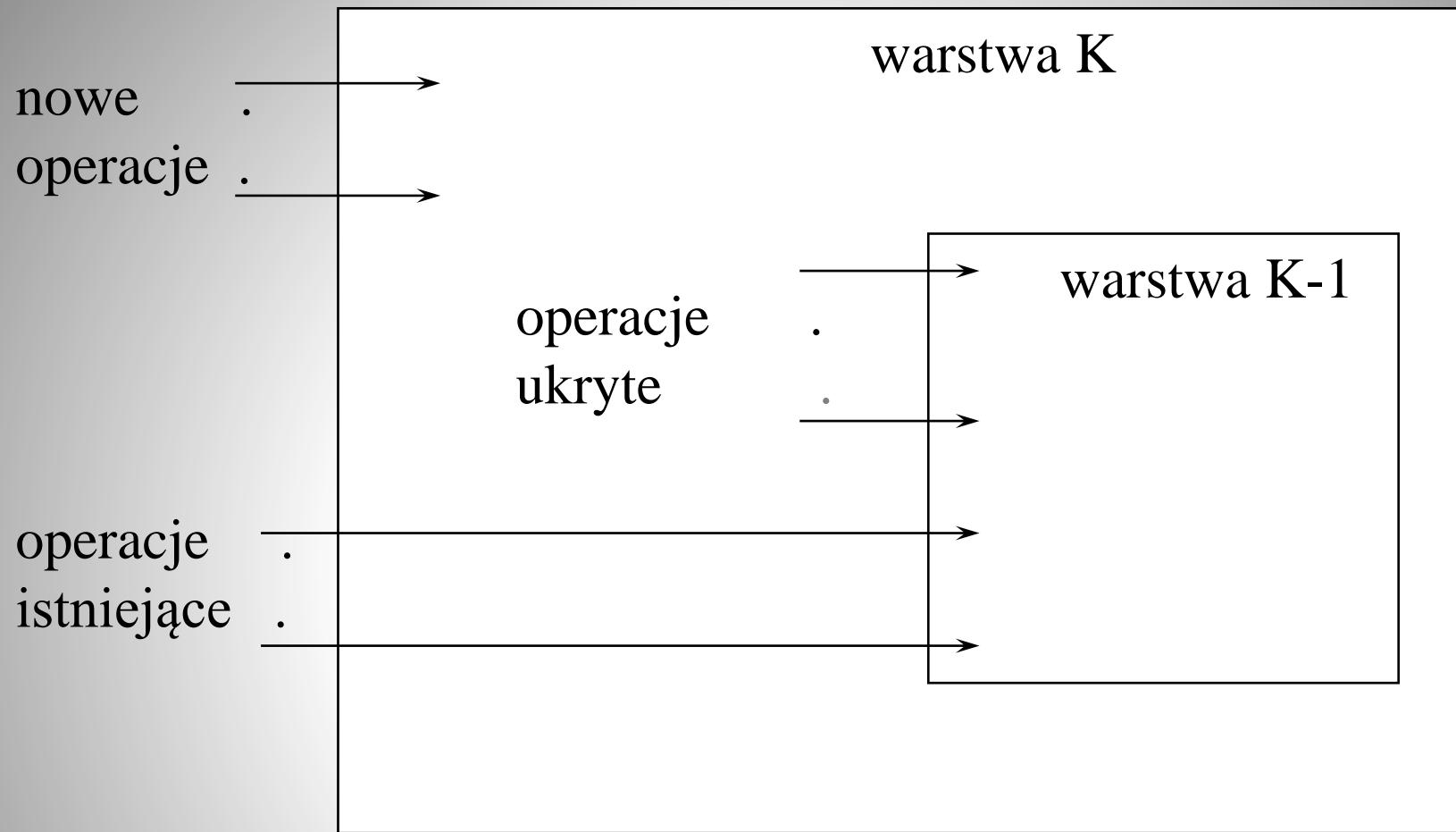
Interfejs między jądrem a sprzętem

sterowniki terminali terminale	sterowniki urządzeń dyski i taśmy	sterowniki pamięci pamięć operacyjna
-----------------------------------	--------------------------------------	---

Struktura systemu UNIX

Struktura warstwowa

- podział na moduły, które są połączone w warstwy
- każda warstwa spełnia funkcje, które zależą tylko od warstwy znajdującej się poniżej
- zaleta: podział na moduły zmniejsza stopień wzajemnych zależności między różnymi składowymi systemu oraz pozwala uniknąć niepożądanych powiązań



Warstwy systemu operacyjnego

Struktura warstwowa systemu THE

- Poziom 5 Programy użytkowników
- Poziom 4 Buforowanie urządzeń we/wy
- Poziom 3 Program obsługi konsoli operatora
- Poziom 2 Zarządzanie pamięcią
- Poziom 1 Planowanie przydziału procesora
- Poziom 0 Sprzęt

Struktura warstwowa systemu Venus

- Poziom 6 Programy użytkowników
- Poziom 5 Programy obsługi i planowania przydziału urządzeń
- Poziom 4 Pamięć wirtualna
- Poziom 3 Kanał we/wy
- Poziom 2 Planowanie przydziału procesora
- Poziom 1 Interpretator rozkazów
- Poziom 0 Sprzęt

Struktura klient-serwer

- podział systemu operacyjnego na moduły, które spełniają pewne wyodrębnione zadania
- moduły te nie są rozmiieszczone w warstwach, ale traktowane mniej lub bardziej równorzędnie
- nie komunikują się między sobą poprzez wywoływanie procedur, ale wysyłają komunikaty za pośrednictwem centralnego programu obsługi komunikatów
- komunikaty mogą być wysyłane w obie strony, dzięki czemu można taką samą drogą przekazywać wyniki
- moduł wysyłający początkowy komunikat - **klient**, moduł odbierający ten komunikat **serwer**

Pamięć operacyjna (main memory)

- obszar bezpośrednio dostępny dla procesora
- rozkazy: **load, store** (PAO \Leftrightarrow rejestr procesora)
- cykl rozkazowy:

- pobranie rozkazu z PAO do rejestrów rozkazów
- dekodowanie
- realizacja (ew. pobranie argumentów z PAO)

Program i dane nie mogą być na stałe w PAO

- za mała
- ulotna

Pamięć pomocnicza (secondary storage)

- rozszerzenie PAO
- trwałe przechowywanie dużej ilości danych
- źródło i miejsce przeznaczenia informacji

sprowadzanie danych do PAO (op. we/wy)

- rejestr sterowników \Leftrightarrow PAO
- we/wy odwzorowane w pamięci (memory mapped I/O)
 - wydzielenie adresów PAO na rejesty urządzzeń (ekran, porty szeregowe i równoległe)

Wysłanie ciągu bajtów przez port szeregowy:

- procesor wpisuje 1 bajt do rejestru danych
 - procesor ustawia bit w rejestrze kontrolnym
 - urządzenie pobiera bajt danych
 - urządzenie zeruje bit w rejestrze kontrolnym
1. Programowane we/wy (*programmed I/O*)
 2. Przesyłanie sterowane przerwaniami
(interrupt driven)

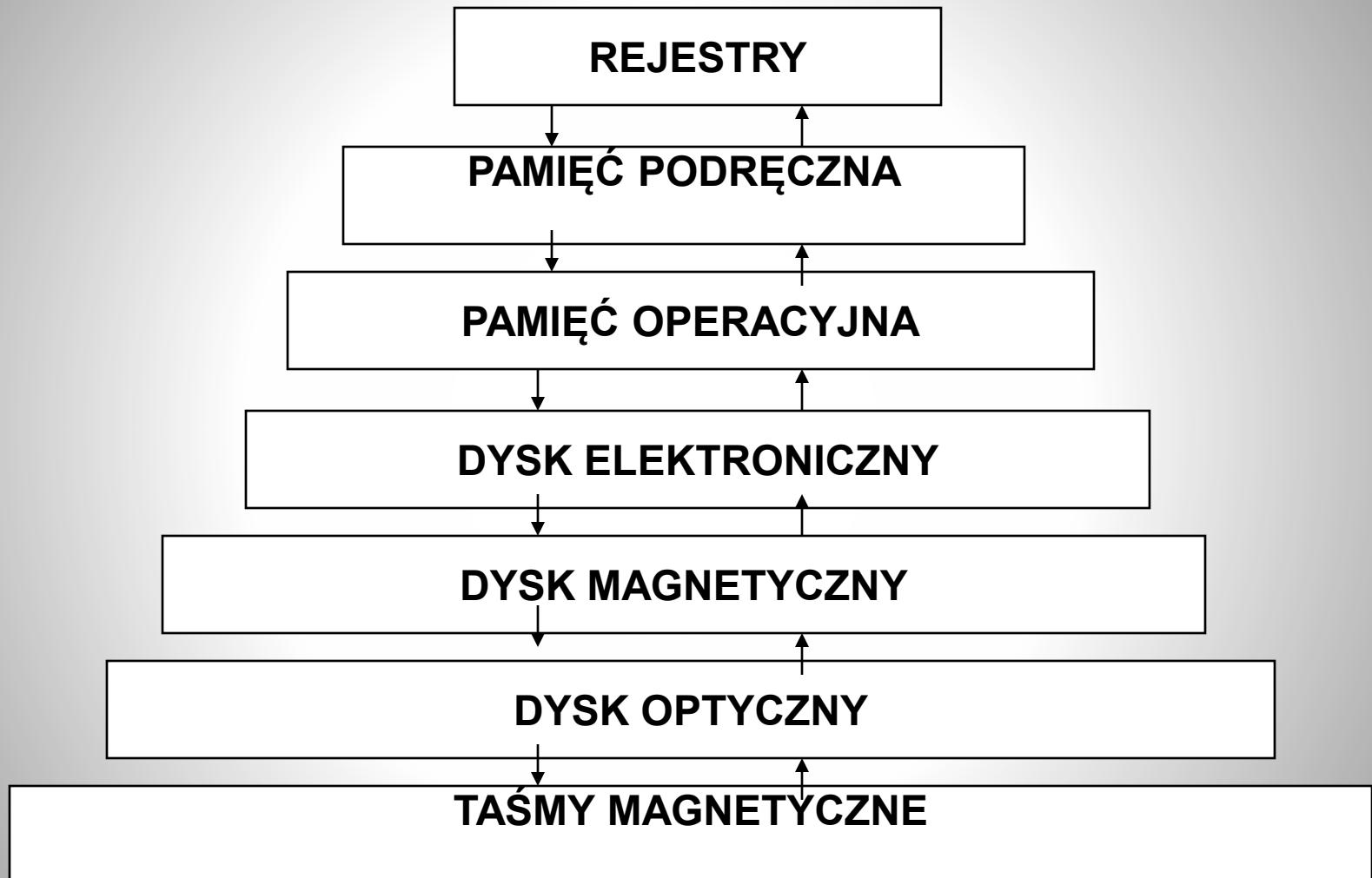
REJESTRY - wbudowane w JC (1 cykl zegara)

dostęp do PAO - za pośrednictwem szyny pamięci
wiele cykli; utykanie procesora (*stall*)

SZYBKA PAMIĘĆ POMOCNICZA (cache)

JC->cache->PAO

HIERARCHIA PAMIĘCI



Pamięć podręczna

- przechowuje informacje przejściowo
- 80-99% dostępów
- polityka zastępowania informacji
- problem zgodności pamięci podręcznej:
 - te same dane na różnych poziomach hierarchicznej struktury pamięci
 A^{++} (liczba A w pliku B na dysku)
 - kopiowanie bloku z liczbą A do PAO
 - kopiowanie do pamięci podręcznej
 - kopiowanie do rejestru wewnętrznego
 - inkrementacja w rejestrze.....

problem w środowisku wieloprocesorowym
cache coherency

PAMIĘĆ POMOCNICZA

- SO dostarcza jednolitego logicznie obrazu przechowywania informacji w oderwaniu od cech fizycznych urządzeń
- PLIK - logiczna jednostka informacji
- system plikowy:
 - zbiór plików
 - struktura katalogów

- Pamięć pomocnicza – dysk
 - Uaktualnianie informacji bez zmiany miejsca
 - Bloki adresowane bezpośrednio
- Unix katalog = plik;
- Plik – wymaga otwarcia
System plików – wymaga zamontowania
- Semantyka spójności
 - Np. unix: pisanie na otwartym pliku – natychmiast widoczne dla innych użytkowników, którzy mają otwarty ten sam plik
 - Istnieje tryb dzielenia – wspólny wskaźnik bieżącego położenia pliku

zadania SYSTEMU PLIKÓW

- Pozwala tworzyć i usuwać pliki
- Umożliwia dostęp do plików w celu czytania i pisania
- Zarządza automatycznie przestrzenią pamięci pomocniczej
- Umożliwia odwoływanie się do plików za pomocą nazw symbolicznych
- Chroni pliki przed skutkami uszkodzenia systemu
- Pozwala na wspólne korzystanie z tych samych plików
- Chroni pliki przed dostępem do nich nieuprawnionych użytkowników

plik może być otwarty przez kilku użytkowników

dwa poziomy tablic wewnętrznych:

- procesowa tablica plików otwartych w procesie
 - tryb otwarcia
 - bieżący wskaźnik do każdego pliku
 - wskaźnik do ogólnosystemowej tablicy plików otwartych
- ogólnosystemowa tablica otwartych plików
 - położenie pliku na dysku
 - daty dostępu
 - rozmiar pliku
 - licznik otwarć

Współużytkowanie plików

- blokowanie części otwartego pliku (*locking*)
- odwzorowanie części pliku w PAO
(*memory mapping*)
 - logiczne przyporządkowanie części wirtualnej przestrzeni adresowej do danej sekcji pliku
 - zamknięcie pliku równoznaczne z zapisem danych na dysk i zwolnieniem pamięci
 - wymaga mechanizmów wzajemnego wykluczania

- dyskowe operacje we/wy w jednostkach równych pojedynczemu blokowi (rekord fizyczny)
- problem zamiany rekordów logicznych na rekordy fizyczne
- fragmentacja wewnętrzna (*internal fragmentation*)

METODY DOSTĘPU

- »dostęp sekwencyjny (*sequential access*)
- »dostęp bezpośredni (*direct access*)
- »dostęp indeksowy (*index*)

STRUKTURY KATALOGOWE

- katalog jednopoziomowy
- katalog dwupoziomowy
- struktury drzewiaste
- acykliczne grafy katalogów
- graf ogólny katalogów

- ochrona
 - prawa dostępu
 - hasła

ORGANIZACJA SYSTEMU PLIKÓW

urządzenia

sterowanie we/wy (sterowniki)

podstawowy system plików (czytanie blokami)

moduł organizacji pliku (adresy logiczne -> fizyczne)

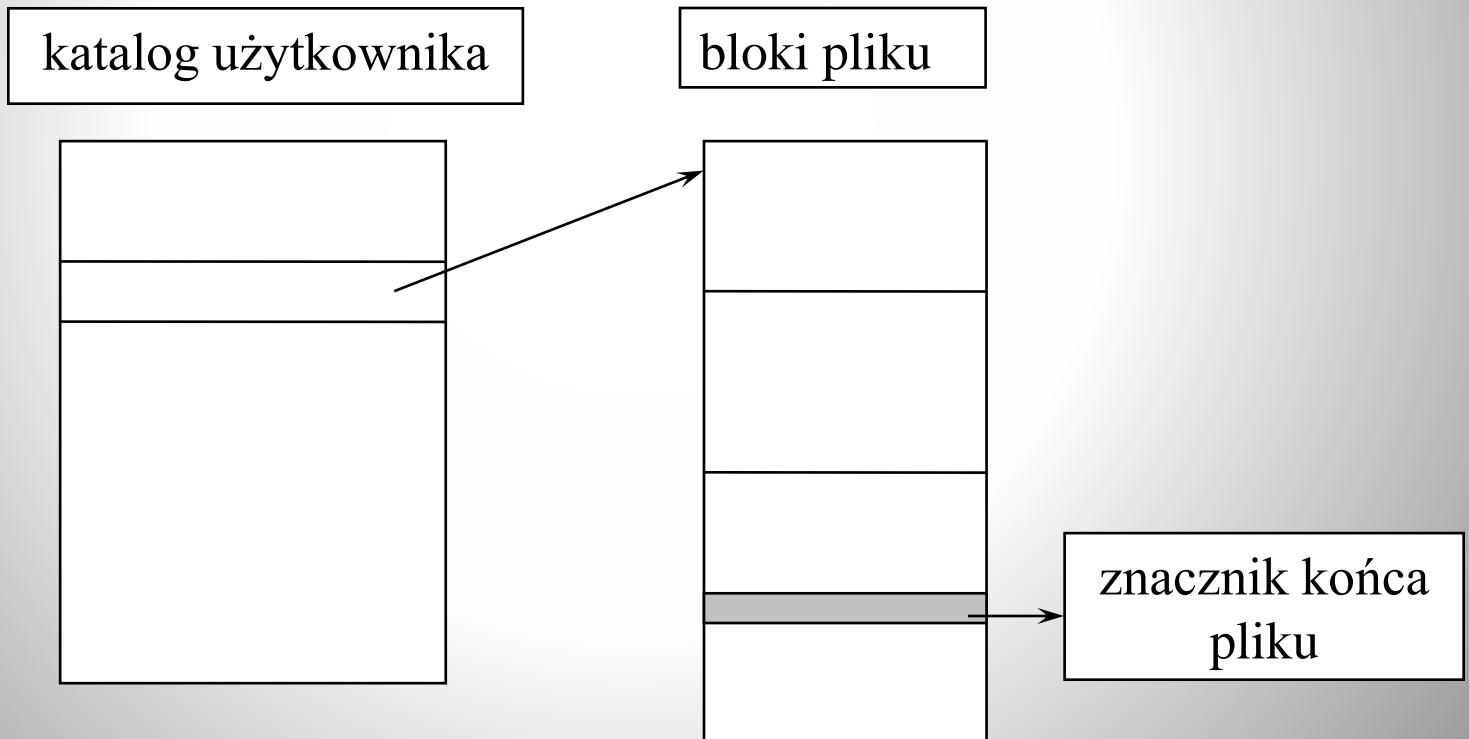
logiczny system pliku (struktura katalogów + ochrona)

ORGANIZACJA PAMIĘCI POMOCNICZEJ

(metody przydziału miejsca na dysku)

- system plików zwartych (przydział ciągły)
- łańcuch powiązanych bloków (przydział listowy)
- mapa plików (tablica alokacji)
- bloki indeksów

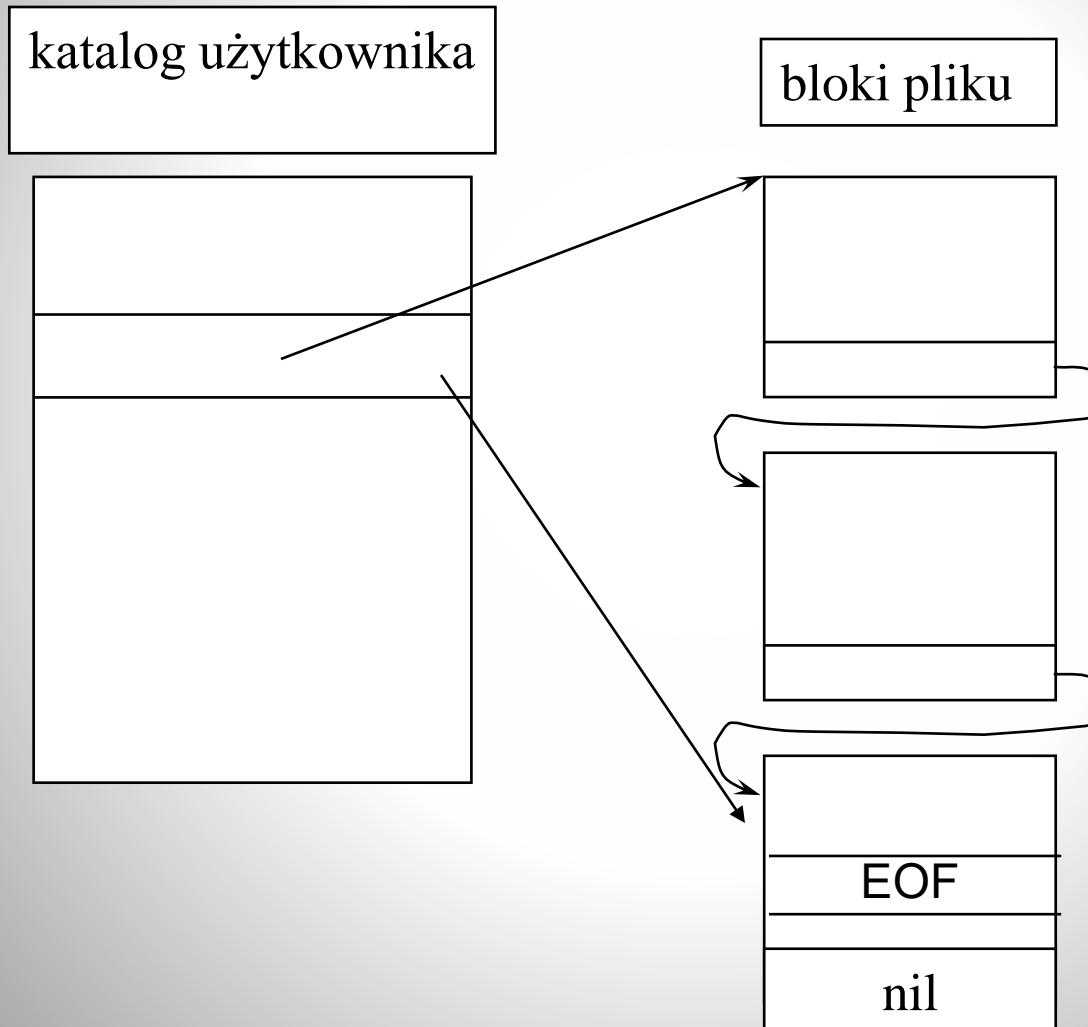
System plików zwartych



WŁAŚCIWOŚCI - ZALETY - WADY

- łatwo implementować dostęp swobodny i sekwencyjny
- trudno uniknąć fragmentacji zewnętrznej
- umożliwia najbardziej elastyczną organizację danych - zniszczenie jednego bloku powoduje tylko lokalną utratę danych
- odpowiednie do takich zastosowań jak bazy danych

Łańcuch powiązanych bloków

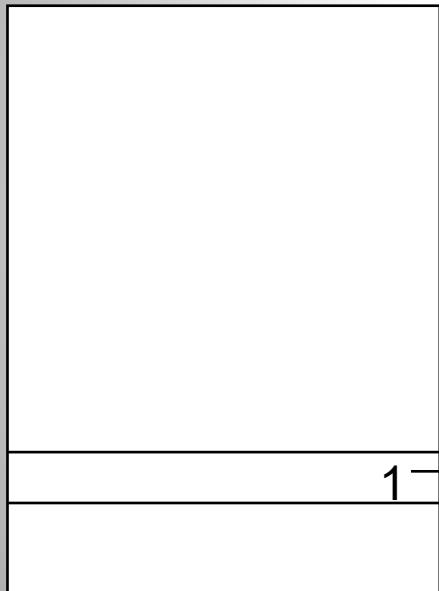


WŁAŚCIWOŚCI - ZALETY - WADY

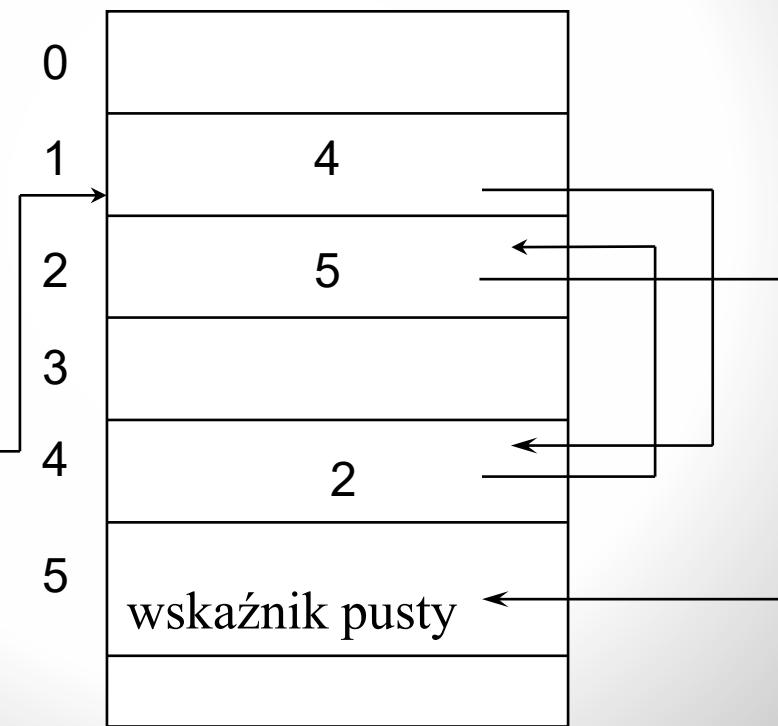
- kilka bajtów, każdego bloku w pliku służy jako wskaźnik do następnego bloku
- wada - konieczność uzyskania dużej liczby dostępów do dysku, zanim znajdzie się koniec pliku
- dostęp do pliku jest z koniecznością sekwencyjny
- metoda ta jest mało elastyczna - skutki uszkodzenia jednego bloku mogą niespodziewanie rozszerzyć się na cały system plików
- nie ma fragmentacji zewnętrznej

Mapa plików

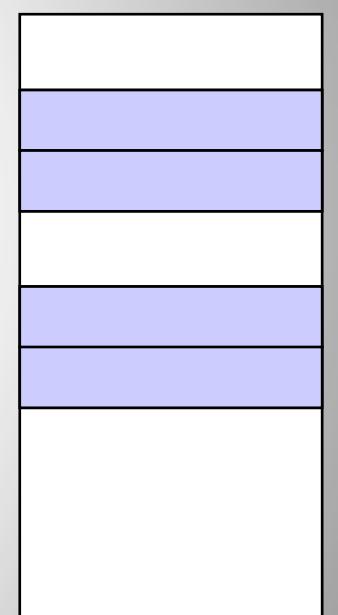
katalog użytkownika



mapa plików



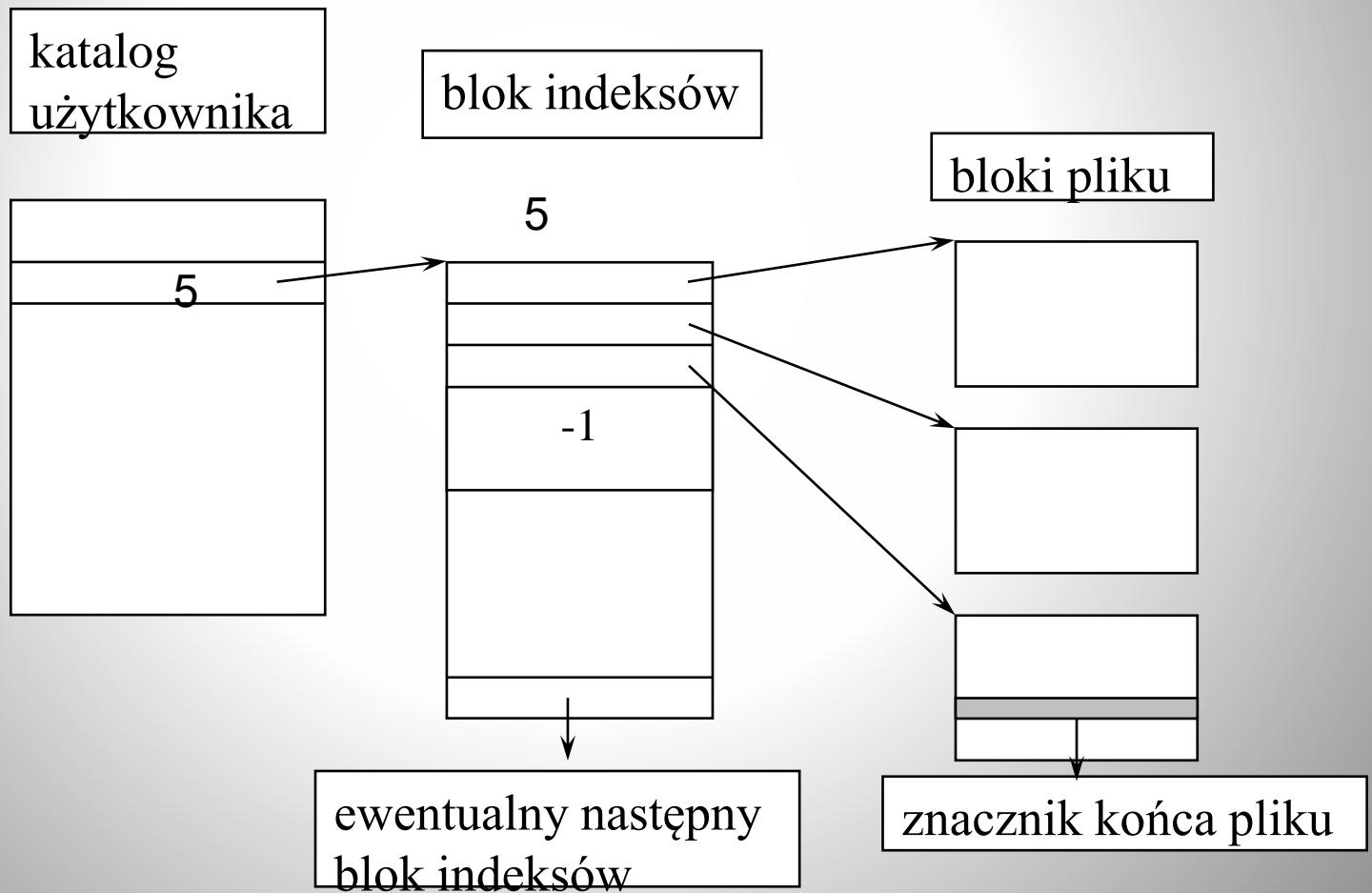
Bloki danych



WŁAŚCIWOŚCI - WADY - ZALETY

- każdy blok na dysku - pozycja w mapie
- bloki nieużywane - 0 w tablicy
- uszkodzenie mapy plików może spowodować poważne straty danych - dwie kopie mapy w różnych rejonach dysku, aby w razie awarii sprzętu nie zniszczyć wszystkich kopii
- znaczny ruch głowic dyskowych
- polepszenie czasu dostępu swobodnego

Bloki indeksów



WŁAŚCIWOŚCI - WADY - ZALETY

- wskaźniki dowiązań do każdego pliku są pamiętane w odrębnych blokach indeksów na dysku
- dla dużego pliku trzeba przeznaczyć kilka bloków indeksów
 - schemat listowy
 - indeks wielopoziomowy
 - schemat kombinowany
- brak fragmentacji zewnętrznej
- umożliwia dostęp bezpośredni

Określając obowiązujący w systemie rozmiar bloku bierze się pod uwagę poniższe kryteria:

- Strata miejsca z powodu bloków nie zapełnionych całkowicie, która zwiększa się w miarę zwiększania rozmiaru bloków (fragmentacja wewnętrzna).
- Strata miejsca związana ze wskaźnikami – im mniejsze bloki, tym więcej użytych wskaźników.
- Jednostki przesyłania danych z urządzeń zewnętrznych do pamięci głównej - rozmiar bloku powinien być wielokrotnością tej jednostki.
- Rozmiar obszaru pamięci głównej potrzebny do wykonania każdej operacji czytania lub pisania odnoszącej się do pliku.