

OBSŁUGA WEJŚCIA-WYJŚCIA

(C) IISI d.KIK PCz 2013

Systemy operacyjne

1

RÓŻNICE POMIĘDZY URZĄDZENIAMI WEJŚCIA WYJŚCIA

- × Zastosowanie
 - × przechowywanie danych, komunikacja, sterowanie
- × Prędkość
 - × sieć, dysk: 10^7 - 10^{12} , klawiatura 10^0 znaków na sekundę
- × Jednostki przesyłania
 - × bity, bajty, słowa
- × Sposób przesyłania
 - × strumień, blok
- × Reprezentacja danych
 - × różne sposoby kodowania na różnych nośnikach
- × Operacje uprzywilejowane
 - × druk na drukarce, czytanie z płyty optycznej

(C) IISI d.KIK PCz 2013

Systemy operacyjne

3

URZĄDZENIA WEJŚCIA-WYJŚCIA

najczęściej posiadają rejestry

- × stanu
- × sterowania
- × danych wejściowych
- × danych wyjściowych

(C) IISI d.KIK PCz 2013

Systemy operacyjne

5

OBSŁUGA WEJŚCIA-WYJŚCIA

- × System operacyjny musi komunikować się ze światem zewnętrznym.
- × Obsługa wejścia-wyjścia jest jednym z najzwyklejszych etapów tworzenia systemu.
 - × powodem jest konieczność stosowania ogólnych technik do różnych urządzeń zewnętrznych, o różnych parametrach technicznych i trybach pracy.

(C) IISI d.KIK PCz 2013

Systemy operacyjne

2

RÓŻNICE POMIĘDZY URZĄDZENIAMI WEJŚCIA WYJŚCIA C.D.

- × Czytanie lub pisanie
 - × klawiatura, drukarka
- × Dostęp sekwencyjny lub swobodny
 - × taśma, dysk
- × Synchroniczność lub asynchroniczność
- × Urządzenie dzielone lub na wyłączność
- × Złożoność sterowania
 - × drukarka, dysk
- × Różne przyczyny wystąpienia błędów
 - × brak papieru w drukarce, zła suma kontrolna

(C) IISI d.KIK PCz 2013

Systemy operacyjne

4

ADRESOWANIE URZĄDZEŃ WEJŚCIA-WYJŚCIA

Procesor odwołuje się do rejestrów urządzeń poprzez porty z przestrzeni:

- × we/wy
 - × instrukcje in i out
- × pamięci
 - × instrukcja mov i wszystkie tryby adresowania

(C) IISI d.KIK PCz 2013

Systemy operacyjne

6

REALIZACJE OPERACJI WEJŚCIA-WYJŚCIA

- × Programowe wejście-wyjście - odpytywanie
 - × procesor zleca obsługę i odpytuje urządzenie do jej zakończenia
- × Operacje wejścia-wyjścia sterowanie przerwami
 - × kolejne dane są transmitowane w kolejnych przerwaniach
- × Bezpośredni dostęp do pamięci (DMA)
 - × procesor zleca wykonanie transmisji kończącej się przerwaniem

(C) IISI d.KIK PCz 2013

Systemy operacyjne

7

ROZWÓJ OBSŁUGI WEJŚCIA-WYJŚCIA

- × Sterowanie bezpośrednio przez procesor
- × Procesor zleca wykonanie operacji kontrolerowi, który bezpośrednio steruje urządzeniem
- × Zastosowanie kontrolera z przerwami
- × Zastosowanie kontrolera z obsługą DMA
- × Zastosowanie procesora we/wy, który po otrzymaniu polecenia od procesora głównego wykonuje właściwy podprogram
- × Procesor we/wy staje się oddzielnym komputerem z własną pamięcią

(C) IISI d.KIK PCz 2013

Systemy operacyjne

8

WYMAGANIA STAWIANE SYSTEMOWI

- × Niezależność od kodów znaków
 - × identyczna interpretacja dla wszystkich urządzeń
- × Niezależność od urządzeń wejścia – wyjścia
 - × tego samego typu: np. który napęd, czy drukarka
 - × różnych typów: np. przesłanie na dysk lub drukarkę realizuje prawie identyczny program
- × Uzyskanie jak największej wydajności
 - × ze względu na powstawanie tzw. wąskiego gardła
- × Jednolite traktowanie urządzeń
 - × celem uproszczenie obsługi i zmniejszenie liczby błędów

(C) IISI d.KIK PCz 2013

Systemy operacyjne

9

SYSTEMOWE MECHANIZMY OBSŁUGI WEJŚCIA – WYJŚCIA

- × Kodowanie znaków w urządzeniach jest często różne i niezgodne z kodowaniem w systemie.
- × Wewnętrzny kod znaków – kod, przez który wykonuje się tłumaczenie znaków dla poszczególnych urządzeń.
- × Stosuje się tablice przekodowujące, aby znaki wewnątrz systemu były reprezentowane w sposób jednoznaczny.

(C) IISI d.KIK PCz 2013

Systemy operacyjne

10

SYSTEMOWE MECHANIZMY OBSŁUGI WEJŚCIA – WYJŚCIA

Urządzenia wirtualne

- × nazywane strumieniami, plikami, zbiorami danych
- × poprzez nie następuje komunikacja z urządzeniami fizycznymi
- × odwołując się do urządzenia najpierw należy uzyskać dostęp do niego od systemu, a po zakończeniu użytkowania należy je zwolnić.

(C) IISI d.KIK PCz 2013

Systemy operacyjne

11

SYSTEMOWE MECHANIZMY OBSŁUGI WEJŚCIA – WYJŚCIA

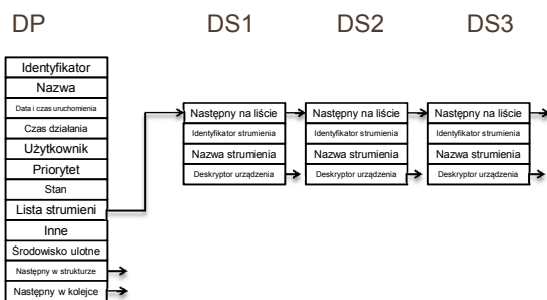
- × Programy działają na wirtualnych urządzeniach (strumieniach) zamiast na rzeczywistych.
- × Po przypisaniu urządzenia do strumienia
 - + `define output1 tape3`
 programista kieruje wyjście na strumień, a przesłaniem do urządzenia zajmuje się system operacyjny.
- × Dzięki temu programy obsługi urządzeń mogą być do siebie podobne
- × Różnice są ukryte przez system operacyjny.
- × Informacje o właściwościach urządzenia znajdują się w deskrypcji urządzenia.

(C) IISI d.KIK PCz 2013

Systemy operacyjne

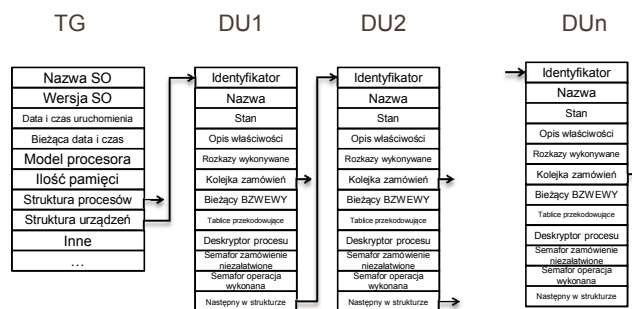
12

LISTA DESKRYPTORÓW STRUMIENI



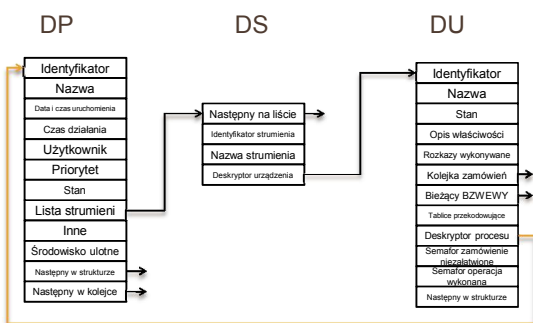
Nazwy strumieni muszą być unikatowe.

STRUKTURA URZĄDZEŃ I DESKRYPTORY URZĄDZEŃ



Struktura urządzeń wskazuje z tablicy głównej listę deskryptorów urządzeń

POWIĄZANIE DESKRYPTORÓW



PODPROGRAMY OBSŁUGI URZĄDZEŃ

- ✘ Programy te muszą być wielowejsciowe, tzn. wiele procesów może się do nich odwoływać.
- ✘ Ich podstawowe zadania to:
 - ✘ tworzenie bloków zamówień wejścia – wyjścia,
 - ✘ informowanie procesu obsługi, że został dla niego przygotowany blok zamówień.

PLIKI SPECJALNE

- ✘ **Pliki specjalne** można rozumieć jako wirtualne urządzenia.
- ✘ Wiele funkcji systemowych dla plików jest również potrzebnych w przypadku urządzeń. Po zażądaniu przydziału urządzenia (i uzyskaniu go) można wykonywać na nim operacje czytania, pisania i (być może) zmiany położenia - jak w operacjach na zwykłych plikach.
- ✘ Podobieństwo między urządzeniami wejścia-wyjścia i plikami może być tak duże, że wiele systemów operacyjnych (UNIX, MS-DOS) łączy je w jedną strukturę plików - urządzeń.

PLIKI SPECJALNE

- ✘ W systemach typu UNIX pliki specjalne do obsługi urządzeń znajdują się w katalogu /dev .
- ✘ W systemie MS-DOS można wyróżnić pewne urządzenia, które są traktowane jak pliki.
 - ✘ port równoległy LPT,
 - ✘ port szeregowy COM,
 - ✘ konsola CON.

PROCEDURY WEJŚCIA-WYJŚCIA

Proces użytkownika zgłasza zamówienia do systemu korzystając z funkcji:

OPWEWY(strumień, tryb, rozmiar, lokalizacja, semafor)

gdzie:

- + OPWEWY – nazwa procedury systemowej
- + strumień – numer strumienia we/wy
- + tryb – tryb dostępu (wysłanie danych, wysunięcie papieru)
- + rozmiar – rozmiar danych
- + lokalizacja – źródło lub miejsce przeznaczenia danych
- + semafor – adres semafora, do którego zostanie przesłany sygnał zakończenia operacji

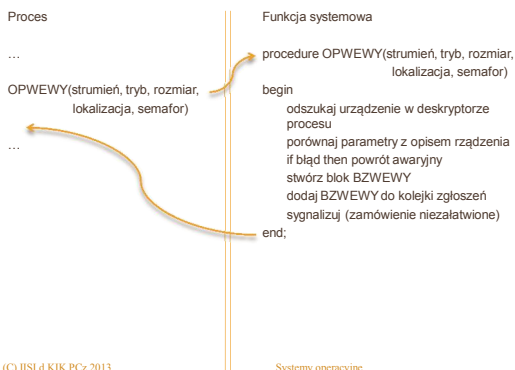
PROCEDURY WEJŚCIA-WYJŚCIA

- ✘ Procedura OPWEWY jest wielowejściowa (może być wywołana przez kilka procesów naraz).

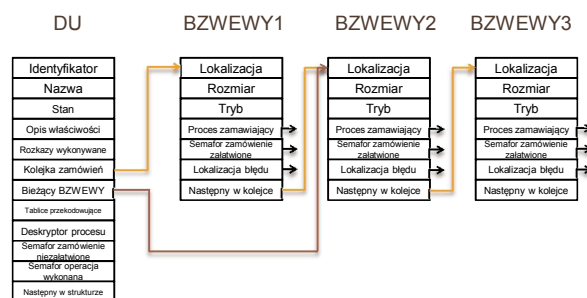
Przebiega ona w następujących krokach:

- ✘ odszukanie urządzenia w deskrytorze procesu
- ✘ sprawdzenie parametrów wywołania procedury z opisem urządzenia
- ✘ zainicjowanie obsługi urządzenia
 - + utworzenie bloku zamówień wejścia-wyjścia (BZWEWY) i dodanie go do kolejki zamówień urządzenia
 - + wysłanie sygnału do procesu obsługi urządzenia informującego o nowym zamówieniu
 - ✘ sygnalizuj (zamówienie niezalutwione)

PROCEDURA WEJŚCIA-WYJŚCIA



DESKRYTOR URZĄDZENIA I KOLEJKA ZAMÓWIEŃ



Kolejka zamówień może być dowolnie zorganizowana.

PROCESY OBSŁUGI URZĄDZEŃ

- ✘ Obsługą zamówień umieszczonych w kolejce urządzenia oraz powiadamianiem procesów-nadawców o wykonaniu operacji we/wy zajmuje się proces obsługi urządzeń.
- ✘ Każde urządzenie posiada osobny proces, jednak podobieństwa pozwalają użyć wspólnych programów.
- ✘ Różnice spowodowane są odmiennymi cechami konkretnych urządzeń.

PROCES OBSŁUGI URZĄDZENIA WYJŚCIOWEGO

while true do {obsługa wyjścia}

begin

czekaj (zamówienie niezalutwione)
 wybranie BZWEWY z kolejki zamówień
 wydobycie szczegółów zamówienia
 przeslij dane z podanej lokalizacji
 tłumacz znaki

zainicjowanie operacji we-wy
 czekaj (operacja wykonana)
 if bład then umieść informację o błędzie
 sygnalizuj (zamówienie zalutwione)
 usunięcie BZWEWY

end;

Dla transmisji poprzez kanał, można zastąpić odpdytywaniam!

PODPROGRAM OBSŁUGI PRZERWANIA

Transmisja przez kanał kończy się przerwaniem:

POP:

lokalizacja deskryptora urządzenia
sygnalizuj(operacja wykonana)

(C) IISI d.KIK PCz 2013

Systemy operacyjne

25

PROCES OBSŁUGI URZĄDZENIA WEJŚCIOWEGO

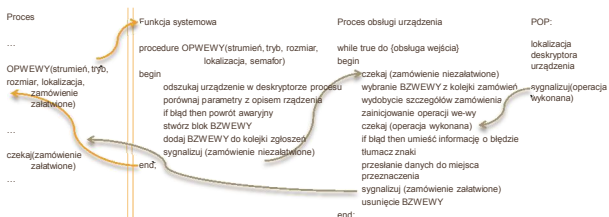
```
while true do {obsługa wejścia}
begin
  czekaj (zamówienie niezalutowane)
  wybranie BZWEWY z kolejki zamówień
  wydobycie szczegółów zamówienia
  zainicjowanie operacji we-wy
  czekaj (operacja wykonana)
  if błąd then umieść informację o błędzie
  tłumacz znaki
  przesłanie danych do miejsca przeznaczenia
  sygnalizuj (zamówienie zalutowane)
  usunięcie BZWEWY
end;
```

(C) IISI d.KIK PCz 2013

Systemy operacyjne

26

CAŁOŚCIOWA PROCEDURA WEJŚCIA-WYJŚCIA

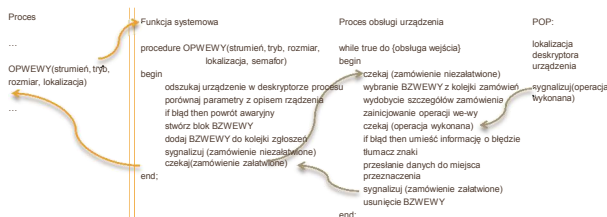


(C) IISI d.KIK PCz 2013

Systemy operacyjne

27

CAŁOŚCIOWA PROCEDURA WEJŚCIA-WYJŚCIA



(C) IISI d.KIK PCz 2013

Systemy operacyjne

28

OBSŁUGA WEJŚCIA-WYJŚCIA

może odbywać się na dwa sposoby:

- ✘ proces zamawiający może wykonywać się równolegle z operacją we/wy
 - + sam musi zadbać o własne wstrzymanie przed odwołaniem się do informacji pochodzącej z wywołanej operacji; musi on wykryć zakończenie operacji we/wy
- ✘ procedura we/wy, będąca częścią systemu operacyjnego, wstrzymuje proces do zakończenia operacji we/wy
 - + semafor zamówienie zalutowane, będący semaforem lokalnym powoduje niezbędne opóźnienie; programista aplikacji nie ponosi odpowiedzialności za wykrycie zakończenia operacji we/wy, ale traci możliwość szybszego wykonywania procesu.

(C) IISI d.KIK PCz 2013

Systemy operacyjne

29

BUFOROWANIE WEJŚCIA-WYJŚCIA

- ✘ Proces powtarzający wielokrotnie operacje na tym samym strumieniu jest ciągle wstrzymywany i przełączany w stan niewykonywalny i w związku z tym traci się dużo czasu, ponieważ operacje te są bardzo kosztowne.
- ✘ Należy spowodować, aby operacje we/wy odbywały się, w miarę możliwości, bez spowalniania procesu.
- ✘ Rozwiązaniem tego problemu może być przesyłanie danych z wyprzedzeniem lub opóźnieniem, czyli buforowanie.

(C) IISI d.KIK PCz 2013

Systemy operacyjne

30

BUFOROWANIE WEJŚCIA-WYJŚCIA

Buforowanie można podzielić na:

- ✘ buforowanie wejścia
- ✘ buforowanie wyjścia
- ✘ buforowanie podwójne
 - ✘ dwa bufory pracują na zmianę
- ✘ buforowanie wielokrotne
 - ✘ usuwa zatory w dostępie do urządzeń we/wy

(C) IISI d.KIK PCz 2013

Systemy operacyjne

31

BUFOROWANIE WEJŚCIA-WYJŚCIA

Podczas używania operacji buforowanych

OPBUFWEWY(strumień, tryb, lokalizacja)

w inny sposób wykonuje się procedury we/wy.

- ✘ Operacja wejściowa obsługuje zamówienia poprzez odczytanie z bufora kolejnego dostępnego elementu.
 - ✘ Procedura generująca blok zamówień we/wy, wysyła sygnał do procesu obsługi urządzenia celem dostarczenia większej ilości danych wejściowych, wywoływana jest jedynie w przypadku opróżnienia bufora.
- ✘ Operacja wyjściowa obsługuje zamówienia poprzez zapisanie do bufora kolejnego elementu.
 - ✘ Procedura generująca blok zamówień we/wy, wstrzymuje proces jedynie w przypadku przepełnienia bufora wyjściowego.
- ✘ Operacja we/wy wraz z procesem obsługi urządzenia działają tak, jak procesy producenta i konsumenta.

(C) IISI d.KIK PCz 2013

Systemy operacyjne

32

URZĄDZENIA PLIKOWE

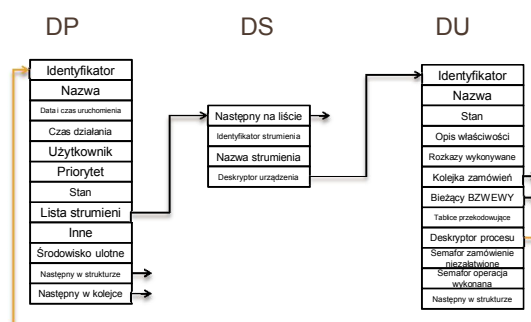
- ✘ Założenie, iż do zapisu ze strumienia wystarczy jedynie identyfikator urządzenia docelowego, nie jest spełnione w przypadku urządzeń o dostępie swobodnym
 - ✘ dysk twardy, płyta optyczna.
- ✘ W takiej sytuacji, niezbędne jest również podanie obszaru, gdzie dany strumień się znajduje lub skąd ma zostać pobrany.
- ✘ W tego rodzaju nośnikach każdy obszar nazywany jest plikiem
 - ✘ posiada on dowolny rozmiar oraz opisującą go jednoznaczną nazwę.

(C) IISI d.KIK PCz 2013

Systemy operacyjne

33

POWIĄZANIE DESKRYTORÓW – URZĄDZENIE NIEPLIKOWE

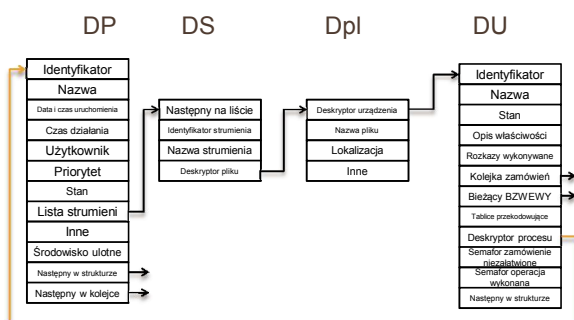


(C) IISI d.KIK PCz 2013

Systemy operacyjne

34

POWIĄZANIE DESKRYTORÓW – URZĄDZENIE PLIKOWE



Odwolując się do urządzenia plikowego pomiędzy deskryptorem strumienia a deskryptorem urządzenia pojawia się deskryptor pliku.

(C) IISI d.KIK PCz 2013

Systemy operacyjne

35

URZĄDZENIA PLIKOWE

- ✘ W przypadku używania urządzenia plikowego, strumień należy powiązać z nazwą pliku, nie jak poprzednio z samym urządzeniem.

```
define input disk2:data7
```

- ✘ Polecenie w takiej postaci definiuje strumień o nazwie input powiązany z plikiem nazwie data7 umieszczony na dysku disk2.

(C) IISI d.KIK PCz 2013

Systemy operacyjne

36

SPOOLER

- ✘ Urządzeniem niepodzielnym nie można przydzielić kilku procesów równocześnie.
- ✘ Procesy chcące korzystać z urządzeń niepodzielnych muszą czekać na zwolnienie kolejki.
- ✘ W danym momencie może być dużo zgłoszeń do urządzenia, podczas gdy później może ono stać bezczynnie.
- ✘ Wskazane jest równomierne rozłożenie zamówień i przeciwdziałanie powstawaniu wąskich gardeł.
- ✘ Rozwiązaniem może być zastosowanie pracy pośredniej. Dane będą przesyłane na nośnik pośredni, jakim najczęściej jest dysk.
- ✘ Następnie proces systemowy przesyła dane z plików na dysku na urządzenie; nazywa się on spoolerem (simultaneous peripheral operation on-line).

(C) IISI d.KIK PCz 2013

Systemy operacyjne

37

SPOOLER

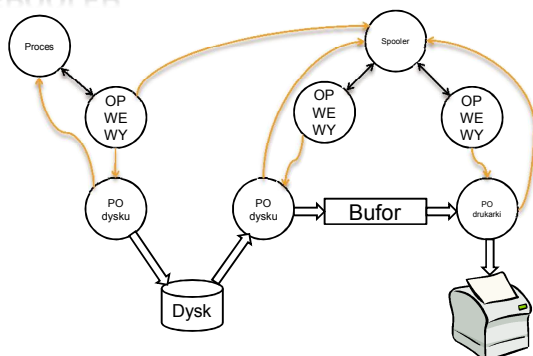
- ✘ Zasadę działania spoolera można przedstawić dla drukarki.
- ✘ Każdemu procesowi otwierającemu strumień drukarki przydziela się plik dyskowy pełniący funkcje wirtualnej drukarki.
- ✘ Po zamknięciu strumienia, plik ten jest dodany do kolejki drukarki.
- ✘ Oszczędzając na czasie oczekiwania na drukarkę, korzystający z niej proces użytkowy może szybciej powracać do kolejki procesów gotowych.

(C) IISI d.KIK PCz 2013

Systemy operacyjne

38

SPOOLER



(C) IISI d.KIK PCz 2013

Systemy operacyjne

39

SPOOLER

Działanie spoolera:

- ✘ sprawdzenie czy jest coś w kolejce
- ✘ wybranie pliku i otwarcie go
- ✘ wykonywanie transmisji do urządzenia do czasu napotkania końca pliku
- ✘ zasygnalizowanie zakończenia przetwarzania pliku

(C) IISI d.KIK PCz 2013

Systemy operacyjne

40

SPOOLER

- ✘ Spooler może wybierać odpowiednie pliki według zasady – FIFO lub dowolnego innego algorytmu.
- ✘ Pozwala przyspieszyć działanie procesów.
- ✘ Czyni zasób niepodzielny podzielnym.
- ✘ Zabiera zasoby dyskowe, czas procesora i pamięć.
- ✘ Nie można go stosować w systemach czasu rzeczywistego (operacje we/wy muszą w tym przypadku być wykonywane natychmiast).

(C) IISI d.KIK PCz 2013

Systemy operacyjne

41

PULA

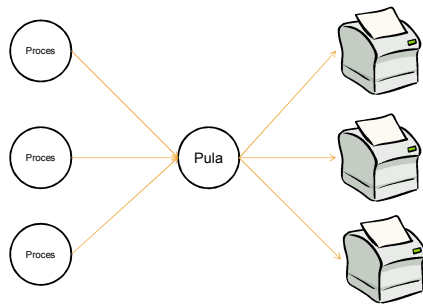
- ✘ Łączy kilka identycznych urządzeń w jedno.
- ✘ Procesy transmitują dane do puli.
- ✘ Pula przesyła dane do wybranego przez siebie urządzenia.
- ✘ Zastosowanie puli zwiększa przepustowość urządzeń.
- ✘ Proces nie musi wybierać sam urządzenia i narażać się na oczekiwanie w przypadku zajętości danego urządzenia.
- ✘ Pula może współpracować ze spoolerem.

(C) IISI d.KIK PCz 2013

Systemy operacyjne

42

PULA DRUKAREK

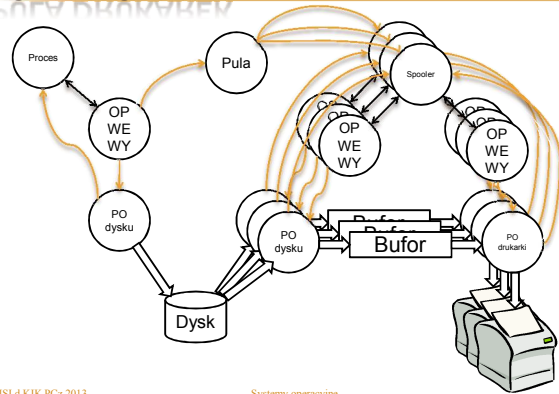


(C) IISI d.KIK PCz 2013

Systemy operacyjne

43

PULA DRUKAREK



(C) IISI d.KIK PCz 2013

Systemy operacyjne

44

MACIERZE DYSKOWE

Macierze dyskowe RAID (redundant arrays of inexpensive disks) są stosowane w celu:

- ✘ zwiększenia wydajności i
- ✘ poprawienia niezawodności przechowywania danych

(C) IISI d.KIK PCz 2013

Systemy operacyjne

45

MACIERZE DYSKOWE

Macierze dyskowe RAID występują w kilku wersjach:

- ✘ RAID0
- ✘ RAID1
- ✘ RAID2
- ✘ RAID3
- ✘ RAID4
- ✘ RAID5
- ✘ RAID6

oraz jako połączenia powyższych poziomów

(C) IISI d.KIK PCz 2013

Systemy operacyjne

46

MACIERZE DYSKOWE

Macierze dyskowe mogą być realizowane w sposób:

- ✘ sprzętowy
 - ✘ korzystając ze specjalnych kontrolerów (drogich!)
- ✘ programowy
 - ✘ przez oprogramowanie systemu operacyjnego

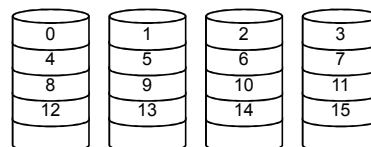
(C) IISI d.KIK PCz 2013

Systemy operacyjne

47

MACIERZE DYSKOWE – RAID0

- ✘ Przestrzeń dyskowa podzielona jest na paski.
- ✘ Paski są umieszczane cyklicznie na kolejnych dyskach.
- ✘ Zwiększenie szybkości dostępu do dysku
- ✘ Brak zabezpieczenia danych na wypadek awarii dysku



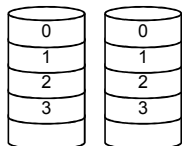
(C) IISI d.KIK PCz 2013

Systemy operacyjne

48

MACIERZE DYSKOWE – RAID1

- Przestrzeń dyskowa podzielona jest na paski.
- Paski są kopiowane na kolejnych dyskach (kopia lustrzana).
- Zwiększenie szybkości odczytu dostępu do dysku.
- Zabezpieczenie danych na wypadek awarii dowolnego dysku.
- Połowa pojemności dostępna dla systemu.



(C) IISI d.KIK PCz 2013

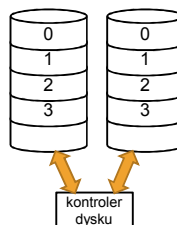
Systemy operacyjne

49

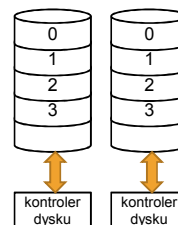
MACIERZE DYSKOWE – RAID1

Może występować w dwóch odmianach:

Mirroring



Duplexing



(C) IISI d.KIK PCz 2013

Systemy operacyjne

50

MACIERZE DYSKOWE – RAID2

- Paski są bardzo małe.
- Stosuje się kod nadmiarowy (korygujący) (f) pozwalający skorygować jeden błąd i wykryć drugi.



(C) IISI d.KIK PCz 2013

Systemy operacyjne

51

MACIERZE DYSKOWE – RAID3

- Parzystość z przeplotem bitów.
- Paski są bardzo małe.
- Błędy wykrywają same dyski na podstawie sum CRC.
- Stosuje się parzystość pozwalającą skorygować błąd na jednym dysku.



(C) IISI d.KIK PCz 2013

Systemy operacyjne

52

MACIERZE DYSKOWE – RAID4

- Parzystość z przeplotem bloków.
- Paski są wielkości bloków.
- Błędy wykrywają same dyski na podstawie sum CRC.
- Stosuje się parzystość pozwalającą skorygować błąd na jednym dysku.



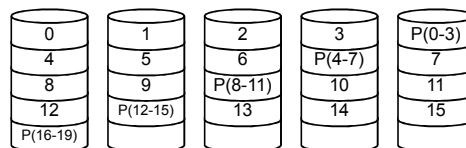
(C) IISI d.KIK PCz 2013

Systemy operacyjne

53

MACIERZE DYSKOWE – RAID5

- Parzystość rozproszona z przeplotem bloków.
- Paski są wielkości bloków.
- Błędy wykrywają same dyski na podstawie sum CRC.
- Stosuje się parzystość pozwalającą skorygować błąd na jednym dysku.



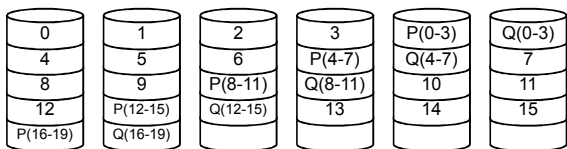
(C) IISI d.KIK PCz 2013

Systemy operacyjne

54

MACIERZE DYSKOWE – RAID6

- ✘ Parzystość rozproszona z przeplotem bloków i podwójną nadmiarowością.
- ✘ Paski są wielkości bloków.
- ✘ Błędy wykrywają same dyski na podstawie sum CRC.
- ✘ Stosuje się dwie funkcje pozwalające skorygować błąd na dwu dyskach.



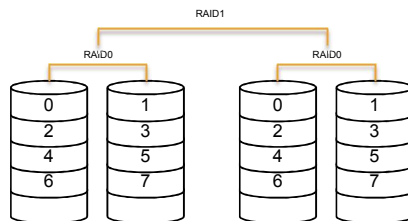
(C) IISI d.KIK PCz 2013

Systemy operacyjne

55

MACIERZE DYSKOWE – RAID 0+1

Lustrzanie pracują dwa zestawy dysków w RAID0



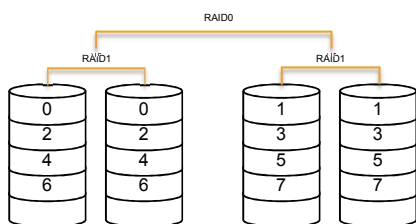
(C) IISI d.KIK PCz 2013

Systemy operacyjne

56

MACIERZE DYSKOWE – RAID 1+0

Dwa zestawy dysków lustrzanych pracują w RAID0



(C) IISI d.KIK PCz 2013

Systemy operacyjne

57

MACIERZE DYSKOWE – RAID 5+

- ✘ RAID 5+0 - Dwa zestawy RAID5 pracują w RAID0
- ✘ RAID 5+1 - Dwa zestawy RAID5 pracują w RAID1

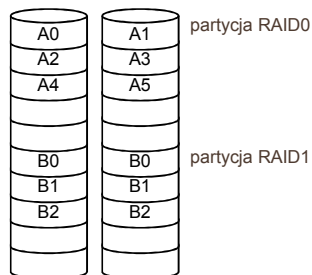
(C) IISI d.KIK PCz 2013

Systemy operacyjne

58

MACIERZE DYSKOWE – MATRIX RAID

Część przestrzeni dyskowej pracuje w RAID0, a część w RAID1.



(C) IISI d.KIK PCz 2013

Systemy operacyjne

59

PAMIĘĆ PODRĘCZNA DYSKU

Realizowana jest:

- ✘ w pamięci operacyjnej systemu
 - ✘ w postaci bufora o zmiennym rozmiarze
 - ✘ zarządzana przez system operacyjny
- ✘ jako dodatkowa pamięć wbudowana w napęd dysku
 - ✘ rozmiar jest ustalony przez producenta dysku
 - ✘ zarządza nią kontroler wbudowany do dysku twardego.

(C) IISI d.KIK PCz 2013

Systemy operacyjne

60