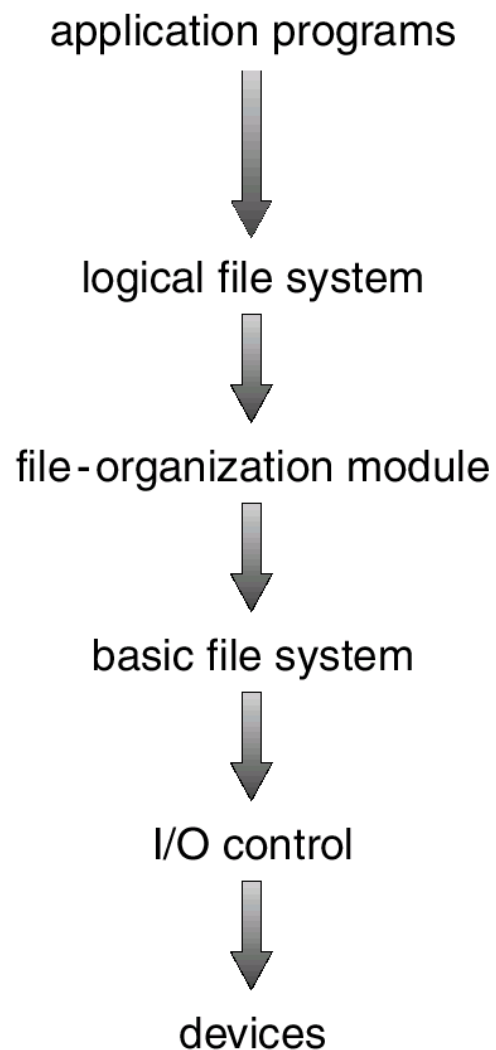


Implementacja systemu plików

Struktura systemu plików

- Struktura pliku:
 - logiczna jednostka magazynowania
 - kolekcja związanych informacji
- System plików rezyduje w pamięci pomocniczej (zwykle dysk)
- Ma organizację warstwową



- I/O control — sterowniki urządzeń i procedury obsługi przerwań
- basic file system — ogólne polecenia czytania/pisania poszczególnych bloków
- organizacja pliku — tłumaczenie numerów logicznych bloków na adresy fizyczne
- logical file system — na podstawie nazwy pliku daje dostęp do bloku kontrolnego pliku (sprawdza również uprawnienia)
- *Blok kontrolny pliku* — struktura zawierająca wszystkie niezbędne informacje o pliku

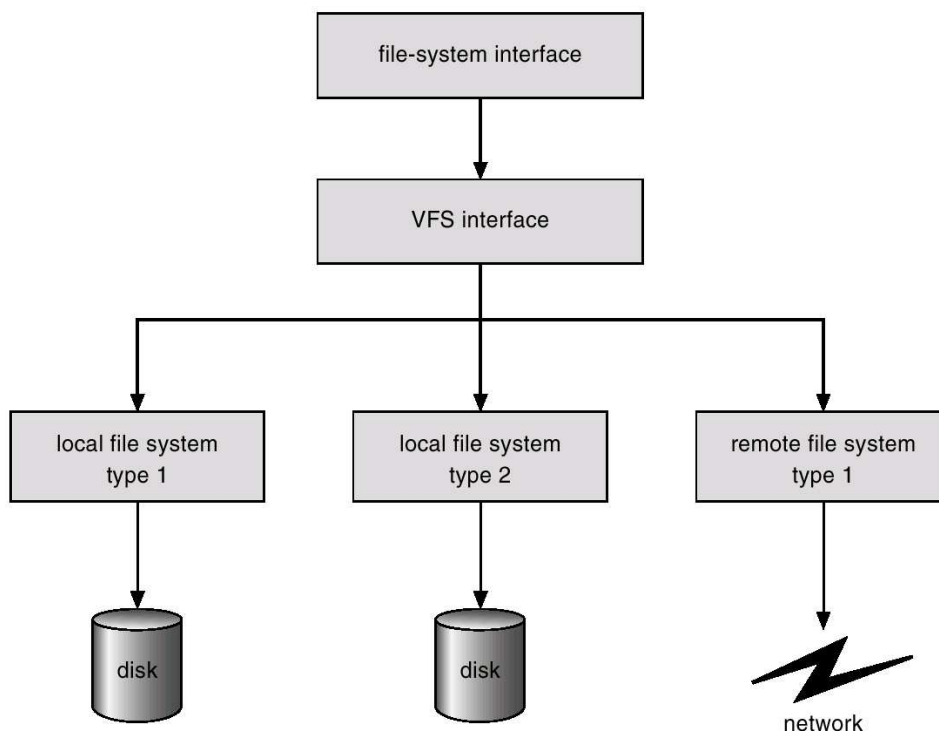
file permissions
file dates (create, access, write)
file owner, group, ACL
file size
file data blocks

Wirtualny system plików

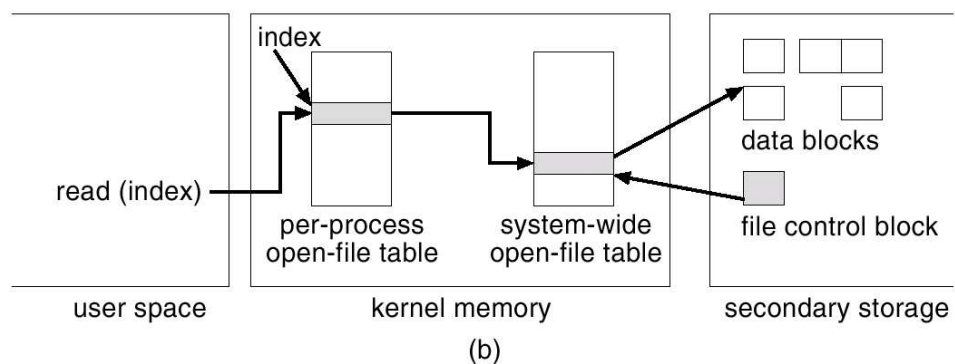
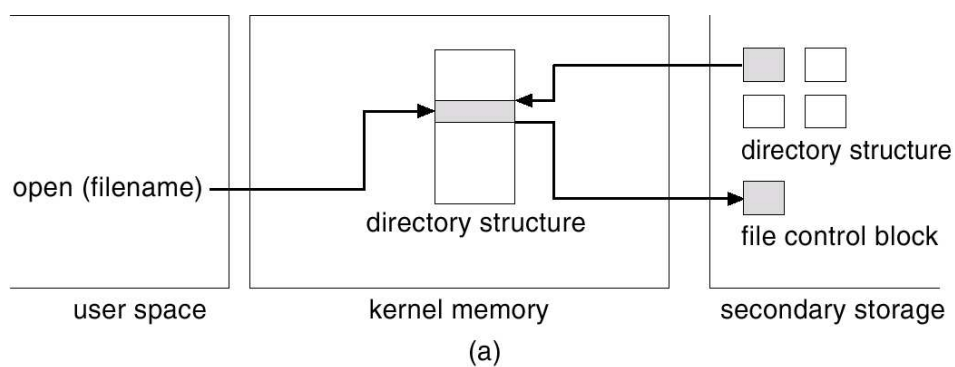
Dodatkowa warstwa (wprowadzona po raz pierwszy przez SUN-a przy implementacji NFS-a)

- obiektowe podejście do implementacji systemu plików
- pozwala na używanie identycznego interfejsu użytkownika do różnych typów systemów plików (nie tylko lokalnych)

- użytkownik faktycznie odwołuje się do wirtualnego systemu plików



Struktury file systemu w pamięci operacyjnej



- Otwieranie pliku — część a)
- Operacja czytania — część b)

Implementacja katalogu

Efektywność implementacji ma istotny wpływ na działanie systemu plików

- Lista liniowa nazw plików
 - łatwe programowanie
 - dość kosztowna obsługa — wyszukiwanie liniowe
- Tablica haszująca — lista liniowa z dostępem przez funkcję mieszającą
 - szybki dostęp
 - konieczność rozwiązania kolizji
 - stała wielkość — zależność funkcji mieszającej od tej wielkości

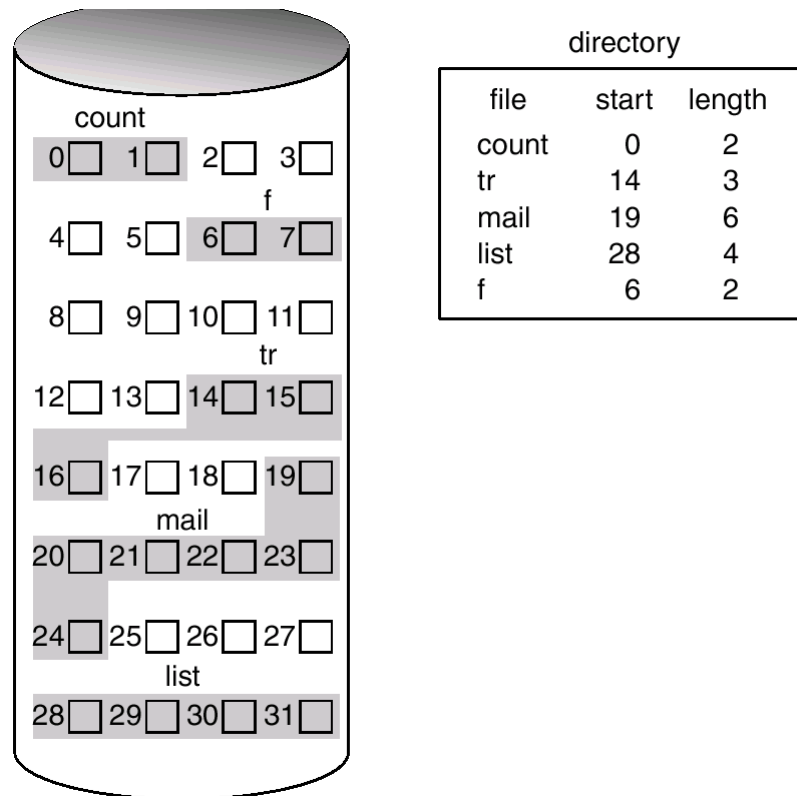
Metody przydziału miejsca na dysku

Jak bloki dyskowe są przydzielane dla plików

Trzy podstawowe schematy:

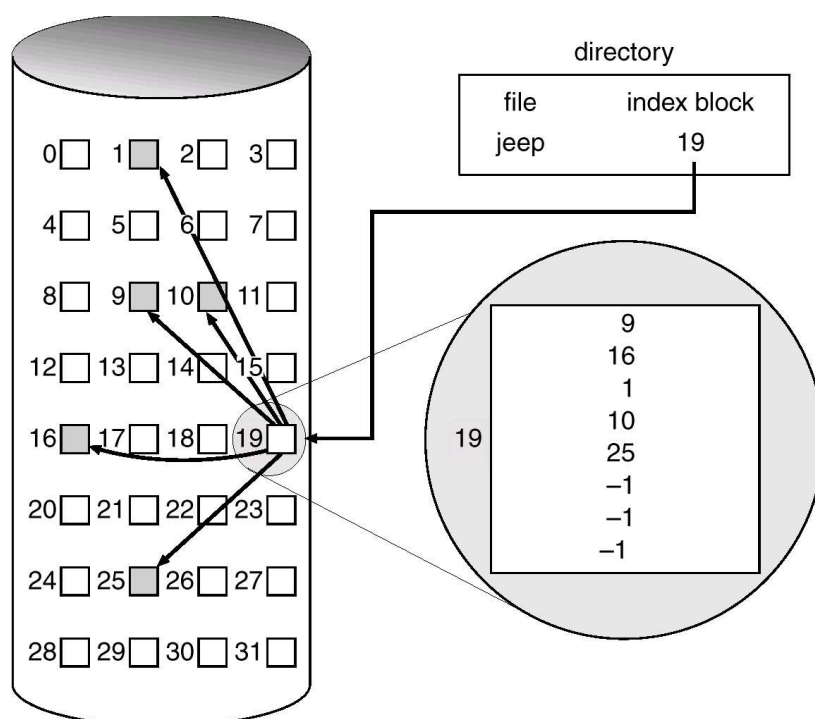
- przydział ciągły
- przydział listowy
- Przydział indeksowy
- Przydział ciągły — plik zajmuje kolejne bloki na dysku
 - prostota: początek i długość
 - fragmentacja zewnętrzna — algorytmy jak w przypadku PAO

- problemy z rozszerzaniem
- określenie miejsca potrzebnego dla pliku, przenoszenie w inne miejsce w przypadku braku miejsca

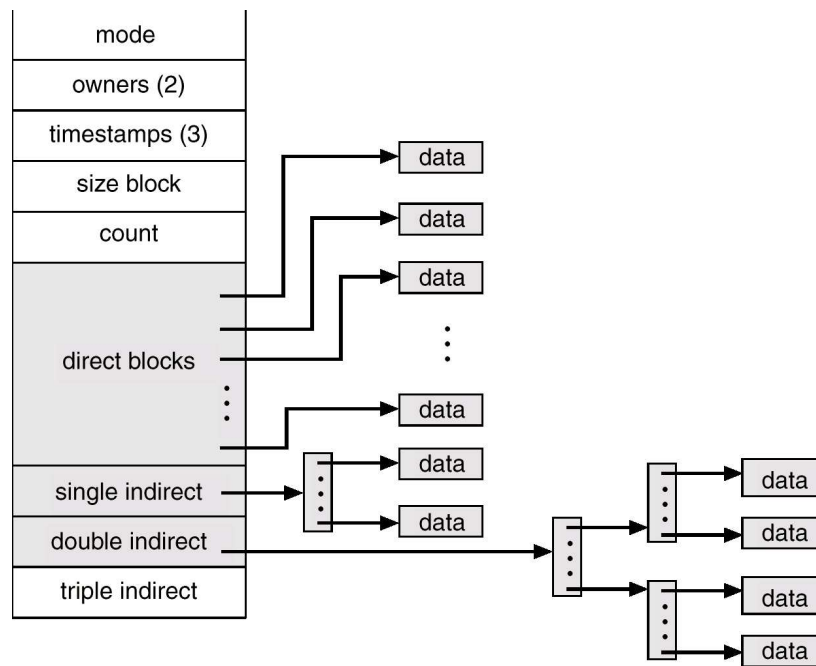


- Modyfikacja schematu polegająca na przydzielaniu dla pliku pewnej liczby obszarów ciągłych (extent-based system)
- Przydział listowy — bloki powiązane w listę wskaźnikową (w ustalonym miejscu bloku wskaźnik do następnego)
 - prostota
 - brak fragmentacji
 - problemy z dostępem bezpośrednim
 - strata miejsca na wskaźniki — grona
 - wrażliwość na błędy i awarie — listy dwukierunkowe

- dostęp bezpośredni
- brak fragmentacji
- wymaga bloku indeksowego
- jak duży powinien być blok indeksowy
 - * dla małych plików cały blok jest zbędny
 - * dla dużych jest zbyt mały — indeks wielopoziomowy albo lista bloków



- Schemat mieszany (Unix/Linux) — w i-węźle pliku 13 pozycji (Linux) na wskaźniki do bloków pliku:
 - pierwsze 10 bloków adresowane bezpośrednio
 - następne przez jeden blok pojedynczo pośredni — indeks jednopoziomowy
 - dalsze przez blok podwójnie pośredni — indeks dwupoziomowy
 - jeszcze dalsze przez blok potrójnie pośredni - indeks trzypoziomowy

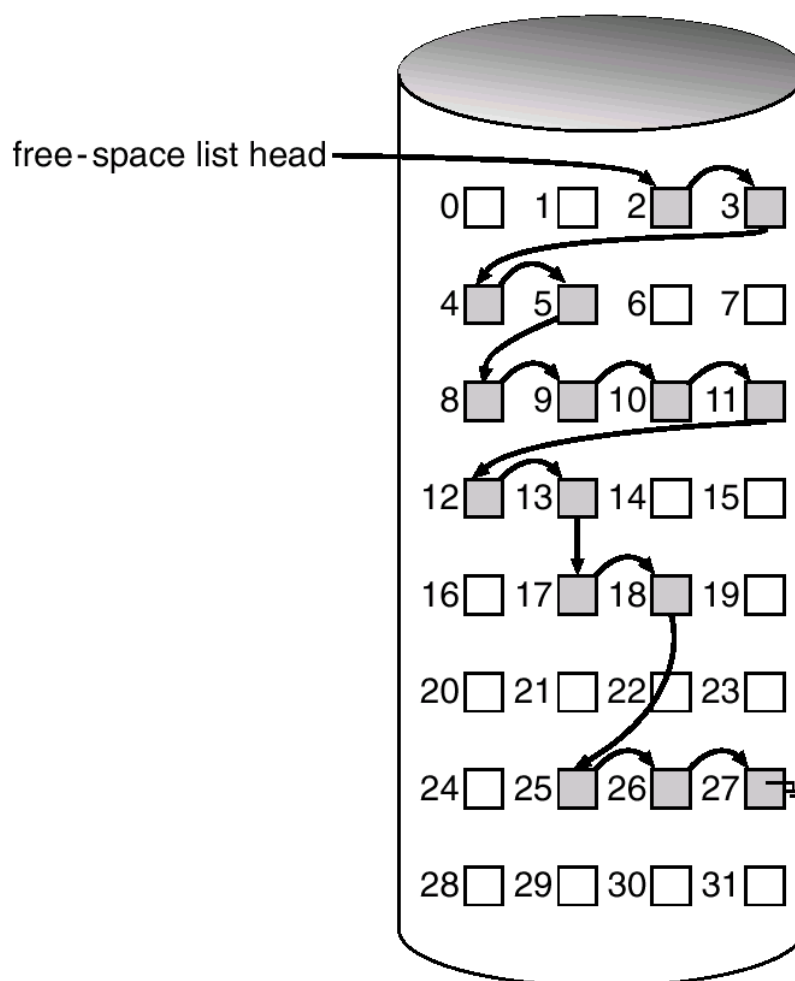


Zarządzanie wolnymi blokami

System musi wiedzieć, które bloki na dysku są wolne.

- Mapa bitowa — jeden bit dla każdego bloku dyskowego
 - np.: 1 – blok wolny, 0 – blok zajęty
 - ustalenie pozycji pierwszej jedynki:
 $(\text{liczba bitów w słowie}) * (\text{liczba wyzerowanych słów}) + \text{pozycja pierwszej jedynki w niezerowym słowie}$
 - problemy z wielkością mapy — słaba efektywność, jeśli mapa poza pamięcią
 - Problem zapewnienia zgodności danych na dysku i w pamięci — odpowiednia kolejność zmian
- Lista powiązana wolnych bloków — przeglądanie kosztowne
- Lista grup wolnych bloków — w bloku są numery $n - 1$ wolnych bloków a ostatni zawiera numer kolejnego elementu listy

- Zlicznie — grupy sąsiadujących wolnych bloków
- Jak to jest w systemach z FAT?



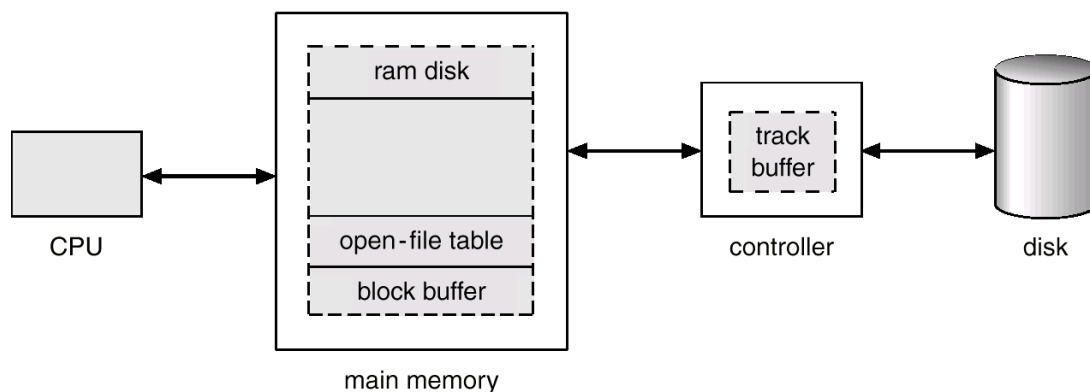
Efektywność i wydajność

- Efektywność wykorzystania miejsca zależy od:
 - algorytmów przydziału i algorytmów obsługi katalogów
 - typów przechowywanych informacji — zmiana zapisu katalogowego wymaga uaktualniania bloku na dysku,
- Wydajność
 - pamięć podręczna bloków dyskowych — w niektórych systemach

cała wolna pamięć traktowana jest jako pula buforów

- strategie zarządzania buforami — np. *free-behind* i *read-ahead*
- RAM dyski w pamięci

- Różne lokalizacje pamięci podręcznych dla dysku

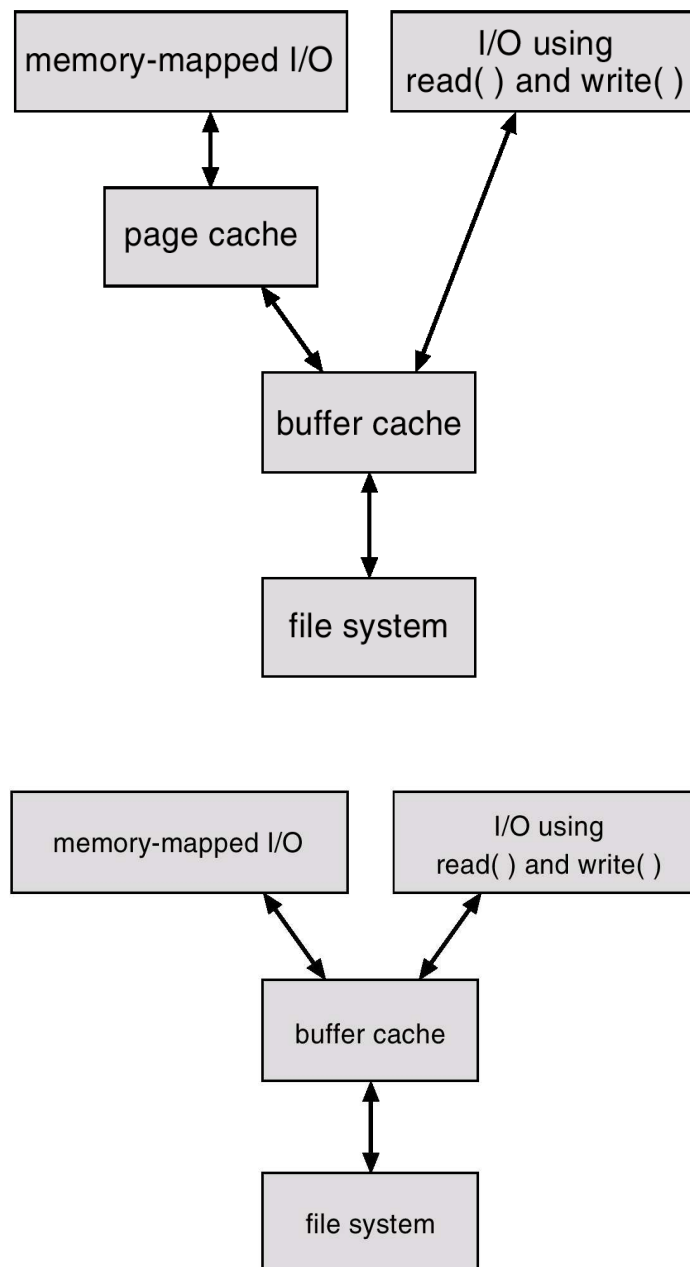


Pamięć podręczna stron

- Przechowuje strony wykorzystując mechanizmy pamięci wirtualnej
- Operacje we/wy na plikach mapowanych do pamięci korzystają z pamięci podręcznej stron
- Procedury we/wy systemu plików korzystają z pamięci buforów dyskowych

Zunifikowana pula buforów

- Wykorzystuje tę samą pamięć podręczną



Rekonstrukcja

Awaria systemu nie powinna doprowadzać do utraty danych albo ich niespójności.

- Sprawdzanie spójności — porównanie danych w strukturze katalogu i blokach dyskowych i próba usunięcia niezgodności

- Składowanie i odtwarzanie — składowanie pełne i przyrostowe (odpowiedni schemat)

Log structured file systems

- Systemy typu log structured (journaling) traktują każdą modyfikację systemu plików jako transakcję (w rozumieniu bazodanowym)
- Każda transakcja zapisywana jest w logu i traktowana jest jako zatwierdzona natychmiast po zapisaniu — faktyczna zmiana nie musi być już wprowadzona
- Transakcje zapisane w logu są wprowadzane asynchronicznie do systemu plików i usuwane z logu
- W razie awarii pozostałe w logu transakcje są realizowane