

# Systemy operacyjne 2016

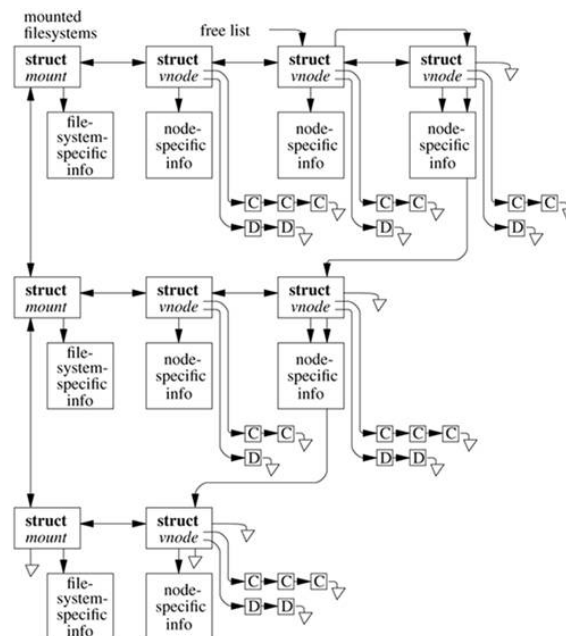
## Lista zadań nr 9

Na zajęcia ?

Należy przygotować się do zajęć czytając następujące rozdziały książek:

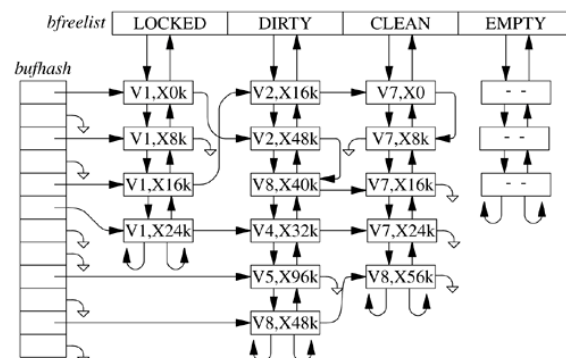
- Stallings (wydanie siódme): 12
- Tanenbaum (wydanie czwarte): 4.1 – 4.5

**UWAGA!** W trakcie prezentacji rozwiązań należy zdefiniować i wyjaśnić pojęcia, które zostały oznaczone **wytluszczoną** czcionką.



*FreeBSD*: Połączenia między strukturami jądra przechowującymi vnode.

D – bufor brudny, C – bufor czysty.



*FreeBSD*: Struktura do buforowania bloków.

V – numer vnode, X – pozycja względem początku pliku.

**Zadanie 1.** Podaj wywołania systemowe jądra *Linux* służące do operowania na **plikach**. Jakie **atrybuty** mogą być skojarzone z plikiem<sup>1</sup>? Opisz różnicę w rozumieniu pojęcia **typu pliku** przez system operacyjny i użytkownika. Jak rozwiązano dostęp do konfiguracji i funkcjonalności specyficznej dla urządzenia – np. zmiany rozdzielczości ekranu, wysunięcia tacki napędu optycznego?

**Zadanie 2.** Opisz pobieżnie następujące metody przydziału przestrzeni dyskowej dla pliku: **ciągła**, **listowa**, **indeksowana**, **i-węzeł**. Jakie struktury danych są wykorzystywane do:

- dostępu do wybranych fragmentów pliku?
- przechowywania informacji o wolnych obszarach?

Jaki jest narzut pamięciowy utrzymywania tych struktur? Podaj górne ograniczenie na ilość operacji dyskowych wymaganych do wczytania dowolnego bloku dysku.

**Zadanie 3.** Rozważmy struktury danych używane przez metody przydziału z poprzedniego zadania. Jakie są konsekwencje ich częściowego uszkodzenia? Jakie błędy można naprawić i w jaki sposób? Na czym polega **defragmentacja** przestrzeni dyskowej? Opisz techniki zapobiegania fragmentacji – tj. stosowanie obszarów (ang. *extent*) i odroczone przydzielanie bloków (ang. *delayed allocation*).

**Zadanie 4.** Rozważmy hierarchiczną strukturę **katalogów** – czym różni się **ścieżka absolutna**, **relatywna** i **znormalizowana**? Podaj wywołania systemowe jądra *Linux* służące do operowania na katalogach. Wymień **atrybuty**, które mogą być powiązane z katalogiem. Jak implementowane są **dowiązania twarde** (ang. *hard link*), a jak **dowiązania symboliczne** (ang. *soft link*)?

**Zadanie 5.** W jaki sposób system plików z i-węzłami przechowuje katalogi? Jak przechowywać długie nazwy plików, aby nie marnotrawić miejsca na dysku? Zaproponuj dyskową strukturę danych do przechowywania dużych katalogów (kilkadziesiąt tysięcy plików). Chcemy z użyciem minimalnej ilości dostępów do dysku:

- pobrać i-węzeł pliku z danego katalogu,
- usunąć wpis z katalogu.

**Wskazówka:** Pomyśl o hybrydowej strukturze danych posiadającej cechy B-drzewa i tablicy mieszającej.

**Zadanie 6.** W jakim celu nowoczesne systemy operacyjne implementują interfejs **wirtualnego systemu plików**? Wyjaśnij znaczenie struktury *vnode* w systemie *FreeBSD*. Proces chce zmodyfikować zawartość pliku o zadanym numerze **deskryptora** – jak system operacyjny ma znaleźć funkcję, która wykonuje to zadanie? Rozważmy **punkt montażowy** systemu plików na przenośnym dysku *USB*. Co się stanie, jeśli użytkownik usunie fizycznie urządzenie z zamontowanym systemem plików?

**Zadanie 7.** Na podstawie rysunku z poprzedniej strony opowiedz jak można zorganizować **pamięć podręczną dla bloków** (ang. *block cache*). Chcemy szybko dowiadywać się czy dany fragment pliku jest w pamięci operacyjnej. Jednocześnie w jakimś wątku jądra chcemy uspoźniać brudne bufor z pamięcią drugorzędną. Kiedy leniwe zapisywanie zawartości brudnych buforów jest niebezpieczne?

**Zadanie 8.** Opisz uniksowy **system kontroli dostępu** do plików. Czemu każdy plik ma **właściciela** i **grupę**? Podaj znaczenie bitów *rxw* dla plików i katalogów. Opisz znaczenie dodatkowych bitów uprawnień: *set-uid*, *set-gid*, *sticky*. Przedstaw algorytm sprawdzania uprawnień dostępu do określonej ścieżki pliku przez zadanego użytkownika. Podaj przykład, w którym standardowy system kontroli dostępu jest zbyt ograniczony i należy użyć **ACL** (ang. *access control list*).

---

<sup>1</sup>Zajrzyj do podręcznika systemowego *stat(2)*, *chattr(1)* i *xattr(7)*.

**Zadanie 9.** Wymień kroki niezbędne do realizacji poniższych operacji i wyjaśnij jak awaria zasilania może przyczynić się do naruszenia **spójności** struktur systemu plików lub zawartości plików.

1. Usunięcie pliku z katalogu.
2. Dopisanie kilku bloków na koniec otwartego pliku.

Pokaż jak wykorzystać **księgowanie** do zapobiegania awariom systemu plików. Czym różni się **księgowanie metadanych** od **księgowania danych**? Dlaczego operacje składowane w **dzienniku** muszą być **idempotentne**?

**Zadanie 10 (bonus).** Księgowanie nie chroni przed usterkami sprzętowymi oraz ludzką głupotą – w tym celu korzystamy z **kopii zapasowych**. Jakie są różnice między **kopią pełną**, **przyrostową** i **różnicową**? Jaki wcześniej poznany mechanizm jest używany do tworzenia **migawek**? Czy mogą one pełnić rolę kopii zapasowych? Gdzie wykorzystuje się **wersjonujące systemy plików**?

**Zadanie 11 (bonus).** Opisz i porównaj funkcje systemu *Linux* do **blokowania plików i rekordów** – odpowiednio `flock(2)` i `fcntl(2)`. Czy systemy uniksowe implementują **blokady obowiązkowe** (ang. *mandatory locks*) czy **doradcze** (ang. *advisory locks*)? Zaproponuj strukturę danych do przechowywania informacji o blokadach rekordów.