Systemy operacyjne 2016

Lista zadań nr 10

Na zajęcia 19 stycznia 2017

Należy przygotować się do zajęć czytając następujące rozdziały książek:

- Stallings (wydanie siódme): 12
- Tanenbaum (wydanie czwarte): 4.3 4.5

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

Zadanie 1. 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 2. 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 3. Podaj sposób organizacji katalogów w systemie plików z i-węzłami. 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.

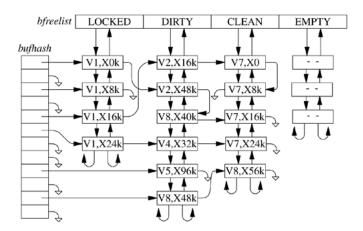
Zadanie 4. Wymień kroki niezbędne do realizacji poniższych operacji i opisz scenariusz, w którym awaria zasilania przyczynia się do naruszenia **spójności** struktur systemu plików lub zawartości plików.

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

Pokaż jak dzięki **księgowaniu** (ang. *journalling*) można zapobiec usterkom systemu plików. Opisz format **dziennika**. Czemu operacje składowane w dzienniku muszą być **idempotentne**? Czym różni się **księgowanie metadanych** od **księgowania danych**?

Zadanie 5. Księgowanie nie chroni przed usterkami sprzętowymi oraz ludzką głupotą – w tym celu korzystamy z **kopii zapasowych** (ang. *backup*). Jakie są różnice między **kopią pełną**, **przyrostową** i **różnicową**? Podaj wcześniej poznany mechanizm używany do tworzenia **migawek** (ang. *snapshot*). Czy mogą one pełnić rolę kopii zapasowych? Gdzie wykorzystuje się **wersjonujące systemy plików**?

Zadanie 6. Na podstawie poniższego rysunku wyjaśnij organizację **pamięci podręcznej dla blo-ków** (ang. *block cache*). Chcemy szybko dowiadywać się czy dany fragment pliku jest w pamięci operacyjnej. Jednocześnie w jakimś podprogramie jądra chcemy uspójniać brudne bufory z pamięcia drugorzędną. Kiedy leniwe zapisywanie zawartości brudnych buforów jest niebezpieczne?



FreeBSD: struktura danych do buforowania bloków

V – numer vnode, X – pozycja względem początku pliku (wielokrotność rozmiaru strony) bufhash – tablica mieszająca do szybkiej lokalizacji bloków LOCKED – lista bloków przypiętych do pamięci operacyjnej, EMPTY – lista bloków wolnych

Zadanie 7. W jakim celu stosuje się **partycjonowanie** dysków? Porównaj dwa schematy organizacji partycji **MBR** (ang. *Master Boot Record*) i **GPT** (ang. *GUID Partition Table*). Opowiedz o ograniczeniach pierwszego schematu i jak drugi je rozwiązuje. Rozważ m.in. liczbę, rozmiar i typ partycji, atrybuty, odporność na uszkodzenia.

Zadanie 8. Wymień główne zadania **program rozruchowego** (ang. *boot loader*). Biorąc za przykład program GRUB (w wersji 2) prześledź etapy od uruchomienia kodu z sektora rozruchowego do uruchomienia jądra. Uzasadnij podział kodu programu GRUB trzy części i wyjaśnij czemu musi on korzystać z funkcji **oprogramowania wbudowanego** (ang. *firmware*), np. BIOS lub UEFI. Do czego służy **linia poleceń jądra**? Czemu oprócz jądra program rozruchowy ładuje również **ramdysk**?

Zadanie 9 (bonus). Wyjaśnij zasadę działania wirtualnych urządzeń blokowych na podstawie linuksowego device mapper¹. Opisz następujące warstwy translacji: mirror, crypt, cache, linear, striped. Jak składać ze sobą warstwy translacji mając do dyspozycji:

- dwa dyski magnetyczne o pojemności 2TB i dysk półprzewodnikowy o pojemności 32GB, aby utworzyć niezawodne i szybkie urządzenie blokowe o pojemności 2TB?
- dwa dyski magnetyczne o pojemności 1TB, aby utworzyć zaszyfrowane urządzenie blokowe o pojemności 2TB?

Porównaj to programowe rozwiązanie ze sprzętowym kontrolerem RAID.

¹https://www.kernel.org/doc/Documentation/device-mapper/