Waldemar Gieracki

SKN „SYSTEM"

SYMULACYJNY MODEL BADANIA EFEKTYWNOŚCI WYBRANYCH ORGANIZACJI ZBIORÓW W PAMIĘCI DYSKOWEJ

Wprowadzenie

W szybkim i burzliwym rozwoju informatyki i sprzętu komputerowego dążącym do zwiększania mocy obliczeniowej maszyn i szybkości przetwarzania, niemałą rolę odgrywa problem masowych pamięci zewnętrznych. Np. taśma magnetyczna dawniej zaliczana do pamięci szybkich dziś jest zbyt wolna w stosunku do szybkości maszyn i realizowanego przetwarzania. Coraz bardziej powszechne stają się bardzo szybkie urządzenia pamięci dyskowej. O ile jednak posługiwanie się pamięcią na pakietach dyskowych wymaga wyższych kwalifikacji i doświadczenia o tyle jednak oferuje większą efektywność przetwarzania i operowania na danych. Poznanie wszystkich możliwości posługiwania się pamięcią dyskową jest niezbędne, gdyż w przeciwnym razie może spowodować wzrost kosztów lub niską efektywność przetwarzania. W ramach działalności w Studenckim Kole Naukowym „SYSTEM" postanowiono zająć się bliżej pamięcią zewnętrzną na pakietach dyskowych. Postanowiono sprawdzić możliwość przetwarzania zbiorów w pamięci dyskowej. Chodziło o porównanie efektywności poszczególnych rodzajów operacji zbiorów w pamięci dyskowej w typowym przetwarzaniu i konkretnym języku programowania. Pod pojęciem typowego przetwarzania rozumie się przebiegi zakładania zbioru oraz aktualizacji tzn. wprowadzenie, modyfikacje i kasowanie rekordów w zbiorze.

Wybór metody rozwiązania problemu:

Opracowano wstępnie 3 metody: metodę porównań czasowych, metodę badania na zbiorach rzeczywistych i metodę symulacyjną.

Metoda porównań czasowych polega na pomiarze czasów przebiegu różnych programów które wykorzystują zbiory na pakietach dyskowych.

Mierzony czas całkowity Tc miałby postać:

$$Tc = Tm + Tp$$

gdzie: Tc – czas całkowity przebiegu

Tm – czas związany z manipulowaniem na zbiorach (odczyt i zapis)

Tp – czas realizacji pozostałych funkcji programu nie związanych ze zbiorami w pamięci dyskowej.

Metoda ma jednak duże zasadnicze wady:

1) W ramach mierzonego czasu Tc mieściłby się oprócz czasu działania na zbiorach Tm czas Tp który zniekształciłby wyniki. Ustalenie samego czasu Tm jest niemożliwe ze względu na brak procedur pomiaru czasu Tm w programach użytkowych.
2) Różne programy użytkowe były ze sobą nieporównywalne ze względu na wielkość, realizowane funkcje, typy organizacji zbiorów itp.

Metoda badania na rzeczywistych zbiorach miała polegać na sporządzeniu kopii kilku wybranych zbiorów na potrzeby dokonania pomiarów czasu przetwarzania. Zbiory te miały jednak z reguły różną strukturę danych i różne organizacje dostępów i wymagałyby dużego nakładu pracy przy opracowaniu programów testowo-pomiarowych.

Ostatecznie jako metodę badawczą zaproponowano metodę symulacyjną.

Plan eksperymentu symulacyjnego.

W I etapie opracowano założenia i przeprowadzono badanie efektywności następujących organizacji pamięci dyskowej programowanych w języku FORTRAN:

a) zbiorów seryjnych
b) zbiorów o dostępie bezpośrednim nazywanych czasami zbiorami nieseryjnymi
c) zbiorów tworzonych przez standardowe podprogramy dla przesyłania tablic lub elementów tablic do i z pamięci dyskowej.

Przyjęto podział na zbiory z danymi w postaci zredagowanej tzn. znakowo– dziesiętnej kodowanej binarnie i niezredagowanej binarnej. Zbiór seryjny tworzy się umieszczając kolejno rekordy jeden za drugim bez pozostawienia wolnego miejsca. Kolejność rekordów w zbiorze jest taka jak kolejność ich wprowadzenia. W zbiorach seryjnych nie można dokonywać wstawiania ani rozszerzania rekordów. Ze zbioru seryjnego można pobierać rekordy tylko w takim porządku w jakim zostały umieszczone.

Do zbiorów o dostępie bezpośrednim należą zbiory sekwencyjne, które tworzy się umieszczając rekordy w obszarze zbioru w kolejności wyznaczonej przez ten klucz. Możliwe jest selektywne pobieranie rekordów ze zbioru sekwencyjnego jeśli zapewni się system indeksowania zbioru poprzez indeksowanie bezpośrednie lub tablice indeksowe. W naszym przypadku wykorzystywane będą zbiory z samoindeksowaniem.

Zbiorem o dostępie bezpośrednim jest też zbiór algorytmizowany losowo, który tworzy się stosując tzw. algorytm adresacji, pozwalający na powiązanie wartości rekordu i numeru porcji, w której jest on umieszczony. W celu umieszczenia rekordu w zbiorze oblicza się wg zadanego algorytmu numer porcji i umieszcza go w tej porcji.

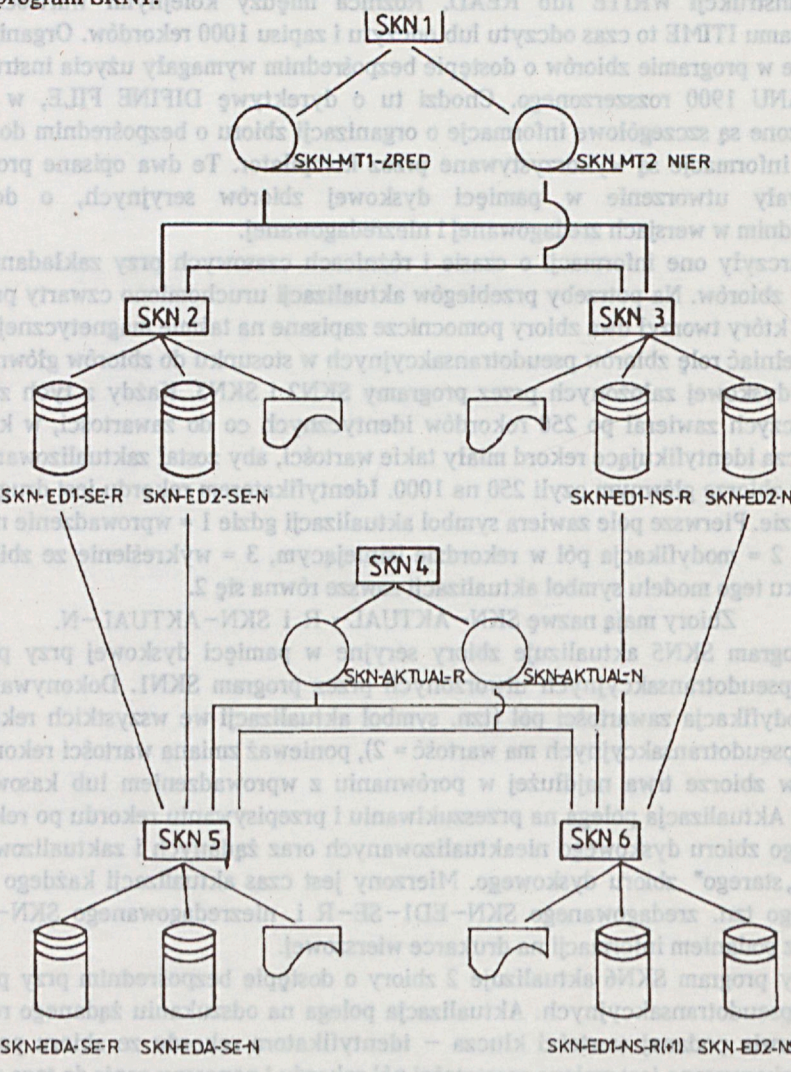
Realizacja eksperymentu symulacyjnego.

W eksperymencie bierze udział 8 programów napisanych w języku FORTRAN 1900. Ogólny schemat przebiegu przedstawiony jest na rys. 1.

Pierwszy z serii, program SKN1 generuje za pomocą multiplikatywnego generatora liczb pseudolosowych dane i tworzy dwa zbiory po 1000 rekordów w wersji zredagowanej i niezredagowanej i zapisuje je na taśmie magnetycznej. Rekordy są stałej długości, a każdy składa się z 8 pól różnego typu. Pierwsze pole traktowane jest jako klucz – identyfikator rekordu i zawiera numer kolejny od 1 – 1000. Pozostałe

zawierają dane typu TEXT liczby typu całkowitego i rzeczywistego. Rekord w wersji zredagowanej ma długości 87 znaków czyli 23 słowa. Rekord niezredagowany ma długość 16 słów + słowo licznikowe[1]. Różnica w długości wynika z używania przez system ODRA 1300 różnej jednostki długości przy zapisie na nośnik magnetyczny (taśmowy lub dyskowy). W zapisie zredagowanym jednostką długości jest 1 znak 6-bitowy, a przy niezredagowanym 2 słowa maszynowe dla jednej danej. Widać tu możliwość oszczędzania miejsca na nośniku magnetycznym ale pod pewnymi warunkami, o których wspomnę w ostatniej części.

Drugi program SKN2 jest programem pomiarowym. Zakłada on 2 zbiory seryjne w pamięci dyskowej przepisując dane wzorcowe ze zbiorów na taśmach utworzonych przez program SKN1.



[1] Słowo w EMC ODRA 1304 ma długość 24 bity i zawiera 4 znaki (1 znak ma 6 bitów). Słowo licznikowe długości 24 bitów dopisywane jest przez system maszyny i zawiera informacje o długości rekordu w słowach - wykorzystywane jest przez system automatycznego programowania pamięci zewnętrznych.

Pierwszy zbiór seryjny w pamięci dyskowej o nazwie SKN–ED1–SE–R ma postać zredagowaną a dane przepisane ze zredagowanej taśmy SKN–MT1–ZRED. Drugi zbiór dyskowy SKN–ED2–SE–N ma dane niezredagowane przepisane z taśmy niezredagowanej SKN–MT1–NIER. Dokonywany jest pomiar czasu zakładania każdego zbioru osobno i drukowane są informacją na drukarce.

Trzeci program SKN3 zakłada zbiór zredagowany oraz zbiór niezredagowany o dostępie bezpośrednim typu DIRECT pobierając dane ze zbiorów na taśmach magnetycznych. Dokonywany jest jak poprzednio pomiar czasów zakładania zbiorów. Pomiaru czasu dokonywano w programach przy użyciu podprogramu ITIME który był wywoływany dwukrotnie: przed pierwszym użyciem instrukcji READ i po ostatnim użyciu instrukcji WRITE lub READ. Różnica między kolejnymi wartościami z podprogramu ITIME to czas odczytu lub odczytu i zapisu 1000 rekordów. Organizacja i tworzenie w programie zbiorów o dostępie bezpośrednim wymagały użycia instrukcji z FORTRANU 1900 rozszerzonego. Chodzi tu o dyrektywę DIFINE FILE, w której umieszczone są szczegółowe informacje o organizacji zbioru o bezpośrednim dostępie, które to informacje są wykorzystywane przez kompilator. Te dwa opisane programy powodowały utworzenie w pamięci dyskowej zbiorów seryjnych, o dostępie bezpośrednim w wersjach zredagowanej i niezredagowanej.

Dostarczyły one informacji o czasie i różnicach czasowych przy zakładaniu obu rodzajów zbiorów. Na potrzeby przebiegów aktualizacji uruchomiono czwarty program – SKN4, który tworzył dwa zbiory pomocnicze zapisane na taśmie magnetycznej, które miały spełniać rolę zbiorów pseudotransakcyjnych w stosunku do zbiorów głównych w pamięci dyskowej założonych przez programy SKN2 i SKN3. Każdy z tych zbiorów pomocniczych zawierał po 250 rekordów identycznych co do zawartości, w których pola klucza identyfikujące rekord miały takie wartości, aby został zaktualizowany co 4 rekord w zbiorze głównym czyli 250 na 1000. Identyfikatorem rekordu jest drugie pole w rekordzie. Pierwsze pole zawiera symbol aktualizacji gdzie 1 = wprowadzenie nowego rekordu, 2 = modyfikacja pól w rekordzie istniejącym, 3 = wykreślenie ze zbioru. W przypadku tego modelu symbol aktualizacji zawsze równa się 2.

Zbiory mają nazwę SKN–AKTUAL– R i SKN–AKTUAL–N.

Piąty program SKN5 aktualizuje zbiory seryjne w pamięci dyskowej przy pomocy zbiorów pseudotransakcyjnych utworzonych przez program SKN1. Dokonywana jest tylko modyfikacja zawartości pól (tzn. symbol aktualizacji we wszystkich rekordach zbiorów pseudotransakcyjnych ma wartość = 2), ponieważ zmiana wartości rekordu (jego pól) w zbiorze trwa najdłużej w porównaniu z wprowadzeniem lub kasowaniem rekordu. Aktualizacja polega na przeszukiwaniu i przepisywaniu rekordu po rekordzie do nowego zbioru dyskowego nieaktualizowanych oraz żądanych i zaktualizowanych pozycji „starego" zbioru dyskowego. Mierzony jest czas aktualizacji każdego zbioru dyskowego tzn. zredagowanego SKN–ED1–SE–R i niezredagowanego SKN–ED2–SE–N z podaniem informacji na drukarce wierszowej.

Szósty program SKN6 aktualizuje 2 zbiory o dostępie bezpośrednim przy pomocy zbiorów pseudotransakcyjnych. Aktualizacja polega na odszukaniu żądanego rekordu na podstawie podanej wartości klucza – identyfikatora rekordu ze zbioru pomocniczego. Dokonywana jest zmiana zawartości pól rekordu i ponowny zapis do tego samego zbioru dyskowego. Wyszukanie rekordu dokonuje się przy pomocy zmiennej

stowarzyszonej używanej w instukcji READ a opisanej w dyrektywie DEFINE FILE. Wyszukiwanie przyśpiesza nieznacznie użycie instrukcji FIND z podaniem wartości zmiennej stowarzyszonej. Jak poprzednio dokonywany jest pomiar czasu aktualizacji i wydruk informacji na drukarkę.
Wszystkie programy zostały skompilowane przy użyciu kompilatora dyskowego XFAT w Systemie Operacyjnym GEORGE–3 EMC ODRA 1305. Uruchomienia opisanych programów odbywały się przy użyciu własnych makroinstrukcji.

W wyniku uruchomienia wszystkich programów otrzymano od programów i systemu operacyjnego szereg informacji, które chciałbym częściowo omówić.

Wyniki eksperymentu

Uruchomienie odbywało się przy pełnym monitorowaniu poprzez zlecenie FULLTRACE i nadzorowaniu wykonania ekstrakodu ALLOT i PERI poprzez zlecenie MONITOR ON, OPEN ALLOT[2)].
Wszystkie zbiory w programach zarówno taśmowe jak i dyskowe miały przyjętą standardową we FORTRANIE długość bloku – dla taśm 128 słów, a dla porcji na dyskach 1 blok 128 słowowy zarówno w programie jak i w dyrektywie CREATE dla zbiorów dyskowych w parametrze BUCKET przeznaczonej dla systemu operacyjnego.

Przy uruchomieniu programu SKN1 i zapisie 1000 rekordów system dokonał 207 transmisji zwanych dalej też transferami na taśmę z danymi zredagowanymi i 1007 transferów na taśmę z danymi niezredagowanymi.

Natomiast rekordy niezredagowane, które były fizycznie krótsze, bo tylko 18 słowowe były zapisane po jednym rekordzie w bloku i transmitowane na nośnik.
Na początku wspomniano, że przy ich użyciu można zaoszczędzić na ilości zajętej pamięci magnetycznej.
Warunkiem jest jednak najściślejsze dopasowanie długości rekordu do długości bloku, np. ideałem dla bloku 128 słowowego jest rekord o 64 polach, może to być 1 wiersz macierzy 64 kolumnowej lub zapisane w szeregu 2 wiersze 32 kolumnowej. W sytuacji 1 rekordu w bloku mamy jeszcze jedną korzyść. Zazwyczaj fizycznie mamy dostęp do 1 bloku a logicznie w ramach bloku do 1 rekordu, w tym przypadku jest to sytuacja tego samego rekordu. Cofnięcie o blok to też cofnięcie o rekord. W programie SKN2 system z taśmy zredagowanej dokonał 205 transferów a z niezredagowanej 1005 a więc zblokowane były identycznie jak poprzednie. Natomiast do zbioru zredagowanego na dysku program wykonał 207 transferów w tym 7 systemowych i 1025 transferów do zbioru niezredagowanego w tym 25 systemowych. W kilkakrotnych uruchomieniach czasy zapisów do zbiorów wahały się od 40 do 50 kilku sekund dla każdego zbioru z kilku sekundową różnicą na korzyść zbioru niezredagowanego w ramach jednego

---

2) Pełne monitorowanie poprzez zlecenie FULLTRACE pozwala na pełne informowanie użytkownika o stanie wykonania zadania w maszynie, ekstrakody ALLOT i PERI są to rozkazy wykonywane na poziomie systemu operacyjnego maszyny, nadzorują one tryb otwierania i dostępu do zbiorów użytkownika.

przebiegu. Podane czasy pomiaru nie są czystym czasem transmisji i działania instrukcji programu,
gdzie:

T_d – czas dostępu do zbioru,
T_p – czas działania instrukcji programu nie związanych z dostępem do zbioru,
T_o – czas oczekiwania na przydział procesora w systemie wieloprogramowym.

Czas ten jest w pewnym stopniu zniekształcony na skutek wieloprogramowego działania maszyny. Zakładając jednak, że obciążenie maszyny było stałe w pewnym przedziale czasu 5–10–15 minut, a programy uruchomione były jeden po drugim z limitami czasu 1 minuty na wykonanie, to wyniki można traktować jako pewne proporcje czasów (tzn. jeśli jeden np. działa 20 s., a drugi 60 to różnica jest trzykrotna), zakładając jednakowo równomierną obsługę programów przez system operacyjny w ramach wieloprogramowości. Nie zdarzyło się ani razu, aby któryś z programów przekroczył przyznany limit 1 minuty na wykonanie, co świadczy o w miarę jednakowym czasie wykonania. Porównując czasy zakładania zbiorów i ilości wykonywanych transmisji można stwierdzić, że transfery danych zredagowanych wykonywane są znacznie wolniej. Gdyby umieszczono w bloku tylko 1 rekord zamiast 5 to 1000 transmisji trwałoby prawdopodobnie 5 razy dłużej, co w przyszłości zostanie jeszcze sprawdzone. Wniosek więc taki, że gdyby udało się lepiej wykorzystać blok w zbiorze niezredagowanym, to ilość i czas przesyłania informacji byłyby efektywniejsze. W programie SKN1 czas zakładania zbiorów o dostępie bezpośrednim wahał się w tych samych granicach co poprzednio dla zbiorów seryjnych, lecz w ramach 1 przebiegu czas zapisu zbioru niezredagowanego był niższy o 20–30% od czasu zapisu zbioru zredagowanego.

W programie aktualizacji zbiorów seryjnych czasy aktualizacji 250 rekordów wynosiły w granicach 60-80 sekund z korzyścią dla zbiorów niezredagowanych. Zbiory były odczytywane i przeszukiwane w całości. W programie aktualizacji o dostępie bezpośrednim czasy wahały się w granicach od kilkunastu do dwudziestu kilku sekund i były średnio 4 krotnie krótsze w stosunku do odpowiednich zbiorów w organizacji seryjnej. W tym programie zbiory te nie były przeszukiwane w całości, a tylko odczytywane były rekordy żądane, a więc transmisji było 4 razy mniej. Wynika z tego, że czasy odczytu i zapisu dla tej samej ilości rekordów są podobne w obu organizacjach pamięci. Algorytm systemowy umieszczania i znajdowania rekordu w zbiorze o dostępie bezpośrednim jest na tyle nieskomplikowany, że zabiera czas w stopniu nie zauważalnym. Różnica i efektywność z tym związana polega na ilości koniecznych odczytów i zapisów w zbiorze. W przypadku zbioru seryjnego było to 1000 zapisów i odczytów w zbiorze o zawartości 1000 rekordów, a w zbiorze o dostępie bezpośrednim było to 250 odczytów i zapisów w zbiorze 1000 rekordów przy aktualizowaniu 250 pozycji.

Przy omawianiu eksperymentu mowa jest o czasie przetwarzania i dostępu do zbiorów. Należy sobie jednak uświadomić, że poprzez czas mamy do czynienia z kosztem, gdyż czas przetwarzania kosztuje i jest to koszt całkowicie wymierny. Tak więc minimalizacja czasu przetwarzania to minimalizacja kosztu dla użytkownika.

Zakończenie

Wyniki przeprowadzonego eksperymentu pozwoliły na wstępne sformułowanie wniosków ogólnych. Przy dostępie seryjnym (sekwencyjnym) najbardziej czasochłonne są operacje przesyłania określonej ilości danych z dysku do pamięci operacyjnej maszyny.

Biorąc pod uwagę czynnik czasu, dostęp do zbiorów przy metodzie seryjnej jest wolny, natomiast przy metodzie indeksowo–sekwencyjnej jest bardzo szybki. Przy dokonywaniu aktualizacji zbiorów przy metodzie seryjnej zakres wykonywanych czynności jest duży w przeciwieństwie do metody indeksowo – sekwencyjnej. Aktualizacja zbiorów seryjnych jest procesem czasochłonnym ze względu na konieczność wyszukiwania i identyfikowania poszczególnych rekordów. Zbiory indeksowo – sekwencyjne powinny znajdować zastosowanie jako zbiory systemowe zawierające słowniki, tablice itp., do których potrzebny jest bardzo szybki dostęp po niewielką ilość informacji. Zbiory o organizacji seryjnej, pomimo podanych wyżej wad, mogą być stosowane jako zbiory użytkowe w przypadku gdy nie jest wymagana duża szybkość przetwarzania.

Literatura

Bańkowski J., Fiałkowski K.: Programowanie w języki FORTRAN PWN Warszawa 1982

Borak S., Klaczak J., Korczak S., Płoski Z.: System Operacyjny GEORGE-3 WNT Warszawa 1981

Gontarczyk T.: „Adaptacyjna optymalizacja struktury zbiorów danych technologicznych" PWN W-wa Łódź 1982

Pachelski W.: „Fortran dla maszyn ODRA serii 1300" WNT Warszawa 1980

„Automatyczne programowanie pamięci o bezpośrednim dostępie" Biblioteka podprogramów WZE ELWRO nr 13044/3 Wrocław 1971/72

Pamięć dyskowa maszyn cyfrowych serii ODRA 1300, Skrypt Politechniki Wrocławskiej Wrocław 1977.