ZESZYTY NAUKOWE POLITECHNIKI ŚLĄSKIEJ 1998
Seria: INFORMATYKA z. 35 Nr kol. 1402

Piotr IREK, Jacek KMONK, Dorota PIERZCHAŁA
Politechnika Śląska, Instytut Informatyki

PROJEKT HURTOWNI DANYCH NA WYBRANYM PRZYKŁADZIE; WPŁYW AGREGATYZACJI DANYCH NA EFEKTYWNOŚĆ REALIZACJI ZADAŃ WYSZUKIWANIA

Streszczenie. W artykule przedstawiono sposób projektowania hurtowni danych dla systemów obsługi bieżącej dziekanatów umożliwiającej wykonywanie typowych zestawień na poziomie całej uczelni (rektoratu). Dla hurtowni wypełnionej przykładowymi danymi zaprezentowano porównanie czasów uzyskiwania odpowiedzi na typowe zapytania operujące na danych podstawowych oraz na danych wstępnie zagregatyzowanych. Na podstawie wykonanych badań określono opłacalność agregatyzowania danych w hurtowni.

DATA WAREHOUSE DESIGNING EXAMPLE; INFLUENCE OF DATA AGGREGATION ON SELECTION QUERIES EFFICIENCY

Summary. The way of data warehouse designing for deaneries' operating systems making possible performing typical computations for whole university is described in the paper. The execution time comparison of typical queries on detail and lightly aggregated data for the warehouse filled up with random data is presented. After performed researches the profits of aggregating data are shown.

1. Wstęp - cel i zakres badań

W dużych, skomputeryzowanych przedsiębiorstwach mamy często do czynienia z sytuacją, gdzie każda jednostka przechowuje bieżące informacje we własnej bazie danych (systemie obsługi bieżącej). Bazy danych i aplikacje poszczególnych jednostek zwykle nie współdziałają ze sobą lub istnieje tylko ograniczona możliwość wymiany danych pomiędzy nimi.

[1] Opracowanie powstało częściowo w ramach Projektu Celowego KBN Nr 1764/CT11-8/97.

Z potrzeby stworzenia systemu dającego możliwość złożonej globalnej analizy danych pochodzących z różnych systemów obsługi bieżącej zrodziła się koncepcja hurtowni danych.

Hurtownia danych stanowi centralną składnicę (repozytorium) kopii danych zawartych w bazach danych systemów obsługi bieżącej, przechowywanych w sposób zapewniający możliwość uwzględnienia aspektu czasu i tematycznie zorientowanych. Hurtownia powinna też ułatwiać złożoną analizę danych i w tym celu może przechowywać oprócz danych podstawowych (pochodzących z systemów obsługi bieżącej) pewne dane nadmiarowe, na przykład dane zagregatyzowane. Mówi się o nieulotności danych w hurtowni, co oznacza, że dane raz w niej umieszczone nie powinny być aktualizowane ani usuwane, są one tylko udostępniane do odczytu.

Przedmiotem opisywanych badań było opracowanie projektu hurtowni danych zbierającej informacje z systemów obsługi bieżącej dziekanatów i umożliwiającej wykonywanie typowych zestawień na poziomie całej uczelni (rektoratu).

W ramach badań opracowano projekt struktury hurtowni danych bazując na strukturze przykładowej bazy danych systemu obsługi bieżącej w dziekanacie i uwzględniając najczęściej wykonywane zadania wyszukiwania.

Zaproponowano metody aktualizacji danych w hurtowni na podstawie zmian zawartości baz danych w poszczególnych dziekanatach.

Przeprowadzono eksperymenty w systemie Microsoft Access polegające na:

- stworzeniu hurtowni danych według przygotowanego projektu i wypełnieniu jej losowo wygenerowanymi danymi,
- wykonaniu kilku typowych zadań wyszukiwania na danych podstawowych oraz przechowywanych w hurtowni danych wstępnie zagregatyzowanych,
- porównaniu wyników czasowych wykonywania powyższych zestawień i określeniu opłacalności agregatyzowania danych w hurtowni.

Sformułowano wnioski z przeprowadzonych badań oraz propozycje ich rozszerzenia w przyszłości.

2. Cel tworzenia hurtowni danych

2.1. Struktura systemu obsługi bieżącej

System obsługi bieżącej dziekanatu gromadzi podstawowe informacje na temat studentów, otrzymywanego przez nich stypendium oraz toku ich studiów. Przechowywane są również informacje słownikowe dotyczące wydziałów, kierunków, specjalności, a także wykładanych przedmiotów. Można wyróżnić następujące rodzaje przechowywanych danych:

- dane osobowe studentów przechowywane w tabeli STUDENCI,
- wyniki dydaktyczne studentów (oceny z zaliczeń i egzaminów) przechowywane w tabelach ZALICZENIA i EGZAMINY,
- dane dotyczące organizacji dydaktyki (nazwy wydziałów, kierunków, specjalności, przedmiotów, wykazy zaliczeń i egzaminów) przechowywane w tabelach WYDZIALY, KIERUNKI, SPECJALNOSCI, PRZEDMIOTY, ZAJECIA i SESJA_EGZAMINACYJNA,
- dane o przebiegu studiów (zamknięcie semestru, urlopy, skreślenia, zamknięcie studiów) przechowywane w tabelach SEMESTRY i ABSOLWENCI,
- dane dotyczące stypendiów przechowywane w tabelach STYPENDIA, KONTA_BANKOWE, FUNDATORZY.

2.2. Cele tworzenia hurtowni danych dla systemu obsługi bieżącej dziekanatu

Do podstawowych celów hurtowni możemy zaliczyć:

1. Możliwość śledzenia historii zmian danych.

System obsługi bieżącej nie gromadzi pełnych danych historycznych. Na przykład, dla danego studenta można sprawdzić jego aktualny status, ale nie można znaleźć informacji o jego statusach na poprzednich semestrach.

Dla następujących danych nie jest przechowywana historia ich zmian:

- dane osobowe o studentach (np. zmiana miejsca zamieszkania),
- tok studiów (np. po powrocie z urlopu informacja o pobycie na nim jest gubiona),
- dane o przedmiotach, specjalnościach, kierunkach, wydziałach.

Jedynie dane o stypendiach i wynikach dydaktycznych studentów są w pełni historyczne.

Ze względu na te ograniczenia na podstawie danych zgromadzonych w bazie nie można dokonać pełnego porównania stanu obecnego i stanu z lat poprzednich. Przykładowo nie można stwierdzić, jak zmienia się liczba studentów przebywających w kolejnych latach akademickich na urlopach.

Często istnieje potrzeba przygotowania szeregu zestawień opartych na danych gromadzonych w bazie z uwzględnieniem ich zmian w czasie. Na przykład należy przygotować zestawienie zmian liczby studentów, którzy warunkowo zaliczyli semestr w przeciągu kilku ostatnich lat.

2. Możliwość wykonywania zapytań obejmujących swym zasięgiem dane pochodzące z różnych systemów obsługi bieżącej.

Dane z poszczególnych dziekanatów uczelni po wstępnym przetworzeniu gromadzone są w hurtowni. Umożliwia to wykonywanie szeregu zestawień i zapytań przekrojowych, które do tej pory były przygotowywane ręcznie.

3. Przyspieszenie realizacji często wykonywanych zadań.

W hurtowni znajdują się tabele przechowujące bardzo duże ilości danych. Wykonanie prostych zapytań dotyczących tych danych jest bardzo czasochłonne - może trwać nawet kilka dni. Jedną z metod przyspieszenia wyszukiwania informacji w hurtowni danych jest denormalizacja struktury bazy danych.

W celu dalszego przyspieszenia wyszukiwania przechowuje się dodatkowo dane sumaryczne wyznaczone z danych szczegółowych. Proces wyznaczania tych danych zwany jest agregatyzacją. Struktura tabel agregatów projektowana jest pod kątem zapytań zadawanych przez użytkowników.

2.3. Etapy tworzenia struktury hurtowni danych

Ze względu na odmienne przeznaczenie danych w systemach obsługi bieżącej i hurtowni danych oraz na różnice w typowych zadaniach operujących na tych danych, struktura baz danych w obu przypadkach musi spełniać inne kryteria.

Poniżej zostaną przedstawione modyfikacje poczynione na strukturze bazy danych systemu obsługi bieżącej w celu dostosowania jej do potrzeb hurtowni danych.

2.3.1. Usunięcie danych nadmiarowych

Ze względu na objętość danych oraz zastosowania hurtowni nie wszystkie informacje pochodzące z systemów obsługi bieżącej muszą być w niej przechowywane.

Z systemu zostały usunięte następujące tabele: SESJA_EGZAMINACYJNA, ZAJECIA, KONTA_BANKOWE i FUNDATORZY. Dane przechowywane w tych tabelach są nieistotne dla hurtowni danych.

Z pewnych tabel usunięto nadmiarowe kolumny:

- W tabelach STUDENCI i STYPENDIA zostały odrzucone kolumny służące do wyznaczenia kwoty stypendium. Hurtownia nie wyznacza stypendium, przechowuje tylko jego wysokość.
- W tabelach EGZAMINY i ZALICZENIA zostały odrzucone nadmiarowe kolumny, których zawartość może być wyznaczona na podstawie innych danych, oraz kolumny nie wykorzystywane nawet w systemie obsługi bieżącej.

2.3.2. Podział danych ze względu na częstość ich zmian

W hurtowniach danych unika się operacji modyfikacji zawartości tabel (mówi się o trwałości danych przechowywanych w hurtowniach). Zmiany zawartości baz danych w systemach obsługi bieżącej powinny raczej powodować generację nowych wierszy

w hurtowni, co pozwala na śledzenie historii zmian danych. Jednocześnie należy dbać o to, by przyrost danych w hurtowni był możliwie jak najmniejszy.

Aby nie powielać wielokrotnie wierszy zawierających duże ilości danych (dużą liczbę kolumn) z powodu częstych zmian zawartości niektórych kolumn, korzystnie jest podzielić niektóre tabele na kilka mniejszych fragmentów według kryterium częstości zmian danych.

Z tego powodu tabela STUDENCI została podzielona na dwie części - STUDENCI i TOK. Pierwsza zawiera dane nie zmieniające się wcale lub bardzo rzadko (np. zmiana nazwiska studenta), zaś druga - dane mogące zmieniać się częściej (np. grupa dziekańska, do której należy student).

Podobnie tabela SEMESTRY została podzielona na dwie tabele: SEMESTRY i SEMESTRY_DOD. Tabela SEMESTRY zawiera informacje, które mogą się zmieniać kilka razy w ciągu jednego semestru, zaś SEMESTRY_DOD informacje, które są stałe w ciągu jednego semestru.

2.3.3. Denormalizacja tabel

Wysoce znormalizowana struktura bazy danych ułatwia zapewnienie spójności danych oraz przyspiesza zadania ich modyfikacji. Ma to duże znaczenie dla systemów OLTP (ang. On Line Transaction Proccessing), jakimi są systemy obsługi bieżącej. W hurtowniach danych, gdzie przeważają zadania wyszukiwania, a modyfikacje wykonywane są tylko przez administratora w wyznaczonym czasie w sposób zapewniający spójność danych, powinno się dokonać celowych denormalizacji mając na celu optymalizację czasu wykonywania najbardziej typowych zadań.

W tym celu (głównie dla zmniejszenia liczby złączeń tabel) została dokonana następująca zmiana struktury bazy danych:

- z tabel WYDZIALY, KIERUNKI i SPECJALNOSCI została utworzona jedna tabela WYDZ_KIER_SPEC,
- do tabel EGZAMINY i ZALICZENIA została dodana kolumna zawierająca nazwę przedmiotu powielona z tabeli PRZEDMIOTY,
- do tabeli STYPENDIA zostały dodane kolumny identyfikatorów wydziału i kierunku pochodzące z tabeli WYDZ_KIER_SPEC.

2.3.4. Dodanie atrybutu czasu

W celu śledzenia historii zmian danych do tabel, w których nie ma informacji o czasowej ważności danych, zostały dodane dodatkowe kolumny *data_od* (początek ważności danych) i *data_do* (koniec ważności danych).

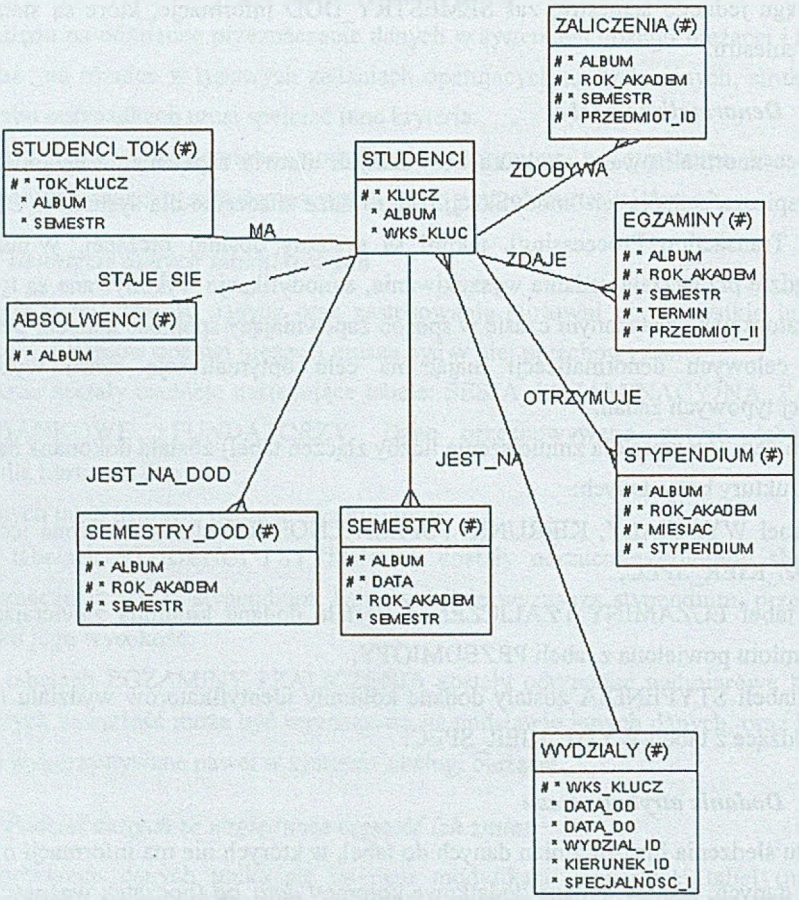
Tabele uzupełnione o te kolumny to: STUDENCI, TOK, WYDZ_KIER_SPEC, PRZEDMIOTY.

2.3.5. Dodatkowe modyfikacje

W systemie obsługi bieżącej dziekanatu dane takie, jak: nazwy przedmiotów, wydziałów i kierunków identyfikowane są przez skróty ich nazw. Nie ma tam potrzeby śledzenia zmiany tych nazw i nie ma możliwości odnotowania faktu, że np. kierunek studiów mimo zmiany swojej nazwy pozostaje logicznie tą samą jednostką organizacyjną.

Problem ten rozwiązano w hurtowni danych przez wprowadzenie dodatkowych identyfikatorów, o wartościach niezależnych od zmian związanych z nimi nazw.

Po wprowadzeniu powyższych zmian otrzymano strukturę hurtowni danych, której najistotniejsze elementy przedstawia rysunek 1.



Rys. 1. Ogólna struktura hurtowni danych
Fig. 1. General structure of data warehouse

2.4. Agregatyzacja danych

Większość zadań wyszukiwania w hurtowniach danych związana jest z prezentacją pewnych zestawień zawierających najczęściej takie informacje, jak sumy, średnie, liczebność danych itp.

Uzyskanie takich informacji z tabel zawierających dane podstawowe wymaga grupowania dużej ilości danych i wykonania na nich operacji agregujących, co zwykle jest bardzo czasochłonne. W celu skrócenia czasu wykonania tych operacji do hurtowni danych zostały wprowadzone tabele zawierające dane zagregatyzowane. Tabele te są tworzone pod kątem spodziewanych potrzeb użytkowników.

W omawianej hurtowni danych zostały zaprojektowane następujące tabele agregatów:

- AGR_LICZBY_STUDENTOW - zawiera liczbę studentów określonego typu na poszczególnych kierunkach, latach akademickich i semestrach studiów,
- AGR_OCEN_STUDENTOW - zawiera semestralne i roczne średnie ocen dla poszczególnych studentów,
- AGR_OCEN_GLOBALNIE - zawiera semestralne i roczne średnie ocen studentów na poszczególnych kierunkach,
- AGR_PRZEDZIALOW_OCEN - zawiera liczby studentów posiadających roczne średnie ocen o wartościach mieszczących się w określonych przedziałach,
- AGR_PRZEDZIALOW_OCEN_2 - o zawartości jak AGR_PRZEDZIALOW_OCEN i organizacji umożliwiającej zadawanie innego typu zapytań,
- AGR_STYPENDIOW - zawiera sumaryczne kwoty różnych typów stypendiów wypłacanych na poszczególnych kierunkach oraz liczby studentów pobierających te stypendia.

W tabelach AGR_OCEN_STUDENTOW i AGR_OCEN_GLOBALNIE oprócz kolumn zawierających średnie ocen zostały dodane kolumny zawierające liczby tych ocen i ich sumy (jako wartości addytywne) w celu umożliwienia agregatyzacji wyższego poziomu. Bez tych dodatkowych kolumn wyznaczenie np. średnich dla poszczególnych grup dziekańskich czy całej uczelni wymagałoby przetwarzania danych szczegółowych.

Tabela AGR_PRZEDZIALOW_OCEN jest zoptymalizowana pod kątem wyszukiwania liczby studentów posiadających średnie ocen zawarte w typowych przedziałach, natomiast tabela AGR_PRZEDZIALOW_OCEN_2 umożliwia znacznie łatwiejsze wyznaczanie liczby studentów dla dowolnych przedziałów ocen.

3. Problemy migracji danych z systemów obsługi bieżącej do hurtowni

Zawartość hurtowni danych powinna odzwierciedlać informacje zgromadzone w poszczególnych systemach obsługi bieżącej (w tym przypadku w dziekanatach). W tym celu informacje w nich zgromadzone muszą okresowo zasilać hurtownię danych. Proces ten jest praco- i czasochłonny oraz powtarzalny, więc wymaga pewnej automatyzacji.

Zakładamy, że wszystkie systemy obsługi bieżącej stanowią jednorodne środowisko pod względem oprogramowania i systemu zarządzania bazą danych. Ponieważ to samo środowisko (MS Access) przyjęto dla hurtowni danych, w procesie migracji danych można pominąć występujący zwykle w warunkach rzeczywistych etap konwersji danych oraz ujednolicania ich struktury.

Ogólnie rozróżnia się trzy rodzaje danych ładowanych do hurtowni ze środowisk operacyjnych (systemów obsługi bieżącej):

- ładowanie danych archiwalnych (jednorazowe),
- ładowanie danych, które aktualnie zawierają poszczególne bazy danych (jednorazowe),
- odzwierciedlanie w hurtowni danych zmian, jakie zaszły w bazach danych systemów obsługi bieżącej od ostatniego ładowania (okresowe).

W rzeczywistych systemach z reguły rezygnuje się z ładowania danych archiwalnych, ze względu na ich mniejszą przydatność oraz minimalizację objętości hurtowni danych. Ładowanie danych istniejących aktualnie w bazach danych odbywa się jednokrotnie i z reguły nie przedstawia większych trudności.

Najtrudniejszym zadaniem jest natomiast okresowe odświeżanie zawartości hurtowni danych na podstawie zmian, jakie zachodzą w poszczególnych bazach danych. Nie jest łatwe do zrealizowania pełne wychwytywanie tych zmian bez odczuwalnego wpływu na wydajność działania poszczególnych systemów OLTP. Istnieje pięć podstawowych technik używanych dla zminimalizowania przeszukiwania danych w systemach obsługi bieżącej:

- Dodanie do wszystkich tabel w poszczególnych bazach danych kolumny zawierającej ślad ostatniej aktualizacji - w ten sposób zawężamy przeglądanie danych do danych zaktualizowanych (bądź dodanych) po ostatnim odświeżaniu hurtowni. Ta metoda wymaga odrębnego odnotowywania faktów ewentualnego usuwania danych.
- Tworzenie w ramach systemów obsługi bieżącej tzw. plików delta, zawierających tylko informacje o zaistniałych zmianach. To bardzo wydajny sposób, gdyż dane, które się nie zmieniają, nie są niepotrzebnie przetwarzane w procesie ładowania.

Odpowiednio przygotowane pliki delta umożliwiają również znaczne przyspieszenie obliczania nowych wartości danych zagregatyzowanych.

- Jeśli nie jest możliwe tworzenie plików delta przez aplikacje, podobne informacje można uzyskać z tworzonych automatycznie plików dziennika (logu) lub audita. Wadą jest fakt, że często pliki te zablokowane są na wyłączność przez serwer bazy danych. Struktura takich plików nie zawsze jest znana programistom, a poza tym najczęściej zawierają one wiele więcej informacji niż to jest potrzebne w procesie ładowania.
- Najkorzystniejszym wyjściem byłaby modyfikacja kodu aplikacji systemów obsługi bieżącej, aby przygotowywać dogodne, specjalnie spreparowane dla hurtowni dane. Najczęściej jednak taka modyfikacja jest nierealna.
- Najmniej wydajnym, lecz czasem jedynym możliwym do zastosowania sposobem jest tworzenie migawek bazy danych i porównywanie ich w celu detekcji zmian.

Ze względu na rodzaj przechowywanych w hurtowni danych można podzielić je generalnie na dwie grupy:

- prosto kumulowane - dane w hurtowni są trwałe, zarówno w postaci szczegółowej, jak i zagregatyzownej; proces ładowania danych do hurtowni polega tylko na uzupełnieniu danych i aktualizacji agregatów,
- sumaryzowane (ang. rolling summary) - starsze dane nie są przechowywane w postaci szczegółowej, tylko w postaci agregatów; im dane starsze, tym mniej szczegółów zawierają. Proces ładowania wiąże się tu z większą reorganizacją danych.

Odrębny problem stanowi dobór odpowiedniej częstotliwości odświeżania hurtowni danych i jest on ściśle związany z zastosowaniem hurtowni.

3.1. Detekcja zmian w bazach danych obsługi bieżącej dziekanatów

System zarządzania bazami danych Access nie tworzy plików dziennika ani audita, dlatego nie jest możliwe śledzenie zmian zawartości baz danych na tej podstawie. Automatycznie nie są tworzone przez rozpatrywaną aplikację żadne pliki delta, ani nie jest dodawany znacznik czasu ostatniej zmiany danych.

Jedynym sposobem na wychwycenie zmian zawartości danych w takiej sytuacji jest więc wykonanie migawek stanu bazy danych i ich porównywanie. Najkorzystniejszym rozwiązaniem byłaby takie przystosowanie aplikacji do współpracy z hurtownią, aby tworzone były pliki delta w trakcie bieżącej pracy systemu.

Do dalszych rozważań założymy, że istnieją pliki delta, które mogą również powstać w wyniku porównania migawek bazy danych.

3.2. Odświeżanie zawartości hurtowni danych na podstawie plików delta

Proces odświeżania można generalnie podzielić na dwa etapy:

- aktualizacja danych szczegółowych na podstawie zmian we wszystkich systemach obsługi dziekanatów,
- aktualizacja agregatów przechowywanych w hurtowni.

Jeśli założymy, że hurtownia ma model kumulacyjny, pierwszy z tych etapów polegać będzie na prostym uzupełnieniu tabel zawierających dane szczegółowe:

- dla danych nowo utworzonych w bazach danych powstaną nowe wiersze w odpowiednich tabelach,
- fakt usunięcia danych może zostać odnotowany poprzez wpisanie w kolumnie *data_do,* oznaczającej datę aktualności tych danych w systemie obsługi bieżącej, wartości daty usunięcia odpowiedniego wiersza z bazy danych,
- fakt modyfikacji danych jest odnotowany poprzez umieszczenie daty modyfikacji w kolumnie *data_do* oraz umieszczenie nowego wiersza z wartościami aktualnymi.

Jeżeli ze względu na objętość hurtowni zrezygnujemy z trwałego przechowywania wszystkich szczegółowych danych - można najstarsze, nieprzydatne dane usuwać po załadowaniu nowych danych.

Częstotliwość wykonywania tej fazy zależy od wymagań dotyczących aktualności danych szczegółowych w stosunku do zawartości baz danych systemów obsługi bieżącej. Ze względu na funkcję hurtowni można tu zaproponować comiesięczne odświeżanie danych.

Faza aktualizacji agregatów polegać będzie głównie na tworzeniu nowych wierszy w tabelach zawierających dane zagregatyzowane. Aktualizacja danych już istniejących dotyczy tabel AGR_OCEN_STUDENTOW i AGR_OCEN_GLOBALNE, których wiersze są tworzone po zakończeniu semestru zimowego, a uzupełniane po zakończeniu semestru letniego, czyli całego roku akademickiego. Zakładamy, że danych zagregatyzowanych z hurtowni nie usuwa się.

Nowe wiersze powstają w wyniku przetwarzania tylko nowych danych, załadowanych w pierwszej fazie. Ponieważ niektóre z danych sumarycznych powstają w wyniku złożonego i czasochłonnego przetwarzania danych podstawowych (np. dane przechowywane w tabeli AGR_PRZEDZIALOW_OCEN), korzystniej byłoby obliczać je na podstawie zawartości pliku delta, a nie danych załadowanych do tabel podstawowych. Powodem tego jest duża objętość danych podstawowych w hurtowni oraz zaobserwowana w trakcie badań niezbyt wydajna optymalizacja wykonania zapytań w systemie Access.

Ze względu na charakter danych zagregatyzowanych przechowywanych w zaprojektowanej hurtowni, faza obliczania wartości zagregatyzowanych nie musiałaby występować podczas każdego odświeżania danych w hurtowni, jeśli występowałoby ono

częściej niż raz w miesiącu. Comiesięcznie tworzone byłyby wiersze w tabeli AGR_STYPENDIUM, pozostałe dane wymagają odświeżania tylko raz w semestrze lub raz do roku (AGR_PRZEDZIALOW_OCEN i AGR_PRZEDZIALOW_OCEN_2).

Proces odświeżania danych powinien blokować bazę danych hurtowni na wyłączność, nie dopuszczając równocześnie wykonywanych zadań wyszukiwania ze strony klientów hurtowni.

4. Porównanie czasów wykonywania zapytań operujących na danych szczegółowych i zagregatyzowanych

W celu określenia opłacalności przechowywania danych nadmiarowych w tabelach agregatów przeprowadzono badania porównawcze na eksperymentalnej hurtowni danych.

4.1. Warunki sprzętowo-programowe

Badania przeprowadzono z wykorzystaniem oprogramowania Microsoft Access 2.0 dla MS Windows 3.x, zainstalowanego na komputerze wyposażonym w procesor Pentium taktowany zegarem o częstotliwości 90 MHz oraz 16 MB pamięci operacyjnej.

Baza hurtowni danych została na czas eksperymentu umieszczona na dysku lokalnym w celu wyeliminowania wpływu zmian obciążenia sieci na uzyskiwane wyniki.

Zostały stworzone programy wypełniające hurtownię przykładowymi danymi. Baza danych o strukturze przedstawionej w rozdziale 2 niniejszego opracowania, do której zadawano zapytania,miała objętość ponad 250 MB. Zawierała m.in. dane osobowe dotyczące 10 000 studentów studiujących na 45 kierunkach, informacje na temat wpisów 708 111 zaliczeń i 212 543 egzaminów. Dane zagregatyzowane zostały obliczone na podstawie całej zawartości tabel danych szczegółowych i zawierają następujące liczby wierszy:

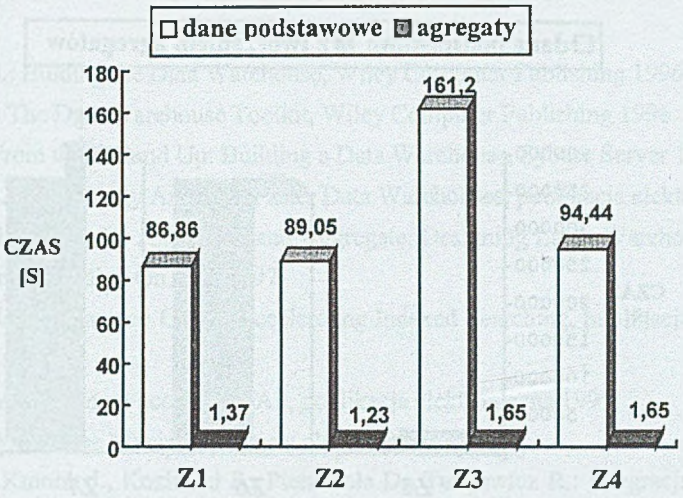
- 5239 w tabeli AGR_LICZBY_STUDENTOW,
- 1125 w AGR_OCEN_GLOBALNIE i AGR_PRZEDZIALOW_OCEN,
- 35451 w AGR_OCEN_STUDENTOW,
- 29250 w AGR_OCEN_STUDENTOW_2.

4.2. Zadania wyszukiwania

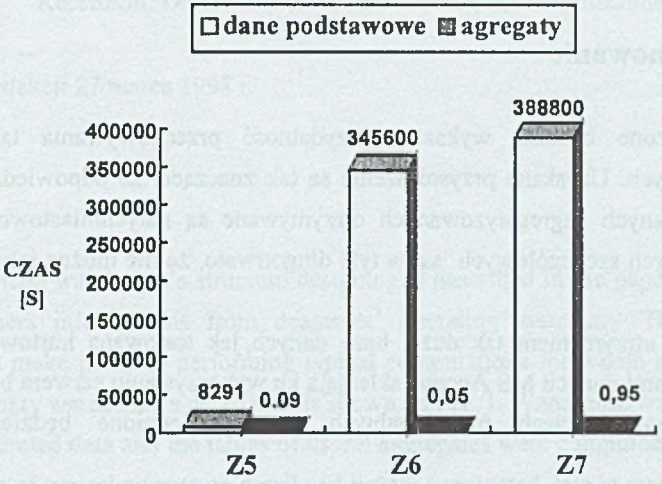
Przeprowadzone zadania wyszukiwania zostały sformułowane na podstawie dokonywanych okresowo zestawień w dziekanatach i rektoracie. Na rysunkach 2,3,4 przedstawione są średnie czasy wykonywania następujących zapytań:

- Z1 - podać liczby studentów na poszczególnych kierunkach wpisanych na semestr zimowy w roku akademickim 1996/1997,
- Z2 - podać sumaryczne liczby studentów na poszczególnych wydziałach i kierunkach wpisanych warunkowo i bezwarunkowo na semestr zimowy w roku akademickim 1996/1997,
- Z3 - dla każdego wydziału i kierunku podać kwoty wypłaconych stypendiów poszczególnych rodzajów oraz liczby studentów pobierających je na drugim roku studiów w styczniu 1996 roku,
- Z4 - podać łączne kwoty wszystkich rodzajów stypendiów wypłaconych na poszczególnych wydziałach w 1996 roku oraz łączne liczby studentów pobierających te stypendia,
- Z5 - dla każdego wydziału i kierunku podać średnie roczne ocen z egzaminów i zaliczeń oraz średnie całkowite uzyskane na drugim roku studiów w roku akademickim 1996/1997,
- Z6 - dla każdego wydziału i kierunku podać liczby studentów na drugim roku studiów w roku akademickim 1996/1997, którzy uzyskali średnie ocen większe od 4.5,
- Z7 - dla każdego wydziału i kierunku podać liczby studentów, którzy uzyskali w roku akademickim 1996/1997 średnie ocen większe od 4.5.

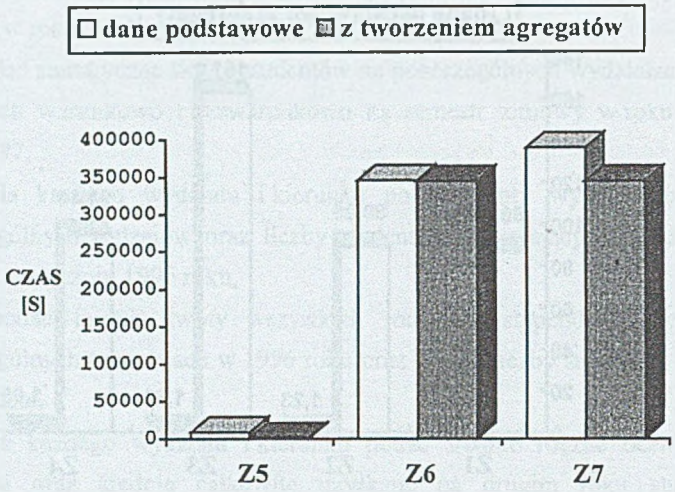
Na rysunkach 2 i 3 prezentowane są uzyskane średnie czasów odpowiedzi na zapytania, które były kierowane na przemian do tabel zawierających dane szczegółowe i do tabel agregatów. Jak można było się spodziewać, stworzenie tabel agregatów wpływa na znaczne przyspieszenie uzyskiwania wyników tworzonych zestawień. Czasy uzyskiwania odpowiedzi na niektóre zapytania (np. Z6 i Z7), operujące na danych szczegółowych, rzędu kilku dni, są nieakceptowalne. Zwykle w hurtowniach danych tworzy się tabele agregatów przyspieszające wykonanie często powtarzających się zapytań. Badania wykazały, że w przypadku, gdy agregaty powstają w wyniku złożonych operacji na dużych ilościach danych, stworzenie tabel agregatów może być opłacalne nawet dla jednokrotnego wykonania zapytania. Zostało to przedstawione na rysunku 4, gdzie pierwsza kolumna (w legendzie - dane podstawowe) to czas wykonania zapytania na danych podstawowych, zaś druga (w legendzie - z tworzeniem agregatów) to suma czasu tworzenia agregatów i wykonania zapytania na danych zagregatyzowanych.



Rys. 2. Czas wykonywania zapytań - cz. 1
Fig. 2. Time of querying data - part 1



Rys. 3. Czas wykonywania zapytań - cz. 2
Fig. 3. Time of querying data - part 2



Rys. 4. Czas wykonywania zapytań - cz. 3
Fig. 4. Time of querying data - part 3

5. Podsumowanie

Przeprowadzone badania wykazały przydatność przechowywania tabel agregatów w hurtowni danych. Uzyskane przyspieszenia są tak znaczące, że odpowiedzi na zapytania operujące na danych zagregatyzowanych otrzymywane są natychmiastowo, podczas gdy operacje na danych szczegółowych są na tyle długotrwałe, że nie można ich wykonywać na bieżąco.

Trudności z utrzymaniem tak dużej bazy danych, jak testowana hurtownia w systemie zarządzania bazami danych MS Access, skłaniają ku wykorzystaniu serwera baz danych SQL o większych możliwościach funkcjonalnych, co uwzględnione będzie w przyszłych badaniach.

Planowane jest również porównanie funkcjonowania hurtowni danych o strukturze zaprezentowanej w niniejszym opracowaniu z hurtownią o strukturze opartej na wielowymiarowym modelu danych.

LITERATURA

1. Inmon W.H.: Bulding the Data Warehouse, Wiley Computer Publishing 1996.
2. Kimball R.: The Data Warehouse Toolkit, Wiley Computer Publishing 1996.
3. Mundy J.: From the Ground Up: Building a Data Warehouse, Sybase Server 1995.
4. Edelstein H.: Technology Analysis: Faster Data Warehouses, publikacja elektroniczna 1995.
5. Meredith M. E., Khader A.: Divide and Aggregate: Designing Large Warehouses, Database Programming & Design On Line 1997.
6. Bontempo Ch. J., Saracco C. M.: Accelerating Indexed Searching, publikacja elektroniczna 1997.
7. Carickhoff A.: A New Face For OLAP, publikacja elektroniczna 1997.
8. Baum D.: Warehouse Mania, publikacja elektroniczna 1996.
9. Frączek J., Kmonk J., Kozielski S., Pierzchała D., Tutajewicz R.: Integracja baz danych - przegląd metod i narzędzi; analiza potrzeb na wybranym przykładzie, ZN Pol. Śl., ser. Informatyka z. 34, Gliwice 1998

Recenzent: Dr hab. inż. Stanisław Wołek, Prof. Politechniki Rzeszowskiej

Wpłynęło do Redakcji 27 marca 1998 r.

Abstract

The way of data warehouse's structure designing is described in the paper. The example warehouse gathers informations from deaneries' operating systems. The aim of the warehouse is to make possible performing typical computations for whole university. The structure of deanery warehouse's detail data is shown on Fig. 1. Warehouse was filled up with the random generated data and the tables of useful aggregates were computed. The execution time comparison of typical queries on detail and lightly aggregated data is presented on Fig. 2 - Fig.3. Fig. 4 shows the comparison of performing query on detail data and the sum of computing aggregates and performing query on it. The researches proved the profits of storing aggregated data within the warehouse.