ZESZYTY NAUKOWE POLITECHNIKI ŚLĄSKIEJ 1999
Seria: INFORMATYKA z. 37 Nr kol. 1421

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

BUDOWA HURTOWNI DANYCH I APLIKACJI ANALITYCZNYCH NA PRZYKŁADZIE NARZĘDZI SAS I ORACLE[1]

Streszczenie. Artykuł zawiera przegląd podstawowych różnic między bazą danych a hurtownią danych. Przedstawiono kolejne etapy budowy i utrzymania hurtowni danych oraz przykłady teoretycznych rozwiązań typowych problemów badawczych z nimi związanych.

Zaprezentowano również praktyczne rozwiązania zasygnalizowanych problemów w dziedzinie hurtowni danych na przykładzie systemów SAS i Oracle.

BUILDING THE DATA WAREHOUSE AND OLAP APPLICATIONS BY EXAMPLE OF ORACLE AND SAS SYSTEMS

Summary. The paper contains review of fundamental differences between database and data warehouse. The steps of building and maintenance of data warehouse and examples of theoretical solving of typical research problems connected with it is presented.

The practical implementation of solving of indicated problems in data warehousing by examples of Oracle and SAS systems is shown.

1. Wstęp

W dużych przedsiębiorstwach większość procesów związanych z przetwarzaniem informacji na bieżąco jest wspomaganych komputerowo, a informacje te przechowywane są najczęściej w relacyjnych bazach danych. Wraz ze wzrostem ilości gromadzonych informacji pojawiły się tendencje zmierzające do wykorzystania danych pochodzących z różnych systemów obsługi bieżącej w procesach wspomagania podejmowania decyzji. W celu umożliwie-

[1] Opracowanie powstało częściowo w ramach Badań Własnych KBN Nr 408/RAU2/98

nia przeprowadzania złożonych, efektywnych analiz danych, nie obciążających systemów wykorzystywanych w bieżącej działalności, informacje z baz danych odzwierciedla się w centralnym repozytorium, zwanym hurtownią danych. Jest to termin stosunkowo nowy i obecnie jedna z najdynamiczniej rozwijających się dziedzin informatyki. Podstawowe różnice między tradycyjną bazą danych a hurtownią danych prezentuje poniższa tabela.

Tabela 1

Różnice między bazą danych a hurtownią danych

Baza danych	Hurtownia danych
dane zorganizowane pod kątem aplikacji, fragmentaryczne w obrębie całego przedsiębiorstwa, w celu wspmagania bieżącej działalności poszczególnych działów	dane zorganizowane zgodnie z problemem analiz, najczęściej kompleksowe spojrzenie na przedsiębiorstwo, w celu wsparcia procesu podejmowania decyzji
dane aktualne (dane historyczne bywają najczęściej archiwizowane i nie są dostępne na bieżąco)	dane w pełni historyczne - hurtownia ma umożliwiać analizy zmian danych w czasie, dane z reguły nie są usuwane
dane są często aktualizowane (istniejące dane są poddawane modyfikacjom)	dane raz wprowadzone nie powinny ulegać modfikacji; zmiany wartości danych powinny raczej generować nowe dane w hurtowni
dane wysoko znormalizowane, nieredundatne, w celu uniknięcia anomalii przy aktualizacji	dane celowo zdenormalizowane, redundantne (przechowywane są np. agregaty oraz metadane)
dane zoptymalizowane pod kątem przetwarzania transakcyjnego	optymalizacja pod kątem analiz (wyszukiwania), a nie aktualizacji
wiele tabel, nieduże ilości danych	niewiele tabel, ogromne ilości danych
dane przechowywane w dwuwymiarowych tabelach	dane wielowymiarowe
tabele w dużym stopniu wypełnione	występuje problem rzadkiego wypełnienia wielowymiarowych baz danych
indeksy: B-drzewa w różnych wariantach, tablice mieszające	indeksy takie jak w bazach danych i różne warianty indeksów bitmapowych
typowe zapytanie operuje często na poziomie poszczególnych wierszy	analiza dużych porcji danych, a informacje na poziomie pojedynczych wierszy często nieistotne
zapytanie daje często zbiory wynikowe o dużym rozmiarze (duża liczba wierszy transmitowanych do komputera klienta)	daje zwykle zbiory wynikowe niewielkich rozmiarów - podsumowania, kilka ekstremalnych wartości itp.
analizy danych są proste (filtrowanie, wyszukiwanie wg wzorca, proste funkcje agregujące)	złożone analizy danych (data-mining, odkrywanie wiedzy) a z operacji podstawowych: zwijanie, rozwijania, obroty, rozszerzone funkcje agregujące
cele analiz (zapytań) dobrze zdefiniowane	nie zawsze zdefiniowane cele analiz
czas uzyskania odpowiedzi na pytanie jest krytyczny (co najwyżej rzędu sekund)	czas nie jest czynnikiem krytycznym, czas wykonania złożonej analizy wyrażony w minutach jest w pełni akceptowalny

cd. tabeli 1

muszą występować blokady danych - dane są jednocześnie selekcjonowane i modyfikowane przez użytkowników	nie występują problemy współbieżnego dostępu do danych - dane są przeznaczone tylko do odczytu, a w czasie ładowania danych dane nie są dostępne dla analiz
wiele jednoczesnych połączeń do bazy danych, wielu użytkowników	mniejsze liczby użytkowników, najczęściej grono ograniczone do decyzyjnej kadry kierowniczej

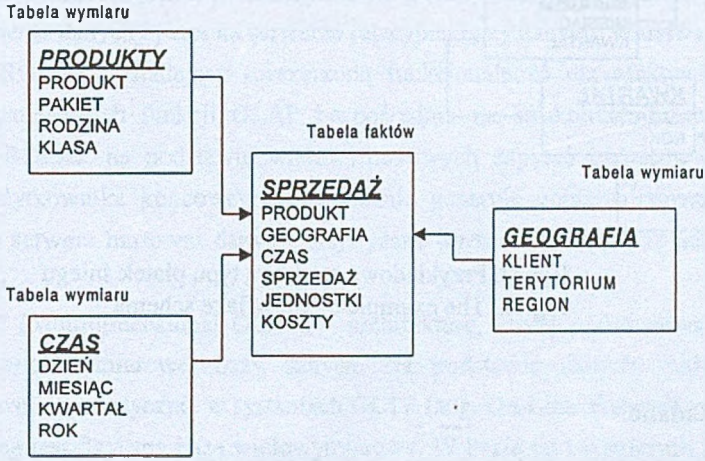
2. Etapy projektowania i utrzymywania hurtowni danych

2.1. Projekt struktury

Przed przystąpieniem do omawiania struktur hurtowni danych należy zapoznać się z podstawowymi definicjami:

- fakty - analizowane dane numeryczne (np. wielkość sprzedaży),
- wymiary - pogrupowane tematycznie atrybuty faktów (np. czas),
- hierarchie wymiarów - relacje typu rodzic-potomek pomiędzy atrybutami wewnątrz jednego wymiaru (np. relacje między atrybutami wymiaru czasu – dzień, miesiąc, kwartał, rok).

Najczęściej spotykaną strukturą hurtowni danych jest struktura gwiazdy. Najprostszym typem struktury gwiaździstej jest baza składająca się tylko z jednej tabeli faktów, której wiersze odpowiadają faktom oraz z tabel wymiarów. Każdy wiersz w tabeli faktów zawiera zwykle analizowane fakty oraz klucze obce każdej z tabel tworzących wymiary.

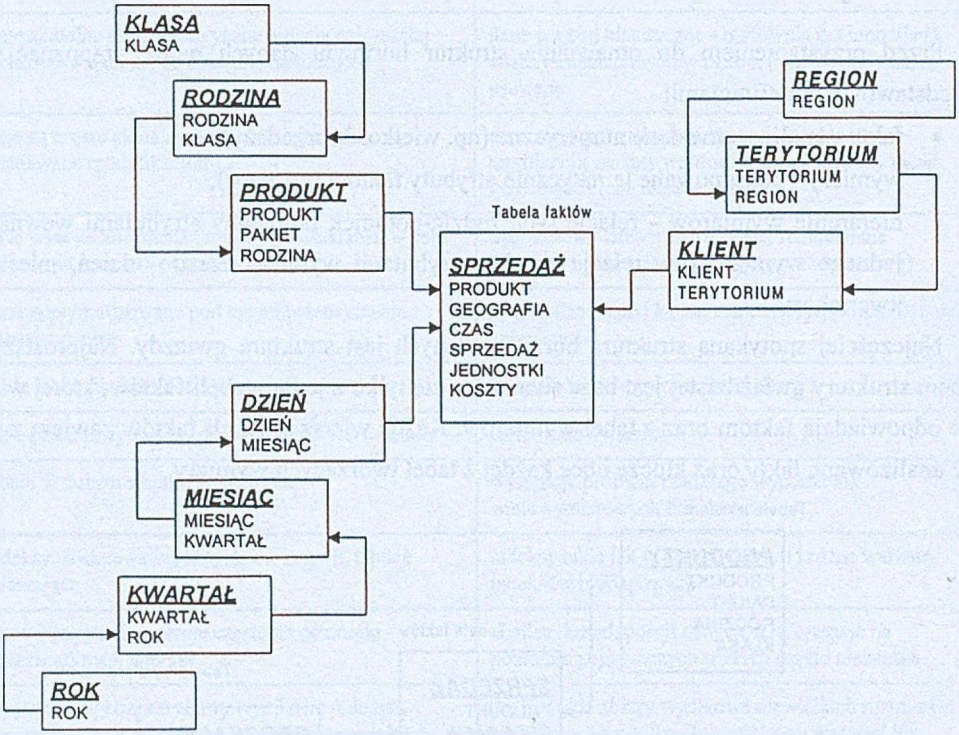


Rys. 1. Przykładowa struktura gwiazdy
Fig. 1. The example of star schema

W tej strukturze jeden wymiar reprezentowany jest zawsze przez jedną tabelę wymiaru. Każda tabela wymiaru składa się z kilku poziomów atrybutów opisujących dany wymiar. Poziomy te zwykle są uporządkowane, tworząc jedną lub więcej hierarchii wymiarów. Ponieważ tabele wymiarów nie są znormalizowane, to schemat ten charakteryzuje się niejawną hierarchią atrybutów wymiarów. Przykładową strukturę połączenia w gwiazdę przedstawia rys. 1.

Innym możliwym typem struktury gwiaździstej jest baza danych zawierająca kilka tabel faktów, które wykorzystują te same tabele wymiarów. Spotyka się również schemat bazy danych, w którym tabele wymiarów są znormalizowane. Schemat taki nosi nazwę struktury płatka śniegu. Istotną cechą tego schematu jest jawność występowania hierarchii wymiarów.

Taką przykładową strukturę przedstawiono na rys. 2.



Rys. 2. Przykładowa struktura typu płatek śniegu
Fig. 2. The example of snowflake schema

2.2. Metadane

Aby możliwe było budowanie, utrzymywanie i używanie danych z hurtowni, potrzebne są pewne dane opisujące zawartość hurtowni, sposoby pozyskiwania danych z systemów opera-

cyjnych oraz opisujące sposoby agregatyzacji danych. Dane te nazywane są metadanymi (danymi o danych).

Można wyróżnić dwie grupy metadanych:

- biznesowe - atrybuty i właściwości hurtowni danych używane przez użytkowników końcowych,
- techniczne - opisują przepływ danych z systemów obsługi bieżącej do hurtowni danych.

Metadane mogą być współdzielone pomiędzy kilka hurtowni danych. Zawierają wtedy m.in. informacje o serwerach, na których rezydują hurtownie danych, informacje o dostępnych Systemach Zarządzania Bazami Danych (SZBD), administratorach hurtowni itp.

2.3. Architektury systemów OLAP

Ponieważ serwery relacyjnych baz danych nie posiadają wystarczających mechanizmów do przeprowadzania skomplikowanych operacji analizy danych, powstało wiele architektur systemów OLAP (ang. On-Line Analytical Processing) ułatwiających takie analizy. Wśród nich możemy wyróżnić:

- DOLAP (Desktop OLAP) - architekturę składającą się z dwu warstw. Pierwsza z nich to serwer relacyjnej bazy danych, na którym rezyduje hurtownia danych. Najczęściej hurtownia ma strukturę gwiazdy. Druga warstwa to kompletna aplikacja lub narzędzie służące do budowy aplikacji. Narzędzia drugiej warstwy umożliwiają wykonywanie bardzo rozbudowanych analiz.
- ROLAP (Relational OLAP) - architekturę o trzech warstwach. Podstawowa warstwa to hurtownia danych oparta na serwerze relacyjnej bazy danych. Warstwa pośrednia to serwer ROLAP posiadający rozszerzoną funkcjonalność dla efektywnej realizacji wielowymiarowych funkcji OLAP bezpośrednio na strukturach hurtowni danych. Serwer ROLAP na podstawie wielowymiarowych zapytań biznesowych zadanych przez użytkownika końcowego dynamicznie generuje zoptymalizowane zapytania SQL do serwera hurtowni danych. Najwyższa warstwa to klienckie narzędzia użytkownika.
- MOLAP (Multidimensional OLAP) – architekturę, której podstawową częścią jest serwer wielowymiarowej bazy danych. Na podstawie danych znajdujących się w hurtowni i sporadycznie w systemach OLTP (ang. On-Line Transaction Processing) budowana jest fizyczna baza wielowymiarowa. W bazie tej tworzonych jest wiele indeksów, co znacznie przyspiesza wykonywanie złożonych analiz. Istotnymi wadami

tego rozwiązania jest ograniczony rozmiar bazy oraz jej małe wypełnienie, co grozi złym wykorzystaniem pamięci, szczególnie przy rzadko upakowanych zbiorach.

- HOLAP (Hybrid OLAP) - jest to rozwiązanie, w którym część danych jest przechowywana w hurtowni danych, a część (najczęściej dane zagregatyzowane) w bazie wielowymiarowej.

2.4. Ekstrakcja i ładowanie danych

Proces ekstrakcji i ładowania danych ma na celu zasilanie hurtowni danymi. Zwykle przebiega on w dwóch etapach. Pierwszy z nich to zasilenie hurtowni danymi historycznymi. Proces ten ma charakter jednorazowy. Hurtownia jest wtedy zasilana danymi historycznymi z systemów transakcyjnych, jak i z nośników archiwalnych. Drugi etap to okresowe ładowanie hurtowni danymi bieżącymi.

Przy realizacji obu etapów można wyróżnić następujące kroki:

- identyfikacja potrzebnych danych,
- właściwa ekstrakcja danych,
- czyszczenie danych wraz z odrzuceniem danych niepoprawnych,
- przekształcenie we właściwe formaty,
- jeśli to możliwe przygotowanie agregatów,
- załadowanie danych do hurtowni.

2.5. Okresowe ładowanie danych

Zawartość hurtowni danych powinna odzwierciedlać informacje zgromadzone w systemach obsługi bieżącej. Jednym z problemów w dziedzinie hurtowni danych jest okresowe odświeżanie zawartości hurtowni danych na podstawie zmian, jakie zachodzą w poszczególnych bazach danych. Nie jest łatwe do zrealizowania wydajne 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 śladu ostatniej aktualizacji,
- tworzenie w ramach systemów obsługi bieżącej tzw. plików delta, zawierających tylko informacje o zaistniałych zmianach,
- jeśli nie jest możliwe tworzenie plików delta przez aplikacje, podobne informacje można uzyskać z tworzonych automatycznie plików dziennika (logu) lub audytu,

- modyfikacja kodu aplikacji systemów obsługi bieżącej, aby przygotowywać dogodne, specjalnie spreparowane dla hurtowni dane,
- najmniej wydajnym, lecz czasem jedynym możliwym do zastosowania, sposobem jest tworzenie migawek bazy danych i porównywanie ich w celu detekcji zmian.

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

2.6. Agregatyzacja danych

W hurtowni danych typowe zapytania polegają na podsumowywaniu, czyli agregowaniu dużych ilości danych. W celu zoptymalizowania czasu odpowiedzi na najczęciej zadawane zapytania, prócz danych elementarnych, przechowuje się też dane zagregowane.

Istnieją dwie podstawowe strategie dotyczące przechowywania danych sumarycznych (zagregowanych) w hurtowni danych:

1. Tworzenie nowych tabel faktów dla agregatów. Projektowane są nowe tabele wymiarów, odpowiednie dla wyższego poziomu ziarnistości danych oraz tabele faktów dla wszystkich poziomów agregacji (po wszystkich rozmiarach). W każdej z tabel faktów reprezentującej dane sumaryczne należy zdefiniować w każdym z agregowanych wymiarów dodatkowe, sztuczne klucze. Zalety takiego rozwiązania są następujące :

- żadna z aplikacji nie będzie podwójnie zliczała wartości, jak może to mieć miejsce przy alternatywnym podejściu - dodaniu pól zawierających poziom agregatyzacji do wspólnej tabeli faktów dla danych szczegółowych i zagregowanych,
- tabele z agregatami mogą być odrębnie tworzone, usuwane, ładowane i indeksowane, to rozwiązanie nie tworzy żadnych dodatkowyh rekordów, pól i kluczy w stosunku do alternatywnego,
- upraszcza strukturę metatabeli,
- prostszy jest wybór kluczy tabeli wymiarów zagregowanych,
- można łatwiej dopasować rozmiary dla pól agregatów, zwiększające się wraz z poziomem agregacji,
- łatwiejsze jest zarządzanie indeksami,
- w odpowiednio zaprojektowanej hurtowni danych użytkownicy końcowi i twórcy aplikacji nie muszą uwzględniać istnienia tabel agregatów dzięki nawigatorowi agregatów - dodatkowej warstwie oprogramowania, omówionej w dalszym ciągu niniejszego opracowania.

2. Tworzenie pól zawierających informację na temat poziomu agregatyzacji danych we wspólnej tabeli faktów. Agregaty znajdują się wówczas w podstawowej tabeli faktów,

korzysta się też z podstawowych tabel wymiarów, uzupełniając je odpowiednio o znaczniki informujące, czy dane reprezentujące fakty są szczegółowe czy zagregatyzowane oraz podające poziom ich agregacji. To rozwiązanie generuje dokładnie tę samą liczbę rekordów, jak w przypadku odrębnych tabel dla poszczególnych poziomów agregacji.

Wielką wadą tego rozwiązania jest wymóg, by każde zapytanie w systemie zawierało warunek nałożony na każdy z poziomów w każdym wymiarze. W przeciwnym wypadku może dojść do wielokrotnego zliczania danych.

Przy przechowywaniu danych sumarycznych należy pamiętać, że ilość danych zawartych w hurtowni bardzo szybko przyrasta, a wobec problemu rzadkości wypełnienia tabel należy dokładnie przemyśleć konieczność gromadzenia nadmiarowych danych, jakimi są agregaty. Dlatego zaleca się, by każdy rekord z danymi sumarycznymi powstawał na podstawie co najmniej 10, a najlepiej 20 i więcej rekordów z danymi szczegółowymi.

Dla uniezależnienia narzędzi do generowania raportów oraz oprogramowania od konieczności wprowadzania zmian wraz z pojawianiem się lub zmianami w obrębie tabel agregatów buduje się czasem dodatkową warstwę oprogramowania zwaną nawigatorem agregatów. Jego działanie polega na automatycznej konwersji zapytań użytkownika, formułowanych tylko dla podstawowej tabeli faktów (i tabel wymiarów) w taki sposób, by były przekierowywane do tabel z agregatami na odpowiednim poziomie.

Rolą tej warstwy oprogramowania jest również zbieranie statystyk dotyczących zapytań użytkownika, w celu określenia potrzeb zbudowania nowych tabel agregatów lub usunięcia zbędnych, rzadko używanych.

Nawigator agregatów powinien być uruchmiony na osobnym komputerze, widoczny wspólnie dla wszystkich użytkowników, ewentualnie razem z serwerem bazy danych. Statystyki prowadzone przez nawigatora agregatów ułatwiają podjęcie decyzji, które dane w systemie powinny być średnio, a które wysoko zagregatyzowane.

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.

2.7. Metody indeksowania w hurtowniach danych

Cechą charakterystyczną zapytań w systemach obsługi bieżącej jest przetwarzanie małej liczby rekordów lub operowanie na pojedynczych rekordach, zaś warunki selekcji zazwyczaj są proste. Tradycyjne metody indeksowania, takie jak indeksowanie za pomocą B-drzew czy tabel mieszających pozwalają w efektywny sposób uzyskać wyniki dla tego typu zapytań.

Specyfika pytań w hurtowniach danych jest inna, często przedmiotem analiz jest znaczny zbiór rekordów, warunki selekcji mogą być skomplikowane. Tradycyjne metody indeksowania nie są wystarczająco wydajne w takich przypadkach.

Indeksy bitmapowe pozwalają w sposób bardziej efektywny wykonywać zapytania typowe dla hurtowni danych.

Działanie indeksu bitmapowego zostanie przedstawione na przykładzie. Załóżmy, że tworzymy indeks dla kolumny Kol1 tabeli Tab1. Tabela ta ma 10 000 000 wierszy, w kolumnie Kol1 znajduje się N rożnych wartości danej - W_0, W_1, W_2, ..., W_{N-1}. Tworzenie indeksu polega na zbudowaniu N map bitowych, po jednej dla każdej różnej wartości przechowywanej danej. Każda mapa bitowa ma 10 000 000 pozycji. Załóżmy, że rozpatrujemy mapę bitową dla wartości W_2. Każda pozycja mapy bitowej zawiera 0 lub 1 oraz wskaźnik na odpowiadający mu rekord. Jeżeli dana pozycja wskazuje na rekord, w którym wartość w kolumnie Kol1 jest równa W_2, wówczas w tej pozycji mapy bitowej znajduje się 1. Jeżeli dany rekord zawiera inną wartość, to pozycja ta zawiera 0. Taka budowa indeksu pozwala na szybkie wyznaczanie rekordów będących odpowiedzią na zapytania zawierające rozbudowane warunki z operatorami AND i OR. Pozwala także na szybkie umiejscowienie w bazie danych wielu rekordów będących wynikiem odpowiedzi na zapytanie (wczytanie do pamięci właściwej mapy bitowej, po czym odpowiednich rekordów).

Wadą tego typu indeksu jest jego wielkość w przypadku bardzo dużego zróżnicowania wartości w kolumnie, na której jest zakładany (ekstremalnie jest to widoczne dla wartości unikalnych). Z tego powodu w praktycznych rozwiązaniach stosuje się tradycyjne indeksy dla bardzo zróżnicowanych danych, a także kombinacje indeksów tradycyjnych oraz bitmapowych. W celu zmniejszenia wielkości indeksów stosuje się ich kompresję (nadają one się bardzo dobrze do tego celu ze względu na swoją zero-jedynkową naturę), co znacznie zmniejsza ich wielkość, wydłuża niestety czas operacji na nich [7]. Na rynku znajdują się jednak systemy komercyjne wykonujące operacje algebry Boole'a na tego typu indeksach w formie spakowanej.

W systemach hurtowni danych stosuje się strukturę gwiazdy do przechowywania danych. W celu szybszego dostępu do danych przechowywanych w takiej strukturze wykorzystuje się indeksy przedstawiające strukturę wzajemnych powiązań między danymi w gwieździe (join indexes).

W systemach typu hurtownie danych odczyt danych następuje często kolumnami, nie zaś wierszami, jak w systemach obsługi bieżącej. Z tego powodu systemy przechowujące dane na dysku kolumnami mogą być bardziej efektywne. W szczególności można tu wspomnieć o systemach przechowujących wszystkie dane w indeksach.

2.8. Aplikacje analityczne

W systemach obsługi bieżącej rolą aplikacji jest wspomaganie bieżącego przetwarzania danych, czyli wprowadzanie i proste przetwarzanie danych. Przetwarzanie to często ogranicza się do operacji na pojedynczych rekordach.

W hurtowniach danych analiza dotyczy często bardzo dużej ilości danych, dane te podlegają zazwyczaj operacjom agregatyzacji, a niewielkie zbiory wynikowe są przedstawiane w sposób przejrzysty. Aplikacje służące do analizy danych w hurtowni prezentują dane w sposób umożliwiający łatwe wykonywanie operacji na modelu wielowymiarowym, co pozwala użytkownikowi na dostęp do informacji potrzebnych do podejmowania decyzji strategicznych.

Aplikacje te umożliwiają m.in. wykonywanie operacji uszczegółowiania i uogólniania danych oraz ich przedstawiania w różnych perspektywach. Możliwe jest także wykonywanie bardzo skomplikowanych obliczeń i porównań danych oraz prognozowanie i znajdowanie ukrytych związków między danymi.

W systemach wspomagających podejmowanie decyzji najczęściej zadaje się pytania typu:

- podaj 10 produktów najlepiej (najgorzej) sprzedawanych w ostatnim miesiącu,
- podaj sprzedaż produktów w poszczególnych regionach w zeszłym roku,
- podaj sprzedaż produktów w poszczególnych sklepach dla wybranego regionu w pierwszym tygodniu maja tego roku,
- porównaj sprzedaż produktu w kolejnych latach,
- wyznacz prognozę sprzedaży na kolejny rok,
- znajdź czynniki wpływające na zwiększenie (zmniejszenie) sprzedaży,
- podaj, jak sprzedaż jednego produktu wpływa na sprzedaż innych.

3. System SAS

System SAS jest zestawem narzędzi wspierającym proces przekształcania danych zawartych w różnego rodzaju systemach komputerowych w informację użyteczną dla podejmowania decyzji. Można wyróżnić kilka logicznych etapów takiego procesu, które

mogą być technicznie zorganizowane w aplikację, system oprogramowania lub hurtownię danych. Wszystkie te etapy wspierane są przez system SAS:

- dostęp do danych - import, eksport oraz bezpośredni dostęp do danych w wielu popularnych formatach, przechowywanie danych we własnych formatach SAS-a,
- przetwarzanie - zaawansowane narzędzia do przetwarzania danych - własny język, procedury, wsparcie dla języka zapytań SQL,
- analiza - narzędzia ekonometryczne, statystyczne i matematyczne, optymalizacja, modelowanie, symulacje,
- prezentacja - raportowanie tekstowe, graficzne, wizualizacja geograficzna, interaktywna analiza statystyczna i prezentacja wielowymiarowa,
- tworzenie aplikacji - narzędzia klasy RAD i VRAD do obiektowego tworzenia aplikacji.

System SAS działa na komputerach o różnych architekturach, pracujących pod kontrolą różnych systemów operacyjnych (m.in. Windows, Unix, VMS) i pozwala na przykład na:

- korzystanie z danych w sieci niezależnie od platformy sprzętowo-programowej,
- wykorzystanie mocy obliczeniowej komputerów pracujących w sieci,
- tworzenie oprogramowania na wybranym typie sprzętu, które może być potem uruchamiane na dowolnej innej platformie.

Na szczególną uwagę projektanta i użytkownika hurtowni danych zasługują, prócz wyżej wymienionych, takie funkcje i narzędzia systemu SAS, jak:

- MDDB - funkcje wspierające tworzenie i wykorzystywanie wielowymiarowych baz danych,
- DATA WAREHOUSE ADMINISTRATOR - narzędzie umożliwiające zarządzanie procesem tworzenia i użytkowania hurtowni danych w scentralizowany i uporządkowany sposób,
- bogactwo bibliotek i funkcji wspierających eksplorację danych (data mining).

3.1. Przechowywanie danych w systemie SAS

Narzędzia systemu SAS mogą wykorzystywać dane gromadzone w różnych formatach. Wspierany jest dostęp do danych w popularnych serwerach relacyjnych baz danych, takich jak m.in.: Oracle, Informix, Sybase, MS SQLServer, DB2, a także w plikach systemów klasy xBase (*.dbf). SAS posiada również własny format przechowywania informacji - zbiory danych (SAS datasets) oraz wielowymiarowe bazy danych (MDDB). Dane mogą być przechowywane w sposób heterogenicznie rozproszony - np. część danych w tradycyjnym serwerze, część w zbiorach SAS-a, a dane zagregowane w wielowymiarowej bazie danych.

Aby uporządkować w pewien sposób informacje na temat źródeł danych wykorzystywanych w systemie, wszelkie dane poddawane obróbce w systemie SAS przechowywane są w tzw. *Informacyjnej Bazie Danych* (IBD). Podstawowym elementem tej bazy danych jest biblioteka. Biblioteka może zawierać zbiory z danymi i tzw. katalogi, w ramach których występują obiekty inne niż zbiory danych. Pojęcie biblioteki jest analogiczne do pojęcia katalogu w systemie DOS/Windows. Biblioteka pozwala na logiczne grupowanie wielu obiektów z danymi. W bibliotekach przechowuje się obiekty różnych typów: DATA - zbiór danych, VIEW - perspektywa, CATALOG - katalog z różnymi obiektami informacyjnymi nie będącymi tabelami, MDDB - wielowymiarowa baza danych, PROGRAM - skompilowana postać programu w języku SAS 4GL DATA-STEP, ACCESS - opis dostępu do danych zapisanych w innych formatach. Katalogi są miejscem przechowywania informacji z danymi w różnych formatach, nie będących tabelami z danymi typu DATA, VIEW, ACCESS, MDDB. Katalog jest w przeciwieństwie do biblioteki obiektem fizycznym, a nie jedynie referencją. Skasowanie katalogu powoduje usunięcie wszystkich danych w nim zawartych. Zawartością katalogu mogą być m.in.:

- SOURCE - plik tekstowy,
- IMAGE - grafika rastrowa w formacie SAS,
- GRSEG - grafika wektorowa w formacie SAS,
- FRAME - aplikacja graficzna stworzona za pomocą SAS/AF,
- PROGRAM - aplikacja stworzona za pomocą SAS/AF,
- SCL - kod źródłowy dla aplikacji SAS/AF,
- EIS - aplikacja stworzona za pomocą SAS/EIS,
- FORMAT - mechanizm formatowania danych ,
- REPORT - definicja raportu dla procedury REPORT,
- CLASS - klasa dla obiektowych narzędzi SAS.

3.1.1. Zbiory danych w systemie SAS

Podstawowym sposobem przechowywania danych w systemie SAS jest umieszczanie ich w tabelach zwanych zbiorami danych. Zbiory danych umieszczane są w bibliotekach. Istnieją dwa podstawowe typy tabel - dane (DATA) i perspektywy (VIEW), nierozróżnialne z punktu widzenia ich wykorzystania przez język SAS 4GL. W tabelach można przechowywać teksty oraz informacje numeryczne (są tylko dwa typy danych - tekstowy i numeryczny), nie można natomiast bezpośrednio przechowywać grafiki, dźwięku, wideo itp. Obiekty tego typu przechowywane są w katalogach SAS-a lub w plikach zewnętrznych.

Zbiory danych w systemie SAS przypominają tabele relacyjne - zorganizowane są w wiersze i kolumny. Kolumny tabeli zwane są zmiennymi, a wiersze - obserwacjami. Do

kolumn odwołujemy się poprzez ich nazwy, do wierszy - bądź przez ich numer, bądź przez podanie tzw. zmiennej klucza. Wszystkie dane w jednej kolumnie muszą być tego samego typu. Dodanie nowych kolumn do tabeli wiąże się zwykle ze skopiowaniem całej zawartości tabeli; dodanie nowego wiersza powoduje dopisanie go na końcu istniejącej tabeli.

Zbiory danych składają się logicznie z dwóch obszarów: obszaru nagłówka oraz obszaru zawierającego dane. Perspektywy (zbiory typu VIEW) w obszarze zawierającym dane przechowują tylko opis uzyskania danych, np. sposób pobrania ich z innej tabeli, wyliczenie, itp. Zbiory tego typu integrują często dane pochodzące z kilku plików danych. Zazwyczaj poprzez perspektywy nie można modyfikować danych, ponieważ perspektywa zawiera tylko opis, a nie same dane (nie są to perspektywy zmaterializowane).

Zbiory danych mogą być indeksowane (jest tylko jeden rodzaj indeksów, oparty na strukturze B-drzewa) lub sortowane. Zbiory danych nie są udostępniane przez dodatkowe oprogramowanie, jak relacyjne bazy danych - za pośrednictwem oprogramowania serwera. Dlatego optymalizacja w dostępie do tych danych spoczywa właściwie na oprogramowaniu klienta. Ponieważ zbiory te przeznaczone są dla hurtowni danych - nie rozpatruje się tu problemu transakcji, czy współbieżnej aktualizacji danych. Dla zastosowań odpowiadających bazom danych klient-serwer stworzony został serwer SPDS - Scalable Performance Data Server, dostępny wyłącznie na platformy wieloprocesorowe. Serwer ten wspiera mechanizmy indeksowania bitmapowego, charakterystyczne dla hurtowni danych (patrz punkt 2.7).

Narzędzia klienta systemu SAS nie stawiają specjalnych wymagań co do struktury tabel, jednak pewne operacje znacznie prościej wykonuje się na pojedynczej tabeli, co warto mieć na uwadze przy projektowaniu logicznej struktury hurtowni.

3.1.2. Wielowymiarowe bazy danych

Jednym z komponentów systemu SAS jest serwer wielowymiarowych baz danych (SAS/MDDB Server). W systemie SAS wielowymiarowa baza danych tworzona jest najczęściej na podstawie istniejącego zbioru danych, w celu efektywniejszej wielowymiarowej analizy danych, z reguły z poziomu SAS/EIS. Dla niektórych operacji SAS/EIS tworzy w locie tymczasowe wielowymiarowe bazy danych. Wielowymiarowa baza danych zawiera dane zagregatyzowane.

Definiowanie wielowymiarowej bazy danych polega na określeniu:

- zbioru danych, na podstawie którego tworzymy wielowymiarową bazę danych,
- statystyk, które mają być przechowywane,
- zmiennych, na podstawie których obliczane są statystyki - fakty,
- zmiennych dzielących informacje na klasy - wymiary,
- hierarchii wymiarów.

Dane w wielowymiarowej bazie danych mogą być aktualizowane na podstawie bieżącej zawartości zbioru danych. Aktualizacja ta nie jest jednak automatyczna, to znaczy nie ma żadnego mechanizmu sprawdzającego na bieżąco zgodność zawartości wielowymiarowej bazy danych ze zbiorem, na podstawie którego powstała.

3.1.3. Dostęp do tradycyjnych, relacyjnych baz danych

Za mechanizmy dostępu do serwerów relacyjnych baz danych odpowiada moduł SAS/ACCESS. Istnieje również możliwość dostępu do zewnętrznych danych w standardzie ODBC.

Interfejs pomiędzy systemem SAS a popularnymi serwerami baz danych może być realizowany na różne sposoby:

- Za pomocą procedury ACCESS można stworzyć pliki-deskryptory relacyjnych tabel lub perspektyw dwojakiego rodzaju:
  - ⇒ access descriptors - przechowujące podstawowe informacje na temat struktury i atrybutów tabeli, do której chcemy uzyskać dostęp. Są to informacje, takie jak: dane na temat połączenia z bazą danych, nazwa tabeli, nazwy kolumn i ich typy. W pliku deskryptora mogą być również przechowywane odpowiednie dane dla systemu SAS - nazwy zmiennych, ich formaty itp. W typowym przypadku jeden plik odpowiada jednej tabeli relacyjnej, jest to główne źródło danych na temat tabeli dla systemu SAS, którego nie można jednak wykorzystać bezpośrednio w programach. Służy on głównie do stworzenia innego pliku, zwanego view descriptor.
  - ⇒ view descriptors - zawierające informacje pozwalające na wykorzystywanie danych z bazy relacyjnej - ich przeglądanie, modyfikację oraz ładowanie danych do zbiorów danych systemu SAS.
- Istnieje możliwość bezpośredniego użycia mechanizmu zwanego *Interface View Engine*, który jest wykorzystywany w sposób przezroczysty dla użytkownika w momencie tworzenia i używania plików *access descriptor* i *view descriptor*. Za pomocą procedury *SQL Pass-Through* można uzyskać bezpośrednie połączenie do bazy danych na relacyjnym serwerze oraz można wykorzystywać dialekt języka SQL charakterystyczny dla danego serwera w taki sposób, w jaki jest to realizowane z poziomu narzędzi klienckich tego serwera.
- Wykorzystując procedurę DBLOAD, można stworzyć nową tabelę na relacyjnym serwerze oraz wypełnić ją danymi pochodzącymi ze zbioru danych systemu SAS.

Ze względu na optymalizację dostępu do danych przy wyszukiwaniu w relacyjnych bazach danych używa się do pewnych celów mechanizmów plików deskryptorów, a do

innych - procedury *SQL Pass-Through*. Generalnie przewagą tego drugiego rozwiązania jest korzystanie w pełni z optymalizatora relacyjnego serwera, a przede wszystkim z plików indeksów dla przyspieszenia realizacji zapytań. Mechanizm ten stosuje się również w aplikacjach SAS/AF, w których odbywa się przetwarzanie transakcyjne danych relacyjnych.

Dostęp do relacyjnych baz danych wykorzystywany bywa zarówno w procesie ładowania danych z systemów obsługi bieżącej do hurtowni (wstępnego i okresowego), jak też spotyka się rozwiązania, w których większość danych w hurtowni przechowywana jest na specjalnie do tego przystosowanym serwerze relacyjnych baz danych (np. Oracle).

3.2. Przetwarzanie klient - serwer w systemie SAS

Oprogramowanie SAS wspiera różne modele przetwarzania klient-serwer, w środowisku heterogenicznych sieci komputerowych. Wymiana danych oraz podział logiki aplikacji pomiędzy różne systemy wspierane są przez następujące mechanizmy:

- Remote Computing Services - przetwarzanie zdalne - służące głównie do przetwarzania danych w miejscu (na komputerze), na którym dane rezydują, co może znacząco zminimalizować rozmiar danych przesyłanych w sieci. Dowolna część logiki aplikacji może być wykonywana zdalnie, na serwerach.
- Remote Library Services - przezroczysty dostęp do zdalnych źródeł danych (między różnymi platformami). Mechanizm ten pozwala na przesył danych poprzez sieć do stacji klienta, w celu ich obróbki lokalnej. W tym samym czasie inni użytkownicy na różnych platformach mają dostęp do tych samych danych. Rozwiązanie to jest korzystne dla niedużych zbiorów danych i aplikacji transakcyjnych.
- Data Transfer Services - pozwala na przesyłanie danych punkt-punkt (peer-to-peer) w postaci całych bibliotek, zbiorów SAS-owych, katalogów graficznych i zewnętrznych plików między komputerem lokalnym a systemami zdalnymi. Można realizować ładowanie i wyładowywanie danych w wyznaczonym czasie, w godzinach mniejszego obciążenia sieci. Można podzielić dane na serwerze, udzielając użytkownikom uprawnień do selekcji i modyfikacji wyłącznie danych związanych z poszczególnymi aplikacjami.
- DDE - Dynamic Data Exchange - pozwala aplikacjom systemu SAS wymieniać dane z każdym innym oprogramowaniem, wspierającym standard DDE (np. Lotus 1-2-3, MS Excel). Wymiana ta może być jednorazowa lub można ustalić trwałe połączenie dla dynamicznej modyfikacji danych.
- Wsparcie dla mechanizmów OLE - łącznie z automatyzacją OLE i OCX.

- Nowe narzędzia pozwalające na dostęp i zarządzanie danymi zewnętrznymi, na przykład za pomocą takich mechanizmów jak FTP, SOCKET i inne.

3.3. Tworzenie aplikacji w systemie SAS

O funkcjonalności hurtowni danych w dużym stopniu decydują aplikacje wykorzystywane przez końcowych użytkowników, realizujące dostęp do danych i ich prezentację. Narzędziami do obiektowego, szybkiego tworzenia aplikacji systemu SAS są SAS/EIS i SAS/AF.

3.3.1. SAS/EIS

Moduł SAS/EIS jest systemem do tworzenia i uruchamiania systemów informowania kierownictwa (Executive Information System). Tworzenie aplikacji za pomocą oprogramowania SAS/EIS odbywa się metodą "wskaż i kliknij" (point and click) i nie musi być realizowane przez wykwalifikowanych programistów, a przez końcowych użytkowników. Jest to tworzenie aplikacji bez konieczności pisania kodu źródłowego. Najważniejsze funkcje SAS/EIS to:

- raportowanie,
- graficzna prezentacja danych,
- wielowymiarowa prezentacja danych (operacje drill-down i roll-up),
- ekstrakcja danych do modelowania i analizy,
- raporty porównawcze,
- prognozowanie business'owe,
- analiza danych,
- możliwość połączenia wszystkich tych funkcji w spójną prezentację.

Obiekty systemu SAS/EIS (jest ich 37) pogrupowane są w następujące kategorie:

- wykresy business'owe,
- raporty business'owe,
- dostęp do danych,
- menu definiowane użytkownika,
- narzędzia,
- przeglądarki.

Narzędzia SAS/EIS przy dostępie do właściwych informacji korzystają z matedanych, w utworzonej specjalnie do tego celu metabazie. Tabele z danymi, które mają być prezentowane w aplikacjach, muszą być odpowiednio zarejestrowane w metabazie. Automatycznie w momencie rejestrowania zbioru przypisywane są następujące atrybuty:

- dla zbioru: nazwa zbioru danych, typ zbioru (dane czy perspektywa), nazwa biblioteki,
- dla kolumn: nazwy, typy i długości kolumn, etykiety, formaty wejściowe i wyjściowe, sposób użycia w aplikacji.

Dla wielowymiarowej analizy danych należy określić dodatkowy atrybut - hierarchia. Wartością tego atrybutu jest lista kolumn definiujących wymiar analizowania danych (np. wymiar CZAS, w skład którego będą wchodziły kolumny ROK, MIESIĄC i DZIEŃ).

3.3.2. SAS/AF

Oprogramowanie SAS/AF jest środowiskiem do tworzenia graficznego interfejsu użytkownika, dokładniej dostosowanego do jego potrzeb. Typowe wykorzystanie SAS/AF to:

- pisanie wolnostojących, sterowanych za pomocą menu, aplikacji pobierających, manipulujących i zachowujących dane, tworzących raporty i prezentujących graficznie dane,
- tworzenie systemów menu, kontrolujących ścieżki wykorzystania aplikacji przez użytkownika końcowego,
- projektowanie okien sprawdzających poprawność informacji oraz okien pobierających informacje do przetworzenia przez programy i powodujących start przetwarzania,
- projektowanie okien pomocy dla użytkownika, skojarzonych z oknami lub polami aplikacji,
- rozszerzenie możliwości oprogramowania SAS/EIS.

3.4. SAS/Warehouse Administrator

SAS/Warehouse Administrator jest narzędziem służącym do budowy i utrzymywania wielu hurtowni danych i składnic danych. Dzięki temu narzędziu zarządzanie hurtownią jest scentralizowane, choć fizyczna hurtownia może być rozproszona.

Oprogramowanie to umożliwia:

- generowanie, przechowywanie i przeglądanie metadanych,
- szeregowanie zadań i procesów,
- wizualne definiowanie logicznej struktury hurtowni.

Poprzez przyjazny dla użytkownika interfejs możemy:

- definiować encje danych, atrybuty i relacje. Relacje mogą być definiowanie pomiędzy:
  - ⇒ danymi operacyjnymi a danymi w hurtowni,
  - ⇒ elementami hurtowni,
- uzyskiwać dostęp do danych operacyjnych,

- przeprowadzać i planować ekstrakcję i transformację danych,
- definiować fizyczne lokalizacje dla danych z hurtowni,
- ładować, przechowywać i agregatyzować dane,
- tworzyć składnice danych, wielowymiarowe obiekty zawierające podsumowania, składnice informacji,
- korzystać ze wspomagania generacji kodu źródłowego.

SAS/Warehouse Administrator posiada dwa główne interfejsy: okno Explorer i Process Editor. Większość atrybutów definiujemy w oknie Explorer, które umożliwia wyświetlanie i edycję metadanych. Process Editor umożliwia definiowanie dodatkowych właściwości elementów oraz definiowanie procesów ładujących dane do hurtowni.

SAS/Warehouse Administrator jest zorganizowany wokół zbioru elementów. Można wyróżnić następujące rodzaje elementów:

- Hurtownia danych - zawiera m.in. metadane dla wszystkich elementów hurtowni. Hurtownia może zawierać pewną liczbę tematów i grup składnic danych.
- Temat - jest logicznym tematycznym zgrupowaniem danych szczegółowych, poziomów podsumowań i składnic informacji.
- Dane szczegółowe - są danymi pobranymi ze źródeł danych operacyjnych i przechowywanymi w jednej lub więcej tabelach w hurtowni danych.
- Poziomy podsumowań - są tabelami agregatów lub wielowymiarowymi danymi (SAS MDDB) pochodzącymi z tabel szczegółów i posiadającymi atrybut czasu.
- Elementy składnicy informacji - elementy, które zawierają lub wyświetlają informacje generowane z danych pochodzących z tabel szczegółów lub poziomów podsumowań.
- Składnice danych - zorganizowane tematycznie podzbiory danych hurtowni ukierunkowane na specjalne potrzeby poszczególnych końcowych użytkowników.

Każdy element posiada zdefiniowany przez administratora hurtowni zbiór atrybutów. Wiele elementów hurtowni może być powiązanych w grupy tworzące nowe elementy hurtowni. Niektóre z nich są hierarchicznie powiązane, co umożliwia dziedziczenie wartości pewnych atrybutów. W celu budowy i ładowania hurtowni danych definiuje się procesy, które wiążą te elementy w pewną całość. Większość elementów definiowana jest w oknie Explorer, a procesy definiowane są w oknie Process Editor.

Dane źródłowe z transakcyjnych i operacyjnych baz danych przechowywane są w Definicjach Danych Operacyjnych (ODDs). Środowiskiem jest kartoteka, która przechowuje metadane współdzielone pomiędzy wiele hurtowni i ODDs.

Istnieje pięć głównych kroków budowy hurtowni przy użyciu narzędzia SAS/Warehouse Administrator:

1. Zdefiniowanie środowisk(-a) - środowisko jest swego rodzaju repozytorium zawierającym współdzielone metadane, takie jak definicje hostów, dostępne SZBD, rodzaje połączeń z SZBD. W dalszych etapach projektu hurtowni w środowisku zostaną zdefiniowane elementy hurtowni danych oraz definicje procesów pozyskiwania danych z systemów transakcyjnych.
2. Zdefiniowanie danych wejściowych do hurtowni danych - danymi wejściowymi dla hurtowni danych mogą być pliki zewnętrzne (np. pochodzące z ekstrakcji danych z systemów OLTP), tabele lub perspektywy systemu SAS, które przechowują dane ze źródeł transakcyjnych i operacyjnych. Dla tych tabel i perspektyw określa się miejsce rezydowania oraz ich strukturę.
3. Zdefiniowanie elementów zawierających dane szczegółowe i zagregatyzowane - krok ten obejmuje definicję takich elementów, jak hurtownie danych, tematy, logiczne tabele szczegółów, tabele szczegółów, tabele agregatów, tabele wielowymiarowe, składnice danych.
4. Zdefiniowanie procesów potrzebnych do przeniesienia danych do hurtowni - SAS/Warehouse Administartor umożliwia mapowanie danych, definiowanie procesów pozyskiwania danych z systemów transakcyjnych, procesów transformacji danych. Umożliwia także szeregowanie tych procesów oraz generację potrzebnego kodu źródłowego.
5. Ładowanie hurtowni - w tym kroku przygotowane dane zostają załadowane do wcześniej przygotowanych elementów hurtowni danych, takich jak logiczne tabele szczegółów, tabele szczegółów, tabele agregatów, tabele wielowymiarowe, składnice danych.

4. System Oracle

Innym przykładem zestawu narzędzi do tworzenia wielowymiarowych baz danych, ich administracji oraz efektywnego wykorzystania informacji w nich zawartych są narzędzia Oracle.

4.1. Przechowywanie danych w hurtowniach danych opartych na narzędziach Oracle

W hurtowniach danych wykorzystujących narzędzia firmy Oracle dane mogą być przechowywane w strukturach wielowymiarowych, relacyjnych lub częściowo w jednych i drugich.

4.1.1. Express Server

Serwer wielowymiarowej bazy danych firmy Oracle nosi nazwę Express Serwer. Serwer ten może być ładowany danymi z trzech źródeł:

- pliki tekstowe,
- SQL-owe bazy danych,
- pliki w formacie EIF.

Express Interchanged Format (EIF) [9] umożliwia bezpośrednią wymianę danych wielowymiarowych.

Czynności administracyjne związane z bazą danych mogą być wykonywane za pomocą języka poleceń Express'a. Język ten umożliwia między innymi definiowanie obiektów bazy, kontrolowanie sposobu przechowywania danych (wyliczane, czy przechowywane stale), tworzenie bazy składającej się z kilku plików (może to przyspieszyć odczyt danych), kontroli dostępu do bazy (na poziomie całej bazy oraz jej obiektów).

W bazie przechowywane są obiekty, z których najważniejsze to: wymiar, zmienna, relacja, formuła [9].

W celu ograniczenia rozmiaru zajmowanej przestrzeni dyskowej przy istnieniu danych rzadkich należy postępować różnie w zależności od tego, czy określona dana nigdy nie występuje dla określonych, znanych, kolejnych wartości pewnych wymiarów, czy występowanie wartości pustej jest przypadkowe – dana nie występuje dla różnych, przypadkowych wartości wymiarów. W pierwszym przypadku wymiar, dla którego pewnych wartości dana nie występuje, powinien być zdefiniowany jako ostatni. Związane jest to ze sposobem zapisu danych na dysku. W drugim przypadku należy zadeklarować daną jako rzadką, co spowoduje efektywniejsze zarządzanie przestrzenią dyskową [9].

4.1.2. Serwer relacyjnej bazy danych

Dane w hurtowniach danych mogą być przechowywane w SZBD zamiast w wielowymiarowej bazie danych. Zaletą tego rozwiązania jest możliwość wykorzystania serwerów relacyjnych baz danych, które zazwyczaj są bardziej dopracowane.

Narzędziem firmy Oracle służącym do projektowania aplikacji i struktury baz danych jest pakiet typu CASE – Oracle Designer/2000. Pakiet ten można także wykorzystać do projektowania struktury hurtowni danych. Narzędzie to umożliwia wygenerowanie i przetwarzanie metadanych opisujących hurtownie danych.

Ładowanie danych do hurtowni w przypadku zastosowania serwera relacyjnego może być wykonane za pomocą:

- programu SQL*Loader,
- narzędzia do importu danych (imp73.exe, exp73.exe),
- programu Oracle Open Gateway.

SQL*Loader umożliwia ładowanie danych z plików tekstowych i binarnych. Przed uruchomieniem tego programu należy przygotować plik kontrolny opisujący źródło danych, miejsce ich przeznaczenia oraz zachowanie programu w razie wystąpienia błędów.

Narzędzie do importu danych służy do czytania skompresowanych plików binarnych stworzonych przez eksport danych z serwera Oracle. Narzędzia do importu i eksportu umożliwiają wymianę danych między dwoma serwerami Oracle.

Oracle Open Gateway służy do pobierania danych z baz innych niż bazy danych zarządzane przez serwer Oracle. Aktualnie dane można pobierać miedzy innymi z takich systemów, jak: DB2, MVS, Informix, Sybase, Teradata. Narzędzie to służy nie tylko do pobierania danych, ale także do ich transformacji.

Serwer bazy danych zarządzający hurtownią danych musi spełniać kilka warunków, z których najważniejszym jest umiejętność operowania na bardzo dużych ilościach danych. Produkty firmy Oracle w wersji 8 jak i 7.3 spełniają ten warunek. Innymi cechami sprzyjającymi zastosowaniu tych serwerów do budowy hurtowni danych są:

- indeksy bitmapowe,
- złączenia z wykorzystaniem tablic mieszających,
- zapytania typu gwiazdy,
- przestrzenie danych tylko do odczytu,
- przetwarzanie równoległe,
- partycjonowanie danych.

Wszystkie te cechy pozwalają na przyspieszenie operacji wyszukiwania w bardzo dużych zbiorach danych.

4.2. Relational Access Manager

Relational Access Manager umożliwia udostępnienie hurtowni danych utworzonej na serwerze relacyjnym narzędziom przystosowanym do pracy z danymi wielowymiarowymi.

Składa się on z trzech elementów [10]:

- Relational Access Administrator – narzędzie to pozwala zdefiniować wielowymiarowy model danych oraz określić, jak Express będzie pobierał dane z RBMS,
- Build Module – pozwala stworzyć bazę danych Exress'a lub uaktualnić istniejącą,
- Runtime module – służy do pobierania danych z systemu relacyjnego na bieżąco.

Wykorzystanie narzędzia polega na kolejnym wykonaniu trzech następujących kroków:

- definicji,
- budowy,
- pracy.

W fazie definicji użytkownik określa strukturę modelu wielowymiarowego oraz mapowanie jego elementów do tabel relacyjnych. W fazie następnej tworzona jest wielowymiarowa baza danych. W ostatniej fazie Relational Access Manager udostępnia tę bazę aplikacją z rodziny Express'a.

W wielowymiarowej bazie danych mogą być przechowywane nie tylko obiekty modelu wielowymiarowego oraz mapowania, ale także same dane, gdy zostanie wybrany hybrydowy tryb pracy.

Nie każdą strukturę tabel relacyjnych można powiązać z obiektami modelu wielowymiarowego. W najbardziej naturalny sposób nadaje się do tego struktura gwiazdy, czy będąca jej rozszerzeniem struktura płatka śniegu[10]. Pola tabeli faktów w modelu gwiazdy można przyporządkować zmiennym, a pola tabel wymiarów wymiarom.

Przyspieszenie dostępu do tabel relacyjnych zostało osiągnięte dzięki zastosowaniu pamięci podręcznej. Jeśli wymaganych danych w niej nie ma, następuje generacja odpowiedniego zapytania SQL-owego, które zostaje wykonane w serwerze relacyjnym.

4.3. Analiza danych

W zależności od zastosowanych narzędzi dane aplikacji mogą być udostępnione w postaci wielowymiarowej lub relacyjnej.

4.3.1. Analiza danych relacyjnych

Stworzona z wykorzystaniem relacyjnego serwera hurtownia danych może być przeglądana za pomocą programu Oracle Discoverer. Program ten należy do grupy narzędzi wspomagających podejmowanie decyzji. Umożliwia on zadawanie zapytań oraz tworzenie raportów bez znajomości języka SQL, wyniki przeszukiwań są przedstawione w przejrzystej i czytelnej formie.

Zanim użytkownik będzie mógł korzystać z pakietu, administrator musi stworzyć warstwę użytkownika końcowego. Warstwa ta składa się z tzw. obszarów działania, folderów i elementów. Obszar działania jest zbiorem informacji o celu biznesowym użytkownika. Foldery przechowują informację o podgrupach w ramach obszaru działania i składają się z elementów.

Każdy użytkownik, bądź grupa użytkowników, mogą mieć własną warstwę użytkownika końcowego, która im oferuje intuicyjny, skupiony na ich obszarze tematycznym widok bazy danych.

Aplikacja umożliwia przeprowadzanie analizy wielowymiarowej, obsługuje techniki uszczegółowienia i uogólnienia (zwijanie i rozwijanie). Discoverer 3.0 wykorzystuje indeksy bitmapowe oraz zapytania typu gwiazdy [11]. W czasie gdy użytkownik pracuje z jednym ze-

stawem danych, może być wykonywane inne zapytanie. Istnieje możliwość przerwania zbyt długo wykonywanego zapytania. Dane będące wynikiem zapytania są przechowywane w pamięci podręcznej na stacji roboczej. Oracle Discoverer zarządza nimi kompresując je i indeksując; zwiększa to efektywność wykonywania operacji charakterystycznych dla struktur wielowymiarowych [11]. Wykorzystanie tabel z wartościami wstępnie zagregatyzowanymi zwiększa także efektywność. Program bez wiedzy użytkownika wybiera tabelę, do której ma być kierowane zapytanie. Może to być tabela z danymi szczegółowymi, gdy nie ma tabel z agregatami, lub jedna z tabel z danymi sumarycznymi. O wyborze decyduje przewidywana szybkość wykonania zapytania oraz możliwość spełnienia warunków zapytania.

4.3.2. Analiza danych wielowymiarowych

Firma Oracle dostarcza trzech klas narzędzi służących do analizy danych wielowymiarowych.

W pierwszej kasie można wymienić pakiet Oracle Express Objects, który jest wizualnie zorientowanym środowiskiem tworzenia aplikacji operujących na danych przechowywanych w strukturach wielowymiarowych.

W drugiej klasie znajduje się Oracle Express Analyzer, będący uniwersalnym narzędziem do analizy danych wielowymiarowych. Narzędzie to służy do raportowania i analizy strategicznej.

Trzecią klasę tworzą trzy gotowe, wyspecjalizowane aplikacje:

- Oracle Financial Analyzer,
- Oracle Financial Controller,
- Oracle Sales Analyzer.

W dziedzinie data mining Oracle opiera się na produktach firm należących do Oracle's Warehouse Technology Initiative (np. NeoVista i DataMind).

5. Podsumowanie

W pierwszej części artykułu przedstawiono przegląd podstawowych różnic koncepcyjnych między bazą danych a hurtownią danych (tablica 1). Zaprezentowano kolejne etapy projektowania, implementowania i utrzymania hurtowni danych, uwzględniając podstawowe modele logicznej i fizycznej struktury danych przechowywanych w hurtowni (rys. 1, rys. 2), nowe metody indeksowania danych i problemy występujące przy zasilaniu hurtowni danymi. Artykuł zawiera również opis typowych rozwiązań teoretycznych w zakresie przechowywania w hurtowni danych zagregowanych.

Wskazano podstawowe wymagania dotyczące aplikacji analitycznych i najczęściej zadawane zapytania w systemach wspomagania podejmowania decyzji.

Kolejne dwie części opracowania zawierają skrócony opis dwóch przykładów praktycznych rozwiązań w dziedzinie hurtowni danych - SAS i Oracle. Przedstawiono sposoby przechowywania danych w tych systemach oraz podstawowe funkcje narzędzi do tworzenia aplikacji.

LITERATURA

1. "Wstęp do systemu SAS" - materiały kursowe, SAS Institute, 1997.
2. "Building a Data Warehouse Using SAS/Warehouse Administrator Software" - Course Notes, SAS Institute 1997.
3. "Building a Data Warehouse Using SAS System" - Course Notes, SAS Institute 1996.
4. "Przetwarzanie Danych" - materiały kursowe, SAS Institute, 1997.
5. "Szybkie tworzenie aplikacji w środowisku SAS/EIS" - materiały kursowe, SAS Institute, 1997.
6. Irek P., Kmonk J., Pierzchała D.: Projekt hurtowni danych na wybranym przykładzie; Wpływ agregatyzacji danych na efektywność realizacji zadań wyszukiwania, ZN Pol. Śl., s. Informatyka z. 34 Gliwice 1998.
7. Edelstein H.: Technology Analysis: Faster Data Warehouses, publikacja elektroniczna 1995.
8. Bontempo Ch. J., Saracco C. M.: Accelerating Indexed Searching, publikacja elektroniczna 1997.
9. Oracle Express. Database Administration Guide - Oracle Corporation 1997.
10. Oracle Express. Relational Access Manager. User's Guide - Oracle Corporation 1996.
11. Oracle Discoverer/2000 - Oracle Corporation 1995.
12. Inmon W.H.: Bulding the Data Warehouse, Wiley Computer Publishing 1996.
13. Kimball R.: The Data Warehouse Toolkit, Wiley Computer Publishing 1996.

Recenzent: Dr hab. inż. Stanisław Wołek Prof. Pol. Rzeszowskiej

Wpłynęło do Redakcji 15 stycznia 1999 r.

Abstract

In the first section of the paper the review of fundamental differences between database and data warehouse is presented (table 1). The steps of projecting, implementation and maintenance of data warehouse, including the basic logical and physical structure's models of data residing in data warehouse (fig. 1, fig. 2), new methods of indexing data, the problems with feeding data warehouse with informations are shown. The paper includes description of typical theoretical solving of storage the aggregates in data warehouse.

The fundamental role of OLAP applications and most frequently asked queries are indicated.

The next two sections are descriptions of practical implementation's examples in data warehousing- SAS and Oracle systems. The way of data storage and basic functions of application development tools in this systems are discussed.