

Aleksandra WERNER
Politechnika Śląska, Instytut Informatyki

TRANSPOZYCJA TABEL TEMPORALNEJ BAZY DANYCH Z MODELU ZDARZEŃ DO MODELU STANU

Streszczenie. Artykuł zawiera podstawy wiedzy z zakresu modeli danych temporalnych i prezentuje możliwości zmiany modelu zdarzeń tabel bazy danych do modelu stanu, wskazując jednocześnie powody przeprowadzenia takiej zmiany. Dodatkowo zaprezentowano w nim wpływ przyjętego modelu tabel na czas realizacji zapytań. Podłożem dla wykonanych eksperymentów stał się fragment rzeczywistej (tzn. aktualnie eksploatowanej) relacyjnej bazy danych, obejmujący swym zakresem farmakoekonomikę szpitala.

Słowa kluczowe: temporalna baza danych, model zdarzeń, model stanu, baza danych *SZPITAL*.

TEMPORAL DATABASE TABLES TRANSITIONS BETWEEN EVENT MODEL AND STATE MODEL

Summary. The fundamental concepts of temporal data models and the possibilities and reasons of title conversion are contained in this paper. Additionally the model influence to the time of executing sample queries is presented in this article. All tests were made on sample real-world *HOSPITAL* database.

Keywords: temporal database, event model, state model, *HOSPITAL* database.

1. Modele danych temporalnych

Według definicji przyjętej w 1992 r. podczas Międzynarodowych Warsztatów ARPA/NSF w Arlington poświęconych infrastrukturze temporalnych baz danych, baza danych wspierająca kilka aspektów czasu, z wyjątkiem takiego czasu będącego jednym z atrybutów bazy, od którego pozostałe atrybuty nie są zależne (tzw. czasu użytkownika), jest **bazą temporalną** (ang. *TDBMS*) [4]. Warto odnotować, że spotykany w literaturze termin

historyczna baza danych jest pojęciem węższym, gdyż sugeruje zachowywanie informacji jedynie o stanach przeszłych, podczas gdy baza temporalna może zachowywać informacje również o przyszłości: planach lub prognozach [5].

W literaturze przedmiotu spotyka się trzy zasadnicze modele danych temporalnych, różniące się między sobą sposobem ujęcia czasu: model „migawka”, model zdarzeń i model stanu [1]. W celu zwiększenia przejrzystości artykułu uściślenie tych terminów zostało poprzedzone definicjami podstawowych pojęć z dziedziny modeli danych.

1.1. Pojęcia podstawowe

Model danych może być rozumiany w kontekście zbioru ogólnych zasad przedstawiania i operowania danymi [7]. Zbiór ten obejmuje, jak podaje m.in. pozycja [4] literatury, trzy główne aspekty:

1. **Definicję danych:** zbiór reguł określających logiczną strukturę danych, w odróżnieniu od niższego poziomu organizacji zapisu stosowanego wewnątrz jądra bazy danych;
2. **Operowanie danymi:** komponent behawioralny, czyli zbiór reguł, dotyczących procesu dostępu do danych i ich modyfikacji;
3. **Integralność danych:** zbiór reguł określających, które stany bazy danych są poprawne (a więc zarazem, jakie operacje prowadzące do modyfikacji danych są dozwolone).

Historycznie rozróżnia się trzy główne generacje modeli danych [9]:

- proste modele danych powstałe w latach 60. XX wieku z systemu plików, w których dane zorganizowane były w strukturę rekordów zgrupowanych w plikach; głównymi dostępnymi operacjami były operacje na rekordach lub ich polach,
- klasyczne modele danych, których rozkwit przypadał na lata 70. XX wieku (w przypadku modelu relacyjnego trwa nadal) i które zachowały podstawową orientację opartą na rekordach; za klasyczne modele danych uważa się obecnie hierarchiczny, sieciowy i – zasadniczo od nich odbiegający – model relacyjny,
- obecnie intensywnie rozwijane i wciąż ewoluujące semantyczne modele danych, dostarczające łatwiejszego niż w modelach poprzednich sposobu odczytywania informacji o semantyce danych (np. obiektowy i temporalny model danych).

1.2. Model zdarzeń (ang. *event model*) TDBMS

W modelu zdarzeń zachodzi rejestracja zdarzeń w czasie. Przez zdarzenie rozumiane jest wystąpienie (instancja) danych temporalnych w pewnej chwili czasu. Tabele zdarzeń (ang. *event tables*) są modyfikowane jedynie przez dodanie nowego zdarzenia (ang. *append only*),

a błędne zapisy są traktowane jako element historii tabeli. Każde zdarzenie (każdy wiersz tabeli) jest znacznikowane elementem temporalnym (czasem), żeby zapisać czas, w którym wystąpiło.

W modelu tym baza widziana jest jako sekwencja zdarzeń – czyli historia obiektu jest wyznaczona przez zbiór kolejnych zdarzeń.

1.3. Model stanu (ang. *state model*) TDBMS

Z definicji stan jest to instancja danych, które istnieją lub trwają przez pewien okres czasu. W modelu stanu zachodzi rejestracja stanu bazy w czasie – czyli baza widziana jest jako sekwencja stanów poszczególnych tabel. Każdy zapisany stan jest znacznikowany okresem czasu, w którym istnieje.

Model ten ma zastosowanie np. w bazach przechowujących informacje o ratach kredytowych, polisach ubezpieczeniowych, jak również jest pomocny przy wykonywaniu bilansów materiałowych. Ze względu na charakter modelu stanu wygodnie jest wiązać ten model z czasem transakcyjnym, w którym dane (fakty) są obecne w bazie (zapamiętane w bazie), niezależnie od tego, kiedy fakt miał miejsce w rzeczywistości.

2. Możliwości transpozycji tabel modelu stanu w tabeli modelu zdarzeń

Wybór modelu projektowanej bazy jest, z punktu widzenia przyśpieszenia realizacji, zapytań, najistotniejszą decyzją projektową. W modelu stanu np. obserwuje się znaczne przyśpieszenie wykonywania zapytań z daną wartością klucza lub zapytań typu zakresowego, podczas gdy w modelu zdarzeń zauważalnie szybciej realizowane są zapytania o stan bazy w chwili t , lub zapytania o historię atrybutu.

Przykładowymi zapytaniami, kierowanymi do przykładowej bazy danych zawierającej informacje o aktualnych zarobkach poszczególnych pracowników przedsiębiorstwa zajmujących pewne stanowiska, mogłyby być:

- *znajdź aktualną pensję pracownika o numerze 21,*
- *sprawdź datę awansu pracownika o numerze 2 na stanowisko „kierownik działu”,*
- *podaj liczbę pracowników zatrudnionych w przedsiębiorstwie w kwietniu 2001 r.,*
- *pokaż historię zarobków pracowników.*

Z uwagi na fakt, że temporalne bazy danych są zwykle implementowane przez nałożenie dodatkowej warstwy na już istniejący dobrze znany (np. relacyjny) system zarządzania bazą danych, struktura tabel składowych bazy wynika z reguły bezpośrednio z diagramu ER,

nieukierunkowanego przecież na utrwalanie informacji czasowej. W efekcie uzyskuje się często nieoptymalny model bazy, który po szczegółowej analizie najczęściej kierowanych do bazy klas zapytań może zostać zmieniony.

Uważa się, że model zdarzeń może być przekształcony do komplementarnego mu modelu stanu przez chronologiczne użycie kolejnych zdarzeń do konstruowania nowych, następujących po sobie stanów. Do osiągnięcia tego celu konieczna jest jednak znajomość stanów: początkowego i końcowego. Również przejście odwrotne jest możliwe do uzyskania przez proste wyliczenie różnic pomiędzy chronologicznie przyległymi (czyli graniczącymi) stanami.

Istnienie ścisłego związku między charakterem wczytywanych danych a sposobem ich zapisu w bazie może jednak w niektórych przypadkach uniemożliwiać zmianę modelu analizowanych tabel. Stwierdzono, że jeżeli dany fakt zachodzi w każdym podprzedziale rozpatrywanego interwału czasowego, wówczas dopuszczalne jest dowolne modelowanie tego faktu i dokonywanie żądanych transformacji między modelami [10]. Jeżeli jednak modelowana historia zdarzeń przyjmuje dyskretne wartości w określonych momentach czasu, wówczas proces zmiany modelu tabeli wymaga zmiany praktycznie wszystkich tabel analizowanej bazy.

3. Analiza przykładowej bazy danych

W badaniach posłużono się aktualnie eksploatowaną bazą danych SZPITAL, rejestrującą na bieżąco wydawanie/przyjmowanie leków. Rozpatrywana baza danych zawierała dane o miejscach składowania towarów w szpitalu, ilościach leków w magazynach (tzw. stanach magazynowych), pacjentach i tzw. ośrodkach powstawania kosztów¹ oraz kontrahentach (dostawcy towaru), zapisane w tabelach o wyszczególnionych poniżej schematach:

```
Dokumenty (id_dokumenty, id_rodzaj_dok, id_kontrahent, id_opk,  
           id_pob, nr_dokumentu, dt_wprowadzenia, dt_wystawienia,  
           dt_platnosci),  
Dostawy (id_dostawy, id_ms, id_towar, seria, dt_waznosci,  
         cena_netto, vat),  
Jednostki_miary_sl (id_jm, nazwa_jm, mnoznik_jm),  
Kontrahenci_sl (id_kontrahent, regon, nazwa_kontrahent, adres,  
               nip_kontr, identyfikator_ksiegowy),  
Magazyny_sl (id_magazynu, nazwa_magazynu, dt_start_mg,  
            dt_stop_mg),  
Miejsca_skladowania_sl (id_ms, id_magazynu, nazwa_ms, cechy_ms,  
                       dt_start_ms, dt_stop_ms),
```

```
Obroty (id_obroty, id_dokumenty, id_dostawy, ilosc, dt_obrotu,
        kierunek_dz),
Opk_sl (id_opk, nazwa_opk, dt_start_opk, dt_stop_opk),
Pacjenci_sl (id_pac, nazwisko, imie, plec, dt_urodzenia, nip_pac,
            pesel, dt_zgonu),
Pobyty (id_pob, id_pac, dt_start_pob, dt_stop_pob),
Postac_sl (id_post, nazwa_post),
Rodzaj_dok_sl (id_rodzaj_dok, nazwa_rodz, schemat_dzialania),
Towary_sl (id_towar, id_jm, id_post, ean, pkwiu, dawka,
           dt_rejestracji, nazwa_tow).
```

Dla bazy danych SZPITAL w jej początkowym kształcie nie można jednoznacznie określić modelu jej danych. Niektóre z tabel prezentowane są bowiem jako tabele zdarzeń (Dokumenty, Dostawy, Obroty), podczas gdy inne – jako tabele stanu (tabela Pobyt i częściowo tabele słownikowe). Co więcej, przeprowadzone w ramach pracy analizy wykazują, że jeżeli zapamiętywane w tabeli dane nie są ciągłe w czasie (okresy występowania pewnego stanu danych nie nakładają się na siebie), nie jest możliwe przekształcenie tabeli modelu stanu do modelu zdarzeń. Można to łatwo prześledzić na przykładzie tabeli *Magazyny_sl*. Załóżmy, że zaistniała konieczność utworzenia magazynu o numerze 25 na okres od 2002-05-30 do 2002-10-30. Nie ma gwarancji, że magazyn ten będzie jeszcze kiedykolwiek później wykorzystany, podobnie jak nie ma pewności, że w ogóle będzie istniał (z powodu braku wolnych pokoi, magazyn może ulec przekształceniu, np. w gabinet diagnostyczny). Dodatkowo, poszczególne magazyny mogą powstawać i być likwidowane niezależnie od siebie (dane nie są od siebie w żaden sposób zależne). Rezygnacja z obu atrybutów *dt_start_mag* i *dt_stop_mg* całkowicie uniemożliwiłaby zachowanie historii zdarzeń, a pozostawienie tylko jednego z atrybutów (np. *dt_start_mag*) niosłoby ze sobą błędną informację dla użytkownika: *magazyn nadal istnieje*. Tak więc przekształcenie tabeli słownikowej *Magazyny_sl* do modelu zdarzeń jest praktycznie niemożliwe ze względu na charakter zapamiętywanych w niej danych.

Podobna sytuacja występuje w tabeli *Miejsca_skladowania_sl*, chociaż jej struktura wstępnie wskazuje, że została zamodelowana w modelu stanu (atrybuty *dt_start_ms*, *dt_stop_ms*). Analogicznie do przedstawionego wcześniej przypadku, miejsca składowania w magazynie mogą ulegać okresowym zmianom: mogą być kupowane nowe lodówki, szafki, montowane dodatkowe półki lub likwidowane stare. Przez cały okres użytkowania cechy charakterystyczne miejsc składowania nie ulegają jednak zmianom (szafka stoi w miejscu nasłonecznionym, lodówka chłodzi do +4 stopni itd.). Najczęściej stosowaną w szpitalach praktyką jest nawet pomijanie wpisu danych do kolumny charakteryzującej cechy miejsca

¹ Ośrodki powstawania kosztów to odbiorcy towaru – np. poszczególne oddziały i pracownie szpitala.

składowania (atrybut `cechy_ms`), co wskazuje na nieistotność, a zarazem niezmiennosc w czasie tych danych. Brak będzie wobec tego zdarzeń, które, występując w określonym czasie, wpływają na zapisane w wierszu wartości atrybutów. Dane zapisane w tej tabeli informują o stanie wyposażenia magazynu w różnych, i co istotne nieciągłych, okresach czasu, tak więc również i dla tej tabeli niemożliwe będzie jej przekształcenie do modelu zdarzeń.

W przypadku tabel: `Pacjenci_sl` oraz `Towary_sl` sama obecność atrybutów czasowych (`dt_urodzenia`, `dt_zgonu`, `dt_rejestracji`) nie wystarcza do stwierdzenia, że są to tabele modelu zdarzeń (mogłoby to dotyczyć tabeli `Towary_sl`) lub modelu stanu (`Pacjenci_sl`). W modelu zdarzeń historia obiektu (tu leku) powinna być wyznaczona przez zbiór kolejnych zdarzeń, ale zauważmy, że atrybut `dt_rejestracji` nie jest znacznikiem czasowym, a jedynie czasem użytkownika. Zarejestrowany w pewnym czasie lek może oczywiście zostać kiedyś wycofany z użycia, ale informacja ta nie zostanie odnotowana w bazie. Podobnie samo istnienie dwóch atrybutów: intuicyjnie „początkowego” (`dt_urodzenia`) i intuicyjnie „końcowego” (`dt_zgonu`) w tabeli `Pacjenci_sl`, nie upoważnia do zaklasyfikowania jej jako tabeli modelu stanu. Dzieje się tak dlatego, że atrybut `dt_zgonu` jest wypełniany określoną wartością tylko w przypadku zgonu pacjenta podczas pobytu w szpitalu (jeżeli pacjent umrze po opuszczeniu szpitala, fakt ten nie jest odnotowywany w bazie). Tym samym stwierdzenie, że wspomniane wcześniej atrybuty czasowe znaczą stan osobowy pacjentów w szpitalu, będzie niezgodne z definicją, która mówi, że *stan jest to instancja danych, które istnieją lub trwają przez pewien okres czasu*.

Wniosek częściowy z przeprowadzonego rozumowania jest następujący: tabele słownikowe nie są dobrym materiałem do przekształceń, gdyż zwykle zawierają dane niewrażliwe na czas (ewentualnie mieszczą czas użytkownika, niezależny od pozostałych atrybutów – jak np. w tabeli `Towary_sl`). Nawet gdyby przyjąć założenie, że tabele słownikowe będą podlegać okresowym modyfikacjom (np. pacjent zmienił nazwisko, lek zmienił producenta itd.), zmiany te będą na tyle incydentalne, że nie będą prowadziły do nadmiernego (i szybkiego) rozrostu tabel – a tym samym nie będą znacząco obciążały systemu (wpływały na czas wykonania zapytania). Również takie tabele słownikowe, które zawierają czas ważności i zostały zamodelowane przykładowo jako tabele stanu (jak to jest np. w przypadku tabeli `Magazyny_sl`), nie powinny być uważane za tabele temporalne, gdyż mają z reguły charakter statyczny.

Newralgicznymi i kluczowymi punktami bazy, (o czym świadczy zarówno ilość zapisanych w nich danych, jak i szybkość ich przyrostu) są tabele `Obroty` i `Pobyty`. Zostaną więc przeanalizowane pod kątem doboru właściwego modelu dla wykonywanych na nich zapytań.

W tabeli `Pobyty` odnotowywany jest każdy pobyt pacjenta w szpitalu, jak również jego każdorazowa wizyta w przychodni przyszpitalnej (atributy `dt_start_pob`, `dt_stop_pob` przyjmują wtedy identyczne wartości). Celowe wydaje się więc odseparowanie danych o jednodniowych pobytach pacjentów na terenie przychodni przyszpitalnej od pozostałych danych, informujących o dłuższym pobycie pacjentów w szpitalu. Korzyści płynące z rozdzielenia danych wydają się dość oczywiste z punktu widzenia najczęściej eksplorowanych przez personel danych. Otóż pytania, dotyczące pacjentów przychodni stanowią jedynie niewielki procent ogółu zadawanych zapytań (pacjentom takim nie są np. wydawane żadne leki z magazynu), chociaż ich wkład w fizyczny rozrost tabeli jest identyczny jak w przypadku pacjentów hospitalizowanych. Jediną konsekwencją decyzji o podziale źródłowej tabeli `Pobyty` będzie ograniczenie w niej nadmiernego przyrostu rekordów, co z kolei pozwoli uniknąć kosztownych łączy np. z tabelą `Obroty`.

Tabela `Obroty` uczestniczy praktycznie w każdej transakcji, dokonywanej w ramach bazy – zarówno podczas wydawania czy przyjmowania zwrotnych leków dla pacjenta bądź wybranego oddziału, jak też przy dostawach towarów od kontrahentów. Można się więc spodziewać, że odpowiednie jej przekształcenie będzie kluczowym procesem, zmierzającym do poprawy efektywności pracy z bazą.

Z uwagi na fakt, iż tabela `Obroty` zawiera atrybut `dt_obrotu`, który niesie ze sobą wiadomość o chwili wystąpienia zdarzenia (przyjęcie dostawy lub wydanie leku w zależności od określonego atrybutem `kierunek_dz` kierunku transakcji), nasuwa się pytanie: *Jak przekształcić tę tabelę do modelu stanu, nie tracąc równocześnie informacji o okresowości występowania zdarzeń?*

3.1. Proponowane rozwiązanie problemu

Jediną możliwość przekształcenia modelu analizowanych tabel stwarza jednoczesna modyfikacja dwóch tabel: interesującej nas tabeli `Obroty` oraz skorelowanej z nią tabeli `Dostawy`.

Aby zrealizować to żądanie, wystarczy dokonać przesunięć wybranych atrybutów tabeli `Obroty` do tabeli `Dostawy`, przy jednoczesnej (niewielkiej) zmianie ich formy. I tak, data wystąpienia obrotu powinna zostać całkowicie wyeliminowana ze źródłowej tabeli `Obroty` i stać się jednym z atrybutów tabeli `Dostawy` (atribut `dt_obrotu`). Natomiast dwa kolejne atrybuty – `ilosc` i `kier_dzialania`, występujące pierwotnie w tabeli `Obroty` – muszą ulec transformacji, przechodząc jako ilość bieżąca towaru z danej dostawy do tabeli `Dostawy` (atribut `ilosc_biezaca`).

Utworzona poleceniem:

```
CREATE TABLE dostawy_stan AS
SELECT d.id_dostawy, d.id_ms, d.id_towar,
```

```

d.seria, d.dt_waznosci, d.cena_netto, d.vat, o.dt_obrotu,
(SELECT
    sum(o2.kierunek_dz * o2.ilosc)
FROM
    obroty o2, dostawy d2
WHERE
    o2.id_dostawy = d2.id_dostawy and
    d2.id_dostawy = d.id_dostawy and
    o2.id_obroty <= o.id_obroty)
AS ilosc_biezaca
FROM obroty o, dostawy d
WHERE o.id_dostawy = d.id_dostawy;

```

przekształcona tabela `Dostawy` zawiera teraz planowe informacje o stanie magazynowym towaru z danej dostawy na dany dzień.

3.2. Proponowane reguły realizacji zmiany modeli tabel

W miarę postępu prac stwierdzono braki w istniejących w literaturze wytycznych, dotyczących transformacji modeli tabel. Zawarte w literaturze reguły, dotyczące transformacji modeli tabel, zalecające „chronologicznego użycia kolejnych zdarzeń do konstruowania nowych, następujących po sobie stanów”, nie mogą sprostać zadaniu zmiany modelu w przypadku gdy do określenia pewnego stanu potrzebne jest sięgnięcie do wielu danych pochodzących z **różnych** tabel.

Z uwagi na to, w celu realizacji procesu transformacji źródłowego modelu tabeli, opracowano i następnie wykorzystano własny zestaw zasad, których treść jest następująca:

1. Wskaż w istniejącej bazie danych zestaw tych danych, które określają żądany stan – ich liczba będzie liczbą nietemporalnych atrybutów nowo tworzonej tabeli stanu.
2. Wyznacz zbiór tych danych, występujących w analizowanym stanie, które podlegają zmianom w czasie.
3. Wyodrębnij spośród danych opisujących rozpatrywany stan te, które nie zmieniają się w czasie.
4. Zdefiniuj zdarzenia, które poprzez modyfikację atrybutów zmieniających się w czasie, będą miały wpływ na zmianę stanu danych.
5. Jeżeli stan określany będzie w punktach czasu, poszerz powstałą tabelę stanu o pojedynczy atrybut czasowy, którego wartość będzie ustalana na podstawie czasu wystąpienia zdarzenia zmieniającego stan bieżący.
6. Jeżeli stan będzie znacznikowany okresem czasu, dodaj do tworzonej tabeli stanu dwie kolumny określające punkt początkowy i końcowy okresu istnienia stanu, przy czym:

- przy każdorazowym wystąpieniu zdarzenia zarejestruj (dopisz do tabeli) nowy stan z początkowym znacznikiem czasowym równym czasowi wystąpienia zdarzenia, a końcowym ustalonym na wartość „do nadal”, modyfikując jednocześnie znacznik końcowy stanu poprzedniego na czas wystąpienia zdarzenia pomniejszony o jeden chronon¹.
- 7. W przypadku wystąpienia któregośkolwiek ze zdarzeń wymienionych w pkt 4, określ nowy stan danych zmieniających się w czasie na podstawie wartości ich stanu poprzedniego i zdarzenia, które go zmieniło.
- 8. Dopisz nowy stan do tabeli, przepisując do pozostałych kolumn atrybutów nietemporalnych wartości odpowiadających im atrybutów z tabel przekształcanej bazy.

4. Uzyskane wyniki

W celu dokonania rzeczywistej oceny problemu przeanalizowano zestaw kilkunastu przykładowych zapytań, których treść zawiera tabela 4.1. Zarejestrowane czasy ich wykonania prezentuje tabela 4.2.

Uzyskane rezultaty wyraźnie wskazują, że najbardziej obciążającymi dla systemu zapytaniami są te o numerach 6 i 16. Po zmianie modelu wybranych tabel bazy zaobserwowano znaczącą poprawę czasu realizacji zapytania 16, gdy tymczasem wykonanie zapytania 6 przebiegało wolniej. Podobne różnice zauważono w pozostałych zapytaniach (p. zapytanie 7 i 4, oraz 10 i 17). Aby uzyskać pełniejszy obraz tendencji, zestawiono ze sobą razem wszystkie dane wynikowe uzyskane w poszczególnych klasach zapytań².

Mimo iż w niektórych przypadkach określenie klasy zapytania było niemożliwe z powodu zbyt ogólnikowego podziału typów zapytań temporalnych³, dało się zauważyć, że w modelu stanu czas odpowiedzi na zapytanie użytkownika uległ istotnemu skróceniu dla zapytań zakresowych, a wydłużeniu dla zapytań o historię atrybutu, natomiast w modelu zdarzeń odwrotnie.

¹ Chronony są reprezentacją podziału linii czasu rzeczywistego na niepodzielne segmenty równego rozmiaru.

² W tabeli 4.1 podano przy większości zapytań identyfikację ich klas.

³ Por. wprowadzoną przez V. Tsotrasa i C. Jensena klasyfikację zapytań do temporalnej bazy danych [8].

Tabela 4.1

Przykładowe zapytania realizowane w bazie danych SZPITAL

Nr zapytania	Treść zapytania	Typ zapytania
1.	Podaj dostawy leków, których ilość bieżąca jest większa od 0 i data ważności nie została przekroczona.	zapytanie o stan bazy
2.	Podaj dostawy leków, których ilość bieżąca jest większa od 0.	
3.	Podaj dostawy leków o danej serii, których ilość bieżąca jest większa od 0.	
4.	Podaj informacje o dostawach takich leków, których stawka VAT wynosi 7%.	zapytanie o historię atrybutu
5.	Podaj dostawy leków, których ilość bieżąca jest większa od 0 i data ważności została przekroczona w danym dniu.	
6.	Podaj dostawy leków, zalegających w magazynie dłużej niż 3 miesiące.	
7.	Podaj dostawy leków, których data ważności zostanie przekroczona w dniu 30.11.2004 r.	zapytanie o stan bazy
8.	Podaj leki, które podano w danym dniu w Zakładzie Radiobiologii.	
9.	Podaj wartość netto obrotów dla danego ośrodka powstawania kosztów w danym okresie.	zapytanie zakresowe
10.	Podaj wartość obrotów dla danego kontrahenta w danym okresie.	zapytanie zakresowe o danej wartości klucza
11.	Podaj, jakie leki zostały podane pacjentowi, o danym numerze PESEL w danym pobycie.	zapytanie o historię atrybutu
12.	Sporządź raport dzienny z dokumentów (z wybranego dnia).	
13.	Podaj, jakie były obroty dla danego towaru w wybranym okresie.	zapytanie zakresowe
14.	Jaki był stan magazynowy danego leku w danym dniu.	
15.	Podaj, jaki był stan magazynowy wszystkich leków w danym dniu.	zapytanie o stan bazy
16.	Podaj nazwiska pacjentów zażywających identyczne leki przez okres dłuższy niż 14 dni.	zapytanie zakresowe
17.	Podać nazwiska pacjentów, którzy zażywali dany lek w czasie swego pobytu w szpitalu w danym roku.	zapytanie zakresowe
18.	Jakie leki przepisano dla pacjenta o nazwisku Kowalski w 1994 r.	

Tabela 4.2

Uśrednione czasy realizacji zapytań do bazy w modelu zdarzeń i stanu

Nr zapytania	Czas wykonania zapytania [s]		Nr zapytania	Czas wykonania zapytania [s]		Nr zapytania	Czas wykonania zapytania [s]	
	model zdarzeń	model stanu		model zdarzeń	model stanu		model zdarzeń	model stanu
1.	2	18	7.	6	42	13.	5	4
2.	5	7	8.	0,656	0,422	14.	2	4
3.	2	6	9.	0,547	0,313	15.	2	17
4.	7	86	10.	8	3	16.	1560	39
5.	1	1	11.	0,344	13	17.	10	2
6.	51	64	12.	6	4	18.	8	10

Obserwowane dysproporcje wymagają ze strony administratora dodatkowej analizy nie tylko wszystkich korzyści, ale i strat, które przyniesie ewentualna zmiana charakteru rozpatrywanej tabeli. Toteż zmiana modelu tabel temporalnej bazy danych, poprzedzona rozpoznaniem charakteru danych wypełniających tabele, powinna być reakcją na niezadowolającą wydajność bazy dla najczęściej wykonywanych zapytań, a nie pochopnym działaniem administratorskim.

5. Podsumowanie

Przytoczony przykład potwierdza istnienie ścisłego związku między charakterem wczytywanych danych i sposobem ich zapisu w bazie, a przeprowadzone w ramach artykułu rozumowanie jasno precyzuje, co powinno być podstawą zmiany modeli tabel. Jedynie wnikliwa obserwacja zmian dokonywanych na danych zapamiętanych w bazie (np. za pomocą histogramów) oraz charakter tych danych (dane ciągle, dane okresowe) mogą wskazać projektantowi systemu właściwy sposób doboru modelu poszczególnych tabel. Stwierdzenie to pozostaje w pewnej sprzeczności z przyjmowanym powszechnie w literaturze poglądem o możliwości automatycznej transpozycji modeli, ale zostało poparte analizą rzeczywistych danych.

LITERATURA

1. Bair J.: It's about Time! Supporting Temporal Data in a warehouse. InfoDB, Vol. 10, No 1, 1997, <http://www.dbaint.com/InfoDBDW.html>
2. Beynon-Davies P.: Systemy baz danych. WNT, Warszawa 1998.
3. Chomicki J., Toman D.: Temporal Logic in Information Systems. Basic Research in Computer Science, 1997, <http://www.brics.dk/LS/97/1/>
4. Pissinou N., Snodgrass R.T., Elmasri R.: Towards an infrastructure for temporal databases. Report of an International ARPA/NSF Workshop, Arlington, Texas 1993.
5. Snodgrass R.T., Ahn I.: A Taxonomy of Time in Databases, ACM 0-89791-160-1/85/005/0236, 1985.
6. Steiner A.P.: A Generalization Approach to Temporal Data Models and their Implementations, Informatik Dept. Zurich 1998, <http://www.timeconsult.com>
7. Tschritzis D.C., Lochovsky F.H.: Modele danych. WNT, Warszawa 1990.
8. Tsotras V.J., Jensen C.S.: A Notation for Spatiotemporal Queries. TimeCenter Technical Report TR-10, 1997, <http://www.cs.auc.dk/TimeCenter/>
9. Ullman J.D., Widom J.: Podstawowy wykład z systemów baz danych, WNT, Warszawa 2000.
10. Werner A.: Semantyka danych temporalnych i zapytań. Konferencja SCR, Ustroń, 2002.

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

Wpłynęło do Redakcji 14 lipca 2004 r.

Abstract

Because of the fact temporal database may be modelled as event or state one, the topic of this article is the problem of models transitions in temporal databases. The article is organized as follows. The first part is connected with fundamental concepts of models (especially temporal models). This is the basic platform to examine the possibilities and reasons of making title conversion, that is discussed in point 2. The example of a database analysis, presented in the next part of the report, identifies the difficult problem of transforming one sort of table into another in case of discontinuous data occurrence. Besides, sample queries were run against different types of temporal databases and the processing time of these queries was measured. The experiments included in fourth section show, that the

query time execution decreases in some cases. It was due to the nature of modelled data. Finally, conclusions are offered in section five.

Adres

Aleksandra WERNER: Politechnika Śląska, Instytut Informatyki, ul. Akademicka 16,
44-100 Gliwice, Polska, ola@ares.iinf.polsl.gliwice.pl .

