STUDIA INFORMATICA 2017
Volume 24 Number 3 (48)

Zygmunt BOK

METODA PROJEKTOWANIA TEMATYCZNEJ HURTOWNI DANYCH ZA POMOCĄ STANDARDU SQL

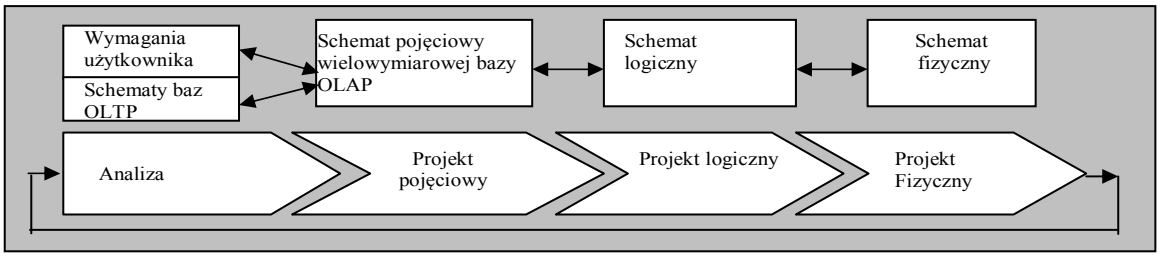
Streszczenie. W tym artykule przedstawiono metodę projektowania tematycznej hurtowni danych, bazującej na algorytmie dynamicznego projektowania hurtowni danych wykorzystującej standard SQL-99. Bazując na tej metodzie, omówiono problem dynamicznego projektowania tematycznej hurtowni danych, biorąc pod uwagę zapytania analityczne formułowane przez użytkownika końcowego. W zaproponowanym algorytmie zaimplementowanym w tej metodzie, każde nowe zapytanie analityczne analizowane jest pod kątem możliwości jego realizacji. Jeśli nie może być wykonane, wówczas poddawane jest dalszej analizie w celu wyodrębnienia ewentualnych zapytań pomocniczych lub cząstkowych (jednoprzebiegowych), tzn. takich zapytań, których wyniki są danymi wejściowymi do nowego zapytania analitycznego. Na podstawie tych wyodrębnionych zapytań pomocniczych lub cząstkowych podejmowana jest decyzja o inkrementalnym, dynamicznym rozszerzaniu schematu za pomocą zaproponowanej metody.

THE SQL-92 STANDARD UTILIZATION IN DATA WAREHOUSE DESIGN

Summary. In this article the dynamic method extending the schema data warehouse that uses the standard SQL-99 has been presented. Based on this method and dynamically extension data warehouse schema method using SQL-99 standard, a data warehouse design problem taking into account analytical queries formulated by end user has been discussed. In the proposed algorithm implemented in this method every new analytical query is analyzed at an angle of it's realizability. If it can not been executed then to isolate possible auxiliary or partially (one-route) queries, eg. such queries whose results are input data to a new analytical query to further analysis is submitted. Based on this isolated auxiliary or partially queries, a decision about incrementally and dynamically data warehouse schema extension is taking with the aid of proposed method.

1. Wprowadzenie

Najczęściej obecnie stosowane metody projektowania struktury baz danych do systemów transakcyjnych wykorzystują technikę diagramów związków encji *ER* (*Entity Relationship*) oraz technikę normalizacji. Podejście to nie zawsze jest jednak właściwe do projektowania baz danych w systemach *OLAP*, gdzie najważniejsza jest efektywność wyszukiwania i ładowania danych [1]. Na podstawie publikacji [2,3] wynika, że projektowanie hurtowni danych wymaga technik kompletnie różnych od tych, które zostały zaadoptowane z systemów transakcyjnych. Obok prac badawczych w zakresie modelowania wymiarowego (*Dimensional modelling*) i technik oraz nowych notacji konceptualnego modelowania do reprezentowania wielowymiarowych danych [4,5,6] w hurtowniach danych, nie podjęto jednak dotychczas znaczących wysiłków zmierzających do rozwinięcia spójnej metodologii [7,8,9] projektowania hurtowni danych na podstawie jednego zaakceptowanego standardu modelowania konceptualnego [10]. Na podstawie prac [11,12,13,14] wynika, że tradycyjny proces projektowania schematu hurtowni danych można przedstawić jako ciąg następujących po sobie i wzajemnie zazębiających się etapów, które pokazano na rys. 1.

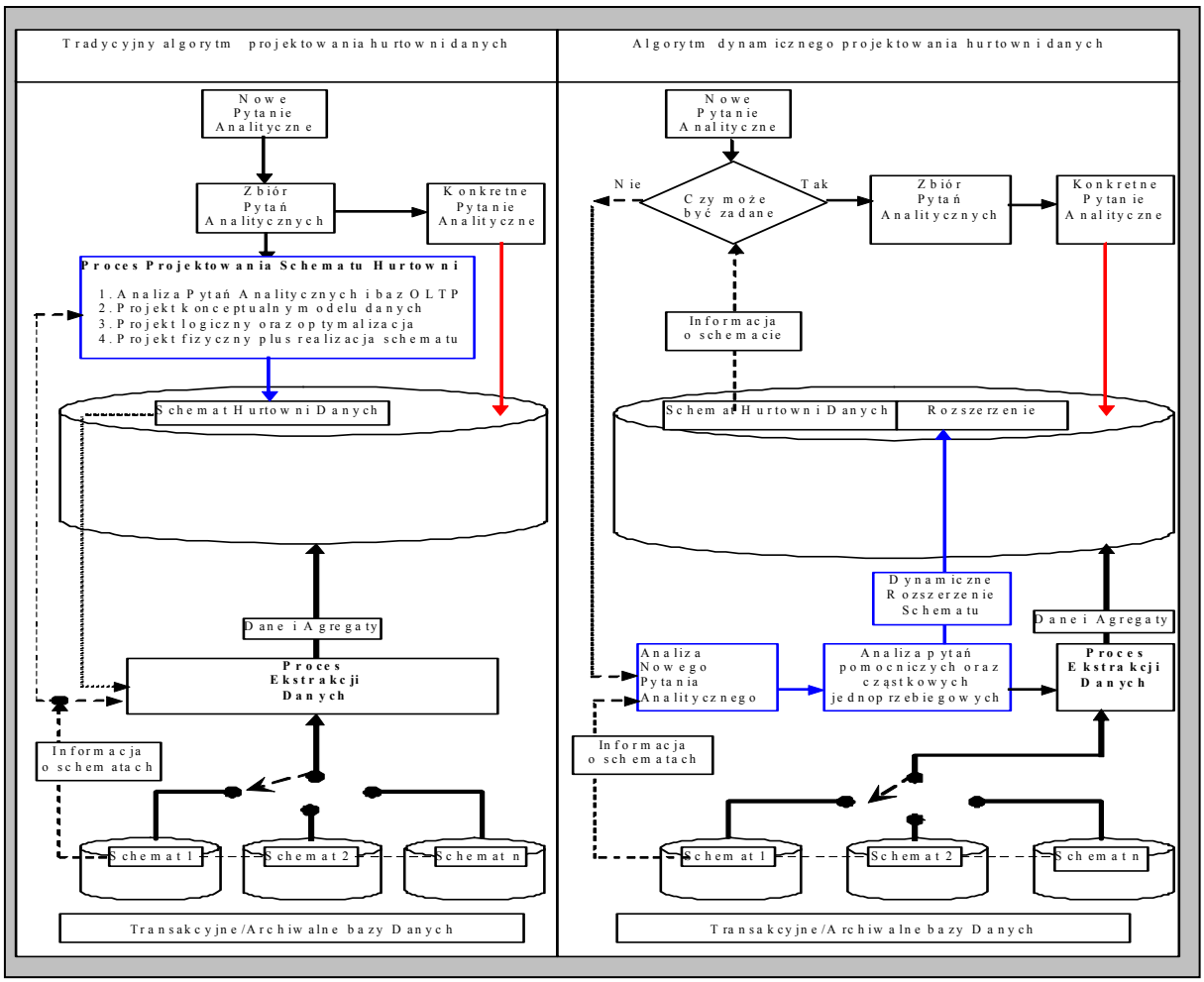


Rys. 1. Tradycyjny proces projektowania schematu hurtowni danych
Fig. 1. The traditional data warehouse schema project process

W przedstawionym zamkniętym cyklu projektowym wyróżniamy cztery zasadnicze etapy projektowania środowiska hurtowni danych, które powtarzane są cyklicznie w przypadku pojawienia się nowych wymagań użytkownika. Zmaterializowane są w postaci zbioru zapytań analitycznych. Stanowią przedmiot pierwszego etapu projektowania obejmującego równocześnie analizę schematów transakcyjnych baz *OLTP*. Na etapie projektu pojęciowego (*conceptual project*) powstaje pojęciowy schemat hurtowni danych, zrozumiały dla użytkowników końcowych. Możliwa jest również na tym etapie weryfikacja ich wymagań, identyfikacja luk oraz analiza celów biznesowych. Powstający schemat pojęciowy jest zarazem niezależny od kwestii implementacyjnych. Jego formalność i kompletność oznacza, że może być poddany jednoznacznej transformacji do schematu logicznego, który próbuje zbalansować paradygmat niezależności magazynowania (*storage-independent*) i naturalnej reprezentacji informacji w kategoriach komputerowo-zorientowanych koncepcji. W końcu następuje etap projektu fizycznego schematu hurtowni danych, operujący na poziomie szczegółów reprezentujących

informację w sprzęcie. Oprócz tradycyjnego podejścia do projektowania schematu hurtowni danych na podstawie wymagań użytkownika końcowego, istnieją również inne. Punktem wyjścia w tych podejściach projektowania schematu hurtowni danych są schematy *ER* z zastanych transakcyjnych systemów *OLTP* [15,16,17], a nie zbiór pytań analitycznych.

Budowa hurtowni danych jest zadaniem złożonym. Realizowana jest przy pomocy wielu narzędzi [18]. Projektant podczas jej budowy napotyka na szereg problemów. Jednym z nich jest istniejący w hurtowniach danych problem dotyczący magazynowania danych. Obejmuje on proces pozyskiwania, konwersji i integracji danych źródłowych, zwany również ekstrakcją danych. Wykonywany jest cyklicznie przez specjalnie opracowane programy. Ekstrakcja danych może być również ręcznie realizowana przez użytkowników systemu [19], na których spoczywa pełna odpowiedzialność za poprawność procesu magazynowania danych i co za tym idzie – za spójność magazynu danych. Mając na uwadze, w skrócie zaprezentowany, aktualny stan wiedzy odnoszący się do tradycyjnego podejścia do problemu statycznego projektowania, budowy hurtowni oraz ekstrakcji danych, dokonano krótkiej syntezy tych informacji, którą schematycznie zaprezentowano na rys. 2.



Rys. 2. Tradycyjne i alternatywne podejście do problemu projektowania hurtowni danych
Fig. 2. The traditional and alternative approach to data warehouse design problem

Na podstawie przedstawionej syntezy zaproponowano, przedstawiony na tym samym rysunku, algorytm dynamicznego projektowania i budowy hurtowni danych. W tym podejściu projektant koncentruje się na wymaganiach użytkownika końcowego, wyrażonych w postaci zapytań analitycznych pojawiających się *ad-hoc*. Jest to podejście alternatywne w stosunku do tradycyjnego podejścia projektowania hurtowni danych, a w szczególności do obowiązującej w nim naczelnej zasady, która zakłada, że schemat hurtowni powinien bezpośrednio wynikać z wcześniej określonego zbioru zapytań analitycznych [20]. Zbiór ten powinien obejmować wszystkie spodziewane typy zapytań analitycznych, które mogą być zadane przez użytkownika. Jest on niezbędny do zaprojektowania 'właściwie' określonego schematu. Pod pojęciem 'właściwie' określony schemat hurtowni danych rozumie się taki jej schemat, który umożliwia realizację większości zapytań analitycznych użytkownika. Z drugiej strony, wysoka nieprzewidywalność i zmienność w czasie wymagań analitycznych użytkownika końcowego skutkuje niestabilnymi podstawami projektowymi. Stanowi to poważny problem i jedną z głównych wad podejścia tradycyjnego. Problem ten jest obecny w najnowszej literaturze dotyczącej integracji różnych, autonomicznych i heterogenicznych zewnętrznych źródeł danych *EDS* (*External Data Sources*), jak choćby podejście typu *data-warehousing* do dynamicznego projektowania hurtowni danych zaproponowane w pracach [21,22,23,24,25,26], czy też podejście typu *query-driven* opisane w pracach [27,28,29,30]. Podejście typu *data-warehousing*, które stało się popularną obecnie technologią, bazuje na centralnym repozytorium danych. Po ekstrakcji danych z EDS'ów, ładowane są one do centralnego repozytorium zwanego hurtownią danych. W podejściu typu *query-driven*, *EDS*'y integrowane są tylko na poziomie logicznym, poprzez łączenie (*merging*) wszystkich lokalnych schematów w jeden schemat globalny (bez integracji zawartości *EDS*'ów). Zapytania użytkownika wyrażone w języku SQL kierowane do schematu globalnego przekształcane są następnie na jedno lub wiele zapytań kierowanych do lokalnych EDS'ów. Uzyskane odpowiedzi łączone są w odpowiedź finalną zwracaną do użytkownika.

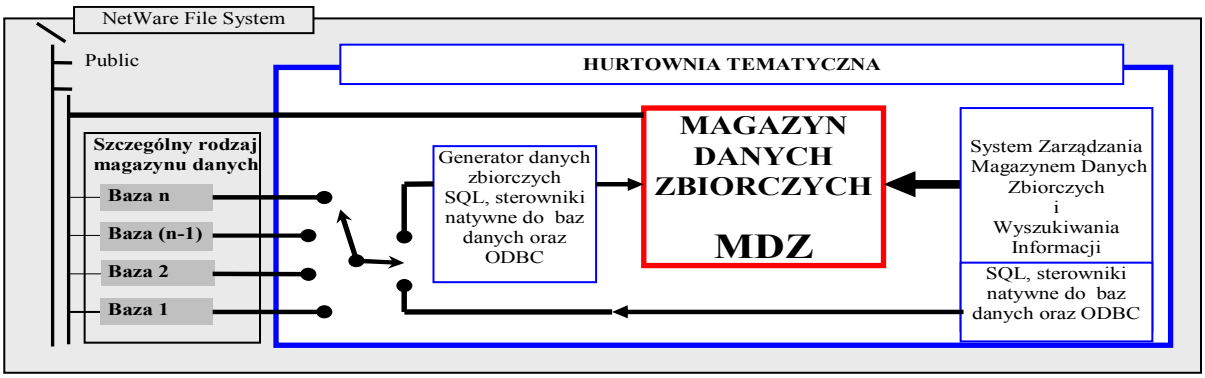
Podczas projektowania hurtowni danych, niezależnie od przyjętego pojęciowego modelu danych, problemem zasadniczym przy definiowaniu i rozwoju schematu pojęciowego wielowymiarowej bazy OLAP jest osiągnięcie pewnych celów. Pierwszym celem w wielowymiarowym modelowania danych jest stworzenie takiej struktury wielowymiarowej bazy danych OLAP, która jest łatwa do zrozumienia i wykorzystania przez użytkownika końcowego, kierującego do niej zapytania analityczne. Drugim celem jest maksymalizacja efektywności wykonania tych zapytań. Cele te osiąga się przez minimalizację liczby tablic i łączących je relacji, przez co redukuje się złożoność wielowymiarowej bazy danych oraz minimalizuje się liczbę złączeń wymaganych do realizacji zapytania analitycznego. W celu realizacji tak określonych celów podczas projektowania wielowymiarowej bazy danych, najlepszym rozwiąza-

niem jest zdefiniowanie jej schematu w postaci gwiazdy [4,20]. Powszechnie akceptowane schematy tego typu zawierają zależności pomiędzy danymi typu *wiele-do-jednego*. Często się zdarza, że mogą zawierać również zależności typu *wiele-do-wielu*. Objawia się to tym, że pomiędzy encją faktów i pewną encją wymiarów istnieją encje typu *wiele-do-wielu*. Istnienie tego typu encji generuje kilka trudnych problemów. Wśród nich wyróżnić można utratę prostoty struktury typu gwiazda, wzrost stopnia złożoności formułowanych pytań analitycznych, spadek efektywności wykonywanych pytań spowodowanych wprowadzeniem większej ilości złączeń. Jednym z rozwiązań tych problemów może być wprowadzenie do schematu hurtowni danych [31] encji łączącej. Jest ona podobna do encji pośredniczącej [32], która powiązana jest, za pomocą encji typu *wiele-do-jednego*, ze znormalizowanymi encjami, początkowo zawierającymi relacje typu *wiele-do-wielu*.

Zaproponowane w tym artykule podejście typu *query-driven*, w postaci algorytmu do dynamicznego projektowania i budowy hurtowni danych, może okazać się uzasadnione również w sytuacji, w której wiedza na temat zbioru zapytań analitycznych na etapie projektowania jest ograniczona oraz jeśli nie wiadomo, kiedy pojawią się nowe zapytania analityczne.

Celem niniejszego artykułu było wykorzystanie tego podejścia w zaproponowanej metodzie do zbudowania tematycznej wielowymiarowej hurtowni danych, na osnowie pewnego przemysłowego systemu informacyjnego dotyczącego sprzedaży wyrobów gotowych. Z tego systemu istniała potrzeba uzyskania informacji zbiorczych zawartych w zbiorze relacyjnych archiwalnych bazach danych, zeskładowanych w oddzielnych folderach systemu plików pewnego serwera sieciowego, dotyczących poprzednich zamkniętych okresów obliczeniowych. Wspomniany zbiór archiwalnych relacyjnych baz danych $\boldsymbol{r} = \{r_{ij}\}$ o schematach $\boldsymbol{R}=\{R_{ij}\}$ potraktowano jako podstawowe repozytorium informacji lub inaczej jako pewny szczególny rodzaj magazynu danych. W tym zbiorze wskaźnikiem $i\in\{1,...,n\}$ oznaczano poszczególne archiwalne bazy danych, natomiast wskaźnikiem $j\in\{1,...,m\}$ kolejne jej relacje. Schemat tego szczególnego magazynu danych definiowany jest jako suma atrybutów schematu abstrakcyjnej relacji uniwersalnej $\boldsymbol{u}$ utworzonej nad zbiorem wszystkich atrybutów ze schematów $\boldsymbol{R}$ [33,34,35,36]. Ponieważ w tym magazynie dla każdego ustalonego *j* zachodzi $R_{1j}=R_{2j}=...=R_{nj}$, zatem jego schemat definiowany jest jako suma atrybutów $\boldsymbol{R}= \{R_1 \cup R_2 \cup \ldots \cup R_j \cup \ldots \cup R_m\}$. Innymi słowy, odpowiednie relacje w poszczególnych bazach składowych opisywanego zbioru baz danych mają taki sam schemat. Z tak pojmowanego szczególnego rodzaju magazynu danych istnieje potrzeba uzyskania informacji zbiorczych według różnych wymiarów, w tym wymiaru czasu, poprzez realizację pojawiających się *ad hoc* zapytań analitycznych dotyczących jednej lub wielu, w zależności od zakresu pytania, oddzielnych baz danych i uzyskania jednoznacznej odpowiedzi. Z drugiej strony, tak określony szczególny rodzaj magazynu danych jest w pewnym sensie podobny do wydzielonych systemów baz danych wspomagają-

cych przetwarzanie analityczne. Architektura takich systemów zakłada bowiem pełną izolację przetwarzania operacyjnego i analitycznego. Informacje powstające w operacyjnych bazach danych tych systemów są replikowane i fizycznie składowane w pewnym magazynie danych do późniejszego przetwarzania analitycznego. Ponieważ w opisywanym przypadku istnieje pełna izolacja pomiędzy bazami archiwalnymi i operacyjnymi, jak również to, że nie ma potrzeby replikowania danych archiwalnych do osobnego magazynu danych, zatem w tym właśnie sensie szczególny rodzaj magazynu danych podobny jest do wydzielonych systemów baz danych wspomagających przetwarzanie analityczne. Wykorzystując to podobieństwo, postanowiono zastosować zaproponowane podejście do zaprojektowania i zbudowania tematycznej wielowymiarowej hurtowni danych, której architekturę przedstawiono na rys. 3.



Rys. 3. Przykład architektury prostej hurtowni tematycznej
Fig. 3. The simple data marts architecture example

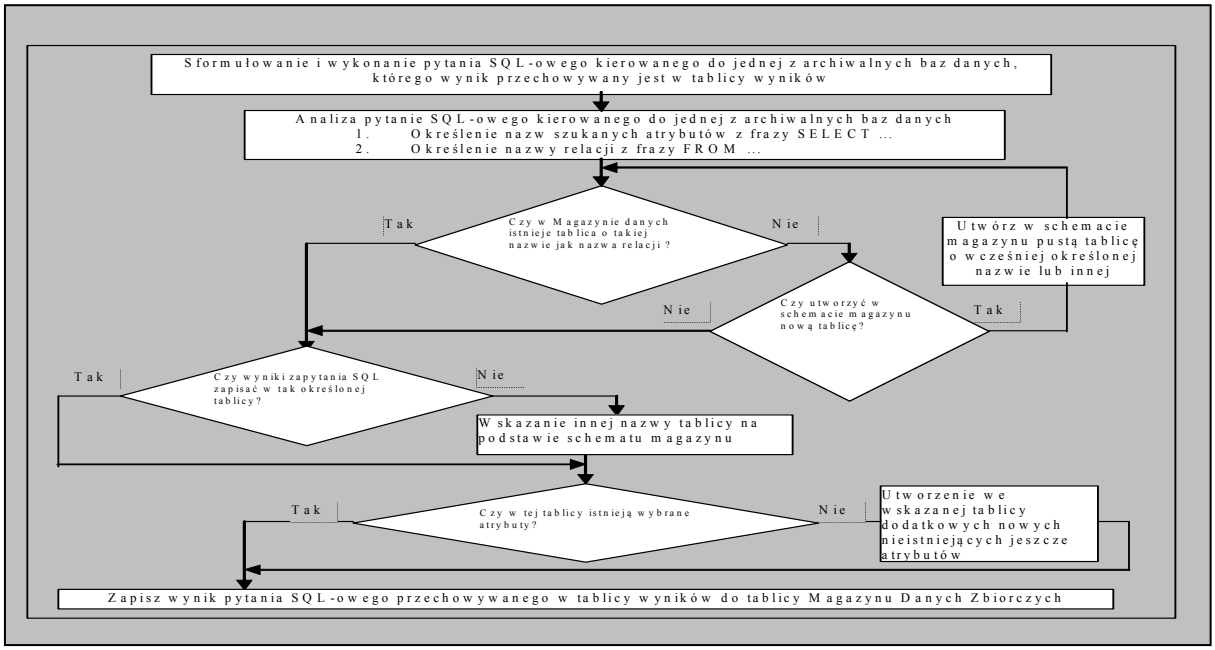
W związku z tym, że szczególny rodzaj magazynu nie zawiera informacji zbiorczych zagregowanych na różnych poziomach niezbędnych do analitycznego przetwarzania, konieczne stało się zaprojektowanie Magazynu Danych Zbiorczych (MDZ), którego celem będzie przechowywanie informacji zbiorczych pochodzących ze szczególnego rodzaju magazynu danych. Jego schemat powinien być tak określony, aby umożliwiał realizację większości potencjalnych zapytań analitycznych. Niestety, o pytaniach tych wiadomo tylko tyle, że powinny rozszerzać zbiór standardowych zestawień, predefiniowanych w aplikacji obsługującej bazy archiwalne. Horyzont czasowy tych zestawień sięga jednego roku, tj. dzień, tydzień, miesiąc, kwartał, rok. Innymi słowy, otrzymywane odpowiedzi na zapytania analityczne, kierowane do MDZ, w wymiarze czasu powinny obejmować zagregowane dane dotyczące kilku lat. Kierowano je do MDZ za pomocą przykładowego systemu zarządzania tym magazynem, który zrealizowano wykorzystując pakiet *SQLWindows Team Developer*, stworzony przez amerykańską firmę komputerową *CENTURA* (dawniej *GUPTA*). Pakiet ten jest w pełni obiektowym narzędziem (4GL), opartym na wstępnie zdefiniowanych klasach obiektowych z wszystkimi korzyściami wynikającymi z programowania obiektowego. Dzięki sterownikom natywnym, pakiet ten umożliwia dostęp do baz danych typu *SQLBase*, *ODBC*, *Informix*, *Ingress*, *Sybas oraz Oracle*, w której definiowano schemat MDZ i gromadzono niezbędne dane.

2. Opis metody bazującej na algorytmie dynamicznego projektowania hurtowni danych

Jak widać na zaprezentowanym już rys. 2, kluczowym elementem w omawianej metodzie projektowania hurtowni danych jest mechanizm dynamicznego rozszerzania jej schematu [37]. Mechanizm ten wykorzystuje wymieniany przez *ANSI/ISO* standard języka programowania *SQL* dla relacyjnych baz, który posiada tę własność, że jednoczy język zapytań *SQL* z językiem zarządzania danymi (*DML*) i schematem bazy danych (*DDL*) i tym samym ma wpływ na fizyczny projekt hurtowni danych. W chwili obecnej większość popularnych obecnie Relacyjnych Systemach Zarządzania Bazą Danych (RSZBD) implementuje język *SQL* na wyjściowym poziomie zgodności ze standardem *SQL-99*. Postanowiono zatem - mimo pewnych ograniczeń - wykorzystać ww. własność do realizacji koncepcji dynamicznego rozszerzania schematu MDZ. Do wspomnianych ograniczeń zaliczyć można między innymi ograniczenia i wady języka *SQL*, w szczególności zaś jego ograniczenia implementacyjne w pakiecie *SQLWindows Team Developer*. Pomimo tych ograniczeń, możliwe było za pomocą tego mechanizmu pobranie z szczególnego magazynu niezbędnych danych i załadowanie ich do MDZ. Podczas pobierania danych poddawano wstępnej filtracji i agregowaniu. Następnie, w trakcie procesu zapisywania pobranych danych do przykładowego MDZ, dynamicznie rozszerzano jego schemat. Podczas dynamicznego rozszerzania tego schematu korzystano ze składni zapytań *SQL*-owych zadawanych do szczególnego magazynu danych, jak również z informacji o strukturze plików składowych tych baz. Tak otrzymane dane, będące wynikiem zadanych zapytań *SQL*-owych, poddawano konwersji niektórych typów danych ze schematu tego magazynu do postaci akceptowanej przez schemat MDZ. Następnie, korzystając z poleceń *SQL*, wstępnie zagregowane i przetworzone dane ładowano do tego magazynu. W zaproponowanej metodzie wyeliminowano jedną z jej najistotniejszych wad. Polegała ona na tym, że wygenerowana postać schematu MDZ była całkowicie zależna od szczególnego magazynu danych. Obecnie, dzięki nowej implementacji mechanizmu dynamicznego jego rozszerzania działającego według zmienionego algorytmu, przedstawionego na rys. 4, możliwe jest utworzenie poprawnej struktury MDZ typu gwiazdy lub płatka śniegu w każdym przypadku.

Pomimo wyeliminowania ww. wady, zaproponowany mechanizm obarczony jest jeszcze innymi wadami. Między innymi wygenerowany schemat MDZ pozbawiony jest tego, co człowiek wnosi do procesu projektowania struktury hurtowni, czyli optymalizacji struktury danych. Przedstawiony mechanizm obejmuje tylko przypadki jednoprzebiegowe, tj. gdy zapytanie SQL-owe kierowane do szczególnego magazynu danych generuje oczekiwane wyniki (dane zagregowane); nie zaimplementowano mechanizmu umożliwiającego pozyskanie wyników oczekiwanych po kilku przebiegach. Dzięki jednak swej prostocie, w przypadku two-

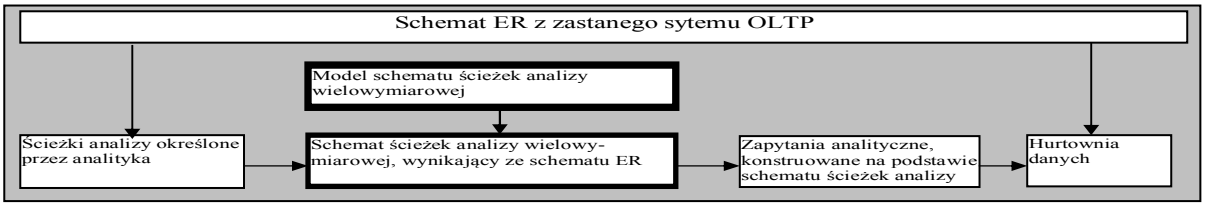
rzenia małych hurtowni tematycznych na podstawie istniejących zastanych przemysłowych systemów informacyjnych, zaproponowany mechanizm dynamicznego rozszerzania schematu hurtowni może okazać się przydatny.



Rys. 4. Algorytm mechanizmu dynamicznego rozszerzania schematu magazynu danych
Fig. 4. The dynamic algorithm schema data warehouse extension mechanism

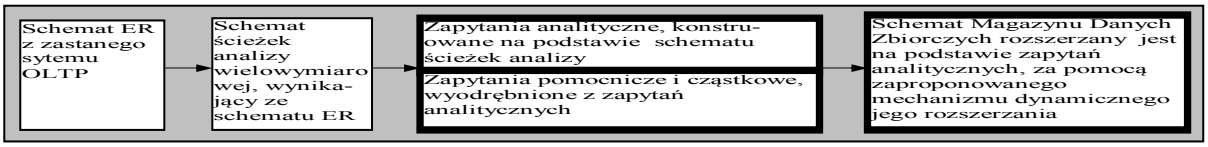
W zaproponowanej metodzie oparto się również na modelu *Multidimensional Aggregation Cube (MAC)* [38] i stowarzyszonych z nim pojęć. Jest to praca, która obok prac [39,40,41], dedykowana jest problemowi modelowania danych, używanych w analizie wielowymiarowej na poziomie pojęciowym, z perspektywy użytkownika końcowego. Praca ta dotyczy projektów pojęciowych, które pod uwagę biorą jego wymagania jako punkt początkowy. Model *MAC* wykorzystano w celu dokonania analizy wpływu zapytań analitycznych na postać dynamicznie rozszerzanego schematu MDZ. W szczególności, wykorzystano wprowadzoną w tym modelu koncepcję ścieżek analizy. Bazując na tej koncepcji zdefiniowano formalny model schematu ścieżek analizy wielowymiarowej. Model ten wykorzystano w projektowaniu wielowymiarowej hurtowni danych [42], do określenia konkretnego schematu ścieżek analizy wielowymiarowej, wynikającego z zastanego szczególnego magazynu danych. Schemat ten stanowił podstawę konstrukcji właściwych, z punktu widzenia tego magazynu, zapytań analitycznych. Formułowano je w oparciu o kombinacje różnych ścieżek analizy określonych w schemacie ścieżek analizy wielowymiarowej, który pokazano na rys. 5, zapewniając tym samym potencjalną możliwość realizacji takich zapytań. Zaproponowane podejście zapewnia ewolucję schematu [40] hurtowni danych w sytuacji, gdy pojawiają się nowe zapytania analityczne. Zapewnia ono również wyeliminowanie wstępnie zdefiniowanego na podstawie wymagań użytkownika zbioru zapytań analitycznych. W tym podejściu, każde nowo pojawiające

się pytanie analityczne poddawane jest analizie pod kątem możliwości jego zrealizowania. Jeśli nie może być zrealizowane, wówczas poddawane jest dalszej analizie w celu wyodrębnienia ewentualnych zapytań pomocniczych lub cząstkowych (jednoprzebiegowych), tzn. takich, których wyniki są danymi wejściowymi do nowego zapytania analitycznego. W trakcie realizacji zapytań cząstkowych podejmowano decyzję o inkrementalnym, dynamicznym rozszerzaniu schematu MDZ za pomocą zaproponowanego mechanizmu.



Rys 5. Konstrukcja zapytań analitycznych na podstawie schematu ścieżek analizy
Fig 5. The analytical queries construction based on analytical paths schema

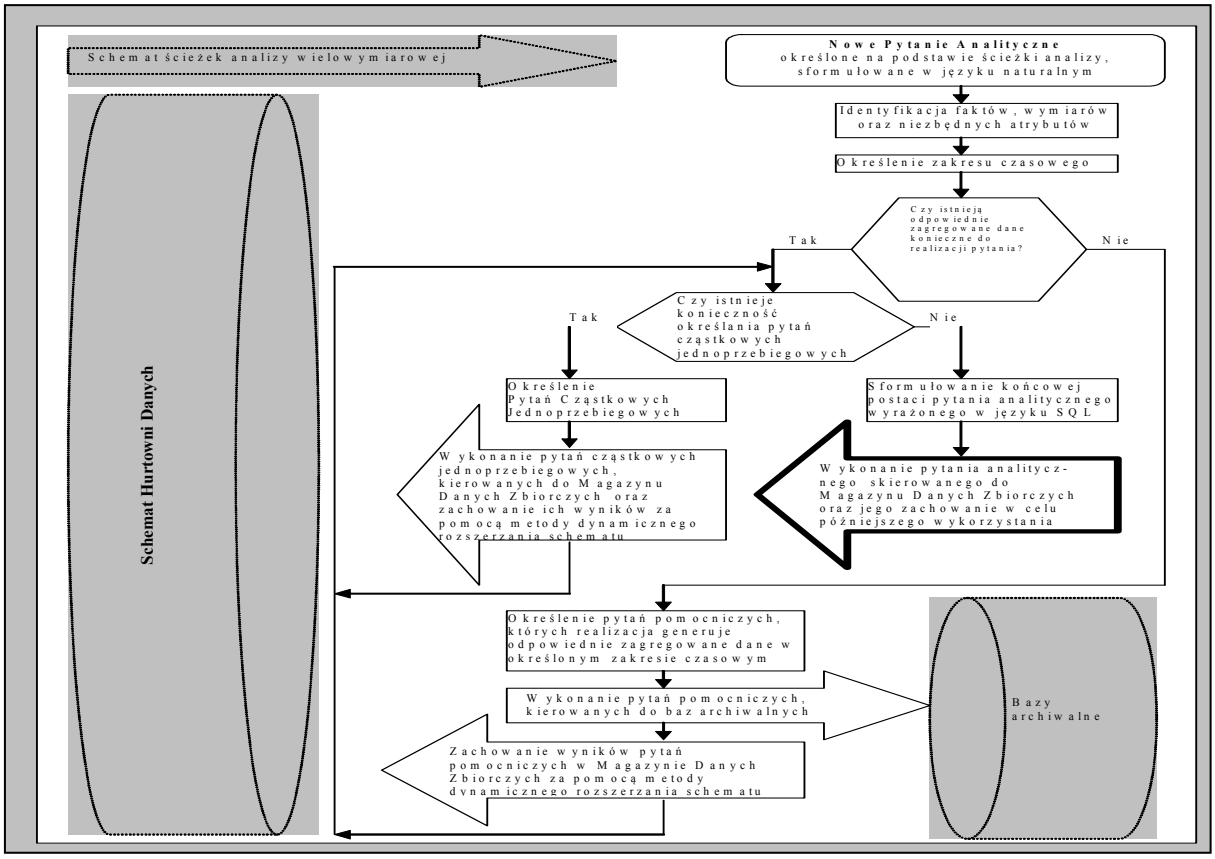
W wyniku zastosowania zaproponowanego podejścia, proces projektowania MDZ można schematycznie przedstawić w postaci ciągu zdarzeń, co uwidoczniono na rys. 6.



Rys 6. Koncepcja projektowania hurtowni na podstawie zapytań analitycznych typu *ad-hoc*
Fig 6. The concept of the data warehouse project based on *ad-hoc* type analytical queries

Pierwszym krokiem podczas analizy nowego zapytania analitycznego za pomocą scenariusza działań pokazanych na rys. 7 jest określenie, które z poszukiwanych informacji odnoszą się do danych ilościowych (faktów), a które do danych kwalifikujących (wymiarów). Celem drugiego kroku jest określenie zakresu czasowego tego zapytania. W trzecim kroku określa się, czy w MDZ istnieją odpowiednie zagregowane dane, konieczne do zrealizowania nowego zapytania analitycznego, czyli czy istnieje konieczność formułowania dodatkowych zapytań cząstkowych generujących odpowiednio zagregowane dane. W zaprezentowanym podejściu, przedstawiony scenariusz analizy zapytań analitycznych prezentuje ogólny sposób postępowania z każdym kolejnym nowym zapytaniem analitycznym wyrażonym w języku naturalnym. Nie zaimplementowano na jego podstawie żadnych mechanizmów do automatycznego podejmowania decyzji o możliwości (lub nie) zadania zapytania analitycznego do MDZ. Nie zaimplementowano również żadnych mechanizmów automatycznie generujących pytania cząstkowe lub pomocnicze. W przyjętym rozwiązaniu powyższe pytania konstruowano *ad-hoc* na podstawie logicznej analizy zapytania analitycznego, schematu ścieżek analizy wielowymiarowej oraz znajomości schematu *ER* szczególnego rodzaju magazynu danych. Ponadto przyjęto, że początkowy schemat MDZ określono za pomocą metody opisanej w publikacji

[**20**]. Metoda ta wykorzystuje tradycyjny model *ER* do projektowania hurtowni danych na podstawie modeli danych z przemysłowych systemów informacyjnych.



Rys. 7. Scenariusz analizy zapytań analitycznych typu *ad-hoc*
Fig. 7. The *ad-hoc* type analytical queries analyse scenario

Cechą charakterystyczną tej metody jest zastosowanie denormalizacji do wstępnie pogrupowanych encji modelu *ER* z zastanego systemu informacyjnego, według przyjętych przez autorów trzech kategorii klasyfikujących, tj. kategorii encji transakcyjnych, klasyfikacyjnych oraz komponentowych. Poprzez zastosowanie operatora agregacji w stosunku do encji transakcyjnych możliwe jest utworzenie nowych encji zawierających zagregowane dane. W przypadku projektowania schematu hurtowni danych typu gwiazda, encja faktów formowana jest na podstawie encji transakcyjnych, natomiast encje określające wymiary tworzone są dla każdej encji komponentowej poprzez denormalizację hierarchicznie powiązanych encji klasyfikujących. Biorąc powyższe pod uwagę, w celu wykazania przydatności zaproponowanej koncepcji projektowania MDZ na podstawie zapytań analitycznych formułowanych na bazie schematu ścieżek analizy wielowymiarowej oraz schematu *ER* zastanych archiwalnych systemów *OLTP*, analizie poddano wpływ niektórych przykładowych zapytań analitycznych na początkową postać schematu magazynu. Zapytania te należały do jednej z trzech kategorii zapytań. Kategoria I obejmuje zapytania z grupy zapytań skutkujących zwiększeniem liczby wymia-

rów magazynu. Kategoria II obejmuje zapytania z grupy zapytań skutkujących zwiększeniem liczbę ścieżek analizy w ramach danego wymiaru. W końcu, kategoria III obejmuje zapytania z grupy zapytań skutkujących zwiększeniem liczby poziomów w ramach danego wymiaru.

2.1. Model schematu ścieżek analizy wielowymiarowej

Jak wspomniano wcześniej bazując na, wprowadzonej w modelu *MAC*, koncepcji ścieżek analizy, zaproponowano formalny model schematu ścieżek analizy wielowymiarowej. Model ten wykorzystano do określenia konkretnego schematu ścieżek analizy wielowymiarowej, stanowiący podstawę konstrukcji właściwych zapytań analitycznych. Dla jego formalnego zdefiniowania oparto się na, cytując za [8,43,44,45,46], poniższych definicjach.

Definicja 1. Grafem G =(V,E) nazywamy zbiór węzłów V={v_1, v_2, ...} oraz zbioru krawędzi E={e_1, e_2, ...}. Krawędź e_k utożsamia się z nieuporządkowaną parą węzłów (v_i, v_j). Węzły v_i, v_j związane z krawędzią e_k nazywa się węzłami końcowymi krawędzi e_k.

Definicja 2 Krawędzią grafu G=(V,E) nazywamy dowolną nieuporządkowaną parę {v_i, v_j} taką, że ($\langle v_i, v_j \rangle \in E) \vee (\langle v_j, v_i \rangle \in E$).

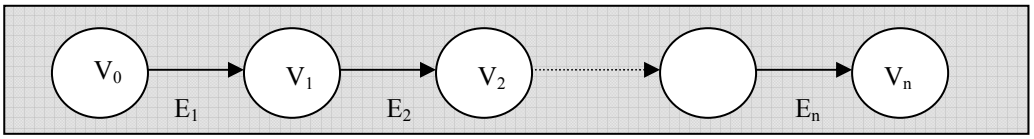
Definicja 3. Ukierunkowanym (zorientowanym) grafem nazywamy taki graf G = (V,E), w którym E jest zbiorem takich uporządkowanych par (v_i, v_j), dla których krawędź łącząca dwa węzły v_i, v_j posiada określony kierunek.

Definicja 4. Ścieżką w ukierunkowanym grafie G=(V, E) nazywamy ciąg krawędzi (v_1,v_2), (v_2,v_3),..., (v_{n-1},v_n).

Definicja 5. Ścieżką acykliczną nazywamy taką ścieżkę, którą można przemierzyć (pokonać) tylko w jeden sposób.

Definicja 6. Ukierunkowanym acyklicznym grafem nazywamy ukierunkowany graf, w którym nie ma ścieżek rozpoczynających i kończących się w tym samym węźle.

Definicja 7. Niech **g** = (V, E) będzie ukierunkowanym acyklicznym grafem, gdzie V jest zbiorem węzłów, natomiast E jest zbiorem krawędzi. Mówimy, że **g,** co pokazano na rys. 8, jest quasi-drzewem z korzeniem w $V_0 \in V$, jeśli każdy wierzchołek $V_j \in V$ może być osiągnięty z v_0 za pomocą przynajmniej jednej ukierunkowanej ścieżki.



Rys. 8. Przykład ukierunkowanego acyklicznego grafu zakorzenionego w V_0
Fig. 8. The example of the directed acyclic graph rooted in V_0

Oznaczmy przez $s_{ij} \subseteq$ **g** ukierunkowaną ścieżkę rozpoczynającą się w V_i i kończącą się w V_j. Oznaczmy dalej przez sub(V_i) $\subset$ **g** quasi drzewo zakorzenione w węźle $V_i \neq V_0$ [**8**].

Definicja 8. Schemat ścieżek analizy wielowymiarowej $\mathbf{S}_{śaw}$=(M, W, P, S) stanowi grupa powiązanych danych, gdzie:

M – jest zbiorem miar. Każda miara $M_n \in M = \{M_1 \cup M_2 \cup \ldots \cup M_m\}$ definiowana jest przez wyrażenia numeryczne pochodzące z systemów informacyjnych,

W – jest zbiorem wymiarów w analizie wielowymiarowej, tj. $W = \{W_1 \cup W_2 \cup \ldots \cup W_w\}$,

P – jest zbiorem wszystkich poziomów analizy, tj. $P = \{P_1 \cup P_2 \cup \ldots \cup P_w\}$,
gdzie $P_i = \{P_{i1} \cup P_{i2} \cup \ldots \cup P_{ij} \cup \ldots \cup P_{ik}\}$ jest zbiorem wszystkich poziomów analizy w ścieżkach analizy pewnego wymiaru $W_i \in W$. W tym zbiorze $P_{ij}=\{p_{ij1} \cup p_{ij2} \cup \ldots \cup p_{ijr}\}$ jest zbiorem poziomów w ścieżkach analizy dla j-tej ścieżki analizy, natomiast k- oznacza ilość ścieżek analizy, r – oznacza ilość poziomów analizy dla j-tej ścieżki analizy,

S – jest zbiorem wszystkich uporządkowanych podzbiorów, każdy składający się ze zbioru uporządkowanych par, tj. $S=\{S_1 \cup S_2 \cup \ldots \cup S_w\}$
gdzie $S_i=\{S_{i1} \cup S_{i2} \cup \ldots \cup S_{ij} \cup \ldots \cup S_{ik}\}$. W tym zbiorze $S_{ij}=\{(p_{ijx}, p_{ijy})_1 \cup (p_{ijx}, p_{ijy})_2 \cup (p_{ijx}, p_{ijy})_u\}$ jest zbiorem uporządkowanych par, w którym i – oznacza numer wymiaru, j - oznacza numer ścieżki analizy, u – oznacza takie uporządkowane pary w których $x<y$ oraz $x,y\in\langle 1, r\rangle$, natomiast r – oznacza ilość poziomów analizy. Uporządkowane pary, określające ukierunkowane ścieżki analizy $s_{ij,xy}=(p_{ijx}, p_{ijy})$, modelują relacje typu *wiele-do-jednego*. Za pomocą ukierunkowanych ścieżek analizy, poziom analizy p_{ijy} może być osiągnięty, wychodząc od poziomu analizy p_{ijx},
gdzie: $p_{ijy} \in \{p_0\} \cup P_{ij}=\{p_0\}+\{p_{ij1}\}+\{p_{ij2}\}+...+\{p_{ijy}\}$
$p_{ijx} \in P_{ij}=\{p_{ij1}\}+\{p_{ij2}\}+...+\{p_{ijx}\}$.

Jeśli zatem dla dowolnego wymiaru $W_i \in W$, każdy zbiór $S_{ij} \in S$ jest takim zbiorem, że graf **g**(V, E), gdzie: $V = \{p_0\} \cup P_{ij}$
$E = S_{ij}$ (j - oznacza numer ścieżki analizy w ramach i-tego wymiaru)
jest takim ukierunkowanym, acyklicznym grafem zakorzenionym w $p_0 \in P$, że każdy poziom analizy $p_{ijx} \in P_{ij}$ znajdujący się na j-tej ścieżce i-tego wymiaru może być osiągnięty wychodząc z poziomu analizy p_0 za pomocą przynajmniej jednej ukierunkowanej ścieżki, wówczas grupa powiązanych danych $\mathbf{S}_{śaw}$=(M,W,P,S) stanowi schemat ścieżek analizy wielowymiarowej.

2.1.1. Ścieżki skrośne w schemacie ścieżek analizy wielowymiarowej

Wykorzystując, wprowadzone definicją 7, pojęcie quasi-drzewa zakorzenionego w węźle $V_i \neq V_0$ oznaczonego symbolem sub(V_i) oraz formalnie zdefiniowanego pojęcia schematu ścieżek analizy wielowymiarowej, do dalszych rozważań wprowadza się pojęcie ścieżki analizy skrośnej, definiowanej na podstawie schematu ścieżek analizy wielowymiarowej.

Definicja 9. Dla danego schematu ścieżek analizy wielowymiarowej $\mathbf{S}_{śaw} = (M,W,P,S)$, prostą skrośną ścieżką analizy nazywa się taką ukierunkowaną ścieżkę analizy, która stanowi sumę dwóch ukierunkowanych ścieżek analizy spełniających następujące warunki:

1º pierwsza z ukierunkowanych ścieżek $s_{ij,\ vx}=(p_{ijv}, p_{ijx})$ pewnego wymiaru $W_i \in W$ zakorzeniona jest w $p_0 \in P_{ij}$, gdzie i – oznacza numer wymiaru, j - oznacza numer ścieżki analizy, natomiast v, x, y – oznaczają numery poziomów analizy,

2º druga z ukierunkowanych ścieżek $s_{ij,\ yz}=(p_{ijy}, p_{ijz})$ z tego samego wymiaru jest quasi-drzewem sub(P_{ij}) zakorzenionym w węźle $P_{ij} \neq p_0$,.

3º ścieżka $s_{ij,\ vz}=(p_{ijv}, p_{ijx}) \cup (p_{ijy}, p_{ijz})$ jest quasi-drzewem z korzeniem w węźle $p_0 \in P$.

Definicja 10. Dla danego schematu ścieżek analizy wielowymiarowej $\mathbf{S}_{śaw} = (M,W,P,S)$, złożoną skrośną ścieżką analizy nazywa się taką ukierunkowaną ścieżkę analizy, która stanowi sumę dwóch lub więcej ukierunkowanych ścieżek analizy spełniających następujące warunki:

1º pierwsza z ukierunkowanych ścieżek $s_{ij,\ vx} = (p_{ijv}, p_{ijx})$ pewnego wymiaru zakorzeniona jest w $p_0 \in P_{ij}$, gdzie i – oznacza numer wymiaru, j - oznacza numer ścieżki analizy, natomiast v, x, y – oznaczają numery poziomów analizy,

2º druga lub dalsze z ukierunkowanych ścieżek $s_{mn,\ yz} = (p_{mny}, p_{mnz})$ z innych wymiarów są quasi-drzewemi sub(P_{mn}) zakorzenione są w węzłach $P_{mn} \neq p_0$,

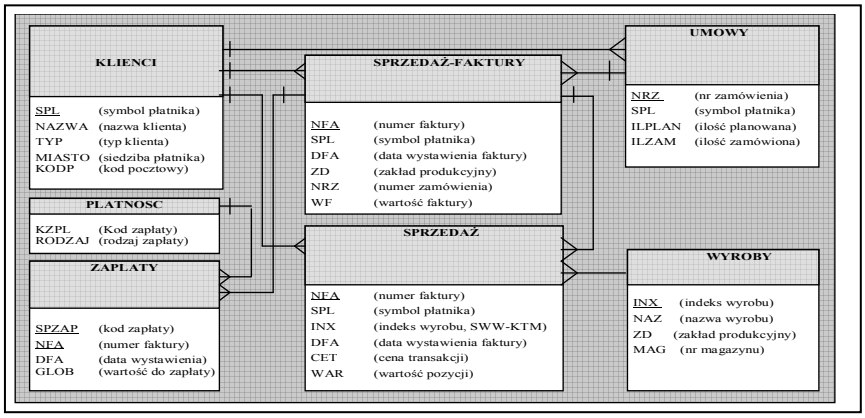
3º ścieżka $s_{cross}=(p_{ijv},p_{ijx}) \cup \ldots \cup (p_{mny},p_{mnz})$, jest quasi-drzewem z korzeniem w węźle $p_0 \in P$.

3. Realizacja dynamicznie rozszerzanego schematu Magazynu Danych Zbiorczych na podstawie zapytań analitycznych

Bazując na przedstawionej koncepcji podejścia do problemu projektowania hurtowni danych na podstawie analizy zapytań analitycznych, dokonywanej za pomocą wcześniej opisanego algorytmu przedstawionego już na rys. 4, w dalszej części pracy przedstawiono wybrane zapytania analityczne, ich analizę i ich wpływ na postać dynamicznie rozszerzanego początkowego schematu MDZ w przykładowej tematycznej hurtowni danych. Pytania te konstruowano na podstawie schematu ścieżek analizy wielowymiarowej, określonego na podstawie modelu danych *ER* ze wspomnianego we wprowadzeniu szczególnego magazynu danych.

3.1. Schemat ER szczególnego magazynu danych

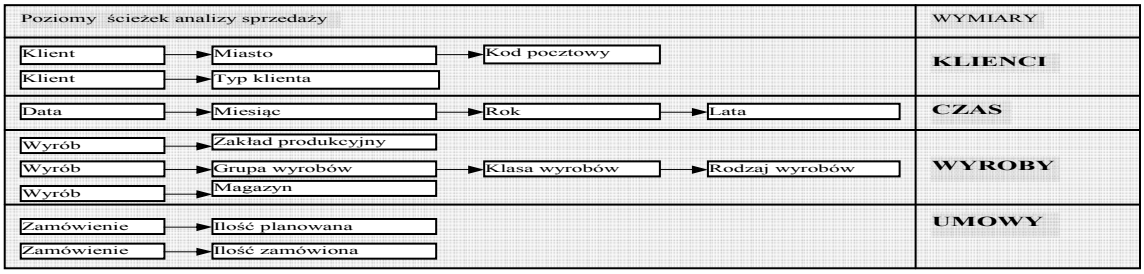
W dalszej części pracy przyjęto, że schemat *ER* łączący fragmenty niektórych encji szczególnego magazynu danych ma postać przedstawioną na rys. 9. Jest on nieodzownym elementem, umożliwiającym zaprojektowanie MDZ według zaproponowanego podejścia i stanowi podstawę do określenia schematu ścieżek wielowymiarowej analizy sprzedaży.



Rys. 9. Fragmenty encji ze szczególnego magazynu danych
Fig. 9. The entity fragments from special data waerhouse

3.2. Ścieżki wielowymiarowej analizy sprzedaży

Wykorzystując schemat *ER* szczególnego magazynu danych, przyjęto, że wybrane zapytania analityczne dotyczyły sprzedaży wyrobów gotowych względem hierarchii różnych wymiarów, tj. ścieżek analizy sprzedaży. Przy ich określaniu skorzystano z koncepcji ścieżek analizy ze wspomnianego już wcześniej modelu *MAC*. Przyjęto ponadto założenie, że początkowy zbiór ścieżek analizy sprzedaży definiowano na podstawie modelu *ER* szczególnego magazynu danych oraz wstępnych potrzeb analityka w zakresie wielowymiarowej analizy informacji. Pokazane na rys. 10 przykładowe ścieżki analizy zgrupowano w cztery wymiary.



Rys. 10. Przykłady ścieżek analizy sprzedaży
Fig. 10. The analytical sales paths examples

Najbardziej szczegółowy poziom każdego wymiaru odpowiada podstawowym własnościom sprzedawanych produktów, tak jak to zarejestrowano w systemie transakcyjnym. W zaproponowanym podejściu, oprócz zapytań konstruowanych na podstawie przedstawionych ścieżek analizy mogą pojawiać się pewne zapytania pomocnicze typu *ad-hoc*, które mogą wymagać zdefiniowania ich własnych ścieżek analizy. W końcu, tak określone przykładowe ścieżki analizy sprzedaży poddano przekształceniu, wykorzystując formalny modelu schematu ścieżek analizy wielowymiarowej, do odpowiedniego początkowego schematu ścieżek wielowymiarowej analizy sprzedaży.

3.3. Początkowy schemat ścieżek wielowymiarowej analizy sprzedaży

Dla przedstawionego powyżej początkowego zbioru przykładowych ścieżek analizy określono, na podstawie definicji 8, przykładowy początkowy schemat ścieżek analizy wielowymiarowej. Stanowi on grupę powiązanych danych $\mathbf{S}_{przykł} = (M,W,P,S)$. W tym określeniu M = {Sprzedaż wartościowa według...,} jest miarą definiowaną i reprezentowaną przez atrybut *WAR* z encji *SPRZEDAŻ* z przedstawionego wcześniej schematu *ER* szczególnego magazynu danych, natomiast W = {KLIENCI, CZAS, WYROBY, UMOWY}. Dla tak określonych wymiarów analizy sprzedaży określono przykładowe ścieżki analizy.

1. Dla wymiaru $\mathbf{W_1}$ = {KLIENCI} określono dwie ścieżki analizy, tj.:
   $P_{11}=\{p_{111} \cup p_{112} \cup p_{113}\}$, gdzie: p_{111} = {Klient}, p_{112} = {Miasto}, p_{113} = {Kod pocztowy}
   $P_{12}=\{p_{121} \cup p_{122}\}$, gdzie: p_{121} = {Klient}, p_{122} = {Typ klienta}
2. Dla wymiaru $\mathbf{W_2}$ = {CZAS} określono jedną ścieżkę analizy, tj.:
   $P_{21}=\{p_{211} \cup p_{212} \cup p_{213} \cup p_{214}\}$, gdzie: p_{211} = {Data}, p_{212} = {Miesiąc}, p_{213} = {Rok}, p_{214} = {Lata}
3. Dla wymiaru $\mathbf{W_3}$ = {WYROBY} określono trzy ścieżki analizy, tj.:
   $P_{31}=\{p_{311} \cup p_{312}\}$, gdzie: p_{311} = {Wyrób}, p_{312} = {Zakład produkcyjny}
   $P_{32}=\{p_{321} \cup p_{322}, \cup p_{323} \cup p_{324}\}$,
   gdzie p_{321} = {Wyrób}, p_{322} = {Grupa wyrobów}, p_{323} = {Klasa wyrobów}, p_{324} = {Rodzaj wyrobów}
   $P_{33}=\{p_{331} \cup p_{332}\}$, gdzie: p_{331} = {Wyrób}, p_{332} = {Magazyn}
4. Dla wymiaru $\mathbf{W_4}$ = {UMOWY} określono dwie ścieżki analizy, tj.:
   $P_{41}=\{p_{411} \cup p_{412}\}$, gdzie: p_{411} = {Zamówienie}, p_{412} = {Ilość planowana}
   $P_{42}=\{p_{421} \cup p_{422}\}$, gdzie: p_{421} = {Zamówienie}, p_{422} = {Ilość zamówiona}

Dla tak określonych zbiorów poziomów analizy w poszczególnych wymiarach, wynikają następujące zbiory uporządkowanych par, tj.:

1. dla wymiaru $\mathbf{W_1}$ = {KLIENCI},
   ścieżka 1: $S_{11}=\{(p_{111}, p_{112}), (p_{111}, p_{113}), (p_{112}, p_{113})\}$
   ścieżka 2: $S_{12}=\{(p_{121}, p_{122}), (p_{121}, p_{123})\}$
2. dla wymiaru $\mathbf{W_2}$ = {CZAS},
   ścieżka 1: $S_{21} = \{(p_{211}, p_{212}), (p_{211}, p_{213}), (p_{211}, p_{214}), (p_{212}, p_{213}), (p_{212}, p_{214}), (p_{213}, p_{214})\}$
3. dla wymiaru $\mathbf{W_3}$ = {WYROBY},
   ścieżka 1: $S_{31} = \{(p_{311}, p_{312})\}$
   ścieżka 2: $S_{32} = \{(p_{321}, p_{322}),(p_{321}, p_{323}),(p_{321}, p_{324}),(p_{322}, p_{323}),(p_{322}, p_{324}),(p_{323}, p_{324})\}$
   ścieżka 3: $S_{33} = \{(p_{331}, p_{332})\}$
4. dla wymiaru $\mathbf{W_4}$ = {UMOWY},
   ścieżka 1: $S_{41} = \{(p_{411}, p_{412})\}$
   ścieżka 2: $S_{42} = \{(p_{421}, p_{422})\}$

które wyznaczają następujące zbiory ukierunkowanych ścieżki analizy $s_{ij, xy}=(p_{ijx}, p_{ijy})$.

I tak, dla wymiaru $\mathbf{W_1}$ = {KLIENCI} mamy następujące ukierunkowane ścieżki analizy:

ścieżka 1, $s_{11, 12} = (p_{111}, p_{112})$, $s_{11, 13} = (p_{111}, p_{113})$, $s_{11, 23} = (p_{112}, p_{113})$, zatem $S_{11} = \{s_{11, 12}, s_{11, 13}, s_{11, 23}\}$
ścieżka 2, $s_{12, 12} = (p_{121}, p_{122})$, $s_{12, 13} = (p_{121}, p_{123})$, zatem $S_{12} = \{s_{12, 12}, s_{12, 13}\}$

Dla wymiaru $\mathbf{W_2}$ = {CZAS}, ukierunkowane ścieżki analizy przedstawiają się następująco:

ścieżka 1, $s_{21,12}=(p_{211},p_{212})$, $s_{21,13}=(p_{211},p_{213})$, $s_{21,14}=(p_{211},p_{214})$, $s_{21,23}=(p_{212},p_{213})$, $s_{21,24}=(p_{212},p_{214})$, $s_{21,34}=(p_{213},p_{214})$

zatem $S_{21}=\{s_{21,12}, s_{21,13}, s_{21,14}, s_{21,23}, s_{21,24}, s_{21,34}\}$

Dla wymiaru $\mathbf{W_3}$ = {WYROBY}, mamy:

ścieżka 1, $s_{31,12}=(p_{311}, p_{312})$, zatem $S_{31}=\{s_{31,12}\}$

ścieżka 2, $s_{32,12}=(p_{321},p_{322})$, $s_{32,13}=(p_{321},p_{323})$, $s_{32,14}=(p_{321},p_{324})$, $s_{32,23}=(p_{322},p_{323})$, $s_{32,24}=(p_{322},p_{324})$, $s_{32,34}=(p_{323},p_{324})$

zatem $S_{32}=\{s_{31,12}, s_{32,13}, s_{32,14}, s_{32,23}, s_{32,24}, s_{32,34}\}$

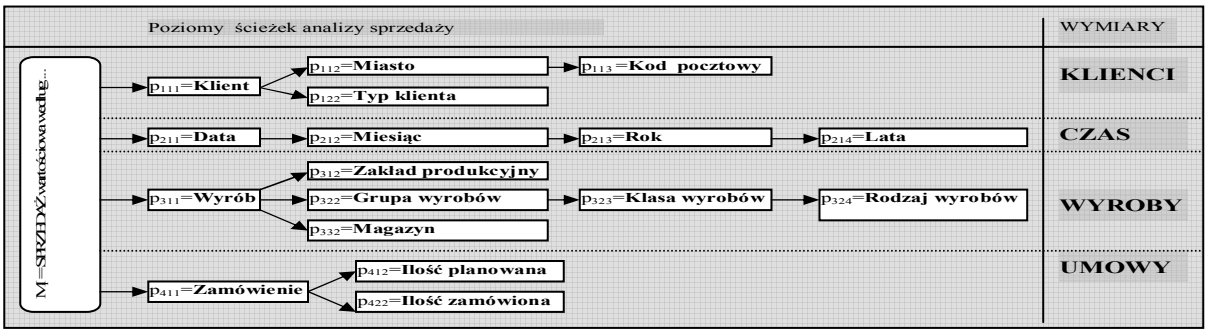
ścieżka 3, $s_{33,12}=(p_{331},p_{332})$, zatem $S_{33}=\{s_{33,12}\}$

Dla wymiaru $\mathbf{W_4}$ = {UMOWY}, mamy:

ścieżka 1, $s_{41,12}=(p_{411}, p_{412})$, zatem $S_{41}=\{s_{31,12}\}$

ścieżka 2, $s_{42,12}=(p_{421}, p_{422})$, zatem $S_{42}=\{s_{42,12}\}$

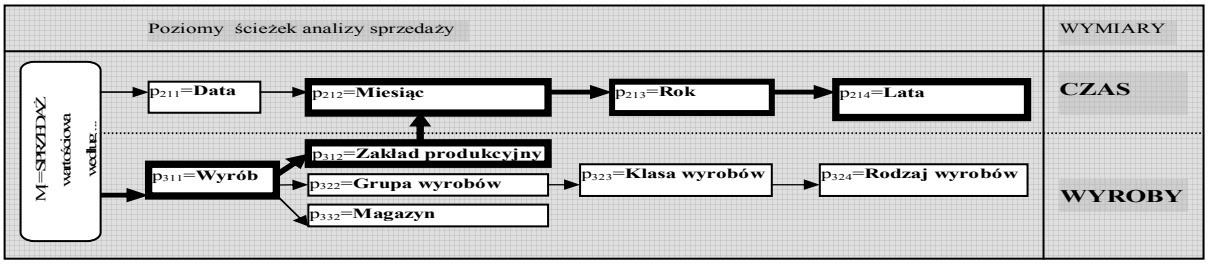
Uwzględniając, że $p_{111}=p_{121}=$ {Klient} oraz $p_{311}=p_{321}=p_{331}=$ {Wyrób}, otrzymano wynikowy schemat ścieżek wielowymiarowej analizy sprzedaży, który przedstawiono na rys. 11.



Rys. 11. Przykład schematu ścieżek wielowymiarowej analizy sprzedaży
Fig. 11. The multidimensional analytical sales paths schema example

3.3.1. Ścieżki skrośne w schemacie ścieżek wielowymiarowej analizy sprzedaży

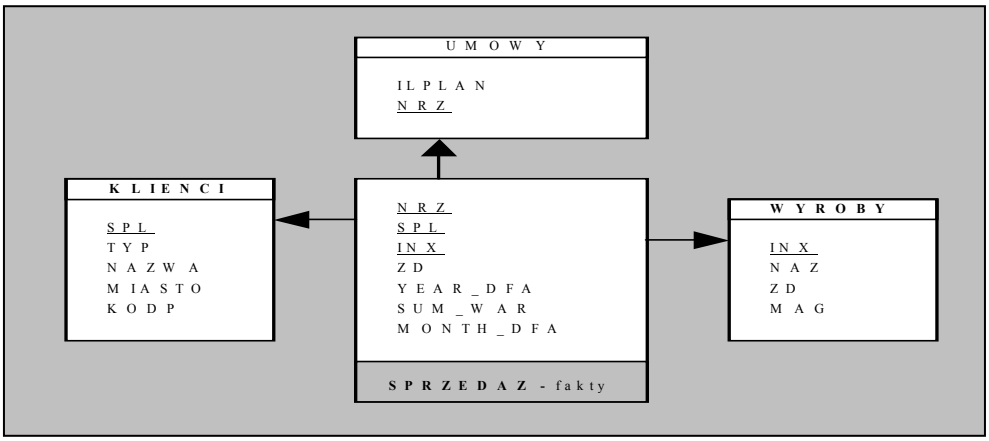
Na podstawie otrzymanego przykładowego schematu ścieżek wielowymiarowej analizy sprzedaży, możliwe są do zestawienia inne, tj. proste i złożone skrośne ścieżki analizy sprzedaży. Jeśli dla takiego schematu przyjąć, że $s_{ij,vx}=s_{31,12}$ oraz $s_{mn,yz}=s_{21,24}$, wówczas dla tak określonych ukierunkowanych ścieżek analizy, złożoną ścieżką skrośną wielowymiarowej analizy sprzedaży, jest ścieżka $s_{cross}=s_{31,12}\cup s_{21,24}$ lub prościej *Wyrób→Zakład Produkcyjny→Miesiąc→Rok→Lata*, którą pokazano na rys. 12.



Rys. 12. Przykład złożonej skrośnej ścieżki analizy sprzedaży
Fig. 12. Example of the cross path folded up of the sales analysis

3.4. Początkowy schemat Magazynu Danych Zbiorczych

Jak już wspomniano we wprowadzeniu, początkowy schemat MDZ określano za pomocą metody [20]. W celu jego określania za pomocą tej metody, przyjęto założenie, że wykorzystane będą tylko te encje *ER* ze szczególnego magazynu danych, które w kontekście początkowego schematu ścieżek analizy wielowymiarowej zawierają istotne informacje. Dla wspomnianego już przykładowego schematu *ER* do encji transakcyjnych należą encje *SPRZEDAŻ-FAKTURY*, *SPRZEDAŻ* oraz *ZAPLATY*. Do encji komponentowych należą zaś encje *KLIENCI*, *WYROBY* oraz *UMOWY*, natomiast do encji klasyfikujących encja *PLATNOSC*. Tak więc, na podstawie cytowanej metody, w tym konkretnym przypadku możliwe do utworzenia są dwa schematy hurtowni danych typu gwiazda, w których encję faktów tworzą encje transakcyjne. Do dalszych prac i analiz wybrano ten ze schematów, w którym encja faktów jest formowana na podstawie encji *SPRZEDAŻ*. Atrybuty grupujące i agregujące w encji faktów tj. *SPRZEDAŻ-fakty* utworzono na podstawie istniejących atrybutów z encji *SPRZEDAŻ*. I tak, do atrybutów grupujących w encji faktów należą *SPL* (Symbol Płatnika), *INX* (Indeks Wyrobu), *MONTH_DFA* (Miesiąc wystawienia faktury) oraz *YEAR_DFA* (Rok wystawienia faktury). Dwa ostatnie atrybuty utworzono na bazie atrybutu *DFA* (Data Faktury) z encji *SPRZEDAŻ*. Do atrybutów agregujących należy *SUM_WAR* (Suma Wartości) utworzony na bazie atrybutu *WAR* (Wartość Pozycji) z encji *SPRZEDAŻ*. Do formownia encji wymiarów na podstawie encji komponentowych wykorzystano encje *KLIENCI, WYROBY* oraz *UMOWY*. Encji *PLATNOSC* należącej również do encji komponentowych nie brano na tym etapie pod uwagę z tego względu, iż wymiaru *PŁATNOŚCI* nie uwzględniono w schemacie ścieżek analizy wielowymiarowej. Tak więc, otrzymany na podstawie tej metody początkowy schemat przykładowego MDZ, przedstawiony na rys. 13, jest schematem typu gwiazda, w którym poziomy wymiaru czasu (lata, rok, miesiące) zdefiniowano w encji faktów. W końcu, MDZ o poniższym schemacie zasilono pochodzącymi z szczególnego magazynu danych odpowiednio zagregowanymi na różnych poziomach danymi.



Rys. 13. Przykład początkowego schematu Magazynu Danych Zbiorczych
Fig. 13. The initial data warehouse schema example

3.4.1. Ocena początkowego schematu Magazynu Danych Zbiorczych

Nawet gdyby początkową postać tego schematu otrzymano za pomocą zastosowanej metody z kompletnego schematu *ER* szczególnego magazynu danych, to i tak trudno byłoby ocenić, czy możliwe jest zrealizowanie 'większości' zapytań analitycznych i tym samym stwierdzić, że jest on 'właściwie' określony. Powszechnie zadawane i kierowane do hurtowni danych zapytania analityczne można pogrupować w pewne klasy. Klasy te w jawny sposób określono na podstawie analizy przykładowych zapytań analitycznych zawartych w pracy [38]. Wyróżniono w ten sposób trzy klasy zapytań analitycznych.

A) Klasa pytań dotyczących jednej miary, zawierających:
   - pytania typu **Q1**, dotyczące jednej miary względem jednej ścieżki z dwóch wymiarów,
   - pytania typu **Q2**, dotyczące jednej miary względem dwóch ścieżek z jednego wymiaru.

B) Klasa pytań dotyczących dwóch miar, czyli
   - pytania typu **Q3**, dotyczące dwóch miar względem jednej ścieżki z dwóch wymiarów.

C) Klasa pytań dokonujących selekcji opartej na wcześniej zagregowanych danych na różnych poziomach; są to pytania zagnieżdżone typu **Q4**, dotyczące jednej miary względem jednej ścieżki kilku wymiarów, w których zagnieżdżony operator selekcji bazuje na wcześniej zagregowanych danych.

Tak więc, za wyjątkiem pytań typu Q3, których na tym etapie rozwoju początkowego schematu MDZ nie można formułować, ponieważ określony wcześniej schemat ścieżek analizy wielowymiarowej dotyczy tylko jednej miary (tj. sprzedaż według ...), pozwala jednak on na zrealizowanie każdego innego pytania analitycznego należącego do pozostałych klas.

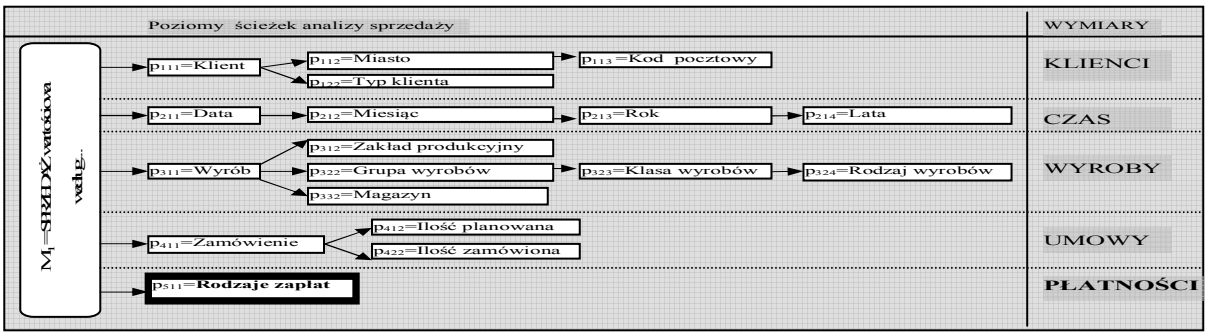
3.5. Dynamiczne rozszerzanie schematu Magazynu Danych Zbiorczych

W celu wykazania przydatności zaproponowanej koncepcji projektowania hurtowni danych, analizie poddano wpływ niektórych przykładowych zapytań analitycznych na początkową postać schematu MDZ. Należały one do jednej z trzech, wspomnianych we wprowadzeniu, kategorii zapytań. Poniżej przedstawiono przykłady zapytań analitycznych, na podstawie których w dynamiczny sposób rozszerzano początkowy schemat MDZ.

3.5.1. Wpływ zapytań analitycznych na postać początkowego schematu MDZ, należących do kategorii pytań zwiększających liczbę wymiarów

Zapytanie analityczne I: '*Podaj grudniową sprzedaż według rodzajów płatności w latach 1993-1994*'. Pytanie to stanowi dobrą ilustrację sytuacji, w której zaproponowane i omawiane w niniejszej pracy podejście okazuje się uzasadnione. Sytuacja taka może być spowodowana różnymi względami, najczęściej takimi, w której wiedza na temat zbioru pytań analitycznych na etapie projektowania jest ograniczona lub jeśli nie wiadomo, kiedy pojawią

się nowe zapytania analityczne. Jest to przykład pytania typu Q1 z klasy A, należącego równocześnie do I kategorii pytań zwiększających ilość wymiarów MDZ. Wymiarem w tym wypadku stanowią *PŁATNOŚCI* za sprzedany towar. Ponieważ jest to zapytanie analityczne, którego nie skonstruowano w oparciu wcześniej określony schemat ścieżek analizy wielowymiarowej, należy zatem tak go rozszerzyć, aby uwzględniał zaistniałą sytuację. Rozszerzono go zatem o brakujący wymiar i ścieżkę analizy. Schemat ten po rozszerzeniu przybrał postać uwidocznioną na rys. 14.



Rys. 14. Nowy wymiar PŁATNOŚCI w rozszerzonym schemacie ścieżek analizy
Fig. 14. The new dimension PŁATNOŚCI in the extended analytical paths schema

Na podstawie tego schematu, właściwą ścieżką analizy do konstrukcji ww. zapytania jest złożona ścieżka skrośna postaci: *Rodzaje zapłaty→Miesiąc→Rok→Lata*. Ponieważ w tym zapytaniu żądano pewnych informacji zbiorczych, których w MDZ nie ma, zatem ww. zapytanie analityczne nie mogło być do niego na tym etapie kierowane. Analizę tego zapytania prowadzono według przedstawionego już algorytmu analizy zapytań analitycznych, za pomocą bloku odpowiedzialnego za identyfikację faktów, wymiarów oraz niezbędnych atrybutów. Jej wyniki pozwalają stwierdzić, że pytanie to dotyczy zagregowanych danych dotyczących sposobów realizacji należności za sprzedane towary w tych latach, czyli sprzedaż wartościowa według rodzajów zapłaty. Selekcji żądanych informacji należało dokonać względem nieistniejącego jeszcze w schemacie MDZ wymiaru PŁATNOŚCI w funkcji wymiaru CZAS-u. Aby zapytanie analityczne mogło być zrealizowane, zachodziła najpierw konieczność określenia i wykonania następującego zapytania pomocniczego: '*Jak kształtowała się wartość sprzedaży według poszczególnych rodzajów płatności w poszczególnych miesiącach w latach 1991-2001*'. Wyrażone w składni języka SQL przyjęło następującą postać:

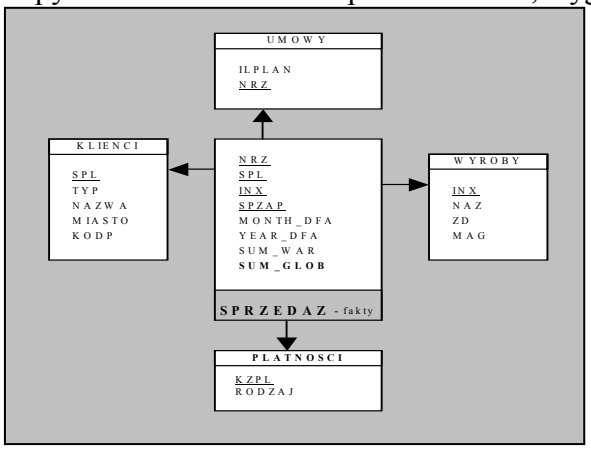
SELECT YEAR(DFA), MONTH(DFA), SPZAP, SUM(GLOB) **FROM** ZAPLATY **GROUP BY** YEAR(DFA), MONTH(DFA), SPZAP

Jak wcześniej wspomniano, przy określaniu początkowego schematu MDZ nie uwzględniono encji *ZAPLATY*, należącej do encji komponentowych. Ponieważ relacja o tej samej nazwie ze szczególnego magazynu danych, zawiera istotne z punktu widzenia realizacji tego pytania informacje, dlatego też ona jest adresatem kierowanego do niej tego zapytania, we wszystkich składowych bazach danych. Wykonanie powyższego zapytania za pośrednictwem systemu

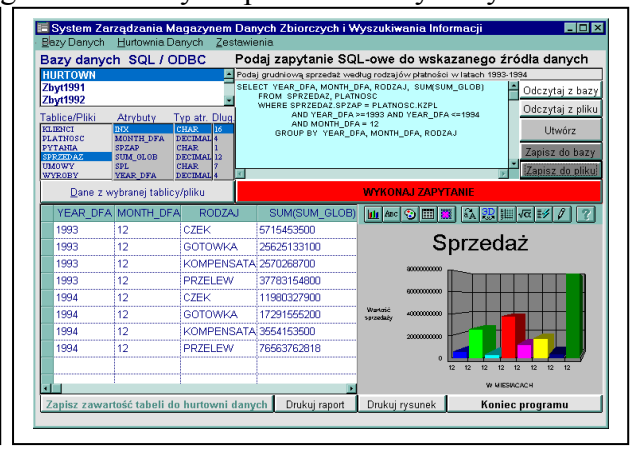
zarządzania MDZ, a następnie zapisanie otrzymanego rezultatu w tablicy faktów MDZ, spowodowało podczas zapisywania wyników jej dynamiczne rozszerzenie, za pomocą zaproponowanej metody. Przyjęła ona ostatecznie postać: *SPRZEDAZ(INX, SPL, SPZAP, YEAR_DFA, MONTH_DFA, SUM_WAR, SUM_GLOB)*. Jak już wspomniano, wymiary względem których należało dokonać selekcji potrzebnych informacji to *PŁATNOŚCI* w funkcji *CZAS*-u. Jak do tej pory wymiaru *PŁATNOŚCI* w MDZ jeszcze nie określono. Aby to zapytanie mogło być zrealizowane, zachodzi konieczność określenia pytania pomocniczego postaci: „*Wybierz poprawne dane o rodzajach płatności”,* czyli „**SELECT** KZPL, RODZAJ **FROM** PLATNOSC”. Pytanie to skierowano najpierw do poszczególnych składowych baz w szczególnym magazynie danych, a następnie zapisano otrzymany rezultat w MDZ. Podczas zapisywania wyników dynamicznie rozszerzono jego schemat o nową tablicę, tj.: *PLATNOSC(KZPL, RODZAJ)*, co przedstawiono na rys. 15. Zatem zapytanie analityczne ‘*Podaj grudniową sprzedaż według rodzajów płatności w latach 1993-1994*’ wyrażone w języku SQL przyjmuje ostatecznie poniższą postać:

SELECT YEAR_DFA, MONTH_DFA, RODZAJ, SUM(SUM_GLOB) **FROM** SPRZEDAZ, PLATNOSC
WHERE SPRZEDAZ.SPZAP = PLATNOSC.KZPL **AND** YEAR_DFA >=1993 **AND** YEAR_DFA <=1994 **AND** MONTH_DFA = 12
GROUP BY YEAR_DFA, MONTH_DFA, RODZAJ

Zapytanie to skierowano wprost do MDZ, wygenerowało wynik przedstawiony na rys. 16.



Rys. 15. Nowy wymiar PLATNOSCI schemacie MDZ
Fig. 15. The new dimension PLATNOSCI in the MDZ schema

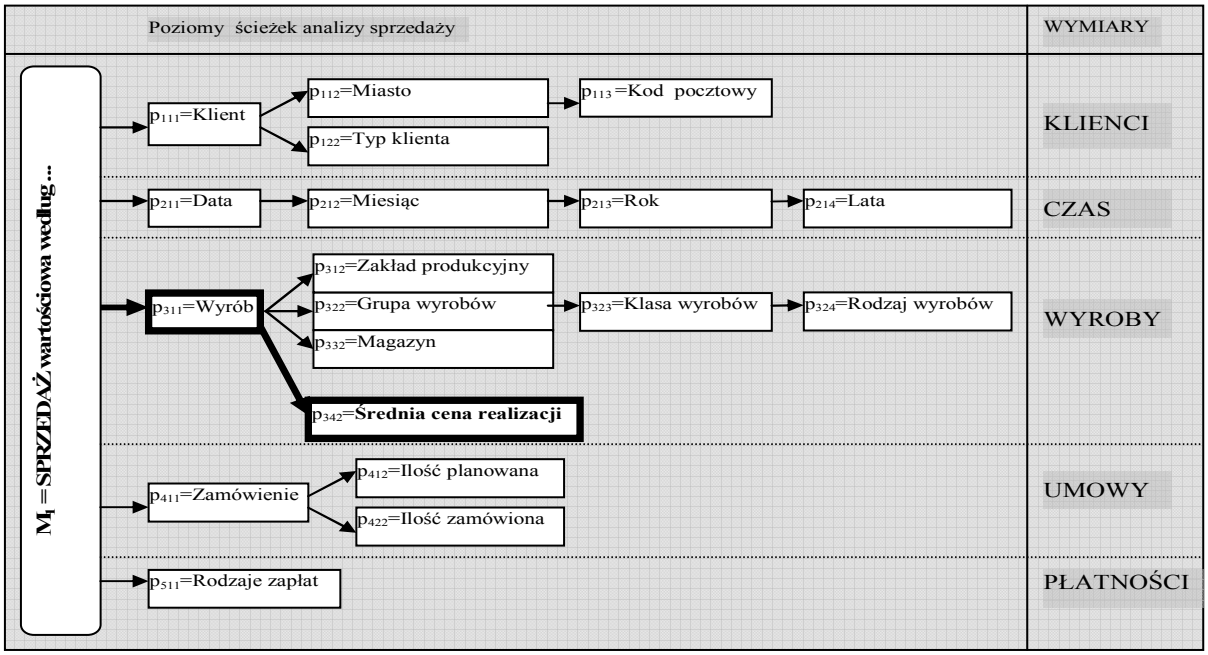


Rys 16. Pytanie: Podaj grudniową sprzedaż według rodzajów płatności w latach 1993-1994
Fig 16. Question: Give the december’s sales according to payment kind in 1993-1994 years

3.5.2. Wpływ zapytań analitycznych na postać początkową schematu MDZ, z kategorii zapytań zwiększających liczbę ścieżek analizy w ramach pewnego wymiaru

Zapytanie analityczne II: ‘*Podaj marcową średnią cenę realizacji transakcji sprzedaży wyrobu X w latach 1991-1994*’. Jest to przykład pytania typu Q1 z klasy A, należącego równocześnie do II kategorii pytań zwiększających liczbę ścieżek analizy w ramach pewnego wymiaru. Ponieważ jest to zapytanie analityczne, którego nie skonstruowano w oparciu o

wcześniej określony schemat ścieżek analizy wielowymiarowej, zatem należy go tak rozszerzyć, aby uwzględniał zaistniałą sytuację. Po jego rozszerzeniu o brakującą ścieżkę analizy w wymiarze *WYROBY*, uzyskano nowy schemat ścieżek, przedstawiony na rys. 17.



Rys. 17. Nowa ścieżka analizy w rozszerzonym schemacie ścieżek analizy
Fig. 17. The new analytical path in the extended analytical paths schema

Tak więc ścieżką analizy właściwą do konstrukcji ww. zapytania analitycznego jest złożona ścieżka skrośna postaci: *Wyrob→Średnia cena realizacji→Miesiąc→Rok→Lata*. Analiza tego zapytania prowadzona według algorytmu analizy pytań analitycznych pozwala stwierdzić, że wyniki realizacji tego pytania to stanowią podzbiór pewnego zbioru, zawierającego zagregowane dane dotyczące średnich cen realizacji transakcji sprzedaży wyrobów w poszczególnych miesiącach w latach 1991-2001. Zatem, aby ww. zapytanie analityczne mogło być zrealizowane, zachodzi najpierw konieczność określenia i wykonania następującego zapytania pomocniczego: '*Podaj średnie ceny realizacji transakcji sprzedaży wyrobów w poszczególnych miesiącach roku...*', czyli:

SELECT YEAR(DFA), MONTH(DFA), SPRZEDAZ.INX, AVG(CET) **FROM** SPRZEDAZ
GROUP BY YEAR(DFA), MONTH(DFA), SPRZEDAZ.INX

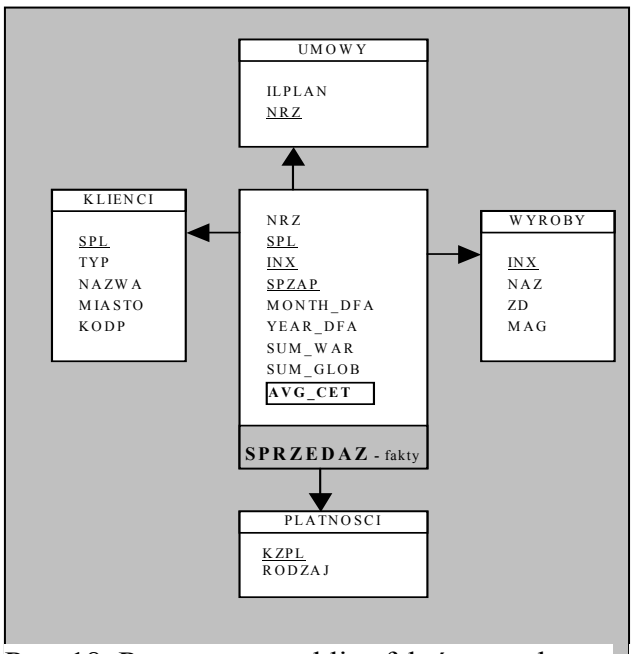
Wykonanie powyższego zapytania, skierowanego do składowych baz danych z lat 1991-2001 roku za pośrednictwem systemu zarządzania MDZ, a następnie zapisanie otrzymanego rezultatu w tablicy faktów w MDZ, spowoduje dynamiczne jej rozszerzenie o nowy atrybut *AVG_CET*. Przyjmuje ona teraz postać następującą: *SPRZEDAZ(INX, SPL, SPZAP, YEAR_DFA, MONTH_DFA, SUM_WAR, SUM_GLOB, AVG_CET)*. W ten sposób dokonano dalszego rozszerzenia schmatu MDZ do postaci, którą przedstawiono na rys. 18. Tak więc,

rozszerzony MDZ posiada już odpowiedni schemat i zawiera odpowiednio zagregowane informacje niezbędne do poprawnego skonstruowania wspomnianego II zapytania analitycznego. I tak poszukiwana w tym zapytaniu miara z tablicy faktów to *sprzedaż według... średnich cen realizacji*, którą reprezentuje atrybut *AVG_CET*. W tym zapytaniu poszukiwany jest pewien wyrób, którego nazwa reprezentowana są przez atrybut *NAZ* (Nazwa Klienta z tablicy *WYROBY*). Wymiary względem których dokonuje się selekcji to *WYROBY* oraz *CZAS* (zawarty w tablicy faktów). W MDZ znajdują się zagregowane dotyczące średnich cen realizacji transakcji sprzedaży wyrobów w poszczególnych miesiącach z lat 1991-1994.

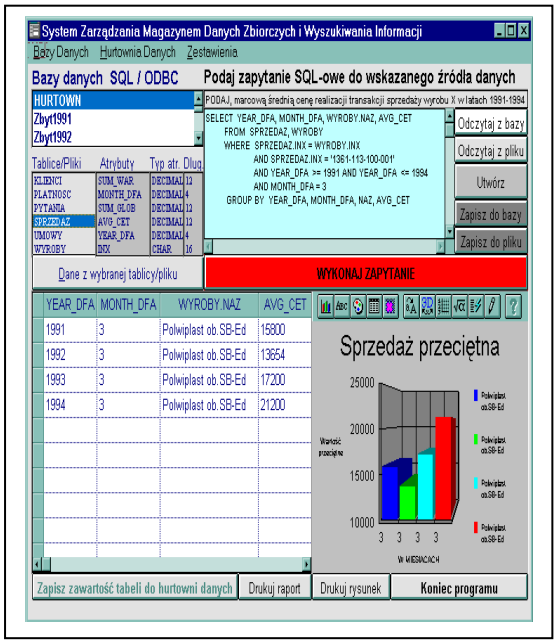
Zatem w końcowym zapytaniu analitycznym wyrażonym w języku SQL zaangażowane będą atrybuty grupujące *YEAR_DFA, MONTH_DFA, NAZ* oraz *AVG_CET*. Poszukiwane *marcowe średnie ceny realizacji transakcji sprzedaży wyrobu X w latach 1991-1994* stanowią odpowiedź na postawione II zapytanie analityczne skierowane wprost do MDZ. Pytanie to wyrażone w języku SQL przyjmuje ostatecznie poniższą postać:

```
SELECT  YEAR_DFA, MONTH_DFA, WYROBY.NAZ, AVG_CET FROM  SPRZEDAZ, WYROBY
    WHERE  SPRZEDAZ.INX = WYROBY.INX
        AND SPRZEDAZ.INX = '1361-113-100-001'
        AND YEAR_DFA  >= 1991 AND YEAR_DFA  <= 1994 AND MONTH_DFA = 3
    GROUP BY  YEAR_DFA, MONTH_DFA, NAZ, AVG_CET
```

Po skierowaniu tego zapytania do MDZ otrzymano wynik, który przedstawiono na rys. 19. Reasumując, powyżej przedstawiono analizę zapytania analitycznego, którego wpływ na postać schamatu MDZ zaznaczył się rozszerzeniem tablicy faktów o nowy atrybut.



Rys. 18. Rozszerzona tablica faktów w schemacie magazynu danych
Fig. 18. The extended fact table in the data warehouse schema



Rys 19. Pytanie: Jaka była marcową średnią cenę realizacji transakcji sprzedaży wyrobu X w latach 1991-1994
Fig 19. Question: What was the march's average transaction realization sales prize of the X product in 1993-1994 years

W ramach dyskusji nad pytaniami analitycznymi należącymi do II kategorii pytań zwiększających liczbę ścieżek analizy w ramach pewnego wymiaru, poniżej przedstawiono analizę innego zapytania. Wpływ tego zapytania na postać schematu MDZ zaznaczył się w inny sposób. Realizacja tego zapytania spowodowała rozszerzenie dotychczas otrzymanego schematu MDZ do postaci, którą zakwalifikować można do postaci typu płatka śniegu.

Zapytanie analityczne IIa: '*Jaka była grudniowa sprzedaż zakładu X w latach 1991-1993*'

Jest to przykład pytania typu Q1 z klasy A, należącego równocześnie do wspomnianej II kategorii pytań. W przeciwieństwie do II zapytania analitycznego, jest to zapytanie, które skonstruowano w oparciu o wcześniej określony schemat ścieżek analizy wielowymiarowej, tj. w oparciu o złożoną ścieżkę skrośną postaci: *Wyrob→Zakład produkcyjny→Miesiąc→Rok→Lata*. W tym zapytaniu poszukiwane są zakłady produkcyjne reprezentowane przez atrybut *ZD* (Zakład). Wymiary względem których dokonuje się selekcji, to *WYROBY* oraz *CZAS*. Dalej można stwierdzić, że zapytanie dotyczy zagregowanych danych dotyczących sprzedaży poszczególnych zakładów w kolejnych miesiącach, na przestrzeni lat 1991-1993. Aby udzielić poprawnej odpowiedzi na ww. zapytanie, wymagane jest zrealizowanie pytania pomocniczego, tj.: '*Jaka była wartość sprzedaży poszczególnych zakładów w poszczególnych latach*'. Wyrażone w języku SQL przyjmuje następującą postać:

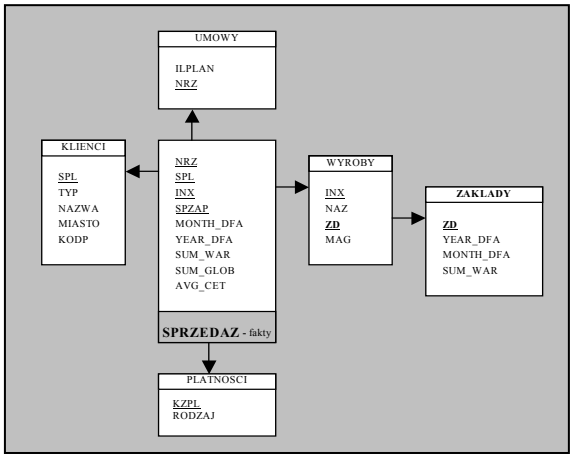
```
SELECT YEAR(DFA), MONTH(DFA), ZD, SUM(WAR) FROM SPRZEDAZ
    GROUP BY YEAR(DFA), MONTH(DFA), ZD
```

Skierowano go za pośrednictwem systemu zarządzania MDZ do poszczególnych archiwalnych baz danych z lat 1991-2001. Danych zagregowanych stanowiących wynik tego zapytania nie zapisano jednak w tablicy faktów MDZ. Korzystając z algorytmu mechanizmu dynamicznego rozszerzania schematu MDZ, wyniki zapisano w odrębnej, nowo utworzonej tablicy wymiarów o nazwie *ZAKŁADY*, tworząc w ten sposób nowy wymiar dla wyższego poziomu ziarnistości danych. Tak wiec za pomocą tego mechanizmu, jej schemat określono następująco: *ZAKŁADY(YEAR_DFA, MONTH_DFA, ZD, SUM_WAR)*. W ten sposób dokonano dalszego rozszerzenia schmatu MDZ, którego schemat typu płatka śniegu przedstawiono na rys. 22. Zaprezentowano w ten sposób możliwość dowolnego kształtowania schematu.

W tym miejscu należy zauważyć, że zastosowanie metody [20], do nawet pełnego schematu *ER* szczególnego magazynu danych, nie może wygenerować, przedstawionego na tym rys. 20, schematu MDZ z tak określonym wymiarem. Jest to spowodowane tym, że schemat *ER* w ogóle nie zawiera encji komponentowej *ZAKLADY*. Pomimo tego, że istniejący w tej metodzie operator agregacji, który zastosowany do encji transakcyjnej umożliwia utworzenie nowej encji zawierającej zagregowane dane, niemniej jednak w celu utworzenia takiej encji konieczna byłaby znajomość analizowanego zapytania już na etapie projektowania schematu MDZ. Na tym prostym przykładzie widać, że początkowy schemat MDZ określony tą metodą nie jest schematem 'właściwym', w związku z tym nie jest możliwa realizacja większości

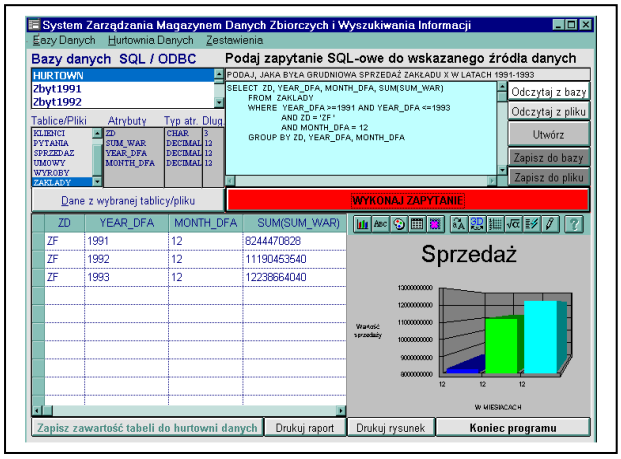
zapytań analitycznych. Realizacja nowych pytań *ad-hoc* wymaga jego rozszerzenia. Reasumując, przy tak określonym schemacie MDZ, zapytanie analityczne '*Jaka była grudniowa sprzedaż zakładu X w latach 1991-1993*', wyrażone w języku SQL może przyjąć poniższą postać, którego wynik przedstawiono na rys. 21.

```
SELECT ZD, YEAR_DFA, MONTH_DFA, SUM(SUM_WAR) FROM ZAKLADY
    WHERE YEAR_DFA >=1991 AND YEAR_DFA <=1993 AND ZD = 'ZF' AND MONTH_DFA = 12
    GROUP BY ZD, YEAR_DFA, MONTH_DFA
```



Rys. 20. Schemat magazynu danych typu płatek śniegu

Fig. 20. The snow flake schema type of the data warehouse



Rys. 21. Pytanie: Jaka była grudniowa sprzedaż zakładu X w latach 1991-1993

Fig. 21. Question: Wat was the december's sales X factory in 1991-1993 years

3.5.2. Wpływ pytań analitycznych na postać schematu MDZ, z kategorii pytań zwiększających liczbę poziomów w pewnej ścieżce analizy pewnego wymiaru,

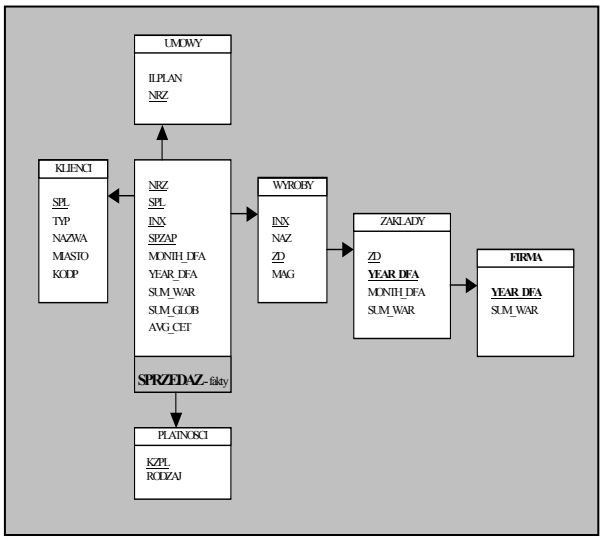
Zapytanie analityczne III: '*Podaj udział sprzedży majowej w stosunku do globalnej w latach 1991-1993*'. Jest to przykład zapytania typu Q1 z klasy A, należącego równocześnie do III kategorii pytań zwiększających liczbę poziomów w pewnej ścieżce analizy pewnego wymiaru. W przeciwieństwie do zapytania IIa jest to zapytanie analityczne, dla konstrukcji którego dokonano dalszego rozszerzenia schematu ścieżek analizy wielowymiarowej. Dodano do niego nową ścieżkę złożoną ścieżkę prostą postaci: *Wyrob→Zakład produkcyjny→firma*, która wraz ze ścieżką prostą z wymiaru *CZAS* daje złożoną skrośną ścieżkę postaci: *Wyrob→Zakład produkcyjny→firma→Miesiąc→Rok→Lata*. Dalej można stwierdzić, że zapytanie dotyczy zagregowanych danych dotyczących sprzedaży globalnej w latach 1991-1993. Zatem, aby udzielić poprawnej odpowiedzi na ww. zapytanie wymagane jest zrealizowanie pytania pomocniczego: '*Jaka była wartość sprzedaży w poszczególnych latach'*. Wyrażone w języku SQL przyjmuje następującą postać:

```
SELECT YEAR(DFA), SUM(WAR) FROM SPRZEDAZ GROUP BY YEAR(DFA)
```

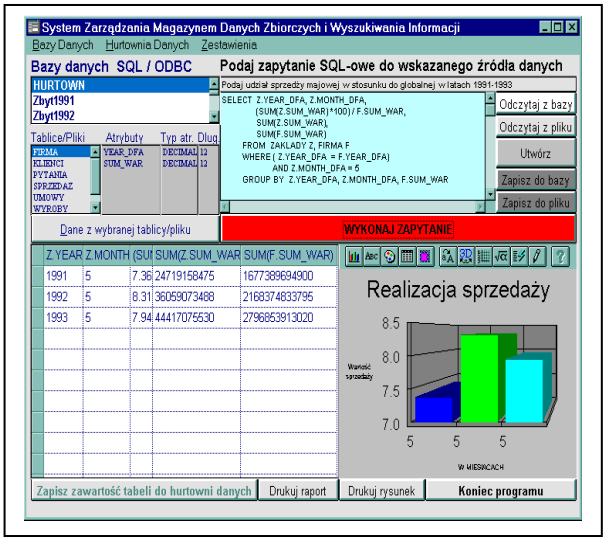
Pytanie to skierowano za pośrednictwem systemu zarządzania MDZ do poszczególnych składowych baz danych z lat 1991-2001. Wyników tego pytania, podobnie jak w zapytaniu IIa,

nie zapisano w tablicy faktów MDZ, lecz w dynamiczny sposób utworzonej tablicy *FIRMA(YEAR_DFA, SUM_WAR)*. W ten sposób dokonano dalszego rozszerzonia schmatu MDZ, którego schemat typu płatka śniegu przedstawiono na rys. 22. Przy tak określonym schemacie MDZ, zapytanie analityczne '*Podaj udział sprzedży majowej w stosunku do globalnej w latach 1991-1993*', wyrażone w języku SQL przyjmuje poniższą postać, którego wynik przedstawiono na rys. 23.

```
SELECT  Z.YEAR_DFA, Z.MONTH_DFA, (SUM(Z.SUM_WAR)*100)/F.SUM_WAR, SUM(Z.SUM_WAR),
        SUM(F.SUM_WAR)
    FROM  ZAKLADY Z, FIRMA F
    WHERE ( Z.YEAR_DFA  = F.YEAR_DFA) AND Z.MONTH_DFA = 5
    GROUP BY  Z.YEAR_DFA, Z.MONTH_DFA, F.SUM_WAR
```



Rys. 22. Rozszerzony schemat magazynu danych typu płatek śniegu
Fig. 22. The extended snow flake schema type of the data warehouse



Rys. 23. Pytanie: *Podaj udział sprzedży majowej w stosunku do globalnej w latach 1991-1993*
Fig. 23. Question: *Give the may's participation sale relative to global in 1991-1993 years*

3.6. Wnioski i ocena wpływu niektórych pytań analitycznych na postać dynamicznie rozszerzanego początkowego schematu Magazynu Danych Zbiorczych

W pracy analizie poddano wpływ niektórych zapytań analitycznych na postać schematu MDZ, które konstruowano na podstawie ścieżek analizy określonych w schemacie ścieżek analizy wielowymiarowej. Badano wpływ zapytań analitycznych należących do wszystkich, opisanych w rozdziale 2, kategorii zapytań. Dzięki zastosowaniu zmodyfikowanego algorytmu mechanizmu dynamicznego rozszerzania schematu MDZ, uzyskano możliwość swobodnego kształtowania jego postaci. Wpływ przykładowych zapytań analitycznych (IIa oraz III), należących do tych kategorii, na postać tego schematu był taki, że początkowy typ schematu MDZ określony jako gwiazda, doprowadzono do schematu postaci typu płatka śniegu. Zade-

monstrowano w ten sposób możliwość swobodnego kształtowania postaci dynamicznie rozszerzanego schematu MDZ. Wniosek zasadniczy wynikający z przeprowadzonej dyskusji jest następujący: zapytania analityczne należące do kategorii zapytań skutkujących zwiększeniem liczbę ścieżek analizy w ramach pewnego wymiaru lub też zapytania analityczne należące do kategorii zapytań skutkujących zwiększeniem liczby poziomów w pewnej ścieżce analizy pewnego wymiaru ze schematu ścieżek analizy wielowymiarowej, mogą, lecz nie muszą skutkować dynamicznym rozszerzeniem schematu MDZ w postaci nowej tablicy.
Jeśli natomiast chodzi o kwestię wpływu dyskutowanych zapytań analitycznych na powstanie w MDZ ewentualnych zależności pomiędzy danymi typu *wiele-do-wielu,* to wniosek wynikający z własności schematu ścieżek analizy wielowymiarowej można wyrazić następująco: ponieważ relacje łączące poziomy w tym schemacie są relacjami grupowania/klasyfikowania, zatem wynikowa postać schematu MDZ zawiera tylko relacje typu *wiele-do-jednego*.

4. Podsumowanie

W artykule przedstawiono aktualny przegląd oraz syntezę stanu wiedzy odnoszącego się do tradycyjnego podejścia do problemu statycznego projektowania, budowy hurtowni oraz ekstrakcji danych. Na podstawie przedstawionej syntezy zaproponowano alternatywne podejście do problemu projektowania i budowy hurtowni danych, biorąc pod uwagę pojawiające się w różnym czasie nowe pytania analityczne. W zaproponowanym podejściu wykorzystano metodę dynamicznego rozszerzania jej schematu. Jest ona obarczona istotnymi wadami, niemniej jednak w przypadku tworzenia małych hurtowni tematycznych, na podstawie istniejących zastanych przemysłowych systemów informacyjnych, może okazać się przydatna dzięki swej prostocie. Ponadto, w zaproponowanym podejściu, zastosowano metodę, która zapewnia ewolucję jej schematu w sytuacji, gdy pojawiają się nowe zapytania analityczne typu *ad-hoc*.
Zalety zaproponowanego podejścia projektowania hurtowni danych są następujące:

1) wyeliminowanie zbioru zapytań analitycznych określonych przez użytkownika końcowego, niezbędnego do zaprojektowania właściwego schematu hurtowni danych,
2) zapytania analityczne użytkownika mogą być formułowane *ad-hoc* w oparciu o kombinacje wcześniej określonych na podstawie zastanego modelu danych bazy *OLTP* ścieżek analizy, zapewniając tym samym możliwość realizacji takich pytań.

Natomiast podstawową wadą zaproponowanego podejścia jest konieczność określenia ścieżek analizy, na podstawie wymagań analityka oraz schematu *ER* z zastanego systemu informacyjnego. W przypadku braku takiego schematu, niewłaściwe zrozumienie przez projektanta związków w modelu danych tego systemu, prowadzić może do ustalenia niewłaściwych ścieżek analizy, co w konsekwencji prowadzi do formułowania zapytań analitycznych, któ-

rych nie można zrealizować. W przeciwieństwie do rozwiązania tradycyjnego, w którym zakłada się, że schemat hurtowni powinien bezpośrednio wynikać z wcześniej określonego zbioru pytań analitycznych, zaproponowane rozwiązanie jest bardziej elastyczne, ponieważ na etapie projektowania hurtowni zbędna staje się znajomość pełnego zbioru zapytań analitycznych. Znacznie łatwiej dla projektanta współpracującego z analitykiem jest określenie potencjalnych ścieżek analizy, niż kompletnego zbioru potencjalnych zapytań analitycznych.

LITERATURA

1. Grzywak A., Kozielski S., Irek P., Kmonk J., Pierzchała D.: Hurtownie danych. Projekt Celowy nr 8T11C 009 96C/2995, 1998.
2. Kimbal R.: A Dimensional Modeling Manifesto. DBMS, August 1997.
3. Chaudhuri S., Dayal U.: An Overview of Data Warehousing and OLAP Technology. Appears in ACM Sigmod Record, March 1997.
4. Moody D., Kortink M.: From ER Models to Star Schemas, 14th Australasian Conference on Information Systems, Page 1, 2003.
5. Abell A., Samos J., Saltor F.: A DataWarehouse Multidimensional Data Models Classication, Politecnica de Catalunya.
6. Phipps C., Davis K. C.: Automating Data Warehouse Conceptual Schema Design and Evaluation, DMDW, 2002.
7. Golfarelli M., Rizzi S.: Designing the Data Warehouse: Key Steps and Crucial Issues. In the Journal of the Computer Science and Information Management, Vol.2, N.3,1999.
8. Golfarelli M, Rizzi S.: A Methodological Framework for Data Warehouse Design. In Proceedings of the Acm International Workshop on Data Warehousing, 1998.
9. Sen A., Atish P. Sinha A.P.,: A Comparison of Data Warehousing Methodologies, COMMUNICATIONS OF THE ACM, Vol. 48, No. 3, 2005.
10. Wrembel R., Koncilia C.: Data Warehouses and OLAP Concepts, Architectures and Solutions, IRM Press, ISBN 1-59904-364-5, 2007.
11. Vassiliadis P., Sellis T.: A Survey on Logical Models for OLAP Databases. National Technical University of Athens, Technical Report DWQ NTUA 301, 1999.
12. Sapia C., Blaschka M, Höfling G., Dinter B.: Extending the E/R Model for the Multidimensional Paradigm. In Proceedings of the 1st Intl. Workshop on Data Warehouse and Data Mining (DWDM'98), Volume 1552, pages 105-116, 1998.
13. Hüsemann B., Lechtenbörger J., Vossen G.: Conceptual Data Warehouse Design. In Proceedings of the International Workshop on Design and Management of Data Warehouses (DMDW'2000), 2000.

14. Lechtenbörger J., Vossen G.: Multidimensional Normal Forms for Data Warehouse Design. University of Muenster, Bericht Nr. 9/01-1, 2001.
15. Phipps C., Davis K.: Automating Data Warehouse Conceptual Schema Design and Evalution. In Proceedings of 4th International Workshop (DMDW'2002), 2002.
16. Boehnlein M, Ende A.: Deriving Initial Data Warehouse Structures from the Conceptual Data Models of the Underlying Operational Information Systems. In Proceedings of the ACM Second International Workshop on Data Warehousing and OLAP, (DOLAP'99), 1999.
17. Golfarelli M., Maio D., Rizzi S.: Conceptual Design of Data Warehouses from E/R Schemas. In the Proceedings of the Hawaii International Conference On System Sciences, 1998.
18. Pajer M., Granat J., Majdan M., Kuśmierek M., Chudzian C., Salwa M., Sobieszek J.: Metody i technologie budowy hurtowni danych, Instytut Łączności, Praca nr 06.30.002.7, 2007.
19. Koszlajda T.: Technologia Magazynów Danych. III Konferencja użytkowników i developerów, Zakopane, 1997.
20. Moody D., Kortink M.: From Enteprise Models to Dimensional Models: A Methodology for Data Warhouse and Data Mart Design. In Proceedings of the International Workshop on Design and Management of Data Warehouses (DMDW2000), 2000.
21. Kotidis Y., Roussopoulos N.: DynaMat: A Dynamic View Management System for Data Warehouses. Sponsored by NASA, Grant NAG 5-2926, by NSA/Lucite under contract CG9815, In Proc. ACM/SIGMOD'99.
22. Theodoratos D., Sellis T.: Dynamic Data Warehouse Design. In Proceedings of the International Conference on Data Warehousing and Knowledge Discovery, (DaWa'99), Springer LNCS 1676, pp. 1 10, 1999.
23. Tekaya K.: Dynamic Distributed Data Warehouse Design, IRMA International Conference, 2007.
24. Zhao J., Schewe K.D.: Dynamic Data Warehouse Design with Abstract State Machines, Journal of Universal Computer Science, vol. 15, no. 1, 355-397, 2009.
25. Koehler H., Schewe K.D., Zhao J.: Dynamic Data Warehouse Design as a Refinement, Fourth Asia-Pacific Conference on Conceptual Modelling (APCCM 2007), 2007.
26. Lahanas S.: The Dynamic Data Warehouse, DATAVERSITY, 2011.
27. Czejdo B., Messa K., Morzy T., Morzy M., Czejdo J.: Data Warehouses With Dynamically Changing Schemas and Data Sources, 2008.
28. Pang C., Taylor K., Zhang X., Cameron M.: Generating multidimensional schemata from relational aggregation queries, 2004.

29. Solodovnikova D., Niedrite L., Niedritis A.: Query-Driven Method for Improvement of Data Warehouse Conceptual Model, International Conference on Information Systems Development, ESF project No. 2009/0216/1DP/1.1.1.2.0/09/APIA/VIAA/044, 2012.
30. Solodovnikova D., Niedrite L., Niedritis A.: Query-Driven Method for Improvement of Data Warehouse Conceptual Model, 2013.
31. Song I., Rowen W.: An Analysis of Many-to-Many Relationships Between facts and Dimension Tables in Dimensional Modeling. In Proceedings of the International Workshop on Design and Management of data Warehouses (DMDW'2001), 2001.
32. Muller R.J.: Bazy danych, język UML w modelowaniu danych. ISBN 83-7279-000-0, wydawnictwo MIKOM, 2000.
33. Grzywocz J.: Własności języka zapytań a opis bazy danych. Zeszyty Naukowe Politechniki Śląskiej, s. Informatyka, z. 25, 1994.
34. Boruta A., Grzywocz J., Kozielski S.: Wykorzystanie elementów języka naturalnego w systemie wyszukiwania opartym na modelu relacji uniwersalnej. Zeszyty Naukowe Politechniki Śląskiej, s. Informatyka, z. 27, 1994.
35. Kozielski S.: Języki zapytań relacyjnych baz danych a rozpraszanie obliczeń w sieci komputerowej. Zeszyty Naukowe Politechniki Śląskiej, s. Informatyka, z. 24, 1994.
36. Grzywocz J.: Metody opisu baz danych jako podstawa automatyzacji procesu wyszukiwania. Zeszyty Naukowe Politechniki Śląskiej, s. Informatyka, z. 29, 1995.
37. Bok Z.: Integracja relacyjnych baz danych w zastanych przemysłowych systemach informatycznych, Studia Informatica, Volume 23, Nr 4, Politechnika Śląska, 2002.
38. Tsois A., Karayannidis N.: MAC: Conceptual Data Modeling for OLAP. National Technical University of Athens, Zografou 15773, Athens, Greece, In Proceedings of the 3rt International Workshop DMDW'2001, 2001.
39. Niemi T., Nummenmaa J., Thanisch P.: Constructing OLAP Cubes Based On Queries. In Proceedings of the ACM International Workshop on Data Warehousing, 2001.
40. Blaschka M, Sapia C., Höfling G.: On Schema Evolution in Multidimensional Databases. In the Proceedings of First International Conference on Data Warehousing and Knowledge Discovery, 1999.
41. Cheung D., Zhou B., Kao B., Lu H., Lam T., Ting H.: Requirement Based Data Cube Schema Design. In Proceedings of the Eighth International Conference on Information Knowledge Management, pages 162-169, 1999.
42. Vassiliadis P.: Modeling Multidimensional Databases, Cubes and Cube Operations, National Technical University of Athens.
43. Kacprzyk J., Stańczak W.: Teoria grafów i jej zastosowania w informatyce, 1980.

44. Ignasiak E.: Teoria grafów i planowanie sieciowe. Państwowe Wydawnictwo Ekonomiczne, 1982.
45. Jankowski B.: Grafy, algorytmy w Pascalu. Wydawnictwo ”Mikom”, ISBN 83-7158-077-0, 1988.
46. Black P.E.: Directed acyclic graph, in Dictionary of Algorithms and Data Structures, 2004.

Recenzent:

Wpłynęło do Redakcji 1 grudnia 2017 r.

Abstract

In this article the dynamic method extending the schema data warehouse that uses the standard SQL-99 has been presented. Based on this method and dynamically extension data warehouse schema method using SQL-99 standard, a data warehouse design problem taking into account analytical queries formulated by end user has been discussed. In the proposed algorithm implemented in this method every new analytical query is analyzed at an angle of it’s realizability. If it can not been executed then to isolate possible auxiliary or partially (one-route) queries, eg. such queries whose results are input data to a new analytical query to further analysis is submitted. Based on this isolated auxiliary or partially queries, a decision about incrementally and dynamically data warehouse schema extension is taking with the aid of proposed method. If it can not been executed then to isolate possible auxiliary or partially (one-route) queries, eg. such queries whose results are input data to a new analytical query to further analysis is submitted. Based on this isolated auxiliary or partially queries, a decision about incrementally and dynamically data warehouse schema extension is taking with the aid of proposed method. In proposed approach to data warehouse design problem, the Multidimensional Aggregation Cube data model and associated with it terms and concepts has been selected in order to accomplish analytical queries influence to the shape of dynamically extended warehouse data schema. In particular, based on introduced by MAC model author’s analysis paths conception, a formal multidimensional schema analysis paths model was proposed, which was base to construction correct – from the legacy OLTP systems viewpoint - analytical queries.