

IBM DB2 Cube Views



Przykłady tworzenia modeli biznesowych

Wersja 8

Uwaga

Uwaga: Przed korzystaniem z tych informacji oraz z produktu, którego one dotyczą, należy zapoznać się z treścią sekcji “Uwagi” na stronie 29.

Dokument ten zawiera informacje dotyczące praw własności IBM . Informacje te są dostarczane na warunkach umowy licencyjnej i są chronione prawem autorskim. Informacje zawarte w niniejszej publikacji nie dotyczą jakichkolwiek gwarancji udzielanych na produkty, a jakiegokolwiek oświadczenia znajdujące się w tym podręczniku nie mogą być interpretowane jako gwarancje.

Publikacje firmy IBM można zamawiać przez Internet lub za pośrednictwem lokalnego przedstawiciela IBM.

- Aby zamówić publikacje przez Internet, należy przejść do strony IBM Publications Center pod adresem www.ibm.com/shop/publications/order
- Aby odnaleźć lokalnego przedstawiciela firmy IBM, należy przejść do strony IBM Directory of Worldwide Contacts pod adresem www.ibm.com/planetwide

Aby zamówić publikacje związane z programem DB2 w dziale sprzedaży i marketingu DB2 w Stanach Zjednoczonych lub Kanadzie, należy zadzwonić pod numer 1800-IBM-4YOU (425-4968).

Przesyłając informację do IBM, Użytkownik udziela IBM niewyłącznego prawa do używania i rozpowszechniania tych informacji w dowolny sposób, jaki uzna za właściwy, bez żadnych zobowiązań wobec ich autora.

Spis treści

Informacje o podręczniku	v	Szczegóły scenariusza	17
Kto powinien przeczytać ten podręcznik	v	Czynności wymagane do utworzenia miary	18
Rozdział 1. Obliczanie ruchu i wartości zapasów w magazynie w czasie	1	Rozdział 5. Ranking wysokości obrotów	21
Szczegóły scenariusza	1	Szczegóły scenariusza	21
Czynności wymagane do utworzenia miar	3	Czynności wymagane do utworzenia miary	23
Rozdział 2. Kojarzenie kosztów reklamy z wysokością sprzedaży	7	Rozdział 6. Użycie danych czasowych w tabeli faktów do utworzenia wymiaru czasu	25
Szczegóły scenariusza	7	Szczegóły scenariusza	25
Czynności wymagane do utworzenia miary	7	Czynności wymagane przy tworzeniu atrybutów i wymiaru	27
Rozdział 3. Obliczanie zysku i marży sklepu	13	Uwagi	29
Szczegóły scenariusza	13	Znaki towarowe	31
Czynności wymagane do utworzenia miar	13		
Rozdział 4. Zliczanie zamówień internetowych	17		

Informacje o podręczniku

Ta książka zawiera informacje uzupełniające w stosunku do podręcznika *DB2 Cube Views - Podręcznik instalacji i użytkowania*. Zawarto w niej zbiór przykładowych modeli praktycznego wykorzystania metadanych programu DB2 Cube Views w zastosowaniach biznesowych.

Kto powinien przeczytać ten podręcznik

Podręcznik ten powinni przeczytać administratorzy baz danych posługujący się metadanymi OLAP i programem DB2 Universal Database (DB2). Konieczna jest znajomość następujących zagadnień:

- zagadnienia związane z techniką OLAP, takie jak schematy gwiazdziste;
- obiekty metadanych programu DB2 Cube Views, takie jak modele kostki, obiekty faktów, wymiary, łączenia, miary i atrybuty.

Rozdział 1. Obliczanie ruchu i wartości zapasów w magazynie w czasie

Przedsiębiorstwo zajmujące się handlem detalicznym, XYZ Retail, korzysta z magazynu do przechowywania zapasów przed wysłaniem ich do sklepu w celu sprzedania. W bazach danych XYZ Retail rejestrowane są na bieżąco dane na temat stanu zapasów w magazynie. Teraz zachodzi potrzeba poddania tych danych analizie. Dla kierownictwa szczególnie interesujące są dwa aspekty funkcjonowania magazynu:

- ruch towarów do i z magazynu,
- wartość towarów znajdujących się w magazynie w danym momencie.

Pierwszy aspekt, czyli ruch towarów, wymaga rozpatrzenia dostępnych danych w wymiarze czasu. Drugi aspekt, czyli wartość towarów, jest statycznym obrazem stanu magazynu w wybranym momencie.

Szczegóły scenariusza

W bazach danych firmy XYZ Retail znajduje się tabela faktów zawierająca następujące kolumny związane z magazynem: QUANTITY_IN (ilość przyjęta), QUANTITY_OUT (ilość wydana), CURRENT_QUANTITY (ilość bieżąca), PRODUCT_VALUE (wartość towaru), PRODUCT_ID (identyfikator towaru) i TIME_ID (czas). Dane są wprowadzane do tabeli w odstępach tygodniowych. W bazie danych znajduje się także tabela Product (Produkt) i Time (Czas). Przykładowy zestaw danych z tabeli faktów przedstawia Tabela 1.

Tabela 1.

PRODUCT_ID	TIME_ID	QUANTITY_ID	QUANTITY_OUT	CURRENT_QUANTITY	PRODUCT_VALUE
1234	1	5	0	5	5
1234	2	20	10	15	5
1234	3	10	20	5	5

Wartość PRODUCT_ID w każdym z trzech przykładowych wpisów jest identyczna, ponieważ produkt tego samego typu może być wielokrotnie przyjmowany i wydawany z magazynu.

Administrator bazy danych XYZ Retail musi utworzyć trzy różne miary:

Flow In (Przyjęcia)

Odwieczidla ruch towarów w kierunku do magazynu.

Flow Out (Wydania)

Odzwierciedla ruch towarów z magazynu.

Current Value (Wartość bieżąca)

Odzwierciedla wartość towarów w magazynie w danym momencie.

Przy tworzeniu pierwszych dwóch miar, Flow In i Flow Out, należy utworzyć odwzorowania odpowiednio do kolumn QUANTITY_IN i QUANTITY_OUT, po czym zsumować dane po wszystkich wymiarach. Jest to przypadek miary w pełni addytywnej, ponieważ agregacja danych odbywa się wyłącznie przy użyciu funkcji SUM we wszystkich wymiarach. Tabela 2 przedstawia przykładowy zestaw danych z kolumn QUANTITY_IN i QUANTITY_OUT za trzy miesiące dla produktu z PRODUCT_ID 1234. Miary Flow In i Flow Out sumują te miesięczne wartości w celu obliczenia ogólnych ilości towaru przyjmowanych i wydawanych z magazynu w skali kwartału.

Tabela 2. Obliczenie przykładowych danych dla miar w pełni addytywnych Flow In i Flow Out dla produktu PRODUCT_ID 1234

	Styczeń	Luty	Marzec	Kwartał 1
QUANTITY_IN	5	20	10	35
QUANTITY_OUT	0	10	20	30

Miary w pełni addytywne to nierzadko najprostsze i najczęściej tworzone miary. Często służą one jako elementy składowe bardziej złożonych miar. W przypadku miar odwołujących się do danych liczbowych program Centrum OLAP domyślnie tworzy miary w pełni addytywne.

Aby utworzyć trzecią miarę, Current Value, należy obliczyć wynik mnożenia kolumn PRODUCT_VALUE i CURRENT_QUANTITY. Na przykład, jeśli dla PRODUCT_ID=1234 wartość jednostkowa wynosi 5, wartość miary Current Value odpowiadająca danym przykładowym przedstawia Tabela 3.

Tabela 3. Obliczanie przykładowych wartości miary Current Value dla danych przykładowych i PRODUCT_ID 1234

	Styczeń	Luty	Marzec
CURRENT_QUANTITY	5	10	20
Current Value	25	50	100

W następnym kroku dane te należy zagregować wzdłuż wymiarów. Ponieważ jednak miara ta oblicza wartość z określonej chwili, sumowanie jej w wymiarze czasu nie miałyby sensu. Dlatego agregacja będzie sumować dane po wymiarze Product i obliczać średnią z wyników w czasie. Taką miarę nazywa się póładdytywną, ponieważ tylko część agregacji realizowana jest z użyciem funkcji SUM.

Miary obliczające dane chwilowe, czyli reprezentujące określony moment w czasie - na przykład wynik comiesięcznej inwentaryzacji - często są miarami póładdytywnymi, ponieważ sumowanie miesięcy w kwartały nie daje sensownych wyników. Jeśli dany towar pozostaje w magazynie przez cały miesiąc, jest on uwzględniany w wartości CURRENT_QUANTITY za kolejne trzy miesiące oddzielnie. Gdyby wartości CURRENT_QUANTITY zostały zsumowane w czasie, towar przechowywany w magazynie przez trzy miesiące byłby liczony trzykrotnie. Jak przedstawia to Tabela 4, wartość 25 za kwartał 1 nie jest użyteczna z punktu widzenia obsługi magazynu. Jak wynika z tabeli, w magazynie nigdy nie znajdowało się 25 sztuk produktu, toteż wynik w tej postaci jest pozbawiony sensu.

Tabela 4. Obliczanie przykładowych danych dla kolumny CURRENT_QUANTITY za pomocą funkcji SUM dla wymiaru czasu i dla PRODUCT_ID 1234

	Styczeń	Luty	Marzec	Kwartał 1
SUM(CURRENT_QUANTITY)	5	15	5	25

Zamiast korzystać z funkcji SUM we wszystkich wymiarach, można użyć innych funkcji agregujących w wymiarze czasu, takich jak AVG, MIN i MAX. Na przykład przy identycznym zestawie danych przykładowych za styczeń, luty i marzec można użyć drugiej funkcji agregującej w wymiarze czasu, jak to przedstawia Tabela 5. Wynikiem będą sensowne wartości dla kwartału. Miara Current Value może reprezentować średnią sumaryczną wartość towaru w magazynie za kwartał, ewentualnie minimalną lub maksymalną wartość w dowolnym momencie w ciągu kwartału.

Tabela 5. Obliczanie przykładowych danych dla kolumny CURRENT_QUANTITY za pomocą funkcji AVG, MAX i MIN dla wymiaru czasu i dla PRODUCT_ID 1234

	Styczeń	Luty	Marzec	Kwartał 1
AVG(CURRENT_QUANTITY)	5	15	5	8,3
MAX(CURRENT_QUANTITY)	5	15	5	15
MIN(CURRENT_QUANTITY)	5	15	5	5

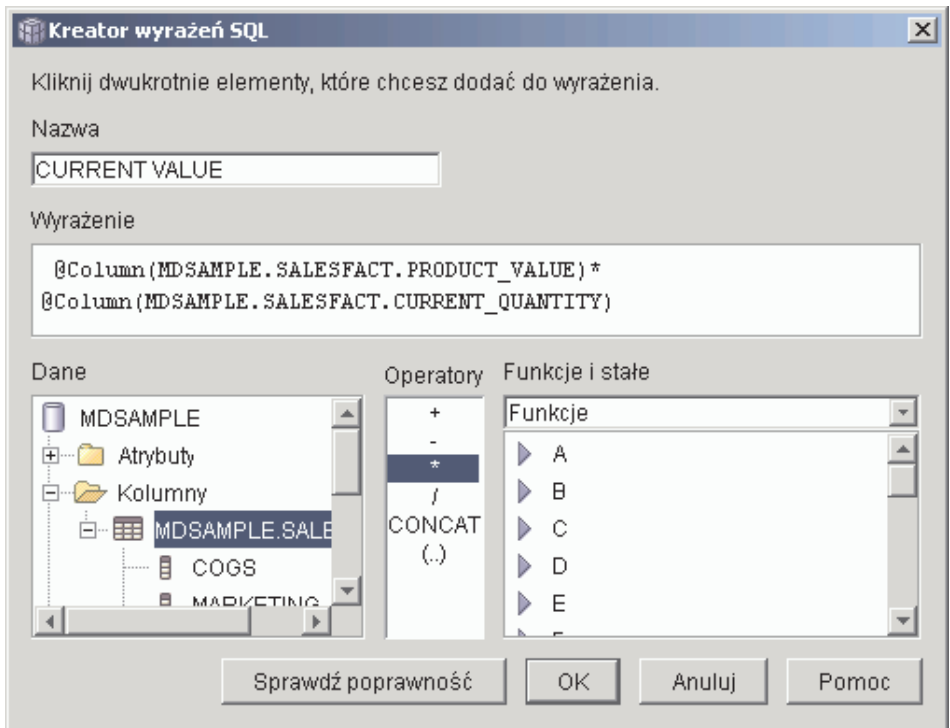
Czynności wymagane do utworzenia miar

W poniższej procedurze opisano sposób użycia okna Właściwości faktów w Centrum OLAP do utworzenia miar Flow In, Flow Out i Current Value w istniejącym obiekcie faktów:


1. Aby otworzyć okno Właściwości faktów, kliknij prawym przyciskiem myszy obiekt faktów w drzewie obiektów Centrum OLAP, a następnie kliknij opcję **Edytuj miary**. Otwarte zostanie okno Właściwości faktów.

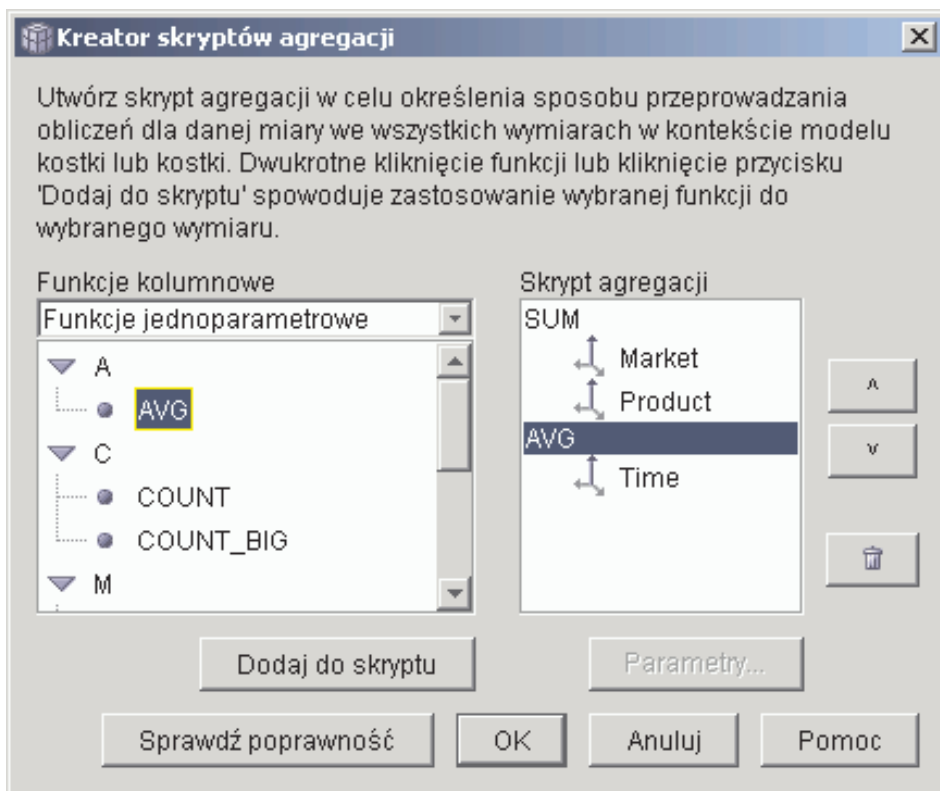
2. Utwórz miarę Flow In:
 - a. Na stronie Miary kliknij opcję **Utwórz miarę obliczaną**, aby utworzyć miarę Flow In. Otwarte zostanie okno Kreatora wyrażeń SQL.
 - b. W oknie Kreatora wyrażeń SQL w polu **Nazwa** wpisz FLOW IN.
 - c. Aby utworzyć wyrażenie obliczające przyjęcia, wykonaj poniższe czynności:
 - Rozwiń folder **Kolumny** i tabelę faktów na liście Dane.
 - Kliknij dwukrotnie kolumnę **QUANTITY_IN**, aby dodać ją do wyrażenia.
 - Kliknij przycisk **OK**, aby zamknąć okno Kreatora wyrażeń SQL. Na stronie Agregacje nie trzeba zmieniać domyślnej funkcji agregującej, którą jest SUM. Funkcja SUM jest domyślnym wyborem dla miary Flow In, ponieważ dane źródłowe są liczbowe i ponieważ miara odwołuje się do kolumny, a nie do istniejących miar.
3. Utwórz miarę Flow Out:
 - a. Na stronie Miary kliknij opcję **Utwórz miarę obliczaną**, aby utworzyć miarę Flow Out. Otwarte zostanie okno Kreatora wyrażeń SQL.
 - b. W oknie Kreatora wyrażeń SQL w polu **Nazwa** wpisz FLOW OUT.
 - c. Aby utworzyć wyrażenie obliczające wydania, wykonaj poniższe czynności:
 - Rozwiń folder **Kolumny** i tabelę faktów na liście **Dane**.
 - Kliknij dwukrotnie kolumnę **QUANTITY_OUT**.
 - d. Kliknij przycisk **OK**, aby zamknąć okno Kreatora wyrażeń SQL. Na stronie Agregacje nie trzeba zmieniać domyślnej funkcji agregującej, którą jest SUM. Funkcja SUM jest domyślnym wyborem dla miary Flow Out, ponieważ dane źródłowe są liczbowe, zaś miara odwołuje się do kolumny, a nie tylko do istniejących miar.
4. Utwórz miarę Current Value:
 - a. Na stronie Miary kliknij opcję **Utwórz miarę obliczaną**, aby utworzyć miarę Current Value. Otwarte zostanie okno Kreatora wyrażeń SQL.
 - b. W oknie Kreatora wyrażeń SQL w polu **Nazwa** wpisz CURRENT VALUE.
 - c. Aby utworzyć wyrażenie miary Current Value, wykonaj poniższe czynności:
 - Rozwiń folder **Kolumny** i tabelę faktów na liście **Dane**.
 - Kliknij kolumnę **PRODUCT_VALUE** na liście **Dane**.
 - Kliknij dwukrotnie operator ***** na liście **Operatory**.
 - Kliknij kolumnę **CURRENT_QUANTITY** na liście **Dane**.

Rys. 1 na stronie 5 przedstawia wyrażenie obliczające wartość bieżącą utworzone w oknie Kreatora wyrażeń SQL.



Rysunek 1. Gotowe wyrażenie miary Current Value w oknie Kreatora wyrażen SQL

- d. Kliknij przycisk **OK**, aby zamknąć okno Kreatora wyrażen SQL.
- e. Na karcie Agregacje kliknij agregację dla miary Current Value i kliknij na liście pozycję **Skrypt agregacji**. Otwarte zostanie okno Kreatora skryptów agregacji. Domyślny skrypt agregacji będzie zawierał funkcję SUM dla wszystkich wymiarów.
- f. W razie potrzeby przenieś wymiar Time niżej, wybierając pozycję **Time** i klikając przycisk , przez co wymiar trafi na ostatnią pozycję na liście wymiarów w skrypcie. Następnie, kiedy wymiar Time będzie wybrany, kliknij dwukrotnie funkcję **AVG** na liście **Funkcje kolumnowe**. Skrypt agregacji, który przedstawia Rys. 2 na stronie 6, sumuje dane po wszystkich wymiarach z wyjątkiem Time, w którym to wymiarze obliczana jest średnia danych.



Rysunek 2. Skrypt agregacji dla miary Current Value

- g. W oknie Kreatora skryptów agregacji kliknij przycisk **Sprawdź poprawność**, aby sprawdzić, czy skrypt agregacji jest poprawny. Kliknij przycisk **OK**, aby zapisać skrypt agregacji i zamknąć okno.
5. Kliknij przycisk **OK**, aby zapisać zmiany w obiekcie faktów i zamknąć okno Właściwości faktów.

W wyniku wykonanej procedury otrzymaliśmy trzy miary obliczane związane z zapasami w magazynie. Miar tych można używać do analizowania prawidłowości w ruchu towarów do i z magazynu.

Rozdział 2. Kojarzenie kosztów reklamy z wysokością sprzedaży

Pewien dealer samochodowy zastanawia się nad zwiększeniem nakładów na reklamę. Aby podjąć trafną decyzję, kierownictwo zamierza najpierw przeanalizować dane historyczne dotyczące związków między wydatkami na reklamę a wysokością sprzedaży. Należy przy tym ustalić, czy zmiana wydatków na reklamę miała jakikolwiek wpływ na sprzedaż, a w szczególności, czy nasilenie reklamy daje się ściśle skorelować ze zwiększeniem sprzedaży.

Szczegóły scenariusza

W bazie danych używanej przez dealera istnieje tabela faktów z kolumnami Sales (Sprzedaż) i Ad Costs (Koszty reklamy). Baza danych zawiera także kilka tabel wymiarów. Administrator może utworzyć miarę korzystając z funkcji DB2 CORRELATION do przeprowadzenia obliczeń badających poziom korelacji między wysokością kosztów i sprzedażą. Funkcja CORRELATION wymaga podania dwóch parametrów wejściowych. W tym przypadku jako parametrów administrator użył kolumn Sales i Ad Costs.

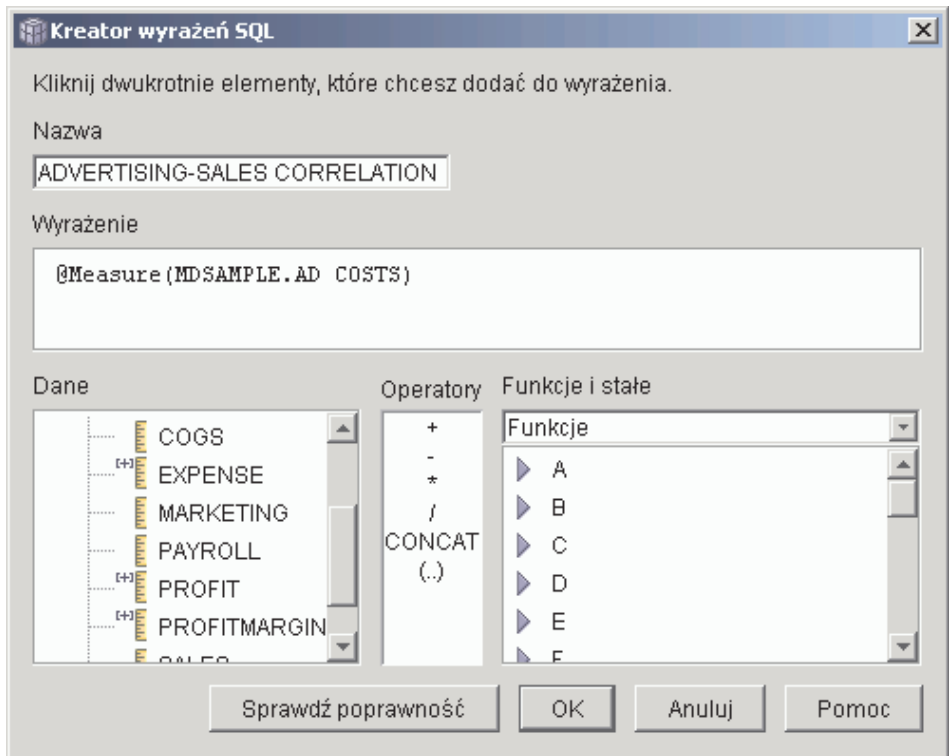
Wieloparametrowa funkcja agregująca musi wystąpić w skrypcie agregacji jako pierwsza. Funkcję wieloparametrową można zastosować po wszystkich wymiarach lub najpierw dla wszystkich wymiarów oprócz wymiaru czasu, po czym wymiar czasu zagregować inną funkcją, na przykład MAX. W wyrażeniu SQL odpowiedniej miary będzie istniało bezpośrednie odwzorowanie do kolumny Ad Costs. Wyrażenie SQL będzie pierwszym z dwóch parametrów funkcji wieloparametrowej. Administrator zdefiniuje drugi parametr jako wyrażenie SQL odwzorowane bezpośrednio do kolumny Sales. Funkcja CORRELATION jest zdefiniowana jako jedyna funkcja agregacji, przez co miara może obliczyć korelację statystyczną między kosztami reklamy a wynikami sprzedaży we wszystkich wymiarach.

Czynności wymagane do utworzenia miary

W poniższej procedurze opisano sposób użycia okna Właściwości faktów w Centrum OLAP do utworzenia miary Advertising-Sales Correlation (Korelacja reklamy ze sprzedażą) w istniejącym obiekcie faktów:

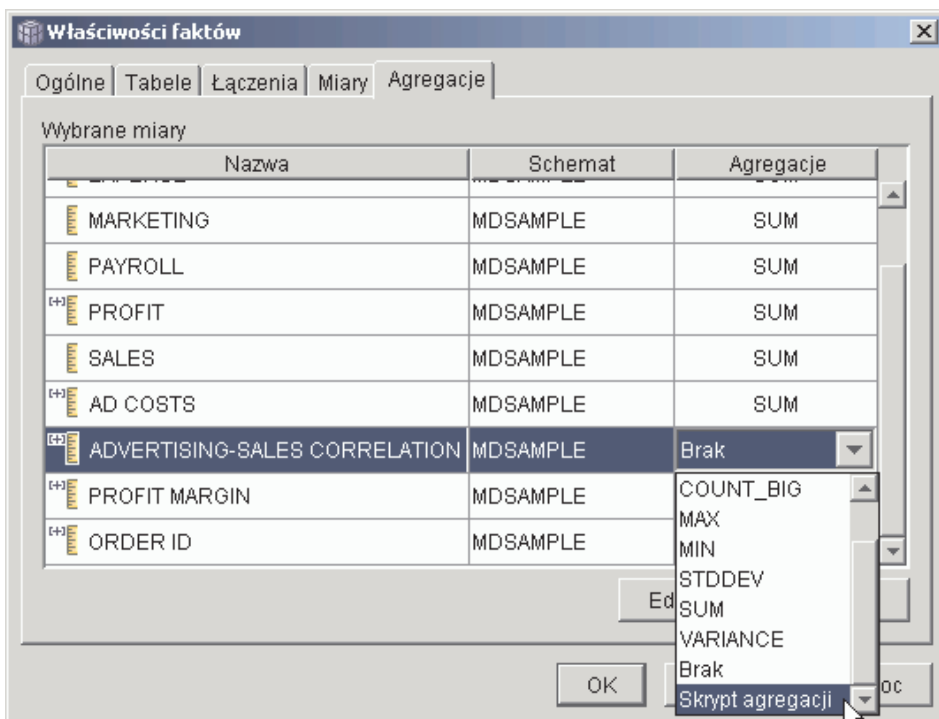
1. Otwórz okno Właściwości faktów, klikając prawym przyciskiem myszy obiekt faktów w drzewie obiektów Centrum OLAP, a następnie kliknij opcję **Edytuj miary**.
2. Kliknij przycisk **Utwórz miarę obliczaną**. Otwarte zostanie okno Kreatora wyrażen SQL.

3. W oknie Kreatora wyrażeń SQL, w polu **Nazwa**, wpisz ADVERTISING-SALES CORRELATION.
4. Zdefiniuj wyrażenie miary, które posłuży także jako pierwszy parametr funkcji wieloparametrowej CORRELATION w skrypcie agregacji. Aby zdefiniować wyrażenie, rozwiń folder **Miary** na liście **Dane** i kliknij dwukrotnie miarę **AD COSTS** w celu dodania jej do listy **Wyrażenie**. Rys. 3 przedstawia wyrażenie utworzone w oknie Kreatora wyrażeń SQL.



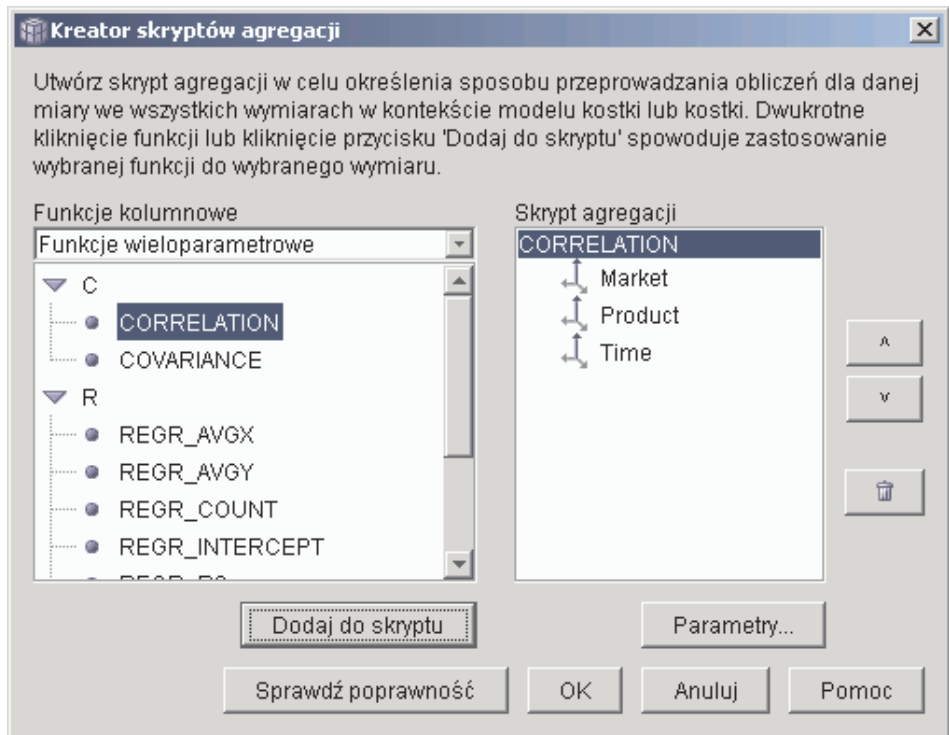
Rysunek 3. Gotowe wyrażenie korelujące koszty reklam ze sprzedażą w oknie Kreatora wyrażeń SQL

5. Na stronie Agregacji kliknij funkcję agregacji dla miary **ADVERTISING-SALES CORRELATION** i wybierz opcję **Skrypt agregacji**, jak to przedstawia Rys. 4 na stronie 9. Otwarte zostanie okno Kreatora skryptów agregacji.



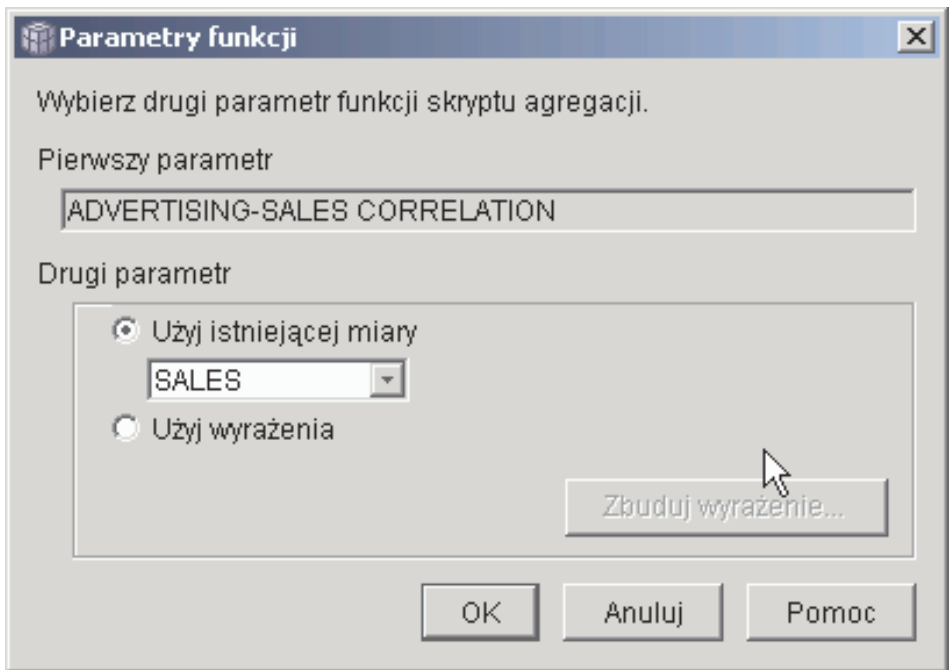
Rysunek 4. Strona Agregacje w oknie Właściwości faktów

6. W polu **Funkcje kolumnowe** wybierz opcję **Funkcje wieloparametrowe**. Na liście funkcji wieloparametrowych wybierz funkcję **CORRELATION** i kliknij przycisk **Dodaj do skryptu**. Rys. 5 na stronie 10 przedstawia funkcję CORRELATION na pierwszym miejscu listy wymiarów w skrypcie.



Rysunek 5. Skrypt agregacji dla miary Advertising-Sales Correlation

7. Kliknij przycisk **Parametry**, aby określić drugi parametr funkcji CORRELATION. Otwarte zostanie okno Parametry funkcji. Wybierz opcję **Użyj istniejącej miary** i wybierz pozycję **SALES**. Rys. 6 na stronie 11 przedstawia okno Parametry funkcji z miarą Sales wybraną jako drugi parametr. Kliknij przycisk **OK**, aby zapisać wybrane opcje i zamknąć okno Parametry funkcji.



Rysunek 6. Miara Sales wybrana jako drugi parametr w oknie Parametry funkcji

8. W oknie Kreatora skryptów agregacji kliknij przycisk **Sprawdź poprawność**, aby sprawdzić, czy skrypt agregacji jest poprawny. Kliknij przycisk **OK**, aby zapisać skrypt agregacji i zamknąć okno.
9. Kliknij przycisk **OK**, aby zapisać zmiany w obiekcie faktów i zamknąć okno Właściwości faktów.

Utworzona została miara korelująca dwa rodzaje informacji w bazie danych. Na jej podstawie łatwiej będzie podejmować decyzje o wydatkach na reklamę w przyszłości, analizując tendencje zmian dotychczasowych.

Rozdział 3. Obliczanie zysku i marży sklepu

Kierownik sklepu z zabawkami chce mieć możliwość analizowania wpływu różnych czynników, takich jak pora roku i typ towaru, na zysk i marżę. Zanim można będzie przystąpić do zaawansowanej analizy, administrator bazy danych musi utworzyć miary zysku i marży. W następnej kolejności administrator utworzy dodatkowe miary, które pozwolą skorelować rozmaite czynniki z zyskiem i marżą oraz porównywać je ze sobą.

Szczegóły scenariusza

Baza danych sklepu zawiera tabelę faktów, w której znajdują się kolumny Sales (Sprzedaż), Costs of Goods Sold (COGS - Koszt sprzedanych towarów) i Expense (Wydatki), a oprócz tego kolumny klucza obcego dla każdej z siedmiu tabel wymiarów. Administrator utworzył już miary o nazwach Sales, COGS i Expense, które są odwzorowane odpowiednio do kolumn Sales, COGS i Expense. Miary Profit (Zysk) i Profit Margin (Marża) można zdefiniować wyłącznie w oparciu o te istniejące już miary.

Aby uzyskać miarę Profit, administrator w oknie Wyrażenie SQL tworzy miarę obliczającą wielkość $SALES-(COGS+EXPENSE)$, po czym sumuje wyniki obliczeń po wszystkich wymiarach. Miarę Profit można utworzyć, odwołując się do istniejących miar, kolumn lub ich kombinacji.

Po utworzeniu miary Profit administrator może utworzyć miarę Profit Margin. Miara Profit Margin stanowi procentowy stosunek dwóch istniejących miar, $(Profit / Sales)*100$, i nie wymaga własnej funkcji agregującej. Funkcja agregująca nie jest wymagana, ponieważ miara odwołuje się tylko do innych miar, których dane są już zagregowane. Gdy administrator korzysta z miary złożonej, czyli takiej, która odwołuje się tylko do innych miar, obliczając ich stosunek, nie ma potrzeby definiowania dodatkowej agregacji. Większość funkcji agregujących, takich jak SUM, nie daje sensownych wyników w odniesieniu do wartości będących ilorazami. Na przykład jeśli sklep z zabawkami osiągał w czterech kolejnych kwartałach marżę na poziomie 40%, 32%, 28% i 37%, sumowanie tych wartości w wymiarze czasu dałoby wynik 137% w skali roku, czyli wartość pozbawioną sensu.

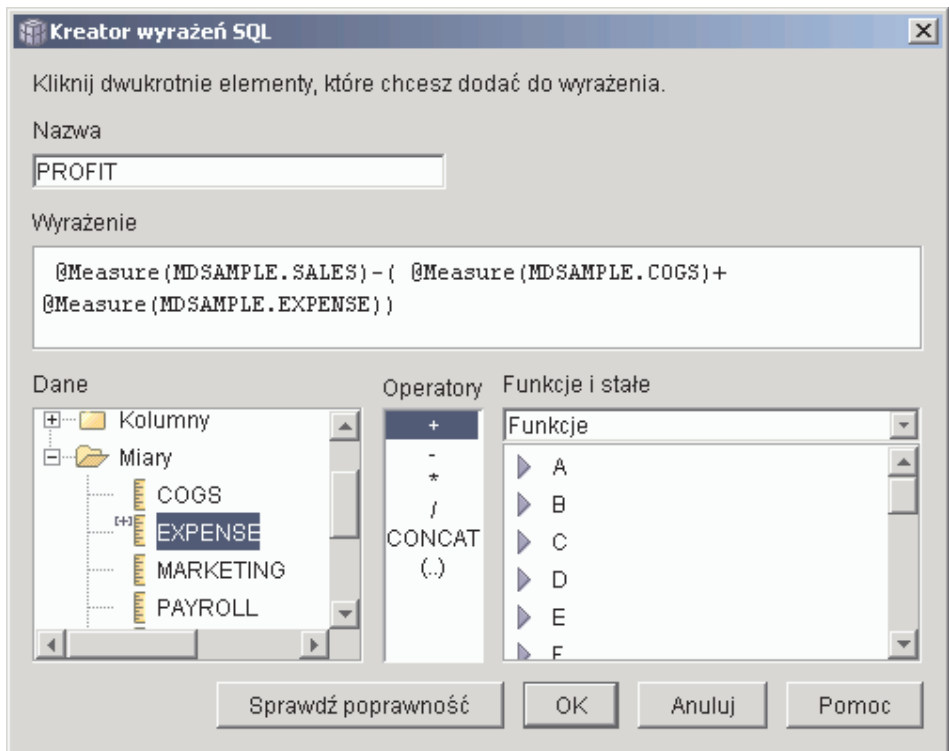
Czynności wymagane do utworzenia miar

W poniższej procedurze opisano sposób użycia okna Właściwości faktów w Centrum OLAP do utworzenia miar Profit i Profit Margin w istniejącym obiekcie faktów:

1. Aby otworzyć okno Właściwości faktów, kliknij prawym przyciskiem myszy obiekt faktów w drzewie obiektów Centrum OLAP, a następnie kliknij opcję **Edytuj miary**. Otwarte zostanie okno Właściwości faktów.
2. Utwórz miarę Profit:

- a. Na stronie Miary kliknij przycisk **Utwórz miarę obliczaną**. Otwarte zostanie okno Kreatora wyrażeń SQL.
- b. W oknie Kreatora wyrażeń SQL, w polu **Nazwa**, wpisz PROFIT.
- c. Aby zdefiniować wyrażenie obliczające zysk, rozwiń folder **Miary** na liście **Dane** i wykonaj poniższe czynności:
 - Kliknij dwukrotnie miarę **SALES** na liście **Dane**, aby dodać ją do wyrażenia.
 - Kliknij dwukrotnie operator **-** na liście **Operatory**.
 - Kliknij dwukrotnie miarę **COGS** na liście **Dane**.
 - Kliknij dwukrotnie operator **+** na liście **Operatory**.
 - Kliknij dwukrotnie miarę **EXPENSE** na liście **Dane**.
 - W polu **Wyrażenie** podświetl następujący fragment wyrażenia:
 @Measure(MDSAMPLE.COGS)+@Measure(MDSAMPLE.EXPENSE)
 i kliknij dwukrotnie operator **(.)** na liście **Operatory**, aby ująć zaznaczoną część wyrażenia w nawiasy.

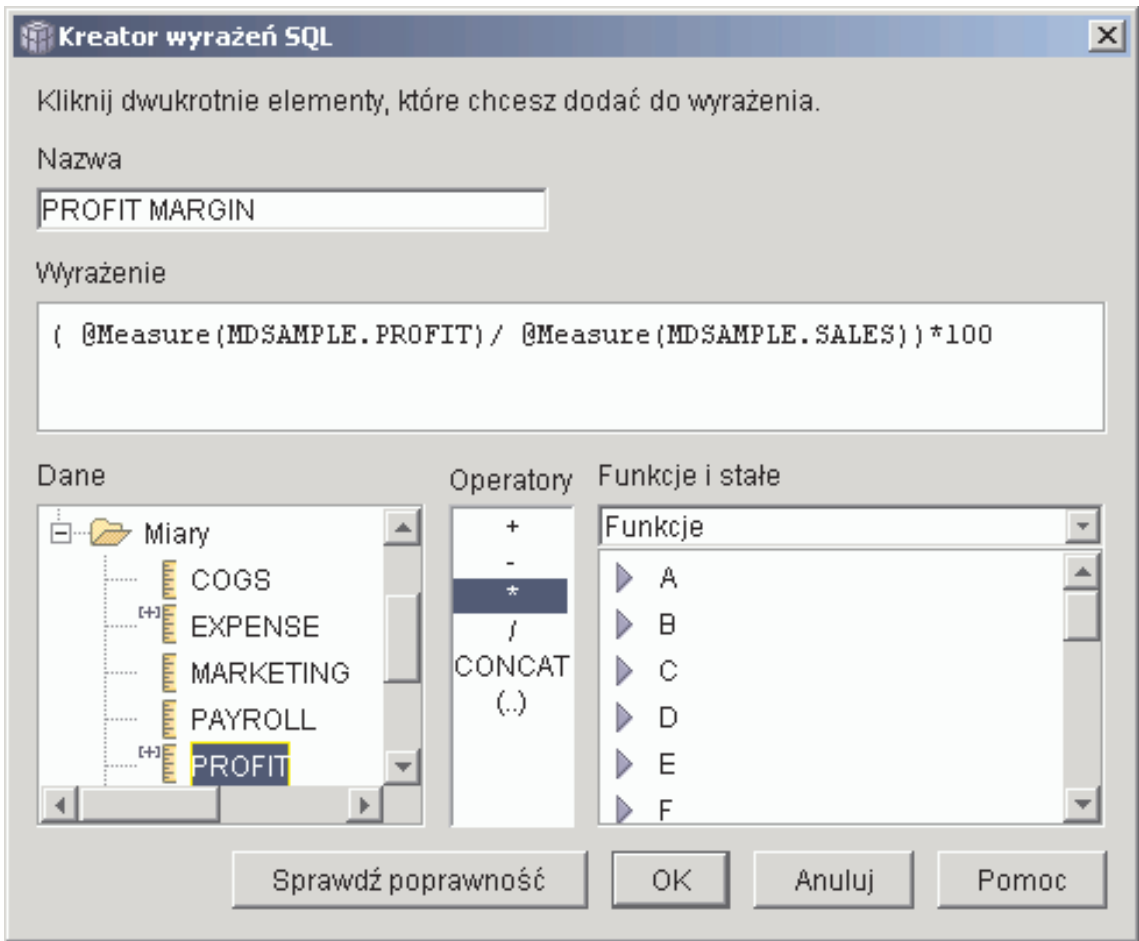
Rys. 7 przedstawia wyrażenie obliczające zysk utworzone w oknie Kreatora wyrażeń SQL.



Rysunek 7. Gotowe wyrażenie obliczające zysk w oknie Kreatora wyrażeń SQL

- d. Kliknij przycisk **OK**, aby utworzyć miarę Profit i zamknąć okno Kreatora wyrażeń SQL.
 - e. Na stronie Agregacje kliknij agregację dla miary Profit i wybierz funkcję SUM. Miara Profit jest gotowa.
3. Utwórz miarę Profit Margin:
- a. Na stronie Miary kliknij opcję **Utwórz miarę obliczaną**. Otwarte zostanie okno Kreatora wyrażeń SQL.
 - b. W oknie Kreatora wyrażeń SQL, w polu **Nazwa**, wpisz PROFIT MARGIN.
 - c. Aby zdefiniować wyrażenie obliczające marżę, rozwiń folder **Miary** na liście **Dane** i wykonaj poniższe czynności:
 - Kliknij dwukrotnie miarę **PROFIT** na liście **Dane**, aby dodać ją do wyrażenia.
 - Kliknij dwukrotnie operator/ na liście **Operatory**.
 - Kliknij dwukrotnie miarę **SALES** na liście **Dane**.
 - Zamknij całe wyrażenie w nawiasach, wpisując je w polu **Wyrażenie** field.
 - Ustaw kursor na końcu wyrażenia i kliknij dwukrotnie operator * na liście **Operatory**.
 - Dopisz liczbę 100 na końcu wyrażenia w polu **Wyrażenie**.

Rys. 8 na stronie 16 przedstawia wyrażenie obliczające marżę utworzone w oknie Kreatora wyrażeń SQL.



Rysunek 8. Gotowe wyrażenie obliczające marżę w oknie Kreatora wyrażen SQL

- d. Kliknij przycisk **OK**, aby utworzyć miarę Profit Margin i zamknąć okno Kreatora wyrażen SQL.

Na stronie Agregacje Centrum OLAP dobiera domyślnie ustawienie BRAK w polu agregacji dla miar złożonych, dlatego nie ma potrzeby zmiany funkcji agregującej.

4. Kliknij przycisk **OK**, aby zamknąć okno Właściwości faktów i zapisać obie nowe miary dodane do obiektu faktów.

Gdy obie te miary są już gotowe, można na ich podstawie wykonywać dodatkowe analizy.

Rozdział 4. Zliczanie zamówień internetowych

Pewne przedsiębiorstwo zajmujące się handlem detalicznym kilka lat temu poszerzyło zakres działalności o sprzedaż przez Internet. Obecnie jego kierownictwo pragnie przeanalizować efektywność internetowego kanału sprzedaży. Jednym z pierwszych wskaźników, jakie są potrzebne w takiej analizie, jest liczba zamówień zrealizowanych za pośrednictwem Internetu.

Szczegóły scenariusza

W bazie danych przedsiębiorstwa znajduje się tabela faktów dla zamówień internetowych, zawierająca kolumny ORDER_ID, PRODUCT_ID, QUANTITY i TIME_ID. W kolumnie PRODUCT_ID zapisywany jest każdy produkt sprzedany w ramach danego zamówienia, a w kolumnie QUANTITY figurują odpowiednie ilości sprzedanych produktów. Zamówienia zawierające więcej niż jeden produkt zapisywane są w wielu wierszach, których liczba odpowiada liczbie odrębnych produktów w zamówieniu. Na przykład Tabela 6 zawiera informacje o trzech zamówieniach, z których Zamówienie 1 opiewało na trzy produkty A, jeden produkt O i jeden produkt G.

Tabela 6. Zawartość części tabeli faktów

ORDER_ID	PRODUCT_ID	QUANTITY
1	A	3
1	O	1
1	G	1
2	L	1
2	Q	2
3	P	5

Administrator bazy danych może utworzyć miarę Order Count (Liczba zamówień), która będzie zliczać każdą niepowtarzalną wartość w kolumnie ORDER_ID. W kodzie SQL definiującym miarę Order Count musi wystąpić parametr DISTINCT oraz funkcja COUNT realizująca agregację po wszystkich wymiarach. Wyrażenie SQL tej miary utworzy listę odrębnych zamówień, uzyskaną przez zliczanie w wyniku agregacji. Jako że miara nie realizuje sumowania, nazywana jest miarą nieaddytywną.

Miary nieaddytywne są użyteczne także w sytuacji, gdy zachodzi potrzeba zliczania danych tekstowych, znakowych lub innego typu. Przykładem będzie użycie miary nieaddytywnej do policzenia kodów pocztowych, pod które realizowane były dostawy.

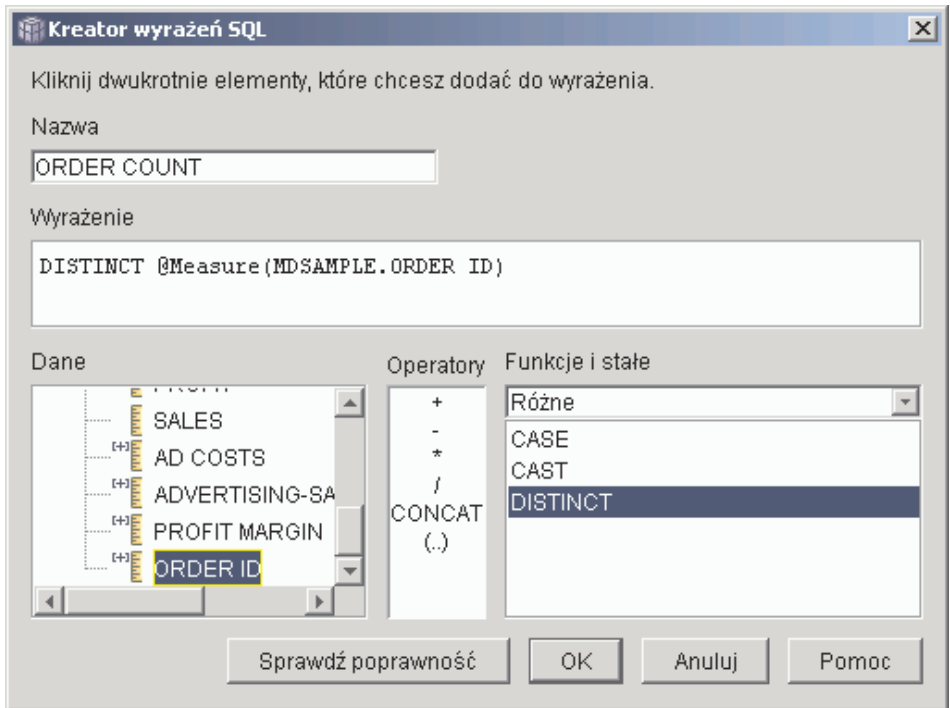
W tym przykładzie administrator postanowił zdefiniować miarę Order ID (ID zamówienia), bezpośrednio odwzorowaną do kolumny ORDER_ID. Można jednak użyć samej kolumny ORDER_ID w identyczny sposób. Domyślna agregacja jest dobierana w zależności od tego, czy dana kolumna lub miara jest wykorzystywana w wyrażeniu SQL, jednak w obu przypadkach należy zmienić domyślną agregację na COUNT, jak to opisano w punkcie “Czynności wymagane do utworzenia miary”.

Czynności wymagane do utworzenia miary

W poniższej procedurze opisano sposób użycia okna Właściwości faktów w Centrum OLAP do utworzenia miary Order Count w istniejącym obiekcie faktów:

1. Aby otworzyć okno Właściwości faktów, kliknij prawym przyciskiem myszy obiekt faktów w drzewie obiektów Centrum OLAP, a następnie kliknij opcję **Edytuj miary**. Otwarte zostanie okno Właściwości faktów.
2. Na stronie Miary kliknij przycisk **Utwórz miarę obliczaną**. Otwarte zostanie okno Kreatora wyrażen SQL.
3. W oknie Kreatora wyrażen SQL, w polu **Nazwa**, wpisz ORDER COUNT.
4. Aby zdefiniować wyrażenie zliczające zamówienia, rozwiń folder **Miary** na liście **Dane** i wykonaj poniższe czynności:
 - W polu **Funkcje i stałe** wybierz pozycję **Różne**. Na liście różnych funkcji i stałych kliknij dwukrotnie parametr **DISTINCT**.
 - Kliknij dwukrotnie miarę **ORDER ID** na liście **Dane**.

Rys. 9 na stronie 19 przedstawia wyrażenie zliczające zamówienia utworzone w oknie Kreatora wyrażen SQL.



Rysunek 9. Kompletnie wyrażenie zliczające zamówienia w oknie Kreatora wyrażen SQL

5. Kliknij przycisk **OK**, aby zamknąć okno Kreatora wyrażen SQL.
6. Na stronie Agregacje kliknij agregację dla miary Order Count i wybierz funkcję **COUNT**.
7. Kliknij przycisk **OK**, aby zapisać zmiany w obiekcie faktów i zamknąć okno Właściwości faktów.

Wynikiem powyższych operacji jest miara zliczająca liczbę odrębnych wartości w wierszu Order ID (Numer zamówienia). Miary tej można używać w połączeniu z innymi miarami przy prowadzeniu dalszych analiz danych.

Rozdział 5. Ranking wysokości obrotów

Pewne przedsiębiorstwo w ciągu ostatnich kilku lat intensywnie rozbudowywało swoją sieć sklepów z artykułami biurowymi. Obecnie kierownictwo rozważa zamknięcie najmniej aktywnych sklepów, aby obniżyć koszty i zwiększyć zyski. Historia realizowanych obrotów jest ważnym czynnikiem przy podejmowaniu decyzji o zamknięciu danego sklepu. Analitycy muszą mieć możliwość uszeregowania placówek pod względem wysokości obrotów i drążenia danych w celu dokonania dokładniejszej analizy.

Szczegóły scenariusza

W bazie danych sklepów biurowych znajduje się tabela faktów zawierająca między innymi kolumnę Sales (Obroty). Baza danych zawiera także kilka tabel wymiarów. Administrator może utworzyć miarę Sales Rank (Ranking obrotów) na bazie funkcji RANK. Jest to specjalna funkcja OLAP dostarczana z programem DB2 Universal Database (DB2 UDB).

Program DB2 Cube Views obsługuje następujące funkcje OLAP dostarczane z programem DB2 UDB:

RANK

Porządkuje wiersze i każdemu z nich przypisuje ocenę. Ocena jest definiowana jako 1 plus liczba poprzedzających wierszy o odrębnych pozycjach. Jeśli nie da się ustalić wzajemnej kolejności niektórych wierszy ze względu na występowanie w nich powielonych wartości, wierszom tym przypisywana jest jednakowa wartość oceny. W razie powielenia wartości w wierszach wartości oceny mogą nie tworzyć ciągłej sekwencji. Tabela 7 na stronie 23 przedstawia przykład wyników działania funkcji RANK w odniesieniu do zbioru przykładowych wartości.

Typowa składnia funkcji RANK wygląda następująco:

```
RANK ( ) OVER (ORDER BY  
wyrażenie klucza sortowania kolejność-wyrażenia)
```

gdzie *wyrażenie-klucza-sortowania* jest zbiorem danych poddawanych rankingowi, a *kolejność-wyrażenia* jest to słowo kluczowe **ASC** lub **DESC**, powodujące uporządkowanie wartości wyrażenia klucza sortowania w porządku rosnącym lub malejącym. W programie DB2 Cube Views istnieje wymóg, by *wyrażenie-klucza-sortowania* było istniejącą miarą, a nie kolumną ani atrybutem. Ponadto program DB2 Cube Views nie zapewnia obsługi

klauzuli PARTITION BY, która działa z funkcją pochodzącą z programu DB2 UDB. Więcej informacji o funkcji RANK zawiera Centrum informacyjne DB2 UDB.

DENSERANK

Porządkuje wiersze i każdemu z nich przypisuje ocenę. Ocena jest obliczana jako 1 plus liczba wszystkich wierszy poprzedzających danych wiersz. Przy takim sposobie liczenia wyniki rankingu będą tworzyć ciągłą sekwencję, bez przerw w numeracji. Tabela 7 na stronie 23 przedstawia przykład wyników działania funkcji DENSERANK w odniesieniu do zbioru przykładowych wartości.

Typowa składnia funkcji DENSERANK wygląda następująco:

```
DENSERANK ( ) OVER (ORDER BY wyrażenie-klucza-sortowania kolejność-wyrażenia)
```

gdzie *wyrażenie-klucza-sortowania* jest zbiorem danych poddawanych rankingowi, a *kolejność-wyrażenia* jest to słowo kluczowe **ASC** lub **DESC**, powodujące uporządkowanie wartości wyrażenia klucza sortowania w porządku rosnącym lub malejącym. W programie DB2 Cube Views istnieje wymóg, by *wyrażenie-klucza-sortowania* było istniejącą miarą, a nie kolumną ani atrybutem. Ponadto program DB2 Cube Views nie zapewnia obsługi klauzuli PARTITION BY, która działa z funkcją pochodzącą z programu DB2 UDB. Więcej informacji o funkcji DENSERANK zawiera Centrum informacyjne DB2 UDB.

ROWNUMBER

Oblicza kolejny numer wiersza w ustalonym porządku, począwszy od numeru 1 dla wiersza pierwszego. Jeśli nie zostanie użyta klauzula ORDER BY, numery są przypisywane wierszom w sposób przypadkowy.

Typowa składnia funkcji ROWNUMBER wygląda następująco:

```
ROWNUMBER ( ) OVER  
([ORDER BY wyrażenie-klucza-sortowania kolejność-wyrażenia])
```

gdzie *wyrażenie-klucza-sortowania* jest zbiorem danych poddawanych rankingowi, a *kolejność-wyrażenia* jest to słowo kluczowe **ASC** lub **DESC**, powodujące uporządkowanie wartości wyrażenia klucza sortowania w porządku rosnącym lub malejącym. W programie DB2 Cube Views istnieje wymóg, by źródłem danych tej funkcji była istniejąca miara, a nie kolumna lub atrybut. Ponadto program DB2 Cube Views nie zapewnia obsługi klauzuli PARTITION BY, która działa z funkcją pochodzącą z programu DB2 UDB. Więcej informacji o funkcji ROWNUMBER zawiera Centrum informacyjne DB2 UDB.

Wymienione tu funkcje OLAP nie figurują na liście funkcji i stałych w Kreatorze wyrażen SQL.

Tabela 7. Wyniki rankingu uzyskane po zastosowaniu funkcji RANK i DENSERANK dla przykładowego zestawu wartości

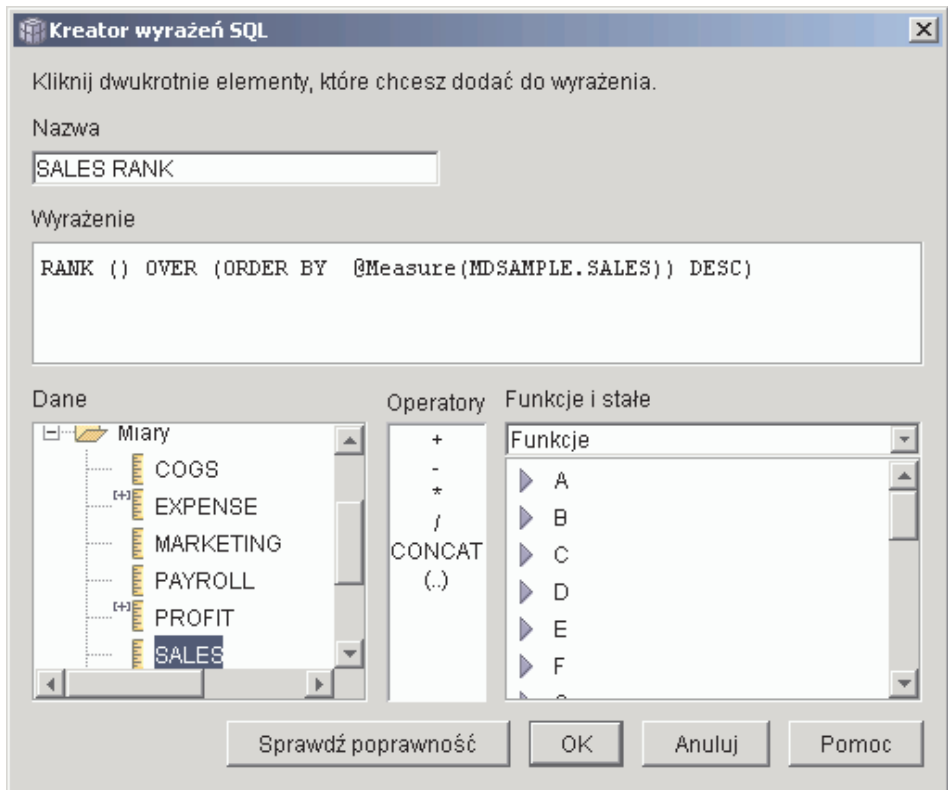
Wartości wiersza	Kolejność	Wyniki rankingu z funkcji RANK	Wyniki rankingu z funkcji DENSERANK
100	1	1	1
35	2	2	2
23	3	3	3
8	4	4	4
8	4	4	5
6	5	6	6

Czynności wymagane do utworzenia miary

W poniższej procedurze opisano sposób użycia okna Właściwości faktów w programie Centrum OLAP do utworzenia miary Sales Rank w istniejącym obiekcie faktów:

1. Aby otworzyć okno Właściwości faktów, kliknij prawym przyciskiem myszy obiekt faktów w drzewie obiektów Centrum OLAP, a następnie kliknij opcję **Edytuj miary**. Otwarte zostanie okno Właściwości faktów.
2. Na stronie Miary kliknij opcję **Utwórz miarę obliczaną**, aby utworzyć miarę Sales Rank. Otwarte zostanie okno Kreatora wyrażeń SQL.
3. W oknie Kreatora wyrażeń SQL, w polu **Nazwa**, wpisz SALES RANK.
4. Wykonaj poniższe czynności w celu utworzenia wyrażenia miary Sales Rank:
 - W polu Wyrażenie wpisz następujący kod funkcji: RANK () OVER (ORDER BY measure DESC).
 - Rozwiń folder **Miary** na liście **Dane**.
 - Podświetl wyraz **measure** w kodzie funkcji w polu **Wyrażenie** i kliknij dwukrotnie miarę **SALES** w celu dodania jej do wyrażenia.

Wyrażenie w ostatecznej postaci przedstawia Rys. 10 na stronie 24.



Rysunek 10. Gotowe wyrażenie miary Sales Rank w oknie Kreatora wyrażen SQL

5. Kliknij przycisk **Sprawdź poprawność**, aby zyskać pewność, że wyrażenie jest poprawne. Kliknij przycisk **OK**, aby zamknąć okno Kreatora wyrażen SQL.
Na stronie Agregacji należy zmienić domyślną funkcję agregującą, którą jest Brak. Opcja Brak jest wybierana domyślnie dla miary Sales Rank, ponieważ źródło danych jest liczbowe i odnosi się tylko do istniejących miar.

Korzystając z funkcji RANK w wyrażeniu miary do uporządkowania kolumny Sales w kolejności malejącej, analityk może schodzić do bardziej szczegółowych warstw danych w dowolnych innych wymiarach, na przykład w celu ustalenia placówki o najgorszych wynikach sprzedaży w minionym roku dla wybranej linii produktów.

Rozdział 6. Użycie danych czasowych w tabeli faktów do utworzenia wymiaru czasu

Przedsiębiorstwo zajmujące się handlem detalicznym, XYZ Retail, modeluje dane ze swoich transakcji sprzedaży w programie DB2 Cube Views, aby skorzystać z oferowanych przezeń funkcji analitycznych. Jednak z uwagi na transakcyjny charakter danych, jedyna dostępna informacja czasowa to data skojarzona z odpowiednią transakcją. Informacje o czasie modelowane w wymiarze czasu są niezbędne jako kontekst dla wielu typowych obliczeń, takich jak analizy tendencji w sprzedaży według kwartału i obliczanie średniej wartości zapasów w poszczególnych tygodniach.

Wielu administratorów unika przechowywania danych na temat czasu w postaci daty lub datownika transakcji, ponieważ brak transakcji za pewien dzień powoduje powstawanie przerw w ciągłości danych, co stwarza później problemy przy ich agregowaniu i wyświetlaniu. Zazwyczaj lepszym rozwiązaniem jest modelowanie danych czasowych w tabeli czasu. Jednak administrator bazy danych w firmie XYZ Retail jest przekonany, że każdego dnia rejestrowana będzie przynajmniej jedna transakcja i decyduje się pozostawić strukturę danych w obecnej postaci.

Szczegóły scenariusza

Firma XYZ Retail prowadzi tabelę faktów zawierającą mierzalne parametry każdej transakcji, takie jak Sales (Przychód), Costs (Koszt), Quantity Sold (Ilość sprzedana) i Date (Data). Ponadto w bazie danych znajdują się tabele wymiarowe Region oraz Product. Problemem jest to, że dane czasu przechowywane są w tabeli faktów, a nie w osobnej tabeli wymiaru. Administrator musi więc utworzyć obiekt wymiaru bazujący na informacjach o dacie znajdujących się w obiekcie faktów.

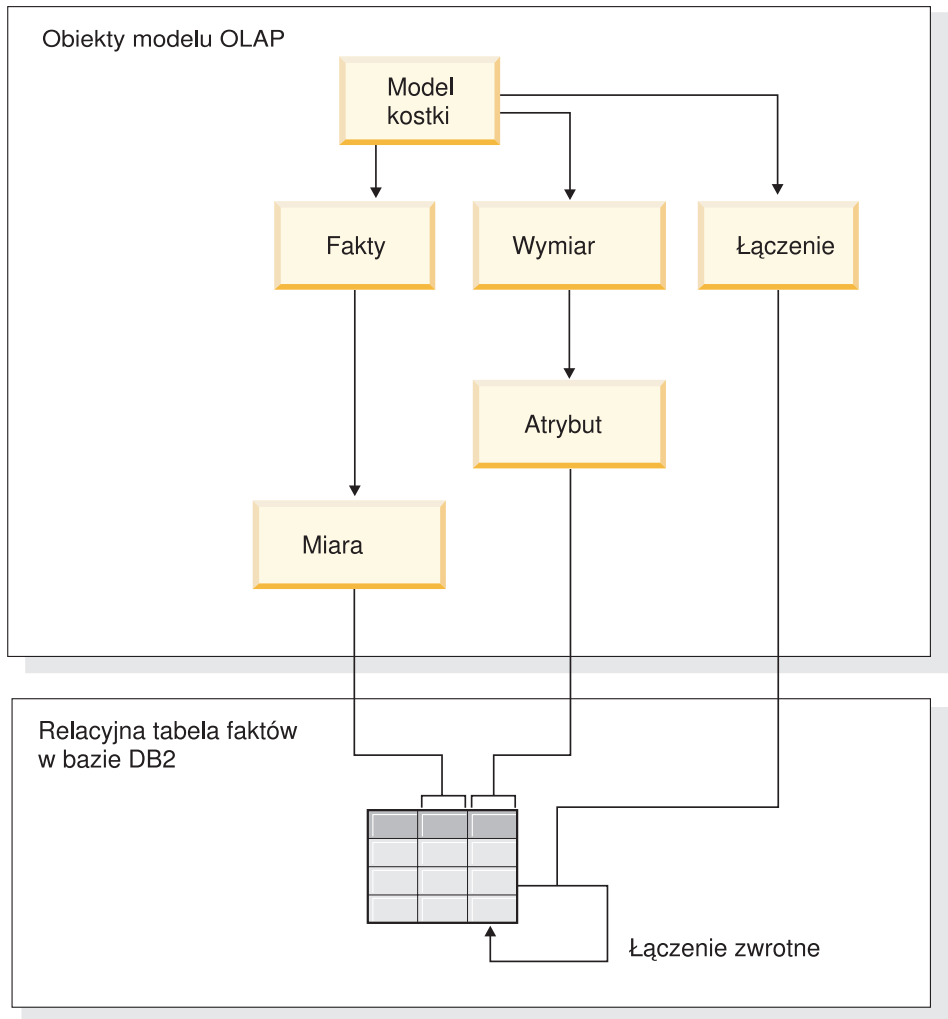
Tworząc wymiar czasu w oparciu o pojedynczą kolumnę dat w tabeli faktów, należy spełnić dwa szczególne wymagania:

- Ponieważ wszystkie obiekty wymiarów w poprawnym modelu kostki muszą być połączone z obiektem faktów, a zarówno obiekt wymiaru czasu, jak i obiekt faktów bazują na jednej i tej samej tabeli faktów, obiekt wymiaru czasu musi być połączony z obiektem faktów za pomocą łączenia zwrotnego, które połączy tabelę faktów z nią samą.
- Administrator musi zdefiniować atrybuty obliczane agregujące daty do poziomu przydatnego w analizie, takiego jak tydzień, miesiąc, kwartał lub rok.

Łączenie zwrotne jest to typ łączenia, w którym tabela jest łączona ze sobą. W tym przypadku będzie to tabela faktów. Łączenie zwrotne powinno łączyć jedną lub więcej kolumn, które wspólnie identyfikują jednoznacznie każdy wiersz w tabeli faktów. Najlepszą opcją będzie wybór klucza głównego. Jeżeli jednak klucz podstawowy nie jest

zdefiniowany, dobrą alternatywą będzie zestaw kolumn używany do łączenia tabeli faktów z tabelami wymiarów. Aby zoptymalizować model kostki, trzeba zdefiniować klucz podstawowy. Łączenie to musi mieć stosunek liczościowy 1:1 i być typu wewnętrznego.

Rys. 11 przedstawia sposób odwzorowania między obiektem faktów, wymiarem bazującym na tabeli faktów i połączeniem faktów z wymiarem a jedną i tą samą tabelą faktów.



Rysunek 11. Sposób łączenia tabeli z nią samą przy użyciu łączenia zwrotnego

Czynności wymagane przy tworzeniu atrybutów i wymiaru

W poniższej procedurze opisano sposób użycia Kreatora wymiarów w Centrum OLAP do utworzenia wymiaru Time (Czas) i atrybutów obliczanych bazujących na tabeli faktów:

1. Aby otworzyć Kreatora wymiarów, kliknij prawym przyciskiem myszy model kostki w drzewie obiektów Centrum OLAP i kliknij opcję **Utwórz wymiar**. Otwarty zostanie Kreator wymiarów.
2. Na karcie Nazwa wpisz **Time (Czas)** w polu **Nazwa**. Opcjonalnie można zmienić nazwę biznesową i wpisać komentarz. Kliknij przycisk **Dalej**.
3. Wybierz tabelę faktów modelu kostki. Kliknij przycisk **Dalej**. Nie ma potrzeby określania łączy wymiarowych, ponieważ wymiar zawiera tylko jedną tabelę. Na stronie Łączenia wymiaru kliknij przycisk **Dalej**.
- 4.
5. Na stronie Atrybuty wymiaru wybierz kolumnę **Timestamp** (Datownik).
6. Opcjonalnie: Utwórz dodatkowe atrybuty obliczane, które będą agregować datowniki w większe porcje, takie jak miesiąc, kwartał lub rok. Aby utworzyć atrybuty obliczane, kliknij przycisk **Utwórz atrybut obliczany**, aby otworzyć Kreatora wyrażeń SQL, a następnie zdefiniuj dla poszczególnych atrybutów wyrażenia obliczające miesiące, kwartały lub lata na podstawie źródłowych datowników. Po zdefiniowaniu poszczególnych atrybutów obliczanych kliknij przycisk **Sprawdź poprawność**, aby zweryfikować poprawność wyrażenia, a następnie kliknij przycisk **OK**, aby zamknąć Kreatora wyrażeń SQL i powrócić do Kreatora wymiarów. Kliknij przycisk **Dalej** po wybraniu i utworzeniu wszystkich wymaganych atrybutów.
7. Na karcie Typ wymiaru wybierz opcję **Czas**. Kliknij przycisk **Dalej**.
8. Na stronie Łączenie fakt-wymiar kliknij przycisk **Utwórz łączenie**. W otwartym oknie Kreatora łączy utwórz łączenie zwrotne. Wpisz nazwę i kliknij przycisk **Dalej**. Wybierz kolumnę lub zestaw kolumn jednoznacznie identyfikujący każdy wiersz w tabeli faktów, na przykład klucz podstawowy, dla atrybutu lewostronnego i prawostronnego. Wybieraj po jednej parze i klikaj przycisk **Dodaj**, aby dodać parę atrybutów do łączenia. Jako typ łączenia wybierz łączenie wewnętrzne i licznosc 1:1. Po dodaniu wymaganych par atrybutów kliknij przycisk **Zakończ**. Okno Kreatora łączy zostanie zamknięte.
9. Na stronie Łączenie fakt-wymiar kliknij przycisk **Zakończ**.

Dzięki zdefiniowaniu wymiaru czasu w modelu kostki firma XYZ Retail może korzystać z całkiem nowej warstwy znaczeniowej w swych analizach danych. Możliwe będzie teraz wykonywanie analiz związanych z czasem, na przykład dotyczących stanu zapasów.

Uwagi

Niniejsza publikacja została przygotowana z myślą o produktach i usługach oferowanych w Stanach Zjednoczonych.

IBM może nie oferować w innych krajach produktów, usług lub opcji omawianych w tej publikacji. Informacje o produktach i usługach dostępnych w danym kraju/regionie można uzyskać od lokalnego przedstawiciela IBM. Odwołanie do produktu, programu lub usługi IBM nie oznacza, że można użyć wyłącznie tego produktu, programu lub usługi. Zamiast nich można zastosować ich odpowiednik funkcjonalny pod warunkiem, że nie narusza to praw własności intelektualnej IBM. Jednakże cała odpowiedzialność za ocenę przydatności i sprawdzenie działania produktu, programu lub usługi, pochodzących od producenta innego niż IBM, spoczywa na użytkowniku.

IBM może posiadać patenty lub złożone wnioski patentowe na towary i usługi, o których mowa w niniejszej publikacji. Przedstawienie tego dokumentu nie daje żadnych uprawnień licencyjnych do tychże patentów. Zapytania dotyczące licencji można wysłać na piśmie na adres:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

Zapytania w sprawie licencji na informacje dotyczące zestawów znaków dwubajtowych (DBCS) należy kierować do lokalnych działów własności intelektualnej IBM (IBM Intellectual Property Department) lub zgłaszać na piśmie pod adresem:

IBM World Trade
Asia Corporation
Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106, Japan

Poniższy akapit nie obowiązuje w Wielkiej Brytanii, a także w innych krajach, w których jego treść pozostaje w sprzeczności z przepisami prawa miejscowego: INTERNATIONAL BUSINESS MACHINES CORPORATION DOSTARCZA TĘ PUBLIKACJĘ W STANIE, W JAKIM SIĘ ZNAJDUJE (“AS IS”), BEZ UDZIELANIA JAKIKOLWIEK GWARANCJI (W TYM TAKŻE RĘKOJMI), WYRAŹNYCH LUB DOMNIEMANYCH, A W SZCZEGÓLNOŚCI DOMNIEMANYCH GWARANCJI PRZYDATNOŚCI HANDLOWEJ, PRZYDATNOŚCI DO OKREŚLONEGO CELU ORAZ GWARANCJI, ŻE PUBLIKACJA TA NIE NARUSZA PRAW OSÓB TRZECICH. Ustawodawstwa niektórych krajów nie

dopuszczają zastrzeżeń dotyczących gwarancji wyraźnych i domniemanych w odniesieniu do pewnych transakcji; w takiej sytuacji powyższe zdanie nie ma zastosowania.

Informacje zawarte w niniejszym dokumencie mogą zawierać nieścisłości techniczne lub błędy typograficzne. Informacje te są okresowo aktualizowane, a zmiany te zostaną uwzględnione w kolejnych wydaniach tej publikacji. IBM zastrzega sobie prawo do wprowadzania ulepszeń i/lub zmian w produktach i/lub programach opisanych w tej publikacji w dowolnym czasie, bez wcześniejszego powiadomienia.

IBM ma prawo do używania i rozpowszechniania informacji przysłanych przez użytkownika w dowolny sposób, jaki uzna za właściwy, bez żadnych zobowiązań wobec ich autora.

Informacje na temat możliwości stosowania tego programu, takie jak: (i) wymiana informacji między niezależnie tworzonymi programami a innymi programami (włącznie z tym programem) czy (ii) wspólne używanie wymienianych informacji, można uzyskać pod adresem:

IBM Corporation
J46A/G4
555 Bailey Avenue
San Jose, CA 95141-1003
U.S.A.

Informacje takie mogą być udostępnione, o ile spełnione zostaną odpowiednie warunki, w tym, w niektórych przypadkach, uiszczenie odpowiedniej opłaty.

Licencjonowany program opisany w tej publikacji i wszystkie inne licencjonowane materiały dostępne dla tego programu są dostarczane przez IBM na warunkach określonych w Umowie IBM z Klientem, Międzynarodowej Umowie Licencyjnej IBM na Program lub w innych podobnych umowach zawartych między IBM i użytkownikami.

Informacje dotyczące produktów firm innych niż IBM pochodzą od dostawców tych produktów, z opublikowanych przez nich zapowiedzi lub innych powszechnie dostępnych źródeł. Firma IBM nie testowała tych produktów i nie może potwierdzić dokładności pomiarów wydajności, kompatybilności ani żadnych innych danych związanych z tymi produktami. Pytania dotyczące produktów firm innych niż IBM należy kierować do dostawców tych produktów.

Publikacja ta zawiera przykładowe dane i raporty używane w codziennych operacjach działalności gospodarczej. W celu kompleksowego ich zilustrowania, podane przykłady zawierają nazwiska osób prywatnych, nazwy przedsiębiorstw oraz nazwy produktów. Wszystkie te nazwy/nazwiska są fikcyjne i jakiegokolwiek podobieństwo do istniejących nazw/nazwisk i adresów jest całkowicie przypadkowe.

Wszelkie dane dotyczące wydajności zostały zebrane w kontrolowanym środowisku. W związku z tym rezultaty uzyskane w innych środowiskach operacyjnych mogą się znacząco różnić. Niektóre pomiary mogły być dokonywane na systemach będących w fazie rozwoju i nie ma gwarancji, że pomiary te wykonane na ogólnie dostępnych systemach dadzą takie same wyniki. Niektóre z pomiarów mogły być estymowane przez ekstrapolację. Rzeczywiste wyniki mogą być inne. Użytkownicy powinni we własnym zakresie sprawdzić odpowiednie dane dla ich środowiska.

Informacje dotyczące produktów firm innych niż IBM pochodzą od dostawców tych produktów, z opublikowanych przez nich zapowiedzi lub innych powszechnie dostępnych źródeł. Firma IBM nie testowała tych produktów i nie może potwierdzić dokładności pomiarów wydajności, kompatybilności ani żadnych innych danych związanych z tymi produktami. Pytania dotyczące produktów firm innych niż IBM należy kierować do dostawców tych produktów.

Przy przeglądaniu tych informacji w formie elektronicznej, fotografie i ilustracje kolorowe mogą się nie pojawić.

Znaki towarowe

Następujące nazwy są znakami towarowymi International Business Machines Corporation w Stanach Zjednoczonych i/lub w innych krajach:

DB2
DB2 Universal Database
IBM
Office Connect
Redbooks

Następujące nazwy są znakami towarowymi lub zastrzeżonymi znakami towarowymi innych podmiotów:

Microsoft, Windows, Windows NT, Windows 2000, Windows XP i Microsoft Excel są znakami towarowymi lub zastrzeżonymi znakami towarowymi Microsoft Corporation.

Java oraz wszystkie znaki towarowe i logo dotyczące Java i Solaris są znakami towarowymi Sun Microsystems, Inc. w Stanach Zjednoczonych i/lub w innych krajach.

UNIX jest zastrzeżonym znakiem towarowym w Stanach Zjednoczonych i/lub w innych krajach i jest licencjonowany na zasadzie wyłączności przez X/Open Company Limited.

Linux jest zastrzeżonym znakiem towarowym Linusa Torvaldsa. Red Hat oraz wszystkie znaki towarowe i logo dotyczące Red Hat są znakami towarowymi lub zastrzeżonymi znakami towarowymi Red Hat, Inc. w Stanach Zjednoczonych i/lub w innych krajach.

Nazwy innych firm, produktów i usług mogą być znakami towarowymi lub znakami usług innych podmiotów.