



ADAM PELIKANT

HURTOWNIE DANYCH

Od przetwarzania analitycznego
do raportowania

Spec od hurtowni danych? Zawsze będzie pilnie potrzebny!
Jak stworzyć strukturę hurtowni danych i dokonać ich integracji?
Jak przeprowadzić analizę danych z wykorzystaniem rozszerzenia MDK SQL?
Do czego potrzebne jest raportowanie?

» Idź do

- Spis treści
- Przykładowy rozdział
- Skorowidz

» Katalog książek

- Katalog online
- Zamów drukowany katalog

» Twój koszyk

- Dodaj do koszyka

» Cennik i informacje

- Zamów informacje o nowościach
- Zamów cennik

» Czytelnia

- Fragmenty książek online

» Kontakt

Helion SA
ul. Kościuszki 1c
44-100 Gliwice
tel. 32 230 98 63
e-mail: helion@helion.pl
© Helion 1991–2011

Hurtownie danych. Od przetwarzania analitycznego do raportowania

Autor: Adam Pelikant
ISBN: 978-83-246-2977-0
Format: 158×235, stron: 400



Spec od hurtowni danych? Zawsze będzie pilnie potrzebny!

- Jak stworzyć strukturę hurtowni danych i dokonać ich integracji?
- Jak przeprowadzić analizę danych z wykorzystaniem rozszerzenia MDX SQL?
- Do czego potrzebne jest raportowanie?

Idea hurtowni danych ściśle wiąże się z ich kolosalnymi ilościami, gromadzonymi podczas tysięcy różnych sytuacji – przy dowolnej transakcji, w urzędzie, na lotnisku, w internecie... Nawet nasze połączenia telefoniczne są przechowywane przez operatora. Te wszystkie dane trzeba gdzieś pomieścić, sensownie posegregować i zapewnić sobie możliwość sięgnięcia do wybranego ich zakresu bez długotrwałych poszukiwań. Taką możliwość dają właśnie hurtownie danych – przemysłane, bardzo pojemne bazy, oferujące zarówno integrację wprowadzanych danych, jak i znakomite mechanizmy ich przeszukiwania. Jeśli chcesz poszerzyć swoją wiedzę na temat tworzenia i przeglądania zawartości hurtowni danych, trafiłeś pod właściwy adres!

Książka „Hurtownie danych. Od przetwarzania analitycznego do raportowania” zawiera materiał przeznaczony nie tylko dla studentów wydziałów informatycznych, ale także dla pasjonatów tej tematyki oraz specjalistów zainteresowanych poszerzeniem wiedzy. W możliwie najprostszy, praktyczny sposób opisano w niej składnię i postać zapytań analitycznych, strukturę hurtowni danych oraz kwestię ich integracji i wizualnego tworzenia elementów hurtowni. Znajdziesz tu także omówienie analizy danych z wykorzystaniem rozszerzenia MDX SQL oraz zastosowań raportowania. Zapoznanie się z tymi informacjami oraz przesłanie zgromadzonych tu przykładów pozwoli Ci zrozumieć problemy powstające przy budowie hurtowni danych i wykorzystać tę wiedzę we własnych projektach.

- Zapytania analityczne
- Struktura hurtowni danych
- Integracja danych
- Wizualne tworzenie elementów hurtowni danych
- Analiza danych z wykorzystaniem rozszerzenia MDX SQL
- Raportowanie

Od bazy do hurtowni danych... Skocz na głęboką wodę!

Spis treści

Od autora	5
Rozdział 1. Wstęp	7
Rozdział 2. Zapytania analityczne	13
Podstawy składni	13
Grupowanie w SQL	18
Grupowanie nad oknem logicznym	28
Operator COMPUTE	32
Funkcje agregujące zdefiniowane przez użytkownika	35
Rozdział 3. Struktura hurtowni danych	43
Rola hurtowni danych w procesie przetwarzania	43
Proces integracji danych	46
Elementy hurtowni danych	48
Rozdział 4. Integracja danych	57
Wprowadzenie do Integration Services	57
Prosta migracja danych	59
Kontener FOR LOOP	72
Kontener FOREACH LOOP	83
Sprawdzanie zgodności danych ze słownikiem	94
Uruchamianie pakietów integracyjnych	104
Wykorzystywanie zapytań SQL do migracji danych	109
Rozdział 5. Wizualne tworzenie elementów hurtowni danych	119
Wstęp do Analysis Services	119
Tworzenie podstawowej struktury hurtowni danych	132
Modyfikacja struktury hurtowni danych	146
Kostka o strukturze płata śniegu	160
Tworzenie hurtowni danych z zastosowaniem tabel pośrednich	163
Definiowanie zaawansowanych elementów kostki	173
Struktura uprawnień do korzystania z hurtowni danych	184
Dodatkowe funkcjonalności Analysis Services	187

Rozdział 6. Analiza danych z wykorzystaniem rozszerzenia MDX SQL	203
Podstawy składni zapytań MDX	203
Operacje na zbiorach atrybutów	212
Definiowanie miar ad hoc	218
Definiowanie ad hoc zbiorów atrybutów	225
Zastosowanie wskazania poziomu hierarchii do wyznaczania miar	230
Wyświetlanie wielu poziomów hierarchii	234
Wyznaczanie miar jako wyrażeń dla różnych elementów i poziomów hierarchii	236
Filtrowanie w zapytaniach MDX	246
Wyznaczanie przedziałów i zakresów dla wymiarów	251
Zastosowanie instrukcji warunkowych	260
Zastosowanie funkcji agregujących w zapytaniach MDX	262
Definiowanie złożonej struktury dla wymiaru czasu	267
Definiowanie operacji na zbiorach atrybutów	270
Funkcje analityczne i statystyczne w MDX	273
Podsumowanie wiadomości o zapytaniach wybierających MDX	278
Tworzenie i testowanie nietrwałych struktur wielowymiarowych	287
Rozdział 7. Raportowanie	317
Zastosowanie MS Excel do tworzenia raportów dla hurtowni danych	317
Zastosowanie języków wyższego rzędu do tworzenia raportów	324
Zastosowanie Reporting Services — podstawy	331
Konfigurowanie serwera http dla potrzeb Reporting Services	339
Synchronizowanie raportów	346
Raporty o strukturze macierzowej	356
Definiowanie akcji dla raportów	361
Definiowanie grup hierarchicznych	365
Raportowanie dla danych pochodzących z hurtowni	369
Rozdział 8. Podsumowanie.	
Co dalej z analitycznym przetwarzaniem danych?	383
Skorowidz	387

Wykorzystywanie zapytań SQL do migracji danych

Na zakończenie tego rozdziału trochę przekornie można powiedzieć, że w zasadzie omawiane w nim narzędzie nie jest w ogóle potrzebne. Każdy „twardy” programista powie, że prawdziwą wartość ma tylko takie rozwiązanie, w którym widoczny jest kod. Doda jeszcze, że oprogramować można wszystko. Zwolennicy baz danych i ich rozszerzeń proceduralnych stwierdzą, że niepotrzebne są języki wyższego rzędu i że wystarczy samo środowisko serwera. Niestety, mają dużo racji. Sam wolę programować niż „klikać ikony” czy przeciągać je i tworzyć „ładne obrazki”. Co można zaproponować w zamian? Aby zrealizować przykład konkurencyjnego rozwiązania, utworzymy dwie tabele: *Product*, opisującą towary i ich przynależność do kategorii, przeznaczoną na dane docelowe, oraz tabelę *Błady*, opisującą proces kopiowania. Druga z tabel zawiera „zdublowane” kolumny tabeli docelowej, raz poprzedzone prefiksem *Existing*, a drugi raz *New*. Między tymi grupami wprowadzono znakową kolumnę o nazwie *ActionTaken*.

```
CREATE TABLE Product (  
  PK_Product int PRIMARY KEY,  
  Product_Name varchar(50),  
  Category int,  
  Category_Name varchar(50)  
);
```

GO

```
CREATE TABLE Błady (  
  ExistingProduct_PK int,  
  ExistingProduct_Name nvarchar(50),  
  ExistingCategory int,  
  ExistingCategory_Name nvarchar(50),  
  ActionTaken nvarchar(10),  
  NewProduct_PK int,  
  NewProduct_Name nvarchar(50),  
  NewCategory int,  
  NewCategory_Name nvarchar(50)  
);
```

Właściwą akcją będzie przepisanie danych pochodzących z dwóch tabel do tabeli *Product*. Zastosowane zostanie polecenie *MERGE*, a tabela docelowa będzie opatrzona aliasem *Target*. Po słowie kluczowym *USING* zdefiniowano zapytanie wybierające kopiowane dane z tabel *Towary* i *Kategorie*. Dopuszczalne jest stosowanie wszystkich elementów składniowych występujących w tego typu zapytaniach.

```
MERGE Product AS Target  
USING (  
  SELECT IdTowaru, NazwaTowaru, Kategorie.IdKategorii, NazwaKategorii  
  FROM Towar JOIN Kategorie  
  ON Towar.IdKategorii=Kategorie.IdKategorii  
  WHERE IdTowaru<=10  
)
```

```

AS Source (IdTowaru, NazwaTowaru, IdKategorii, NazwaKategorii)
ON (IdTowaru=Target.PK_Product)
WHEN NOT MATCHED BY Target
THEN
INSERT (PK_Product, Product_Name, Category, Category_Name)
VALUES(IdTowaru, NazwaTowaru, IdKategorii, NazwaKategorii)
WHEN MATCHED AND Target.Category IS NULL
THEN DELETE
WHEN MATCHED AND Target.Category IS NOT NULL
THEN UPDATE SET target.Category = SOURCE.IdKategorii
OUTPUT DELETED.*, $action, INSERTED.* INTO Blady;

```

W przykładzie zdecydowano się na wybranie rekordów, w których identyfikator towaru nie przekracza 10, a zapytanie określające źródło zostało opatrzone aliasem *Source*. Słowo kluczowe *ON* wyznacza sekcję, w której określamy warunki, według których dane ze źródła będą porównywane z danymi zawartymi w obiekcie docelowym. Warunek *WHEN NO MATCHED* definiuje działanie w przypadku wykrycia rekordów, dla których wyrażenie sprawdzające jest fałszywe, co oznacza, że wiersza nie ma jeszcze w obiekcie docelowym. Naturalne jest wtedy wstawienie nowego wiersza. Stosowana jest składnia podobna do zwykłego zapytania *INSERT*, ale pomijana jest nazwa obiektu docelowego, ponieważ definiuje ją nagłówek polecenia *MERGE*. Po słowie kluczowym *Values* występuje separowana przecinkami lista pól z zapytania źródłowego, o kolejności zgodnej z listą pól wymienioną po słowie *INSERT*. Jeśli zasilamy wszystkie pola, lista pól docelowych nie jest obowiązkowa, a kolejność wartości musi być zgodna z kolejnością pól w tabeli docelowej — patrz polecenie *CREATE TABLE*. W przykładowym zapytaniu określono dwojaki sposób działania, kiedy wykryto zgodność między źródłem a celem — *WHEN MATCHED*. W obu sytuacjach sprawdzono dodatkowo, jaką wartość ma pole *Category*. Jeśli w pierwszym przypadku było *NULL*, następowało wykasowanie rekordu z obiektu docelowego, w przeciwnym razie dokonano zmiany tego pola na zgodne z tym, które wystąpiło w źródle. Oczywiście należałoby zmodyfikować wszystkie pola z wyjątkiem pola klucza głównego, ale spowodowałoby to dalszą rozbudowę przykładu. Ostatnim elementem jest zdefiniowanie pól, które mają się znaleźć w tabeli *Blady*, definiującej wykonane procesem operacje. Wykorzystano dwie tabele systemowe tworzone podczas wykonywania tego polecenia, które stanowią analogię dla tabel tworzonych w trakcie pracy procedur wyzwalanych — triggerów. Tabela *DELETED* zawiera stare, natomiast *INSERTED* nowe wartości pól rekordów. Dodatkowo wykorzystano polecenie wewnętrzne *\$action*, zwracające nazwę akcji, która została wykonana dla danego rekordu. Średnik kończący polecenie *MERGE* jest obowiązkowy, co nie ma odniesienia do innych poleceń SQL. Tabela 4.12 zawiera rekordy, jakie pojawiły się w tabeli docelowej, natomiast tabela 4.13 przedstawia rekordy zapisane do tabeli *Blady*.

Pierwsze wykonanie zapytania, dla pustego obiektu docelowego było równoważne wykonaniu zapytania wstawiającego wiersze typu *INSERT ... SELECT*. Powtórzmy ten proces, zmieniając w poleceniu *MERGE* zakres kopiowanych wierszy w klauzuli *WHERE*. Tym razem będą to rekordy, dla których identyfikator wiersza jest z przedziału obustronnie domkniętego $<5, 15>$, jak pokazuje przykład. Pozostałe elementy składniowe nie uległy zmianie. Przed wykonaniem tego polecenia zawartość tabeli docelowej nie była czyszczona.

Tabela 4.12. Skutek wykonania polecenia MERGE — tabela wynikowa

PK_Product	Product_Name	Category	Category_Name
1	Mazurki FCh	9	Muzyka
2	Zestaw Lux	7	Akcesoria
3	Gio 15	5	GPS
4	Leksykon	19	Podręczniki
5	Fizyka	19	Podręczniki
6	Statyw AX	7	Akcesoria
7	Dodatki +3	12	Oprogramowanie
8	Dodatki +7	12	Oprogramowanie
9	Mikrofala	2	RTV
10	Podstawka	7	Akcesoria

Tabela 4.13. Skutek wykonania polecenia MERGE — tabela opisująca wykonywaną akcje

ExistingProduct_PK	ExistingProduct_Name	ExistingCategory	ExistingCategory_Name	ActionTaken	NewProduct_PK	NewProduct_Name	NewCategory	NewCategory_Name
NULL	NULL	NULL	NULL	INSERT	1	Mazurki FCh	9	Muzyka
NULL	NULL	NULL	NULL	INSERT	2	Zestaw Lux	7	Akcesoria
NULL	NULL	NULL	NULL	INSERT	3	Gio 15	5	GPS
NULL	NULL	NULL	NULL	INSERT	4	Leksykon	19	Podręczniki
NULL	NULL	NULL	NULL	INSERT	5	Fizyka	19	Podręczniki
NULL	NULL	NULL	NULL	INSERT	6	Statyw AX	7	Akcesoria
NULL	NULL	NULL	NULL	INSERT	7	Dodatki +3	12	Oprogramowanie
NULL	NULL	NULL	NULL	INSERT	8	Dodatki +7	12	Oprogramowanie
NULL	NULL	NULL	NULL	INSERT	9	Mikrofala	2	RTV
NULL	NULL	NULL	NULL	INSERT	10	Podstawka	7	Akcesoria

```

MERGE Product AS TARGET
USING(
SELECT IdTowaru, NazwaTowaru, Kategorie.IdKategorii, NazwaKategorii
FROM Towar JOIN Kategorie
ON Towar.IdKategorii=Kategorie.IdKategorii
WHERE IdTowaru>=5 AND IdTowaru <=15
)
AS SOURCE (IdTowaru, NazwaTowaru, IdKategorii, NazwaKategorii)
ON (Idtowaru=TARGET.PK_Product)
WHEN NOT MATCHED BY TARGET
THEN
INSERT (PK_Product, Product_Name, Category, Category_Name)
VALUES(IdTowaru, NazwaTowaru, IdKategorii, NazwaKategorii)
WHEN MATCHED AND target.Category IS NULL
THEN DELETE

```

```

WHEN MATCHED AND target.Category IS NOT NULL
THEN UPDATE SET target.Category = SOURCE.IdKategorii
OUTPUT DELETED.*, $action, INSERTED.* INTO bład;

```

Po wykonaniu przedstawionego polecenia zawartość tabeli docelowej uległa zmianie. Pozostawiono do tej pory wpisane rekordy oraz zostały dopisane rekordy wynikające z powiększenia zakresu danych źródłowych. Zawartość tabeli docelowej po wykonaniu tej operacji przedstawia tabela 4.14.

Tabela 4.14. Skutek ponownego wykonania polecenia *MERGE* — tabela wynikowa

PK_Product	Product_Name	Category	Category_Name
1	Mazurki FCh	9	Muzyka
...
5	Fizyka	19	Podręczniki
6	Statyw AX	7	Akcesoria
...
15	Luneta stand	6	Optyka

Ważniejsza jest analiza tabeli *Bład*, zawierającej informacje o przeprowadzonych w trakcie wykonywania przedstawianego zapytania (tabela 4.15). Tak samo jak w przypadku tabeli *Product* również i ta tabela nie była czyszczona, dlatego pierwszych dziesięć rekordów *ExistingProduct_PK=NULL* i *NewProduct_PK<=10* jest świadectwem wykonania poprzedniej operacji. Kolejne wiersze, dla których *ExistingProduct_PK=NewProduct_PK* pokazują, że dla wierszy, które już istniały w tabeli docelowej i ponownie próbowano je zapisać, wykonano zapytanie modyfikujące *UPDATE*. Ostatnia grupa rekordów, *ExistingProduct_PK=NULL*, to ta, dla których nie istniały jeszcze rekordy i tak samo jak przy pierwszej próbie wykonania polecenia *MERGE* zostały dopisane do tabeli.

Tabela 4.15. Skutek ponownego wykonania polecenia *MERGE* — tabela opisująca wykonywaną akcję

ExistingProduct_PK	ExistingProduct_Name	ExistingCategory	ExistingCategory_Name	ActionTaken	NewProduct_PK	NewProduct_Name	NewCategory	NewCategory_Name
NULL	NULL	NULL	NULL	INSERT	1	Mazurki FCh	9	Muzyka
...
NULL	NULL	NULL	NULL	INSERT	10	Podstawka	7	Akcesoria
5	Fizyka	19	Podręczniki	UPDATE	5	Fizyka	19	Podręczniki
...
10	Podstawka	7	Akcesoria	UPDATE	10	Podstawka	7	Akcesoria
NULL	NULL	NULL	NULL	INSERT	11	Statyw	7	Akcesoria
...
NULL	NULL	NULL	NULL	INSERT	15	Luneta stand	6	Optyka

Przykład realizacji polecenia *MERGE* jest z przyczyn formalnych bardzo uproszczony, ponieważ w przypadku zgodności kluczy podstawowych należałoby dodatkowo sprawdzić zgodność wartości wszystkich pól zapytania określającego źródło oraz tabeli docelowej. Gdy zgodność jest pełna, nie powinno się podejmować żadnej akcji, tak aby nie wydłużać czasu przetwarzania. Formalnie lepiej jest sprawdzić warunek odwrotny i w razie niezgodności jednego z pól wykonać odpowiednią modyfikację. Oczywiście w dużej mierze sposób realizacji zapytania zależy od postawionych przed procesem zasilania wymagań formalnych.

W prezentowanym przykładzie przyjęto, że źródłem są tabele zapisane po stronie schematu relacyjnego na serwerze MS SQL. W praktyce możliwe jest stosowanie jako źródła danych pliku tekstowego. Proces takiej migracji realizowany jest dzięki wykorzystaniu polecenia *BULK INSERT*. Należy w nim zdefiniować nazwę tabeli docelowej oraz po słowie kluczowym *FROM* kwalifikowaną nazwę pliku źródłowego wraz z rozszerzeniem. W sekcji *WITH* podawana jest ujęta w nawiasy, rozdzielana przecinkami lista parametrów importu z określeniem ich wartości.

```
BULK INSERT Product
FROM 'C:\Książka_hurtownie_helion\produkty.csv'
WITH
(
  FIELDTERMINATOR = ',' ,
  ROWTERMINATOR = '\n' ,
  FIRSTROW = 2 ,
  MAXERRORS = 100 ,
  ERRORFILE = 'C:\Książka_hurtownie_helion\bldy.txt'
)
```

W przykładzie podane zostały parametry określające:

- ♦ *FIELDTERMINATOR* — separator rozdzielający kolejne pola rekordu, najczęściej przecinek lub średnik;
- ♦ *ROWTERMINATOR* — separator rozdzielający kolejne rekordy, najczęściej zmiana wiersza;
- ♦ *FIRSTROW* — od którego wiersza należy czytać dane z pliku źródłowego; wartość większa od 1 jest najczęściej spowodowana istnieniem nagłówka albo tym, że plik zawiera nazwy kolumn;
- ♦ *MAXERRORS* — po jakiej liczbie błędów proces ma bezwarunkowo zakończyć działanie;
- ♦ *ERRORFILE* — nazwę pliku tekstowego, do którego będą wpisywane wiersze zawierające błędne dane (oprócz tego tworzony jest plik o dodatkowym rozszerzeniu *Error.Txt*, zawierający komunikaty odnoszące się do tych wierszy).

Ponadto możliwe jest jeszcze zdefiniowanie innych parametrów, które określają:

- ♦ *BATCHSIZE* — liczba wierszy w jednej porcji, każda z nich jest kopiowana w ramach pojedynczej transakcji; w przypadku błędu krytycznego każda z porcji jest oddzielnie zatwierdzana *Commit* lub wycofywana *ROLLBACK*, domyślnie cały plik stanowi pojedynczą transakcję;

- ◆ *CHECK_CONSTRAINTS* — wszystkie ograniczenia zdefiniowane w tabeli docelowej są sprawdzane podczas importu; jeśli nie zostanie ustawiona ta opcja, ograniczenia walidujące *CHECK* oraz kluczy obcych *FOREIGN KEY* nie są sprawdzane w trakcie importowania, a po zakończeniu są oznaczone flagą *NOT-TRUSTED*; ograniczenia *UNIQUE*, *PRIMARY KEY* oraz *NOT NULL* są zawsze sprawdzane;
- ◆ *CODEPAGE* = { 'ACP' | 'OEM' | 'RAW' | 'code_page' } — określa sposób kodowania pól znakowych (*ACP* — konwersja na format ANSI lub Microsoft Windows ISO 1252, *OEM* (domyślna) — następuje konwersja od strony kodowej systemu operacyjnego do strony kodowej zdefiniowanej w SQL Server, *RAW* — nie następuje konwersja, *code_page* — wskazuje jawnie numer strony kodowej), format *UTF8* nie jest obsługiwany;
- ◆ *DATAFILETYPE* { 'char' | 'native' | 'widechar' | 'widenative' } — określa reprezentacje zmiennych znakowych (*char* (domyślnie) — zwykłe typy znakowe, *native* — typy znakowe w narodowej stronie kodowej, *widechar* — zmienne w formacie *unicode*, *widenative* — typy znakowe w narodowej stronie kodowej z wyjątkiem tych, dla których zastosowano *unicode*);
- ◆ *FIRE_TRIGGERS* — powoduje uruchomienie procedur wyzwalanych, utworzonych dla zdarzenia *INSERT*; procedury te są uruchamiane dla każdej zakończonej transakcji i porcji danych;
- ◆ *FORMATFILE* — wskazuje kwalifikowaną nazwę pliku formatującego, utworzonego za pomocą polecenia *bcp*; odwołanie do niego następuje, gdy plik danych zawiera zbyt dużo lub zbyt mało kolumn, kolumny są zapisywane w innej kolejności niż występująca w pliku źródłowym, zmienia się ogranicznik kolumn lub zmieniają się inne elementy formatu; plik ten może być edytowany tak jak każdy plik tekstowy;
- ◆ *KEEPIDENTITY* — wskazuje, że do pól automatycznie inkrementowanych *IDENTITY* wstawiane będą wartości pobrane z pliku; jeśli nie wyspecyfikowano, uruchamiana jest generacja wartości po stronie serwera;
- ◆ *KEEPNULLS* — do pustych pól będą wstawiane wartości *NULL*;
- ◆ *KILOBYTES_PER_BATCH* — określa szacunkowy przydział pamięci na jedną porcję danych; domyślnie jest wartością nieokreśloną, wynikającą z ustawienia parametru *BATCHSIZE*;
- ◆ *LASTROW* — numer ostatniego wiersza, który będzie importowany z pliku;
- ◆ *ORDER* ({column [ASC | DESC]}[,...n]) — określa, w jaki sposób będą sortowane dane w tabeli docelowej; nie ma zastosowania, jeśli został na niej utworzony indeks grupujący *CLUSTERED*;
- ◆ *ROWS_PER_BATCH* — szacunkowa liczba wierszy w zbiorze źródłowym;
- ◆ *TABLOCK* — blokuje dostęp do kopiowanych wierszy przez transakcje odwołujące się do zasilanej tabeli; wynika to z poziomu izolacji transakcji.

Przy okazji można wskazać na dwa ważne przypadki, które powodują krytyczny błąd przetwarzania. Pierwszy z nich ma miejsce wtedy, gdy plik przeznaczony na błędne wiersze już istnieje. Pojawia się wtedy komunikat o postaci:

```
Msg 4861, Level 16, State 1, Line 1
Cannot bulk load because the file "C:\Książka_hurtownie_helion\błady.txt" could not be
opened. Operating system error code 80 (Plik istnieje).
```

Kolejny dotyczy zasilania tabeli, dla której zdefiniowano klucz podstawowy, a w źródle istnieją wiersze powodujące powielenie ich wartości. Komunikat ma wtedy postać:

```
Msg 2627, Level 14, State 1, Line 1
Violation of PRIMARY KEY constraint 'PK__Product__E5E6F73B06CD04F7'. Cannot insert
duplicate key in object 'dbo.Product'.
The statement has been terminated.
```

Aby pokazać działanie kopiowania masowego, przygotowany został plik tekstowy o postaci:

```
T_PK,Tow,K_PK,Kat
21,Mazurki FCh,9,11
22,Zestaw,xxx,Akcesoria
23,Gio 15,7
24,Leksykon;
25,19,Podręczniki
26,Statyw AX,7,Akcesoria
```

Pierwszy wiersz zawiera nazwy kolumn. W kolejnych wierszach zawarte są dane. Tylko wiersz o identyfikatorze 21 zawiera dane we właściwej postaci. W następnych wierszach są zawarte dane o niezgodnym typie albo jest ich zbyt mało, albo zastosowano zły separator. Na skutek wykonania polecenia *BULK INSERT* wszystkie niepoprawne wiersze zostały umieszczone w pliku *błady.txt*.

```
22,Zestaw,xxx,Akcesoria
23,Gio 15,7
24,Leksykon;
25,19,Podręczniki
26,Statyw AX,7,Akcesoria
```

Natomiast komunikaty związane z tymi błędami zawarte są w pliku *błady.txt.Error.Txt* i mają postać:

```
Row 3 File Offset 40 ErrorFile Offset 0 - HRESULT 0x80020005
Row 4 File Offset 65 ErrorFile Offset 25 - HRESULT 0x80020005
Row 5 File Offset 92 ErrorFile Offset 52 - HRESULT 0x80020005
```

Oczywiście wnikliwy Czytelnik powie, że źródła w postaci danych zapisanych w MS SQL Server oraz w plikach tekstowych nie wyczerpują wszelkich możliwości integracji danych pochodzących ze źródeł heterogenicznych. Cóż, można założyć, że każdy komercyjny serwer baz danych może generować plik tekstowy, a ten może być odczytany przy użyciu *BULK INSERT*, ale powoduje to konieczność utworzenia „warstwy pośredniej”, co spowalnia proces migracji. Istnieje jednak możliwość bezpośredniego odpytania serwerów innych typów. Pierwszym rozwiązaniem jest zastosowanie polecenia *OPENROWSET*, w którym należy zdefiniować trzy parametry. Pierwszym jest nazwa dostawcy sterownika do serwera wybranego typu, drugi, charakterystyczny dla

każdego sterownika, z reguły zawiera informacje o instancji bazy danych oraz trybie uwierzytelnienia, trzecim jest zapytanie, które ma zostać wykonane w określonej lokalizacji. Prezentowany przykład odnosi się do serwera MS SQL i ma za zadanie wyświetlać nazwy towarów z tabeli *Towar* przy uwierzytelnieniu dziedziczonym po systemie operacyjnym.

```
SELECT a.*
FROM OPENROWSET('SQLNCLI', 'Server=AP;Trusted_Connection=yes;',
'SELECT NazwaTowaru
FROM BazaRelacyjna.dbo.Towar') AS a;
```

Drugi wariant pokazuje wykonanie tego samego zapytania na tej samej maszynie przy zmianie uwierzytelnienia na określone po stronie serwera bazy danych.

```
SELECT a.*
FROM OPENROWSET('SQLNCLI', 'Data Source=AP;UID=sa;PWD=haslo;',
'SELECT NazwaTowaru
FROM BazaRelacyjna.dbo.Towar') AS a;
```

W trzecim przypadku zmieniona została definicja źródła danych w ten sposób, że zamiast zastosowania zapytania wybierającego odwołano się wprost do tabeli. Jest to równoważne wybraniu z niej wszystkich pól i wierszy.

```
SELECT *
FROM OPENROWSET('SQLNCLI', 'Data Source=AP;UID=sa;PWD=haslo;',
BazaRelacyjna.dbo.Towar)
```

Zamiast podawać łańcuch połączeniowy w sposób jawny, używając nazw parametrów połączenia, możemy uzyskać ten sam efekt, podając tylko ich wartości separowane średnikami.

```
SELECT *
FROM OPENROWSET('SQLNCLI', 'AP';'sa';'haslo',
BazaRelacyjna.dbo.Towar)
```

Można oczywiście powiedzieć, że to jednak nie rozwiązuje problemu, ponieważ w dalszym ciągu poruszamy się w sferze serwerów MS. Pokazywanie wszystkich przykładów dla każdego możliwego do zdefiniowania sterownika do serwera baz danych nie jest celowe. Dlatego ograniczmy się tylko do odpytania źródła w przypadku najsilniejszej konkurencji, jaką jest niewątpliwie Oracle. W prezentowanym przykładzie zastosowany został sterownik Microsoftu dla Oracle o nazwie *MSDAORA*; *sid* określa nazwę serwisu (instancji serwera), z którym się łączymy.

```
SELECT *
FROM OPENROWSET('MSDAORA', 'sid';'uzytkownik';'haslo',
SELECT * FROM Osoby)
```

Innym podejściem jest zdefiniowanie serwera połączonego *LINKED SERVER*. W takim przypadku definicję łańcucha połączeniowego wykonujemy w narzędziach wizualnych MS SQL, a w zapytaniu pozostaje nam jedynie odwołać się do jego nazwy *Połączony* oraz zdefiniować zapytanie lub wskazać na tabelę albo perspektywę.

```
SELECT * FROM OPENQUERY(Połączony,
'SELECT * FROM BazaRelacyjna.dbo.towar') AS test
```

W związku z tym pozostaje odpowiedzieć na podstawowe pytanie. Czy warto korzystać z narzędzi wizualnych Integration Services? Mimo wszystko wydaje się, że tak. Przede wszystkim proponowane przez twórców kontrolki mają wbudowane algorytmy optymalizujące, co może mieć wpływ na czas przetwarzania zadań. W przypadku bardzo dużych wolumenów danych jest to nie do przecenienia. Jedynym problemem z wydajnością przetwarzania może być stosowanie wielu tabel pośredniczących, co znacząco spowalnia pracę (zapis i odczyt z urządzeń fizycznych są znacznie wolniejsze niż wszystkie inne operacje). Dlatego w miarę możliwości należy takich operacji unikać. Jeśli jednak wolimy korzystać bezpośrednio z zapytań, to warto rozważyć wykorzystanie ich jako elementu zadania *Execute SQL*. Kolejną ważną cechą jest możliwość ustanowienia ścisłej kolejności przetwarzania zadań oraz wbudowana obsługa błędów zarówno na poziomie definiowania pojedynczego zadania, jak i później, na poziomie definiowania terminarza w postaci szeregowo przetwarzanych kroków. W pierwszym przypadku istnieje możliwość warunkowego rozwidlenia procesów, co może mieć również wpływ na wydajność.

Skorowidz

A

- Access Data Objects, 61
- Access mode, 66
- Actions, 361
- ADDCALCULATEDMEMBERS, 208, 210, 220, 283
- addytywność funkcji agregującej, 199
- ADO, 64
- ADO.NET, 64
- ADO.NET Destination, 68
- AdomdCommand, 328
- AdomdDataAdapter, 328
- AGGREGATE, 281
- AggregateFunction, 199
- AggregationPrefix, 200
- akcje, 179, 311
 - CommandLine, 183
 - Dataset, 183
 - definiowanie, 180
 - Drillthrough, 182, 183
 - Html, 183
 - Proprietary, 183
 - Report, 183
 - Rowset, 183
 - Statement, 183
 - Szukaj wartosci, 181
 - typy akcji, 183
 - URL, 183, 312
 - Wyświetl mapę, 182
 - Zglebianie, 183
- akcje dla raportów, 361
 - definicja typu akcji, 362
- aliasowanie kolumn, 64
- ALL, 52
- AllMembers, 279
- ALTER CUBE, 306
- ALTER DATABASE, 21
- analityczne przetwarzanie danych, 383
- analityczne struktury OLAP, 121
- analizy wielowymiarowe, 13
- Analysis Services, 57, 119, 121, 187
- Ancestor, 232
- ANCESTOR, 235, 248, 281
- ANCESTORS, 283
- AS, 219, 288
- ASC, 251
- ASCENDANTS, 284
- Assemblies, 37
- ASSEMBLY, 39
- ASSOCIATED_MEASURE_GROUP, 310
- atrybuty wymiaru, 192
- Attribute, 197
- AttributeAllMemberName, 196
- AttributeHierarchyDisplayFolder, 192
- AttributeHierarchyEnabled, 192
- AttributeHierarchyOptimizedState, 192
- AttributeHierarchyOrdered, 193
- AttributeHierarchyVisible, 193
- automatyczne poprawianie błędnych wpisów, 94
- automatyczne tworzenie kluczy, 170
- autoryzacja, 125
- AverageOfChildren, 199
- AVG, 51, 242, 258, 262, 281
- AXIS, 284

B

- Back Up, 10
- BACK_COLOR, 233
- Banach Stefan, 48
- BASC, 252
- BATCHSIZE, 113
- baza danych, 9

BDESC, 253, 271
 biblioteki CLR, 40
 błędne wpisy w tabelach relacyjnych, 94
 błędy, 198
 błędy zapisu, 47
 BOTTOMCOUNT, 284
 BOTTOMPERCENT, 284
 BOTTOMSUM, 284
 Browser, 151
 BULK INSERT, 113, 115
 Business Intelligence, 57
 rodzaje projektów, 57
 tworzenie projektu, 57
 Button, 325
 ByAccount, 199

C

C#, 42, 326
 CACHE, 64
 CALCULATE, 173
 CALCULATIONCURRENTPASS, 281
 CALCULATIONPASSVALUE, 281
 Cancel, 77
 CancelEvent, 76
 CAPTION, 312
 Caption is MDX, 180
 Cardinality, 197
 CASE, 307
 CELL CALCULATION, 314
 CELL PROPERTIES, 233
 CHECK, 89
 CHECK_CONSTRAINTS, 114
 Children, 229, 235, 279
 CLOSINGPERIOD, 240, 281
 CLR, 40
 CLUSTERED, 28
 COALESCEEMPTY, 261, 281
 CODEPAGE, 114
 Collation, 196, 200
 COM Components, 325
 CommandText, 328
 COMPUTE, 32
 Connection, 328
 Connection Manager, 59, 66, 87, 122
 Connection Project, 58
 Connection Properties, 369
 connection string, 46, 122
 ConnectionString, 327
 Containers, 72
 Control Flow, 59
 Control Flows Items, 59
 Copy Database, 10
 CORRELATION, 281

COUNT, 50, 55, 199, 258, 279, 282
 COUSIN, 281
 COVARIANCE, 282
 COVARIANCEN, 282
 CREATE ACTION, 311, 312
 CREATE CELL CALCULATION, 315
 CREATE GLOBAL CUBE, 309
 CREATE KPI, 310
 CREATE MEMBER, 306
 CREATE SET, 309
 CREATE SUBCUBE, 288
 CREATE TABLE, 110
 CreationDate, 76
 CreationName, 76
 CreatorComputerName, 76
 CreatorName, 76
 CROSSJOIN, 192, 212, 215, 267, 269, 284
 CUBE, 20, 21, 25, 26
 Cubes, 141
 Current, 280
 CurrentMember, 225, 260, 263, 279
 CurrentOrdinal, 279
 CurrentStorageMode, 196
 CustomRollupColumn, 193
 CustomRollupPropertiesColumn, 193

D

dane, 48
 Data Connections, 61
 Data Flow, 59, 60, 61, 86, 94, 95
 Data Flow Destinations, 60
 Data Flow Sources, 60
 Data Flow Task, 59
 Data Flow Transformations, 60
 Data Marts, 51
 Data Mining, 157, 384
 Data Set, 365
 Data Source, 121
 DATAFILETYPE, 114
 DataGridView, 325
 DataMember, 279
 DataSet, 328
 DataType, 199
 daty, 47
 DDL, 288
 DEALLOCATE, 36
 DefaultMeasure, 200
 DefaultMember, 193, 279
 definicja kostki, 155
 definicja KPIs, 177, 178, 179
 definicja łańcucha połączeniowego, 63, 126
 definicja mapowania kolumn, 68
 definicja parametru wielowartościowego, 379

- definicja połączenia z bazą, 8, 125
- definicja pól obliczanych dla raportu, 365
- definicja przekierowania błędnych danych z kontrolki reprezentującej źródło, 69
- definicja samozłączenia, 53
- definicja skryptu dla zadania Execute SQL, 96
- definicja właściwości zadania przetwarzania procesu, 81
- definicja zadania Job, 105
- definicja źródła danych ADO, 65
- definicja źródła danych typu plik Excela, 87
- definiowanie
 - akcje, 180
 - akcje dla raportów, 361
 - automatyczne tworzenie kluczy, 170
 - dostęp do atrybutów wymiarów, 186, 187
 - dostęp do definicji hurtowni, 185
 - dostęp do komórek kostki, 186
 - dostęp do kostek, 185
 - dostęp do wymiarów, 186
 - elementy składowe przepływu, 96
 - elementy wymiaru czasu, 152
 - grupy hierarchiczne, 365
 - hurtownia danych jako zewnętrzne źródło danych dla MS Excel, 318
 - mapowanie kolumn, 64
 - miary, 172
 - miary ad hoc, 218
 - nazwane zapytania, 188
 - operacje na zbiorach atrybutów, 270
 - połączenia, 62, 123
 - połączenia dla wymiaru w postaci relacji referencyjnej, 162
 - role, 184
 - sposób zachowania się kontrolki w przypadku pojawienia się wyjątku, 66
 - tabela docelowa, 66
 - typ kalendarza, 153
 - wymiary, 51
 - wymiary ad hoc, 232
 - wymiary czasu, 150
 - wyrazenie filtrujące, 351
 - zaawansowane elementy kostki, 173
 - zbiory atrybutów ad hoc, 225
 - zewnętrzne źródło danych dla MS Excel, 318
 - złożona struktura dla wymiaru czasu, 267
 - zmienne, 73
 - źródło danych ADO, 64
- DELETE FROM, 169
- DENSE_RANK, 30
- DependsOnDimension, 196
- DESC, 252, 271
- DESCENDANTS, 209, 235, 236, 258, 284
- Description, 193, 196, 199
- DESCRIPTION, 312
- Destination Column, 68
- Detach, 10
- diagram perspektywy źródła danych, 147, 148
- diagram schematu relacyjnego bazy danych, 11
- Dimension, 278
- Dimensions, 278
- DIMENSIONS, 54
- Dimensions.Count, 279
- DiscretizationBucketCount, 193
- DiscretizationMethod, 193
- DISPLAY_FOLDER, 310
- DisplayFolder, 199
- DISTINCT, 136, 285, 360
- DistinctCount, 199
- DISTINCTCOUNT, 282
- DMX SQL, 13
- dodawanie do perspektywy źródła danych dodatkowych źródeł danych, 146
- dodawanie kontrolki do panelu narzędziowego Toolbox, 324
- dodawanie miary do kostki, 171
- dodawanie pól do źródła danych raportu, 350
- dodawanie przepływu do definicji pakietu, 60
- dodawanie relacji dla wymiaru, 156
- dodawanie tabeli do definicji wymiaru, 138
- dodawanie wymiaru do definicji kostki, 173
- dołączanie bazy danych do serwera, 9
- dostęp do definicji hurtowni, 185
- drażnienie danych, 52, 312
- DRILLDOWNLEVEL, 285
- DRILLDOWNLEVELBOTTOM, 285
- DRILLDOWNLEVELTOP, 285
- DRILLDOWNMEMBER, 285
- DRILLDOWNMEMBERBOTTOM, 285
- DRILLDOWNMEMBERTOP, 285
- DRILLTHROUGH, 312
- DRILLUPLEVEL, 285
- DRILLUPMEMBER, 285
- DROP, 288, 289
- DROP CELL CALCULATION, 315
- DROP MEMBER, 307
- DROP SUBCUBE, 289, 294
- drzewiasta struktura wymiaru rodzic — potomek, 54
- Dts.Variables(), 78
- Duplicate key, 198
- dynamiczne przetwarzanie zapytań, 36

E

- edycja połączenia z plikiem Excela, 88
- edycja źródła danych, 127, 128
- edytor wyrażień, 93

elementy hurtowni danych, 48
 eliminacja wartości NULL, 260
 ERROR, 287
 Error log path, 198
 ErrorCode, 77
 ErrorConfiguration, 196, 200
 ErrorDescription, 77
 ERRORFILE, 113
 EstimatedCount, 193
 EstimatedRows, 200
 ETL, 46, 47
 Evaluate Expression, 92
 Excel, 87, 317

- definicja układu tabeli przestawnej, 320
- definiowanie połączenia, 318
- definiowanie zewnętrznego źródła danych, 318
- kreator raportu przestawnego, 323
- kreator wykresów przestawnych, 322
- opcje tabeli przestawnej, 321
- tabele przestawne, 321
- wybór hurtowni danych, 319
- wybór kostki dla zdefiniowanego połączenia z hurtownią danych, 319
- wybór serwera hurtowni danych, 319
- wykresy przestawne, 322

 EXCEL, 64
 EXCEPT, 232, 285
 EXCLUDEEMPTY, 258
 Execute SQL, 94, 95, 117
 Execute SQL Task, 169
 ExecutionInstanceGUID, 76
 ExecutionStatus, 77
 ExecutionValue, 77
 EXISTS, 285
 Export Data, 10
 EXTRACT, 285
 Extract, Transform, Load, 46

F

Fact, 157
 FIELDTERMINATOR, 113
 FILE, 64
 File Enumerator, 84
 FILTER, 213, 243, 245, 247, 248, 268, 271, 272, 285
 filtrowanie w zapytaniach MDX, 246
 FIRE_TRIGGERS, 114
 Firstchild, 199, 225, 226, 227, 228, 279
 FirstNonEmpty, 199
 FIRSTROW, 113
 FirstSibling, 279
 FLATFILE, 64

Flow Task, 89
 FONT_FLAGS, 233
 FONT_NAME, 233
 FONT_SIZE, 233
 FOR, 314
 For Loop, 72, 73, 74

- inkrementacja pętli, 74
- licznik, 74
- Script, 74
- zadania przetwarzania procesu, 80

 FORE_COLOR, 233
 Foreach ADO, 85
 Foreach ADO.NET Schema Rowset, 85
 Foreach File, 85
 Foreach From Variable, 85
 Foreach Item, 85
 Foreach Loop, 73, 83

- definicja ograniczeń, 91
- definicja typu mapowania zmiennych, 85
- definicja typu przeglądanych elementów, 84
- definicja zmiennych dla pakietu z zagnieżdżonymi kontenerami, 90
- kody typów obiektów, 85
- zadanie Data Flow, 86

 Foreach Nodelist, 85
 Foreach SMO, 85
 FORMAT_STRING, 233
 FORMATFILE, 114
 formatowanie warunkowe, 233
 FormatString, 199
 FTP, 64
 funkcje agregujące, 14, 50, 262

- funkcje agregujące definiujące miary, 199
- funkcje agregujące zdefiniowane przez użytkownika, 35

 funkcje analityczne, 273
 funkcje języka MDX, 280
 funkcje rankingowe, 31
 funkcje statystyczne, 273
 Fuzzy Grouping, 102, 103, 104

- definicja kontrolki, 102

 Fuzzy Lookup, 95, 96, 100

- definicja tabeli słownikowej, 97
- definiowanie elementów składowych przepływu, 96
- definiowanie kolumn tabeli źródłowej, 97
- mapowanie kolumn tabeli docelowej, 99
- zawansowane opcje, 98

G

Gauge, 374
 GENERATE, 225, 234, 285
 Generate Scripts, 10

generowanie
 kostka, 171
 plan wykonania zapytania, 16
 schemat z poziomu struktury wymiaru, 169
 tabela po stronie relacyjnej, 168

GLOBAL CUBE, 309

GOAL, 310

Goal Expression, 177, 179

Google Maps, 364

GROUP BY, 14, 18

GROUPING SETS, 22, 23, 26

GroupingBehavior, 193

grupa miar, 172

grupowanie, 18

grupowanie nad oknem logicznym, 28

grupowanie rozmyte, 102, 103, 104

grupy hierarchiczne, 365

gwiazda, 55, 56

H

HEAD, 285

HIDDEN, 290

hierarchia rodzic — potomek, 53

hierarchiczna struktura wymiaru o dwóch poziomach, 52

HIERARCHIZE, 285

Hierarchy, 275, 278

HOLAP, 48

HTTP, 64

hurtownia danych, 43, 384
 elementy, 48
 proces przetwarzania, 43
 struktura, 43
 wymiary, 49, 51
 zastosowanie, 44
 zawartość pojedynczej komórki, 51

Hybrid OnLine Analytical Processing, 48

I

ID, 196

Ignore errors count, 198

IIF, 233, 261, 282

indeksy CLUSTERED, 28

INSERT, 110, 168

INSERTED, 110

InstanceSelection, 193

INSTR, 259

instrukcje warunkowe, 260

integracja danych, 46, 57

Integration Services, 57, 66

InteractiveMode, 76

INTERSECT, 286

INVOCATION, 312

IsAggregatable, 193

ISANCESTOR, 280

ISEMPTY, 260, 261, 280

ISGENERATION, 280

ISLEAF, 280

ISSIBLING, 281

Item, 279, 280

J

JDBC, 46

jeden do wielu, 53

język C#, 326

Job, 105

JOIN, 17, 18, 22, 32

K

kalendarze, 151, 152

KEEPIDENTITY, 114

KEEPNULLS, 114

Key error action, 198

Key not found, 198

KeyColumn, 194

KILOBYTES_PER_BATCH, 114

klucz główny wymiaru, 164

klucz obcy, 55, 120

kluczowe wskaźniki wydajności, 177

kod Transact-SQL, 101

kodowanie zmiennych znakowych, 169

kolory okna poleceń systemowych, 82

komórki, 51

kompaktowanie bazy danych, 101

konfiguracja Reporting Services, 339

kontenery, 72
 For Loop, 72, 73, 74
 Foreach Loop, 73, 83
 Sequence, 73

korelacja, 274

kostka, 54, 141, 154, 159
 akcje, 179
 definiowanie zaawansowanych elementów, 173
 dodawanie wymiaru, 173
 elementy składowe w procesie przetwarzania, 145
 GLOBAL CUBE, 309
 kluczowe wskaźniki wydajności, 177
 KPIs, 177
 metody tworzenia, 142
 miary, 142
 miary kalkulowane, 174

kostka
 nazwy kostek, 144
 szablony, 171
 tabela faktów, 142
 właściwości, 200
 wybór miar, 142
 wybór wymiaru, 143
 wymiary, 143
 kostka o strukturze płatka śniegu, 160
 tworzenie, 161
 kowariancja populacji, 274
 kowariancja próbki, 274
 KPIs, 177, 178
 kreator definiowania właściwości obiektu
 źródła danych ADO, 62
 kreator połączenia, 61, 122

L

LAG, 265, 279
 Language, 196
 LastChild, 199, 227, 279
 LastNonEmpty, 199
 LASTPERIODS, 286
 LASTROW, 114
 LastSibling, 279
 LEAD, 237, 279
 LEAVES, 287
 Level, 275, 278
 Levels, 279
 Levels.Count, 279
 liczebność elementów, 50
 LINKED SERVER, 116
 LINKMEMBER, 281
 LINREGINTERCEPT, 282
 LINREGPOINT, 282
 LINREGR2, 282
 LINREGSLOPE, 282
 LINREGVARIANCE, 282
 lista parametrów typu ReadWrite, 79
 LocaleId, 76, 77
 LOOKUPCUBE, 282

Ł

ładowanie danych do schematu, 48
 łańcuch połączeniowy, 46, 63, 122

M

MachineName, 76
 Maintenance Plan Tasks, 59
 Many-to-Many, 157

mapowanie kolumn, 64
 MAX, 199, 282
 MAXERRORS, 113
 MAXROWS, 313
 MDX, 121, 179, 203
 MDX SQL, 13, 44, 203
 MdxMissingMemberMode, 196
 MeasureExpression, 199
 MEASURES, 54
 MEDIAN, 282
 MEMBER, 219
 MemberNamesUnique, 194
 Members, 277, 279, 281
 MembersWithData, 194
 MembersWithDataCaption, 194
 MEMBERTOSTR, 287, 311
 menedżer połączeń, 123
 MERGE, 109, 110, 112
 Merge Join, 170
 MessageBox, 91, 329
 metody języka MDX, 278
 miary, 54
 miary ad hoc, 218
 miary kalkulowane, 174, 176
 właściwości, 199
 migracja danych, 59, 109
 MIN, 199, 283
 MiningModelID, 196
 model płatka, 55, 160
 modyfikacja struktury hurtowni danych, 146
 MOLAP, 48
 MONTH, 365
 most dostępu do danych, 46
 MS Business Intelligence, 317
 MS Management Studio, 167
 MS SQL Agent, 104
 MS SQL Server, 47
 MS SQL Server 2008, 7
 MsgBox, 78
 MSMQ, 64
 MSOLAP100, 64
 MTD, 223, 245, 286
 MultiDimensional eXtension, 44
 Multidimensional OnLine Analytical
 Processing, 48
 MULTIFILE, 64
 MULTIFLATFILE, 64

N

Name, 280
 NameColumn, 194
 NAMETOSET, 286
 NamingTemplate, 194

nazwane zapytania, 188
 tabele źródłowe, 188
 tworzenie, 189
 nazwy baz danych, 125
 nazwy kostek, 144
 nazwy logiczne połączenia, 126
 nazwy symboliczne, 125
 New Database, 9
 New Dimension, 132
 New Measure Group, 172
 New Named Query, 146, 187
 NextMember, 236, 263, 279
 nietrwale struktury wielowymiarowe, 287
 No Relationship, 157
 NON VISUAL, 289
 None, 199
 NONEMPTY, 207, 212, 227, 233, 246, 249, 269, 270
 NONEMPTYCROSSJOIN, 286
 NOTEEMPTY, 243
 NTILE, 30
 NULL, 36, 53, 110, 260
 Null key converted to unknown, 198
 Null key not allowed, 198
 Number of errors, 198
 numeracja wierszy, 28

O

obiekty ASSEMBLY, 39
 Object Explorer, 9, 204
 ODBC, 46, 64
 OfflineMode, 76
 okna logiczne, 30
 określanie terminarzy wykonania zadania, 107
 OLAP, 45, 46, 201
 OLE DB, 46, 66, 123
 OLEDB, 64
 OLEDB connection manager, 86
 OLTP, 45
 On error action, 198
 On Line Analytical Processing, 45
 On Line Transactional Processing, 45
 OPENINGPERIOD, 239, 281
 OPENROWSET, 115
 operacje na zbiorach atrybutów, 212, 270
 optymalizacja zapytań, 15, 17
 ORACLE, 64
 ORDER, 114, 251, 253, 286
 ORDER BY, 28, 33
 OrderBy, 194
 OrderByAttribute, 194
 ORDINAL, 279, 283
 OVER, 28, 39
 OverwriteDataSources, 337

P

PackageID, 76
 PackageName, 76
 pakiety integracyjne, 104
 pamięć, 43
 panel wielozakładowy, 325
 panel zapytań MDX, 204
 PARALLELPERIOD, 238, 281
 Parent, 231, 232, 279
 PARTITION BY, 30, 32, 39
 partycje logiczne, 28
 pełna nazwa kwalifikowana, 41
 PercentComplete, 77
 PERIODSTATE, 243
 PERIODSTODATE, 242, 244, 286
 perspektywa źródła danych, 127, 128, 129
 nazwa logiczna perspektywy, 131
 tworzenie, 128
 widok struktury, 131
 właściwości połączenia, 129
 wybór tabel, 130
 perspektywy, 119, 120
 definiowanie elementów hurtowni, 188
 plan wykonania zapytania, 16
 pliki bazy danych, 10
 płatek śniegu, 56, 160
 podkostka, 288
 podraporty, 354
 podsumowania, 32
 podział na okna logiczne, 30
 Pointer, 66
 pole tekstowe, 325
 połączenie z bazą, 8
 połączenie z plikiem Excela, 88, 92
 porównania rozmyte, 100
 poziomy definiowania miar i wymiarów
 w hurtowni danych, 54
 PREDICT, 283
 PrevMember, 236, 260, 263, 279
 ProactiveCaching, 196, 200
 proces integracji danych, 76
 proces przetwarzania, 43
 ProcessingGroup, 196
 ProcessingMode, 196, 200
 ProcessingPriority, 196, 200
 Product Template, 163, 164
 ProgressCountHigh, 77
 ProgressCountLow, 77
 ProgressDescription, 77
 projekt Analysis Services, 121
 projekt Business Intelligence, 57
 projekt hurtowni danych, 122
 Propagate, 77

- Properties, 280
 - przeciągnij i upuść, 174
 - przekroje, 52
 - przepływ, 60, 89
 - przeszukiwanie elementów zdefiniowanej kolekcji obiektów, 83
 - przetwarzanie, 43
 - przetwarzanie analityczne, 43, 45, 384
 - przetwarzanie struktur mieszanych, 48
 - przetwarzanie transakcyjne, 45
 - przetwarzanie wymiaru, 137
 - przycisk polecenia, 325
 - przypisanie użytkowników do roli, 184
 - puste transakcje, 47
- Q**
- QTD, 223, 245, 286
 - Query Builder, 332, 333, 370
 - Query Designer, 349
- R**
- RANK, 30, 283
 - raportowanie, 317
 - akcje, 361
 - Excel, 317
 - grupy hierarchiczne, 365
 - języki wyższego rzędu, 324
 - Reporting Services, 331
 - synchronizowanie raportów, 346
 - raportowanie dla danych pochodzących z hurtowni, 369
 - definicja pola sterującego wskazaniem wskaźnika Gauge, 375
 - filtrowanie, 373
 - KPI, 374
 - rozmieszczenie pól zwracanych przez zapytanie, 372
 - tworzenie połączenia z hurtownią, 370
 - tworzenie zapytania, 370
 - właściwości skali dla wskaźnika Gauge, 375
 - wskaźniki, 374
 - wybór kostki źródłowej, 371
 - raporty dla hurtowni danych, 317
 - raporty o strukturze macierzowej, 356
 - definicja rozmieszczenia pól, 357
 - definicja właściwości parametru pobierającego dane z zapytania, 359
 - dodawanie źródła danych, 358
 - widok projektu, 357
 - ReadWrite, 79
 - Referenced, 157
 - Regular, 157
 - relacje, 157, 197
 - relacje rodzic — potomek, 52
 - Relational OnLine Analytical Processing, 48
 - RelationshipType, 197
 - Reporting Services, 58, 331
 - definicje pól dla wykresu trójwymiarowego, 377
 - definicje serwera WWW, 341
 - definiowanie właściwości raportu, 337, 338
 - definiowanie wyrażenia filtrującego, 351
 - dodawanie pól do źródła danych raportu, 350
 - filtrowanie wielowartościowe, 380
 - katalog wirtualny, 344
 - konfiguracja, 339
 - konfiguracja serwera WWW, 339
 - nagłówek raportu, 380
 - nazwa raportu, 338
 - parametry wielowartościowe, 379
 - podraporty, 354
 - Preview Report, 336
 - przypisanie do kontrolki akcji przeniesienia do zakładki, 378
 - Report Builder, 345
 - repozytorium, 342, 343
 - serwer WWW, 339
 - sposób odświeżania danych, 349
 - sposób uwierzytelnienia, 341
 - stopka raportu, 380
 - synchronizowanie raportów, 346
 - szata graficzna raportu, 336
 - testowanie serwera WWW, 342
 - tryb autoryzacji, 341
 - tworzenie aplikacji raportującej, 331
 - tworzenie połączenia ze źródłem danych, 332
 - tworzenie raportu, 332
 - tworzenie źródła danych, 331
 - tworzenie źródłowego zapytania dla raportu, 332
 - wizualna prezentacja wyników, 334
 - wskaźniki, 374
 - wybór graficznej postaci raportu, 335
 - wykresy, 377
 - zmiana właściwości raportu, 349
 - źródło danych, 331
- Restore, 10
 - RGB, 233
 - rodzaje kalendarzy, 152
 - rodzaje relacji, 157
 - ROLAP, 48
 - role, 184
 - role o niepełnych uprawnieniach, 185
 - ROLLUP, 20, 21, 23, 24, 25
 - ROLLUPCHILDREN, 283

ROOT, 52, 287
RootMemberIf, 194, 195
ROW_NUMBER, 28, 30
ROWS_PER_BATCH, 114
ROWTERMINATOR, 113
rzeczywisty wymiar czasu, 151

S

samołączenie, 53, 54
SAPBI, 64
Scheduler, 106
Script, 74
ScriptCacheProcessingMode, 200
ScriptErrorHandlingMode, 200
SELECT, 13, 14, 17, 136
Sequence, 73
Serializable, 37
SESSION CUBE, 290
sesyjne zbiory atrybutów, 309
SET, 225
SETTOARRAY, 280
SETTOSTR, 287
Shrink, 10
Shrink Database, 101
Siblings, 276, 279
Similarity threshold, 98
składnia nazewnictwa tabeli, 166
składnice danych, 51
skoroszyt Excela, 87
skrypty SQL, 10
SMOServer, 64
SMTP, 64
Solution Explorer, 126, 184
sortowanie, 251
Source, 196, 199
SourceDescription, 77
SourceID, 77
SourceName, 77
sposób obliczania miary, 174
sprawdzanie zgodności danych ze słownikiem, 94
SQL, 13, 44
SQL Server Destination, 88, 89
SQL Server Management Studio, 7
SqlClient Data Provider, 62
sqlcmd, 82
SQLMOBILE, 64
StartTime, 76
STATEMENT, 312
STATUS, 310
Status Expression, 177, 179
statystyka klienta, 17
STDEV, 283

STDEVP, 283
sterowniki dostępu do danych, 46, 64, 124
Stop on error, 198
stopień izolacji transakcji, 37
StorageLocation, 200
StorageMode, 196, 200
stosowanie zapytań SQL do migracji danych, 109
STRIPCALCULATEDMEMBERS, 286
strojenie bazy danych, 17
STRTOMEMBER, 281
STRTOSET, 286
STRTOTUPLE, 287
STRTOVALUE, 283
struktura hurtowni danych, 43, 49, 132
 modyfikacja, 146
struktura uprawnień do korzystania z hurtowni
 danych, 184
SUBSET, 286
SUM, 50, 199, 232, 242, 258, 283
suma, 50
synchronizacja danych pochodzących z różnych
 źródeł, 48
synchronizacja raportów, 346
system transakcyjny, 45
szablony, 164
szablony kostki, 171
szablony wymiaru, 163
szacowany plan wykonania zapytania, 16

Ś

średnia, 51

T

TabControl, 325
tabela faktów, 142, 144
tabele pośrednie, 163
tabele powiązane, 138
tabele przestawne, 321
TABLOCK, 114
Tabular Data Stream Protocol, 17
TAIL, 286
TargetDataSourceFolder, 337
TargetReportFolder, 337
TargetServerURL, 337
TaskID, 76
TaskName, 76
TaskTransactionOption, 76
TDS, 17
tematyczne hurtownie danych, 51
TERADATA, 64
terminarz wykonania zadania, 107

testowanie
 nietrwale struktury wielowymiarowe, 287
 wskaźniki wydajności, 178
 TextBox, 325
 This, 280
 TOGGLEDRILLSTATE, 286
 Toolbox, 59, 324, 325
 TOPCOUNT, 176, 254, 256, 272, 286
 TOPPERCENT, 255, 256, 257, 286
 TOPSUM, 258, 286
 ToString(), 78
 Transact-SQL, 13, 37, 101
 TREND, 310
 Trend Expression, 178, 179
 trend zmian, 178
 TRUNCATE, 95
 TRUNCATE TABLE, 95, 169
 tuning bazy danych, 17
 TUPLETOSTR, 287
 tworzenie
 analityczne struktury OLAP, 121
 baza danych, 9
 hurtownia danych, 119
 hurtownia danych z zastosowaniem
 tabel pośrednich, 163
 kostka, 141, 154
 kostka o strukturze płatka śniegu, 161
 kostka sesyjna, 290
 miary ad hoc, 218
 miary kalkulowane, 174, 175
 nazwane zapytania, 188
 nietrwale struktury wielowymiarowe, 287
 perspektywa źródła danych po stronie
 hurtowni, 128
 podkostka, 288
 projekt, 121
 projekty Business Intelligence, 57
 raporty dla hurtowni danych, 317
 struktura hurtowni danych, 132
 wymiary, 132, 170
 wyrażenia opisujące miary kalkulowane, 174
 źródło danych, 121, 122
 Type, 195

U

UnaryOperatorColumn, 195
 UNION, 270, 287
 UniqueName, 280
 UnknownMember, 196, 279
 UnknownMemberName, 196
 UNORDER, 287
 UPDATE CUBE, 292, 294, 295
 USE_EQUAL_ALLOCATION, 298
 USE_EQUAL_INCREMENT, 298

USE_WEIGHTED_ALLOCATION, 298
 USE_WEIGHTED_INCREMENT, 298
 uprawnienia do korzystania z hurtowni danych,
 184
 uruchamianie pakietów integracyjnych, 104
 Usage, 195
 USE_EQUAL_ALLOCATION, 297, 298
 USE_EQUAL_INCREMENT, 297, 298
 USE_WEIGHTED_ALLOCATION, 297, 298
 USE_WEIGHTED_INCREMENT, 298, 302
 UserName, 76
 USERNAME, 287
 USING, 109
 ustanawianie klucza wymiaru, 165
 usterki przetwarzania, 47
 usuwanie definicji podkostki, 294
 uwiaryzelnianie, 7

V

VALIDMEASURE, 281
 Value Expression, 179
 ValueColumn, 195
 VAR, 283
 VariableDescription, 77
 VariableID, 77
 Variables, 73
 VARIANCE, 283
 VARIANCEP, 283
 VARP, 283
 VB, 42
 VersionBuild, 76
 VersionComment, 76
 VersionGUID, 76
 VersionMajor, 76
 VersionMinor, 76
 View T-SQL, 101
 Visible, 199, 200
 VISUALTOTALS, 287

W

wartości NULL, 260
 WHEN MATCHED, 110
 WHEN NO MATCHED, 110
 WHERE, 215, 247, 264, 288
 WITH, 19, 219
 WITH CUBE, 21
 WITH ROLLUP, 20
 właściwości atrybutów wymiaru, 192
 właściwości kostki, 200
 właściwości miary, 199
 właściwości relacji, 197

właściwości wymiaru, 196
 właściwości zadania przetwarzania procesu, 81
 WMI, 64
 WriteEnabled, 196
 współczynniki wydajności KPI, 310
 WTD, 223, 245, 287
 wybór łańcucha połączeniowego, 63
 wybór rodzaju relacji, 157
 wybór źródła danych, 62
 wykonywanie zapytania SQL, 169
 wykresy, 377
 wykresy przestawne, 322
 wymiar czasu, 150

- definiowanie elementów, 152
- definiowanie złożonej struktury, 267
- kalendarze, 151

 wymiary, 49, 54

- atrybuty, 138, 192
- atrybuty numeryczne, 54
- atrybuty opisowe, 54
- dodawanie tabeli, 138
- edycja, 135
- kolumna klucza, 133
- kolumna nazwy, 133
- nazwa logiczna, 135
- opis procesu przetwarzania, 137
- przetwarzanie, 135
- sposoby tworzenia, 132
- stan przetwarzania, 136
- szablony, 133
- tabela, 133
- tabele powiązane, 138
- tworzenie, 132
- właściwości, 196
- wybór atrybutów, 134
- wykaz tabel, 139
- wymiary oparte na ciągłych atrybutach, 190
- wyznaczanie przedziałów i zakresów, 251

 wyrażenia MDX, 179
 wyświetlanie

- komunikaty, 80
- wiele poziomów hierarchii, 234

 wyznaczanie miar, 230

- wyznaczanie miar jako wyrażeń dla różnych elementów i poziomów hierarchii, 236

 wyznaczanie podsumowań, 34
 wyznaczanie przedziałów i zakresów dla wymiarów, 251

X

XML, 121

Y

YTD, 223, 243, 287

Z

zadania Job, 104, 105
 zadania Tasks, 72
 zapytania analityczne, 13

- COMPUTE, 32
- CUBE, 20
- elementy, 13
- funkcje agregujące, 14
- generowanie bieżącego planu wykonania zapytania, 16
- GROUP BY, 18
- GROUPING SETS, 23
- grupowanie, 18
- JOIN, 18
- numeracja wierszy, 28
- optymalizacja składni, 15
- plan wykonania zapytania, 16
- ROLLUP, 20, 24
- statystyka klienta, 17

 zapytania DDL, 288
 zapytania definiowane po stronie OLAP, 187
 zapytania MDX, 203, 329

- ADDCALCULATEDMEMBERS, 208, 210, 220
- akcje, 311
- ALTER CUBE, 306
- ANCESTOR, 235, 248
- AS, 219
- AVG, 258, 262
- BASC, 252
- BDESC, 253, 271
- CASE, 307
- CLOSINGPERIOD, 240
- COALESCEEMPTY, 261
- COUNT, 258
- CREATE KPI, 310
- CREATE MEMBER, 306
- CREATE SET, 309
- CREATE SUBCUBE, 288
- CROSSJOIN, 212, 215, 269
- definiowanie miar ad hoc, 218
- definiowanie operacji na zbiorach atrybutów, 270
- definiowanie zbiorów atrybutów ad hoc, 225
- definiowanie złożonej struktury dla wymiaru czasu, 267
- DESC, 271

zapytania MDX

DESCENDANTS, 209, 235, 236, 258
 drażnienie danych, 312
 DROP, 289
 DROP MEMBER, 307
 DROP SUBCUBE, 294
 eliminacja wartości NULL, 260
 EXCLUDEEMPTY, 258
 FILTER, 213, 243, 247, 248, 268, 271
 filtrowanie, 216, 246
 funkcje, 280
 funkcje agregujące, 262
 funkcje analityczne, 273
 funkcje statystyczne, 273
 GENERATE, 225, 234
 GLOBAL CUBE, 309
 IIF, 233, 261
 INSTR, 259
 instrukcje warunkowe, 260
 ISEMPTY, 260, 261
 kostka sesyjna, 290
 LAG, 265
 LEAD, 237
 MEMBER, 219
 nietrwale struktury wielowymiarowe, 287
 NONEMPTY, 207, 212, 227, 249, 269, 270
 ON AXIS, 206
 ON COLUMNS, 205, 206
 ON ROWS, 206
 OPENINGPERIOD, 239
 operacje na zbiorach atrybutów, 212
 ORDER, 251, 253
 PARALLELPERIOD, 238
 PERIODSTATE, 243
 PERIODSTODATE, 242, 244
 podkostka, 288
 porządkowanie malejące, 253
 porządkowanie rosnące, 252
 SESSION CUBE, 290
 sesyjne zbiory atrybutów, 309
 SET, 225
 sortowanie, 251
 SUM, 258
 TOPCOUNT, 254, 256
 TOPPERCENT, 255, 256, 257
 TOPSUM, 258
 tworzenie członka, 306

UNION, 270
 UPDATE CUBE, 292, 294, 295
 USE_EQUAL_ALLOCATION, 297, 298
 USE_WEIGHTED_ALLOCATION, 297, 298
 USE_WEIGHTED_INCREMENT, 302
 wartości null, 207, 216
 WHERE, 215, 216, 217, 247, 264
 WITH, 219
 wskazywanie elementów potomnych
 lub nadrzędnych, 230
 współczynniki wydajności KPI, 310
 wymiary, 211
 wyświetlanie miar, 208
 wyświetlanie wielu poziomów hierarchii, 234
 wyznaczanie miar, 230
 wyznaczanie miar jako wyrażeń dla różnych
 elementów i poziomów hierarchii, 236
 wyznaczanie przedziałów
 i zakresów dla wymiarów, 251
 zakres obliczeń, 223
 zapytania wybierające, 278
 zestaw atrybutów, 208, 209
 zapytania podsumowujące, 33
 zapytania SQL, 119
 zawartość pojedynczej komórki, 51
 zgłębianie danych, 13
 zgodność danych ze słownikiem, 94
 zmiana właściwości raportu, 349
 zmienne, 73
 zmienne systemowe, 74, 76

Ż

źródło danych, 61, 121
 ADO, 61
 baza danych, 125
 edycja, 127
 łańcuch połączeniowy, 122
 menedżer połączeń, 123
 nazwa logiczna połączenia, 126
 sterowniki, 124
 tryb autoryzacji, 125
 tworzenie, 122
 użytkownik, 126

HURTOWNIE DANYCH

Idea hurtowni danych ściśle wiąże się z ich kolosalnymi ilościami, gromadzonymi podczas tysięcy różnych sytuacji — przy dowolnej transakcji, w urzędzie, na lotnisku, w internecie... Nawet nasze połączenia telefoniczne są przechowywane przez operatora. Te wszystkie dane trzeba gdzieś pomieścić, sensownie posegregować i zapewnić sobie możliwość sięgnięcia do wybranego ich zakresu bez długotrwałych poszukiwań. Taką możliwość dają właśnie hurtownie danych — przemyślane, bardzo pojemne bazy, oferujące zarówno integrację wprowadzanych danych, jak i znakomite mechanizmy ich przeszukiwania. Jeśli chcesz poszerzyć swoją wiedzę na temat tworzenia i przeglądania zawartości hurtowni danych, trafiles pod właściwy adres!

Książka „Hurtownie danych. Od przetwarzania analitycznego do raportowania” zawiera materiał przeznaczony nie tylko dla studentów wydziałów informatycznych, ale także dla pasjonatów tej tematyki oraz specjalistów zainteresowanych poszerzeniem wiedzy. W możliwie najprostszy, praktyczny sposób opisano w niej składnię i postać zapytań analitycznych, strukturę hurtowni danych oraz kwestię ich integracji i wizualnego tworzenia elementów hurtowni. Znajdziesz tu także omówienie analizy danych z wykorzystaniem rozszerzenia MDX SQL oraz zastosowań raportowania. Zapoznanie się z tymi informacjami oraz przesiedlenie zgromadzonych tu przykładów pozwoli Ci zrozumieć problemy powstające przy budowie hurtowni danych i wykrywać tę wiedzę we własnych projektach.

- Zapytania analityczne
- Struktura hurtowni danych
- Integracja danych
- Wizualne tworzenie elementów hurtowni danych
- Analiza danych z wykorzystaniem rozszerzenia MDX SQL
- Raportowanie

Od bazy do hurtowni danych... Skocz na głęboką wodę!

nr katalogowy: 5804



Księgarnia internetowa:
<http://helion.pl>



Zamówienia telefoniczne:
0 801 339900
0 601 339900



Helion

Sprawdź najnowsze promocje:
☞ <http://helion.pl/promocje>
Książki mechanicznie czytane:
☞ <http://helion.pl/ksiazki>
Zamów informacje o nowościach:
☞ <http://helion.pl/newsy>

Helion SA
ul. Kołłątaja 1c, 44-100 Gliwice
tel.: 32 230 98 43
e-mail: helion@helion.pl
<http://helion.pl>

helion.pl
księgarnia
internetowa

Cena: 69,00 zł

ISBN 978-83-246-2977-0



9 788324 629770

Informatyka w najlepszym wydaniu