

DANE PRZESTRZENNE W BAZACH DANYCH SYSTEMU MICROSOFT SQL SERVER 2008 R2 – WPROWADZENIE

Streszczenie

Bazy relacyjne coraz częściej zawierają dane, które nie pasują do koncepcji relacyjności. Decydując się na implementację w swych produktach tych nierelacyjnych typów danych producenci systemów zarządzania bazami danych kierują się głównie chęcią skrócenia i uproszczenia procesów tworzenia aplikacji wykorzystujących bazy danych. Niniejszy artykuł wprowadza w tematykę danych przestrzennych dostępnych w systemie Microsoft SQL Server 2008 R2. Owe typy danych stanowią doskonały przykład implementacji nierelacyjnych typów danych w relacyjnych bazach danych.

Abstract

Today the relational databases contain data that do not fit the relational concept more often than several years ago. By implementation of non-relational data types in their products manufacturers of database management systems wish to shorten and simplify the application development processes. This article introduces the spatial data types available in Microsoft SQL Server 2008 R2. These data types are an excellent example of the implementation of non-relational data types in relational databases.

1. WSTĘP

System zarządzania bazami danych Microsoft SQL Server (w tym jego najnowsza wersja – SQL Server 2008 R2) jest, obok Oracle i IBM DB2, jednym z trzech najczęściej wykorzystywanych tej klasy systemów na świecie. W dzisiejszym świecie systemom takim stawia się ogromne wymagania. Jednym z oczekiwań użytkowników jest możliwość przechowywania danych nierelacyjnych, czyli złożonych danych niespełniających podstawowych założeń teorii relacyjności. Przyczynami, dla których takie dane są często przechowywane w relacyjnych bazach danych, jest przyspieszenie procesu tworzenia aplikacji oraz wymiana danych pomiędzy heterogenicznymi systemami.

¹ Mgr inż. Paweł Potasiński jest wykładowcą w Warszawskiej Wyższej Szkole Informatyki. Jest też autorem bloga <http://zine.net.pl/blogs/sqlgeek/about.aspx> i entuzjastą Microsoft SQL Server.

Do nierelacyjnych danych zaliczają się dane przestrzenne, czyli dane geometryczne i geograficzne.

2. ZASTOSOWANIA DANYCH PRZESTRZENNYCH

Zaimplementowanie w systemie zarządzania bazami danych typów danych przestrzennych powoduje, że użytkownik może w dość łatwo rozwiązywać problemy dotyczące danych przestrzennych. Przykłady takich problemów:

1. Problem komiwożacza, czyli problem znalezienia najkrótszej trasy z jednej miejscowości do drugiej z uwzględnieniem dostępnych tras.
2. Problem znalezienia wszystkich banków w promieniu 10 km od centrum miasta.
3. Problem śledzenia położenia pojazdów firmy kurierskiej oraz określania przybliżonego czasu dotarcia przesyłki do adresata.

3. DANE GEOMETRYCZNE

Dane geometryczne w SQL Server 2008 R2 są przechowywane jako kolumny/zmienne typu **geometry**. Są to dane reprezentowane w płaskim euklidesowym układzie współrzędnych (w układzie „płaskiej ziemi”).

Typ ten powinien być stosowany do opisywania obiektów, dla których krzywizna Ziemi oraz jednostka nie mają znaczenia. W szczególności mogą to być niewielkie obiekty (np. działki) lub kształty geometryczne (np. matryce produkcyjne).

Typ geometry nie ma jednostki miary. Zatem to użytkownik decyduje, czy długości linii będą wyrażone w metrach, centymetrach, czy może w innych jednostkach.

4. DANE GEOGRAFICZNE

Typ **geography** dostępny w SQL Server 2008 R2 umożliwia przechowywanie i obsługę danych geograficznych w układzie elipsoidalnym (w układzie „okrągłej ziemi”), w których wykorzystywane są współrzędne geograficzne – długość i szerokość (np. dane pochodzące z urządzeń GPS).

Typ ten powinien być używany w odniesieniu do tych danych, dla których krzywizna Ziemi nie powinna być pomijana.

Położenie wszystkich punktów określone jest za pomocą szerokości geograficznej (mierzonej w stopniach, minutach i sekundach, odległości punktu od bieguna południowego lub północnego) i długości geograficznej (mierzonej w stopniach, minutach i sekundach odległości punktu od południka zerowego).

Jednostka miary dla danych typu geography jest uzależniona od systemu odniesienia, w jakim dane zostały określone.

5. SYSTEMY ODNIESIENIA (SIATKI KARTOGRAFICZNE)

Każda instancja danych przestrzennych posiada **identyfikator systemu odniesienia SRID** (ang. *Spatial Reference Identifier*) zgodny ze standardem EPSG (ang. *European Petroleum Survey Group*). Identyfikator SRID określa dla danych przestrzennych system odniesienia bazujący na określonej elipsoidzie oraz jednostkę miary przypisaną do danego systemu odniesienia.

Kolumny typów geometry lub geography nie muszą przechowywać instancji danych geograficznych o tym samym identyfikatorze SRID. Jednak, aby można było wykonać jakiegokolwiek operacje na dwóch instancjach danych przestrzennych, muszą mieć one ten sam identyfikator SRID. W przeciwnym razie metody typów przestrzennych w SQL Server 2008 R2 będą zwracały wartość NULL. Jako przykład weźmy próbę obliczenia odległości między dwoma punktami, z których każdy ma zdefiniowany inny identyfikator SRID:

```
DECLARE @g1 geography, @g2 geography;
SELECT
    @g1 = geography::STGeomFromText('POINT (1 1)', 4120),
    @g2 = geography::STGeomFromText('POINT (1 2)', 4121);
SELECT @g1.STDistance(@g2);
-----
NULL
```

Dla danych typu GEOMETRY wartością identyfikatora SRID jest 0. Taka wartość SRID oznacza, że dane przestrzenne są określane w niezdefiniowanej płaskiej przestrzeni.

Dla danych typu GEOGRAPHY domyślną wartością identyfikatora SRID jest 4326. Jest to oznaczenie systemu odniesienia opracowanego w programie WGS 84 (ang. *World Geodetic System 1984*). Jeśli chcemy definiować dane w innym systemie odniesienia, musimy w definicji danych przestrzennych podać odpowiednią wartość identyfikatora SRID.

Wszystkie dostępne w SQL Server 2008 R2 wartości identyfikatora SRID (oprócz wartości 0) można przeglądać wykonując zapytania do widoku katalogowego *sys.spatial_reference_systems*.

6. RODZAJE DANYCH PRZESTRZENNYCH

Oba typy – geometry i geography – są zgodne ze specyfikacją stworzoną przez konsorcjum OGC (ang. *Open Geospatial Consortium*). W SQL Server 2008 R2 typy te zostały zaimplementowane jako zbliżone do siebie typy CLR (ang. *Common Language Runtime*, typy stworzone przy użyciu technologii .NET) i są dostarczane w bibliotekach `SqlServerSpatial.dll` oraz `Microsoft.SqlServer.Types.dll`.

Typy przestrzenne są natywnymi typami danych w SQL Server 2008 R2, zatem używanie ich nie wymaga włączania integracji serwera ze środowiskiem CLR.

Oba typy są typami abstrakcyjnymi, zaś rodzaj danych przestrzennych (punkt, wielobok, itd.) użytkownik określa dopiero podczas tworzenia obiektu geometrycznego lub geograficznego. Wspomniane rodzaje danych przestrzennych to:

- POINT – punkt w przestrzeni określony współrzędnymi X i Y (opcjonalnie Z i M).
- MULTIPOINT – uporządkowany zbiór instancji POINT niepołączonych ze sobą.
- LINESTRING – reprezentuje uporządkowany zbiór instancji POINT połączonych odcinkami.
- MULTILINESTRING – zbiór instancji LINESTRING.
- POLYGON – zbiory instancji POINT będących wierzchołkami wielokąta zewnętrznego oraz zera lub więcej wielokątów wewnętrznych, które mogą stykać się pojedynczymi punktami. W przypadku typu danych geography punkty wielokątów należy deklorować w kolejności zgodnie z kierunkiem przeciwnym do ruchu wskazówek zegara. W przeciwnym wypadku wystąpi błąd.
- MULTIPOLYGON – zbiór instancji typu POLYGON.
- GEOMETRYCOLLECTION – zbiór instancji dowolnych obiektów wymienionych powyżej.

7. FORMATY DANYCH PRZESTRZENNYCH

Dane przestrzenne w SQL Server 2008 R2 mogą być reprezentowane za pomocą jednego z następujących formatów:

- Format binarny zgodny z WKB (ang. *Well-Known Binary*) – do wstawiania danych podawanych w tym formacie służą metody o nazwach *ST[rodzaj]FromWKB*, np. *STPointFromWKB*:

```
DECLARE @g geometry;
SET @g = geometry::STPointFromWKB
    (0x010100000000000000000000008400000000000001040, 0);
SELECT @g.STAsText();
-----
POINT (3 4)
```

- Format tekstowy zgodny z WKT (ang. *Well-Known Text*) – najbardziej intuicyjny format, do wstawiania danych podawanych w tym formacie służą metody o nazwach *ST[rodzaj]FromText*, np. *STPointFromText*:

```
DECLARE @g geometry;
SET @g = geometry::STGeomFromText('POINT (3 4)', 0);
SELECT @g.ToString();
-----
POINT (3 4)
```

- Format XML zgodny z GML (ang. *Geography Markup Language*) – do wstawiania danych podawanych w tym formacie służy metoda *GeomFromGml*:

```
DECLARE @g geometry;
SET @g = geometry::GeomFromGml('<Point xmlns="http://www.
  opengis.net/gml">
  <pos>3 4</pos>
</Point>', 0);
SELECT @g.AsGml();
-----
<Point xmlns="http://www.opengis.net/gml"><pos>3 4</pos></Point>
```

8. WYBRANE METODY TYPÓW PRZESTRZENNYCH

Typy przestrzenne w SQL Server 2008 R2 są wyposażone w liczne metody, które umożliwiają wydobywanie informacji na temat instancji danych przestrzennych, a także pozwalają wykonywać operacje na tych danych.

Dla zilustrowania wybranych metod typów przestrzennych użyję następującego fragmentu kodu w języku Transact-SQL:

```
DECLARE
  @Ulica geometry,
  @Osiedle1 geometry,
  @Osiedle2 geometry;
SELECT
  @Ulica = geometry::STGeomFromText('LINestring(0 50, 50 50,
    100 100, 200 100)', 0),
  @Osiedle1 = geometry::STGeomFromText('POLYGON((50 50, 150 50,
    150 150, 100 150, 50 100, 50 50))', 0),
  @Osiedle2 = geometry::STGeomFromText('POLYGON((0 100, 50 100,
    100 150, 200 150, 200 200, 0 200, 0 100))', 0);
```

```
-- Zapytanie 1: Metoda STArea - pole powierzchni
SELECT @Osiedle1.STArea();
-----
-- 8750

-- Zapytanie 2: Metoda STIntersection - czesc wspolna
SELECT @Osiedle1.STIntersection(@Ulica).ToString();
-----
-- LINESTRING (150 100, 100 100, 50 50)

-- Zapytanie 3: Metoda STIntersects - sprawdzenie, czy istnieje
  czesc wspolna
SELECT @Osiedle1.STIntersects(@Ulica)
-----
-- 1

-- Zapytanie 4: Metoda STUnion - zlaczenie
SELECT @Osiedle1.STUnion(@Osiedle2).ToString();
-----
-----
-- POLYGON ((50 50, 150 50, 150 150, 200 150, 200 200, 0 200, 0
  100, 50 100, 50 50))

-- Zapytanie 5: Metoda STLength - dlugosc / obwod
SELECT @Ulica.STLength();
-----
-- 220.710678118655

-- Zapytanie 6: STNumPoints - liosc punktow okreslajacych
  ksztalt
SELECT @Ulica.STNumPoints();
-----
-- 4

-- Zapytanie 7: STEnvelope - najmniejszy prostokat zawierajacy
  figure
SELECT @Osiedle1.STEnvelope().ToString();
-----
-- POLYGON ((50 50, 150 50, 150 150, 50 150, 50 50))
```

W powyższym kodzie zadeklarowane zostały trzy zmienne typu geometry – jedna zmienna zawiera linię łamaną (@Ulica), a dwie figury geometryczne (@Osiedle1 i @Osiedle2).

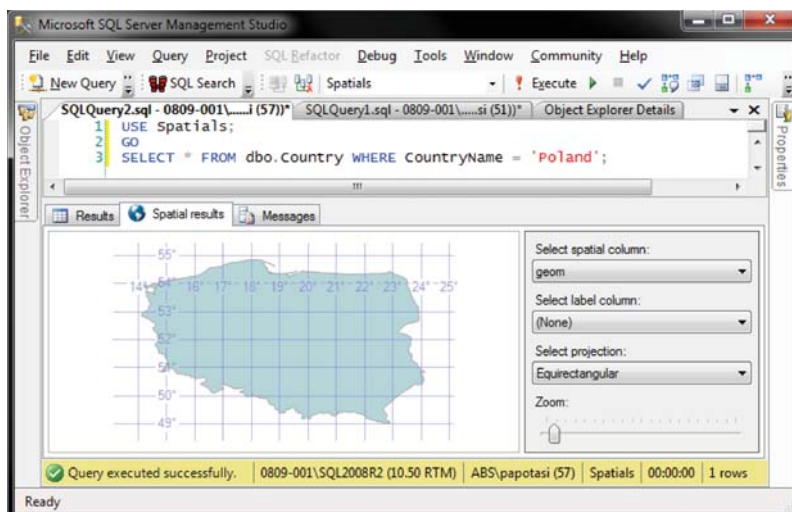
W przykładowych zapytaniach zastosowano kolejno następujące metody:

- *STArea* – zwraca pole powierzchni figur geometrycznych,
- *STIntersection* – zwraca część wspólną dwóch instancji typu przestrzennego,
- *STIntersects* – zwraca 1, jeżeli dwie instancje typu przestrzennego mają choć jeden punkt wspólny,
- *STUnion* – zwraca instancję typu przestrzennego, który jest sumą dwóch instancji,
- *STLength* – zwraca długość/obwód instancji typu przestrzennego,
- *STNumPoints* – zwraca ilość punktów, z których zdefiniowano instancję typu przestrzennego,
- *STEnvelope* – zwraca najmniejszy prostokąt zawierający wskazaną instancję typu przestrzennego.

9. WSPARCIE APLIKACJI DLA TYPÓW PRZESTRZENNYCH

Dane przestrzenne doczekały się wsparcia wielu aplikacji i technologii. Także typy przestrzenne dostępne w SQL Server 2008 R2 mogą liczyć na to, że wiele aplikacji będzie w stanie wykorzystać potencjał typów geometri i geography.

Już korzystając z narzędzia SQL Server Management Studio, które jest instalowane wraz z systemem SQL Server 2008 R2, programista otrzymuje wsparcie w postaci zakładki Spatial results, która w graficzny sposób ilustruje wyniki zapytań wykonywanych na danych przestrzennych (patrz rys. 1).



Rys. 1. Zakładka Spatial results w SQL Server Management Studio

Także technologie służące do tworzenia oprogramowania, jak na przykład .NET, oferują liczne kontrolki i mechanizmy wspierające obsługę danych geometrycznych i geograficznych (patrz rys. 2).



Rys. 2. Kontrolka mapy użyta w aplikacji stworzonej w technologii Silverlight



Rys. 3. Przykładowy raport Reporting Services z wykorzystaniem danych przestrzennych

W SQL Server 2008 R2 dane geograficzne mogą być także wykorzystane do tworzenia raportów przy użyciu platformy Reporting Services. Poniższy przykładowy raport zawiera kontrolkę mapy umożliwiającą wizualizację danych przestrzennych przechowywanych w bazie danych w systemie SQL Server 2008 R2 (na rys. 3 znajduje się raport ilustrujący liczebność ludności województw Polski w 2007 roku).

10. PODSUMOWANIE

Dane przestrzenne znajdują wiele zastosowań we współczesnych aplikacjach biznesowych. Dzięki wizualizacji tych danych raporty i aplikacje stają się bardziej intuicyjne dla użytkownika końcowego, gdyż oprócz zwykłych danych tekstowych otrzymuje on także łatwą w interpretacji postać graficzną. Przestrzenne typy danych w SQL Server 2008 R2 dają programistom ogromne możliwości na drodze do sprostanania rosnącym oczekiwaniom użytkowników dzisiejszych systemów wspierających biznes.

Literatura

1. Danuta Mendrala, Paweł Potasiński, Marcin Szeliga, Damian Widera, *Serwer SQL 2008. Administracja i programowanie*, Helion 2009
2. Alastair Aitchison, *Beginning Spatial with SQL Server 2008*, Apress 2009
3. Microsoft Books Online, <http://msdn.microsoft.com/en-us/library/ms130214.aspx>

