

*Dr inż. Jacek Markus
Warszawska Wyższa Szkoła Informatyki
jacek.markus@gmail.com*

Wykorzystanie przestrzennych typów danych do tworzenia raportów.

Using spatial data types in reports.

Streszczenie:

Do Microsoft SQL 2008 wprowadzono przestrzenne typy danych: geometryczny i geograficzny. Są dostępne narzędzia, pozwalające na odpytywanie typów przestrzennych i rysowania na ich podstawie map. Wiele map można zaimportować do tabeli Microsoft SQL 2008 (lub nowszego) z plików shp. Dla kolumn z typem danych: geography i geometry mogą być tworzone indeksy przestrzenne w celu przyspieszenia zapytań. W Microsoft SQL Server Reporting Services 2008R2 wprowadzono możliwość tworzenia raportów na mapie. Do tworzenia raportów na mapie możemy wykorzystać wbudowane mapy, mapy z plików shp lub zapytanie SQL do tabeli z kolumną z danymi przestrzennymi. W celu stworzenia wykresu na mapie musimy powiązać dane przestrzenne z danymi biznesowymi za pomocą wspólnego klucza. Jeśli dane geograficzne są z zapytania SQL, możemy w tym zapytaniu powiązać je z tabelami biznesowymi za pomocą relacji, według naturalnych kluczy, jeśli jest zdefiniowany klucz obcy, albo bazując na kolumnach zawierających ten sam typ danych oraz te same informacje pozwalające je wykorzystać jako klucz łączący. W plikach shp występują atrybuty, które możemy wykorzystać do powiązania danych przestrzennych z biznesowymi, np. miasto, województwo, kraj.

Słowa kluczowe: przestrzenne typy danych, Microsoft SQL, SSRS

Abstract:

Started Microsoft SQL 2008 Server there is geometry and geography data type, that are spatial type. There are tools to query spatial and base on the point maps. Most maps we can import from shp files to Microsoft SQL 2008 (or newer) tables. We can create spatial index on columns with geometry or geography data type to speed query. In Microsoft SQL 2008R2 Server Reporting Services (or newer) we can create reports on maps. We can create map reports based on builded maps, shp files or query to SQL table with spatial columns. We have to join spatial data with bisnes data on any key. If spatial data are in SQL query we can join based on reference relation, natural if there is foreign key constraint, else based on column with the same data type and the same information. In shp files there are attributes that we can join with data in bisnes data query example city, country or region.

Keywords: spatial data types, Microsoft SQL, SSRS

CHARAKTERYSTYKA PRZESTRZENNICH TYPÓW DANYCH W MICROSOFT SQL 2008

Do przestrzennych typów danych w prowadzonych w Microsoft SQL Server 2008 należą geometry i geography. W typach przestrzennych możemy użyć jednego z typów:

- Point – pozwala uwzględnić położenie jednego punktu opisanego współrzędnymi X,Y oraz opcjonalnie Z (wyniesienie) i M (miara). Zdefiniowanie jednego punktu zajmuje w bazie 22 bajty.
- MultiPoint – pozwala zdefiniować położenie wielu punktów, które nie są ze sobą połączone.
- LineString – pozwala zdefiniować linię łamaną. Każdy z odcinków linii łamanej tworzą punkty. Dwa punkty tworzą jeden odcinek, wiele punktów tworzy wiele odcinków połączonych ze sobą, tak że punkt końcowy pierwszego odcinka jest punktem początkowym kolejnego odcinka, czyli linię łamaną.

- MultiLineString – pozwala zdefiniować wiele linii łamanych, które nie muszą być ze sobą połączone, łączy od zera do wielu obiektów LineString
- Polygon – pozwala zdefiniować powierzchnię ograniczoną wielobokiem. Wierzchołki wieloboku definiują punkty. Pierwszy i ostatni punkt wieloboku musi mieć takie same współrzędne. W wieloboku zewnętrznym może być zdefiniowany wielobok wewnętrzny lub wiele wieloboków wewnętrznych.
- MultiPolygon – pozwala zdefiniować wiele obiektów Polygon.
- GeometryCollection – pozwala połączyć wiele typów obiektów przestrzennych

Opis typów przestrzennych w Microsoft SQL Server jest ogólnodostępny na portalu MSDN: <http://msdn.microsoft.com/en-us/library/ff848797.aspx>

Dane geometry nie uwzględniają krzywizny Ziemi, ani odniesienia do długości czy szerokości geograficznej, tylko położenie w układzie współrzędnych. Mogą być wykorzystane do tworzenie planów, rysunków technicznych. Typ geography uwzględnia krzywiznę Ziemi i ma odniesienie do szerokości i długości geograficznej, jak również możliwość zdefiniowania SRID określającego system odniesienia przestrzennego. Microsoft SQL Server używa domyślnej SRID 4326, która mapuje w systemie odniesienia przestrzennego WGS 84 określającego elipsoidę układu odniesienia oraz jednostce miary dla konkretnego systemu odniesienia.

Podczas definiowania danych można posłużyć się importem/exportem z jednego z trzech znanych formatów:

- WKB (ang. Well Known Binary) – w przypadku danych binarnych
- WKT (ang. Well Known Text) – w przypadku danych tekstowych
- GML (ang. Geography Markup Language) – w przypadku dokumentów XML

Do definiowania danych przestrzennych w odpowiednim formacie służą statyczne metody przypisane dla poszczególnych formatów:

Geometry	WKT	WKB	GML
Point MultiPoint	STPointFromText()	STPointFromWKB()	GeomFromGML()
LineString MultiLineString	STLineFromText()	STLineFromWKB()	GeomFromGML()
Polygon MultiPolygon	STPolyFromText()	STPolyFromWKB()	GeomFromGML()
Geometry Collection	STGeomCollFromText()	STGeomCollFromWKB()	GeomFromGML()
Any Type	STGeomFromText() Parse()	STGeomFromWKB()	GeomFromGML()

Tab.1 Metody statyczne przestrzennych typów danych

Dane przestrzenne w są przechowywane jako duże obiekty binarne (wielkości do 2GB). Do tworzenia rysunku można wykorzystać zapytanie do wielu wierszy tabeli z kolumną typu przestrzennego, więc dane do tworzenia mapy mogą przekraczać rozmiar 2GB, oczywiście przy dużych rysunkach należy uwzględnić wydajność i szybkość serwera. Generowane plany, rysunki czy mapy mogą wykorzystywać warstwy. W tabeli z danymi przestrzennymi lub w tabelach powiązanych referencją mogą być przechowywane dodatkowe dane opisujące, które mogą być wykorzystane do tworzenia opisu mapy czy legendy, a jeśli są unikalne mogą być kluczem łączącym dane biznesowe z danymi przestrzennymi.

Do załadowania danych można użyć np. metody STGeomFromText lub Parse, natomiast metoda ToString pozwala na zamianę geometrycznych danych binarnych na varchar np.:

```
DECLARE @g geometry;
SET @g = geometry::STGeomFromText('POINT (1 2)', 0);
SELECT @g, @g.ToString()
```

UWAGA Wielkość LitER jest ISTotNA!!! Np. zapytanie:

```
DECLARE @g geometry;  
SET @g = geometry::STGeomFromText('POINT (1 2)', 0);  
SELECT @g, @g.ToString()
```

zwróci komunikat, że metoda ToString jest nieznana dla typu geometry.

Można zdefiniować dwa wymiary położenia w układzie współrzędnych, wyniesienie i miarę. Wyniesienie i miara mają domyślnie wartość NULL. Wartości tych atrybutów można pobrać.

```
DECLARE @g geometry;  
SET @g = geometry::Parse('POINT(1 2 3 4)');  
SELECT  
@g.STX AS WymiarX ,  
@g.STY AS WymiarY,  
@g.Z AS WymiarZ,  
@g.M AS Miara;
```

Można użyć wielu punktów. Można załadować dane za pomocą metody STMPPointFromText. Metoda STGeometryN pozwala pobrać wybrany punkt ze zbioru. Zamiast metody ToString można użyć STAsText

```
DECLARE @g geometry;  
SET @g = geometry::STMPPointFromText('MULTIPOINT((1 2), (3 4))', 0);  
SELECT @g.STGeometryN(1).STAsText() AS PierwszyPunkt, @g.STGeometryN(2).STAsText() AS DRUGIPunkt;
```

Możemy użyć linii łamanej otwartej:

```
DECLARE @g geometry= 'LINESTRING(1 1, 1 2, 2 2, 2 1)';  
SELECT @g, @g.ToString()
```

lub linii łamanej zamkniętej, jeśli pierwszy i ostatni punkt ma taką samą wartość:

```
DECLARE @g geometry= 'LINESTRING(1 1, 1 2, 2 2, 2 1, 1 1)';  
SELECT @g, @g.ToString()
```

Można również użyć wielu linii, gdzie każdy obiekt określa punkt początkowy, punkty łamania i punkt końcowy każdej z linii

```
DECLARE @g geometry = 'MULTILINESTRING(  
(0 0, 0 4),  
(-2 2, 2 2),  
(-2 4, 2 0),  
(-2 0, 2 4, 2 3))'  
SELECT @g, @g.ToString();
```

Przy definiowaniu wieloboku należy pamiętać, że powierzchnia znajdzie się wewnątrz wieloboku, co wymaga, aby pierwszy i ostatni punkt miały taką samą wartość.

```
DECLARE @g geometry= 'POLYGON((1 1, 1 2, 2 2, 2 1, 1 1))';  
SELECT @g, @g.ToString()
```

Można również użyć metody STPolyFromText oraz użyć wielu wieloboków

```
DECLARE @g geometry  
SET @g = geometry::STPolyFromText('POLYGON(  
(0 0, 0 3, 3 3, 3 0, 0 0)
```

```
, (1 1, 1 2, 2 1, 1 1))'
, 0);
SELECT @g, @g.ToString();GO
```

Albo zadeklarować wartość przy deklaracji zmiennej:

```
DECLARE @g geometry = 'MULTIPOLYGON(
((2 2, 2 -2, -2 -2, -2 2, 2 2))
,((1 1, 3 1, 3 3, 1 3, 1 1)))';
SELECT @g, @g.ToString()
```

Można tak również połączyć wiele kształtów:

```
DECLARE @g geometry = 'GEOMETRYCOLLECTION(
LINESTRING(1 1, 3 5)
,POLYGON((-1 -1, -1 -5, -5 -5, -5 -1, -1 -1)))';
SELECT @g, @g.ToString()
```

Lub za pomocą metody STGeomCollFromText

```
DECLARE @g geometry;
SET @g = geometry::STGeomCollFromText('GEOMETRYCOLLECTION(
POINT(2 2)
,LINESTRING(1 1, 3 5)
,POLYGON((0 0 2, 1 5 3, 1 0 4, 0 0 2)))', 1);
SELECT @g, @g.ToString()
```

Dla wieloboków można policzyć ich powierzchnię korzystając z metody STAera

```
DECLARE @g1 geometry= 'POLYGON((0 0, 0 1, 1 1, 1 0, 0 0))';
DECLARE @g2 geometry= 'POLYGON((0 0, 0 2, 2 2, 2 0, 0 0))';
DECLARE @g3 geometry= 'POLYGON((0 0, 0 3, 3 3, 3 0, 0 0))';
SELECT
@g1.STArea() AS Pow_g1
,@g2.STArea() AS Pow_g2
,@g3.STArea() AS Pow_g3;
```

Korzystając z metody STDistance można policzyć odległość dwóch punktów

```
DECLARE @a geometry;
DECLARE @b geometry;
SET @a = geometry::STGeomFromText('POINT(0 2)', 0);
SET @b = geometry::STGeomFromText('POINT(0 0)', 0);
SELECT @a.STDistance(@b);
```

Lub odległość dwóch kształtów:

```
DECLARE @a geometry;
DECLARE @b geometry;
SET @a = geometry::STGeomFromText('POLYGON(
(0 0, 2 0, 2 2, 0 2, 0 0))', 0);
SET @b = geometry::STGeomFromText('POLYGON(
(0 -1, -2 -1, -2 -2, 0 -2, 0 -1))', 0);
SELECT @a.STDistance(@b);
```

Najmniejsza odległość jest pomiędzy punktem (0 0) i punktem (0 -1)

Korzystając z metody STLength można sprawdzić długość łamanej

```
DECLARE @g geometry;  
SET @g = geometry::STGeomFromText('LINESTRING(0 0, 0 2, 2 2)', 0);  
SELECT @g.STLength();  
http://msdn.microsoft.com/en-us/library/bb933827.aspx
```

Omówione wcześniej przykłady dotyczące typu geometry mają również zastosowanie również dla typu geography np.:

```
DECLARE @p geography = 'POINT(-122 47)';  
SELECT @p;
```

```
DECLARE @l geography = 'LINESTRING(-120 50, -100 52, -92 42)';  
Select @l;
```

```
DECLARE @g geography = 'POLYGON((-122.358 47.653,  
-122.348 47.649,-122.358 47.658, -122.358 47.653))';  
SELECT @g;
```

Do ładowania danych można wykorzystać metodę STGeomFromText

```
DECLARE @g geography  
SET @g = geography::STGeomFromText('LINESTRING(  
-120 50, -100 52, -92 42)', 4326)  
SELECT @g;
```

Jednym ze źródeł danych przestrzennych są dane z urządzeń GPS w formacie GML:

```
DECLARE @g geography;  
DECLARE @x xml;  
SET @x = '<LineString xmlns="http://www.opengis.net/gml">  
<posList>47.6 22.3 49.6 32.3</posList>  
</LineString>';  
SET @g = geography::GeomFromGml(@x, 4326);  
SELECT @g, @g.ToString(), @g.AsGml();  
Metoda AsGml pozwala uzyskać z typu danych geography dane w formacie GML,
```

W tabeli możemy utworzyć kolumnę typu geography

```
CREATE table Geo (id int primary key, g geography);  
INSERT INTO Geo values  
(0, geography::Point(45, -120, 4326)),  
(1, geography::Point(46, -121, 4326)),  
(2, geography::Point(45, -122, 4326)),  
(3, geography::Point(47, -123, 4326)),  
(4, geography::Point(48, -120, 4326));
```

```
SELECT * FROM Geo
```

Dla kolumny typu geometry lub geography można utworzyć indeks przestrzenny:

```
CREATE spatial index sample_idx on Geo(g);
```

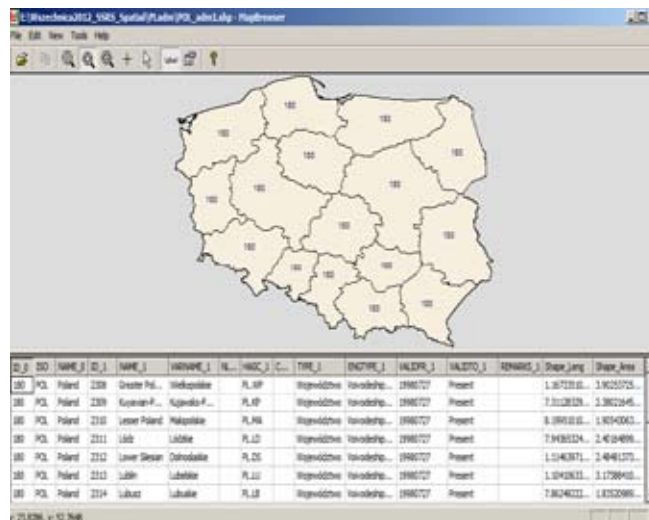
Korzystając z metody STDistance można obliczyć odległość

```
DECLARE @b geography
SET @b = (SELECT g FROM Geo WHERE id = 0)
SELECT id, g.STDistance(@b) FROM Geo
```

Ładowanie danych przestrzennych z plików shp do tabeli SQL

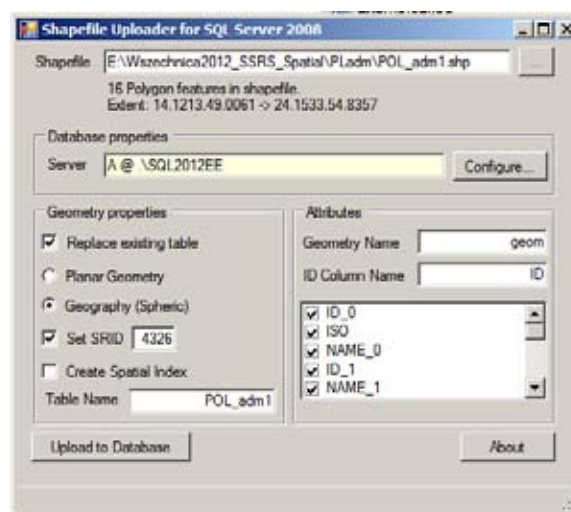
W Internecie jest dostępnych wiele stron z plikami shp. Np strona http://www.vdstech.com/map_data_osm.htm. Są tam dostępne kształty wielu krajów świata zarówno administracyjne jak i fizyczne, w tym mapa administracyjna Polski: pol_adm.zip.

Do przeglądnia plików shp można wykorzystać program MapBrowser, do pobrania ze strony: <http://www.vdstech.com/mapbrowser.htm>



Rys.1 MapBrowser

Pliki shp można załadować do bazy Microsoft SQL narzędzia Shape2Sql.exe, do pobrania z witryny <http://www.sharpgis.net/page/Shape2SQL.aspx>

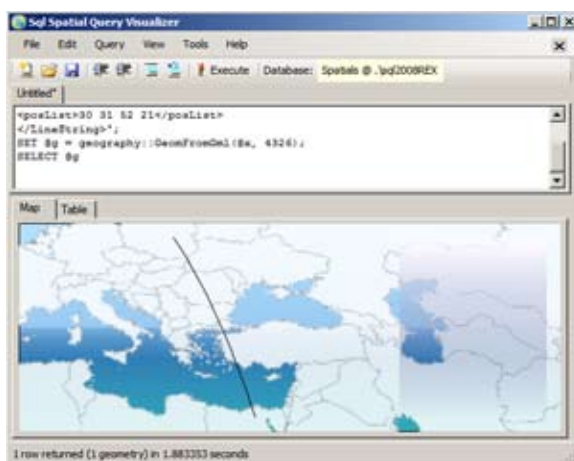


Rys.2 Shape2SQL

W pakiecie jest również dostępne narzędzie SQLSpatial.exe, będące klientem Microsoft SQL, zoptymalizowanym do odpytywania danych przestrzennych. Wykonując w nim zapytanie:

```
DECLARE @g geography;  
DECLARE @x xml;  
SET @x = '<LineString xmlns="http://www.opengis.net/gml">  
<posList>30 31 52 21</posList>  
</LineString>';  
SET @g = geography::GeomFromGml(@x, 4326);  
SELECT @g
```

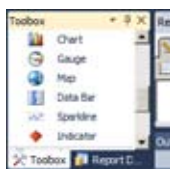
Otrzymamy najkrótszą trasę przelotu z Kairu do Warszawy.



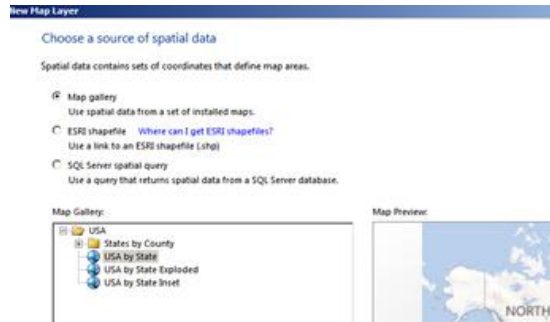
Rys.3 SQLSpatial

TWORZENIE RAPORTU NA MAPIE Z WYKORZYSTANIEM DANYCH PRZESTRZENNYCH Z TABELI SQL

SSRS 2008R2 i SSRS 2012 pozwalają na tworzenie wykresów z wykorzystaniem map (SSRS w Microsoft SQL2008 nie ma tej funkcjonalności). Podobna funkcjonalność była w ubiegłym tysiącleciu dostępna w Microsoft Excel do wersji 2000, później usunięto MsMap z pakietu Office. Do definiowania raportów można wykorzystać Microsoft Business Intelligence Development Studio będące składową Microsoft Visual Studio 2010 lub Report Builder 3.0 (albo nowsze).



Rys.4 Raport na mapie



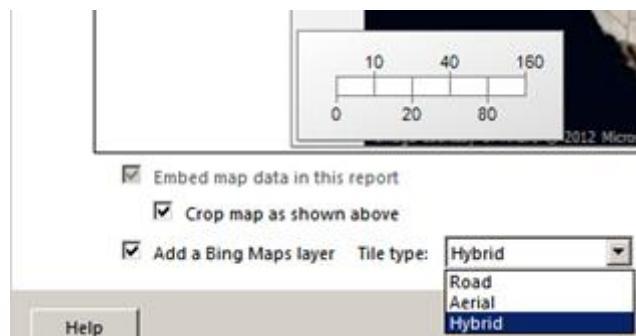
Rys.5 Wybór źródła danych przestrzennych

W SSRS możemy tworzyć mapy z wykorzystaniem predefiniowanych map dotyczących Ameryki Północnej.

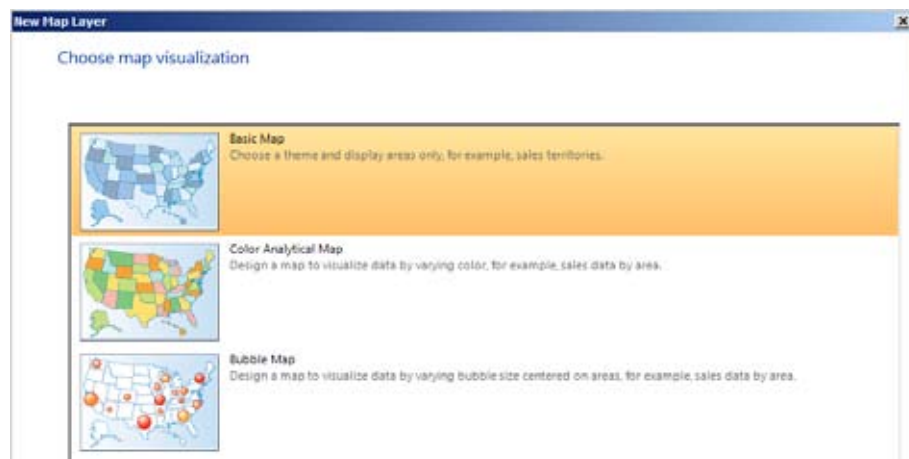
Można dodatkowo wykorzystać warstwy pochodzące z Bing Maps pozwalające obrazować cały świat.

Do generowania warstwy mapy możemy wykorzystać pliki shp, czyli popularny format map dostępny na wielu portalach. Alternatywą jest wykonanie zapytania SQL do tabel zawierających dane geograficzne.

Do wizualizacji danych biznesowych można wykorzystać zakresy barw lub wykres bąbelkowy

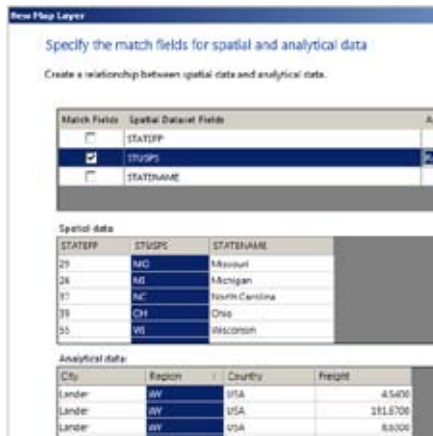


Rys.6. Warstwa mapy Bing

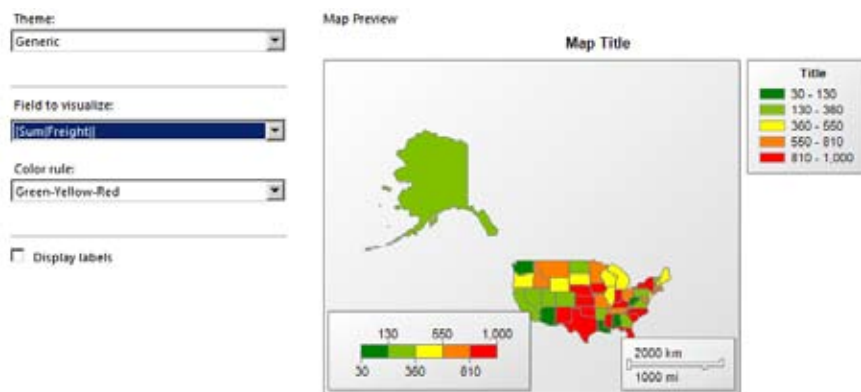


Rys.7. Wybór sposobu wizualizacji mapy

Na warstwie podstawowej mapy możemy umieścić zagregowane dane biznesowe powiązane za pomocą kolumny z atrybutem danych geograficznych poprzez wspólne wartości nazw geograficznych.



Rys.8. Powiązanie danych analitycznych z danymi przestrzennymi

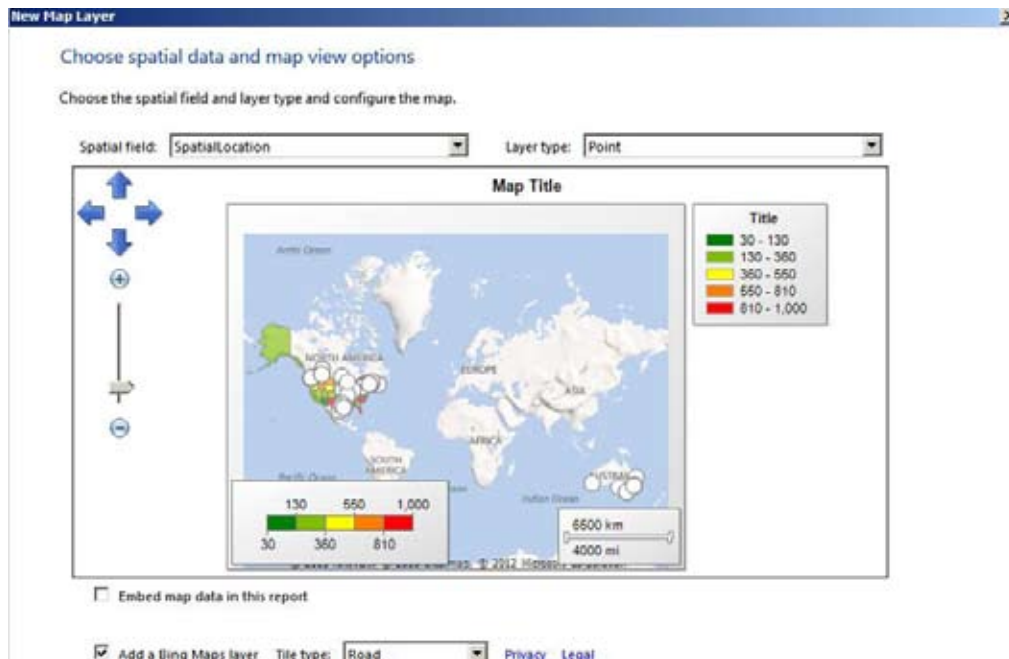


Rys.9. Wybór wizualizacji

Do utworzonego wykresu z danymi biznesowymi na mapie możemy dodawać kolejne warstwy wskazujące punkty, linie czy obszary na mapie utworzone na podstawie zapytania do kolumny z danymi geograficznymi.



Rys.10. Dodanie warstwy



Rys.11. Wybór typu przestrzennego dodatkowej warstwy

Należy przy doborze danych zwrócić uwagę, aby warstwy odnosiły się do spójnych obszarów geograficznych, ponieważ połączenie warstw dotyczących np. Polski i Australii lub mapy województwa Mazowieckiego i punktów geograficznych z całej Europy doprowadzi do nieczytelności wykresu.

PODSUMOWANIE

Funkcjonalność tworzenia wykresów na mapie, która została wprowadzona w SSRS 2008R2 w połączeniu z obsługą danych przestrzennych, w tym typu geography w Microsoft SQL Server 2008 i nowszych pozwala na atrakcyjne obrazowanie wyników biznesowych, co ułatwia i uatrakcyjnia ich analizę.