

Przetwarzanie danych przestrzennych (zadania)

SQL/MM

Krzysztof Jankiewicz
Politechnika Poznańska, Instytut Informatyki

Standard SQL/MM znalazł wiele implementacji. Przykładowymi systemami zarządzania bazami danych, które go wykorzystują są Oracle, PostgreSQL oraz DB2.

Poniższe ćwiczenia, dotyczące standardu SQL/MM, można wykonać przy wykorzystaniu każdej z nich. W przypadku ćwiczeń, które dotyczą tylko jednej z tych baz danych, przed treścią ćwiczenia (w nawiasach) znajduje się odpowiednie oznaczenie wskazujące właściwą dla ćwiczenia implementację.

UWAGA: Polecen z dopiskiem „(DB2)” NIE wykonujemy.

Ćwiczenie 1

Standard SQL/MM Part: 3 Spatial.

A. **(Oracle)** Wykorzystując klauzule CONNECT BY wyświetl hierarchię typu ST_GEOMETRY.

```
select lpad('-',2*(level-1),'|-') || t.owner||'.'||t.type_name||' (FINAL:'||t.final||
', INSTANTIABLE:'||t.instantiable||', ATTRIBUTES:'||t.attributes||', METHODS:'||t.methods||')'
from   all_types t
start with t.type_name = 'ST_GEOMETRY'
connect by prior t.type_name = t.supertype_name
         and prior t.owner = t.owner;
```

```
LPAD('-',2*(LEVEL-1),'|-')||T.OWNER||'.'||T.TYPE_NAME||' (FINAL:'||T.FINAL||'
-----
MDSYS.ST_GEOMETRY (FINAL:NO, INSTANTIABLE:YES, ATTRIBUTES:1, METHODS:41)
|-MDSYS.ST_CURVE (FINAL:NO, INSTANTIABLE:YES, ATTRIBUTES:1, METHODS:52)
|-|-MDSYS.ST_CIRCULARSTRING (FINAL:YES, INSTANTIABLE:YES, ATTRIBUTES:1, METHODS:55)
|-|-MDSYS.ST_COMPOUNDCURVE (FINAL:YES, INSTANTIABLE:YES, ATTRIBUTES:1, METHODS:60)
|-|-MDSYS.ST_LINESTRING (FINAL:YES, INSTANTIABLE:YES, ATTRIBUTES:1, METHODS:56)
|-MDSYS.ST_GEOMCOLLECTION (FINAL:NO, INSTANTIABLE:YES, ATTRIBUTES:1, METHODS:48)
|-|-MDSYS.ST_MULTICURVE (FINAL:NO, INSTANTIABLE:YES, ATTRIBUTES:1, METHODS:50)
|-|-|-MDSYS.ST_MULTILINESTRING (FINAL:YES, INSTANTIABLE:YES, ATTRIBUTES:1, METHODS:50)
|-|-MDSYS.ST_MULTIPOINT (FINAL:YES, INSTANTIABLE:YES, ATTRIBUTES:1, METHODS:48)
|-|-MDSYS.ST_MULTISURFACE (FINAL:NO, INSTANTIABLE:YES, ATTRIBUTES:1, METHODS:49)
|-|-|-MDSYS.ST_MULTIPOLYGON (FINAL:YES, INSTANTIABLE:YES, ATTRIBUTES:1, METHODS:53)
|-MDSYS.ST_POINT (FINAL:YES, INSTANTIABLE:YES, ATTRIBUTES:1, METHODS:49)
|-MDSYS.ST_SURFACE (FINAL:NO, INSTANTIABLE:YES, ATTRIBUTES:1, METHODS:44)
|-|-MDSYS.ST_CURVEPOLYGON (FINAL:NO, INSTANTIABLE:YES, ATTRIBUTES:1, METHODS:54)
|-|-|-MDSYS.ST_POLYGON (FINAL:YES, INSTANTIABLE:YES, ATTRIBUTES:1, METHODS:63)
```

15 rows selected

B. **(Oracle)** Wyświetl nazwy metod typu ST_POLYGON.

```
select distinct m.method_name
from   all_type_methods m
where  m.type_name like 'ST_POLYGON'
and    m.owner = 'MDSYS'
order by 1;
```

Element	Type
GEOM	MDSYS.SDO_GEOMETRY
GET_SDO_GEOM	FUNCTION
GET_WKB	FUNCTION
GET_WKT	FUNCTION
...	
ST_NUMINTERIORRING	FUNCTION
ST_INTERIORRINGN	FUNCTION
ST_POLYGON	FUNCTION
ST_INTERIORRINGSP	FUNCTION
ST_BDPOLYFROMTEXT	FUNCTION
ST_BDPOLYFROMWKB	FUNCTION

C. **(Oracle)** Utwórz tabelę MYST_MAJOR_CITIES o następujących kolumnach:

- FIPS_CNTRY VARCHAR2(2),
- CITY_NAME VARCHAR2(40),
- STGEOM ST_POINT.

Table created

(DB2) Utwórz tabelę MYST_MAJOR_CITIES o następujących kolumnach:

- FIPS_CNTRY VARCHAR(2),
- CITY_NAME VARCHAR(40),
- STGEOM ST_POINT.

Uwaga! W bazie danych DB2 właścicielem typów przestrzennych ST* (a także innych obiektów związanych z przetwarzaniem danych przestrzennych) jest DB2GSE.

Sukces

D. **(Oracle)** Przepisz zawartość tabeli MAJOR_CITIES (znajduje się ona w schemacie ZSBD_TOOLS) do stworzonej przez Ciebie tabeli MYST_MAJOR_CITIES dokonując odpowiedniej konwersji typów.

123 rows inserted

(DB2) Przepisz zawartość tabeli MAJOR_CITIES (znajduje się ona w schemacie ZSBD_OWNER) do stworzonej przez Ciebie tabeli MYST_MAJOR_CITIES. Nie musisz dokonywać żadnych konwersji typów.

Zaktualizowano następującą liczbę wierszy: 123.

Uwaga! W bazie danych DB2 nie ma typowych typów obiektowych. W związku z tym przetwarzanie danych przestrzennych w oparciu o standard SQL/MM nie jest realizowane za pomocą metod. W DB2 możemy mówić o proceduralnej lub funkcyjnej implementacji standardu SQL/MM.

Przykład: Zapytanie które byłoby odpowiedzią na zadanie „Sprawdź ile miejscowości zawiera się w danym państwie” w przypadku bazy danych Oracle wyglądałoby następująco:

```
select B.CNTRY_NAME, count(*)
from MYST_COUNTRY_BOUNDARIES B, MYST_MAJOR_CITIES C
where C.STGEOM.ST_WITHIN(B.STGEOM) = 1
group by B.CNTRY_NAME;
```

Tymczasem w przypadku bazy danych DB2 będzie ono wyglądało tak:

```
select B.CNTRY_NAME, count(*)
from MYST_COUNTRY_BOUNDARIES B, MYST_MAJOR_CITIES C
where DB2GSE.ST_WITHIN(C.STGEOM, B.STGEOM) = 1
group by B.CNTRY_NAME;
```

Ćwiczenie 2

Standard SQL/MM Part: 3 Spatial – konwersja formatów

- A. Wstaw do tabeli MYST_MAJOR_CITIES informację dotyczącą Szczyrku. Załóż, że centrum Szczyrku znajduje się w punkcie o współrzędnych 19.036107; 49.718655. Wprowadź informację przy wykorzystaniu formatu *well-known text* (WKT).

1 row inserted

- B. Wyświetl w formacie *well-known text* (WKT) definicje przestrzenne rzek umieszczone w tabeli RIVERS.

NAME	WKT
Elbe	LINESTRING (14.23346 50.86964, 14.18706 50.891781, 14.1084
50.913361, 14.03845 5	
Vistula	LINESTRING (21.001011 52.268829, 21.021959 52.25135, 21.06971
52.23241, 21.10969	
Vistula	LINESTRING (18.914459 49.70644, 18.832359 49.801899, 18.803619
49.845089, 18.799	
Odra	LINESTRING (14.80079 52.079731, 14.89331 52.037868, 14.9572
52.031139, 15.08692	
Labe	LINESTRING (14.56134 50.28437, 14.60777 50.249771, 14.64611
50.223949, 14.66837	
Goryn'	MULTILINESTRING ((25.565729 49.843349, 25.784321 49.85622,
25.88303 49.87571, 25	
...	

125 rows selected

- C. Wyświetl definicję przestrzenną wprowadzonego przez Ciebie miasta Szczyrk w postaci formatu GML.

```
<gml:Point srsName="SDO:" xmlns:gml="http://www.opengis.net/gml">
  <gml:coordinates decimal="." cs="," ts=" ">19.036107,49.718655 </gml:coordinates>
</gml:Point>
```

Ćwiczenie 3

Standard SQL/MM Part: 3 Spatial – pobieranie własności i miar

- A. **(Oracle)** Utwórz tabelę MYST_COUNTRY_BOUNDARIES z następującymi atrybutami
- FIPS_CNTRY VARCHAR2(2),
 - CNTRY_NAME VARCHAR2(40),
 - STGEOM ST_MULTIPOLYGON.

Table created

- (DB2)** Utwórz tabelę MYST_COUNTRY_BOUNDARIES z następującymi atrybutami
- FIPS_CNTRY VARCHAR(2),
 - CNTRY_NAME VARCHAR(40),
 - STGEOM ST_MULTIPOLYGON.

Sukces

- B. Przepisz zawartość tabeli COUNTRY_BOUNDARIES do nowo utworzonej tabeli dokonując odpowiednich konwersji (w przypadku DB2 konwersja nie będzie potrzebna).

19 rows inserted

C. Sprawdź jakiego typu i ile obiektów przestrzennych zostało umieszczonych w tabeli MYST_COUNTRY_BOUNDARIES.

TYP_OBIEKTU	ILE
ST_MULTIPOLYGON	7
ST_POLYGON	12

D. Sprawdź czy wszystkie definicje przestrzenne uznawane są za proste.

```
B.STGEOM.ST_ISSIMPLE()
-----
1
...
1
19 rows selected
```

Ćwiczenie 4

Standard SQL/MM Part: 3 Spatial – przetwarzanie danych przestrzennych

A. Sprawdź ile miejscowości (MYST_MAJOR_CITIES) zawiera się w danym państwie (MYST_COUNTRY_BOUNDARIES).

ORA-13295: obiekty geometrii znajdują się w różnych systemach współrzędnych

Co jest powodem błędu? Usuń przyczynę.

CNTRY_NAME	COUNT(*)
Denmark	1
Poland	51
...	
Lithuania	1
Ukraine	7

15 rows selected

B. Znajdź te państwa, które graniczą z Czechami.

A_NAME	B_NAME
Austria	Czech Republic
Poland	Czech Republic
Germany	Czech Republic
Slovakia	Czech Republic

C. Znajdź nazwy tych rzek, które przecinają granicę Czech – wykorzystaj tabelę RIVERS (w przypadku bazy Oracle wykorzystaj także konstruktor typu ST_LINESTRING).

CNTRY_NAME	NAME
Czech Republic	Spree
Czech Republic	Morava
Czech Republic	Odra
Czech Republic	Vltava
Czech Republic	Labe

D. Sprawdź, jaka powierzchnia jest Czech i Słowacji połączonych w jeden obiekt przestrzenny.

```
POWIERZCHNIA
-----
126373071900
```

E. Sprawdź jakiego typu obiektem są Węgry z "wykrojonym" Balatonem – wykorzystaj tabelę WATER_BODIES.

OBIEKT	WEGRY_BEZ
ST_POLYGON(SDO_GEOMETRY(2003, 8307, NULL, SDO_ELEM_INFO_ARRAY(1, 1003, 1, 609, 2003, 1), SDO_ORDINATE_ARRAY(22,270275, 48,359993, ...	ST_POLYGON

Ćwiczenie 5

Standard SQL/MM Part: 3 Spatial – indeksowanie i przetwarzanie przy użyciu operatorów SDO_NN i SDO_WITHIN_DISTANCE.

Uwaga! Całe ćwiczenie dotyczy tylko bazy danych Oracle.

- A. **(Oracle)** Wykorzystując operator SDO_WITHIN_DISTANCE znajdź liczbę miejscowości oddalonych od terytorium Polski nie więcej niż 100 km. (wykorzystaj tabele MYST_MAJOR_CITIES i MYST_COUNTRY_BOUNDARIES). Obejrzyj plan wykonania zapytania. (Uwaga: We wcześniejszych wersjach Oracle użycie tych operatorów nawet dla standardowych typów SQL/MM było możliwe tylko z pomocą indeksu przestrzennego. Bez niego zapytanie kończyło się błędem „ORA-13226: interfejs nie jest obsługiwany bez indeksu przestrzennego”).
- B. **(Oracle)** Zarejestruj metadane dotyczące stworzonych przez Ciebie tabeli MYST_MAJOR_CITIES i/lub MYST_COUNTRY_BOUNDARIES.

1 row inserted

- C. **(Oracle)** Utwórz na tabelach MYST_MAJOR_CITIES i/lub MYST_COUNTRY_BOUNDARIES indeks R-drzewo.

Index created

- D. **(Oracle)** Ponownie znajdź liczbę miejscowości oddalonych od terytorium Polski nie więcej niż 100 km. Sprawdź jednocześnie, czy założone przez Ciebie indeksy są wykorzystywane wyświetlając plan wykonania zapytania.

A_NAME	COUNT(*)
Poland	67

Plan hash value: 2247583427

Id	Operation	Name	Rows	Bytes	Cost (%CPU)	Time
0	SELECT STATEMENT		1	7672	7 (0)	00:00:01
1	SORT GROUP BY NOSORT		1	7672	7 (0)	00:00:01
2	NESTED LOOPS		1	7672	7 (0)	00:00:01
* 3	TABLE ACCESS FULL	MYST_COUNTRY_BOUNDARIES	1	3841	5 (0)	00:00:01
* 4	DOMAIN INDEX	MYST_MAJOR_CITIES_IDX				

Predicate Information (identified by operation id):

```
3 - filter("B"."CNTRY_NAME"='Poland')
4 - access("MDSYS"."SDO_WITHIN_DISTANCE"("C"."STGEOM","B"."STGEOM",'distance=100
    unit=km')='TRUE')
```