Bazy Danych

5. Przestrzenne Bazy Danych

Opracował: Maciej Penar

Spis treści

[1. Linki 3](#_Toc513823820)

[2. Dlaczego ten temat? 3](#_Toc513823821)

[3. Cóż to są te „dane przestrzenne”? 4](#_Toc513823822)

[Kontekst 4](#_Toc513823823)

[PRzykład 4](#_Toc513823824)

[Operacje 5](#_Toc513823825)

[4. (6 pkt) Zadania 6](#_Toc513823826)

[Treść właściwa zadania z parkingiem 7](#_Toc513823827)

# 1. Linki

Przydatne linki:

* Oracle 12c: [link](http://www.oracle.com/technetwork/database/enterprise-edition/downloads/index.html)
* Przestrzenne typy danych w Oracle: [link](https://docs.oracle.com/cd/A97630_01/appdev.920/a96630/sdo_objrelschema.htm)
* Przestrzenne dane w Oracle: [link](https://docs.oracle.com/database/121/SPATL/toc.htm)
* Operatory przestrzenne w Oracle: [link](https://docs.oracle.com/cd/B28359_01/appdev.111/b28400/sdo_operat.htm)
* Funkcje przestrzenne w Oracle: [link](https://docs.oracle.com/cd/A87860_01/doc/inter.817/a85337/sdo_objg.htm)

# 2. Dlaczego ten temat?

Każdy bazodanowy „duży gracz” ma pewną funkcjonalność której nie można znaleźć u konkurentów. I tak:

* IBM DB2 posiada mechanizm BLU: shadowing + columnstore + in-memory
* Microsoft SQL Server: posiada przetwarzanie pełnotekstowe / semantic search
* Oracle DB: posiada rozszerzenie temporalne

Z tego względu warto zapoznać się z funkcjonalnością która czyni Oracle DB wyjątkową bazą danych.

# 3. Cóż to są te „dane przestrzenne”?

## Kontekst

Dane przestrzenne (ang. Spatial Data) jest to specjalny typ danych opisujący figury geometryczne np.:

* Punkty
* Linie
* Płaszczyzny (ang. Surfaces)
* Przestrzenie (ang. Planes)
* Hiper-przestrzenie (ang. Hype-planes)

W Oracle istnieje od tego specjalny typ danych: **SDO\_GEOMETRY.** Ogólnie rzecz biorąc konstruowanie tego typu obiektów nie należy do najłatwiejszych.

Obiekt przestrzenny jest zdefiniowany jako (nie wykonywać):

|  |
| --- |
| CREATE TYPE sdo\_geometry AS OBJECT (  SDO\_GTYPE NUMBER,  SDO\_SRID NUMBER,  SDO\_POINT SDO\_POINT\_TYPE,  SDO\_ELEM\_INFO MDSYS.SDO\_ELEM\_INFO\_ARRAY,  SDO\_ORDINATES MDSYS.SDO\_ORDINATE\_ARRAY); |

Gdzie:

* SDO\_GTYPE – definiuje typ obiektu przestrzennego: ile ma wymiaru, jaki rodzaj obiektu to jest
* SDO\_SRID – definiuje w jakim układzie współrzędnych osadzony jest obiekt
* SDO\_POINT – definiuje punkt dla obiektów punktowych
* SDO\_ELEM – definiuje ile i jakie punkty wchodzą w skład topologii
* SDO\_ORDINATES – definiuje parametry obiektów z SDO\_ELEM

## PRzykład

Dana jest tabela:

|  |
| --- |
| **CREATE TABLE** SENSORS**(**  ID **INT GENERATED ALWAYS AS IDENTITY,**  VALUE **NUMERIC,**  SHAPE **SDO\_GEOMETRY**  **)** |

Ładowanie **punktu** o wartości 125 i koordynatach (10,5):

|  |
| --- |
| **INSERT INTO** SENSORS(VALUE, SHAPE)  **VALUES(125, MDSYS.SDO\_GEOMETRY(**2001**,** NULL**, MDSYS.SDO\_POINT\_TYPE**(10, 5, NULL**),** NULL**,** NULL**));** |

Spróbuj załadować:

* Punkt o wartości 65 i koordynatach (0, 0)
* Punkt o wartości 43 i koordynatach (2, 2)
* Punkt o wartości 255 i koordynatach (8, 7)

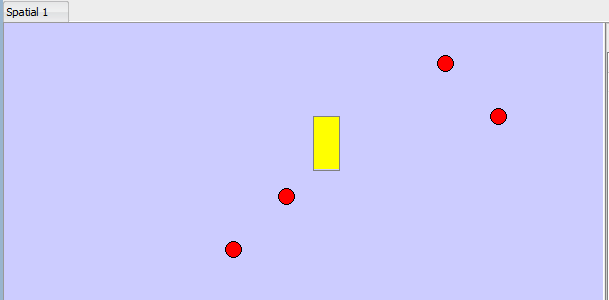
Ładowanie prostokąta o wartości 0 i koordynatach (3,3) do (4,5)

|  |
| --- |
| **INSERT** **INTO** SENSORS(VALUE, SHAPE)  **VALUES**(0, **MDSYS.SDO\_GEOMETRY**(2003, NULL, NULL, **MDSYS.SDO\_ELEM\_INFO\_ARRAY**(1,1003,3), **MDSYS.SDO\_ORDINATE\_ARRAY**(3,3, 4,5))); |

Obiekty przestrzenne mają reprezentację na mapie. Mapę można obejrzeć:

1. View->MapView
2. Klikamy 
3. Wpisujemy nazwę oraz zapytanie: SELECT SHAPE FROM SENSORS

Powinniśmy zobaczyć:



## Operacje

Na obiektach przestrzennych możemy wykonać wiele operacji dostępnych w formie specjalnych funkcji zwanych operatorami oraz funkcjami przestrzennymi. Do najciekawszych należą:

* SDO\_GEOM.SDO\_UNION – do liczenia unii dwóch przestrzeni
* SDO\_GEOM.SDO\_INTERSECTION – do liczenia przecięcia dwóch przestrzeni
* SDO\_GEOM.SDO\_DIFFERENCE – do liczenia różnicy dwóch przestrzeni
* SDO\_GEOM.SDO\_DISTANCE – do liczenia odległości pomiędzy dwoma przestrzeniami
* SDO\_GEOM.SDO\_AREA – do liczenia powierzchni dwu-wymiarowych przestrzeni
* SDO\_GEOM.SDO\_CENTROID – do wyznaczania centroidu przestrzeni

# 4. (6 pkt) Zadania

1. (1 pkt) Narysować różnicę prostokątów: (1,1) (7,5) i (2,1)(5,4), próg dokładności przyjąć 0.1
2. (1 pkt) Policzyć pole z 1), próg dokładności przyjąć 0.1
3. (4 pkt) Załóżmy istnienie tabeli PARKING\_SPOT takiej jak:

|  |
| --- |
| CREATE TABLE PARKING\_SPOT(  NAME VARCHAR(100) NOT NULL UNIQUE,  MAX\_CAP INT CHECK (MAX\_CAP >=0) NOT NULL,  CURRENT\_CAP INT,  PLACE SDO\_GEOMETRY NOT NULL,  CHECK (CURRENT\_CAP <= MAX\_CAP) NOT NULL  ) |

Gdzie:

* NAME to nazwa parkingu
* MAX\_CAP to maksymalna liczba samochodów na parkingu
* CURRENT\_CAP to aktualna liczba samochodów na parkingu
* PLACE to topologia parkingu

Załóżmy że istnieją następujące parkingi:

|  |
| --- |
| INSERT INTO PARKING\_SPOT  VALUES('PARKING A', 100, 98, SDO\_GEOM.SDO\_DIFFERENCE(  MDSYS.SDO\_GEOMETRY(2003, NULL, NULL, MDSYS.SDO\_ELEM\_INFO\_ARRAY(1,1003,3), MDSYS.SDO\_ORDINATE\_ARRAY(8, 8, 12, 14)),  MDSYS.SDO\_GEOMETRY(2003, NULL, NULL, MDSYS.SDO\_ELEM\_INFO\_ARRAY(1,1003,3), MDSYS.SDO\_ORDINATE\_ARRAY(8, 8, 10, 10)),  0.1  )); |
| INSERT INTO PARKING\_SPOT  VALUES('PARKING B', 5, 3,  MDSYS.SDO\_GEOMETRY(2003, NULL, NULL, MDSYS.SDO\_ELEM\_INFO\_ARRAY(1,1003,3), MDSYS.SDO\_ORDINATE\_ARRAY(4, 4, 5, 5))  ); |
| INSERT INTO PARKING\_SPOT  VALUES('PARKING C', 5, 5,  MDSYS.SDO\_GEOMETRY(2003, NULL, NULL, MDSYS.SDO\_ELEM\_INFO\_ARRAY(1,1003,3), MDSYS.SDO\_ORDINATE\_ARRAY(5, 6, 6, 7))  ); |
| INSERT INTO PARKING\_SPOT  VALUES('SUNNY PARKING', 75, 73,  MDSYS.SDO\_GEOMETRY(2003, NULL, NULL, MDSYS.SDO\_ELEM\_INFO\_ARRAY(1,1003,3), MDSYS.SDO\_ORDINATE\_ARRAY(0, 0, 2, 10))  ); |
| INSERT INTO PARKING\_SPOT  VALUES('PARKING DARK', 100, 98, SDO\_GEOM.SDO\_DIFFERENCE(  MDSYS.SDO\_GEOMETRY(2003, NULL, NULL, MDSYS.SDO\_ELEM\_INFO\_ARRAY(1,1003,3), MDSYS.SDO\_ORDINATE\_ARRAY(-2, -2, 16, 16)),  MDSYS.SDO\_GEOMETRY(2003, NULL, NULL, MDSYS.SDO\_ELEM\_INFO\_ARRAY(1,1003,3), MDSYS.SDO\_ORDINATE\_ARRAY(-1, -1, 15, 15)),  0.1  )); |

## Treść właściwa zadania z parkingiem

**(3 pkt) Napisać funkcję która:**

* **Przyjmuje w parametrze pojazd (pojazd można zamodelować w dowolny sposób)**
* **Rezerwuje miejsce na najbliższym dostępnym parkingu**
* **Zwraca nazwę parkingu**

(1 pkt) Jaki jest wynik dla ciągu testowego:

* Samochód 1 jest w punkcie (8,8)
* Samochód 2 jest w punkcie (5.5, 6.5)
* Samochód 3 jest w punkcie (13, 13.5)
* Samochód 4 jest w punkcie (13, 13.5)
* Samochód 5 jest w punkcie (4.8, 1.2)
* Samochód 6 jest w punkcie (4.75, 1.2)
* Samochód 7 jest w punkcie (4.9, 1.2)
* Samochód 8 jest w punkcie (0,0)
* Samochód 9 jest w punkcie (0,0)