Spatial DB Tutorial
(shorten version)

Ugur Demiryurek
Hien To
University of Southern California
Spatial Database

• Spatial databases provide structures for storage, querying and analysis of spatial data

• Spatial data is comprised of objects in multi-dimensional space
Spatial Database

• **Data types in Standard DB**
  – varchar
  – integer
  – real
  – date

• **Data types in Spatial DB**
  – point
  – linestring
  – polygon
  – multipoint
  – multilinestring
  – multipolygon

**Note:**
- The data types in a spatial database include not only the standard database types but also the geometric types used for spatial data.
Standard vs Spatial Database

- Standard database has one dimensional indexes: B-tree, Hash, Bitmap

- Has spatial (multi dimensional) indexes: R-tree, Quad-tree
Architecture

Database → Web Service

JDBC
ODBC

XML
JSON

WWW

Tablet
Spatial Database- Oracle

**Oracle 10g Client Installation**

1. Go to [http://www.oracle.com/technology/software/products/database/index.html](http://www.oracle.com/technology/software/products/database/index.html) and scroll down to “Oracle Database 10g Release 2” section to select your OS specific version. With this tutorial, we will install Microsoft Windows Vista and Windows 2008 version. Note that the installation of other OS is quite similar. If you need more information on installation (and Oracle usage) you can refer to Oracle Documentation Library at [http://www.oracle.com/pls/db102/homepage](http://www.oracle.com/pls/db102/homepage).

2. Click on “Oracle Database 10g Release 2 (10.2.0.3/10.2.0.4) for Microsoft Windows Vista and Windows 2008”. After accepting the “License Agreement” download Oracle Database 10g Client Release 2 (10.2.0.3) file “10203_vista_w2k8_x86_production_client.zip” to your hard disk. Note that, in order to download the file, you will need to register a username and password with Oracle.

[Image of Oracle Database 10g Release 2]

[Link to Oracle Client installation tutorial pdf: http://infolab.usc.edu/projects/TransDec/Oracle_Client_installation_tutorial.pdf](http://infolab.usc.edu/projects/TransDec/Oracle_Client_installation_tutorial.pdf)
Spatial Database - DML

• Table Creation

CREATE TABLE RESTRNTS_Yourid
(  POI_ID      NUMBER,
   POI_NAME    VARCHAR2(35 BYTE),
   POI_ST_NUM  VARCHAR2(10 BYTE),
   ST_NAME     VARCHAR2(80 BYTE),
   PH_NUMBER   VARCHAR2(15 BYTE),
   FOOD_TYPE   VARCHAR2(35 BYTE),
   GEOM        MDSYS.SDO_GEOMETRY
 )
Spatial Database - DML

- Data Insert

Insert into RESTRNTS_YourID (POI_ID, POI_NAME, POI_ST_NUM, ST_NAME, PH_NUMBER, FOOD_TYPE, GEOM) Values
(17597899, 'GAMAGOL KOREAN RESTAURANT', '11900', 'SOUTH ST', '562-4021952', 'KOREAN FOOD', "MDSYS"."SDO_GEOMETRY"(2001,8307, "MDSYS"."SDO_POINT_TYPE"(-118.07935,33.85852,NULL),NULL,NULL));

Insert into RESTRNTS_YourID (POI_ID, POI_NAME, POI_ST_NUM, ST_NAME, PH_NUMBER, FOOD_TYPE, GEOM) Values
(19404550, 'VALENTINO PIZZERIA', '5782', 'E 2ND ST', '562-4383939', 'ITALIAN FOOD', "MDSYS"."SDO_GEOMETRY"(2001,8307, "MDSYS"."SDO_POINT_TYPE"(-118.12126,33.75655,NULL),NULL,NULL));
Spatial Database – Index

• Index Creation
  – INSERT INTO USER_SDO_GEOM_METADATA (TABLE_NAME, COLUMN_NAME, DIMINFO, SRID) VALUES ('RESTRNTS_YourId', 'GEOM', MDSYS.SDO_DIM_ARRAY(MDSYS.SDO_DIM_ELEMENT('LONG', -180.0, 180.0, 0.005), MDSYS.SDO_DIM_ELEMENT('LAT', -90.0, 90.0, 0.005)), 8307);

  – CREATE INDEX INDX_RESTRNTS_YourId ON RESTRNTS_YourId (GEOM) INDEXTYPE IS MDSYS.SPATIAL_INDEX;
Spatial Database – Query

• Nearest Neighbor Query

```
SELECT * FROM RESTRNTS _Yourid A WHERE SDO_NN(A.GEOM,
  mdsys.sdo_geometry(2001,8307,mdsys.sdo_point_type(118.07938,33.8586,NULL),
  NULL, NULL), 'sdo_num_res=1') = 'TRUE';
```

• Range Query

```
Select * FROM RESTRNTS _Yourid A WHERE SDO_FILTER(A.Geom,
  mdsys.sdo_geometry(2003,8307,NULL,mdsys.sdo_elem_info_array(1,1003,3),
  msys.sdo_ordinate_array(-118.07935,33.85852,-118.07920,33.869960)),
  'querytype=WINDOW') = 'TRUE';
```

• More
  – SDO_WITHIN_DISTANCE
  – SDO_RELATE
  – [http://docs.oracle.com/cd/B10501_01/appdev.920/a96630/sdo_index_query.htm](http://docs.oracle.com/cd/B10501_01/appdev.920/a96630/sdo_index_query.htm)
Database Connectivity

• **JDBC**
  
  – JDBC is the Java API for executing dynamic SQL statements.
  
  – The JDBC API is the industry standard for database-independent connectivity between the Java programming language and a wide range of databases.
Database Connectivity

- **Oracle DB Connectivity using JDBC**
  - Make sure that driver class (classes12.jar) is in your classpath

```java
import java.sql.*;
public class TestDBOracle {

public static void main(String[] args) throws ClassNotFoundException, SQLException {
    DriverManager.registerDriver (new oracle.jdbc.driver.OracleDriver());
    String url = "jdbc:oracle:thin:@//128.125.163.168:1521/csci585";

    Connection conn = DriverManager.getConnection(url,"team1","team1");

    conn.setAutoCommit(false);
    Statement stmt = conn.createStatement();
    ResultSet rset = stmt.executeQuery("select POI_NAME from RESTRNTS ");
    while (rset.next()) {
        System.out.println (rset.getString(1));
    }
    stmt.close();
    System.out.println ("Ok.");
}
```