Section 3: April 16th 2009

Plan

- Normal forms

- Project 2
    - Overview
    - JDBC

Normal forms

- Why do we need them?

- What are some examples?
  - 1NF, 2NF
  - 3NF
  - 4NF
  - BCNF

BCNF

- BCNF: If **A1, …, An -> B** is a non-trivial dependency in R, then {A1, …, An} is a superkey for R.

BCNF Exercise

JobInterviews

CandID	Date	Time	EmpID	RoomNo
C21	16-April-09	9.30 AM	E211	CSE550
C21	17-April-09	11.00 AM	E211	CSE550
C5	16-April-09	11.00 AM	E51	CSE218
C2	1-May-09	9.30 AM	E211	CSE218

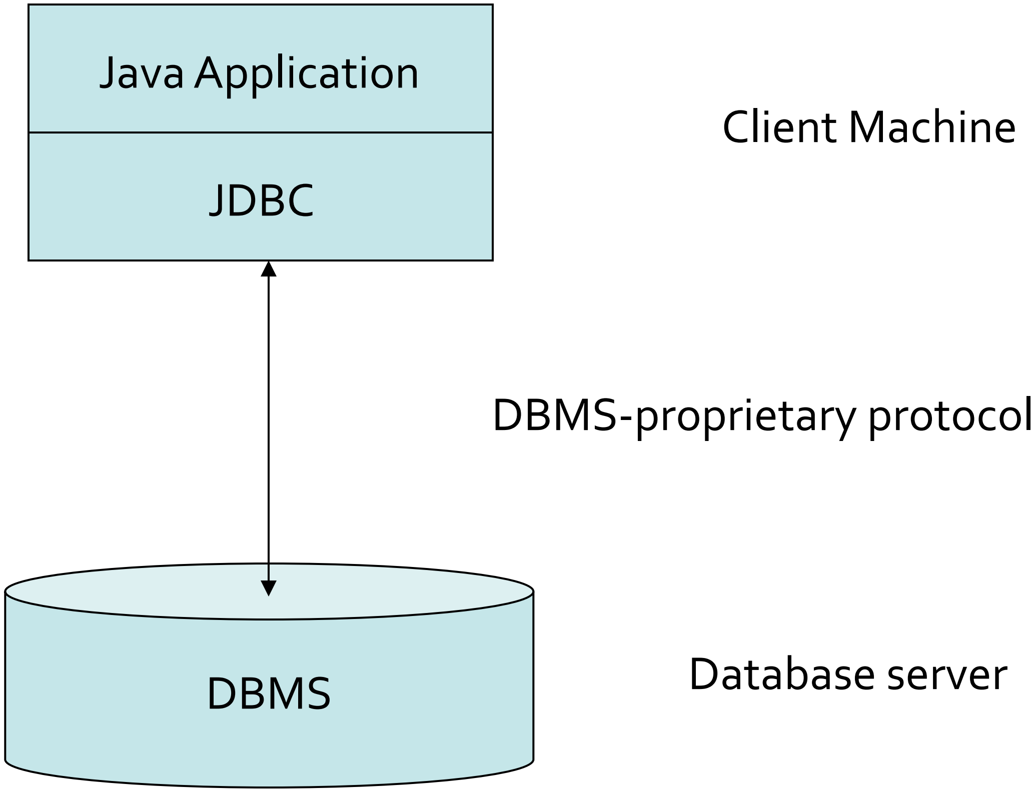
JDBC

- Java API to access database
  1. Connect to a data source
  2. Send queries and update statements
  3. Retrieve and process results

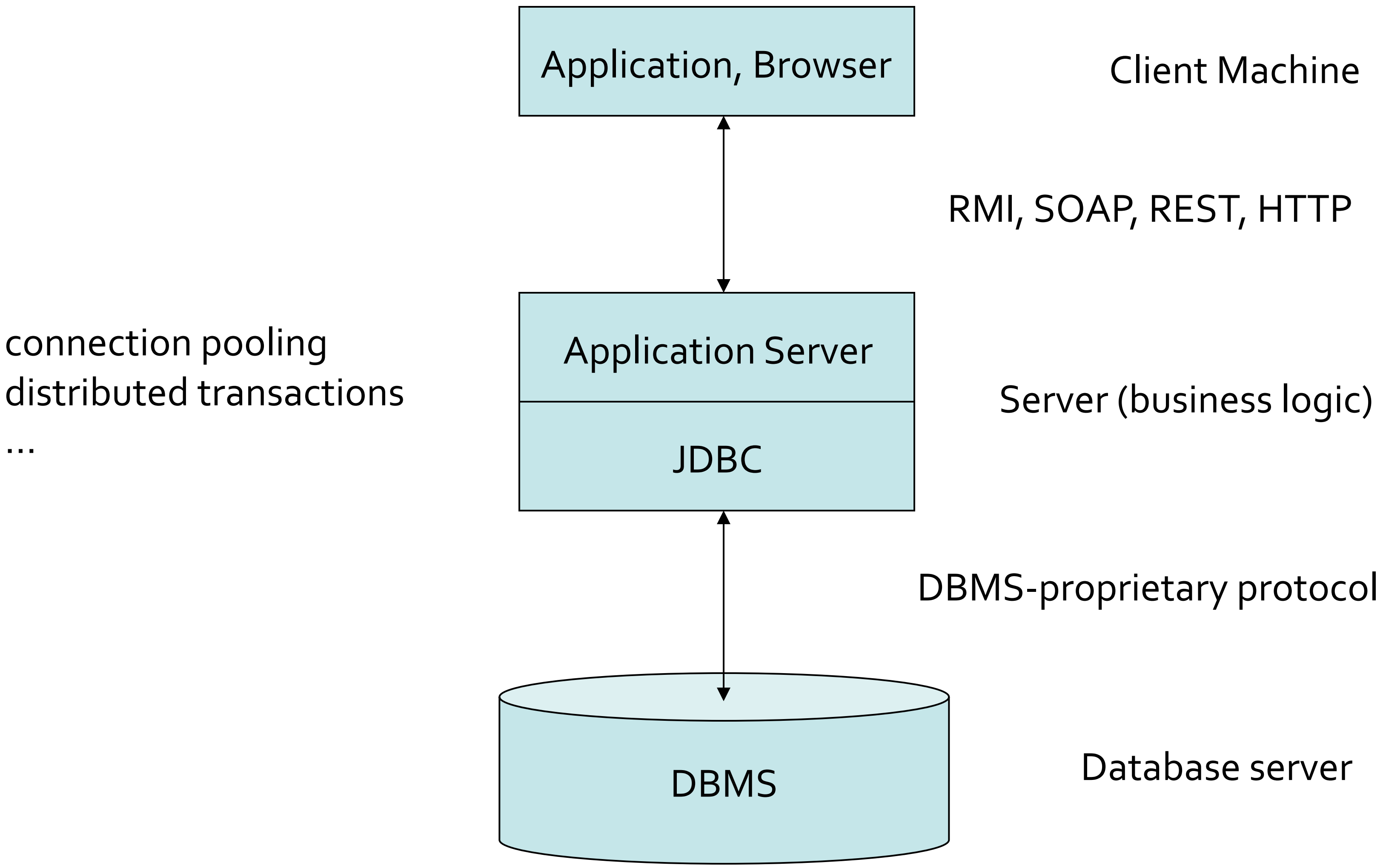
JDBC Example

```
Connection con = DriverManager.getConnection
      ("jdbc:sqlserver://iisqlsrv:database=imdb",
       "myLogin", "myPassword");

Statement stmt = con.createStatement();

ResultSet rs = stmt.executeQuery
      ("SELECT a, b, c FROM Table1");

while (rs.next()) {
      int x = rs.getInt("a");
      String s = rs.getString("b");
      float f = rs.getFloat("c");
}
```

JDBC Architecture
Two-tier model



JDBC Architecture
Three-tier model



Application, Browser

Client Machine

RMI, SOAP, REST, HTTP

connection pooling
distributed transactions
…

Application Server

JDBC

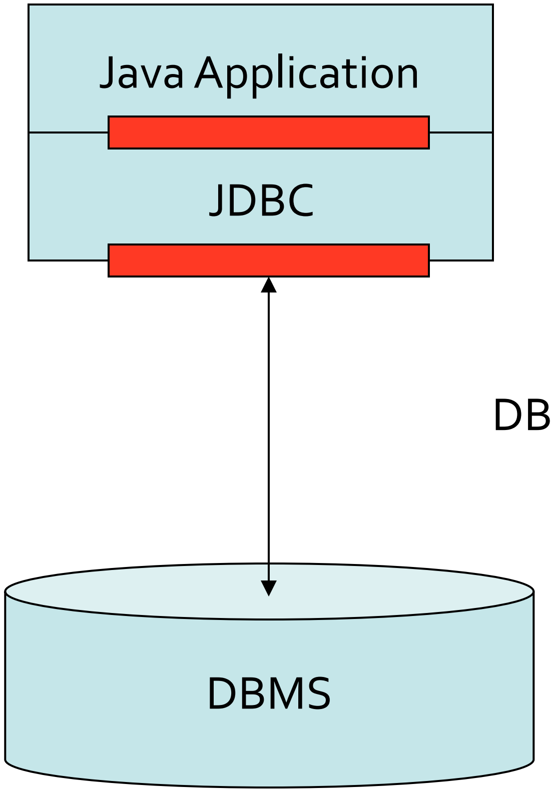
Server (business logic)

DBMS-proprietary protocol

DBMS

Database server

JDBC API



Java Application

JDBC

DBMS

JDBC API for apps

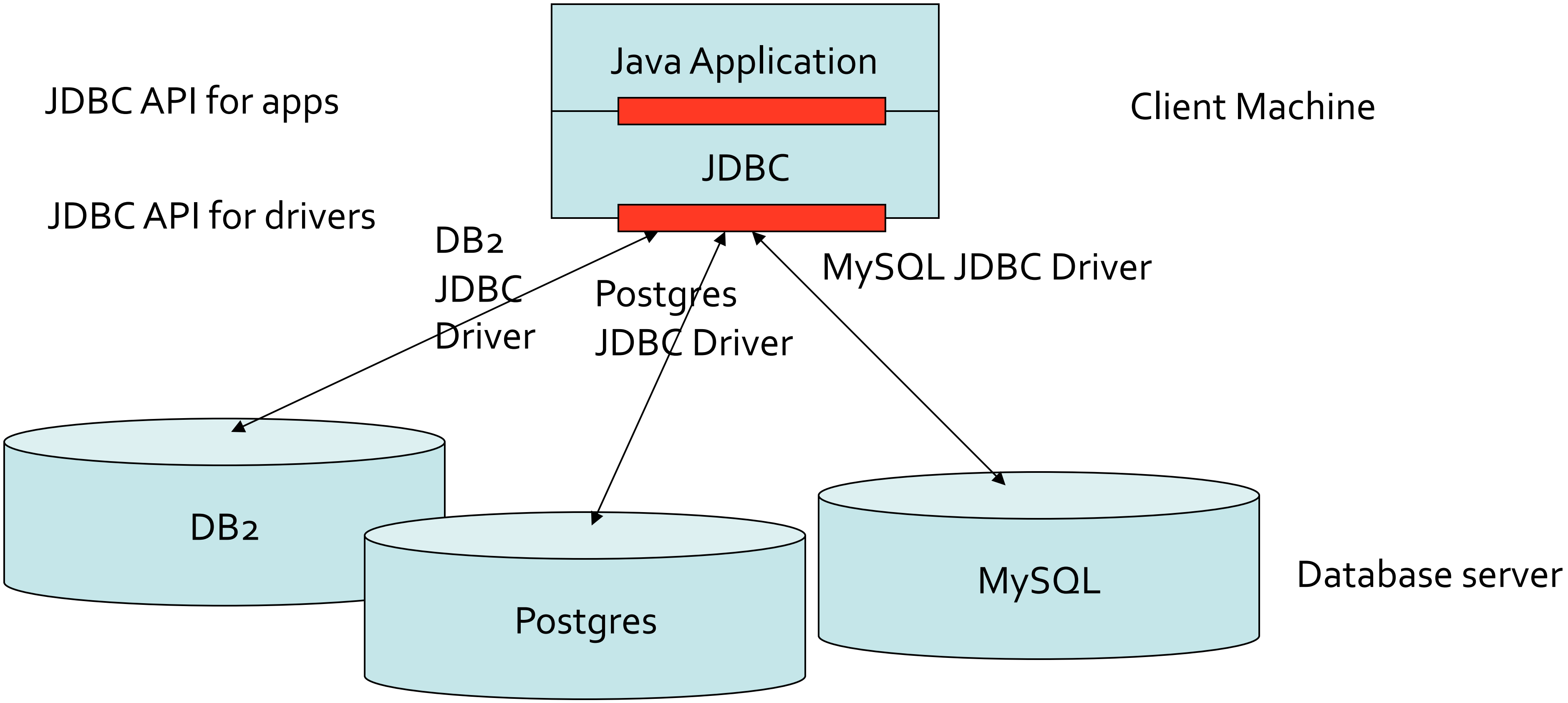
JDBC API for drivers

Client Machine

DBMS-proprietary protocol

Database server

JDBC API


Java Application
JDBC API for apps
Client Machine
JDBC
JDBC API for drivers
DB2
JDBC
Driver
MySQL JDBC Driver
Postgres
JDBC Driver
DB2
Postgres
MySQL
Database server

First need to load driver

- SQL Server Driver
  sqljdbc4.jar
- Postgres Driver
  postgresql-8.3-603.jdbc4.jar

- Put on class path, then tell Java to load it

```
Class.forName(“com.microsoft.sqlserver.jdbc.SQLServerDriver “);
Class.forName(“org.postgresql.Driver”);
Class.forName
  (“com.microsoft.sqlserver.jdbc.SQLServerDriver”);

Connection con = DriverManager.getConnection
      ("jdbc:sqlserver://iisqlsrv:database=imdb",
       "myLogin", "myPassword");

PreparedStatement pstmt = con.prepareStatement
      (“SELECT lname FROM persons WHERE id = ?”);

pstmt.setInt(1, 34);
ResultSet rs = pstmt.executeQuery();

while (rs.next()) {
      String s = rs.getString(“lname");
}

rs.close();

pstmt.close();

con.close();
```

```
Class.forName
  (“com.microsoft.sqlserver.jdbc.SQLServerDriver”);

Connection con = null;

try {
      con = DriverManager.getConnection( … );


      …


} catch (Exception e) {
      e.printStackTrace();
} finally {
      con.close();
}
```

PreparedStatement

```
PreparedStatement pstmt = con.prepareStatement
    (“SELECT lname FROM persons WHERE id = ?”);
…
pstmt.setInt(1, 34);
ResultSet = pstmt.executeQuery();
…

pstmt.setInt(1, 63);
ResultSet = pstmt.executeQuery();

…
```

```
Statement stmt = con.createStatement();
ResultSet rs = stmt.executeQuery
      ("SELECT a, b, c FROM Table1");
```

PreparedStatement

- No need to worry about quotes ‘, “

```
PreparedStatement pstmt = con.prepareStatement
    (“SELECT website FROM shops
       WHERE name = ? OR owner = ?”);
…
pstmt.setString(1, “George’s”);
pstmt.setString(2, “Oh \”wow\”!”);
…
```

```
Statement stmt = con.createStatement();
ResultSet rs = stmt.executeQuery
      ("SELECT website FROM shops
        WHERE name = \”George’s\” OR …");
```

Transactions – Option 1

```
String sBeginTx    = “BEGIN TRANSACTION”;
String sCommitTx   = “COMMIT TRANSACTION”;
String sRollbackTx = “ROLLBACK TRANSACTION”;

PreparedStatement pBeginTx   =
     con.prepareStatement(sBeginTx);
PreparedStatement pCommitTx   =
     con.prepareStatement(sCommitTx);

PreparedStatement pRollbackTx =
     con.prepareStatement(sRollbackTx);
…
pBeginTx.executeUpdate();
…
if (ok) pCommitTx.executeUpdate();
else pRollbackTx.executeUpdate();
```

Transactions – Option 2

```
…
con.setAutoCommit(false);
…
if (ok) con.commit();
else con.rollback();

con.setAutoCommit(true);
```