JDBC

Naveen Kumar K.S
Adith.naveen@gmail.com

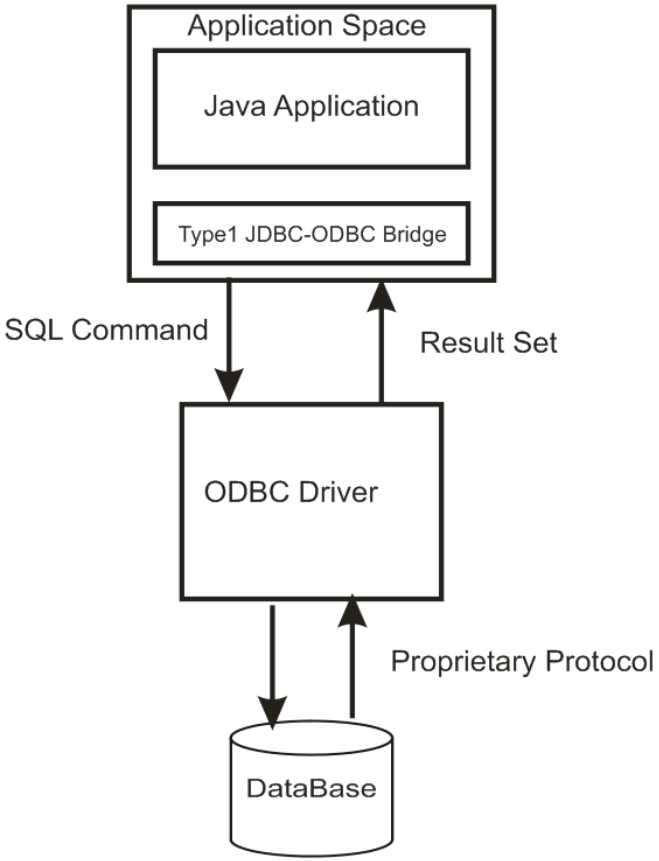
JDBC

- Short for Java Database Connectivity
- Pure Java API used to execute SQL statements
- JDBC provides set of classes and interfaces for developer

JDBC Driver Types

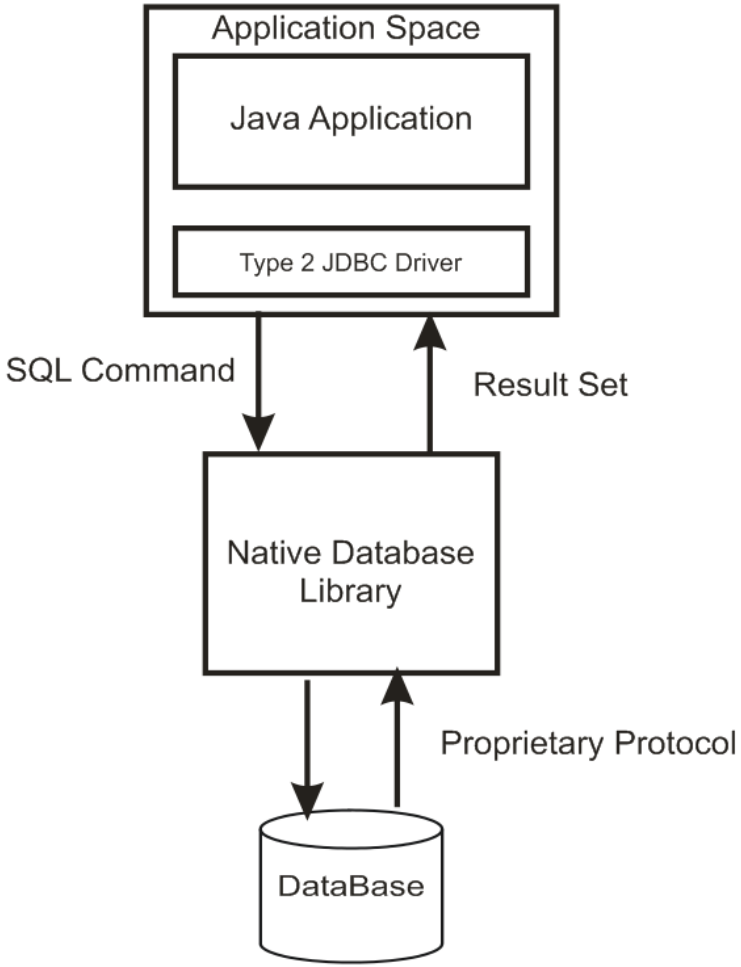
- JDBC-ODBC Bridge, plus ODBC driver
- Native-API, partly Java Driver
- JDBC-net, pure Java Driver
- Native-protocol, Pure Java Driver

JDBC-ODBC Bridge, plus ODBC driver



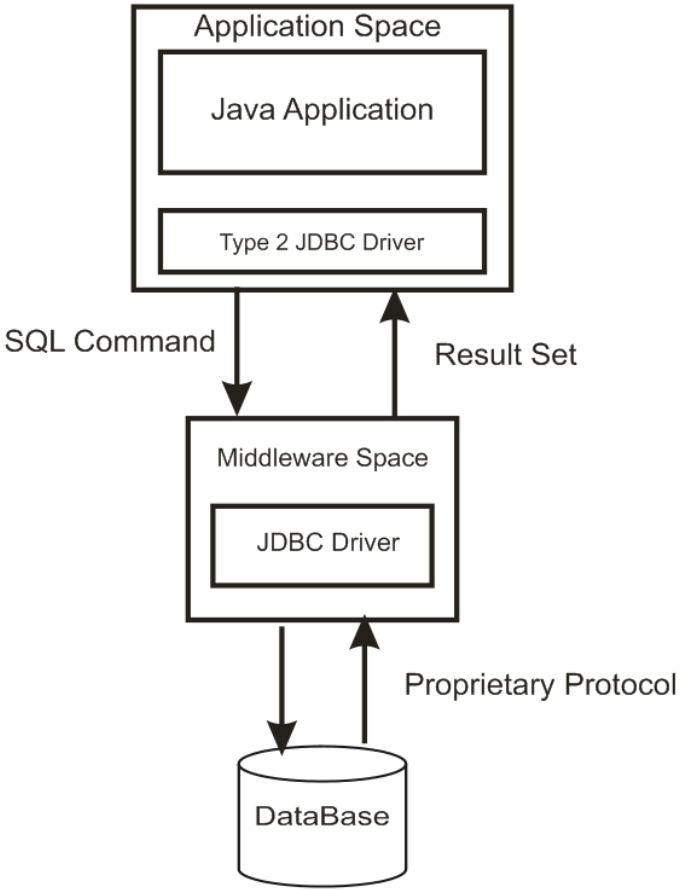
- This is the first type of Driver
- Provided since JDK 1.1
- Used for prototyping

Native-API, partly Java Driver



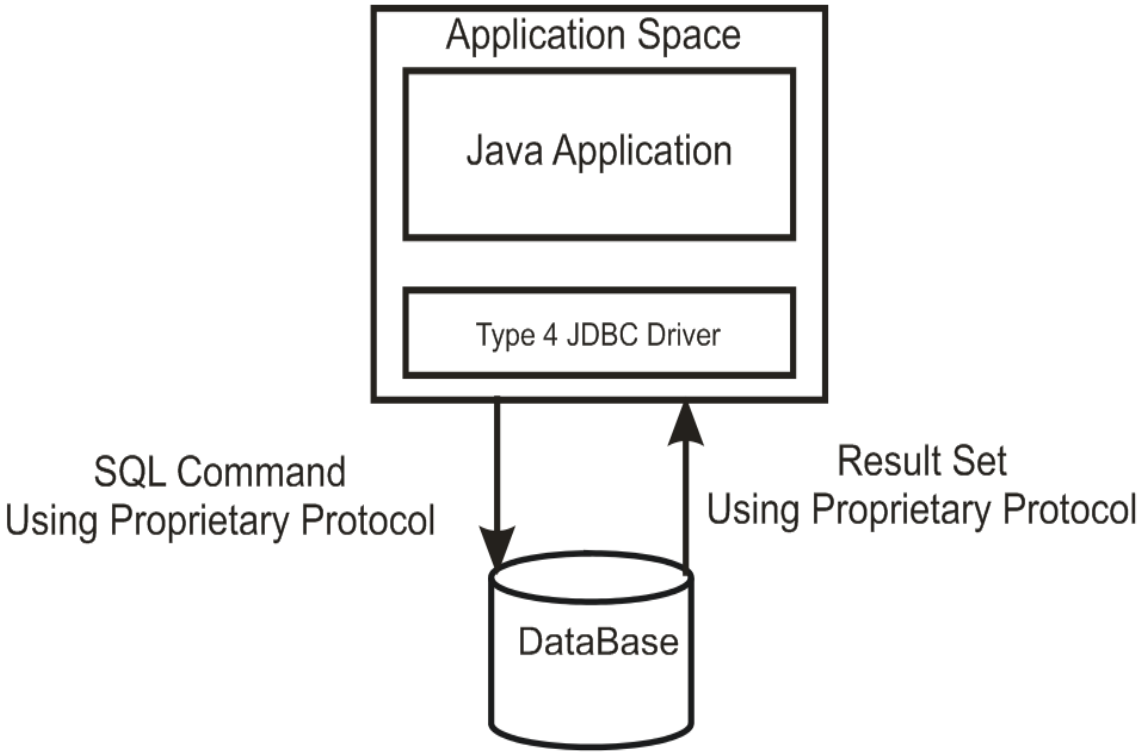
- Converts JDBC Commands into DBMS-specific calls
- Advantage over JDBC-ODBC, since they interface directly with the database

JDBC-net, pure Java Driver



- Three tier solution
- Translates JDBC calls into a DataBase independent protocol
- The results are routed back from middleware to client
- Pure java client
- Swaps DataBase with affection client

Native-protocol, Pure Java Driver



- Pure java drivers
- Direct communication with DB
- No addition translation or middle layers
- Improved performance

Stages of Connection

- Open Connection
- Execute a SQL statements
- Process the results
- Close the connection to DB

Establishing DB Connection

- Load JDBC Driver
  - `Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");`
- Creates instance of the driver
- Register with DriverManager.

Code Fragments

- Open
  - `Connection con = DriverManager.getConnection(url, usrName, passWord);`
- Execute
  - `Statement stmt = con.createStatement();`
- Process
  -
```
While(rs.next()){
    //code
}
```

Continued....

- Close
  - `rs.close();`
  - `con.close();`

Questions….?