JDBC

Author & Presenter -Asfiya Khan

(Senior Technical Trainer)

Agenda

- Why JDBC ?
- What is JDBC ?
- Driver Types in JDBC
- Basic Steps in Using JDBC
- JDBC Architecture

RDBMS

Relational Database:

- A database is a means of storing information in such a way that information can be retrieved from it easily.

- In simplest terms, a relational database is one that presents information in tables with rows and columns.

- From your Java Application you may need to communicate with the database (..to access / modify data etc.)

- Typical Activities your application may need to carry out are:

  - ***Connect to a data source, like a database***
  - ***Send queries and update statements to the database***
  - ***Retrieve and process the results received from the database***

Why JDBC

The need for JDBC :

All databases support SQL (ANSI SQL)

- Different database vendors have introduced their proprietary SQL constructs
- Different database vendors have introduced Application Programming Interfaces for accessing data stored in their respective databases
- Drawback: ?

  If the database changes, all data access logic has to be entirely rewritten. (�much Data access code’ should remain same irrespective of the Database Vendor)

- A *need* was felt to access data from different databases in a consistent and reliable way

What is JDBC

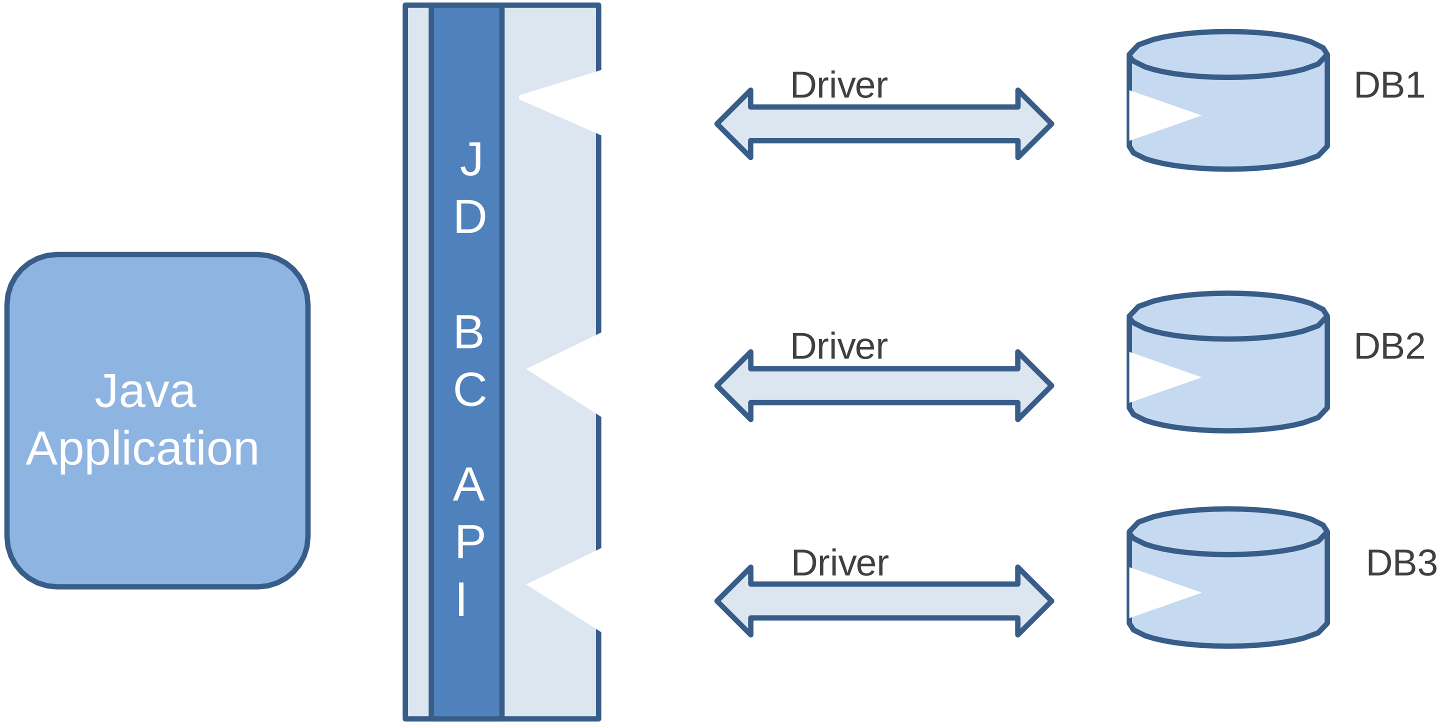
It is not an acronym, but is called Java Database Connectivity

- It is a vendor independent API drafted by Sun to access data from different databases in a consistent and reliable way

- JDBC provides an API by hiding the vendor specific API by introducing the concept of a JDBC driver between the application and the database API Hence, JDBC requires a vendor specific driver

- The *JDBC driver* converts *the* JDBC API calls from the Java application to the DB vendor specific API calls

JDBC API



JDBC API

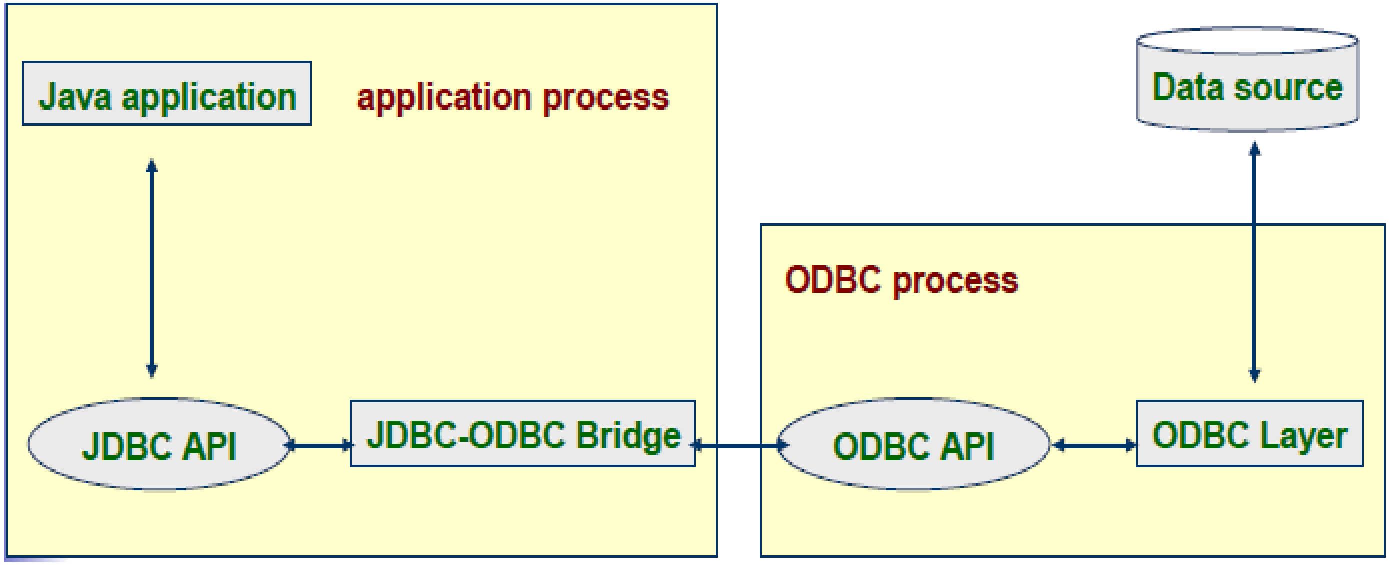
- The JDBC™ API provides programmatic access to relational data from the Java™ programming language.
- Using the JDBC API, applications can execute SQL statements, retrieve results, and propagate changes back to an underlying data source.
- The JDBC API can also interact with multiple data sources in a distributed, heterogeneous environment.

Driver Types in JDBC

There are 4 types of JDBC drivers

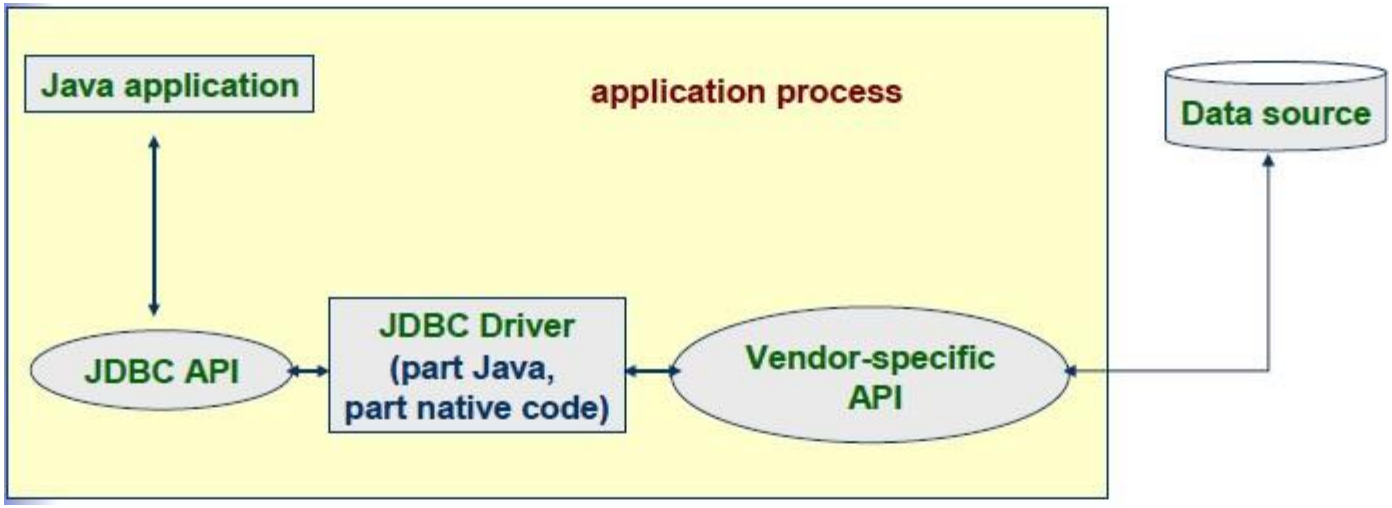
- Type 1 : JDBC – ODBC Bridge
- Type 2 : Native API – Partly Java Driver
- Type 3 : Java – Net Protocol Driver
- Type 4 : All Java Driver

JDBC- ODBC Bridge Driver - Type 1



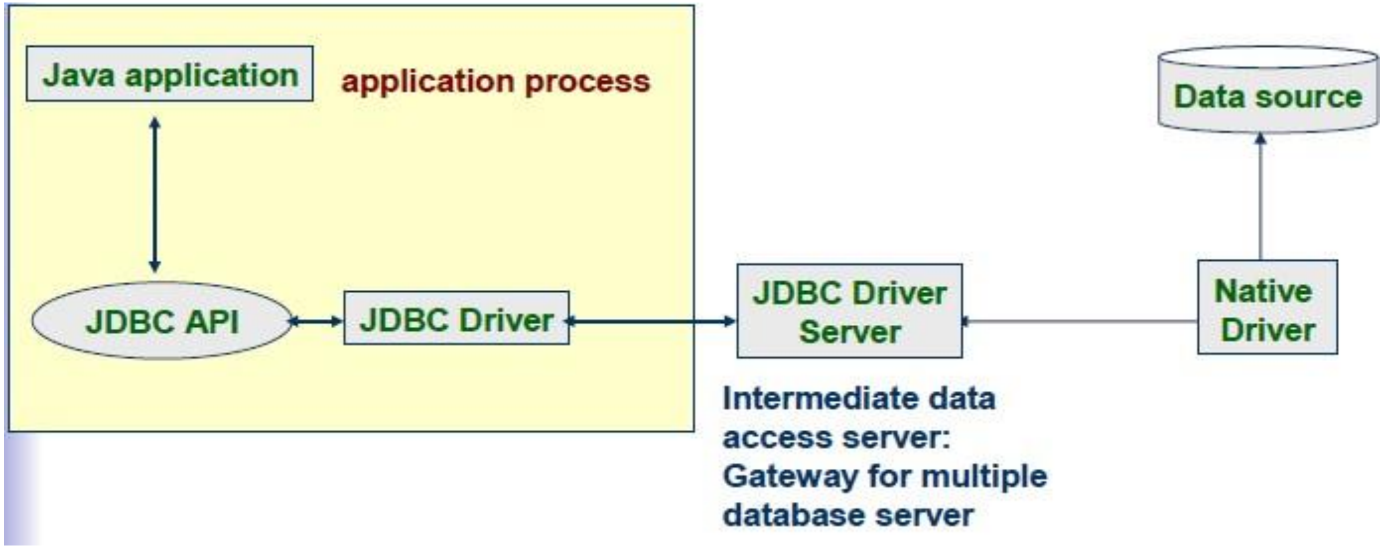
1. Slow performance driver
2. Used for Proto-typing purposes only
3. Inherits limitations of ODBC implementations

Native API – Partly Java Driver – Type 2



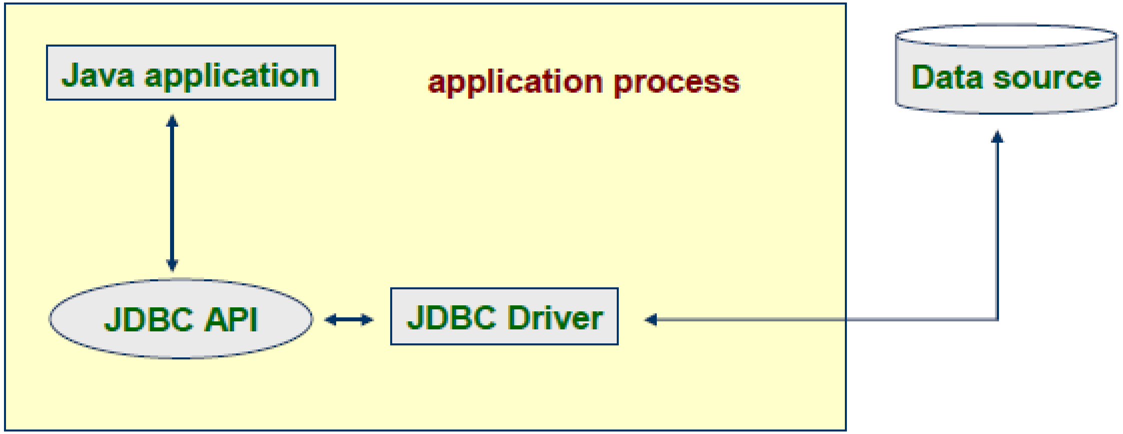
1. Partly java-partly written in native language
2. Native libraries required to be installed on client system
3. Uses native C lib calls for conversion

Java-Net Protocol Driver –Type 3



- Net protocol driver
- JDBC calls are passed through the network to the middle tier server
- The middle tier server then translates the requests to database specific native connectivity interface to further request to the database server
- Deployment is simpler & flexible

All Java Driver – Type 4

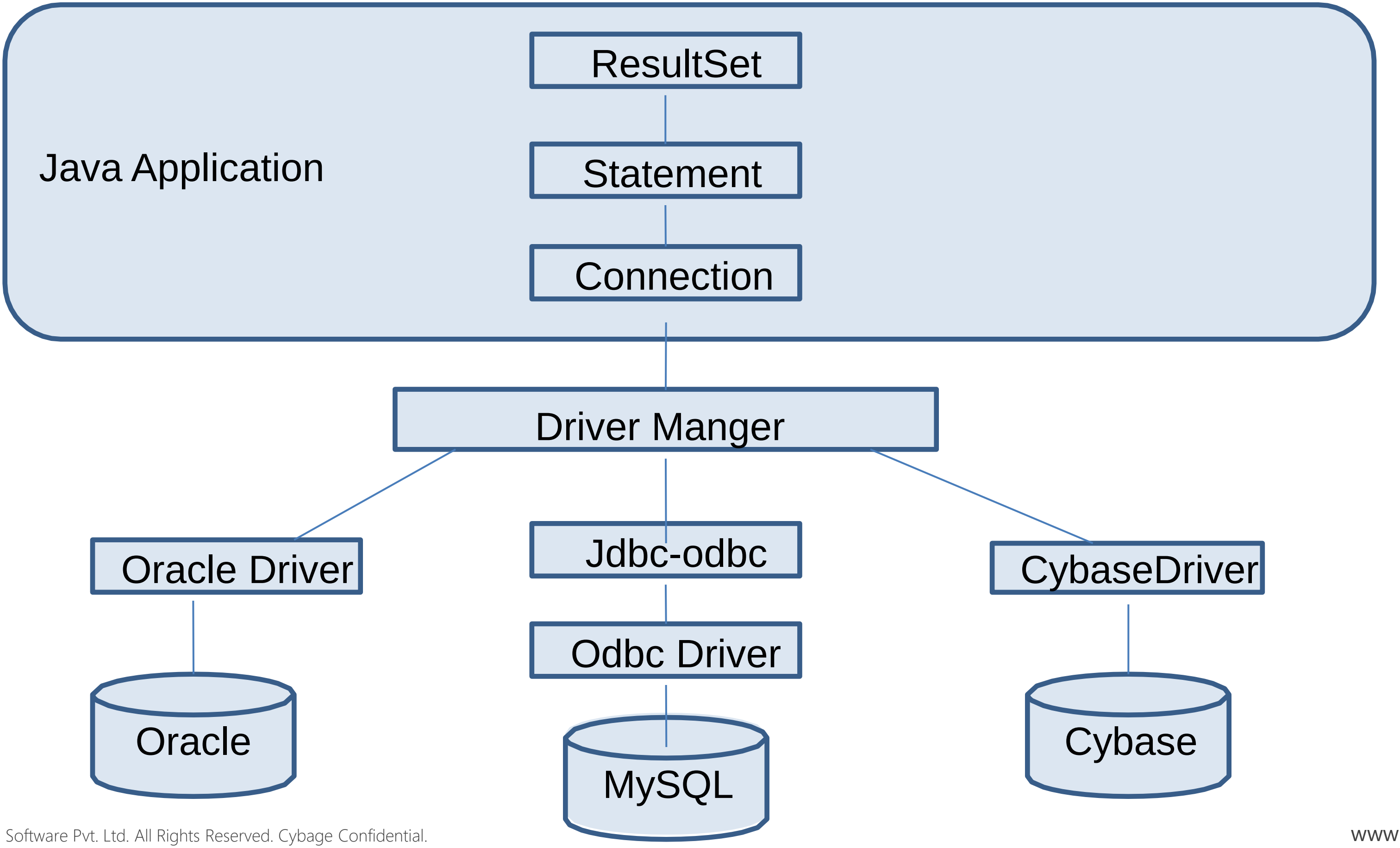


- Is a pure Java driver
- Converts JDBC calls to vendor specific database management system protocol
- client applications can directly communicate with the database server
- Also known as thin driver & performance is very good
- usually comes from DB vendor

Basic Steps In Using JDBC

1. Load the driver
2. Define the connection URL
3. Establish the database connection
4. Create a statement object
5. Execute query
6. Process results
7. Close database connection

JDBC Architecture



JDBC Architecture

JDBC Interfaces

- Driver
- Connection
- Statement
- PreparedStatement
- CallableStatement
- DatabaseMetadata
- ResultSet
- ResultSetMetadata

JDBC classes

- Date
- DriverManager
- Time
- TimeStamp
- Types

Driver Manager

- The DriverManager class is the traditional management layer of JDBC, working between the user and the drivers.

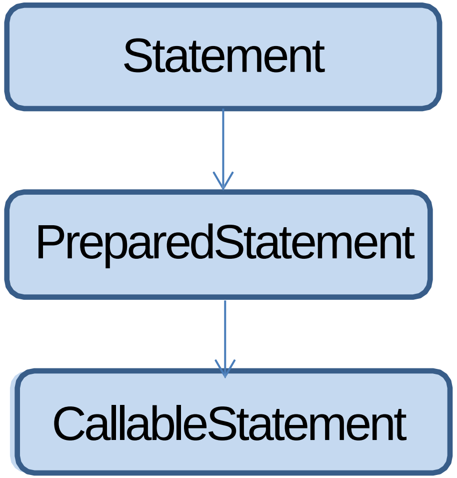
- It keeps track of the drivers that are available and handles establishing a connection between a database and the appropriate driver.

- In addition, the DriverManager class attends to things like driver login time limits and the printing of log and tracing messages.

- It is possible that an application may need to interact with multiple databases created by different vendors

- The JDBC Driver Manager provides the ability to communicate with multiple databases and keep track of which driver is needed for which database

Statement



Statement

PreparedStatement

CallableStatement

Methods in Statement interface:

boolean execute(String sql)

Resultset executeQuery()

int executeUpdate()

Using Statement

The Statament object is used to execute SQL queries against the database.

There are 3 types of Statament objects :

1. Statement :
   - For executing simple SQL statements

2. PreparedStatement
   - For executing pre-compiled SQL statements

3. CallableStatament
   -For executing database stored procedures

ANY
QUESTIONS
?

CYBAGE
Delivering Value. Scientifically.

Thank You!

Any Questions ?

www.cybage.com