

Desenvolvimento de aplicações com SQL

Fernando Lobo

Base de Dados, Universidade do Algarve

Desenvolvimento de aplicações

- Até agora temos dado comandos de SQL de forma interactiva num interpretador de comandos.
- Quando se desenvolve aplicações utilizamos uma linguagem convencional, como C ou Java, e incorporamos comandos de SQL.

Utilização de SQL em programas

- O mais comum é utilizar-se uma *Application Programming Interface* (API)
- Vamos ver alguns exemplos em Java, disponíveis na página web da disciplina.

Programas em Java com SQL

- O mais comum é usar JDBC → Java Database Connectivity.
- JDBC é uma API que facilita a escrita de programas em Java com SQL.

JDBC

- JDBC é uma colecção de classes e interfaces.
- É necessário usar um *driver* específico para um SGBD concreto.
- Usa-se a classe `DriverManager` para estabelecer uma conexão a uma base de dados.

Arquitetura JDBC

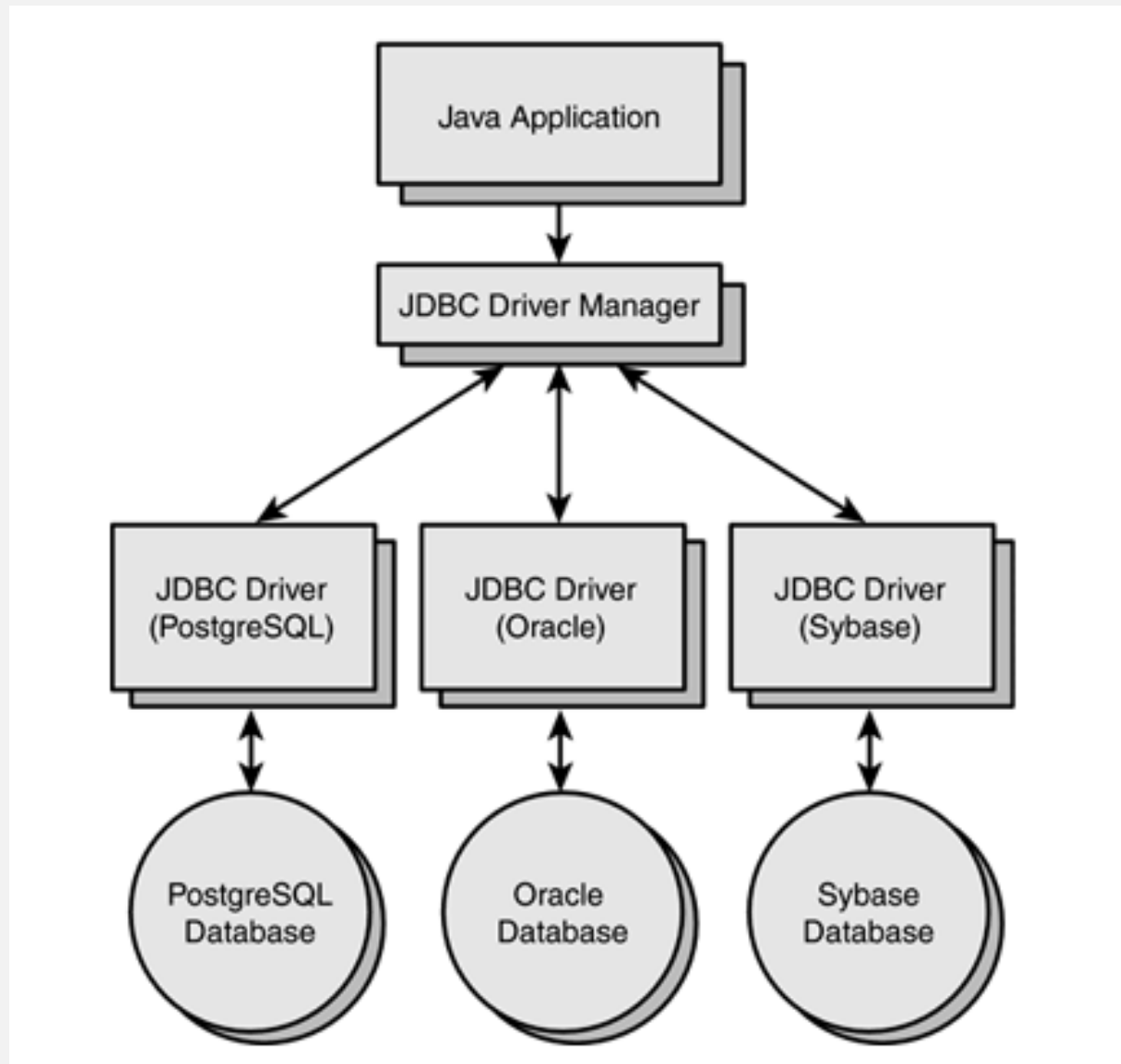


Imagem retirada do livro PostgreSQL de Kerry Douglas e Susan Douglas.

Passos necessários

- 1 Importar o package java.sql

```
import java.sql.*;
```

- 2 Estabelecer uma conexão a uma BD. Exemplo para uma conexão local.

```
String url = "jdbc:postgresql://localhost:5432/filmes";  
Connection c = DriverManager.getConnection(url);
```

Exemplo de url para uma conexão remota,

```
String url = "jdbc:postgresql://www.mysite.com/filmes";
```


- 3 Criar um objecto da classe `Statement`. Utiliza-se o método `createStatement` do objecto da classe `Connection`.

```
Statement stmt = c.createStatement();
```

④ Executar um *Query* ou um *Update*.

- ▶ Update em JDBC refere-se a qualquer operação que altere o estado da base de dados.
- ▶ Pode ser um comando SQL de UPDATE, INSERT, DELETE, CREATE TABLE, etc.

A classe `Statement` tem os métodos

- ▶ `executeQuery`
- ▶ `executeUpdate`

Exemplos

- `executeQuery` retorna um objecto da classe `ResultSet`

```
String cmd = "SELECT nome, ano, duracao " +  
            "FROM filmes " +  
            "WHERE nomeEstudio='Universal';";  
ResultSet rs = stmt.executeQuery(cmd);
```

- `executeUpdate` retorna o número de linhas afectadas pela operação

```
String cmd = "INSERT INTO estudios(nome,morada) " +  
            "VALUES('UALg', 'Campus de Gambelas');";  
int n rows = stmt.executeUpdate(cmd);
```

5 Iterar sobre o ResultSet

- ▶ ResultSet tem uma série de métodos.
- ▶ Tipicamente usa-se o método `next()` dentro de um ciclo `while` para iterar sobre os tuplos/linhas do ResultSet
- ▶ Em cada iteração usamos métodos (`getString`, `getInt`, `getFloat`, etc) consoante o tipo de dados apropriado, para aceder aos componentes de um tuplo.

```
ResultSet rs = stmt.executeQuery(cmd);
while ( rs.next() ) {
    String nome = rs.getString("nome");
    int ano = rs.getInt("ano");
    ...
}
```

⑥ Fechar os objectos criados

```
rs.close();  
stmt.close();  
c.close();
```

Exemplos

Vejamos alguns exemplos, disponíveis na página web da disciplina.

- prog1.java
- insereEstudioUalg.java
- insereEstudio.java
- printStarWars1977.java
- printFilmesUniversal.java
- printFilmesDeEstudio.java

Mais informação

- Há muito mais detalhes que não vimos.
- Pesquisar por JDBC no google.
- Para SGBDs concretos, pesquisar de forma apropriada (Java with PostgreSQL, Java with MySQL, etc.)