

Monday - Week 3

JDBC



Swapping out one JDBC for another

- JDBC's all match the same pattern
- They are thus interchangeable
- Some may require slightly different parameters



SQLite JDBC

- sqlite-jdbc
- org.xerial
- version: 3.25.2
- <https://mvnrepository.com/artifact/org.xerial/sqlite-jdbc>



mySQL JDBC

- mysql-connector-java
- mysql
- version: 8.0.15
- <https://stackoverflow.com/questions/30651830/use-jdbc-mysql-connector-in-intellij-idea>



PostgreSQL JDBC

- postgresql
- org.postgresql
- version: 42.2.5
- <https://mvnrepository.com/artifact/org.postgresql/postgresql>

Couchbase (CouchDB) JDBC

- Seems this one is not so easy to just download
- <https://docs.couchbase.com/server/4.1/connectors/odbc-jdbc-drivers.html>



MongoDB JDBC

- mongodb-driver-sync
- org.mongodb
- version 3.10.1
- <https://mongodb.github.io/mongo-java-driver/>

Monday - Week 3

JDBC Transaction Management



Transactions

- Unit of Work
- Any time you execute a command against the database
- A single action may be granulated into many transactions



ACID

- Atomic
- Consistency
- Isolation
- Durability



Atomic

- A Transaction is either 100% successful or it is a failure



Consistency

- Before the transaction, the database is in a consistent state
- After the transaction, the database must also be in a consistent state



Isolation

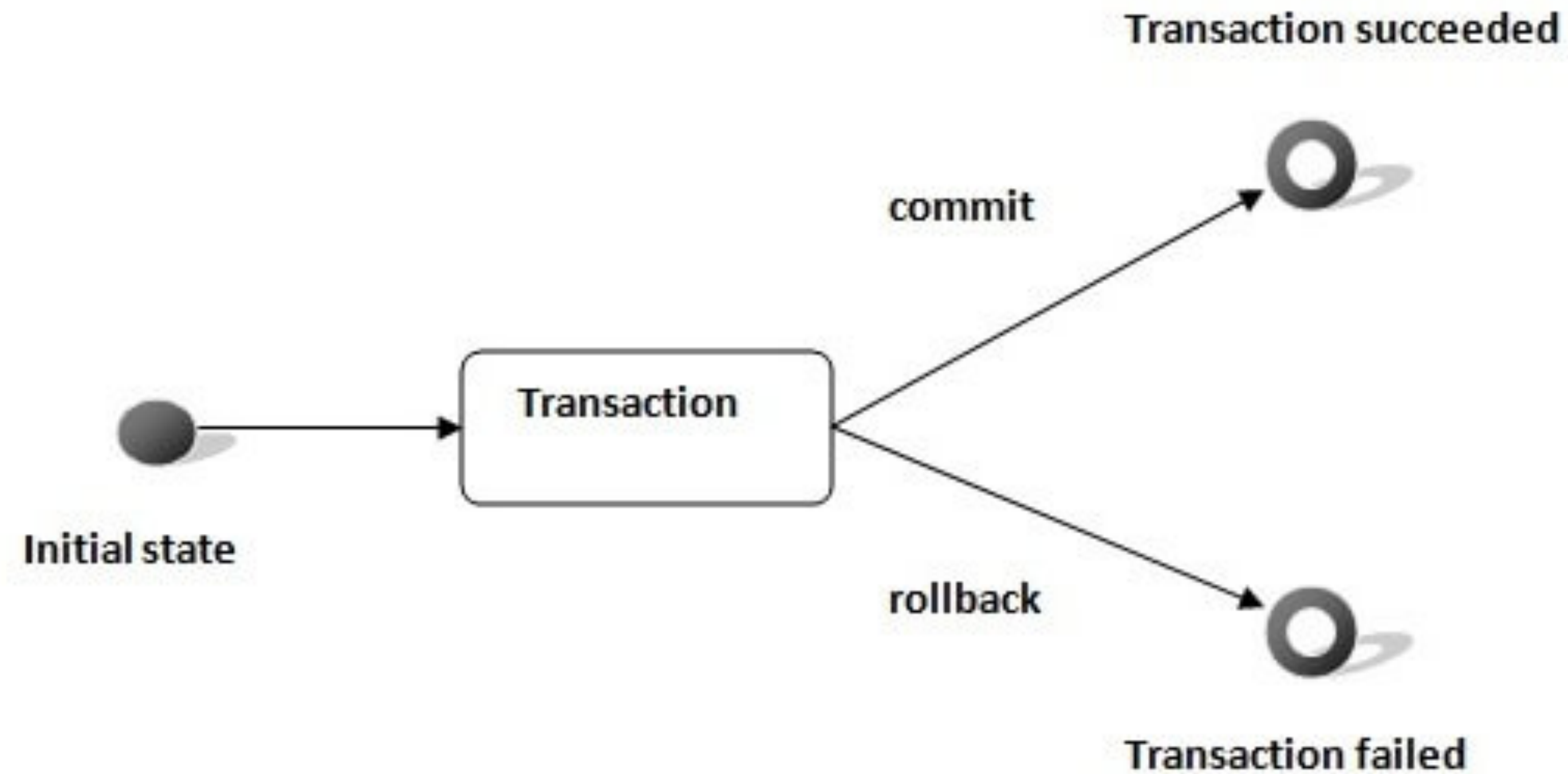
- One transaction is its own entity and does not interact (or need) another transaction to be executed



Durability

- A transaction is a permanent change to the database, it must not be easily undone

Transaction success vs. failure



<https://www.javatpoint.com/transaction-management-in-jdbc>



Rollback?

- <https://www.baeldung.com/java-jdbc>
- Part 7
- (Next Slide)

An example ...

```
1 String updatePositionSql = "UPDATE employees SET position=? WHERE emp_id=?";
2 PreparedStatement pstmt = con.prepareStatement(updatePositionSql);
3 pstmt.setString(1, "lead developer");
4 pstmt.setInt(2, 1);
5
6 String updateSalarySql = "UPDATE employees SET salary=? WHERE emp_id=?";
7 PreparedStatement pstmt2 = con.prepareStatement(updateSalarySql);
8 pstmt.setDouble(1, 3000);
9 pstmt.setInt(2, 1);
10
11 boolean autoCommit = con.getAutoCommit();
12 try {
13     con.setAutoCommit(false);
14     pstmt.executeUpdate();
15     pstmt2.executeUpdate();
16     con.commit();
17 } catch (SQLException exc) {
18     con.rollback();
19 } finally {
20     con.setAutoCommit(autoCommit);
21 }
```

Monday - Week 3

JSON Primer



JSON - JavaScript Object Notation

- Composed of:
 - name/value pairs
 - a list of values



name/vale pairs

```
{  
  name: "Craig",  
  age: 34,  
  city: "Oslo"  
}
```



a list of values

```
[  
    "Hiking",  
    "Photography",  
    "Gaming"  
]
```



Combinations are possible

```
{  
  name: "Craig",  
  age: 34,  
  city: "Oslo"  
  hobbies: [  
    "Hiking",  
    "Photography",  
    "Gaming"  
  ]  
}
```



Converting JSON to/from Strings

- Using JSONObject and JSONArray
- We covered most of this last week
- It should start to make sense as to why you had to do all those conversion before...



Converting JSON to/from POJO

- <https://github.com/google/gson>
 - toJson()
 - fromJson()

Monday - Week 3

Tasks

Reading Task

- Read through:
<https://www.json.org/>
- and:
https://www.w3schools.com/js/js_json_intro.asp



Task 17 - Week Task - Group

For the week you will be building a Java app for storing (and accessing) contact information in a Database.

You will need to design and implement tables to store this information:

- First Name
- Last Name
- Contact Number (multiple possible - work/home/mobile/etc)
- E-mail Address (multiple possible - work/personal/etc)
- Home Address (may be shared by people)
- Date of Birth
- Family/Blood Relationship (Only close ones - mother/father/brother/sister)

Task 17 - Week Task - Group

- Divide yourselves into groups of three or four
- Later in the week we will cover the basics of WebApps
- For now design the database and transactions
- Some design considerations:
 - The app must perform CRUD operations for all tables
 - The app must allow me to search by name/surname/telephone number.
 - For a given contact it must show the related contacts and how they are related.