

DAM - Accés a dades

Tema 2 - Mapeig objecte-relacional (ORM)

Roberto Sanz Requena

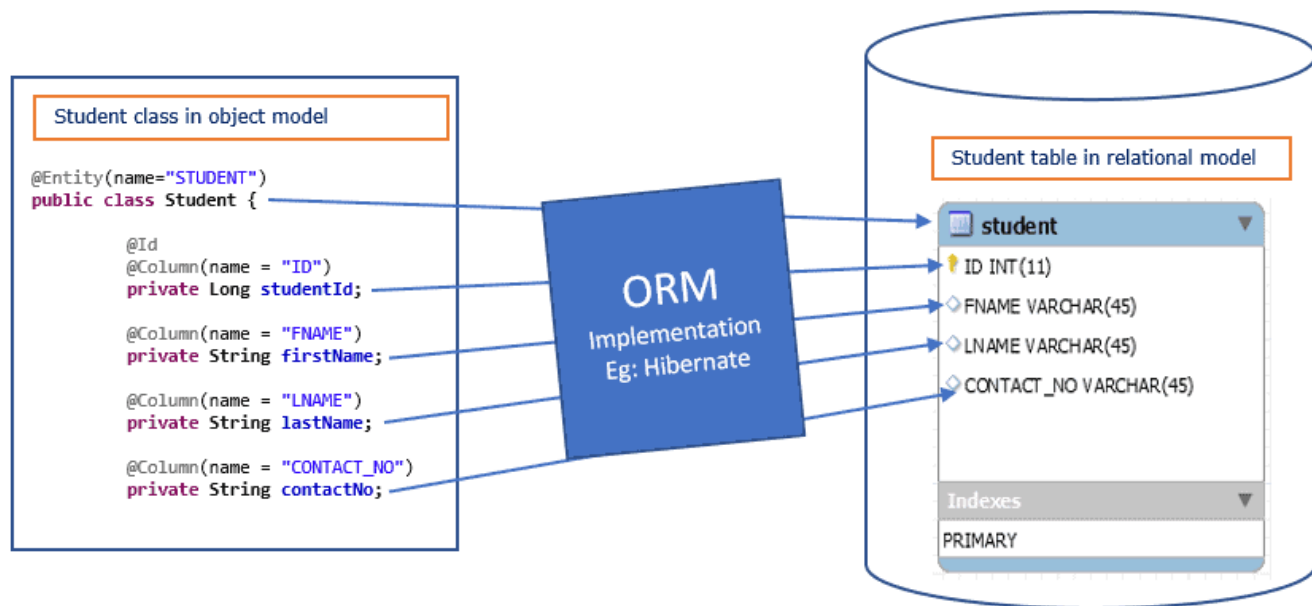
rsanz@florida-uni.es

Índex

- 1. Context**
- 2. Ferramentes de mapeig objecte-relacional**
- 3. Desenvolupament amb Java, Hibernate i MySQL**

1. Context

- Desfasament objecte-relacional (object-relational impedance mismatch): diferències entre al POO i la base de dades relacional.



ORM implements responsibility of mapping the Object to Relational Model.

 JavaByDeveloper

1. Context

- Desfasament objecte-relacional (object-relational impedance mismatch)
- Llenguatges de programació: llenguatge orientat a objectes (p.e. Java) vs llenguatge d'accés a dades (SQL).
- Tipus de dades: en POO tipus de dades primitius i de referència (objectes) vs en accés a dades sols tipus primitius.
- En POO un objecte pot contindre multitud d'atributs (inclosos altres objectes a la seua vegada tinguen atributs): aquest paradigma de programació permet construir qualsevol lògica de negoci. El problema ve quan és necessari fer persistir un objecte en una BDD relacional, que no admet aquesta flexibilitat i requereix fer consultes combinades i joins entre taules, complicant les sentències SQL.

1. Context

- Desfasament objecte-relacional (object-relational impedance mismatch)

¿BDD relacional o POO?	¿BDD relacional o POO?
Encapsulat	No encapsulat
Herència i polimorfisme	No herència ni polimorfisme
Comportament	No comportament
Tipus complets	Tipus primitius
Instància	Fila
No persistència	Persistència

https://es.wikipedia.org/wiki/Adaptaci%C3%B3n_de_impedancias_objeto-relacional

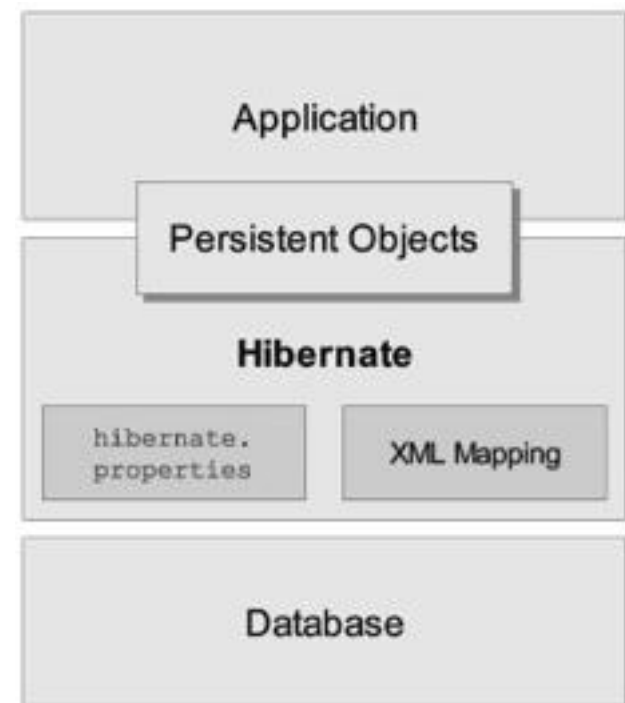
1. Context

- Desfasament objecte-relacional (object-relational impedance mismatch)
 - Alternatives:
 - Bases de dades No-SQL
 - Bases de dades objecte-relacionals
 - Ferramentes de mapeig objecte-relacional

2. Ferramentes de mapeig objecte-relacional

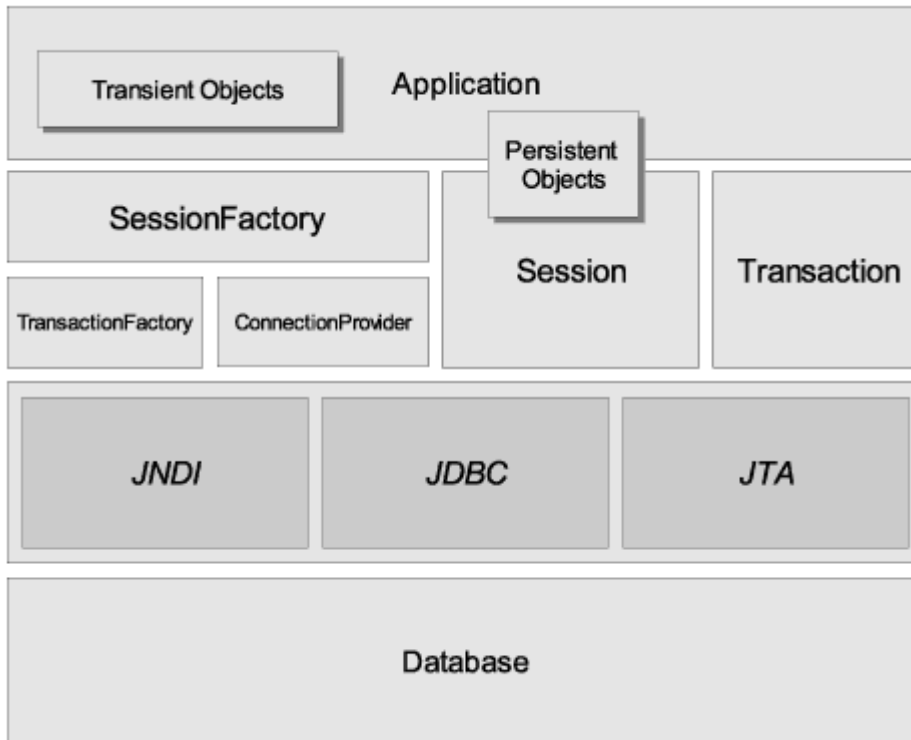
Exemple: Hibernate (Java), NHibernate (C# para .NET)

- Traduir les operacions de POO a operacions sobre la BDD relacional.
- Generar automàticament les sentències SQL necessàries més habituals (HQL, Hibernate Query Language).
- Augmenta el temps d'execució.



2. Ferramentes de mapeig objecte-relacional

Hibernate: arquitectura i funcionament



Objectes transitoris (aplicació Java)



Objectes persistents (BDD)

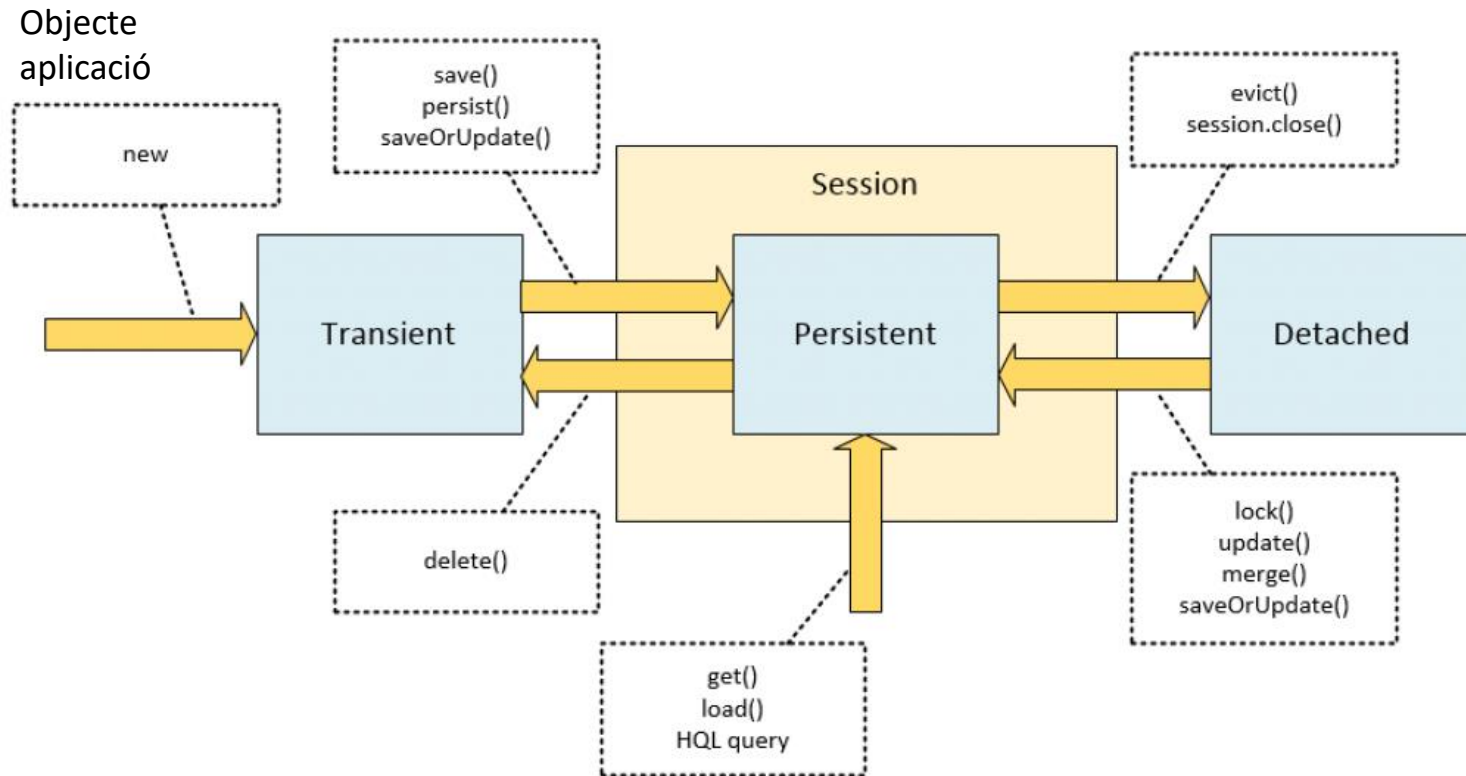
Flux de treball (objectes necessaris):

- Configuració Hibernate
- Session Factory
- Session
- Transaction
- Query
- Criteria

<https://hibernate.org/>
<http://java-white-box.blogspot.com/2014/06/hibernate-que-es-hibernate-arquitectura.html>

2. Ferramentes de mapeig objecte-relacional

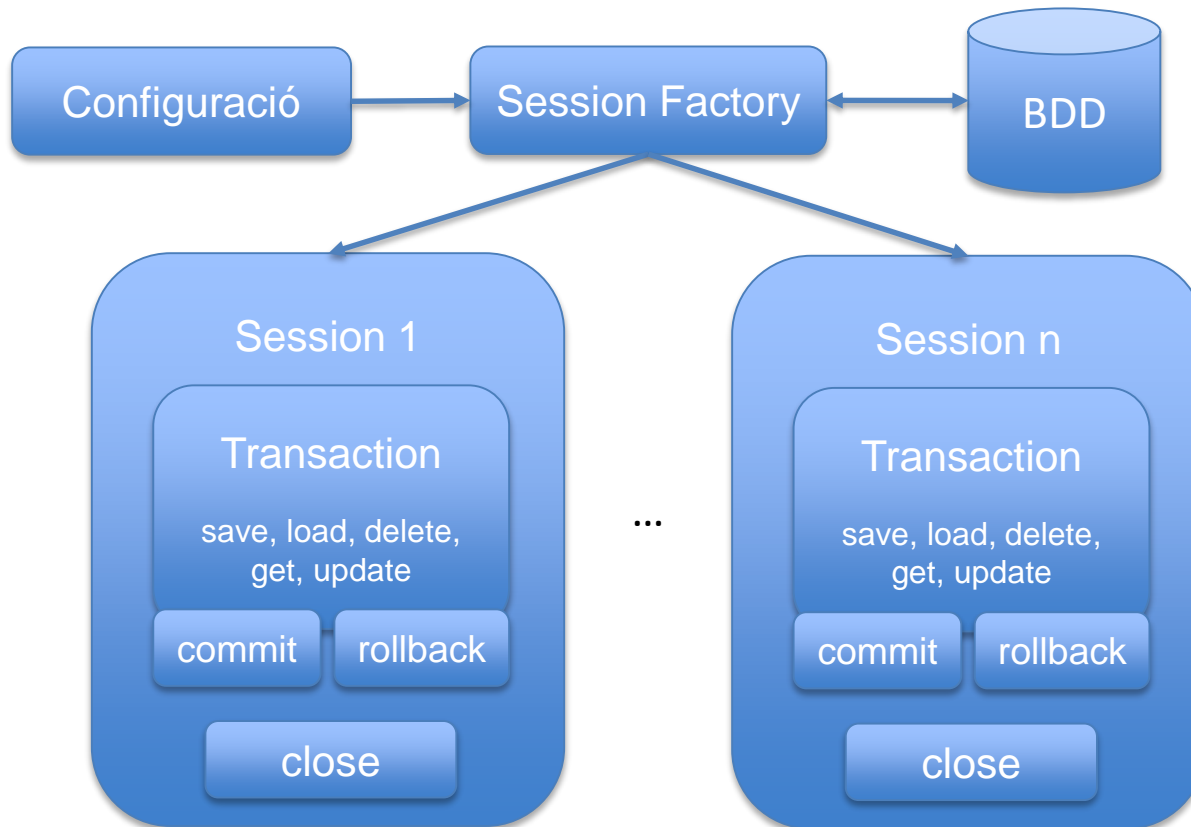
Hibernate: arquitectura i funcionament



<https://hibernate.org/>
<https://www.mysoftkey.com/hibernate/life-cycle-of-objects-in-hibernate/>
<https://www.baeldung.com/hibernate-save-persist-update-merge-saveorupdate>

2. Ferramentes de mapeig objecte-relacional

Hibernate: arquitectura i funcionament



3. Desenvolupament amb Java, Hibernate i MySQL

Hibernate: exemple

1. Preparar BDD MySQL
2. Preparar projecte Eclipse: importar .jar Hibernate i MySQL JDBC
3. Crear fitxer de configuració de Hibernate
4. Crear classes Java que representen les taules de la BDD (POJO)
5. Crear fitxer de mapeig de Hibernate
6. Crear i gestionar sessions de Hibernate
7. Treballar amb transaccions
8. Realitzar operacions CRUD (crear, llegir, actualitzar i esborrar)

<https://hibernate.org/>
<https://www.codejava.net/frameworks/hibernate/writing-a-basic-hibernate-based-program-with-eclipse>

3. Desenvolupament amb Java, Hibernate i MySQL

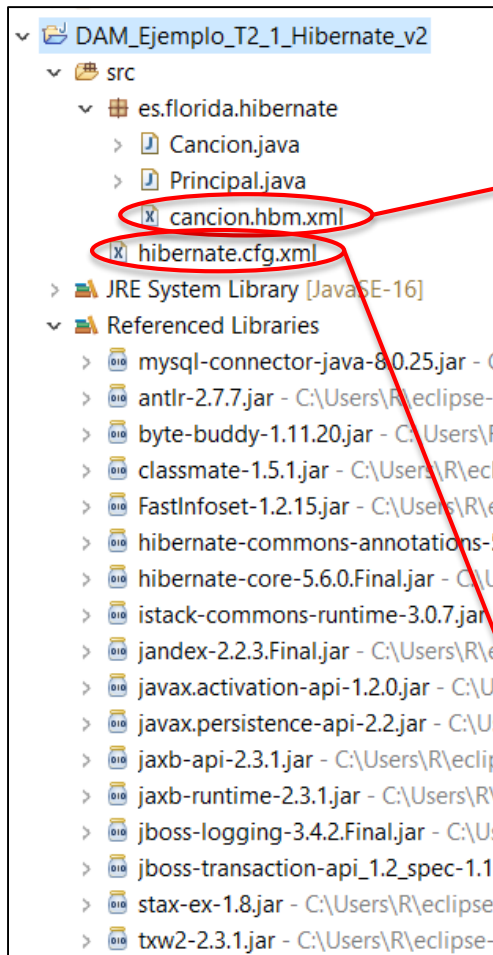
Configuració:

- Descarregar versió 5.6: <https://sourceforge.net/projects/hibernate/>
- Utilitzar només “lib/required”
- Importar tots els .jar en el Build Path del projecte en Classpath (**NO** en Modulepath)
- Importar el .jar de MySQL en Classpath

<https://hibernate.org/>
<https://www.codejava.net/frameworks/hibernate/writing-a-basic-hibernate-based-program-with-eclipse>
<https://examples.javacodegeeks.com/enterprise-java/hibernate/hibernate-crud-operations-tutorial/>

3. Desenvolupament amb Java, Hibernate i MySQL

Configuració:



Objecte

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE hibernate-mapping PUBLIC
    "-//Hibernate/Hibernate Mapping DTD 3.0//EN"
    "http://www.hibernate.org/dtd/hibernate-mapping-3.0.dtd">
<hibernate-mapping package="es.florida.hibernate">
    <class name="Cancion" table="canciones">
        <id name="id" column="id">
            <generator class="increment"/>
        </id>
        <property name="titulo" type="string" column="titulo"/>
        <property name="artista" type="string" column="artista"/>
        <property name="anyo" type="int" column="anyo"/>
        <property name="formato" type="string" column="formato"/>
    </class>
</hibernate-mapping>
```

BDD

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE hibernate-configuration PUBLIC
    "-//Hibernate/Hibernate Configuration DTD 3.0//EN"
    "http://www.hibernate.org/dtd/hibernate-configuration-3.0.dtd">
<hibernate-configuration>
    <session-factory>
        <!-- Database connection settings -->
        <property name="connection.driver_class">com.mysql.jdbc.Driver</property>
        <property name="connection.url">jdbc:mysql://localhost:3306/canciones</property>
        <property name="connection.username">root</property>
        <property name="connection.password"></property>
        <!-- <property name="show_sql">true</property> -->
        <mapping resource="es/florida/hibernate/cancion.hbm.xml"/>
    </session-factory>
</hibernate-configuration>
```

3. Desenvolupament amb Java, Hibernate i MySQL

```
// Carrega la configuracio i crea un session factory
Configuration configuration = new Configuration().configure("hibernate.cfg.xml");
configuration.addClass(Cancion.class);
ServiceRegistry registry = new
    StandardServiceRegistryBuilder().applySettings(configuration.getProperties()).build();
SessionFactory sessionFactory = configuration.buildSessionFactory(registry);

// Obri una nova sessió de la session factory
Session session = sessionFactory.openSession();
session.beginTransaction();

//Ací les operacio/ns CRUD (crear, llegir, actualitzar, borrar)

//Commit de la transacció i tanca de sessió
session.getTransaction().commit();
session.close();
```

```
//Crear nou objecte
Cancion can = new Cancion("Cornfield chase", "Hans Zimmer", 2014, "RAW");
Serializable id = session.save(can);
```

```
//Recuperar llista d'objectes
List listaCanciones = new ArrayList();
listaCanciones = session.createQuery("FROM Cancion").list();
```

¡ATENCIÓN!
Nom de l'objecte mapejat,
NO de la taula de la BDD

3. Desenvolupament amb Java, Hibernate i MySQL

```
//Recuperar un objecte a partir del seu id  
Cancion cancion = (Cancion) session.get(Cancion.class, 1);
```

```
// Actualitza la informació d'un objecte donat el seu id  
Cancion cancion = (Cancion) session.load(Cancion.class, 5);  
cancion.setAnyo(2010);  
cancion.setFormato("MP3");  
session.update(cancion);
```

```
//Borrar un objecte donat el seu id  
Cancion cancion = new Cancion();  
cancion.setId(5);  
session.delete(cancion);
```

```
//Borrar tot  
Query queryObject = session.createQuery("DELETE FROM canciones");  
queryObject.executeUpdate();
```

¡ATENCIÓN!

Ací sí que va el nom de la taula de la BDD

3. Desenvolupament amb Java, Hibernate i MySQL

Exemple sentència SELECT:

```
Statement stmt = con.createStatement();  
ResultSet rs = stmt.executeQuery("SELECT * FROM country");
```

Exemple sentència INSERT:

```
PreparedStatement psInsertar = con.prepareStatement("INSERT INTO tabla (campo1,campo2) VALUES (?,?)");  
psInsertar.setString(1,valorCampo1);  
psInsertar.setString(2,valorCampo2);  
int resultadoInsertar = psInsertar.executeUpdate();
```

Exemple sentència UPDATE:

```
PreparedStatement psActualizar = con.prepareStatement("UPDATE tabla SET campo1 = '" + valorCampo1 +  
    "', campo2 = '" + valorCampo2 + "' WHERE id = 5");  
int resultadoActualizar = psActualizar.executeUpdate();
```

Exemple sentència DELETE:

```
PreparedStatement psBorrar = con.prepareStatement("DELETE FROM tabla WHERE id = 5");  
int resultadoBorrar = psBorrar.executeUpdate();
```


Activitat Entregable 5 - Hibernate

Presentació de l'Activitat Entregable 5 (AE05_T2_1_Hibernate)