
Object-Oriented and Object-Relational Databases

The Object-Relational Impedance Mismatch

Knowledge Objectives

1. Explain the concept of impedance mismatch
2. Explain the historical background / needs behind object data models (ODMs) for databases
3. Enumerate seven desirable features an ODM must have
4. Enumerate ten of the OODBMSs golden rules
5. Name five standard object-oriented features from SQL:1999
6. Give two reasons why object methods may suit better than using attributes
7. Enumerate two pros and cons of choosing OIDs instead of PKs
8. Name two pros and cons of using REFs instead of FKs
9. Enumerate two pros and cons of using type inheritance in Oracle 11g

Understanding Objectives

1. Discuss the performance issues the object-relational impedance mismatch may generate

Application Objectives

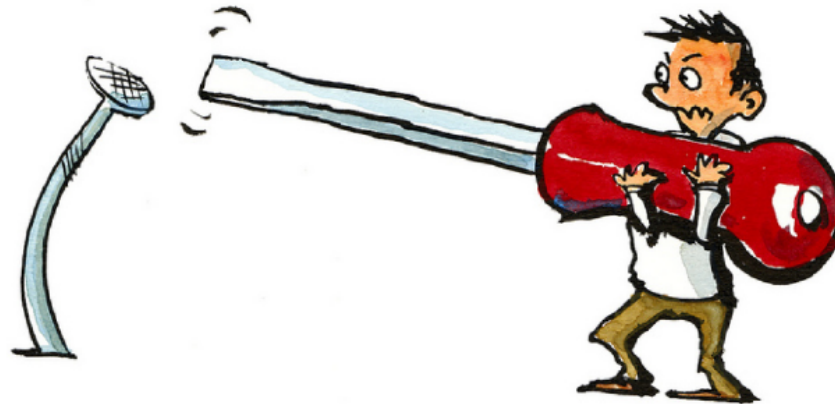
1. Model simple object-relational databases by considering:
 - Objects
 - References to objects
 - Inheritance
 - Multi-valued attributes
2. Design simple (i.e., using basic syntax) object views over relational data and justify its suitability for a given specification

THE OBJECT-RELATIONAL IMPEDANCE MISMATCH **MOTIVATION**

Impedance Mismatch: Some History

Of hammers and nails...

The Law of the Hammer



If the only tool you have is a hammer,
everything looks like a nail.

Abraham Maslow - The Psychology of Science - 1966

Petra Selmer, *Advances in Data Management* 2012

Impedance Mismatch: Some History

The Law of the Relational Database



By HikingArtist.com

If the only tool you have is a relational database,
everything looks like a table.

A Walk in Graph Databases - 2012

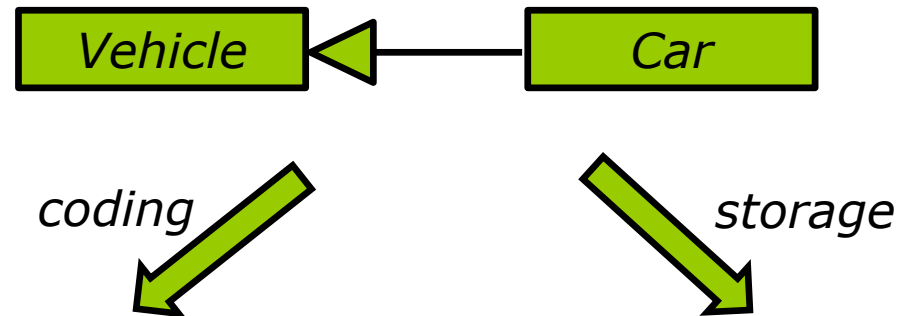
Petra Selmer, *Advances in Data Management 2012*

Alberto Abelló & Oscar Romero

Definition

- ❑ When two technologies interact but they are grounded in different models or paradigms, a translation between the elements of both models is then mandatory to enable interoperability. This overhead (in performance) to execute the map between both models is known as **impedance mismatch**
- ❑ In this course, we will study the potential impedance mismatch **between an application and the database**

Example



```
class Vehicle {
    protected String brand = "Ford";           // Vehicle attribute
    public void honk() {                         // Vehicle method
        System.out.println("Tuut, tuut!");
    }
}

class Car extends Vehicle {
    private String modelName = "Mustang";       // Car
    public static void main(String[] args) {

        // Create a myCar object
        Car myCar = new Car();

        // Call the honk() method (from the Vehicle class) on the myCar object
        myCar.honk();

        // Display the value of the brand attribute (from the Vehicle class)
        System.out.println(myCar.brand + " " + myCar.modelName);
    }
}
```

*Transformation
(JDBC, Hibernate, etc.)*

RDBMS (e.g., PostgreSQL)

Vehicle

PK			subclass

Object-Relational Impedance Mismatch

- ❑ The first impedance mismatch identified and studied, back in the late 70s, was **the object-relational impedance mismatch** between object-oriented programming languages and relational databases. To overcome it, there was a huge amount of work to map object-oriented concepts to relational concepts. As result:
 - Some concepts were **equivalent** (or very similar) and straightforwardly mapped,
 - Some concepts were in **conflict** with each other and no direct mapping could be established – i.e., an inherent contradiction between both models remain,
 - Some concepts were not similar, yet **compatible**, and a mapping could be established in such a way we can avoid contradiction
- ❑ **Being aware of such mappings it is important to acknowledge their impact on performance**
 - The largest the mismatch, the more code to execute to translate from one concept to another

Objectives

- ❑ Understand the position of the main communities with regard to the object-relational impedance mismatch
 - The object-oriented community that advocated for object-oriented databases to minimize the impedance mismatch
 - The object-relational community that advocated to preserve the benefits of the relational model and define and agree a mapping between both models
- ❑ Understand the consequences, performance-wise, of the decisions made by each community and their proposals

Other Types of Impedance Mismatch

- ❑ NOSQL has introduced new data models were introduced
 - Graph data model
 - Document-oriented databases
 - Key-value (\sim hash tables)
 - Streams (\sim vectors and matrixes)
- ❑ Further, other programming paradigms have also appeared or become trendy
- ❑ The impedance mismatch is accordingly a complex issue that may compromise the performance of any [Big Data] system

Summary

- ❑ Impedance mismatch
- ❑ Two main approaches for object data modeling for databases: OODBMSs Vs. ORDBMSs
- ❑ The object-relational impedance mismatch
 - Agreements
 - Disagreements
 - Conflicts
- ❑ The object-relational layer was mostly built on top of the relational layer (the impedance mismatch still holds, even if hidden by automatic processes)
 - PostgreSQL was the first object-relational DBMS, since then, most RDBMS followed

Bibliography

- Atkinson M., Bancilhon F., DeWitt D., Dittrich K., Maier D., and Zdonik S. The Object-Oriented Database System Manifesto. In Proc. 1st Int. Conf. on Deductive and Object-Oriented Databases, North Holland, 1990.
- Stonebraker M., Rowe L., Lindsay B., Gray J., Carey M., Brodie M., Bernstein P., and Beech D. Third generation database system manifesto. ACM SIGMOD Rec., 19(3), 1990.
- Rowe L. and Stonebraker M. The Postgres Data Model. In Proc. 13th Int. Conf. on Very Large Data Bases, 1987.
- Dietrich S.W. and Urban S.D. An Advanced Course in Database Systems: Beyond Relational Databases. Prentice Hall, Upper Saddle River, NJ, 2005.
- Object Database Management Systems: The Resource Portal for Education and Research, <http://odbms.org>