Entity-Relationship Diagram

PB007 Software engineering I

Stanislav Chren

25. 10. 2021

Entity-Relationship Diagram

Entity-Relationship Diagram (ERD) is a data model representing the logical structure of a database.

Its main components are:

- Entities (*Entity types*)
- Relations (*Relation types*)
- Attributes (*Attribute types*)

Class diagram vs. ERD

Class diagram

- models both data and operations
- classes are connected with different types of relationsips (associations, dependencies, generalization, aggregation, composition)
- usually represent business domain concepts

Entity-relationship model

- models data only
- contains only simple relationships
- represents database tables

Object-oriented technologies manipulate with the data through object interaction. *Relational-based technologies* manipulate with the data through relation algebra (SQL).

Object-Relational Mapping

Object-Relational Mapping (ORM) is a technique for conversion of data between relational database and object-oriented language.

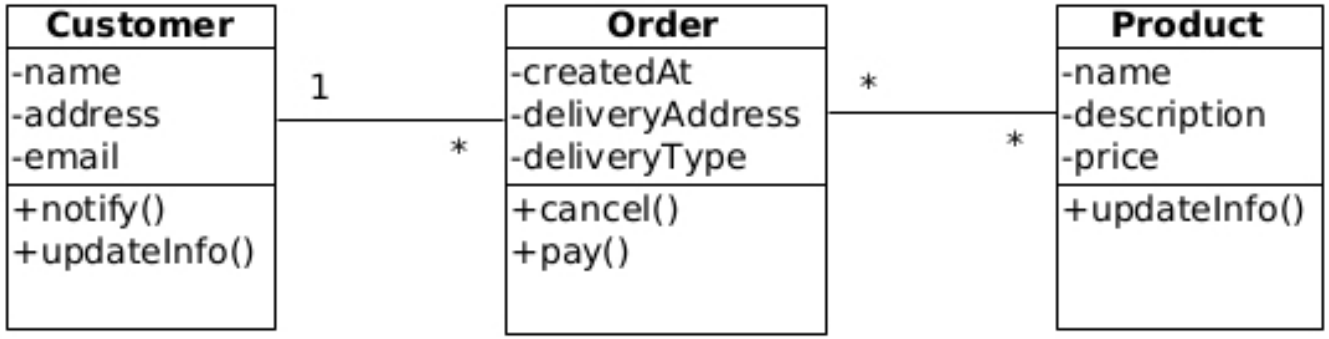
- persistent class defines an entity type (table)
- object defines an entity (table row)
- class attributes become entity attributes (table columns)
- association/aggregation/composition between classes define relation (connection via foreign keys)
- several ways to deal with inheritance: 1:1 mapping, merge to superclass, propagation to subclasses

Note:

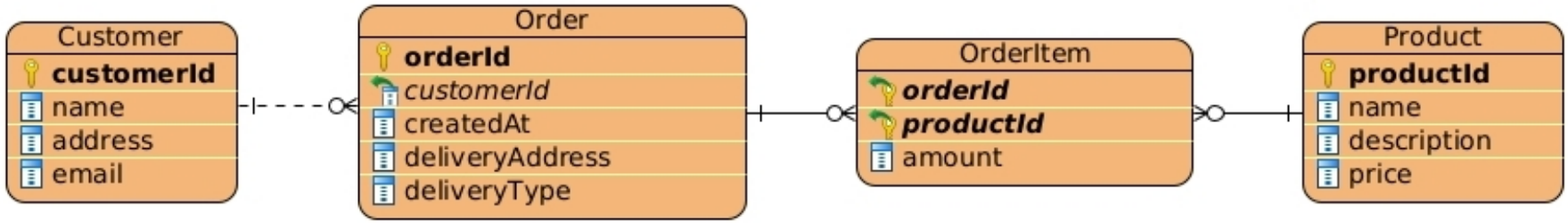
- single class can be mapped to multiple tables
- multiple classes can be mapped to single table
- not all classes must be persistent

Object-Relational Mapping II

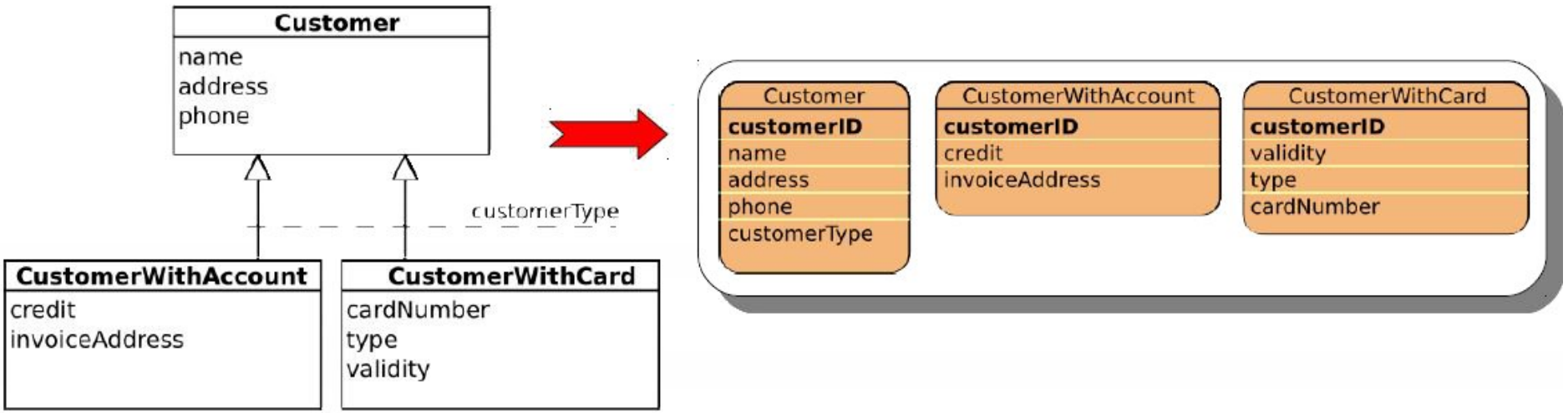
Class diagram:



ERD:

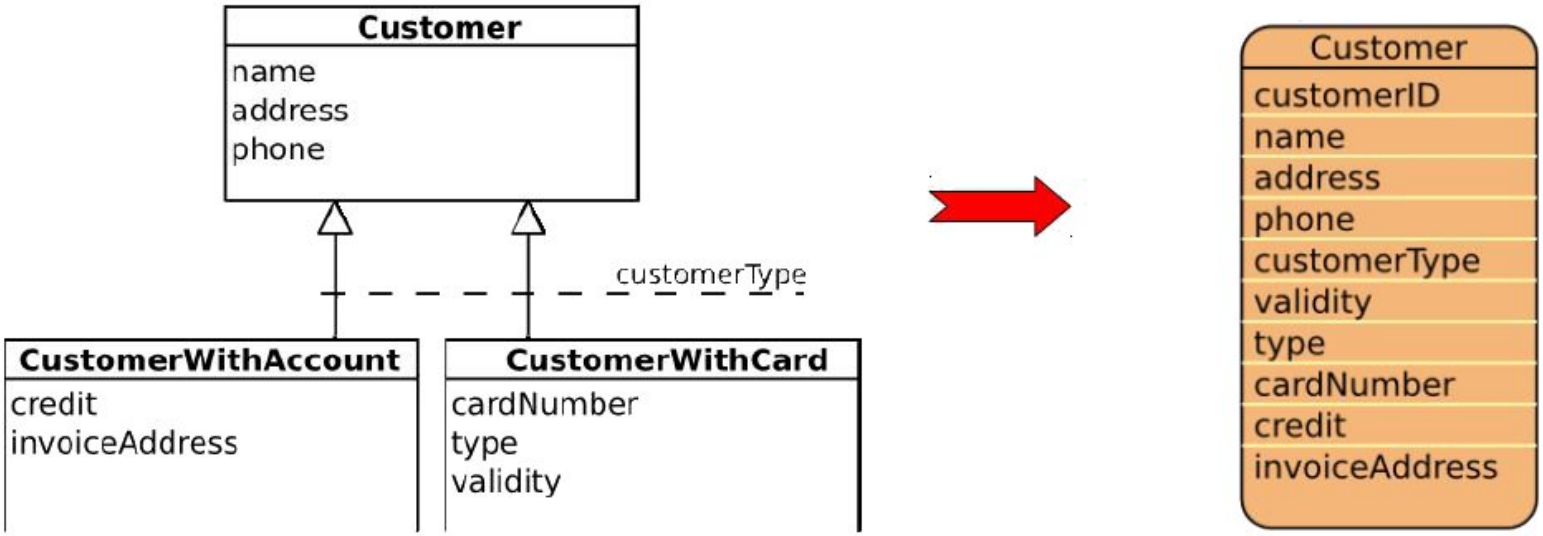


ORM - Inheritance - 1:1 mapping



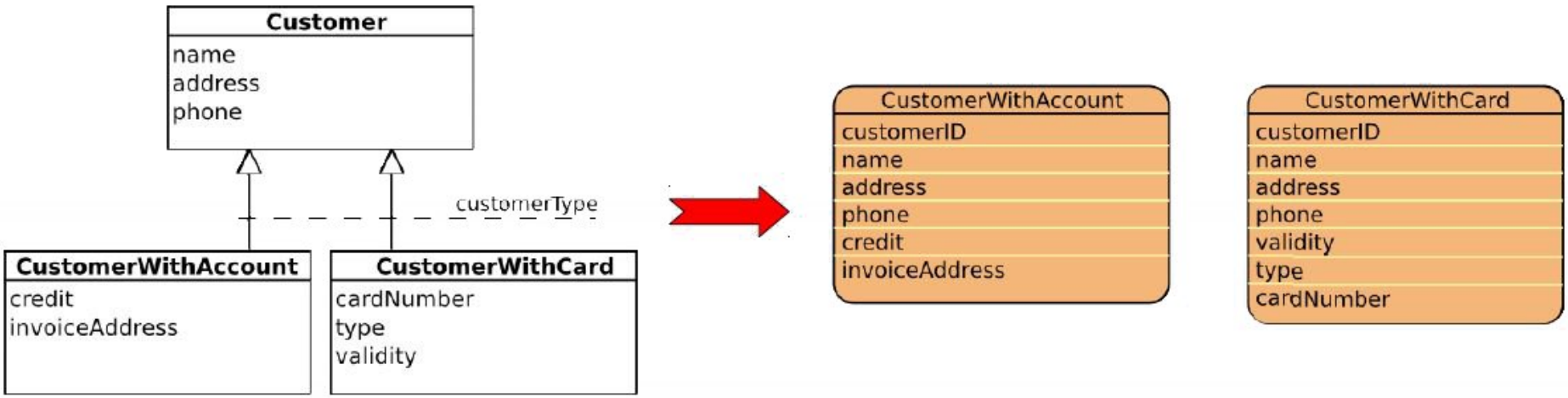
- Each class becomes a table
- Addition of *Type* attribute
- Each class instance is stored in multiple tables $\Rightarrow$ more difficult access to the data

ORM - Inheritance - Merge to superclass



- all subclass attributes are merged to a single table.
- some attributes may contain NULL value - breaking of 4th NF.
- suitable for cases with smaller number of subclasses with few attributes

ORM - Inheritance - Propagation to subclasses



- Superclass attributes are moved to tables for all non-abstract subclasses
- Suitable if:
  - superclass has few attributes
  - there are many subclasses
  - subclasses have many attributes

Normal forms

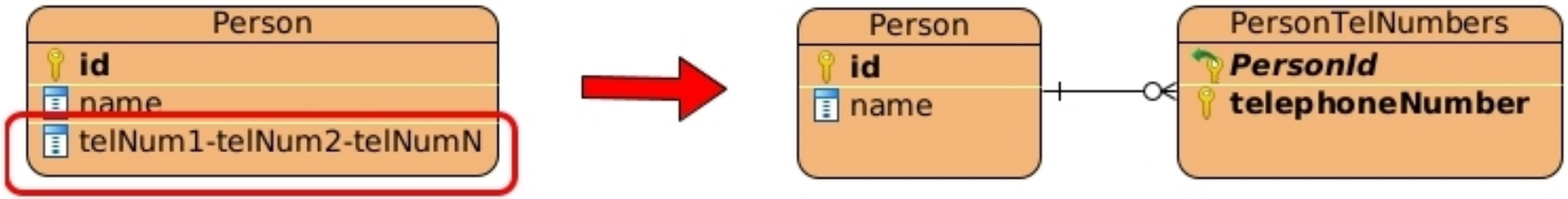
Normal forms are used to achieve good database design.

They help with:

- elimitnation of repetitive data
- reduce table complexity
- prevent anomalies (for update, insert, delete)

1st Normal Form

1st NF requires each attribute to contain only atomic values.



2nd Normal Form

2nd NF requires the schema to satisfy the 1st NF and that each non-key attribute is fully dependable on the whole primary key and other candidate keys.

3rd Normal Form

The schema is in **3rd NF** if it satisfies the 2nd NF and there are no transitive dependencies of non-key attributes on the primary key (and all candidate keys). Therefore, all non-key attributes should be mutually independent.

Additional sources

- http://www.cs.toronto.edu/~sme/CSC340F/2005/slides/tutorial-classes_ERDs.pdf
- http://www.bkent.net/Doc/simple5.htm

Tasks

- Fix the issues in the class diagram.
- Based on the analytical class diagram, create an initial ERD (virtually identical to class diagram + eliminate inheritance).
- Decompose M:N relationships using entities.
- Identify keys for all entities while aiming for maximum efficiency (i.e. do not create artificial id where it is not required).
- Normalize the model into the 3rd NF.
- Create two additional versions of the ERD (at least of its parts) so that the first version satisfies the 2nd NF and violates the 3rd NF and the second version satisfies 1st NF and violates 2nd NF. Add notes explaining the NF violations.
- Submit **pdf report** to the homework vault (**Seminar 07**).
  Deadline:
  - Saturday (Groups 03, 04)
  - Monday (Group 11)
  - Tuesday (Groups 06, 07)
  - Wednesday ([06:00 AM] Groups 08, 09)

Rules for report submission

1. Submit the PDF report, not the VP source file and not an exported image.
2. PDF report must be created using the procedure shown on the seminars including the report settings.
3. The name of the PDF report file should be *lastname1-lastname2-lastname3* of the team members.
4. PDF report must contain all diagrams modelled until now.
5. PDF report must be uploaded to the homework vault by the specified deadline.
6. PDF report must be uploaded to the correct homework vault. The name of the homework vault is always specified on the slides.
7. Each team uploads only a single PDF report for the whole team.
8. Submitted diagrams must be clear and readable.
9. Submitted diagrams should not contain serious mistakes. At least, they should not contain mistakes mentioned in the *Catalogue of common mistakes*.

VP report settings