Object-Oriented Analysis and Modeling

Analysis Data Patterns

Radovan Cervenka

Analysis Data Patterns

→ **Analysis data patterns describe the common problems in modeling data and provide standard generic and repeatable solutions.**

- Purpose:

  - To document good practices in data modeling.

  - To teach (junior) analysts how to build better models.

  - To ease modeling of standard situations and improve readability of models.

- Modeling notation: a simplified version of class diagrams.

  - Classes, associations, generalizations.

- Usually, they model the conceptual/domain level.

- Mostly oriented to the business area.

- There exist also other than analysis data patterns, e.g., *behavioral patterns* modeled as interactions, activities or state machines.

Standardization of Analysis Patterns

- The is no industrial (or other) standard of analysis patterns.
    - In contrast to "GoF" *design patterns*, from Gamma, et. al.

- Several approaches (more-or-less incompatible) exist, e.g.:
    - M. Fowler: Analysis Patterns: Reusable Object Models. Addison-Wesley Professional, October 1996.
    - D. C. Hay: Data Model Patterns: Conventions of Thought. Dorset House Publishing Company, Inc., November 1995.
    - (our main source) L. Sesera, A. Micovsky and J. Cerven: Architektura softverových systemov. Analyticke datove vzory. Slovenska technicka univerzita v Bratislave, 2000.
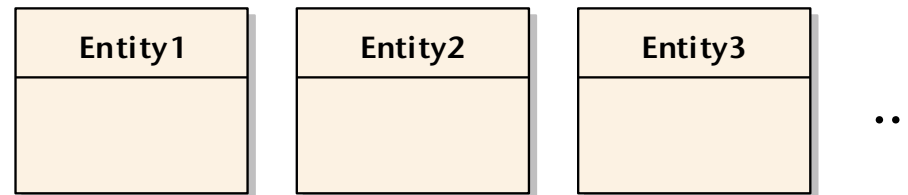   - ...

Types of Analysis Data Patterns

- Optimization "meta" patterns.
    - → Domain-independent analysis patterns used to optimize data models in general.
- Business-related analysis data patterns.
    - → Domain-specific data patterns.
    - Usually explain a step-by-step improvement of a non-optimal model by several applications of "meta" patterns.
    - Covered topics:
        - *People*: humans in a business who perform activities and use objects, classification of people, and their organizational structures.
        - *Objects*: artifacts used in a business, e.g., products, utilized material resources, tools, means, etc.
        - *Activities*: activities of a business process, and the related data, e.g., classification and relationships of activities, work orders, resource utilization, roles in activity execution, execution plans, etc.
        - *Trading*: buying and selling products, procurement of material and tools, orders, contracts, invoices, financial transactions, etc.
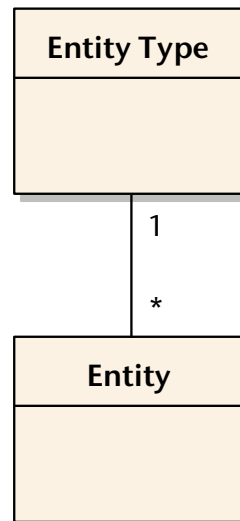
Abstraction of Types

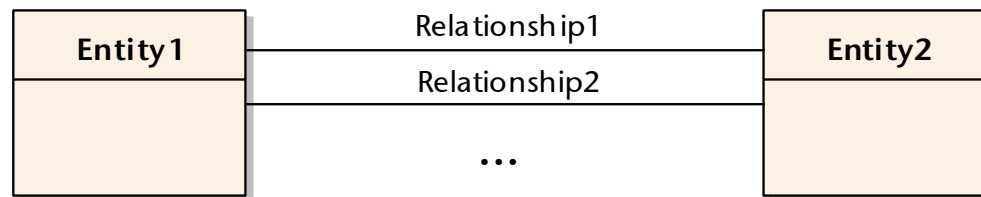- Original model–several entities that specify objects of different types:

- Typing abstraction–different entities (*Entity1*, *Entity2*, ...) are represented by one (*Entity*) and its type is defined dynamically (in runtime):
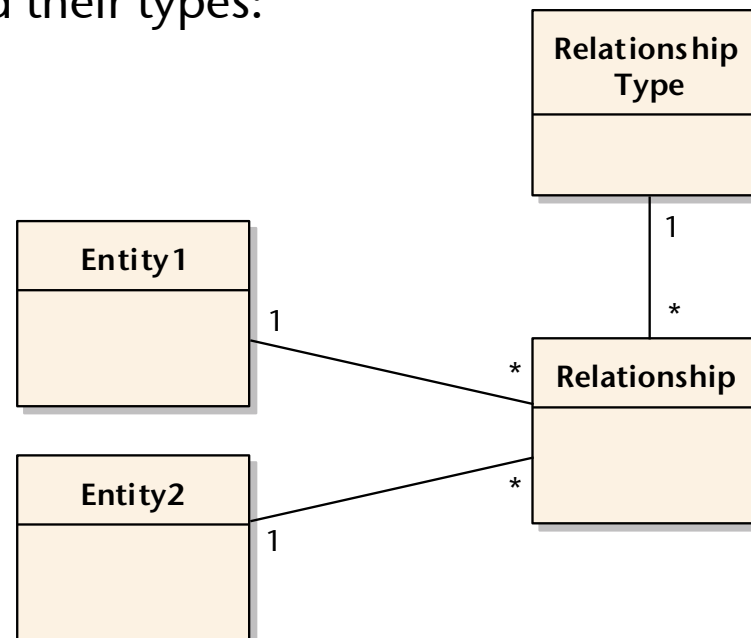
Abstraction of Relationships (Associations)

- Original model—entities with several relationships:

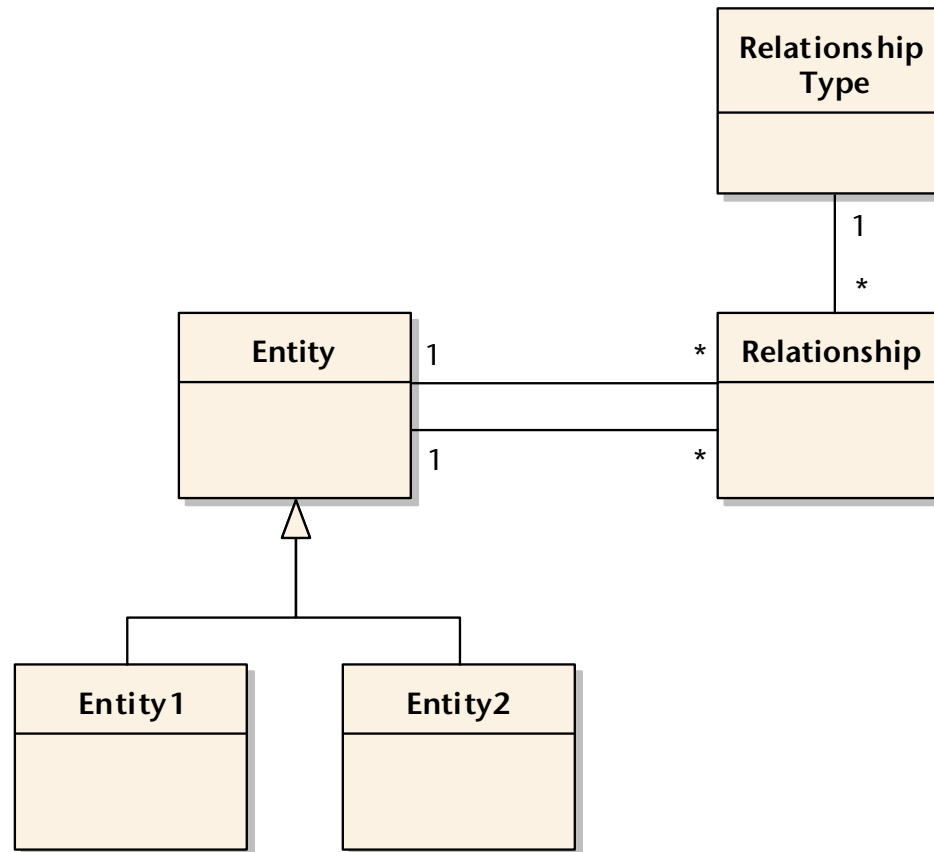- Representing relationships as entities—used to dynamically manage entity relationships and their types:
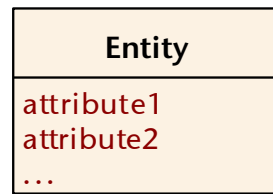
- Simplified form of the relationship abstraction, where the related entities have a common super-type:

Abstraction of Attributes

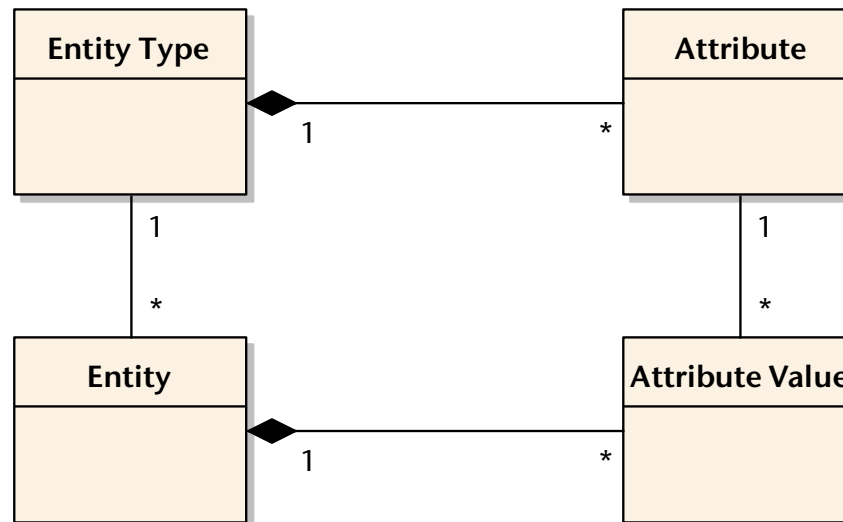- Original model–static set of attributes:

- Representing attributes as entities–dynamic set of attributes:

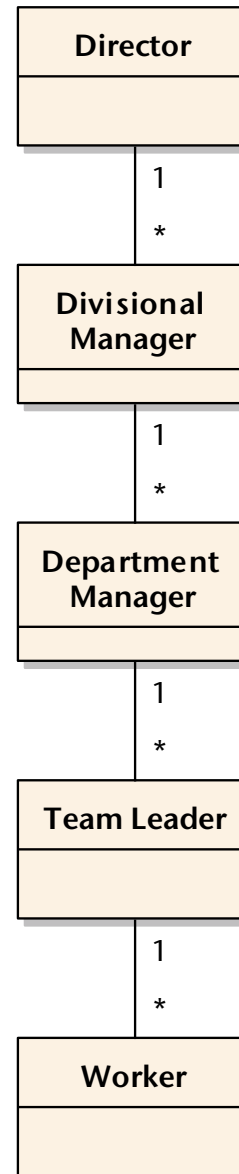- People in family, their attributes, and relationships.
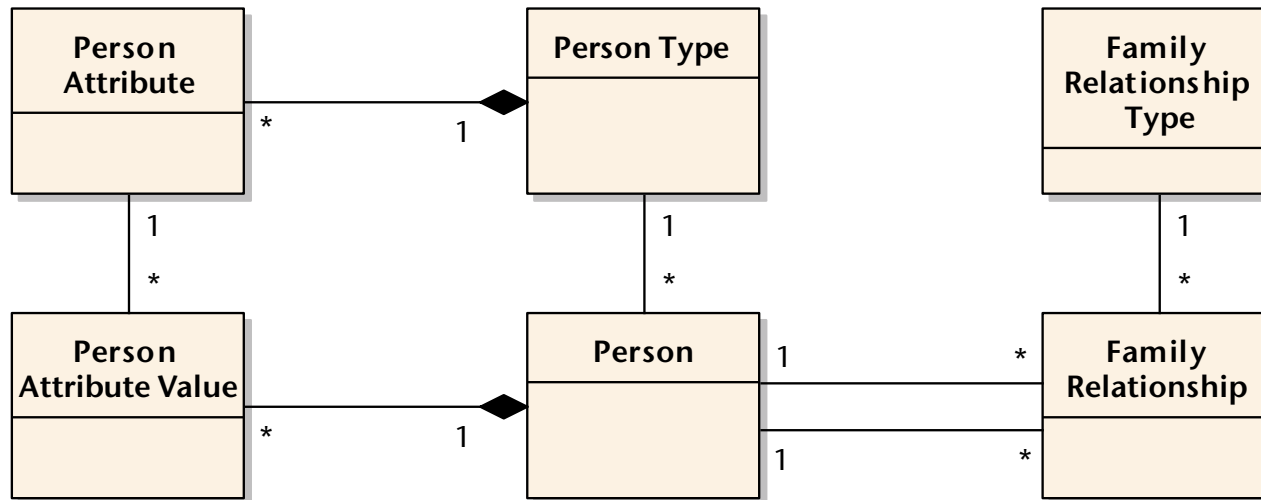
- ■ Direct representation of positions:

```
┌─────────────────┐
│     Director     │
├─────────────────┤
│                 │
└─────────────────┘
         │ 1
         │ *
┌─────────────────┐
│    Divisional   │
│     Manager     │
├─────────────────┤
│                 │
└─────────────────┘
         │ 1
         │ *
┌─────────────────┐
│    Department   │
│     Manager     │
├─────────────────┤
│                 │
└─────────────────┘
         │ 1
         │ *
┌─────────────────┐
│   Team Leader   │
├─────────────────┤
│                 │
└─────────────────┘
         │ 1
         │ *
┌─────────────────┐
│      Worker     │
├─────────────────┤
│                 │
└─────────────────┘
```

10

- ■ Separation of people from their positions:
    - • Dynamic specification of positions and their assignments to persons.
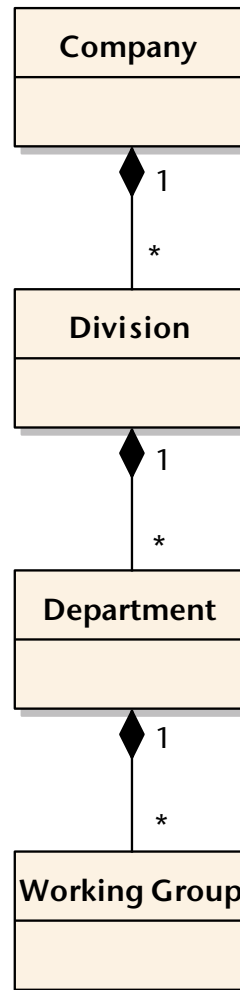   - • Several superiority relationships were reduced to one relationship.
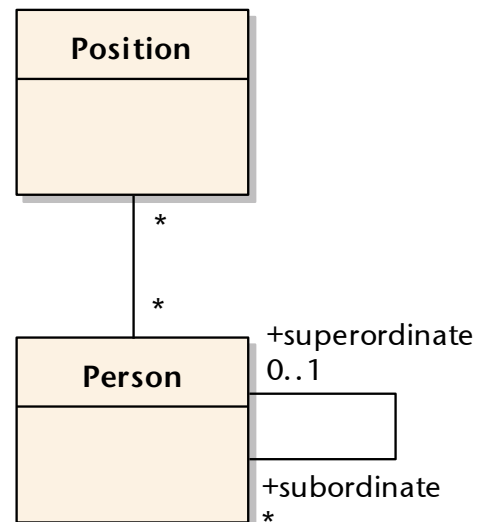
- Abstraction of various work relationships:
  - The possibility to express several different relationship types (including the original superiority relationship) dynamically.
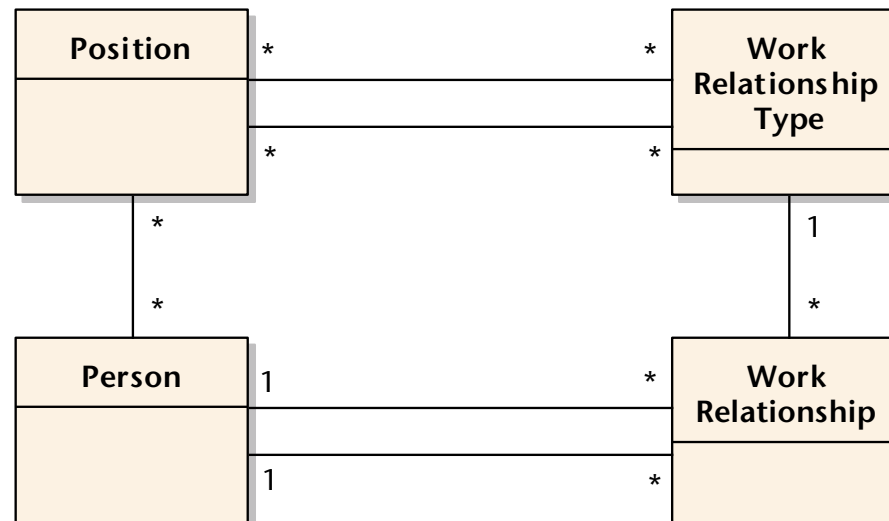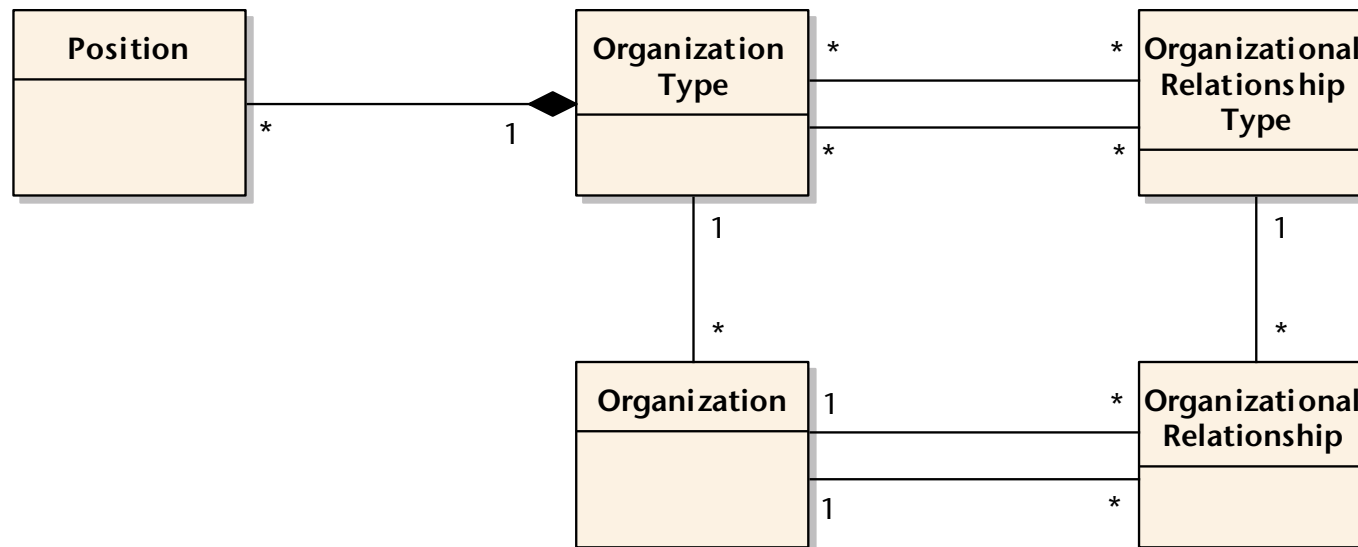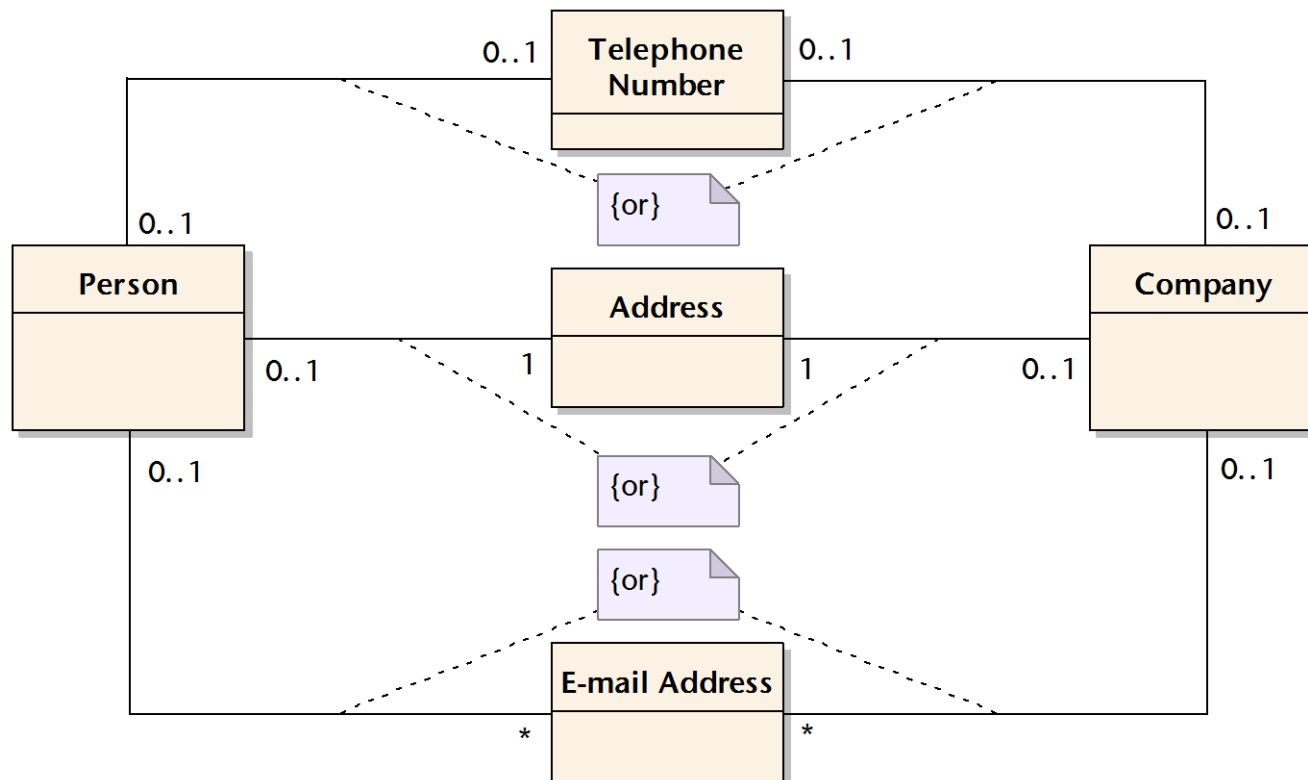
■ Direct representation of an organization structure:

- Flexible representation of an organization structure and its connection to positions:

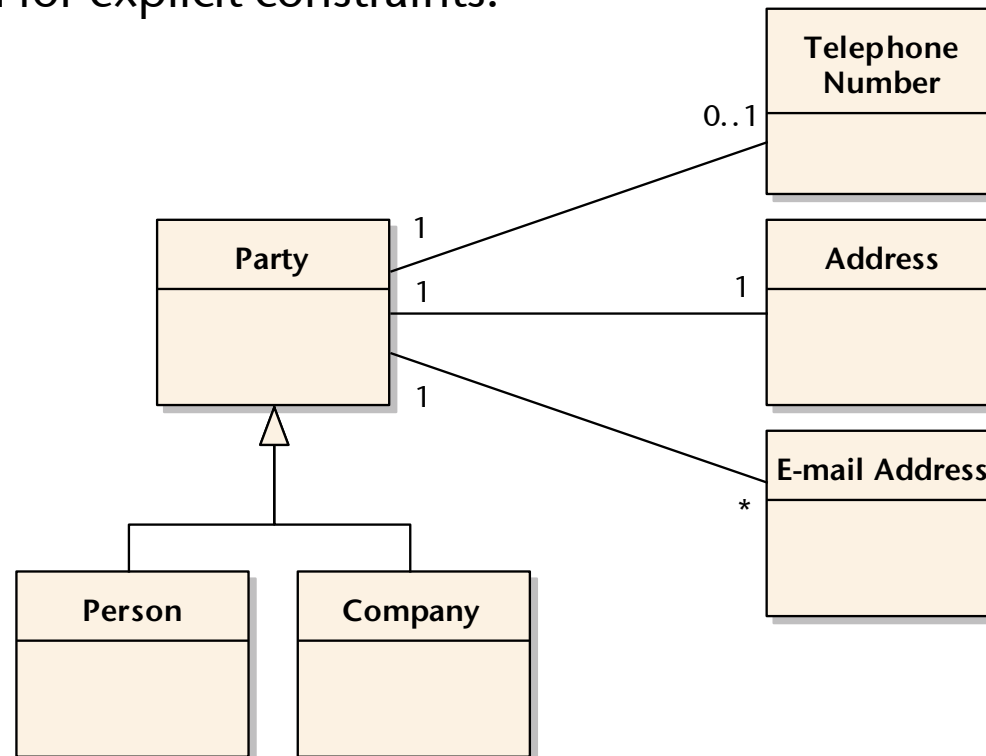 - Application of the type abstraction and the relationship abstraction "meta" patterns.
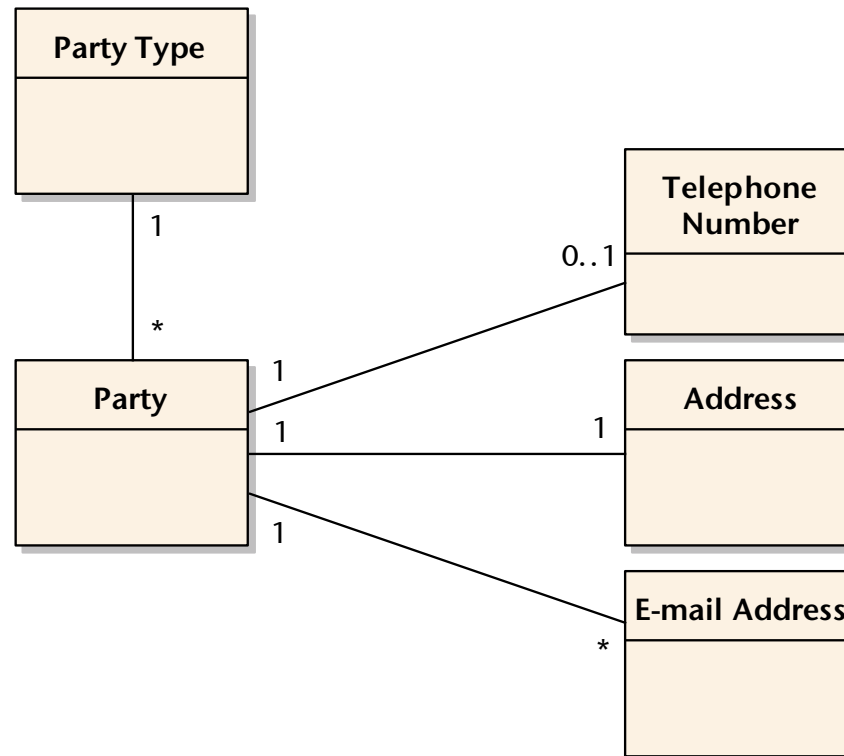
■ Direct representation of an address book:

■ Optimized version of an address book:

- The *Party* entity is a general representation of the *Person* and *Company* entities.

- Reduced amount of associations.

- No need for explicit constraints.

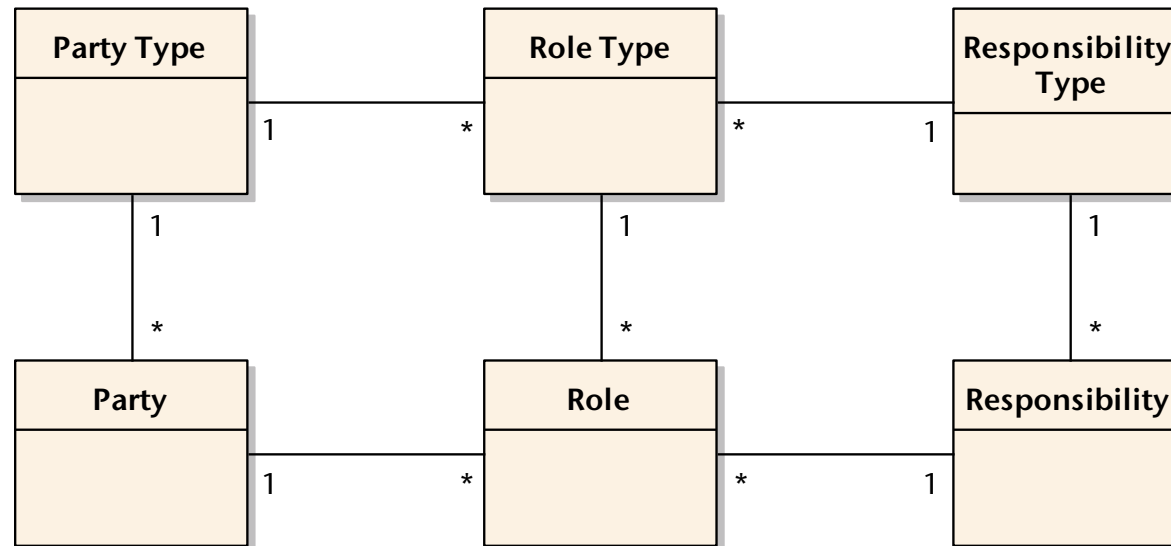- Flexible representation of parties and their types:
  - Application of the type abstraction "meta" pattern.

- Abstraction of the work relationship and organizational relationship by the generic *Responsibility* relationship:

  - *Work Relationship* (*Type*) and *Organization Relationship* (*Type*) are represented by *Responsibility* (*Type*).

  - Reduced amount of entities and relationships.

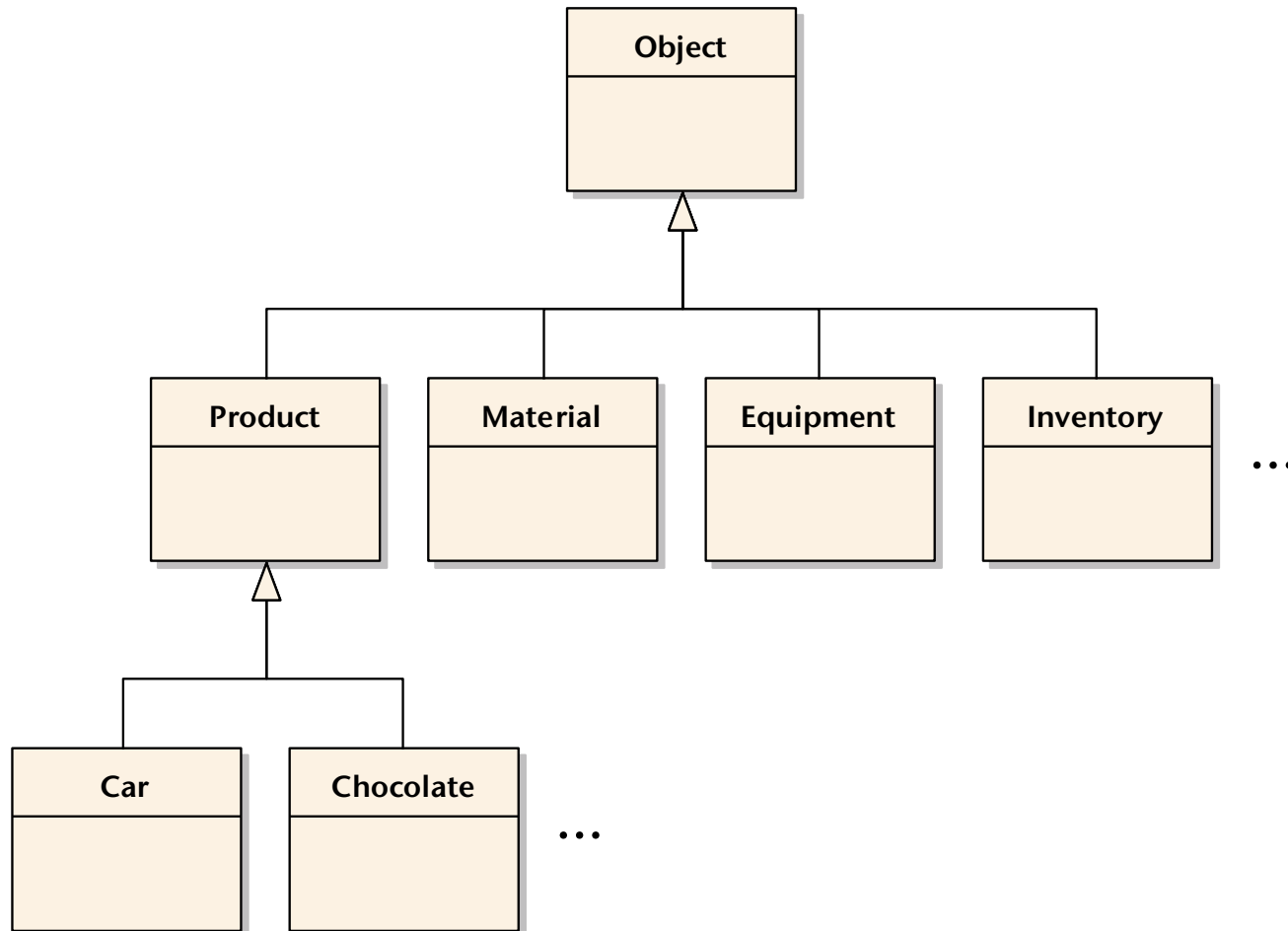 - Constraint: a responsibility can comprise just two parties.
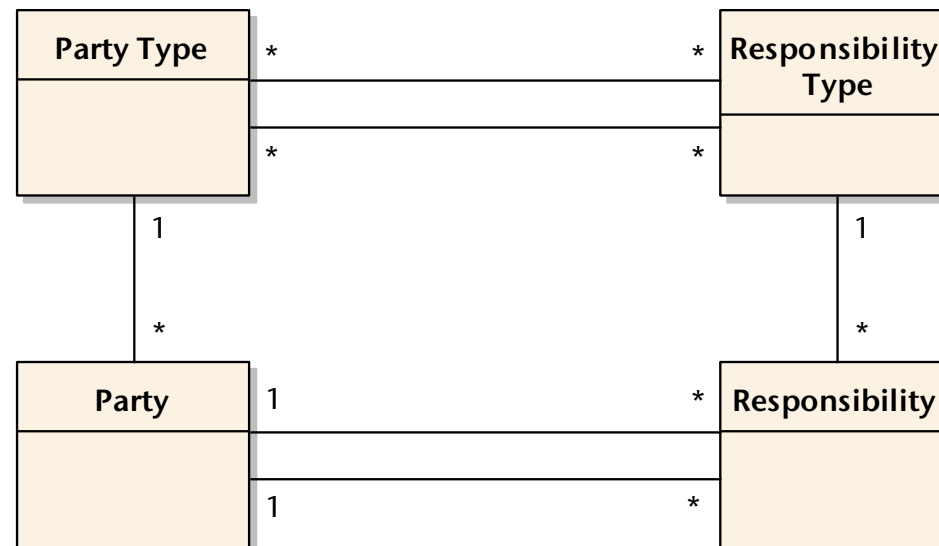
- Extending the responsibility by several parties playing particular *roles*:
  - Each responsibility can have several roles, each played by a (possibly different) party.
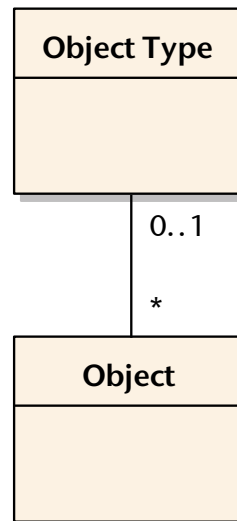 - The came at the type level.

■ Direct representation of business objects:

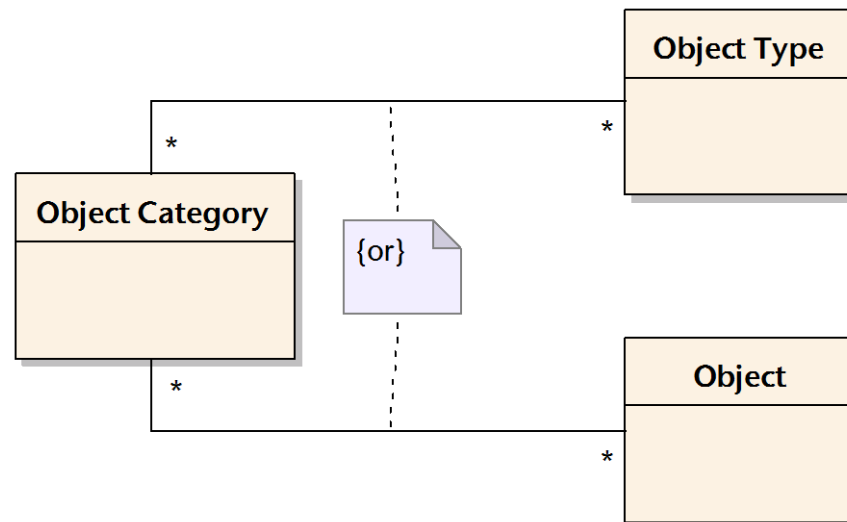■ Flexible representation of objects and their types:

- Application of the type abstraction "meta" pattern.
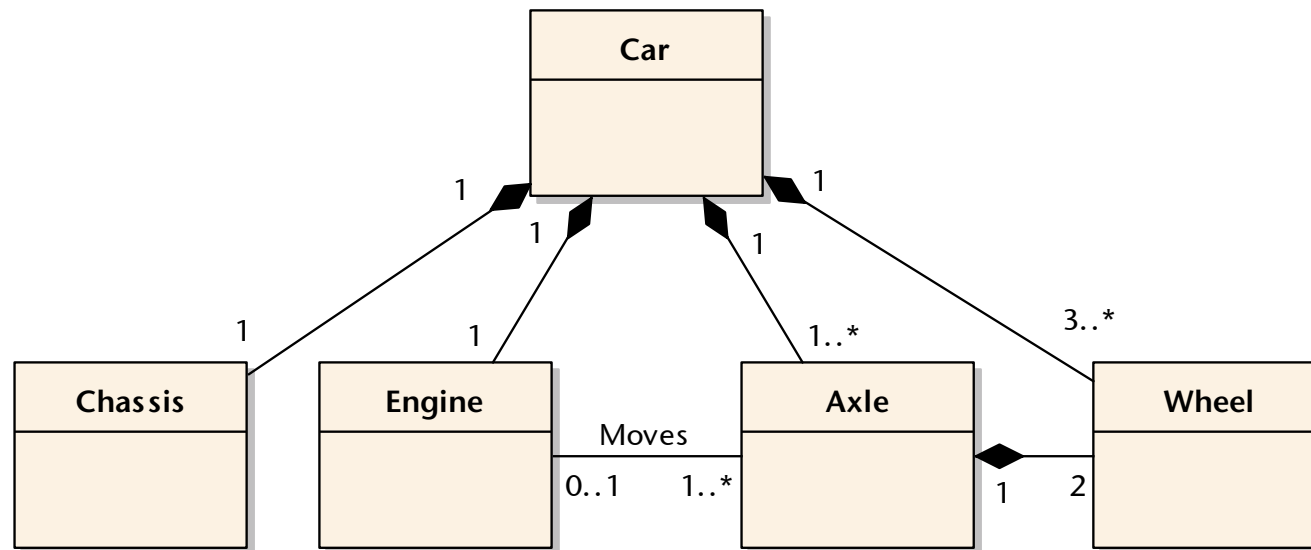
■ Categorization of objects and their types:

- For instance, the object «Nissan Patrol» is of category «off-road».

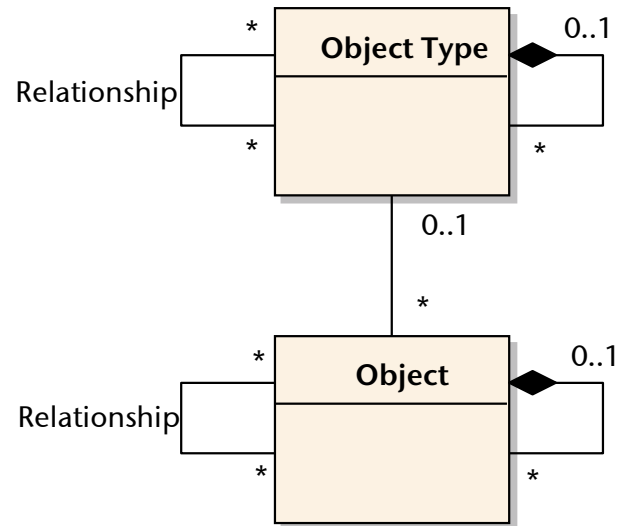- Example of the direct representation of object structures (compositions and other relationships):

- Abstraction of object/object type composition and their other relationships:
  - Relationships expressed directly by associations of entities.
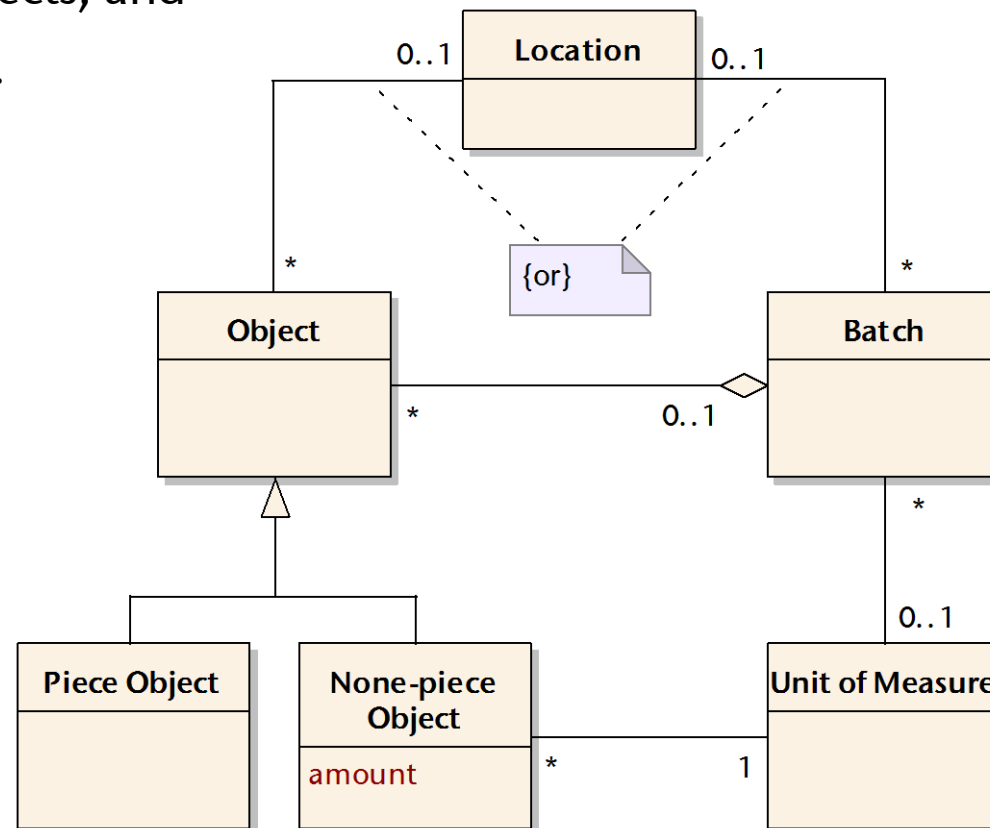
- Abstraction of object composition and other object relationships into a dynamic relationship:

    - Application of the relationship abstraction "meta" pattern.
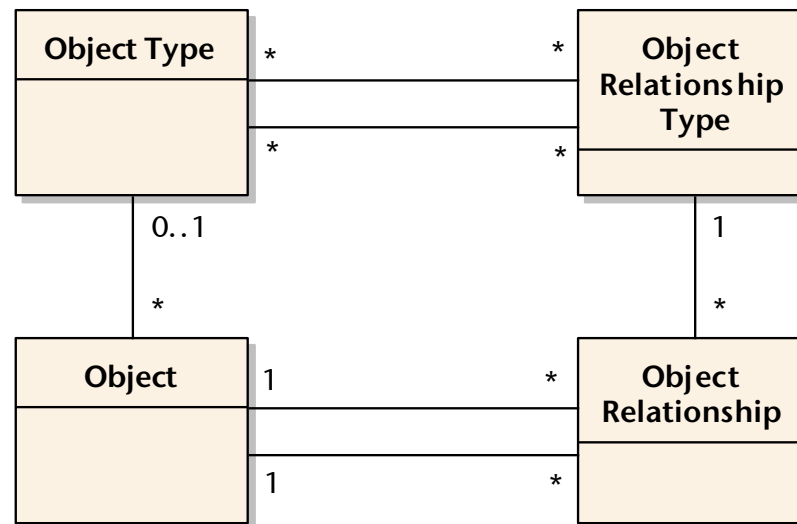
- Other aspects of business objects:
  - placing at a location,
  - batches (groups) of objects,
  - piece and non-piece objects, and
  - measurement of objects.

OOAM.AnalysisPatterns | R. Cervenka

Application of Object Patterns in a Road Net

Flexible Object Type

Flexible Object Relationship

Direct representation of objects

Road Type	Road	Road Segment	Crossway	Crossway Type
class	number internationalNumber	length		

Road Object

Road Object Stationing

Segment Traffic Direction

Bridge

Tunnel

Barrier

Road Equipment

Traffic Lights

Traffic Sign

Catching Safety Installation

start
0..1
0..1
end

from   to   through