

Com S 362

Object-Oriented Analysis & Design

Spring 2018 – Week 4, Lecture 11

Friday, February 2, 2018

Domain Model

Identifying the concepts

Domain Model

In Object-Oriented projects, the Domain Model is the analysis artifact that most directly translates to the design domain.

The Domain Model

- Captures the most important types of objects in the context
- Representation of “conceptual classes”
- Provides a “visual dictionary” of the design domain

Note:

The domain model is *not* designed from expectations about software classes or objects.

Domain Model

Key Benefits

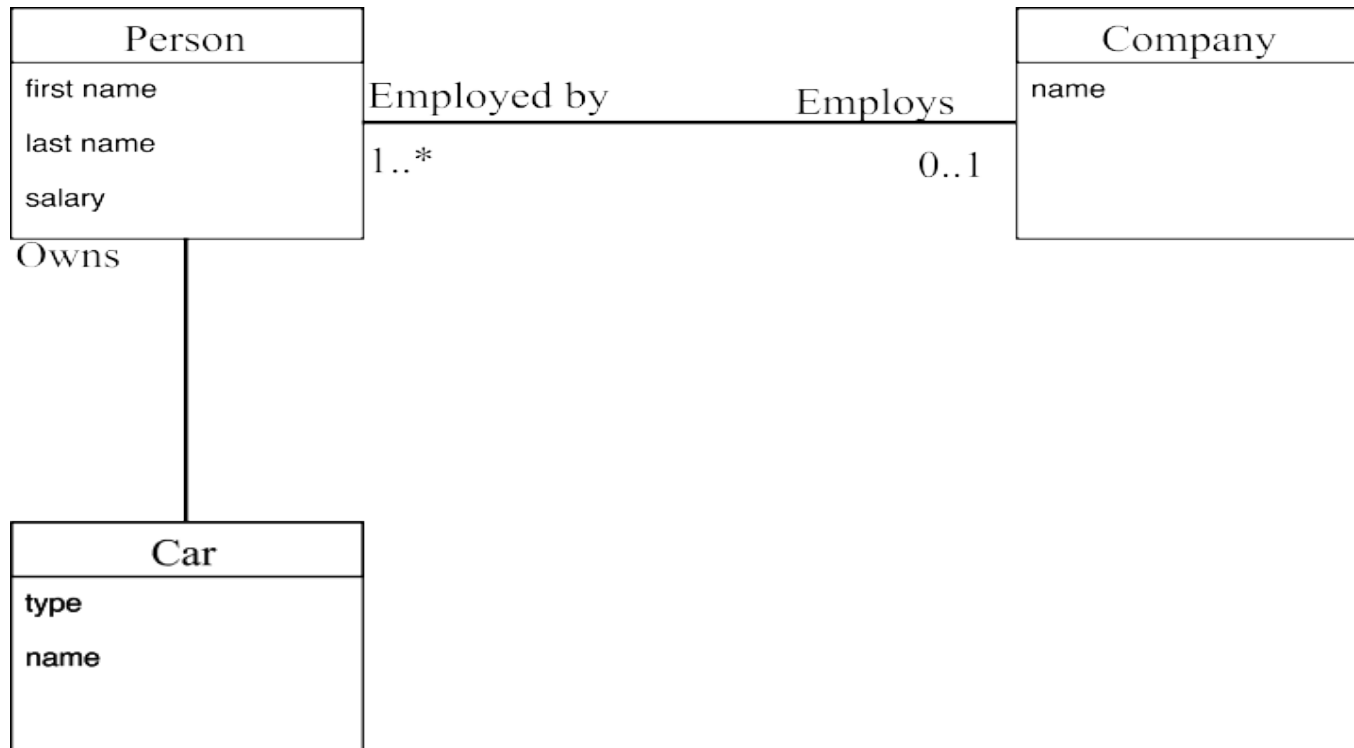
A graphical Domain Model:

- Supplies a visual representation of a conceptual framework of the relevant things in the problem space
- Provides a list of well-defined terms
 - Using matching names in programs and the domain model helps with understanding what an object represents
- Drives consensus in meaning and relationships; helps refine understanding of the domain, thus closing the gap between mental and software models.
- By focusing on concepts, Domain Models encourage designs that organize code as separate concerns.
 - Concepts change less radically and less frequently than work flows, for example.

Based on course notes by Dr. Hridish Rajan.

Domain Model

An Example Domain Model



Based on course notes by Dr. Hridish Rajan.

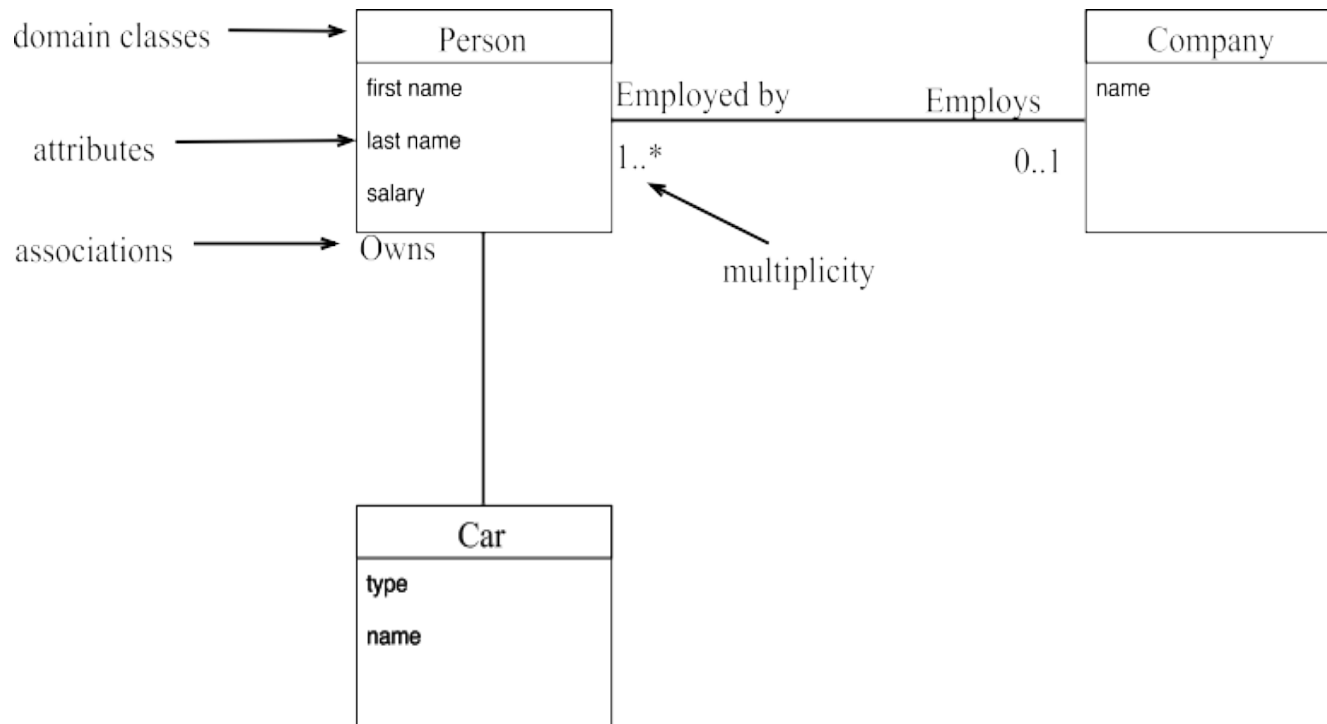
Scope of Domain Models

A Domain Model can include:

- **Domain classes** - representation of types of conceptual objects
- **Attributes** - description of a named slot of domain classes which holds a separate value for different class instances
- **Associations** - description of relationship between domain classes
- **Multiplicity** - how many instances of class A can be associated with an instance of class B.

Domain Model

Annotated Example



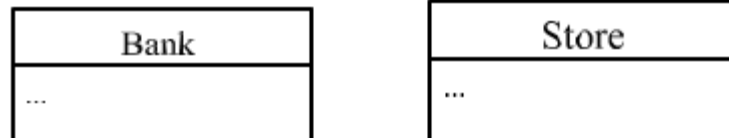
Based on course notes by Dr. Hridish Rajan.

Domain Model

Conceptual Classes

- Domain models includes a collection of conceptual domain classes
- Ideas, things, or objects

Conceptual classes in UML:



Domain Model

Attributes

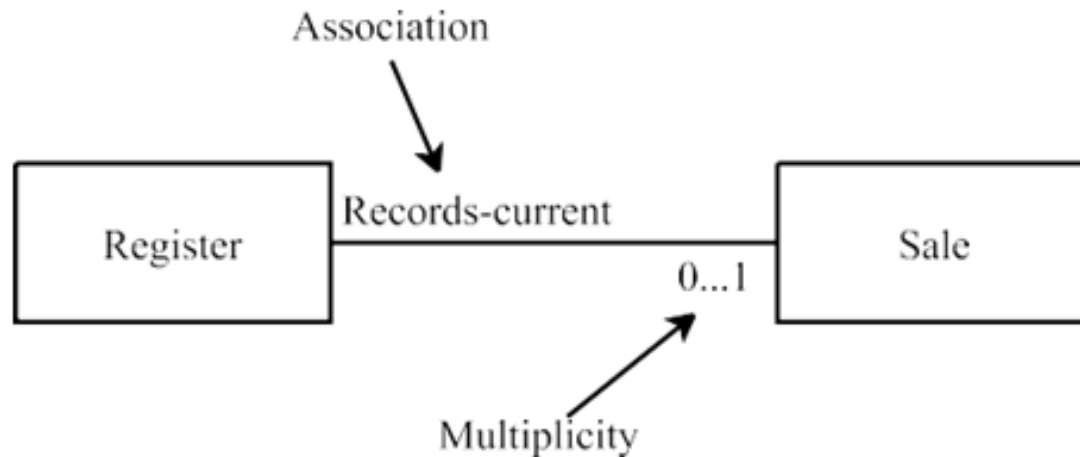
- Capture Logical data value of an object
 - Represent an object's subject matter expertise
- Usually used when the requirements suggests or implies the need to remember information
 - *A cashier needs an ID*
 - *A store needs a name and address*



Domain Model

Associations

- Records and shows the relationship between classes
- What objects is need to be remembered of?
 - A Sale instance will need to know what SaleLineItem instances are associated with it.



Domain Model

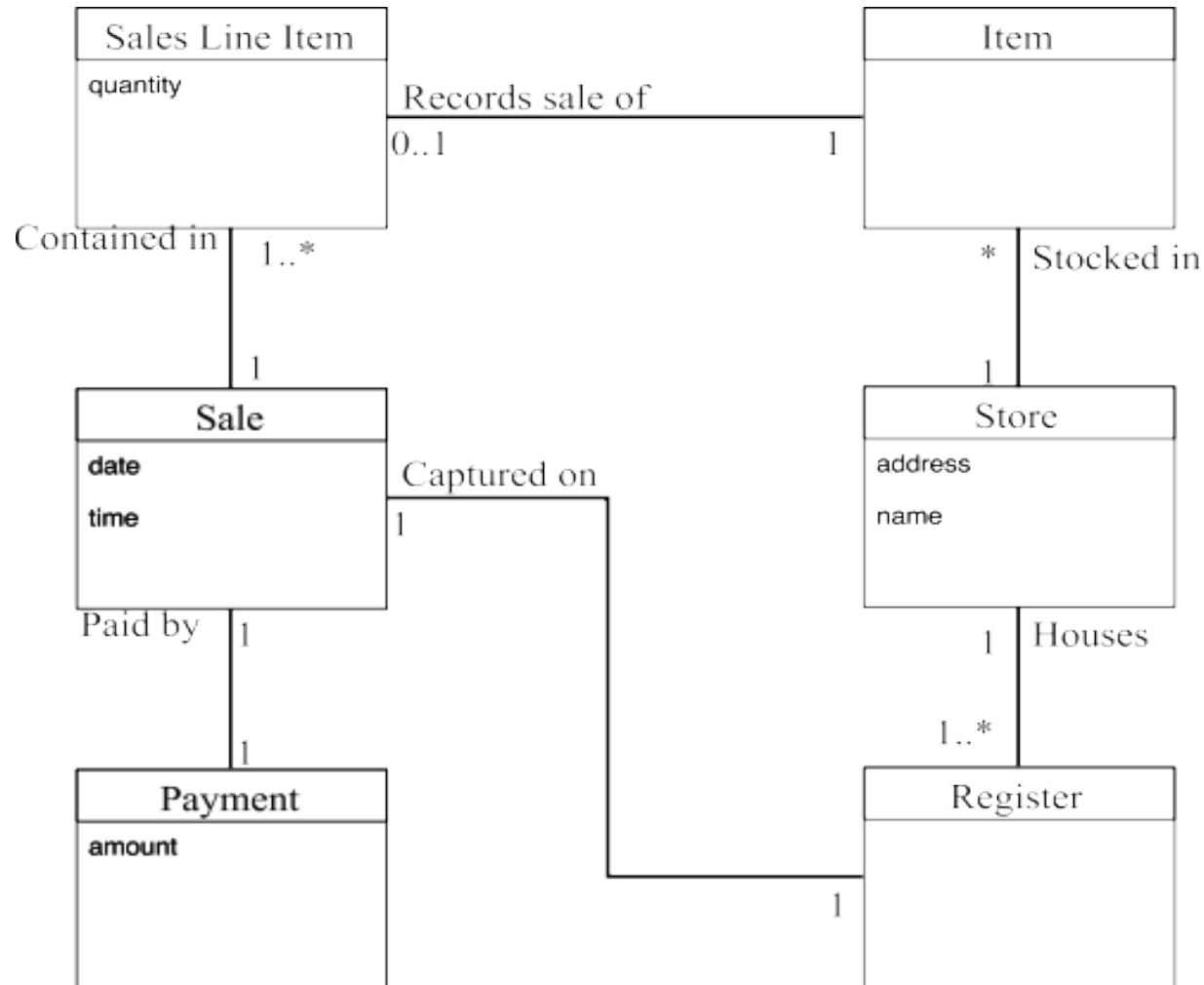
Multiplicity

Multiplicity - how many instances of class A can be associated with an instance of class B

- An instance of Store can be associated with many (zero or more) Items
- Detailed with *, 0..n, 1 *annotations* on appropriate associations.
- Represents what object counts are valid at a particular moment.
- For example a car can be owned by multiple persons over time, but can only be owned by one person at a time (excluding joint ownership).

Domain Models

Another Example



Example adapted from
Craig Larman. Applying UML
and Patterns: An Introduction to
Object-Oriented Analysis and
Design and the Unified
Process.

Domain Model

Some points to keep in mind

- Domain models are **static** representations, NOT dynamic information, such as entities sending messages to another
- The objects represent real world concepts, NOT software entities, such as `AbsolutePathString`
 - We are not writing programs
 - Domain models aids designers and developers with visualizing the domain of the problem, and helps with thinking by using the terms in the model

Looking Forward

Finding Domain Concepts

How do we identify domain model objects (conceptual classes, attributes and associations)?

- Reuse previous models, experience
- Category list
- Identify noun phrases (from use cases we have constructed so far)
 - Physical objects
 - Organizational units
 - Roles
 - Categories (of physical objects, for example)

Domain Model Summary

Key Points

Domain models: used to understand classes of the domain and the interaction between classes

- Elements of a domain models are often derived from use cases constructed earlier in the object-oriented analysis process
- Constructing a domain model helps with the object-oriented design

Construction of Domain models

- Identify domain classes
- Identify attributes and association relationships