COMP 3700: Software Modeling and Design

(Domain and Class Modeling)

Topics

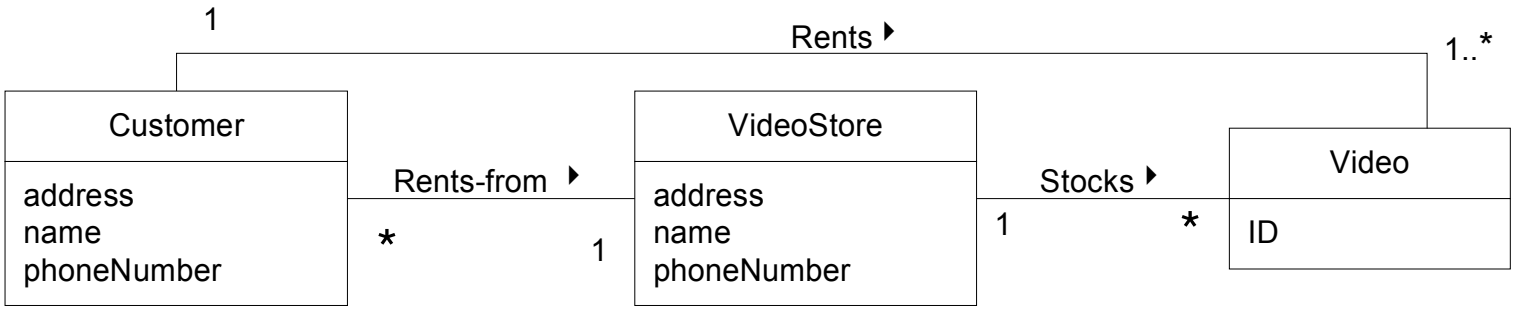
- Object and Class Concepts
- Links and Association Concepts
- Generalization and Specialization
- A Simple Class Model
- Navigation of Class Models

Domain Modeling

- Partitions and illustrates the important domain concepts.
- A classic object-oriented analysis activity.
- What are the objects of interest in the this domain?
  - their attributes?
  - their relationships?
- IMPORTANT: Not software objects, but a “visual dictionary” of domain concepts.

A Domain Model Does Not Represent Software Objects

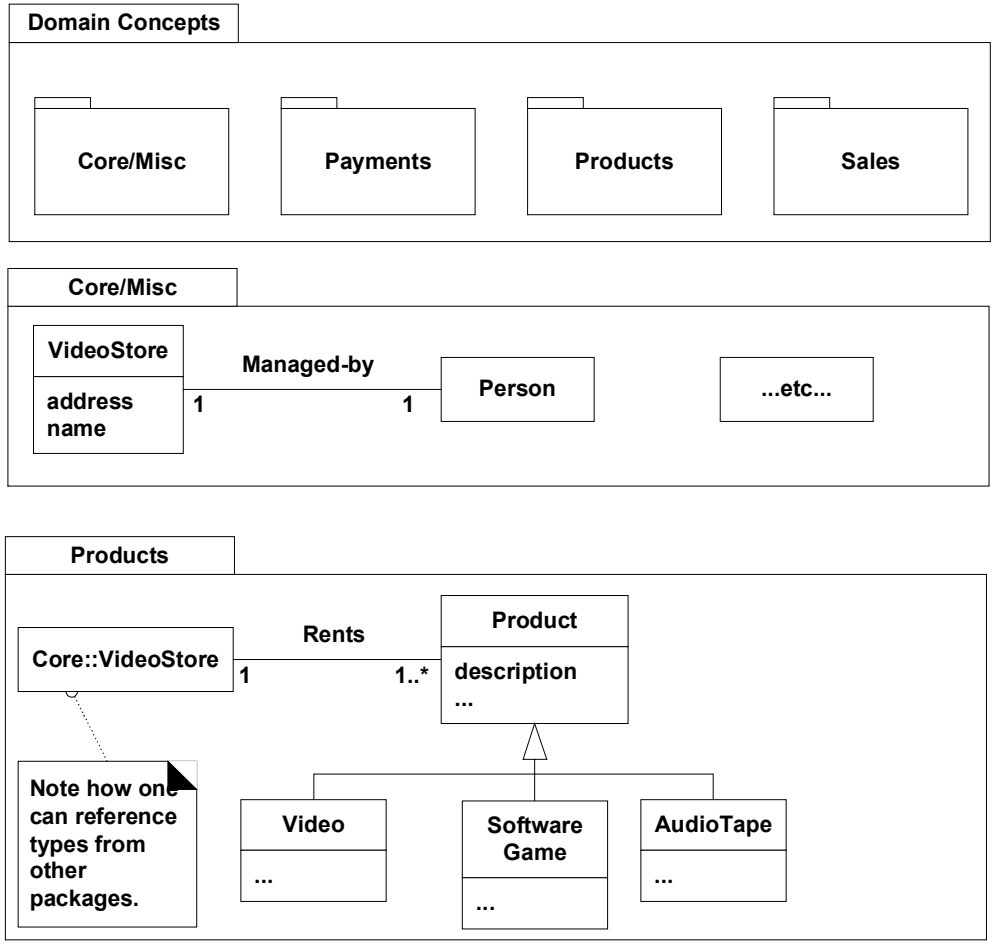
- A model of domain concepts, not of software objects.
  - A "visual dictionary" of important words in the domain.
- Uses UML *static structure diagram* notation.



How to Make a Domain Model

- List the candidate conceptual classes using the Conceptual Class category list or linguistic analysis
- Draw them in a domain model
- Add the associations necessary to record relations
- Add the necessary attributes to fulfill the information requirements.

Partitioning the Domain Model



Objects and Classes

- Object is a concept, abstraction, or thing that has meaning for an application.
- Objects appear as proper nouns in problem descriptions (e.g., Joe smith, IBM, process 7648)
- An object is an instance - or occurrence – of a class.
- A class describes a group of objects with the same properties (attributes), behavior (operations), and relations. (e.g., person, company, process).

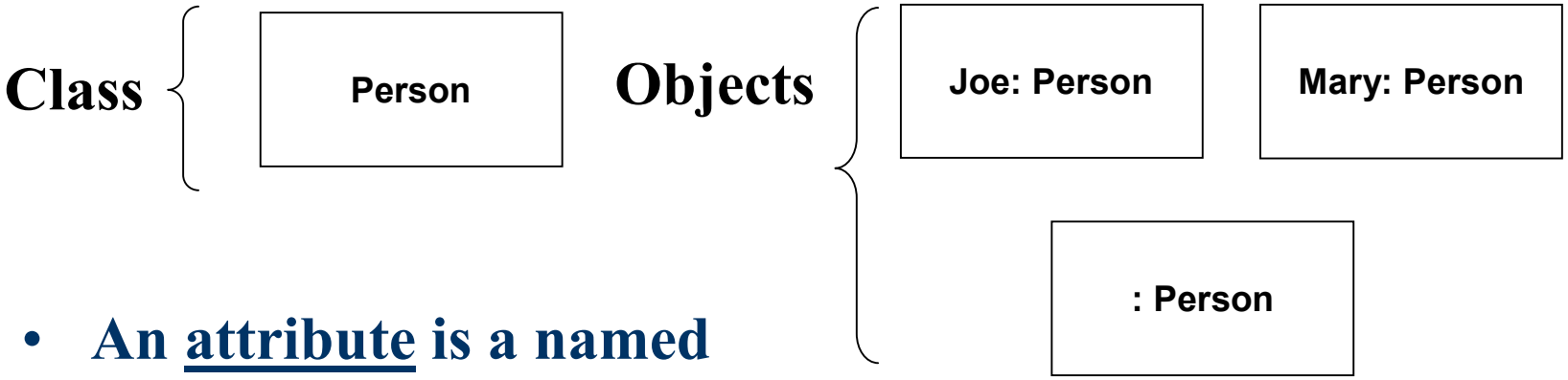
Finding Domain Concepts

- **Linguistic Analysis: Identify the nouns and noun phrases in textual descriptions.**
  - **Care must be applied with this method: a mechanical noun-to-class mapping isn't possible, and words in natural languages are ambiguous.**
- **Specification: Design a library catalog system. The system must support the registration of patrons, checking books in and out patrons, adding and removing of books, and determining which patron has a book.**

Approaches

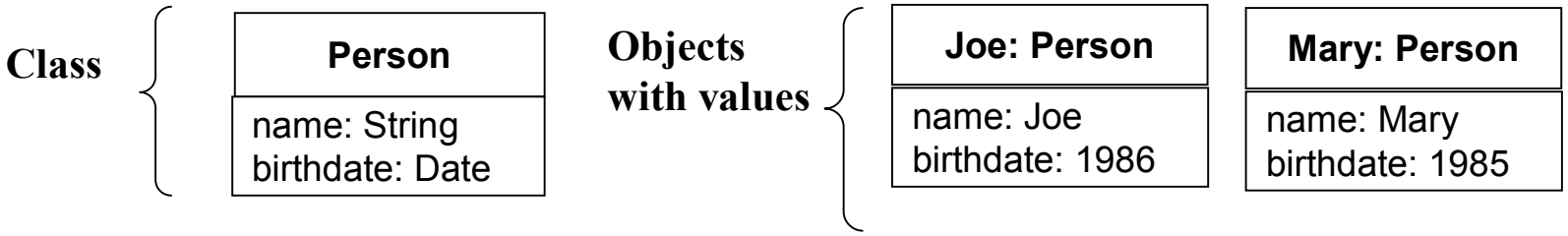
- **Abbott and Booch suggest:**
  - **Use nouns, pronouns, noun phrases to identify objects and classes**
  - **Singular → object, plural → class**
  - **Not all nouns are really going to relate to objects**
- **Coad and Yourdon suggest:**
  - **Identify individual or group "things" in the system or problem**
- **Ross suggest:**
  - **People, places, things, organizations, concepts, events**
- **Danger signs:** class name is a verb, is described as performing something

Attributes



- An attribute is a named property of a class that describes a value held by each object of the class.
- Object is to class as value is to attribute

Attributes

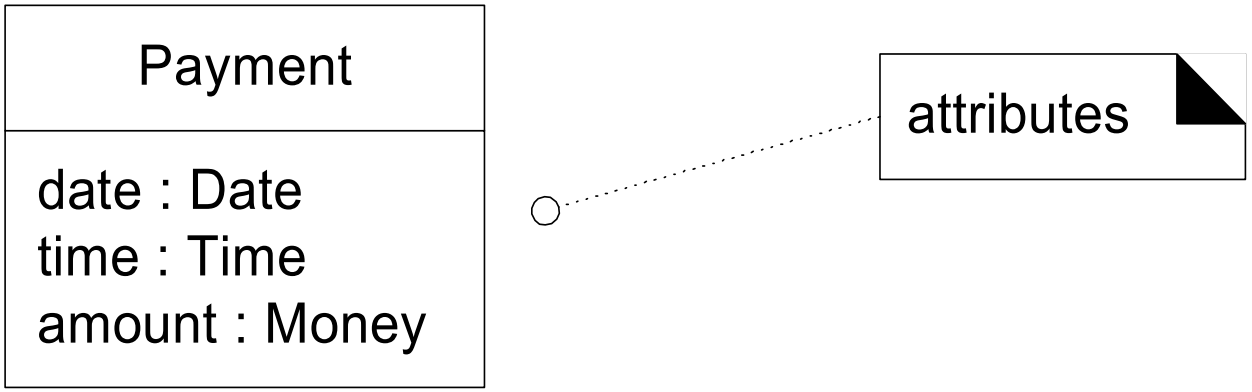


- An attribute is a named property of a class that describes a value held by each object of the class.
- Object is to class as value is to attribute

- One can find attributes by looking for adjectives or by abstracting typical values

Attributes

- **Show only "simple" relatively primitive types as attributes.**



- **Connections to other concepts are to be represented as associations, not attributes.**

Operations (During the design phase – not domain modeling)

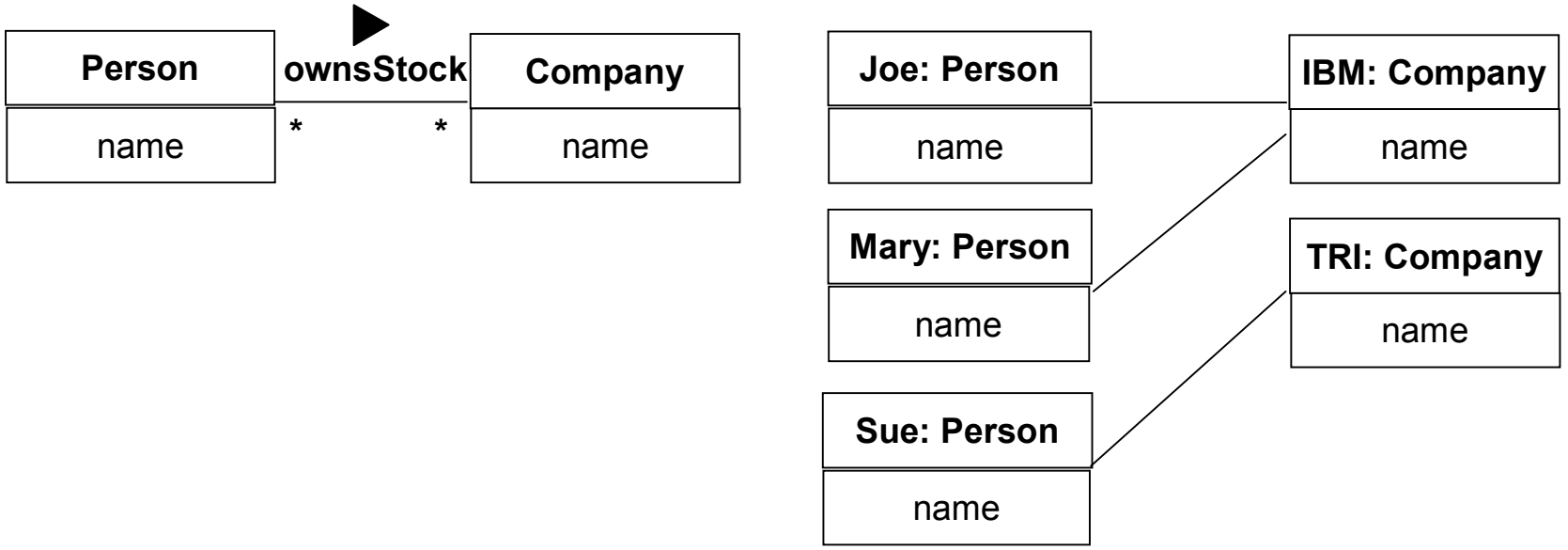
Class

Person	File	Geometric Object
name: String birthdate: Date	filename size	color position
changeJob changeAddress	print	move(delta: Vector) rotate(angle: float)

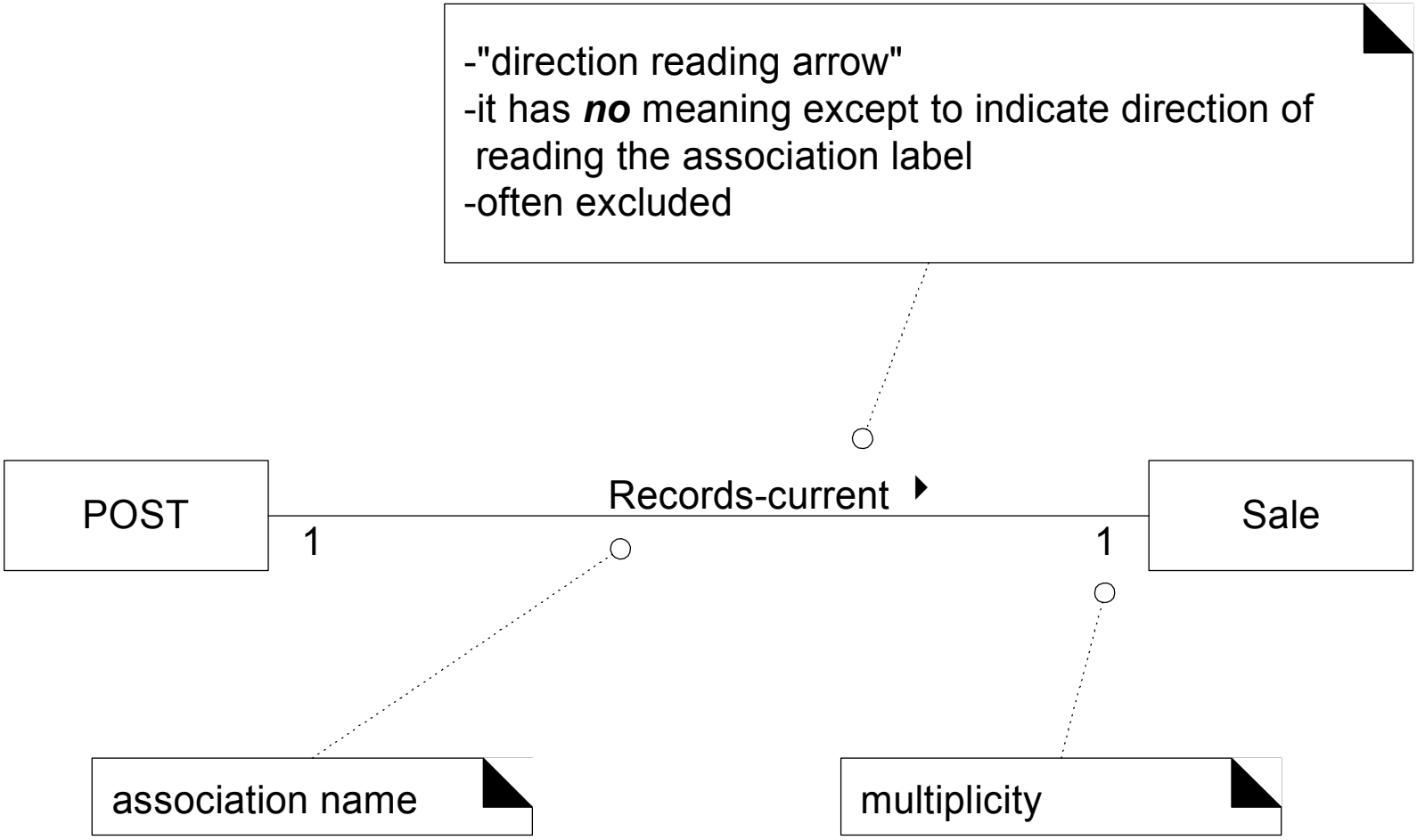
- **An operation is a function or procedure that may be applied to or by objects in a class (e.g., hire, fire, payDivident for company)**

Links and Association Concepts

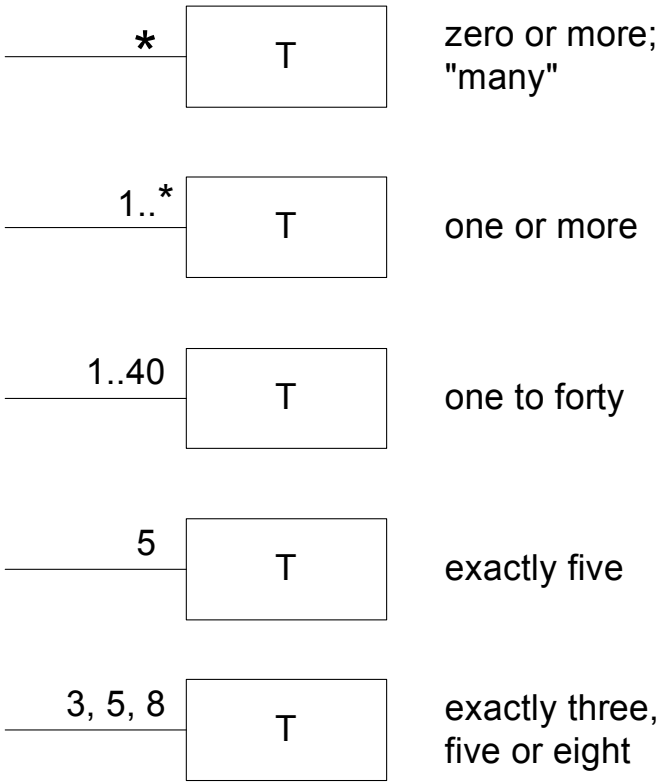
- A link is a physical or conceptual connection among objects. (e.g., Joe works-For IBM).
- An association is a description of a group of links with common structure and semantics.

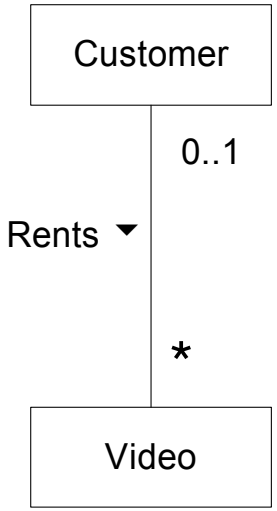


Associations



Multiplicity

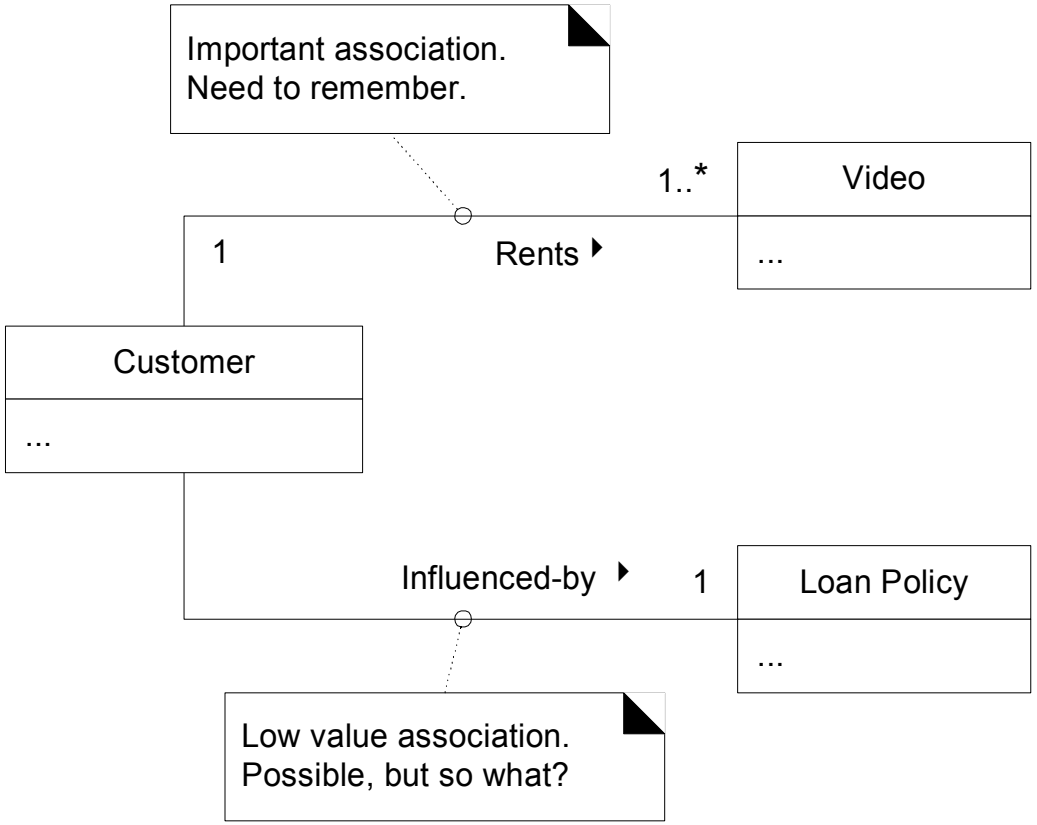




One instance of a Customer may be renting zero or more Videos.

One instance of a Video may be being rented by zero or one Customers.

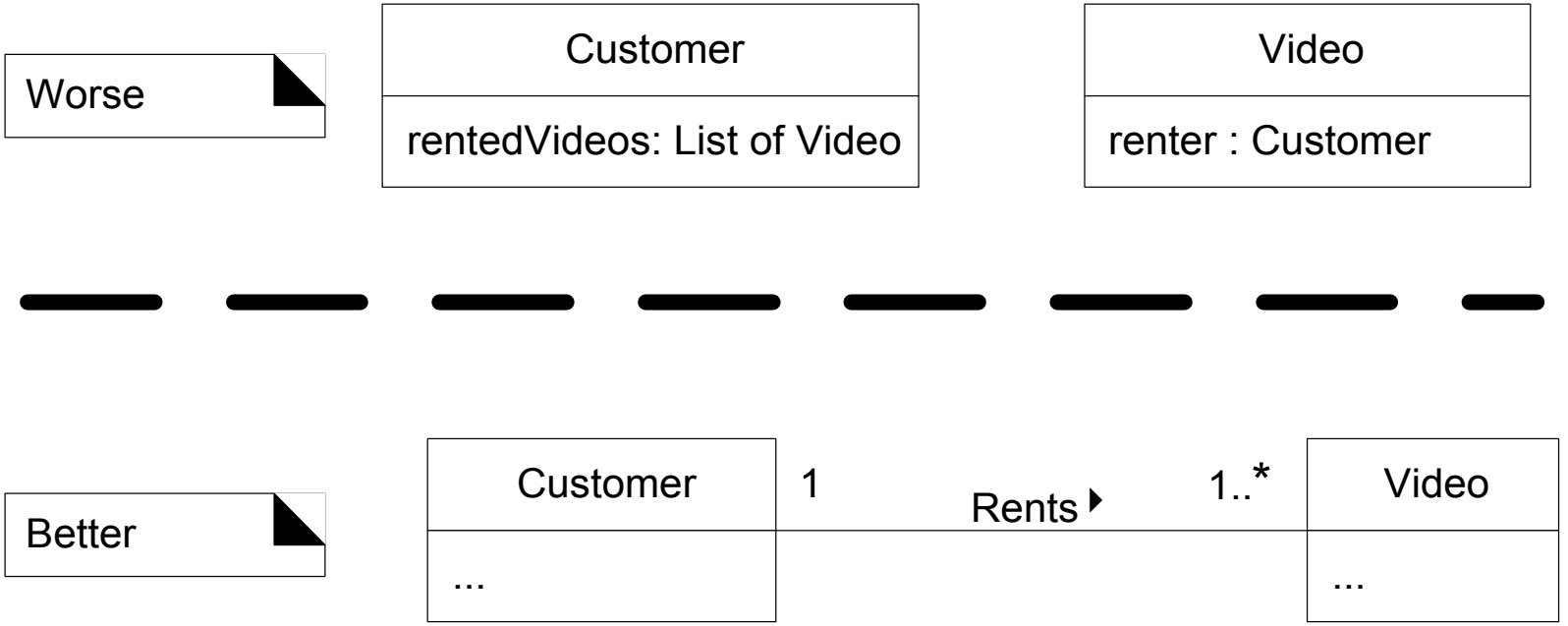
Focus on Important Associations



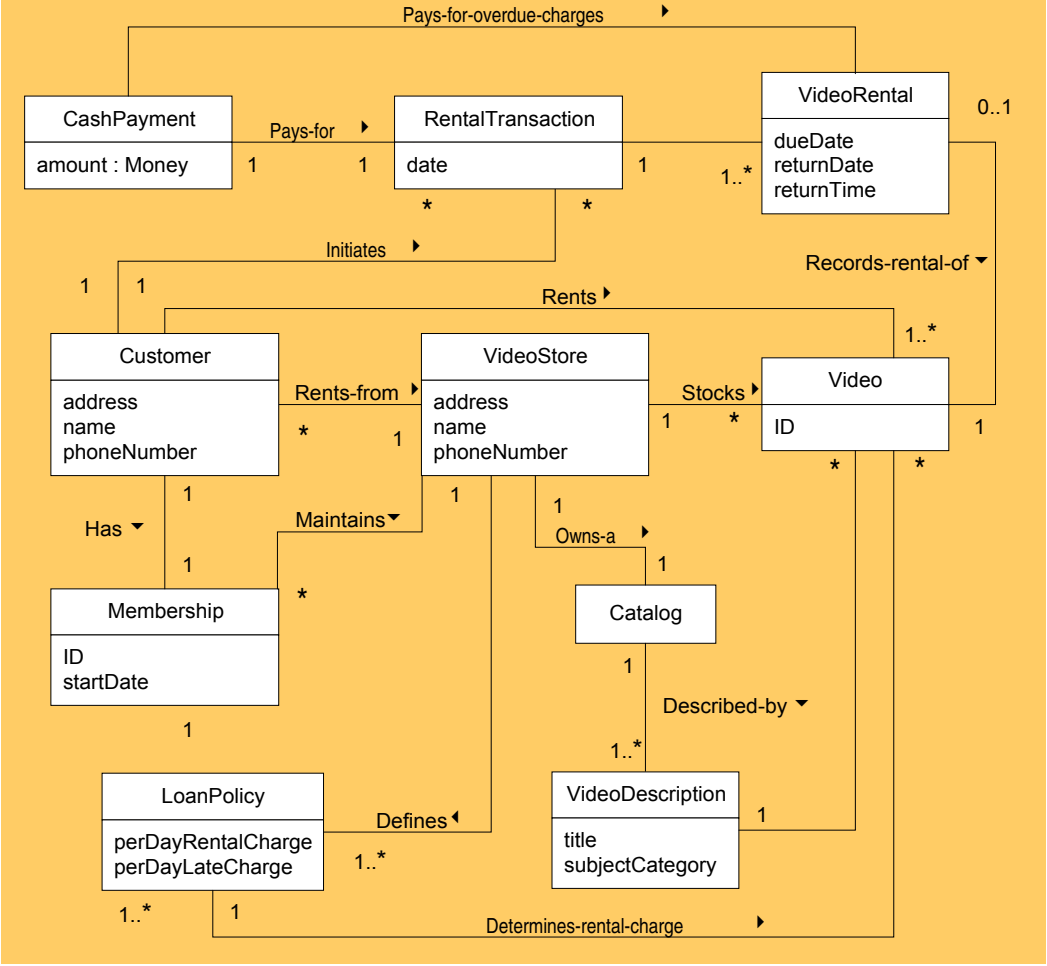
- Name an association based on TypeName – VerbName – TypeName format

Do Not Use Attributes To Relate Concepts

- **Why not?**

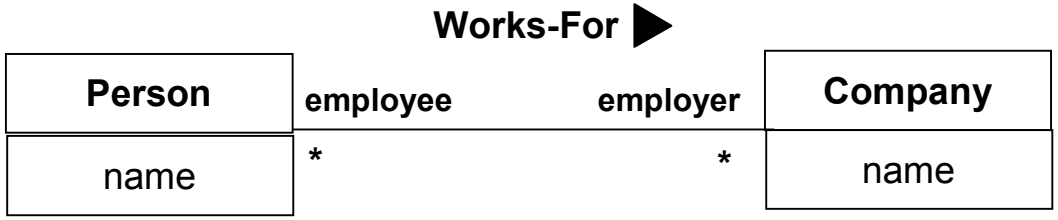


A Sample Class Model

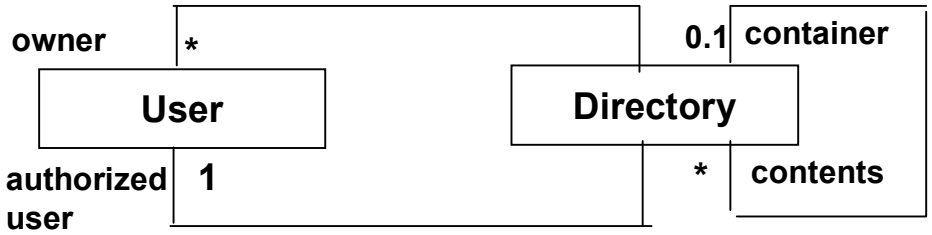


Association End Names

- You can assign association ends not only multiplicity, but also a name.

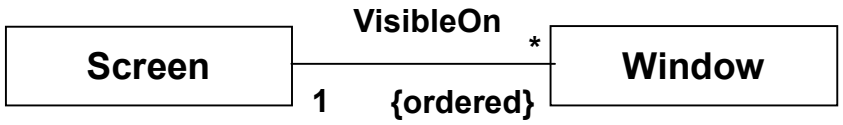


- Association end names are necessary for associations between two objects of the same class



Ordering, Bags, and Sequences

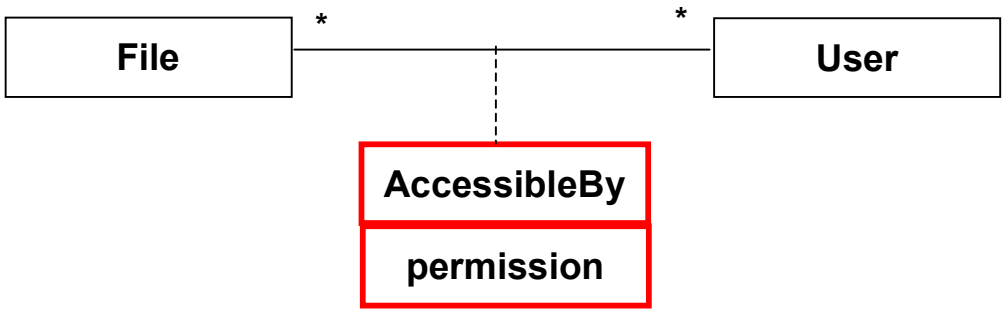
- Sometimes objects on a "many" association end have explicit order.



- A sequence is an ordered collection of elements with duplicates allowed.

Association Classes

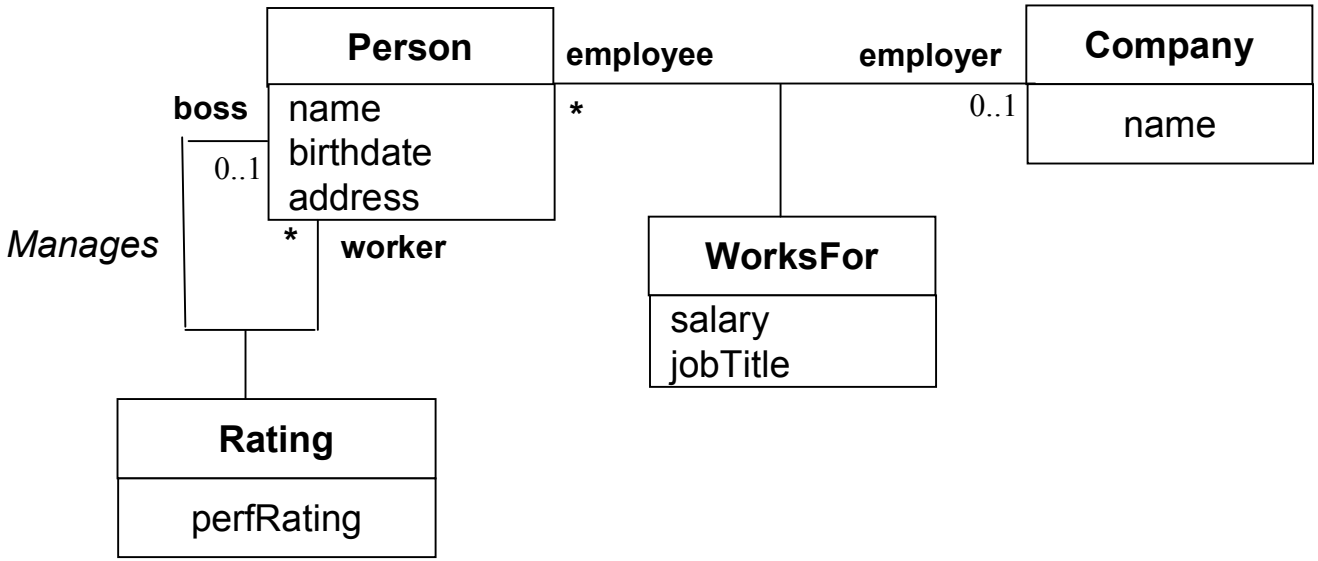
- An association class is an association that is also a class.
- Many to many associations provide a compelling rationale for association classes.



/etc/file1	read	Joe
/etc/file1	read/write	Mary
/etc/file2	read/write	Joe

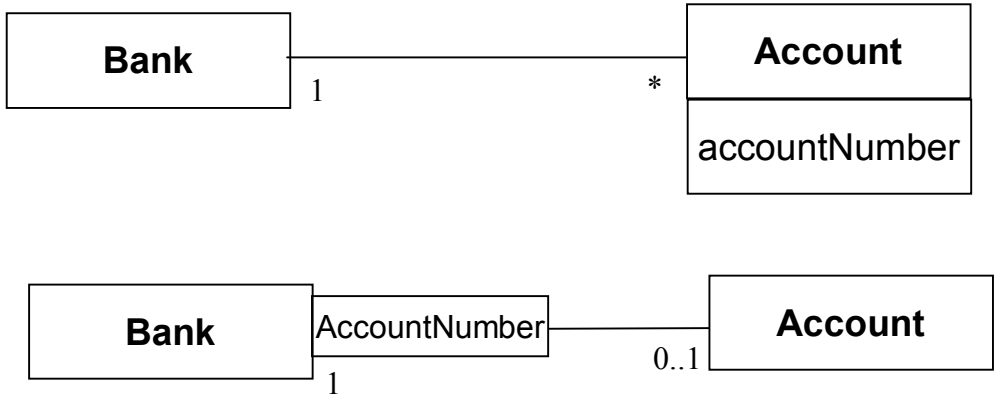
Association Classes

- **Example: Each person working for a company receives a salary and has a job title. The boss evaluates the performance of each worker.**



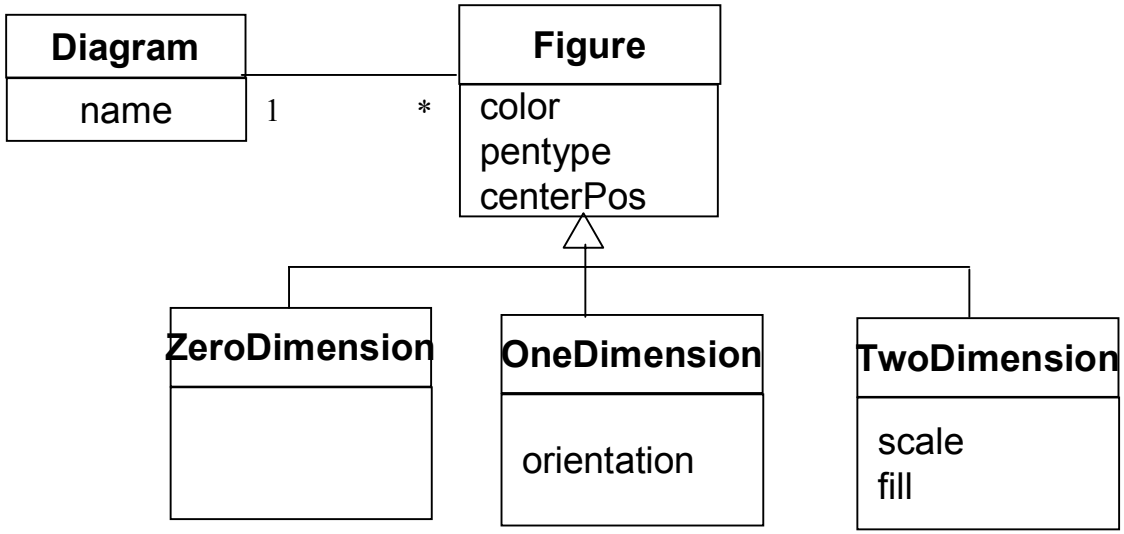
Qualified Associations

- **A qualified association is an association in which an attribute called the qualifier disambiguates the objects for a "may" association end.**



Generalization and Inheritance

- Generalization is the relationship between a class (the superclass) and one or more variations of the class.
- Generalization organizes classes by their similarities and differences, structuring the description of objects.



Use of Generalization

- Polymorphism
- Structuring the description of objects – Creating a taxonomy .
- Reuse of code and data

Navigation of Class Models

- Navigation enables exercising a model to uncover hidden flaws and omission.
- Example: Managing Credit Card Accounts
  - An institution may issue many credit card accounts, each identified by an account number. Each account has a maximum credit limit, a current balance, and a mailing address. The account serves one or more customers who reside at the mailing address. The institution periodically issues a statement for each account. The statement lists a payment due date, finance charge, and minimum payment. The statement itemizes various transactions that have occurred throughout the billing interval: cash advances , interest charges, purchases, fees, and adjustments to the account. The name of the merchant is printed for each purchase.