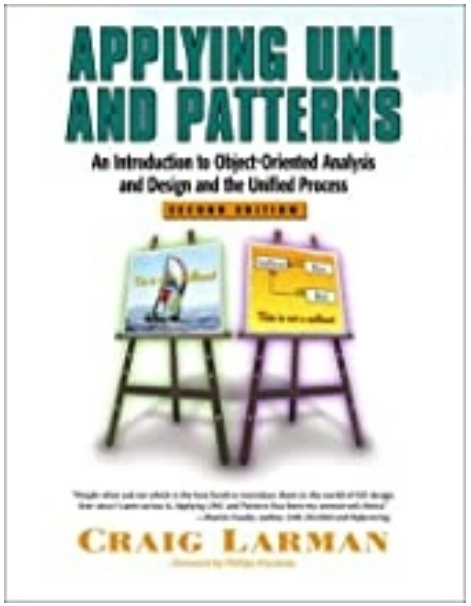
Domain Models

From Larman, Chapter 9

This is one of the most important chapters in the book.



What is a Domain Model?

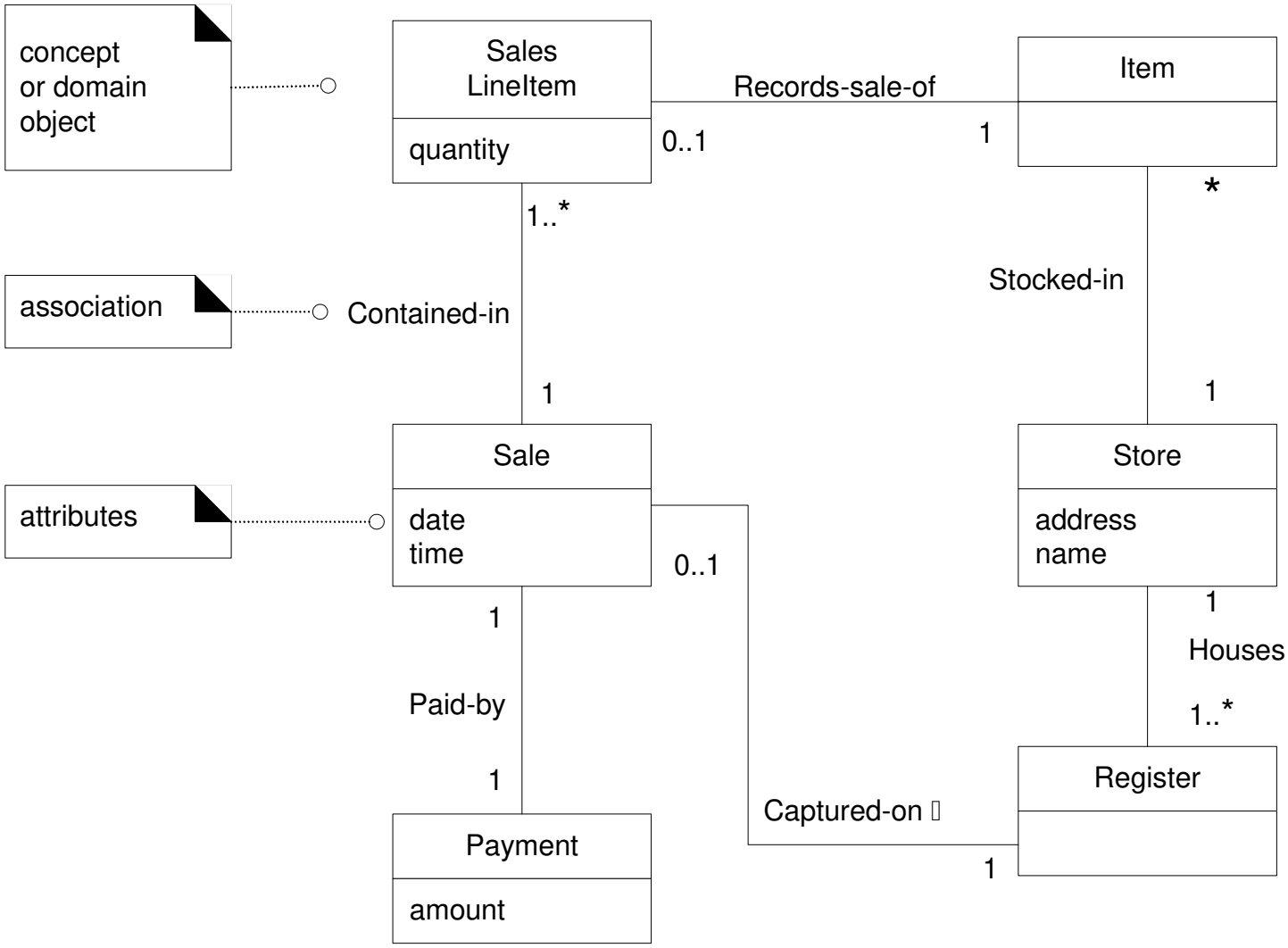
- Model of the problem domain, showing concepts, important attributes, and relationships.

- Not a software model.

Concepts in "Make a Sale" for a Point of Sale (POS) Application

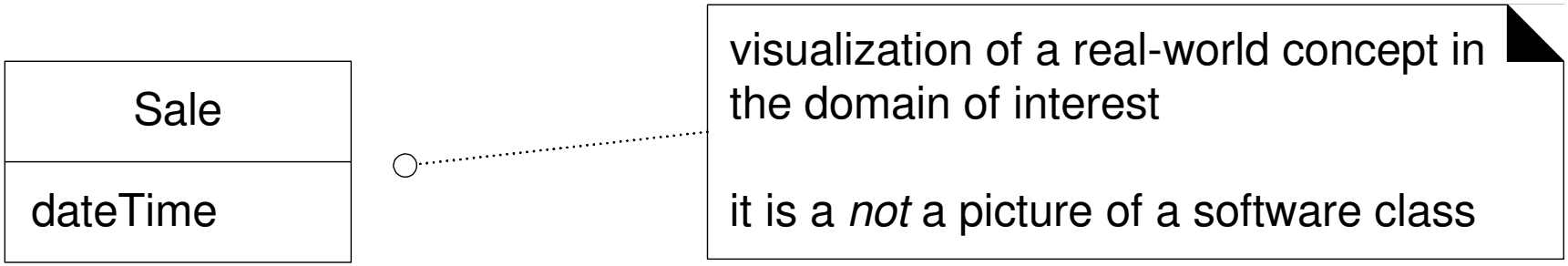
Register	Item	Store	Sale
Sales LineItem	Cashier	Customer	Ledger
Cash Payment	Product Catalog	Product Description	

Domain Model for POS based on "Make a Sale"



A Domain class

A Domain Class represents a "thing" or concept in a model of the application domain. It is not a software class.



In a Point-of-Sale application, a "Sale" is an important part of the model.

3 Techniques to Discover Domain Classes

1. Look for **noun phrases** in User Stories

2. Use a **category list**

3. Use a similar **existing project**

Look for Noun Phrases in Use Cases

This is a Use Case for how "make a sale" is performed..

1. Customer arrives at POS with items to purchase.
2. Cashier starts a new sale on register.
3. Cashier scans an item's barcode.
4. POS looks up Product using barcode and adds a line item to the sale.
5. Cashier repeats steps 2-4 until no more items.
6. Register displays sale total with tax.
7. Cashier asks Customer for payment.
8. Customer pays for items.
9. Cashier enters payment into POS.

Look for **Noun Phrases**

1. Customer arrives at POS with items to purchase.
2. Cashier starts a new sale on register.
3. Cashier scans an item's barcode.
4. POS looks up Product using barcode and adds a line item to the sale.
5. Cashier repeats steps 2-4 until no more items.
6. Register displays sale total with tax.
7. Cashier asks Customer for payment.
8. Customer pays for items.
9. Cashier enters payment into POS.

Noun Phrases

Noun phrases are candidates for domain classes, but not all are classes. Some noun phrases may be attributes, instance names, or not important to the model.

Customer

Cashier

POS

Sale

Register

Product

Line item

Payment

barcode

items

sale total

probably *attributes* rather than *domain classes*

Conceptual Category List

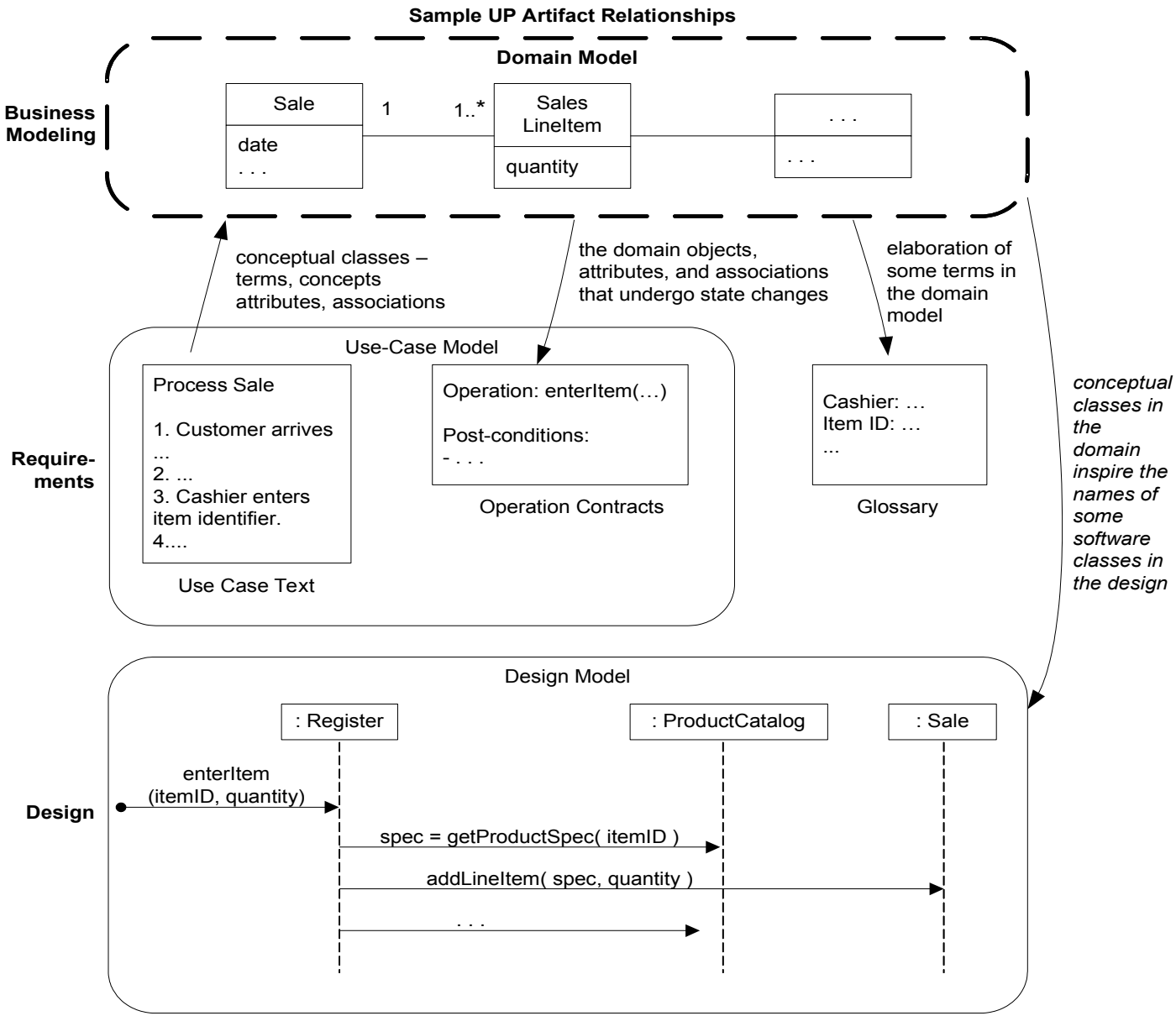
A list of common *categories* for domain classes, to help you think of things.

Table 9.1 in Larman's book

- Business transactions
- Product or service related to transaction
- Where is transaction recorded?
- Catalogs
- Roles of people related to actors in UC

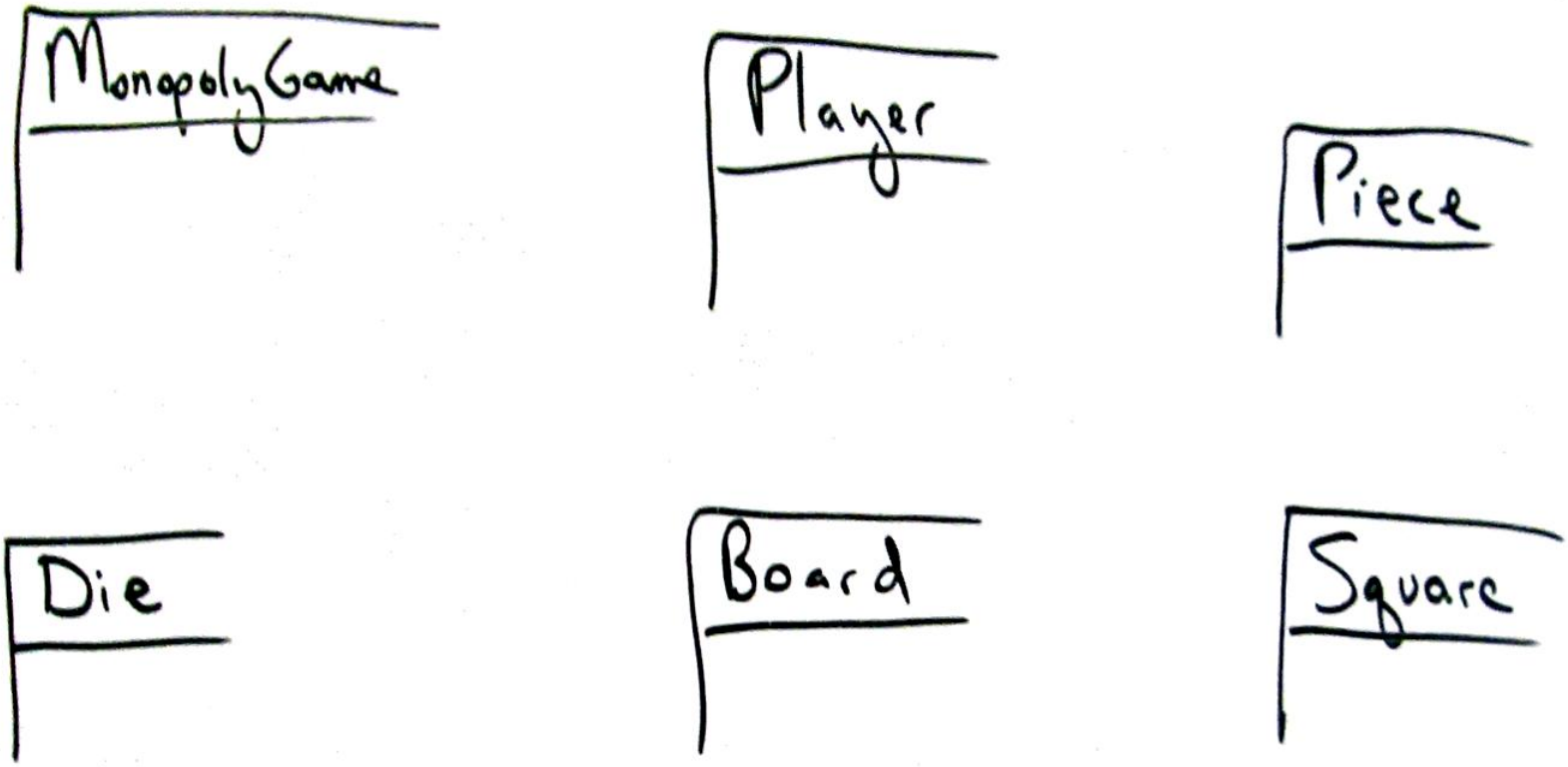
Link to the list posted on classroom.

Design Documents for "Make a Sale"



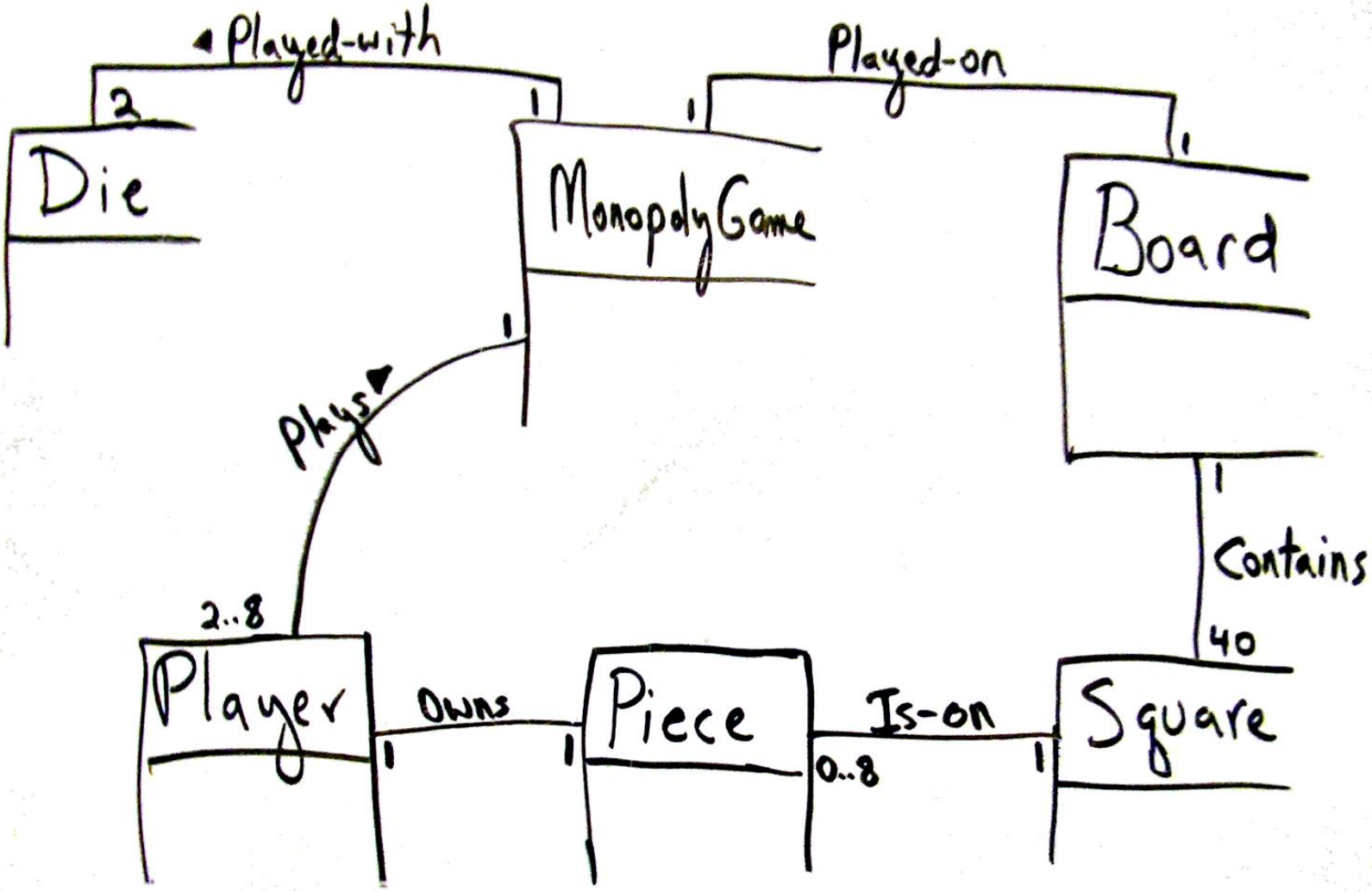
Domain Model for Monopoly Game

Don't try to be beautiful or complete

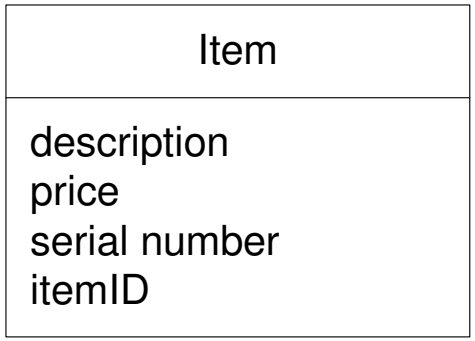


*What is **missing** here?*

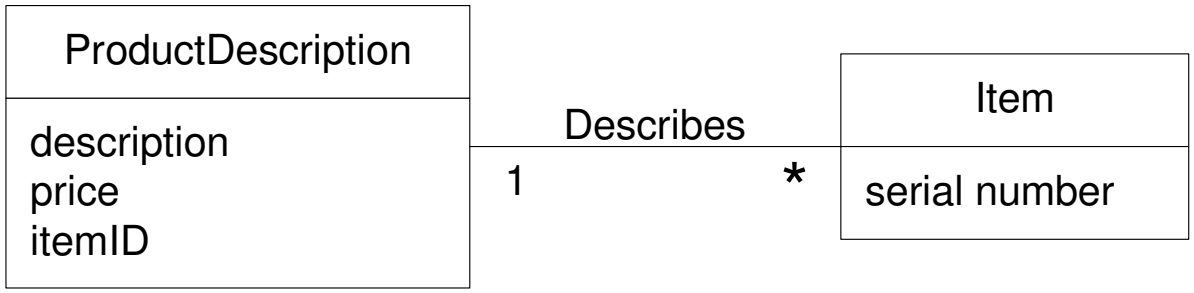
Express relationships



Prefer Associations over attributes

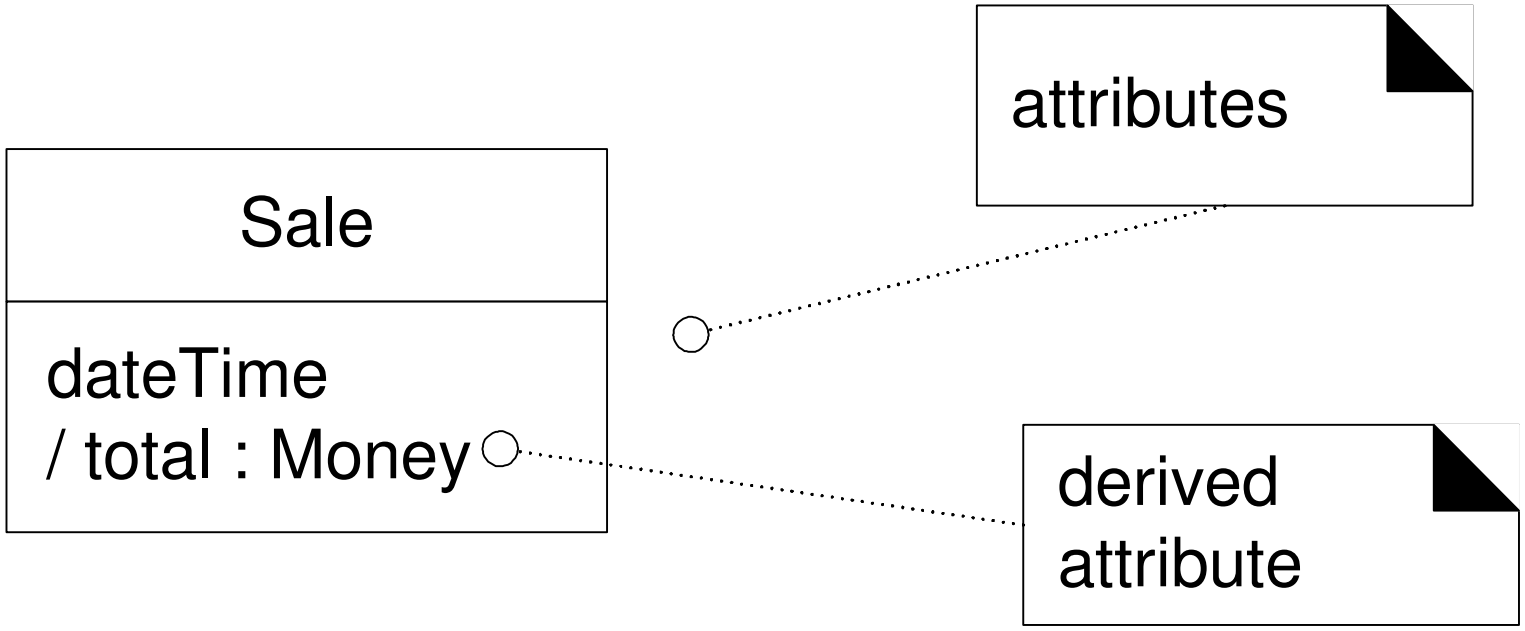


Worse

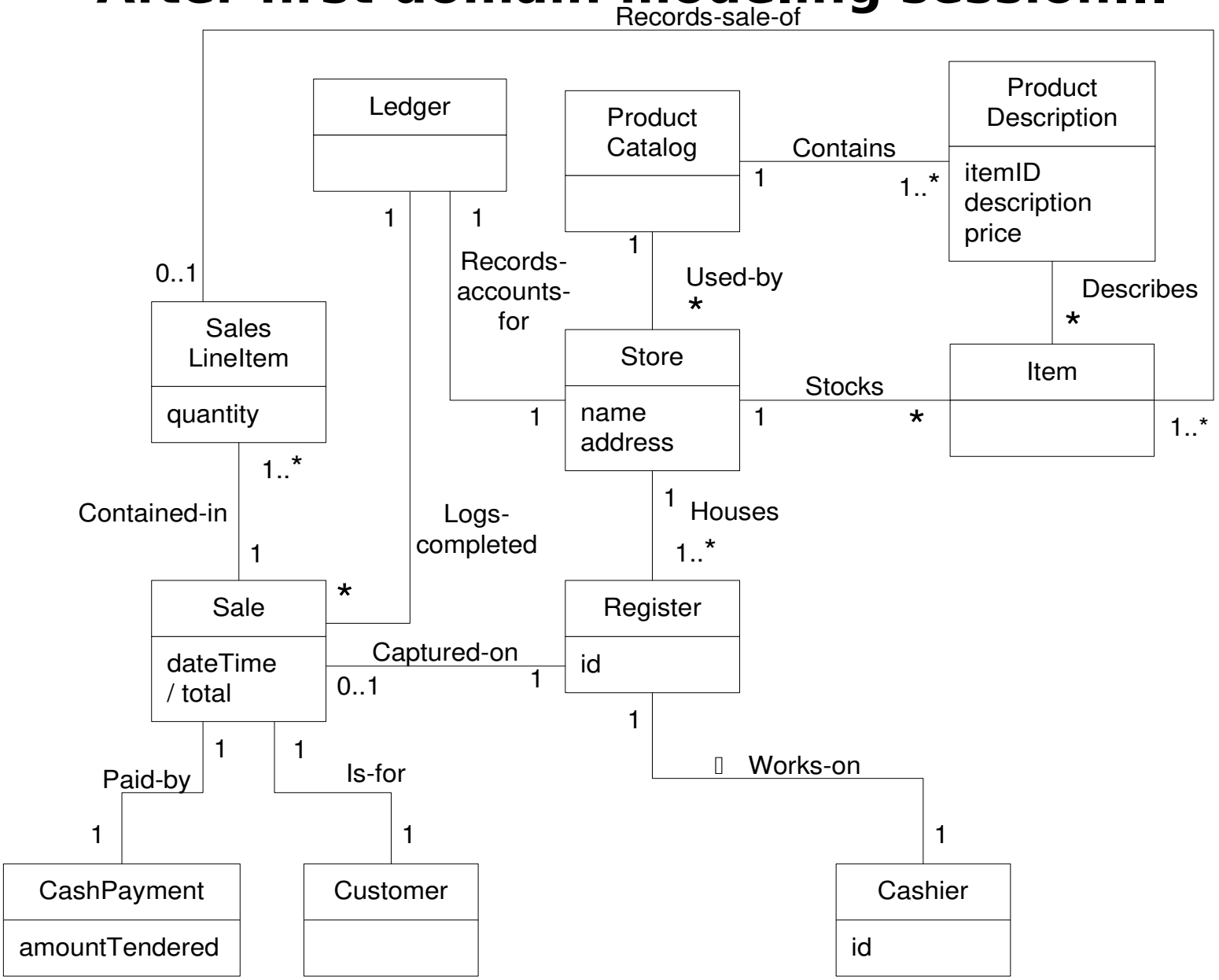


Better

OK to show "derived" attributes



After first domain modeling session...



Your Turn

Construct a domain model for your project.

- Analysis of your Use Cases (or User Stories)
- Use a category list
- Don't be influenced by what other teams are doing – design for yourself