More Domain Model Refinement

Shawn Bohner

Office: Moench Room F212

Phone: (812) 877-8685

Email: bohner@rose-hulman.edu

Q1

ROSE-HULMAN INSTITUTE OF TECHNOLOGY

Today’s Agenda

- Looking Ahead to finishing the term strong
  - Optional Final Exam
  - Design Studios
- More Domain Model Refinement

Final Exam

- 8:00am on Monday, Feb. 22nd
- Exam is *Optional*
  - If you don't take the exam, we'll use your first exam grade as your final exam grade
  - Sign-up for exam by Tuesday of 10th week
    - If you sign-up, you must take the exam
    - Taking the exam can improve or lower your grade

Design Studio: Project Focused

- A chance to get more help on your projects
- Participate in more active learning design sessions without taking more of your time
- Starting Thursday: 20 minutes sessions
  - ~5 minutes: team describes problem and current solution (if any)
  - ~3 minutes: class thinks about questions, alternative approaches
  - ~12 minutes: on-board design with team modeling and instructor advising/facilitating

Suggestions for Design Studio Topics

- Look for problems where there are lots of “which class should be responsible for X” questions

- Challenging design problems you’ve already faced and how you solved them

- Current design problems that you haven't solved yet

- Future extensions that will need to be considered

Design Studio Sign-up

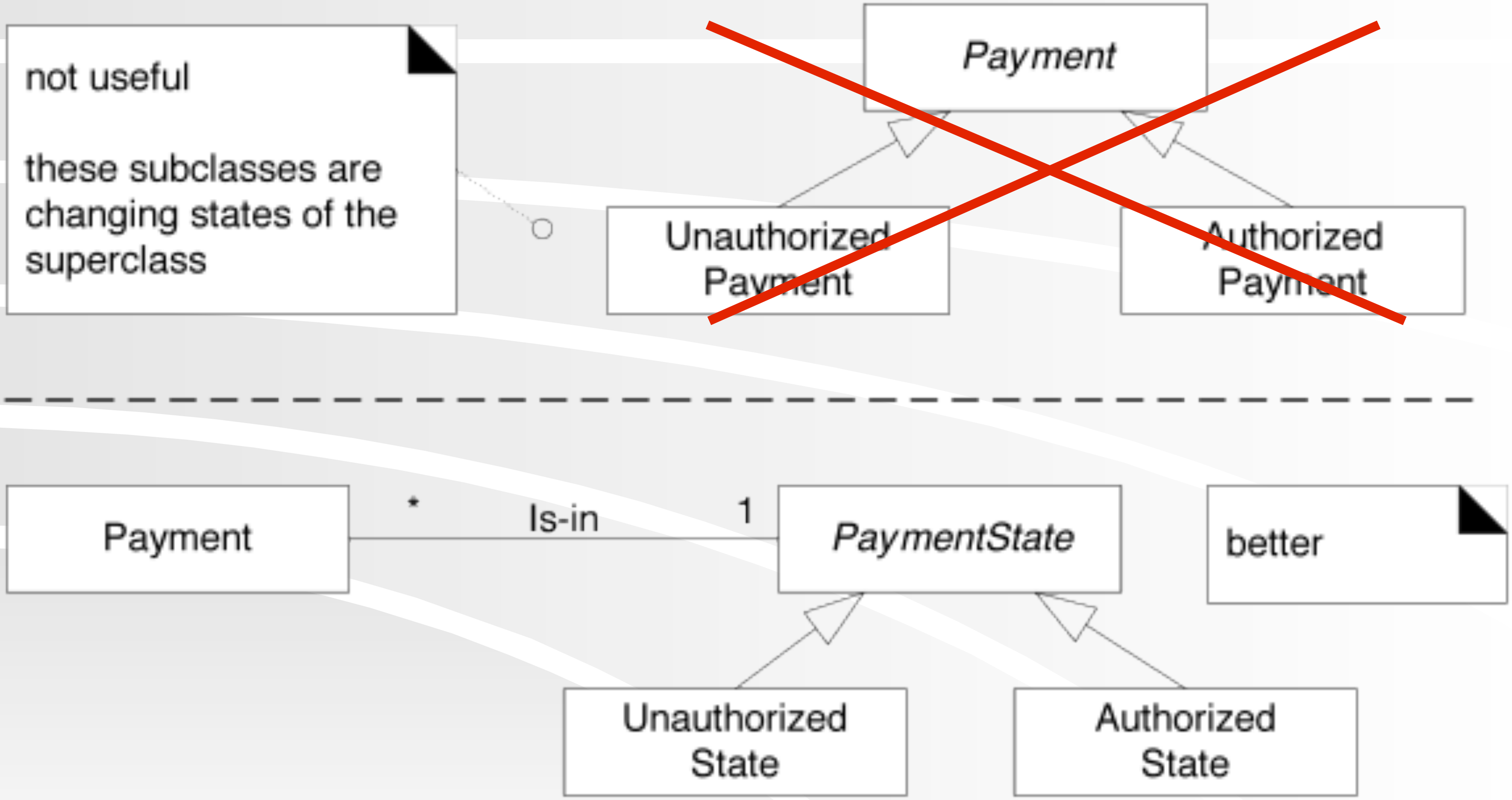
	Monday	Tuesday	Thursday
8th week	Yesterday	Today	T11
9th week	T16	T12	T13
10th week	T15	T14	Course Wrap-up

More on Domain Modeling

Modeling Changing States

- Suppose a concept X has multiple states
  - Draft vs. Sent Email
  - Purchase Request vs. Purchase Order
- Do not model the states as subclasses of X!
- Instead:
  - Define a State hierarchy and associate X with State or,
  - Ignore showing the states in the domain model

State Example



not useful

these subclasses are changing states of the superclass

better

Association Classes: Consider…

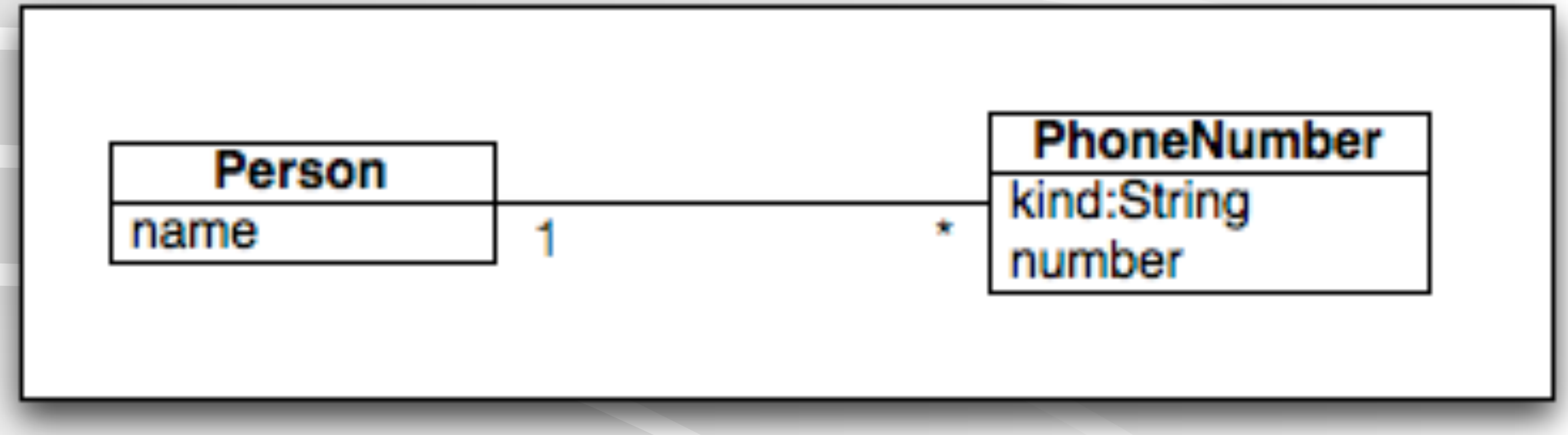
In NextGen POS:

- Authorization Services assign a *merchant ID* to each store
- Payment Authorization Request from store to service must use the *merchant ID*
- A store has a different *merchant ID* for each service

Where should the *merchant ID* appear in the domain model?

Guideline

If a class C can simultaneously have many values for the same attribute A, put A in another class associated with C



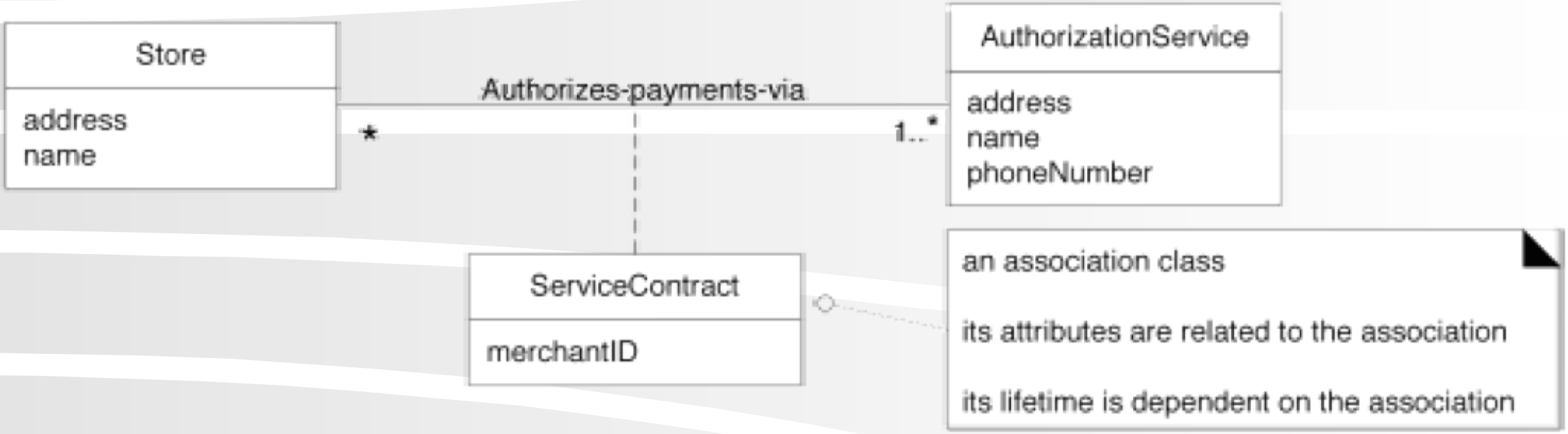
Guideline

Association class might be useful in a domain model if:

- The association has a related attribute
- Instances of the association class can only last as long as the association does
- There is a many-to-many association between two concepts and information is needed to distinguish the pairs

Q2

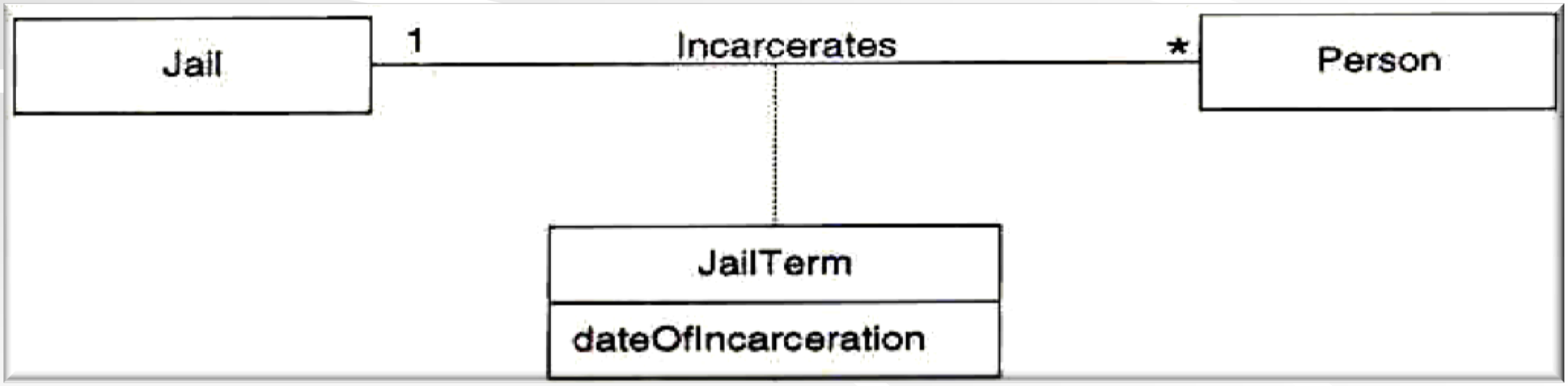
Association Class Examples



an association class

its attributes are related to the association

its lifetime is dependent on the association

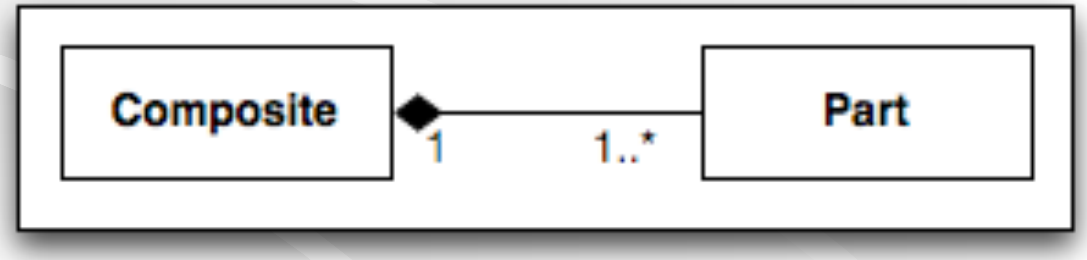


Q3

Recall, Composition

A composition relationship implies:

- An instance of the part belongs to only *one* composite instance at a time
- The part must *always belong* to a composite
- The composite is responsible for creating/deleting the parts



Not to be confused with the composite pattern, which is similar. (sigh)

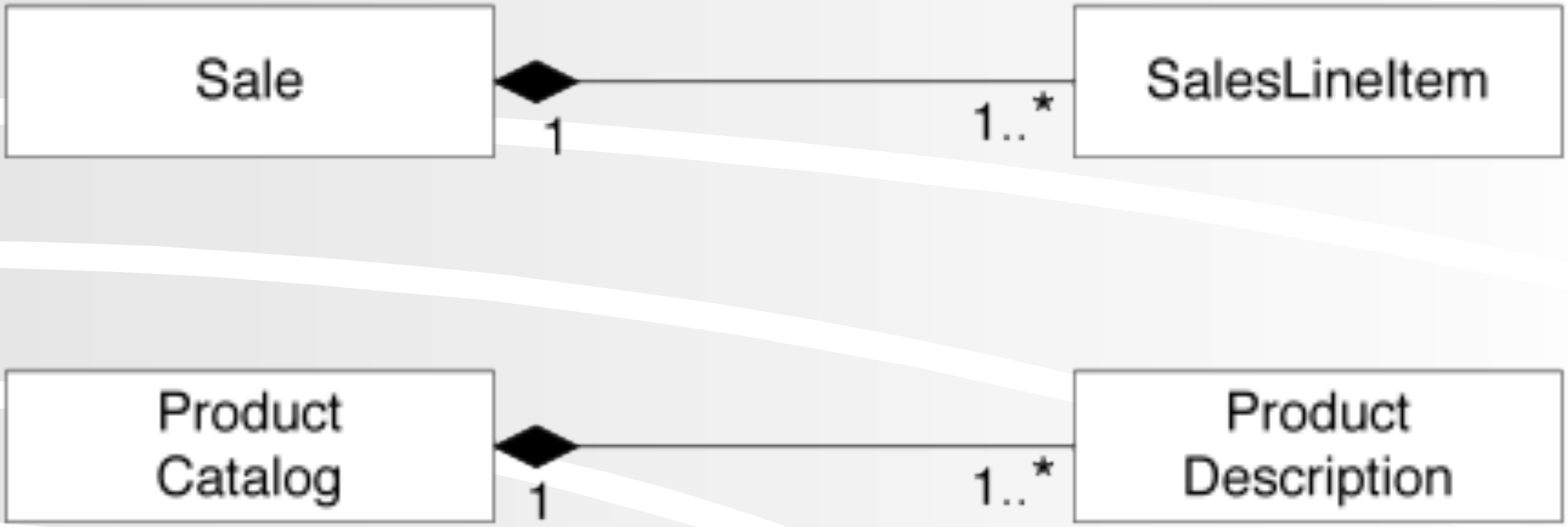
Composition in Domain Models

- Guideline: If in doubt, leave it out.

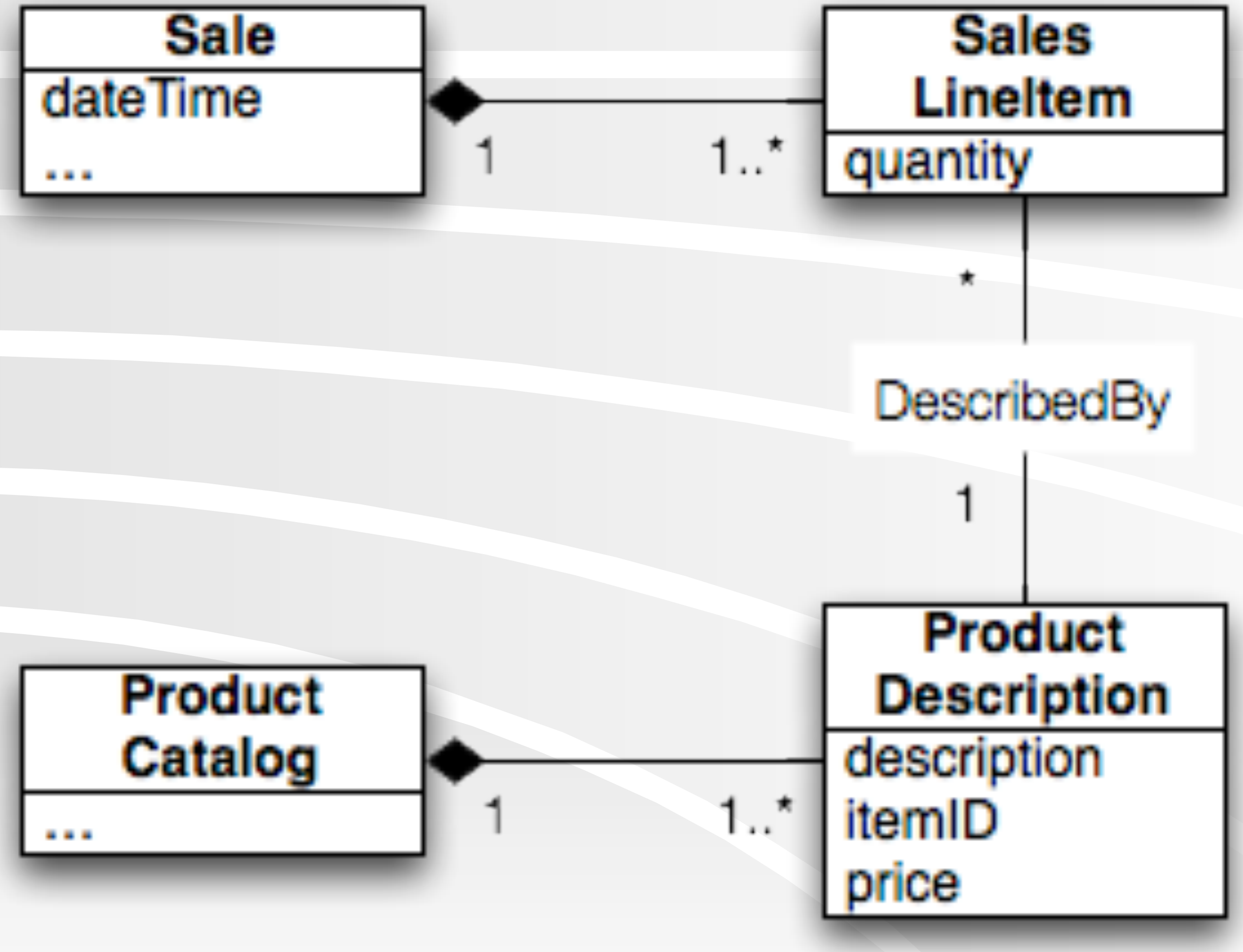
- But, consider showing composition when:
  - Lifetime of part is bounded within lifetime of composite
  - An obvious whole-part physical assembly exists
  - Composite properties propagate to the parts
  - Composite operations propagate to the parts

Q4

Composition in NextGen Domain Model

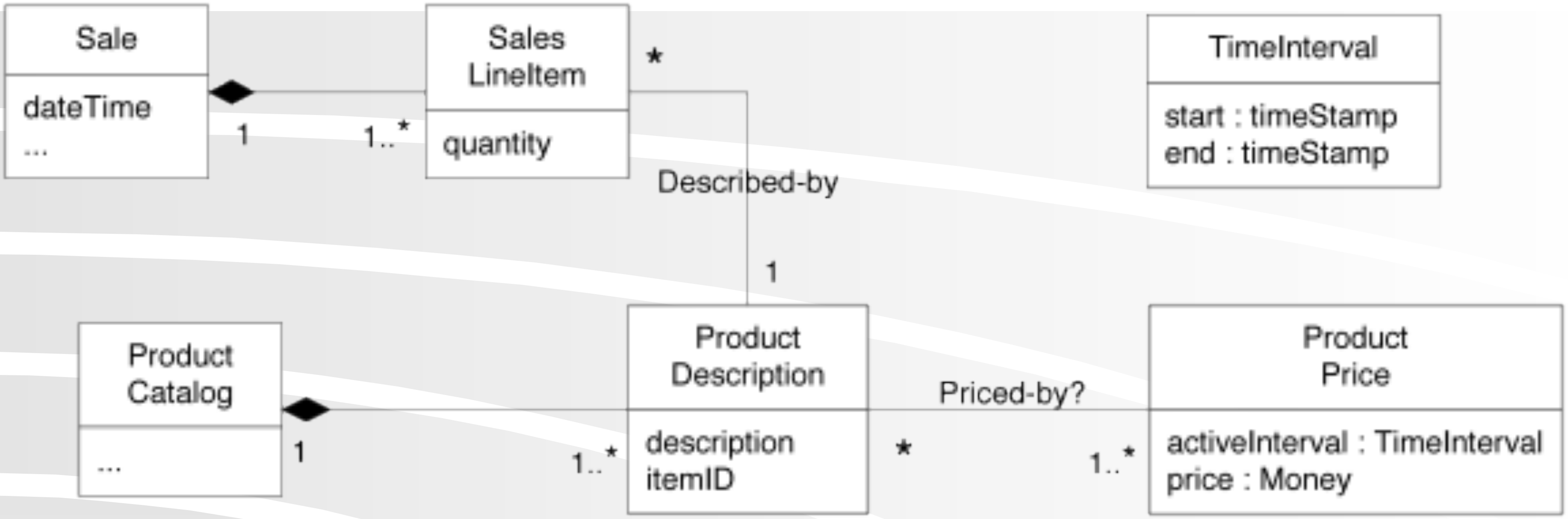


Problem: What happens to old Sales when a product's price changes?



Solutions?

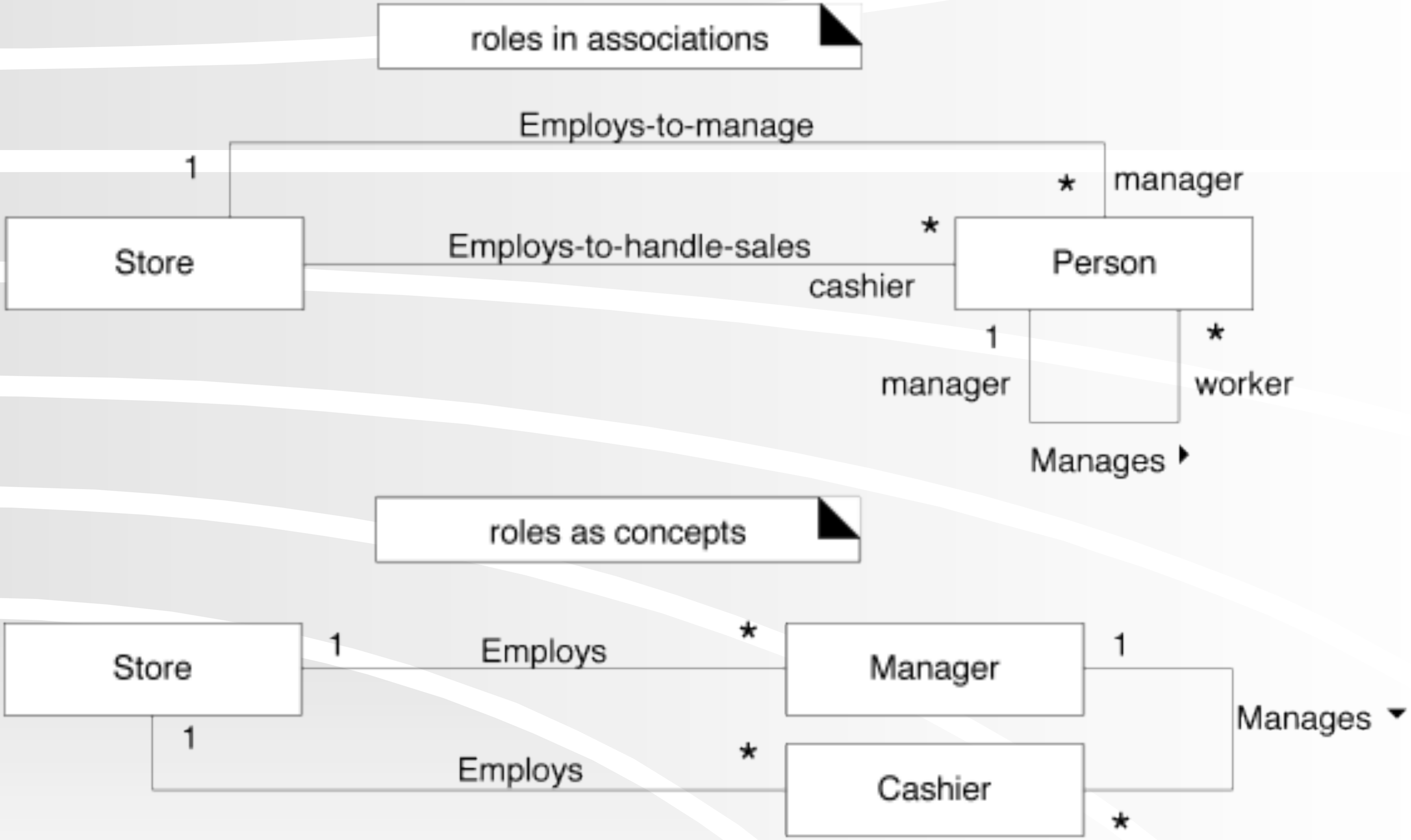
Time Intervals



Q5

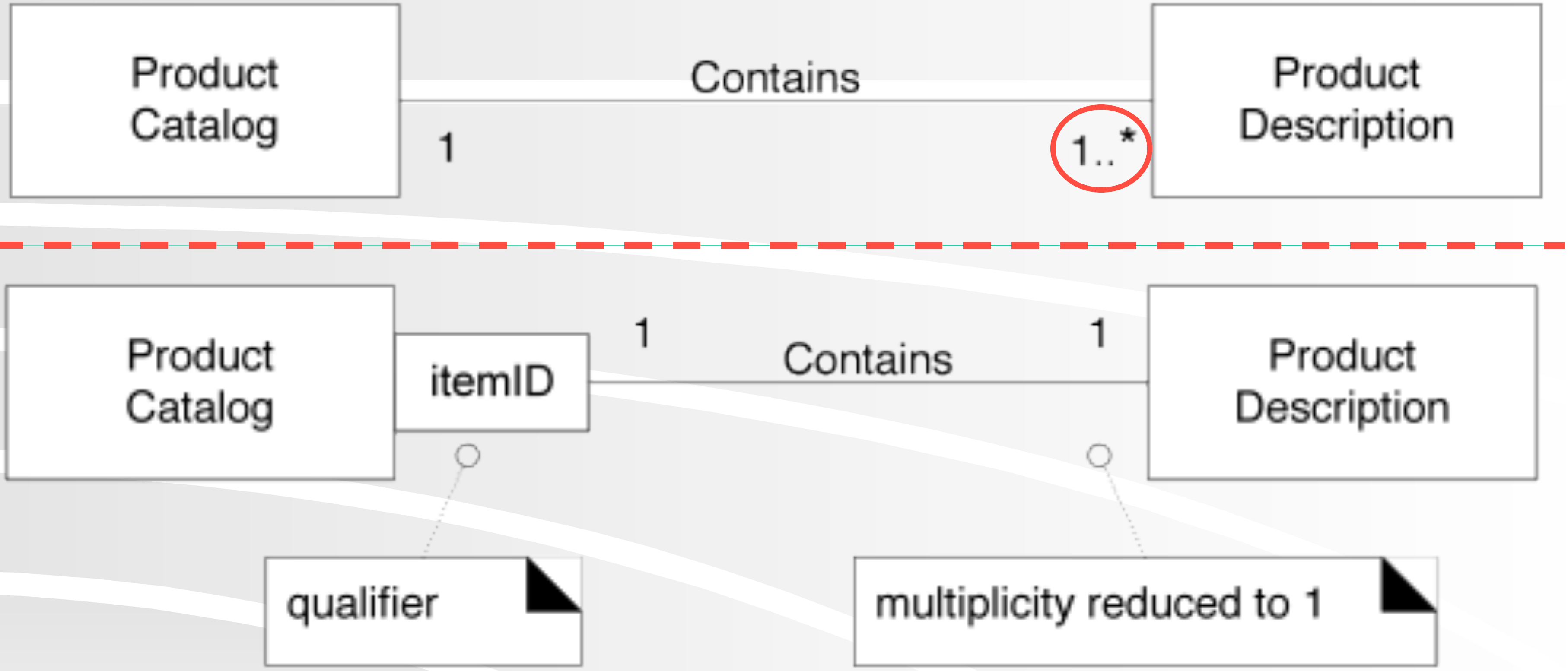
Rolls…

Er, Roles…



Pros and cons?

Qualified Associations

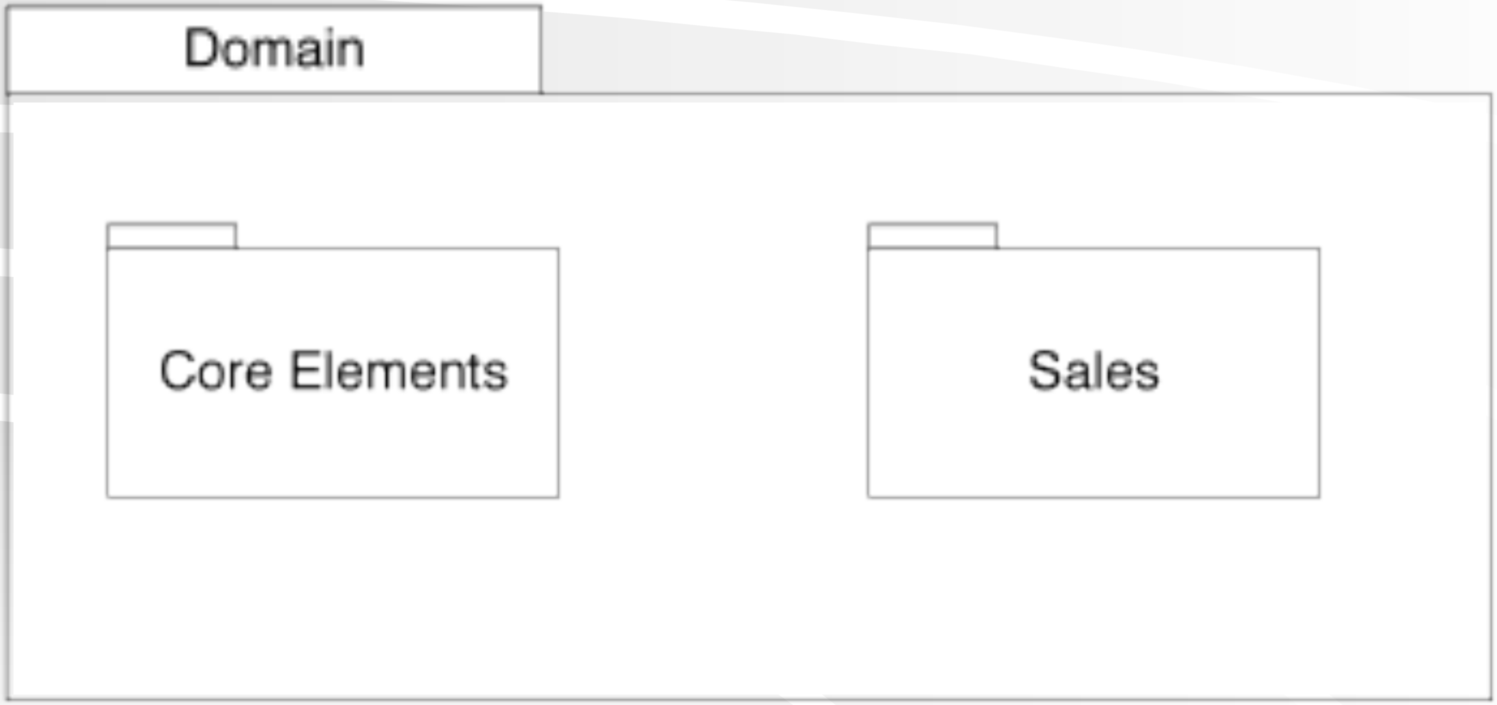


Don't overdo it…

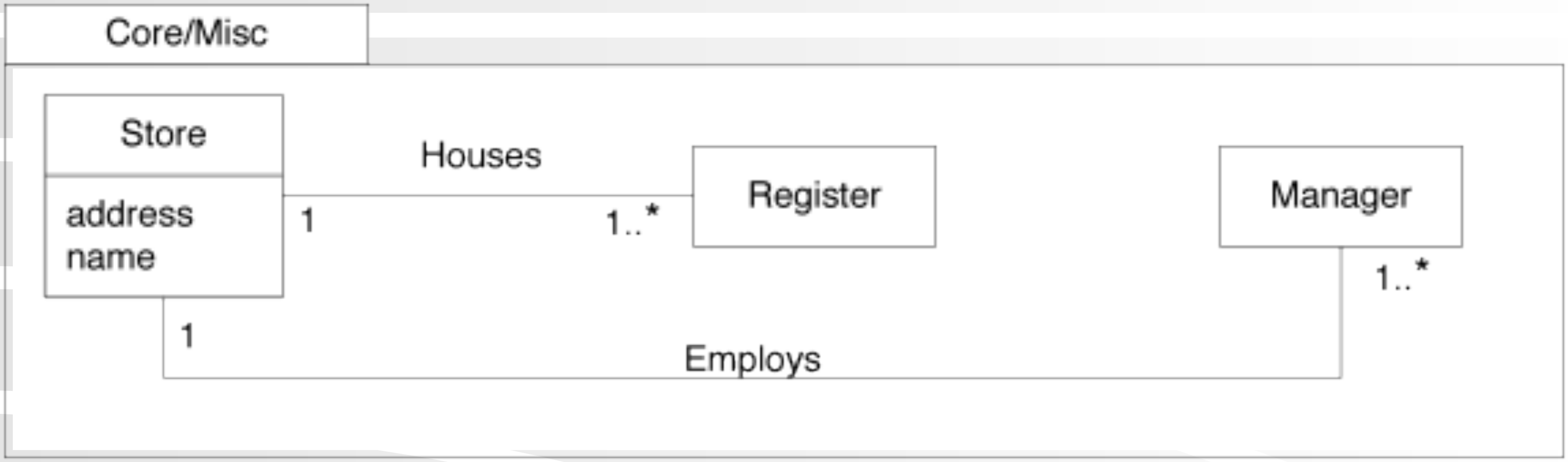
Q6

Splitting Domain Model into Packages

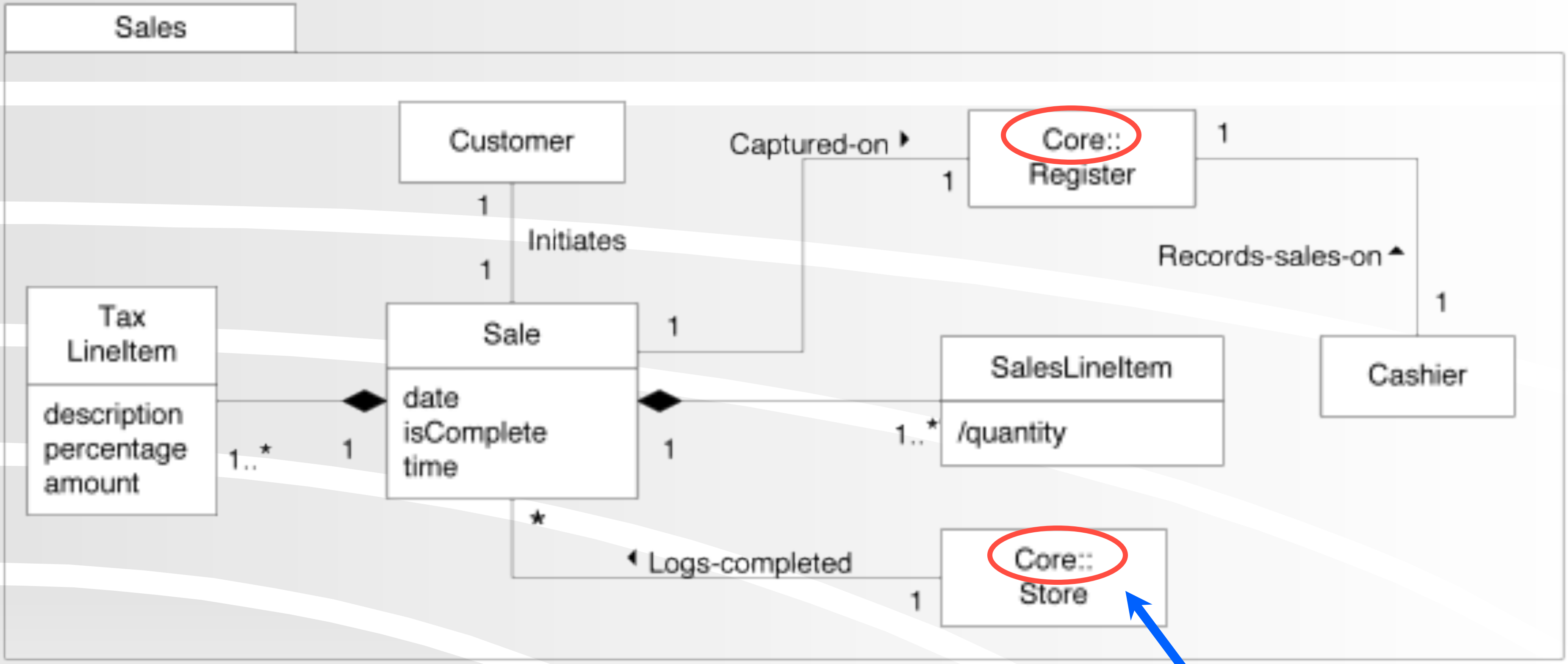
- Supports parallel analysis work
- NextGen POS example:



NextGen POS Core Package



NextGen POS Sales Package with Assocations from Core Package



Store is “owned” by Core package, but also shown here to illustrate associations

Q7

If Domain is big enough to partition into packages, Group conceptual Classes that:

- Are in the same subject area
- Are in the same class hierarchy
- Participate in the same use cases
- Are strongly associated

Q8

Homework and Milestone Reminders

- Read Chapters 32 and 33
- Homework 7– Gang of Four (GoF) Patterns on Video Store Design
  - Due by 5:00pm Tuesday, February 2nd, 2010
- Milestone 5– Iteration 3 Junior Project System with finalized Design Document
  - Due by 11:59pm Friday, February 19th, 2010.