

# 面向对象程序的分析与设计 Object-Oriented Analysis and Design

Lecture 13  
Review

Prof. S. Xu

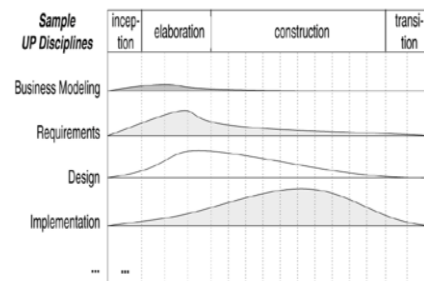
## Contents

- Review on UP

## Review on UP

3

## UP Disciplines and phases



The relative effort in disciplines shifts across the phases.

This example is suggestive, not literal.

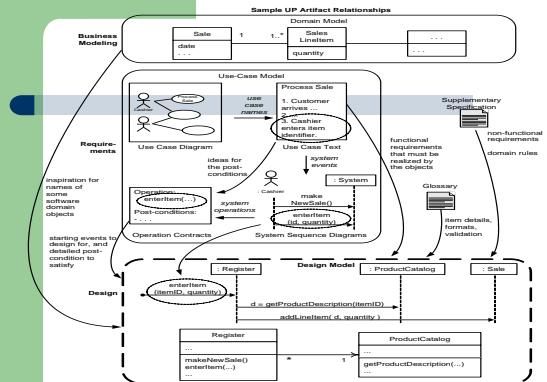
4

## Customize UP, The UP Development Case

Table 2.1. Sample Development Case. s - start; r - refine

Discipline	Practice	Artifact	Incep.	Elab.	Const.	Trans.
Business Modeling	agile modeling req. workshop	Domain Model	I1	E1..En	C1..Cn	T1..T2
Requirements	req. workshop vision box exercise dot voting	Use-Case Model	s	r		
		Vision	s	r		
		Supplementary Specification	s	r		
		Glossary	s	r		
Design	agile modeling test-driven dev.	Design Model		s	r	
		SW Architecture Document		s		
		Data Model		s	r	
Implementation	test-driven dev. pair programming continuous integration coding standards	...				
Project Management	agile PM daily Scrum meeting	...				
...						

## Review



## Review

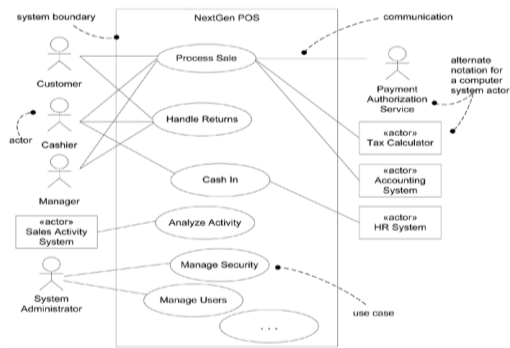
- Do Use Case Modelling (first part)
  - use case description
  - Use case diagram
  - Supplementary document
- Create Domain Model
  - based on requirements and **use cases**
- Do Use Case Modelling (second part)
  - create **System Sequence Diagrams** and identify system operations
  - Clarify system operations with **Operation Contracts**
- Conduct Design Modelling
  - Assign "doing" responsibilities with **Interaction Diagrams** (Communication and Sequence Diagrams)
  - Assign "knowing" responsibilities with **Design Class Diagrams**

## Review

- Create Implementation Model
  - **Convert Design document into Code**
    - there is a **translation process**
      - from **UML class diagrams** to **class definitions**, and
      - from **interaction diagrams** to **method bodies**.

## Use Case Modelling (Part 1)

## Use Case Diagrams



## Use Case Description

### Use Case UCI: Process Sale

Primary Actor: ...

... as before ...

#### Main Success Scenario:

Actor Action (or Intention)

System Response

1. Customer arrives at a POS checkout with goods and/or services to purchase.

2. Cashier starts a new sale.

3. Cashier enters item identifier.

Cashier repeats steps 3-4 until indicates done.

6. Cashier tells Customer the total, and asks for payment.

7. Customer pays.

4. Records each sale line item and presents item description and running total.

5. Presents total with taxes calculated.

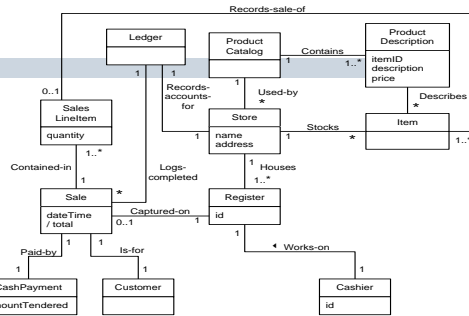
8. Handles payment.

9. Logs the completed sale and sends information to the external accounting (for all accounting and commissions) and inventory systems (to update inventory). System presents receipt.



## Domain Modelling

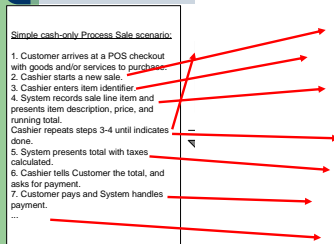
## Domain Model



## Use Case Modelling (Part 2)

## SSD vs. use cases

- An SSD shows system events for one scenario of a use case, therefore it is generated from inspection of a use case.



15

## Example

## Contract C02: enterItem

**Operation:** enterItem(itemID: ItemID, quantity: integer)

**Cross References:** Use Cases: Process Sale

**Preconditions:** There is a sale underway.

**Postconditions:**

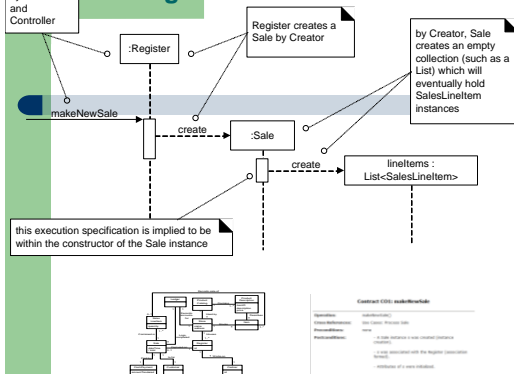
- A SalesLinetItem instance sli was created **[instance creation]**
- sli was associated with the current Sale **[association formed]**
- sli.quantity became quantity **[attribute modification]**
- sli was associated with a ProductDescription, based on itemID match **[association formed]**

The categorizations such as "(instance creation)" are a learning aid, not properly part of the contract.

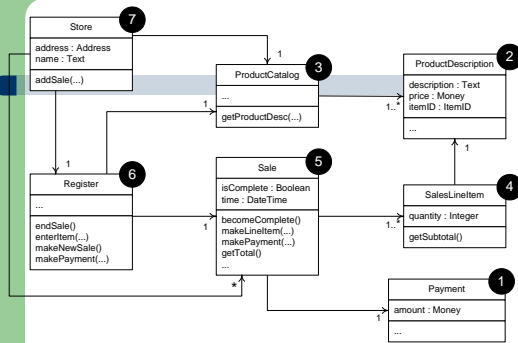
16

## OOD (use case realization)

## to Design makeNewSale?

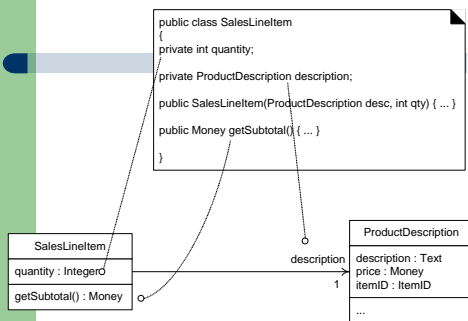


## Final Design Class Diagram

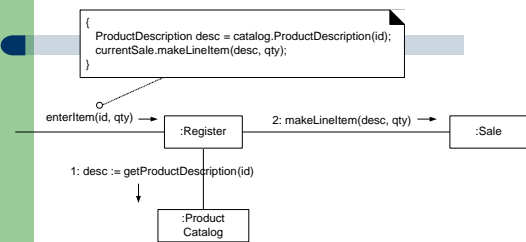


## Create Implementation Model

## Creating Classes from DCDs



## Creating Methods from Interaction Diagrams



- What is cloud computing?

- More on cloud computing