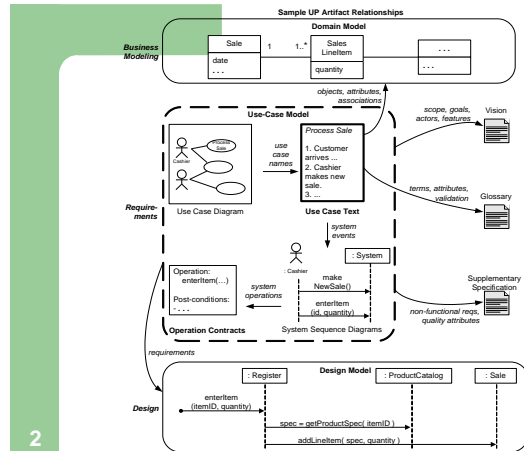


# 面向对象程序的分析与设计 Object-Oriented Analysis and Design

Lecture 6

Prof. S. Xu



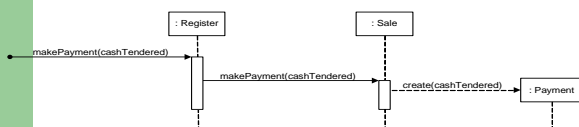
## Contents

- UML Sequence Diagram
- SSD (System Sequence Diagram) and Operation Contracts

## UML Sequence Diagram

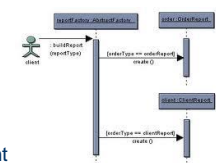
## Modeling Dynamic Behavior

- Sequence diagram
  - Depict object interaction by highlighting the **time ordering** of method invocation



## Sequence Diagrams

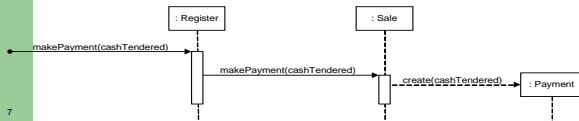
- Describe the flow of messages, events, actions between objects
- Show time sequences that are not easily depicted in other diagrams
- Typically used during analysis (SSD) and design (SD) to document and understand the logical flow of your system



Emphasis on time ordering!

## Sequence Diagram Key Parts

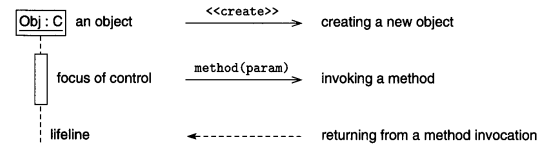
- **participant**: object or entity that acts in the diagram
- **message**: communication between participant objects
- **the axes** in a sequence diagram:
  - **horizontal**: which object/participant is acting
  - **vertical**: time (down -> forward in time)



7

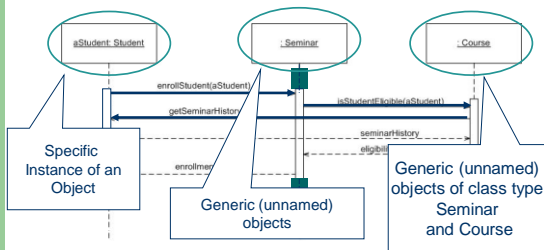
## Sequence Diagram Key Parts

- Message passing and return



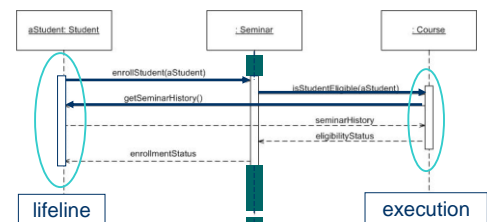
8

## Components

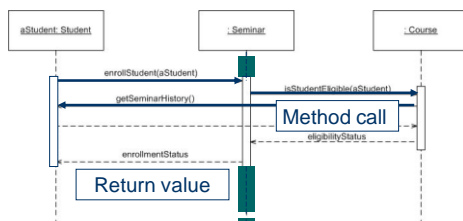


Objects: aStudent is a specific instance of the Student class

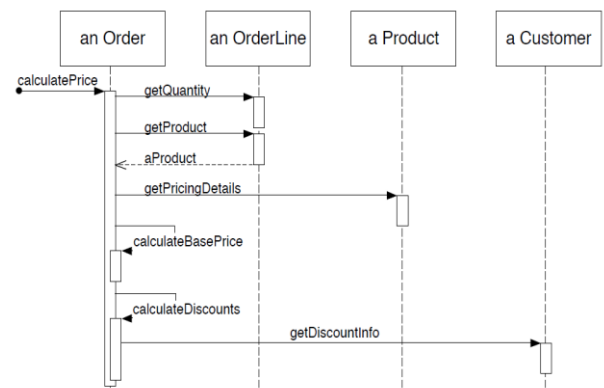
## Components



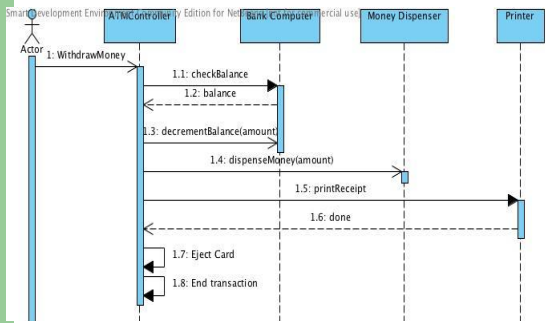
## Components



## Sequence diagram example



Another Example



More Information on Sequence Diagram

- More info on Sequence Diagram will be introduced in future lecture when needed



14

- [Future PC](#)

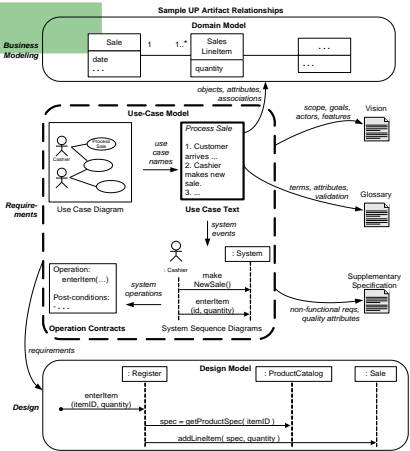
UP  
SSD (System Sequence Diagram)  
and Operation Contracts

16

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

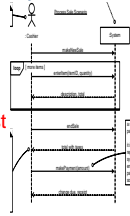


18

## System Sequence Diagrams (SSD)

### What is SSD?

- A system sequence diagram (SSD) is an artifact that illustrates input and output events related to the systems under discussion..

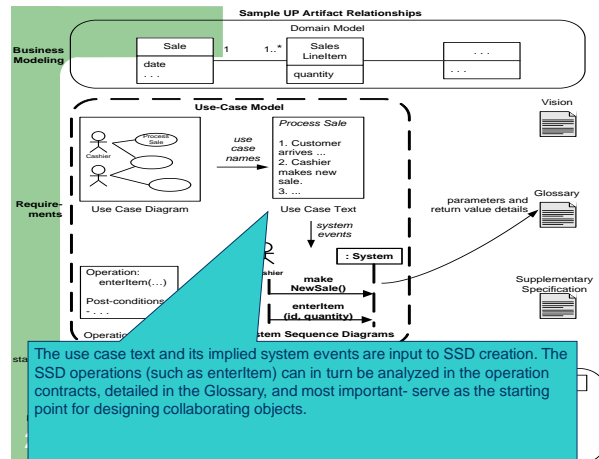


### Why study SSD?

- They are input to **operation contracts** and most importantly - **object design**.
- Domain model is static
  - We want to explore the dynamic aspect of the project domain.
  - A **visual representation** of use case.

- UML sequence diagram is used to illustrate SSD.

19

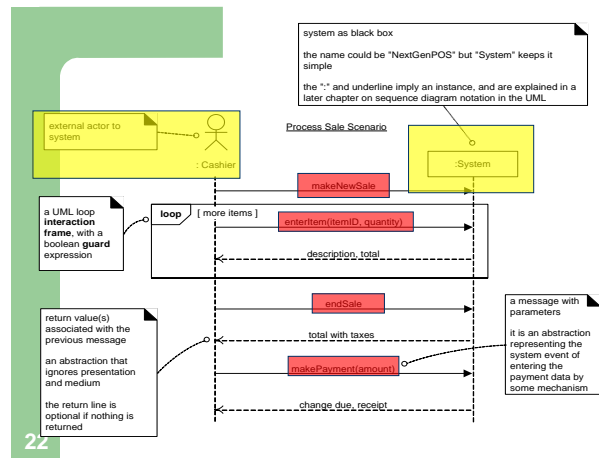


## Example NextGen SSD:

An SSD shows, for a particular course of events within a use case,

- the external actors that interact directly with the system,
- the system (as a black box), and
- the system events that the actors generate.

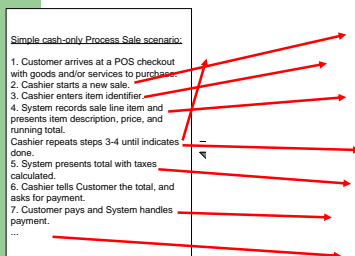
Time proceeds downward, and the ordering of events should follow their order in the scenario.



22

## 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.



23

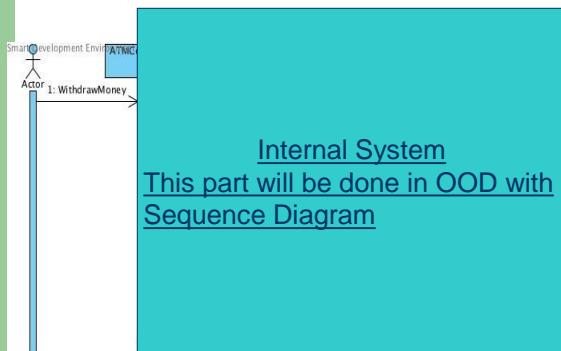
## Naming System Events and Operations

- scan(itemId) or enterItem(itemId)?

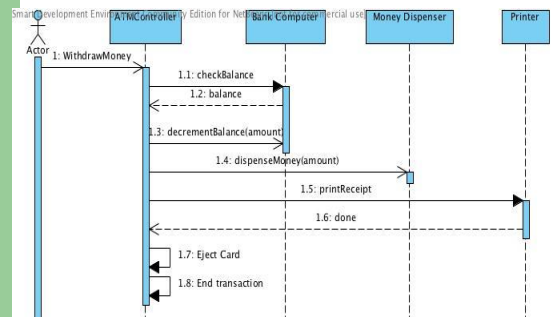
- Thus "enterItem" is better than "scan" (that is, laser scan) because it captures the intent of the operation abstract and nonconcrete. "enterItem" is a better name because it is used to capture the system event. It could be via laser scanner, keyboard, or input.
- It also improves clarity to start the name of a system event with a verb (add..., enter..., end..., make...). Since the requests.

24

## SSD in OOA vs SD in OOD



## SSD in OOA vs SD in OOD



## Review

- What is illustrated in SSD?

27

## Review

An Automatic Teller Machine (ATM) is a computer based machine, connected to a network, that offers, as basic functions to users, access to bank account (balance, bank transfers), deposit money and retrieval of money.

The ATM will service one customer at a time. A customer will be required to enter ATM Card number, personal identification number (PIN) – both of which will be sent to the database for validation. In the case of a **cash withdrawal**, a second message will be sent after the transaction has been physically completed (cash dispensed). If the database determines that the customer's PIN is invalid, the customer will be required to re-enter the PIN before a transaction can proceed. If a transaction fails for any reason other than an invalid PIN, the ATM will display an explanation of the problem, and will then ask the customer whether he/she wants to do another transaction.

The ATM will provide the customer with a printed receipt for each successful transaction, showing the date, time, machine location, type of transaction, account(s), amount, and ending and available balance(s) of the affected account ("to" account for transfers).

28

## Review

- Identify all primary use cases and related actors. Draw the results in a **use case diagram to show the major functions of ATM system**
- Write (**describe**) the **"Withdraw Money"** Use Case of the ATM system using the following template: Use Case name, participating actors, entry condition, exit condition, flow of events (for the main scenario only).
- Bounded by the provided **"Withdraw money"** use case, create a **system sequence diagram** in the UML notation. Show actors and system events

29

- Future Retail Industry



### Example: enterItem Postconditions

#### • Instance Creation and Deletion

- After the *itemID* and *quantity* of an item have been entered, what new object should have been created?
  - A *SalesLineItem*. Thus:
  - A *SalesLineItem* instance sli was created (instance creation).

37

### Example: enterItem Postconditions

#### • Attribute Modification

- After the *itemID* and *quantity* of an item have been entered by the cashier, what attributes of new or existing objects should have been modified?
  - The *quantity* of the *SalesLineItem* should have become equal to the *quantity* parameter. Thus:
  - *sli.quantity* became *quantity* (attribute modification).

38

### Example: enterItem Postconditions

#### • Associations Formed and Broken

- After the *itemID* and *quantity* of an item have been entered by the cashier, what associations between new or existing objects should have been formed or broken?
  - The new *SalesLineItem* should have been related to its *Sale*, and related to its *ProductDescription*. Thus:
    - sli was associated with the current *Sale* (association formed).
    - sli was associated with a *ProductDescription*, based on *itemID* match (association formed).

39

### Exercise

Assume that you have to develop a computer system for the AU. The library contains books and each book may have several copies. Only members of the library may borrow books. The system checks whether the potential borrower is a member of the library, and whether there is a reservation on the book. If both check succeed the system records that the book is on loan. Otherwise it refuses the loan.

A library member may ask to extend the loan of the book. The system then checks whether there is a reservation on the book. If so, the system refuses to extend the loan. Otherwise it records the extension of the loan.

### Exercise

- Draw a use case diagram that represents the activities of the members of the library
- Draw a **system sequence diagram** that shows scenario of the use case **Borrow copy of book** in which the user is permitted to borrow the book, and the system records that the book is on loan
- Describe the **operation contracts** for those system operations found above.

### Review

- True or false (circle one): An operation contract details an entire use case
- Which section of an operation contract is the most important?

**Review**

- One of the four phases of Unified Process include
  - 1) Specification;
  - 2) Design
  - 3) Testing ;
  - 4) Construction

43

**Review**

- A well designed Object Oriented System is: \_\_\_\_\_
  - A. Is loosely cohesive and loosely coupled.
  - B. Is tightly cohesive and loosely coupled.
  - C. Is tightly cohesive and tightly coupled.
  - D. Is loosely cohesive and tightly coupled

44

- What are future wearable devices?