(11224) INTRODUCTION TO SOFTWARE ENGINEERING

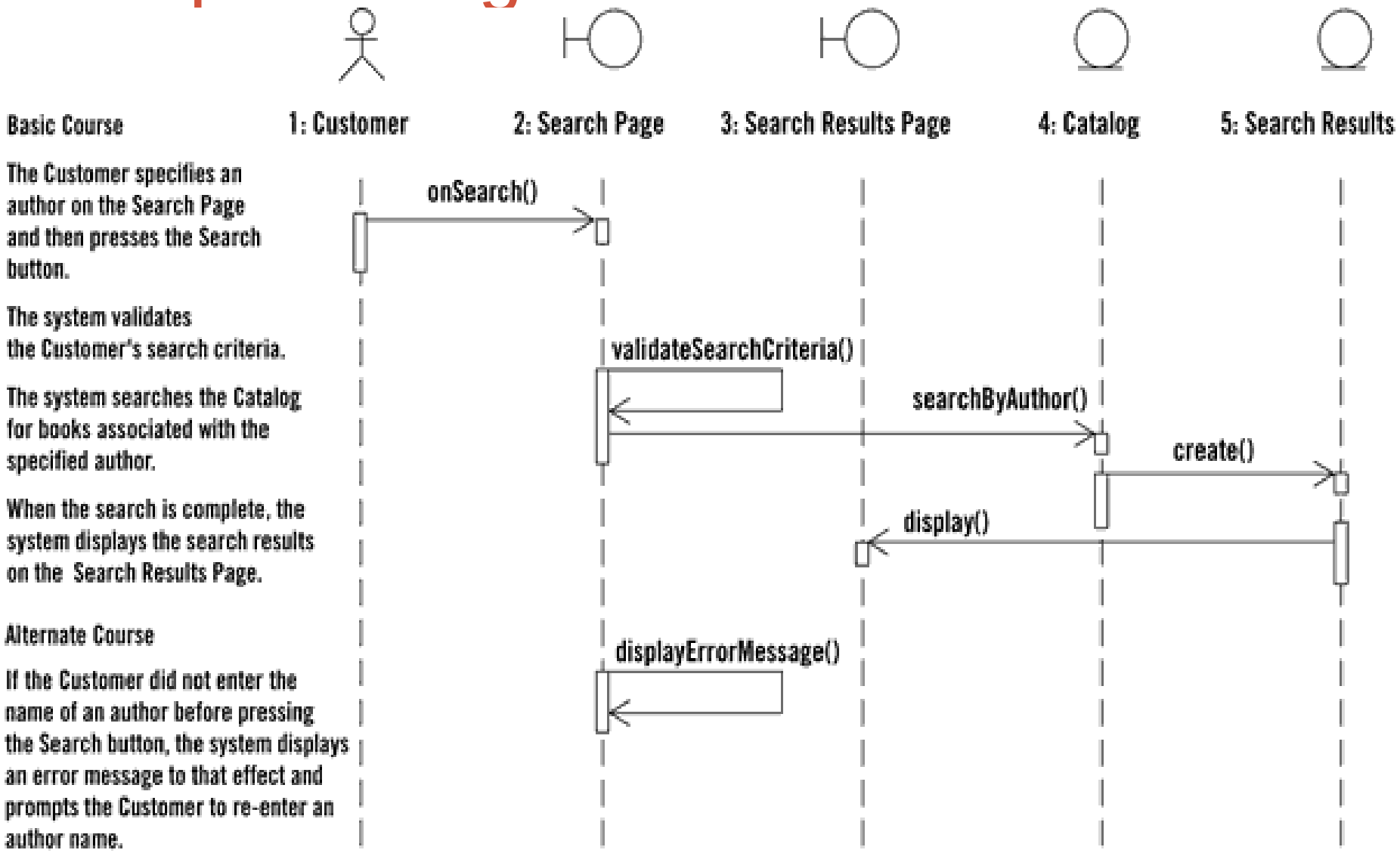
Lecture 17: UML Sequence Diagram

Shereen Fouad

Sequence Diagrams

- Show the interactions between objects with an emphasis on time ordering
- Show time sequences that are not easily depicted in other diagrams
- Document the dynamics in an object-oriented system.
- Describe the flow of messages, events, actions between objects
- Show concurrent processes and activations

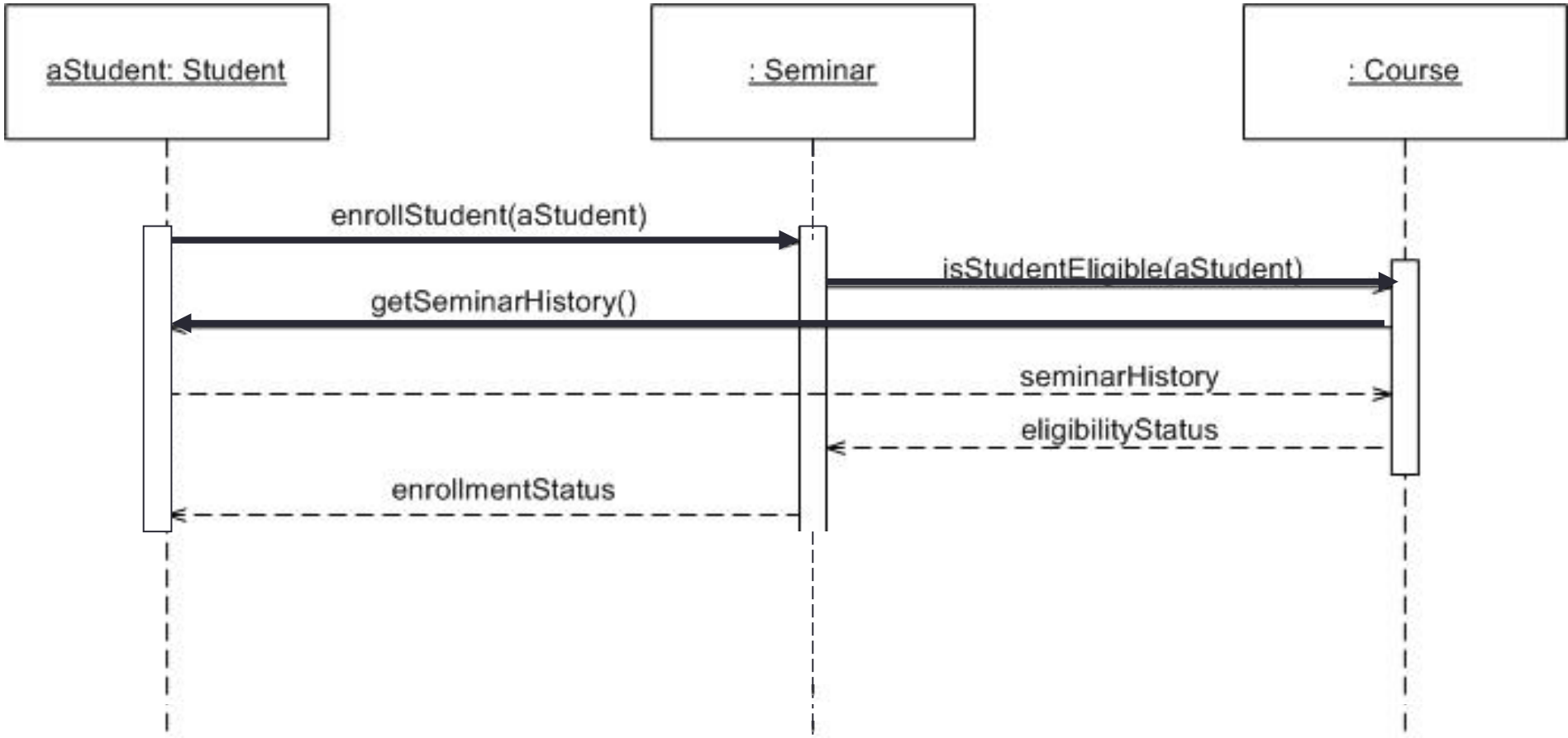
Sequence dg. from use case



Components a sequence diag.

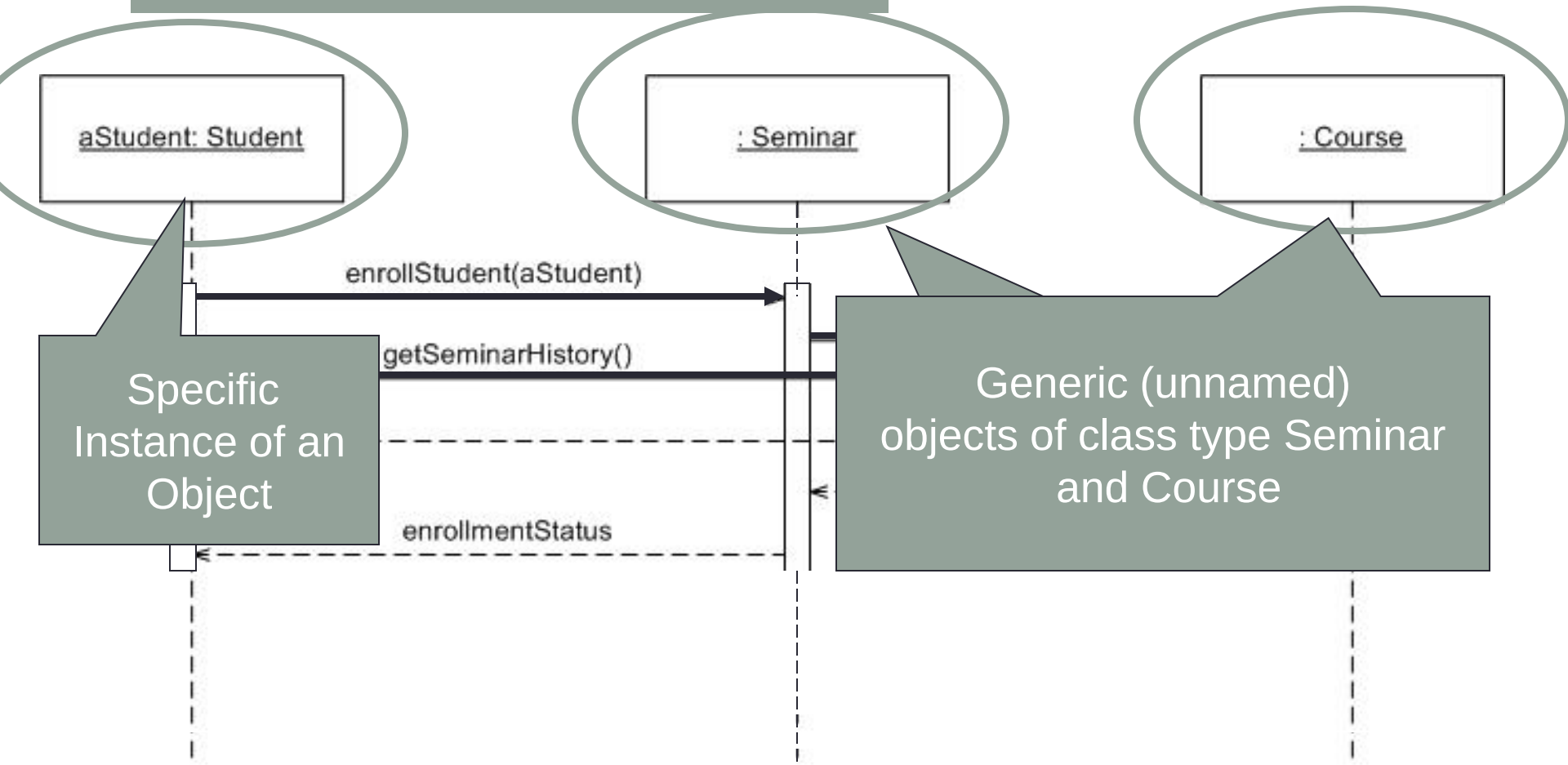
- **participant**: an object or entity that acts in the sequence diagram
- **message**: communication between participant objects
- the **axes** in a sequence diagram:
  - horizontal: which object/participant is acting
  - vertical: time (down -> forward in time)

Sequence Diagram

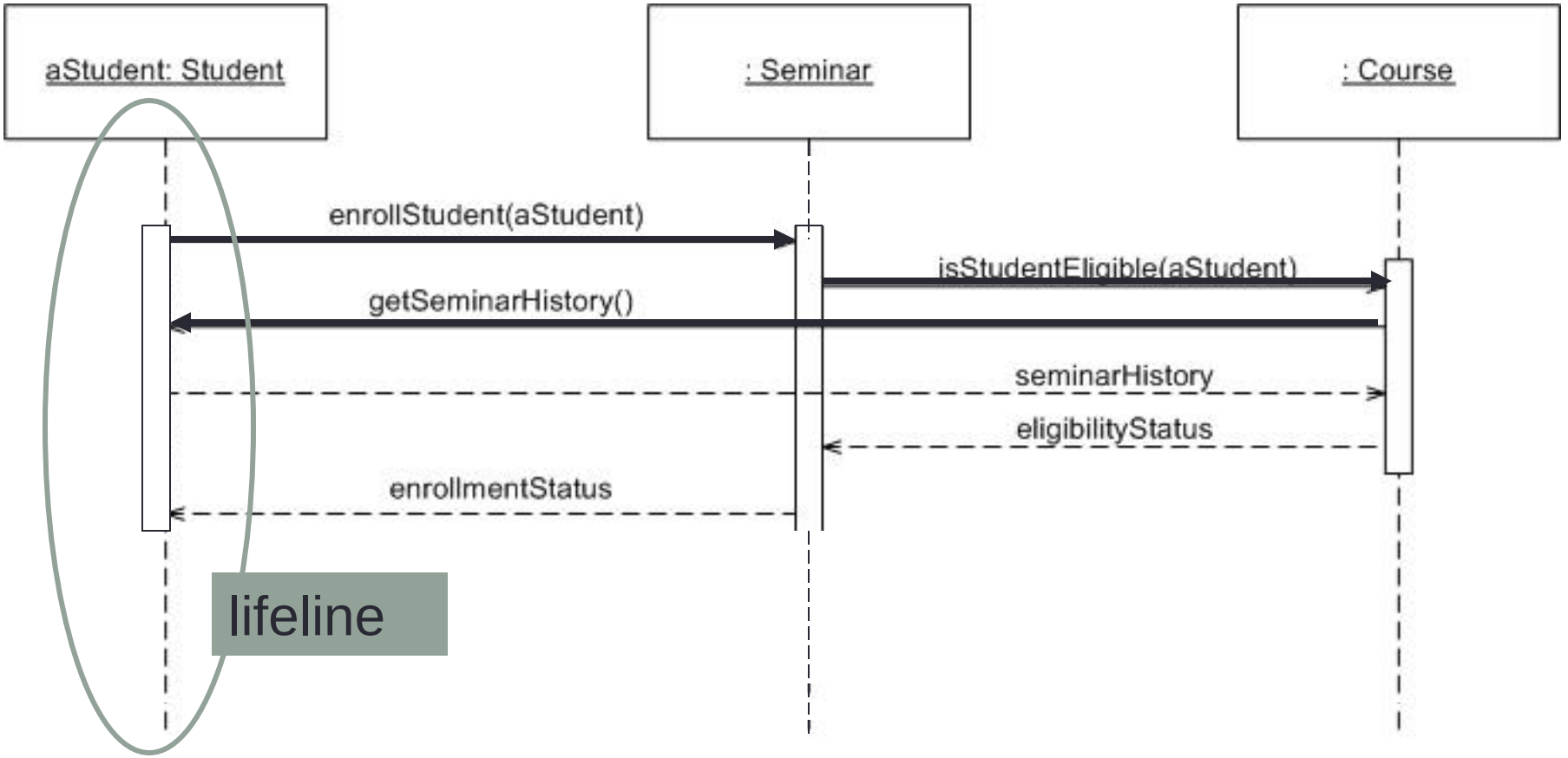


Components: Objects

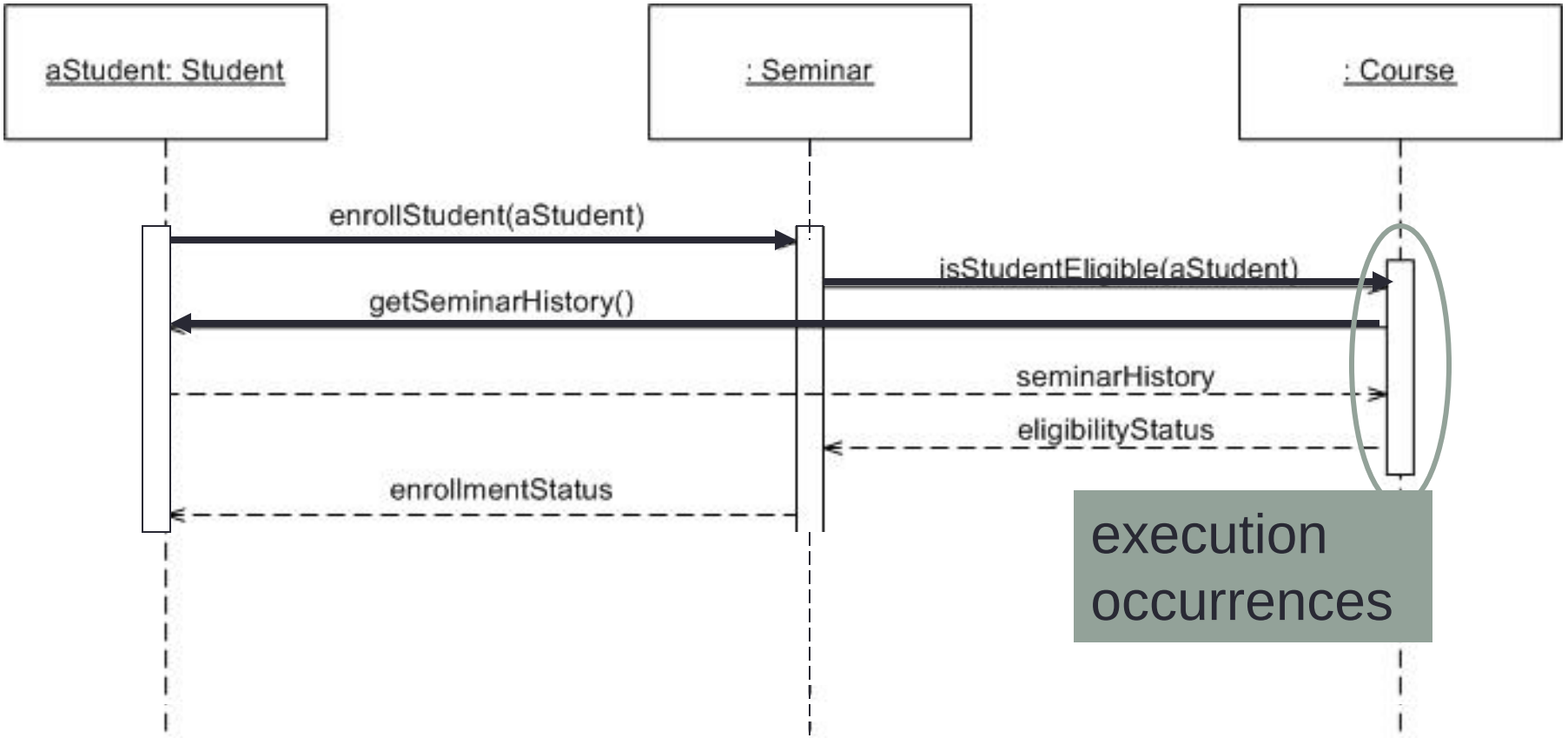
Objects: aStudent is a specific instance of the Student class



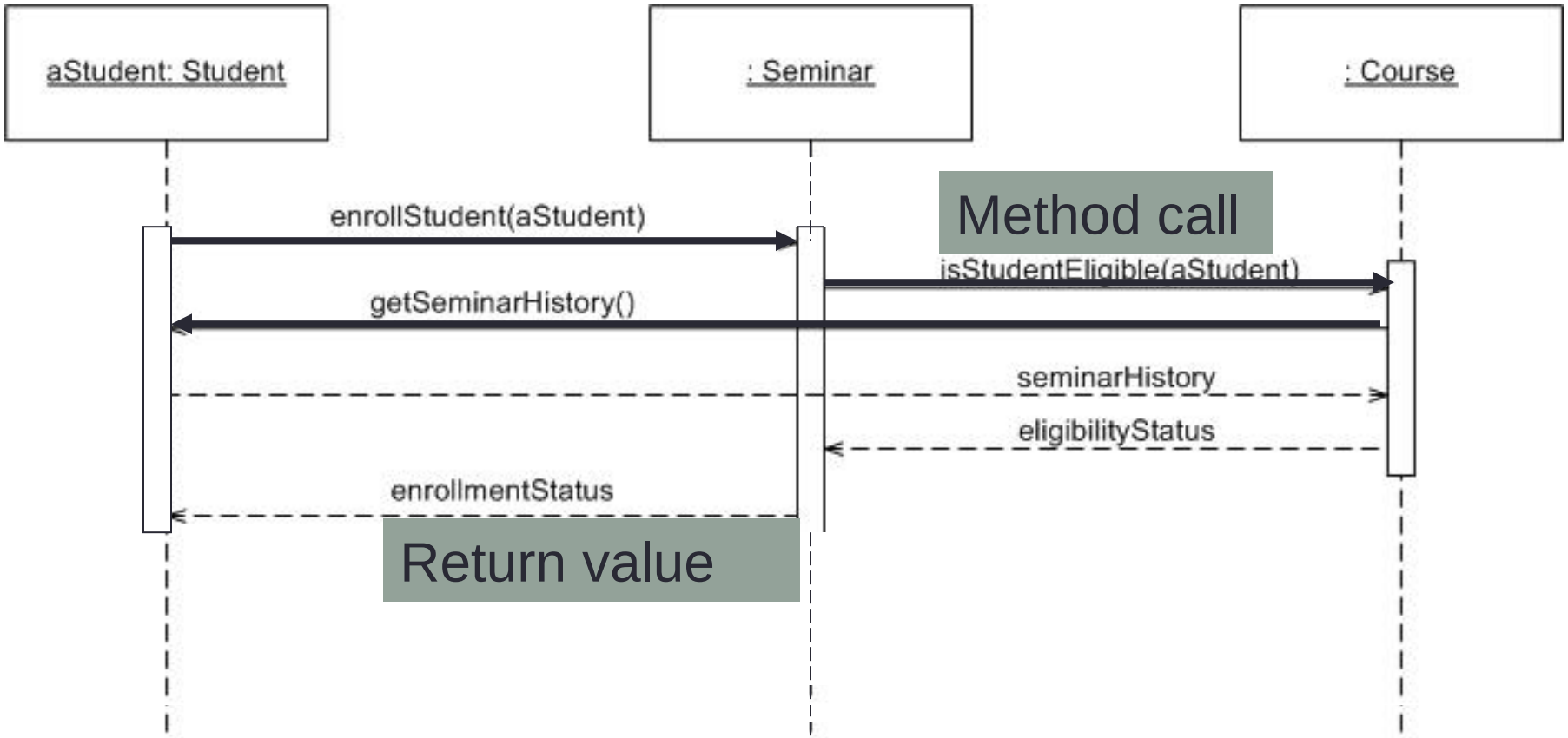
Components: Lifeline



Components: Execution



Components: Messages

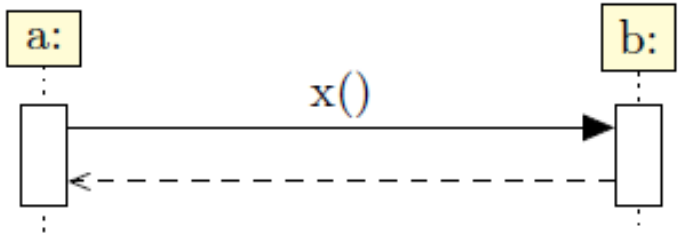


the interaction starts near the top of the diagram and ends at the bottom

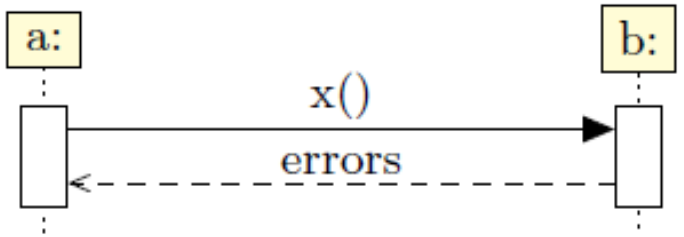
if ActivityA happens before ActivityB, ActivityA must be above activity A

Synchronous: A synchronous message between active objects indicates wait semantics; the sender waits for the message to be handled before it continues. This typically shows a method call.

Reply: This shows the return message from another message.

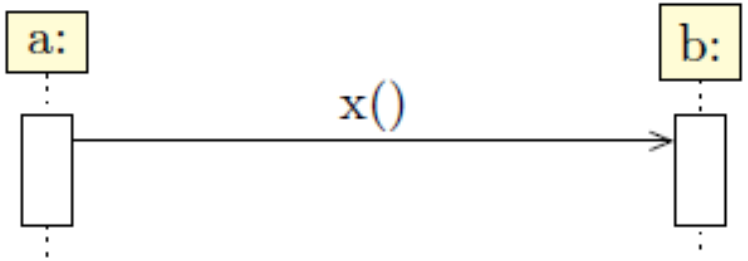


The contents of the return value can be indicated on the return message arrow:



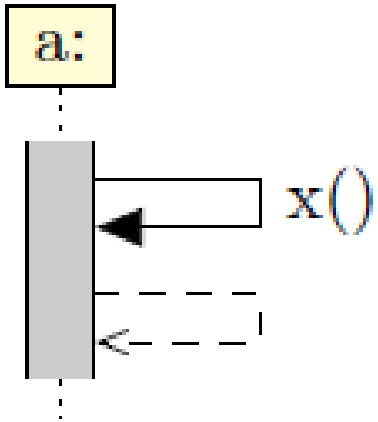
Asynchronous: With an asynchronous flow of control, there is no explicit return message to the caller. An asynchronous message between objects indicates no-wait semantics; the sender does not wait for the message before it continues. This allows objects to execute concurrently.

(a "send-and-forget" message)



Recursive message

Method calls can, of course, be from an object to itself.



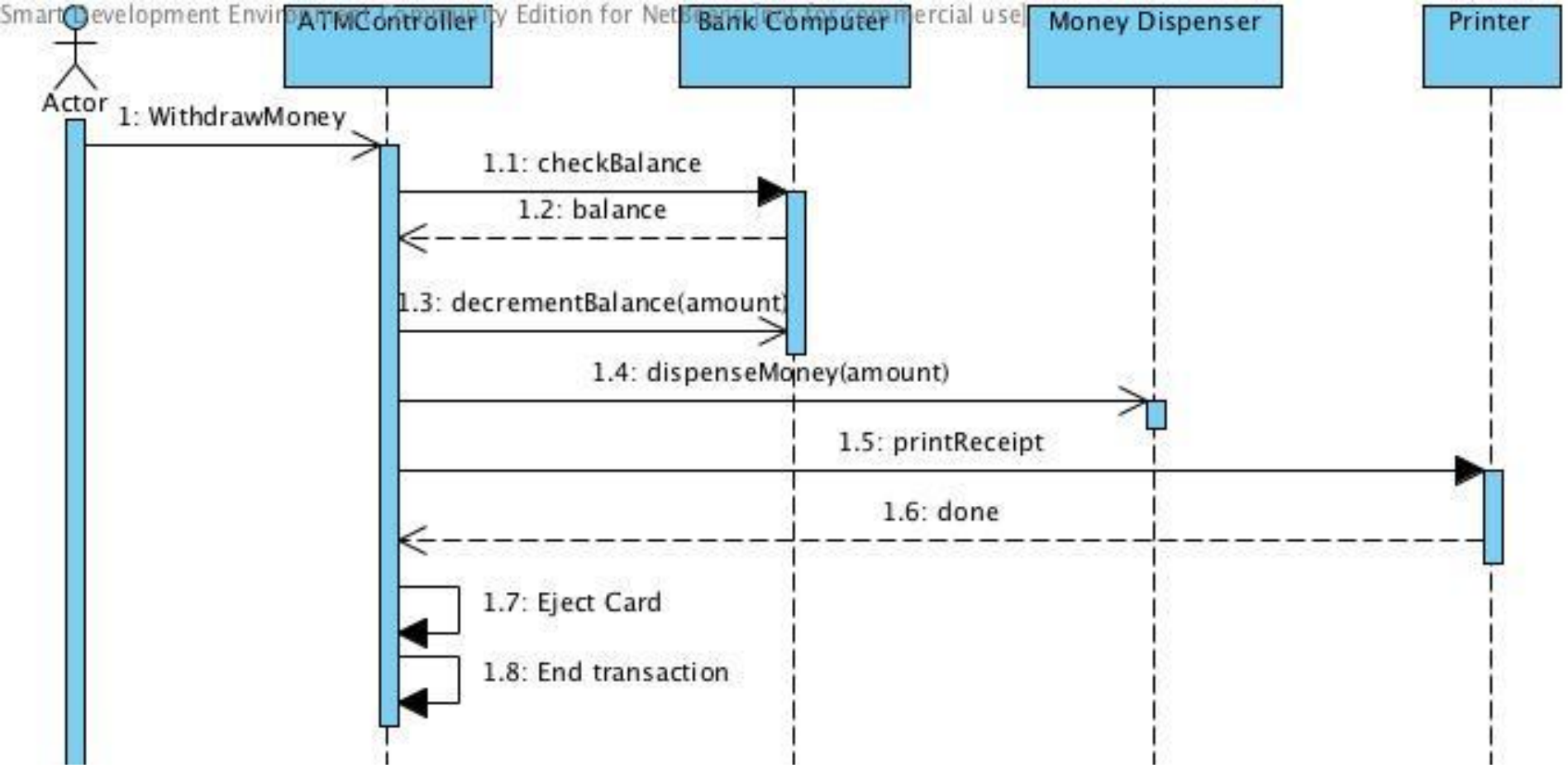
In class Exercise

Draw a sequence diagram for withdrawing money from an ATM machine

Treat each part of the ATM as a class

ATM Controller
Bank Database
Money dispenser
Printer

ATM Machine Exercise



Quiz !

---------- is used in sequence diagram to indicate that some method of the corresponding object is active

a) Life line

b) Execution occurrences

c) Recursive message

d) Life time

What are the heuristics which sequencing diagram follows?

a) Put pairs of lifelines that interact heavily next to one another

b) Position lifelines to make message arrows as short as possible

c) Position lifelines to make message arrows go from left to right

d) All of the mentioned

What is true about a Sequence Diagram? [2 answers]

a) It describes the behavior in many Use Cases.

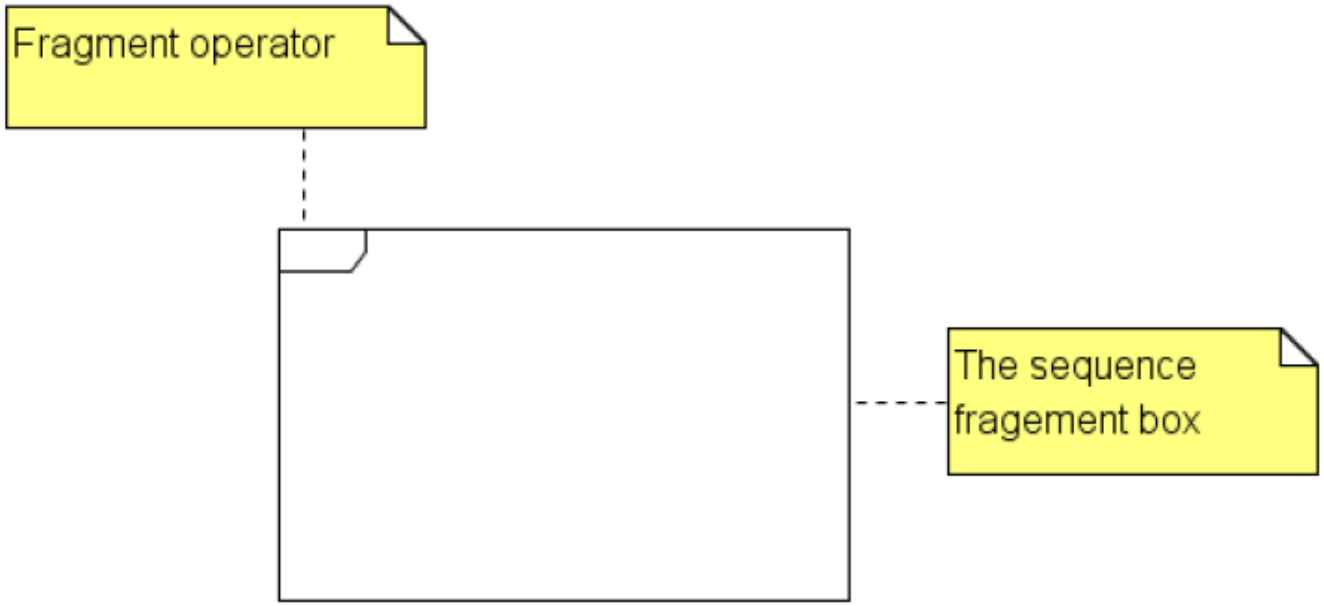
b) It describes the behavior in a single Use Case.

c) It describes the behavior of a single object.

d) It describes the behavior of several objects

Combined fragment

- frame: box around part of a sequence diagram to indicate selection or loop
  - `if` -> (opt) [condition]
  - `if/else` -> (alt) [condition], separated by horizontal dashed line
  - loop -> (loop) [condition or items to loop over]

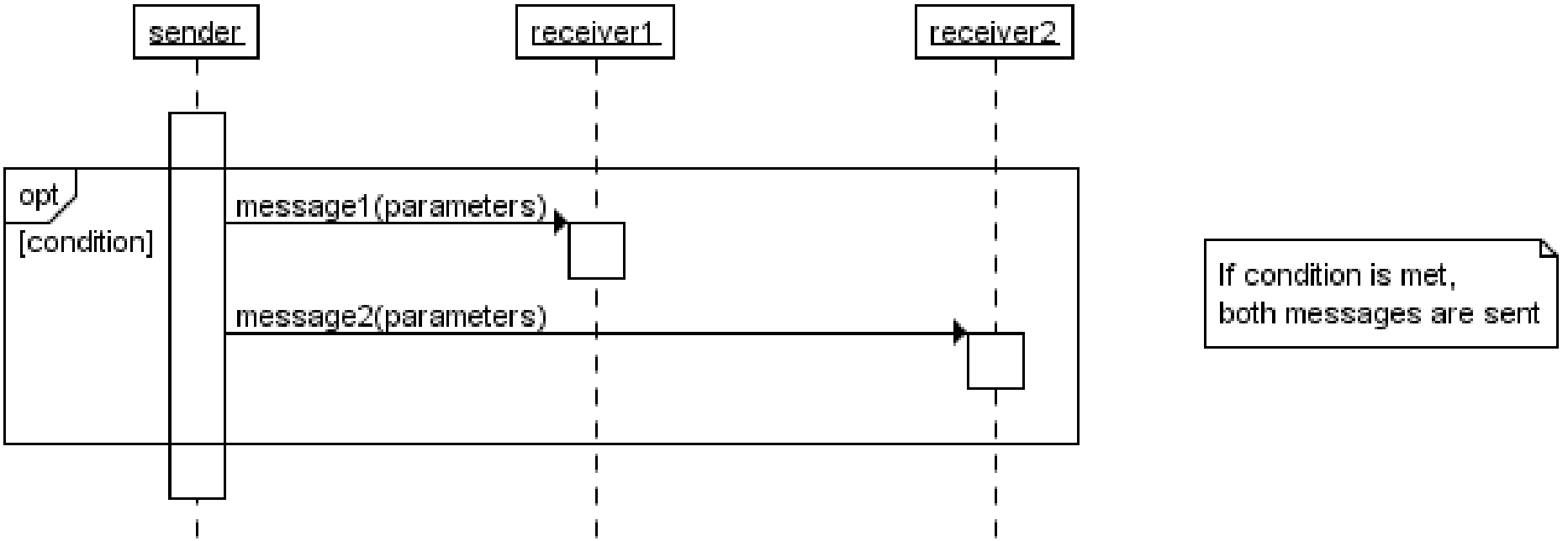


Combined fragment

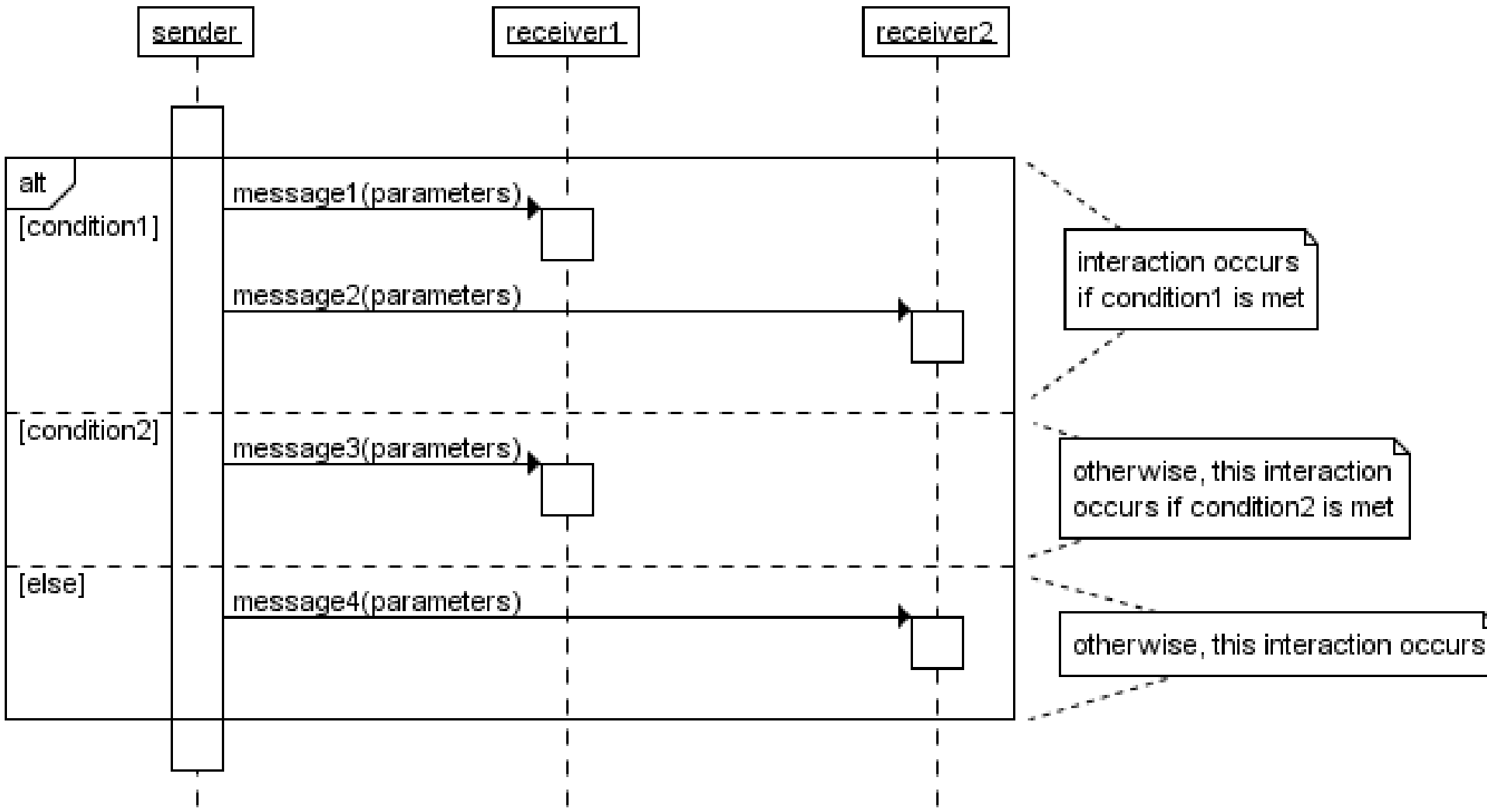
Operator	Meaning
alt	**Alternative multiple fragments:** only the one whose condition is true will execute.
opt	**Optional:** the fragment executes only if the supplied condition is true. Equivalent to an alt only with one trace.
par	**Parallel:** each fragment is run in parallel.
loop	**Loop:** the fragment may execute multiple times, and the guard indicates the basis of iteration.

Conditional interaction

If the condition indicated in the top left corner is satisfied then the block is executed. There is no else part.

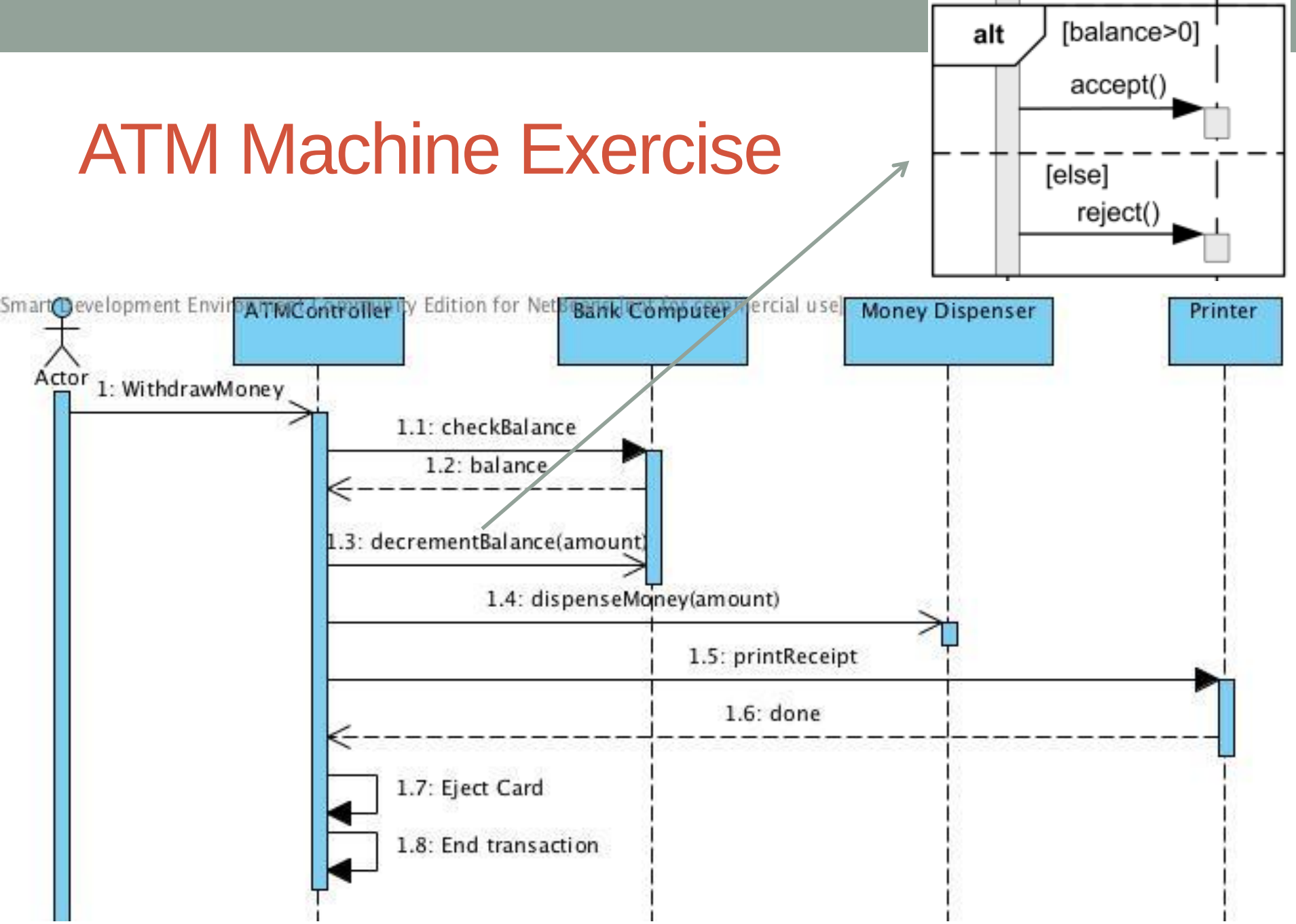


A guarded message or 'opt' combined fragment is somewhat similar to the if-construct in a programming language.

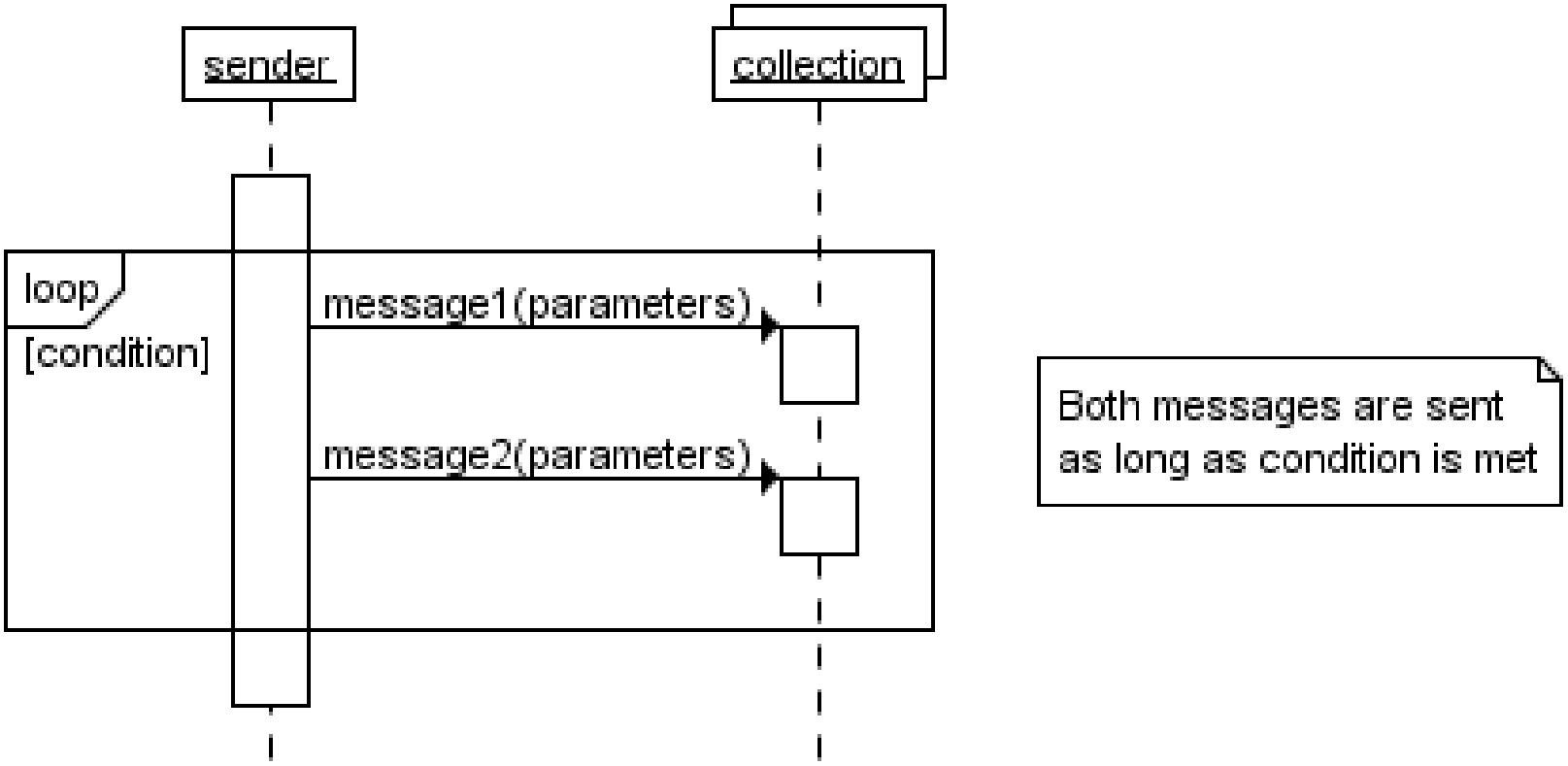


An 'alt' combined fragment is similar to nested if-then-else and switch/case constructs in programming languages.

ATM Machine Exercise



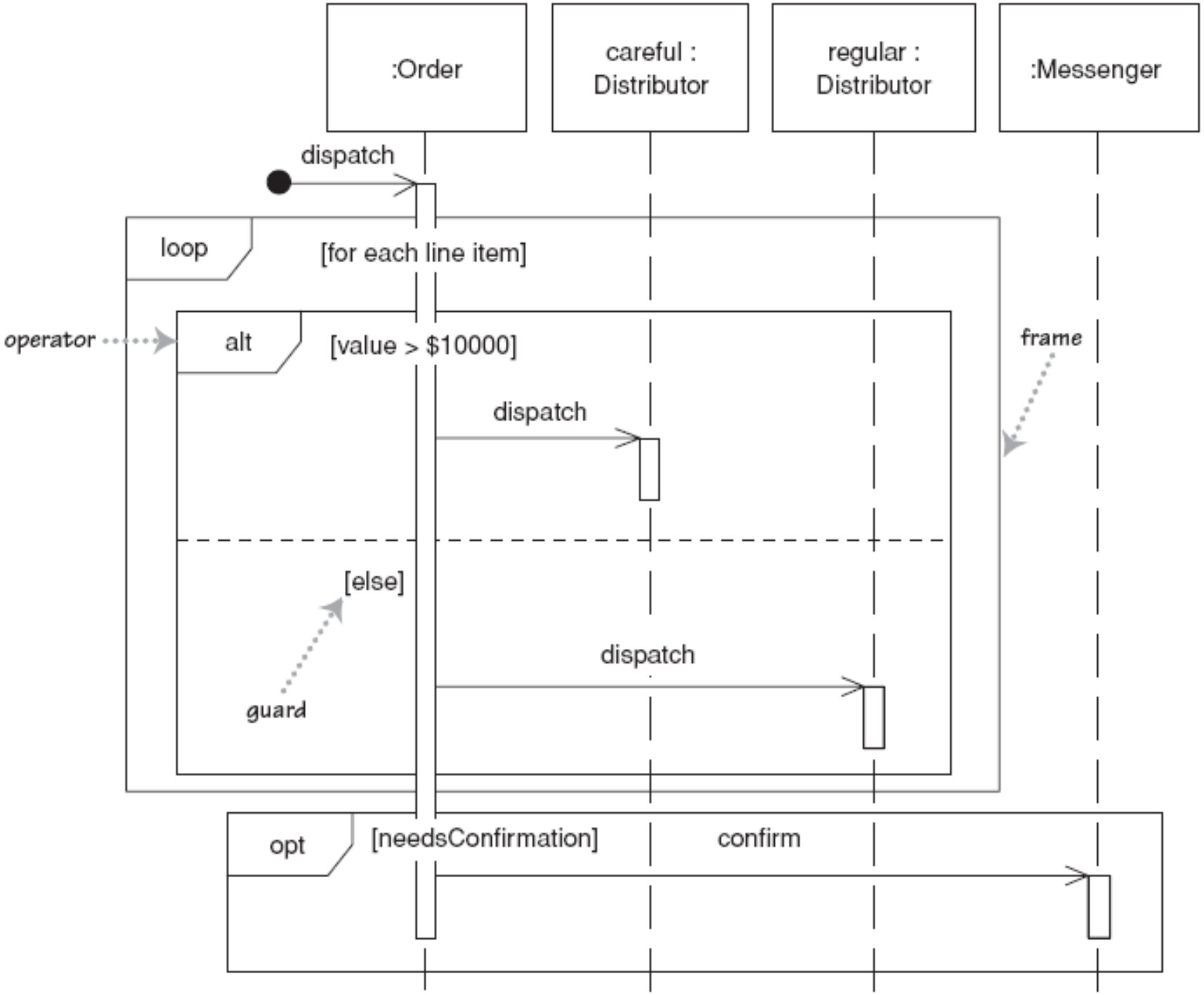
Iterative interaction



While the condition in the top left corner is satisfied, the block is repeated.

In class Exercise

- Draw a sequence diagram for dispatching items in an order.
- *For each item in the order do*
  - *if the value of item > 10000$*
  - *then*
  - *dispatch item through a careful distributer*
  - *else*
  - *dispatch item through a regular distributer*
- Treat the following parts as a separate class
    - order
    - careful distributer
    - regular distributer


:Order
careful : Distributor
regular : Distributor
:Messenger
dispatch
loop
[for each line item]
operator
alt
[value > $10000]
frame
dispatch
[else]
guard
dispatch
opt
[needsConfirmation]
confirm

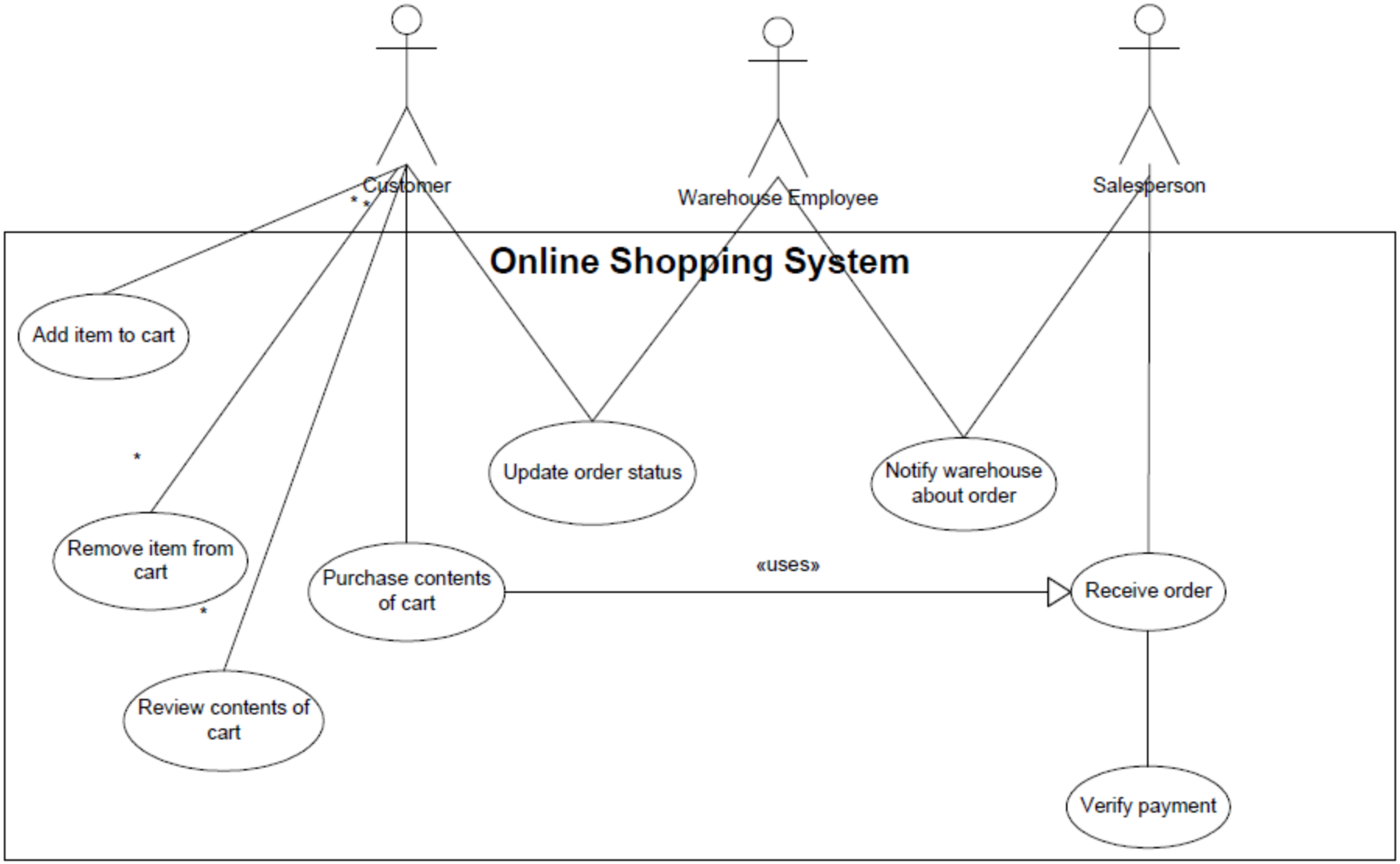
Exercise

- Draw a sequence diagram for booking a movie ticket online

- Treat the following parts as a separate class
    - Admin
    - Registered user
    - Unregistered user
    - Movies
    - Book ticket
    - Payment

Exercise

Develop a sequence diagram showing the interactions involved in the below use case



References

- A number of slides in this talk is based on:
  - Alan P. Sexton hand-outs (Introduction to Software Engineering. The University of Birmingham. Spring Semester 2014)
  - http://www.tracemodeler.com/articles/a_quick_introduction_to_uml_sequence_diagrams/
- For More diagrams check:
  - http://www.ibm.com/developerworks/rational/library/3101.html

Thank YOU ☺