COURSE NAME

OBJECT ORIENTED ANALYSIS AND DESIGN

CSC 2210

(UNDERGRADUATE)



# CHAPTER 4

# SEQUENCE DIAGRAM

VICTOR STANY ROZARIO
ASSISTANT PROFESSOR
DEPARTMENT OF COMPUTER SCIENCE, AIUB
Web: https://cs.aiub.edu/profile/stany

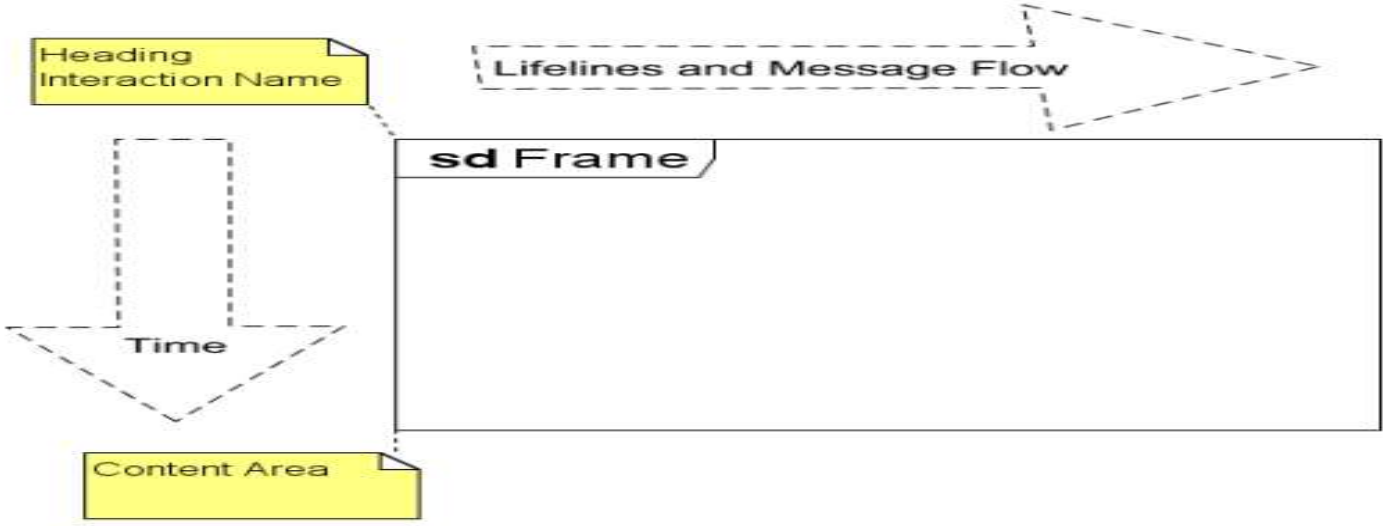
RG Google Scholar LinkedIn

# SEQUENCE DIAGRAM

- Sequence Diagrams are **interaction diagrams** that detail how operations are carried out and models:
  - a single scenario executing in the system
  - communication behavior of individuals exchanging information to accomplish some task.
- *Sequence diagram*—shows interacting individuals along the top and message exchange down the page.
- Relation of UML diagrams to other exercises:
  - CRC cards -> Class diagram
  - Use cases -> Sequence diagrams

# KEY PARTS OF SEQUENCE DIAGRAM

- **Frame:** a rectangle with a pentagon in the upper left-hand corner called the name compartment.
  - **sd** - interaction Identifier is either a simple name or an operation specification as in a class diagram



# KEY PARTS OF SEQUENCE DIAGRAM

- Participant: an object or entity (actor, system) that acts in the sequence diagram (rectangle identifier)
- Lifelines/Axes: Participating individuals are arrayed across the diagram as *lifelines*. The dashed line shows the period when an individual exists.
  - **Horizontal:** which object/participant is interacting
  - **Vertical:** time (down -> forward in time)
- Message: communication between participant objects.

# MESSAGES AND MESSAGE ARROWS

- *Synchronous*—The sender suspends execution until the message is complete (sender waits for reply)
- *Asynchronous*—The sender continues execution after sending the message (no waiting time)
- *Message return to the sender / instance, object creation*

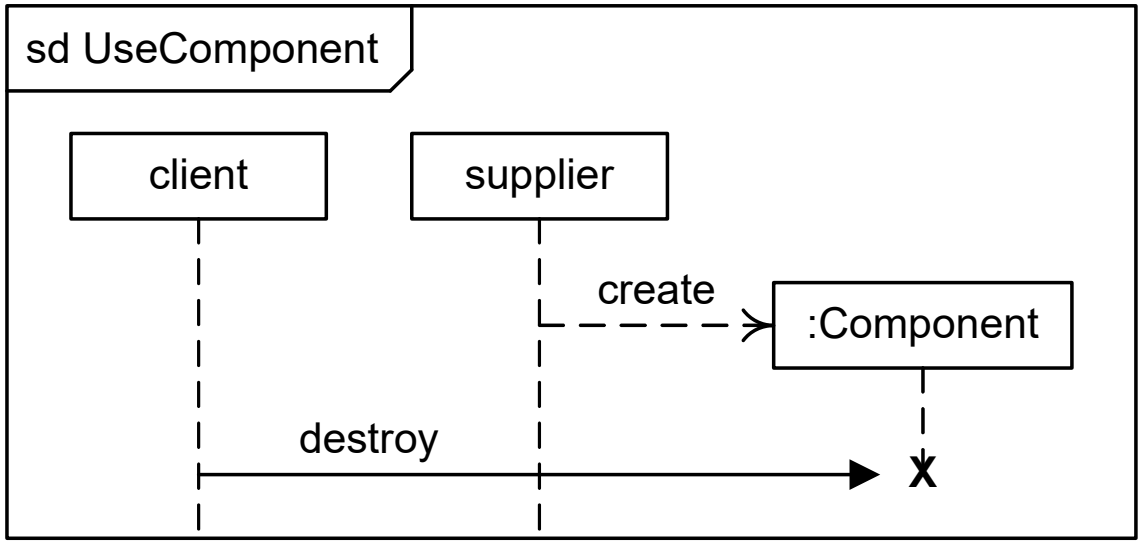
# LIFELINES

- *Participating individuals are arrayed across the diagram as lifelines:*
  - *Rectangle containing an identifier*
  - *Dashed line extending down the page*
- *The vertical dimension represents time; the dashed line shows the period when an individual exists.*

# LIFELINE CREATION AND DESTRUCTION

- *An new object appears at the point it is created.*
  - *Not clear from UML specification*
- *A destroyed object has a truncated lifeline ending in an X.*
- *Persisting objects have lifelines that run the length of the diagram.*

# LIFELINES EXAMPLE



# LIFELINE IDENTIFIER FORMAT

*name*[ *selector* ] : *typeName*

- *name*—simple name or "self"; optional
- selector—expression picking out an individual from a collection
  - Format not specified in UML
  - Optional; if omitted, so are the brackets
- *typeName*—Type of the individual
  - Format not specified in UML
  - Optional; if omitted, so is the colon
- Either *name*, *typeName*, or both must appear

# LIFELINE IDENTIFIER EXAMPLES

*name*[ *selector* ] : *typeName*

- player[i] : Player
- player[i]
- : Player
- board

# SEQUENCE DIAGRAM HEURISTICS

- Put the sender of the first message leftmost.
- Put pairs of individuals that interact heavily next to one another.
- Position individuals to make message arrows as short as possible.
- Position individuals to make message arrows go from left to right.

# SEQUENCE DIAGRAM ELEMENTS

**The reference numbers on the figure denotes:**

1. **Object lifeline**
2. **Message/Stimulus**
3. **Iteration**
4. **Self-reference**
5. **Return**
6. **Anonymous object**
7. **Object name**
8. **Sequence number**
9. **Condition**
10. **Basic comment**

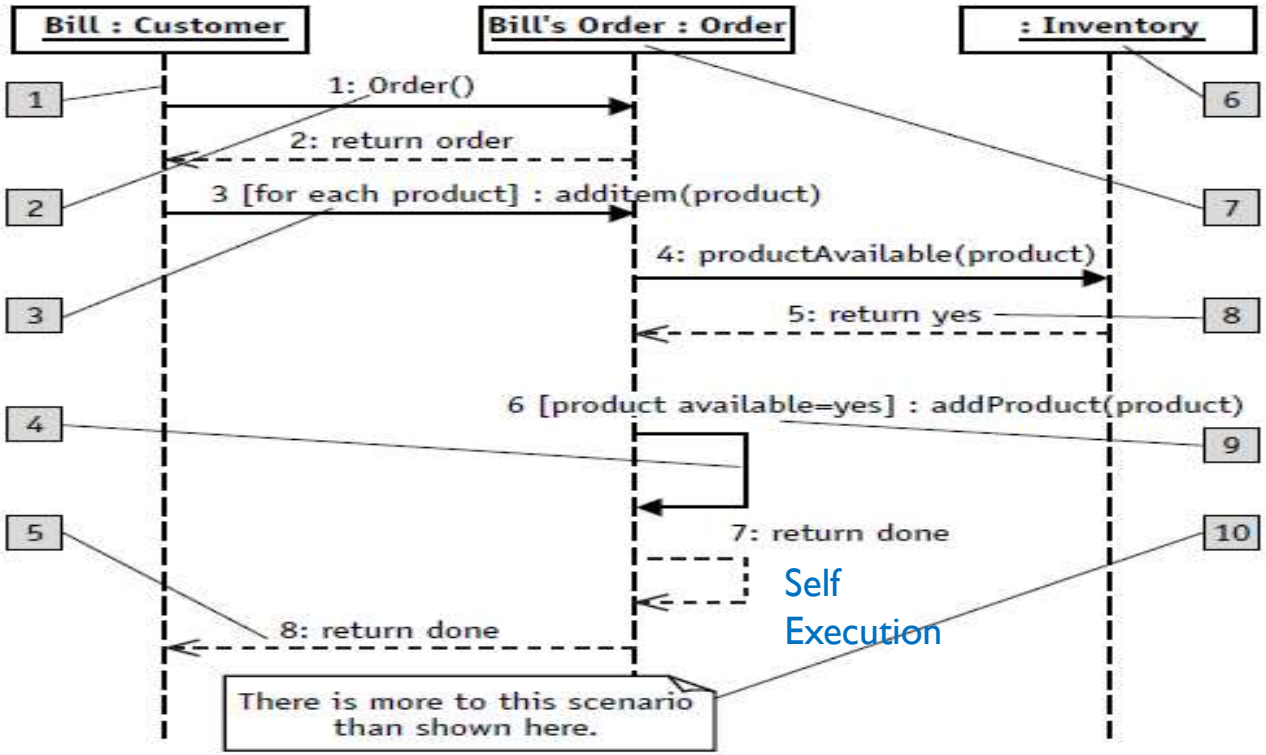



*Figure 16-2 Elements of the Sequence diagram notation*

# ELEMENTS (EXTENDED)

1. Activation:
   - The start of the vertical rectangle, the activation bar
2. Deactivation:
   - The end of the vertical rectangle, the activation bar
3. Timeout event:
   - Typically signified by a full arrowhead with a small clock face or circle on the line
4. Asynchronous event:
   - Typically signified by a stick arrowhead
5. Object termination symbolized by an X

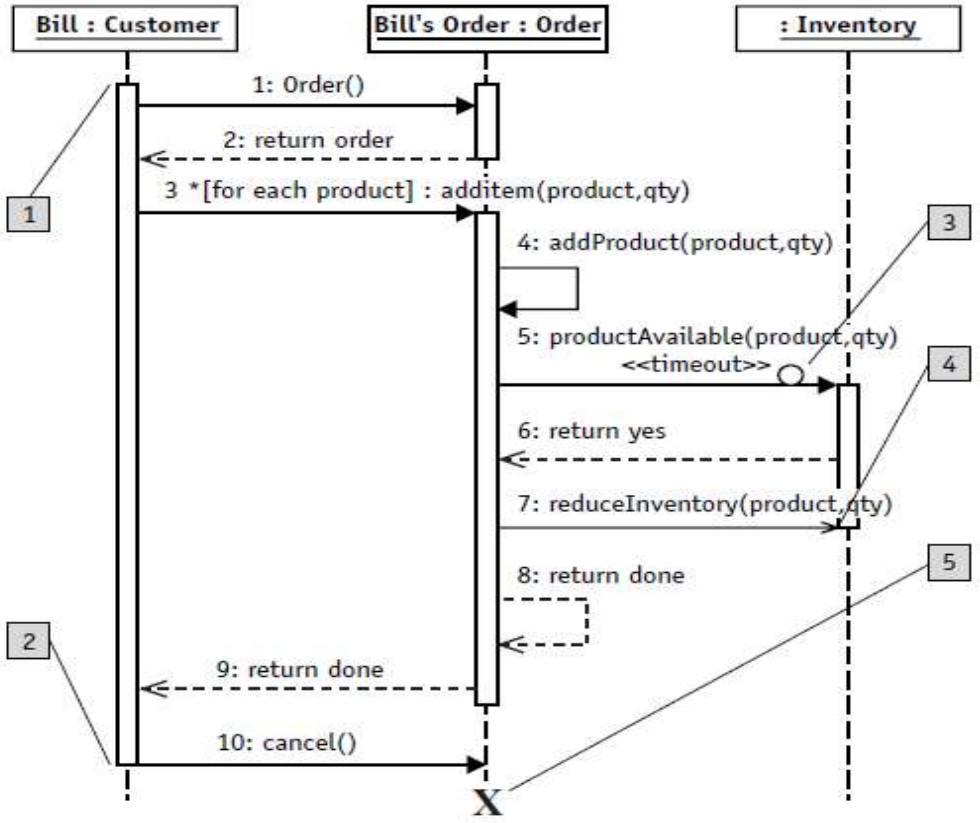
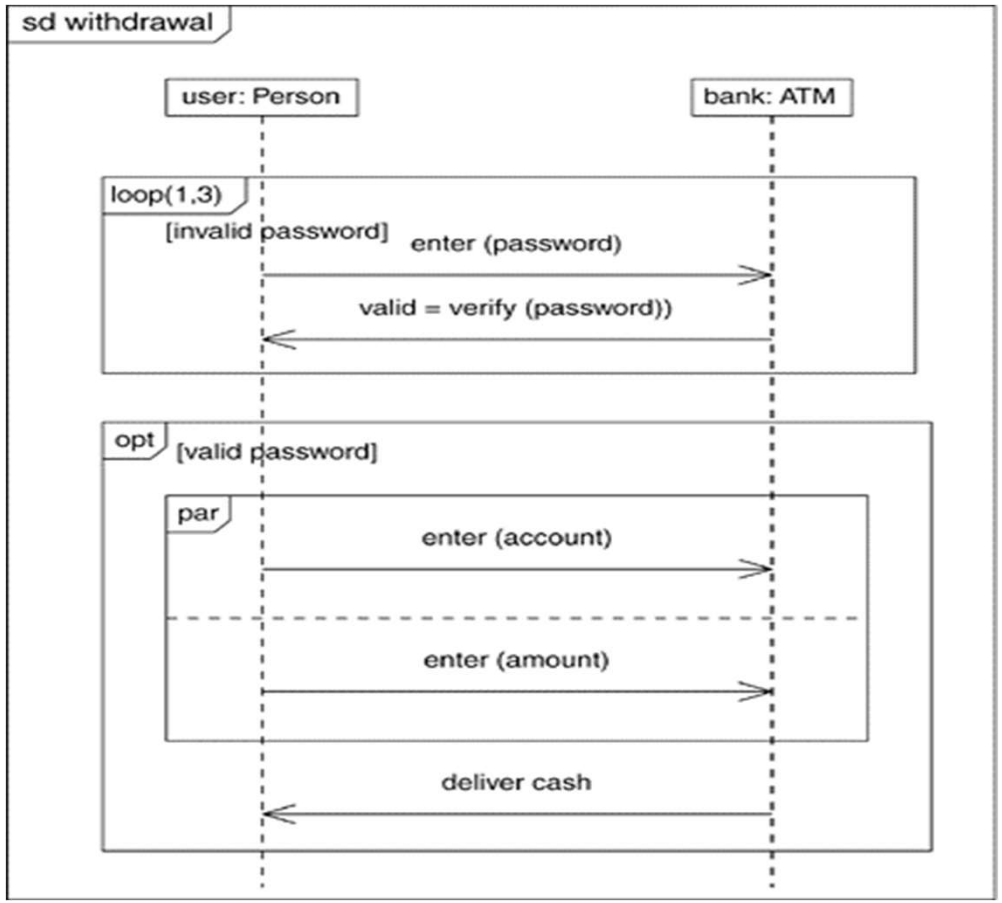



*Figure 16-3 Extended elements of the Sequence diagram notation*

# SEQUENCE DIAGRAM 2.0

- **Structured Control Operators**
  - **'OPT': Optional**
  - **'ALT': Alternate**
  - **'PAR': Parallel**
  - **'Loop': Iterative**
  - **'Ref': Reference**
  - ***There are many operators, but these are most common***

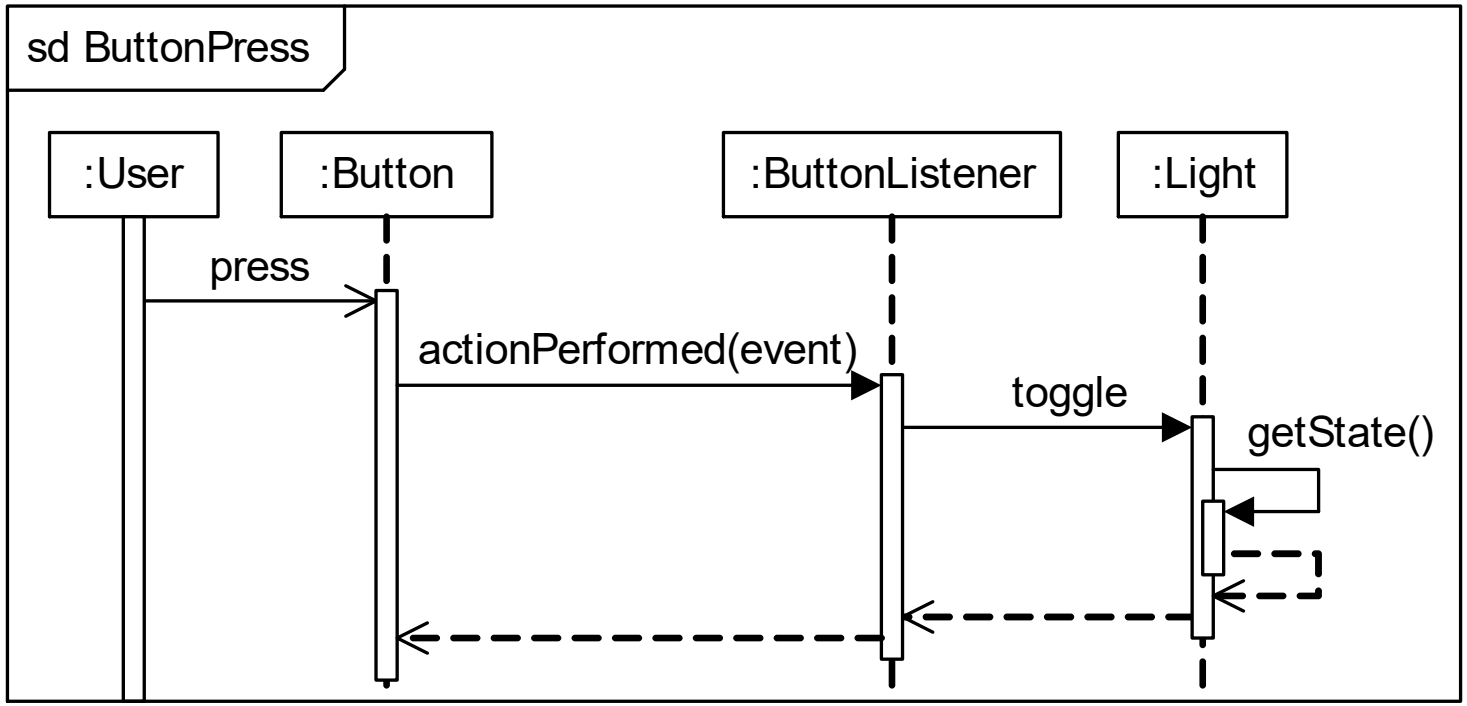
# SEQUENCE DIAGRAM WITH STRUCTURED CONTROLS



# EXECUTION OCCURRENCE

- An operation is **executing** when some process is running its code.
- An operation is **suspended** when it sends a synchronous message and is waiting for it to return.
- An operation is **active** when it is executing or suspended. The period when an object is active can be shown using an ***execution occurrence*** (Thin white or grey rectangle over lifeline dashed line)

# EXECUTION OCCURRENCE EXAMPLE



# CASE STUDIES

**Case 1**

In a withdrawal transaction of an ATM Machine system the customer inserts his ATM card in the machine. The machine then verifies the customer authentication using the information provided in customer account. After successful verification the machine takes withdrawal request from the customer and checks whether the request is valid or not. A valid request is carried out by the machine. In case of unsuccessful verification the customer request is denied.

# CASE STUDIES

**Case 2**

In a library management system of a university a member can place a request to book a journal to the librarian. Before the librarian can complete the booking the member has to be verified of his status whether he is allowed to borrow journals or not. The journal then has to be located whether it is in the campus where the request was made or it is in a different campus. If the journal is in a different campus the librarian makes a request for the journal to be sent at the requested campus. The librarian then informs the member about the time required for the journal to reach and completes the booking.

# CASE STUDIES

## Case 3

A doctor includes the instruction in the patient advice when a patient requires a bed or room in the hospital. The advice is then passed to the office clerk. The office clerk checks the present booking database to get the available room and bed list. Facilities of the rooms and the beds are then extracted from the room list. Available room and bed numbers and facilities are sent to the patient. The patient chooses a room or bed and then the office clerk writes patientid, room or bed no, doctorid and doctor advice in the booking database. Finally a copy of the admission is sent to the associated nurse and a copy is given to the patient

# CASE STUDIES

## Case 4

When an air cooler starts, the control switch sends signal to the blower fan to start at the level that is already set in the control switch. Then the control switch sends the level of temperature to the sensor. The sensor senses the outside temperature by sending a request to the thermostat and if the outside temperature is higher then the level sent by the control switch, the sensor send signal to the compressor to switch on. When the sensor outside temperature goes down to the set level, the thermostat sends signal to the sensor and the sensor in turn sends signal to the compressor to switch off. The process keeps on running in a cycle as long as the air cooler stays on.

# REFERENCES


- Booch, G., Rumbaugh, J. & Jacobson, I. (2005). *The unified modeling language user guide*. Pearson Education India.