

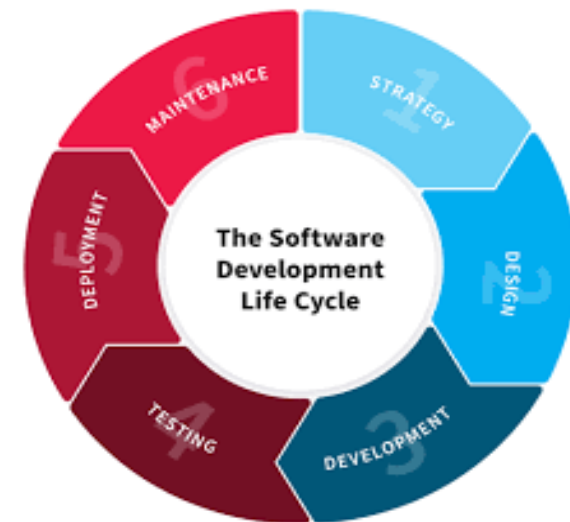


System Analysis Course

Week 05: UML (Sequence diagram)

Ahmed Kord

Rana Khattab





Outline

- ❖ UML
 - ❖ Interaction Diagrams
 - ❖ Sequence diagrams
- ❖ Practical Part – on Software Program



UML (Unified Modeling language)

UNIFIED
MODELING
LANGUAGE





Interaction Diagrams

- ❖ From the name **Interaction** it is clear that the diagram is used to describe some type of interactions among the different elements in the model. So, this interaction is a part of dynamic behavior of the system.
- ❖ This interactive behavior is represented in UML by two diagrams known as **Sequence diagram** and **Collaboration diagram**. The basic purposes of both the diagrams are similar.
- ❖ **Sequence diagram** emphasizes on time sequence of messages and **collaboration diagram** emphasizes on the structural organization of the objects that send and receive messages.



Outline

- ❖ UML
 - ❖ Interaction Diagrams
 - ❖ **Sequence diagrams**
 - ❖ Collaboration diagram
 - ❖ Statechart diagrams
- ❖ Practical Part – on Software Program



Sequence diagrams

- ❖ A **sequence** shows a series of messages exchanged by a selected set of objects in temporally limited situation.
- ❖ Describe the **flow** of messages, events, actions between objects.
- ❖ Show **concurrent** processes and activations.
- ❖ Show **time sequences** that are not easily depicted in **other diagrams**.
- ❖ The system sequence diagram models a scenario of the system interactions with the environment for a **given use case**.
- ❖ Specify the **messages** passed between objects using horizontal arrows including messages to/from external actors.
- ❖ **Time** increases from Top to bottom.



Sequence diagrams Cont.

❖ **X-axis** is objects

- Object that initiates interaction is **left most**.
- Object to the **right** are increasingly more subordinate.

❖ **Y-axis** is time

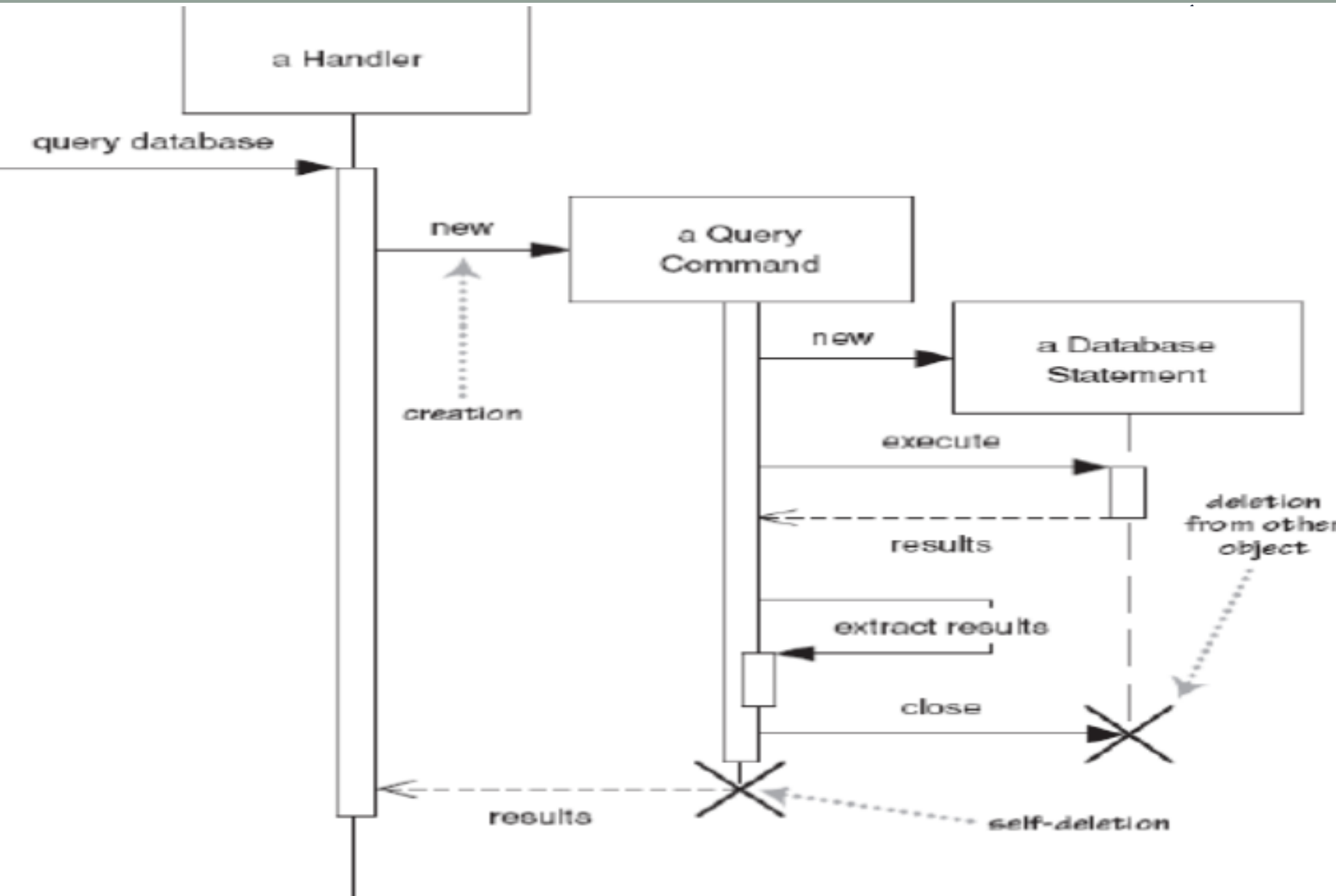
- Messages sent and received are ordered by time.



SYMBOLS

- ❖ **participant**: an object or entity that acts in the sequence diagram.
- ❖ The **Life-Line** represents the object's life during the interaction.
 - Object life-lines represent the existence over a period of time.
 - *deletion*: an X at bottom of object's lifeline.

Sequence diagrams Example 1



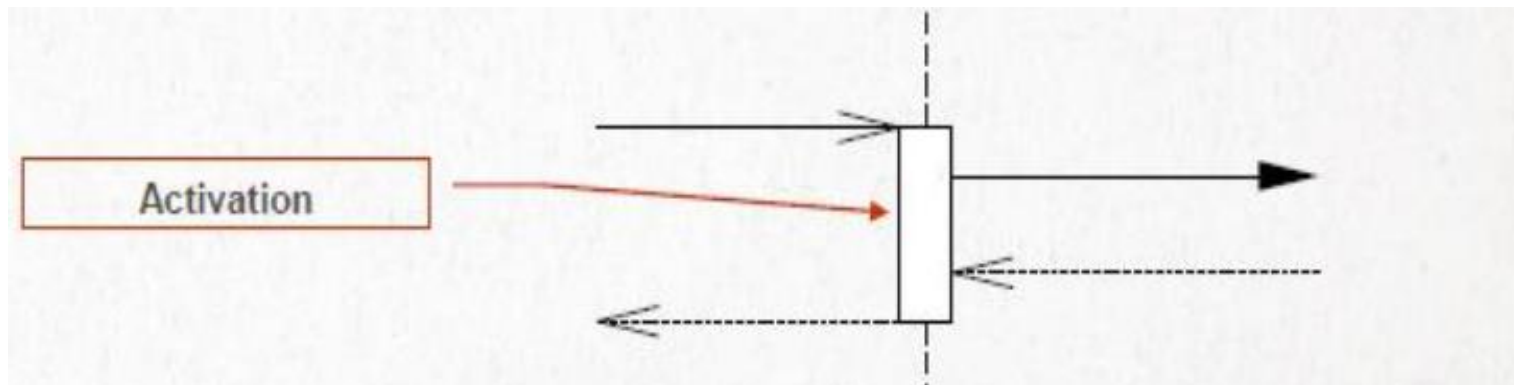


SYMBOLS - Activation

- ❖ **Activation:** box over object's lifeline; drawn when object's method is on the stack.
 - either that object is running its code, or it is on the stack waiting for another object's method to finish.
 - nest to indicate recursion.



SYMBOLS - Activation Cont.





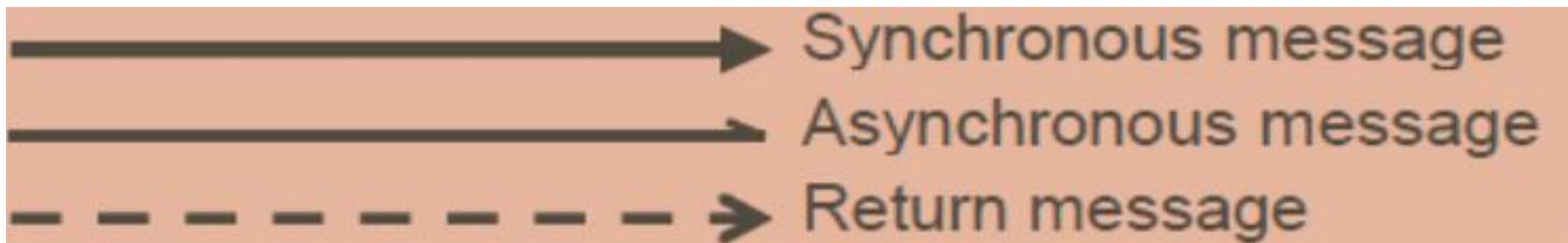
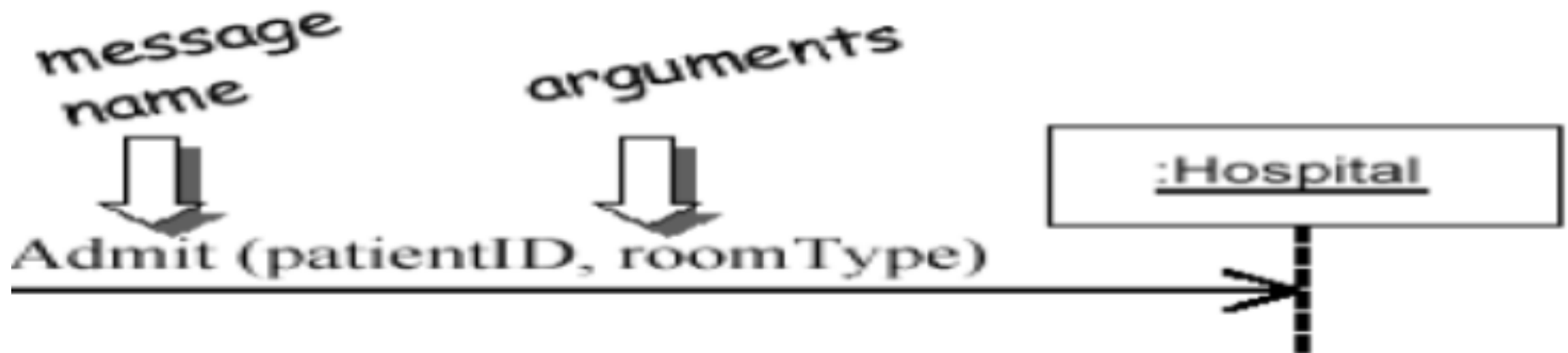
SYMBOLS - Message Passing

❖ MESSAGE PASSING:

- **Message**: communication between participant objects.
- write message name and arguments above arrow.
- *Self calls are also allowed.*
- Arrows:
 - **Synchronous call**(you're waiting for a return value) – triangle arrow-head.
 - **Asynchronous call** (not waiting for a return) – open arrowhead.
 - **Return call** – dashed line.



SYMBOLS - Message Passing Cont.





Call Message

❖ Call Message

- ❖ A message defines a particular communication between Lifelines of an Interaction.
- ❖ Call message is a kind of message that represents an invocation of operation of target lifeline.

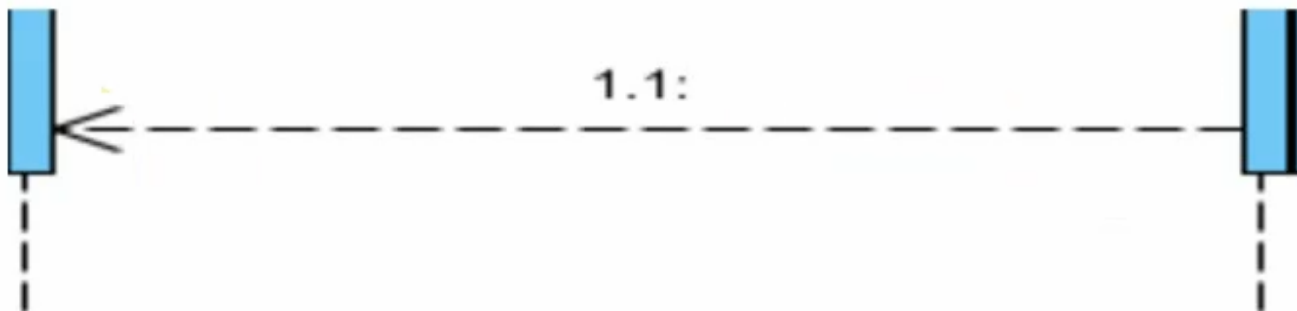




Return Message

❖ return Message

- ❖ Return message is a kind of message that represents the pass of information back to the caller of a corresponded former message.

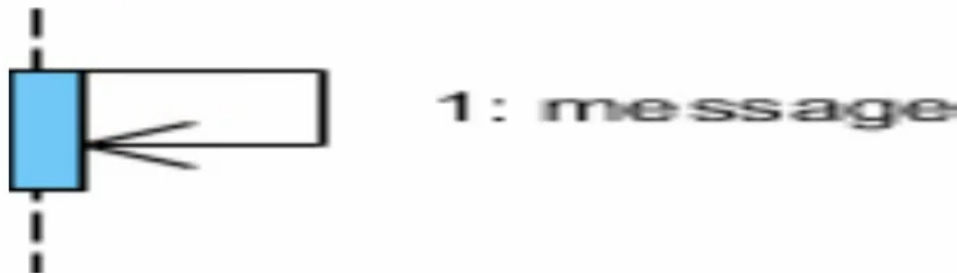




Self Message

❖ Self Message

❖ Self message is a kind of message that represents the invocation of message of the same lifeline.

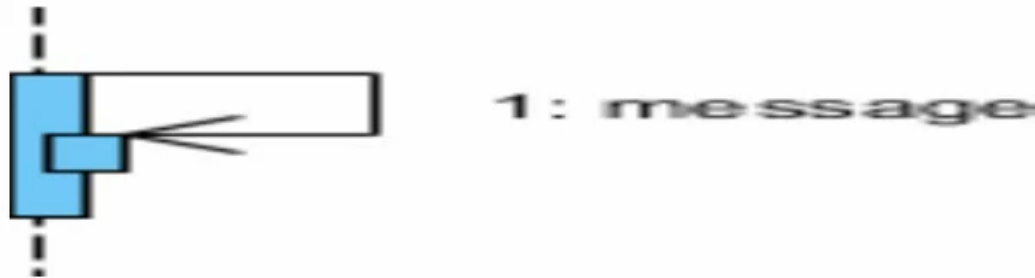




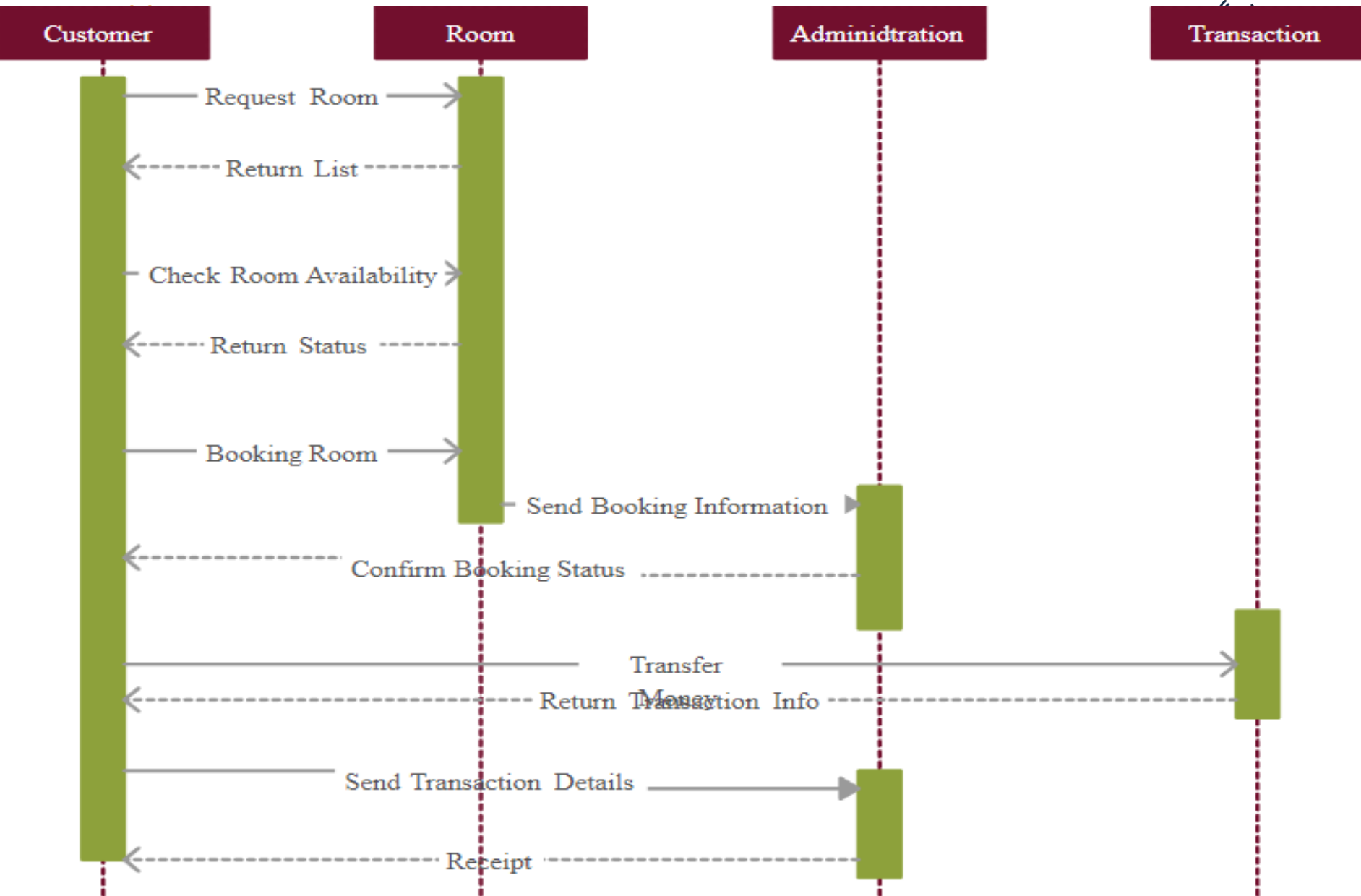
Recursive Message

❖ Recursive Message

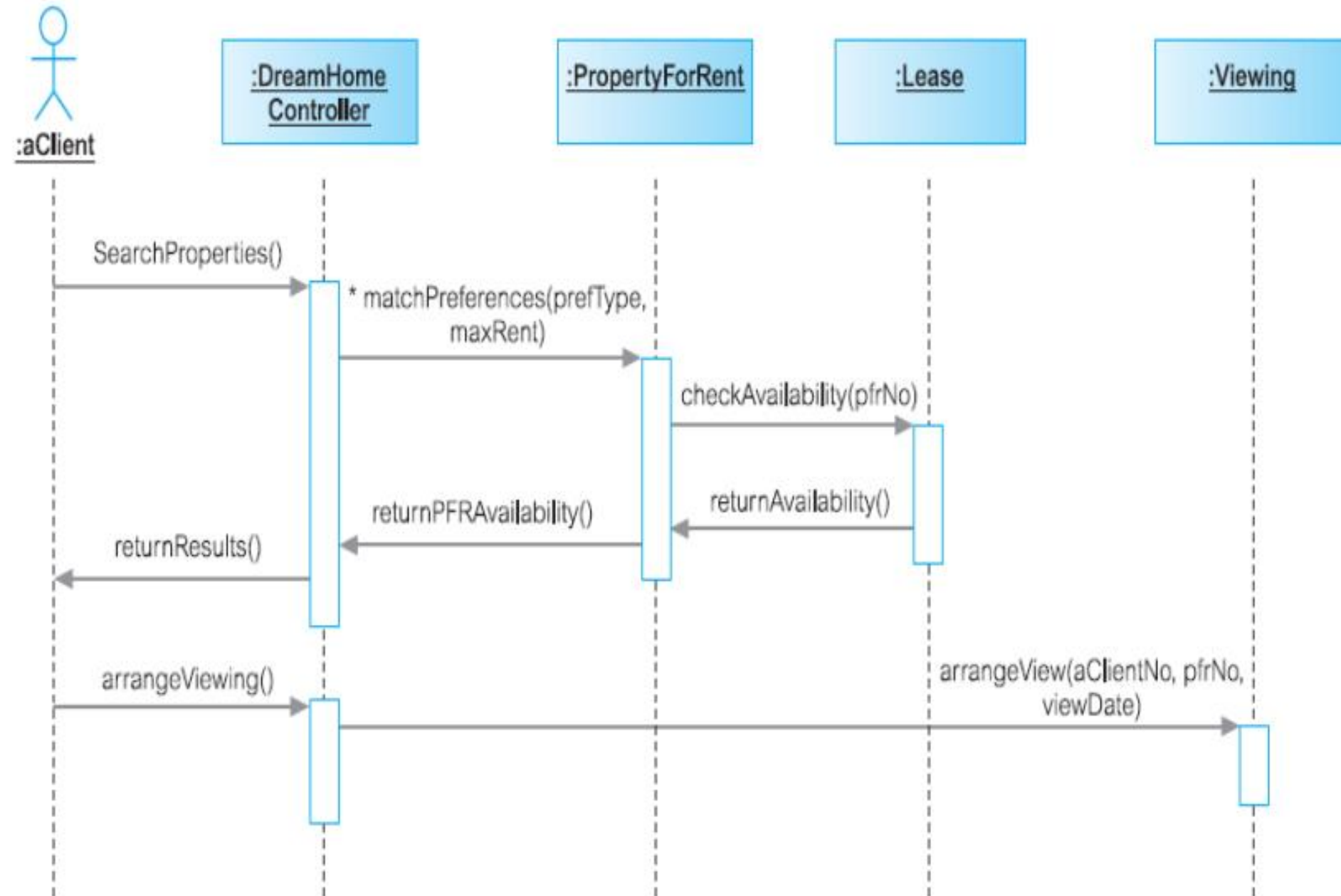
- ❖ Recursive message is a kind of message that represents the invocation of message of the same lifeline. It's target points to an activation on top of the activation where the message was invoked from.



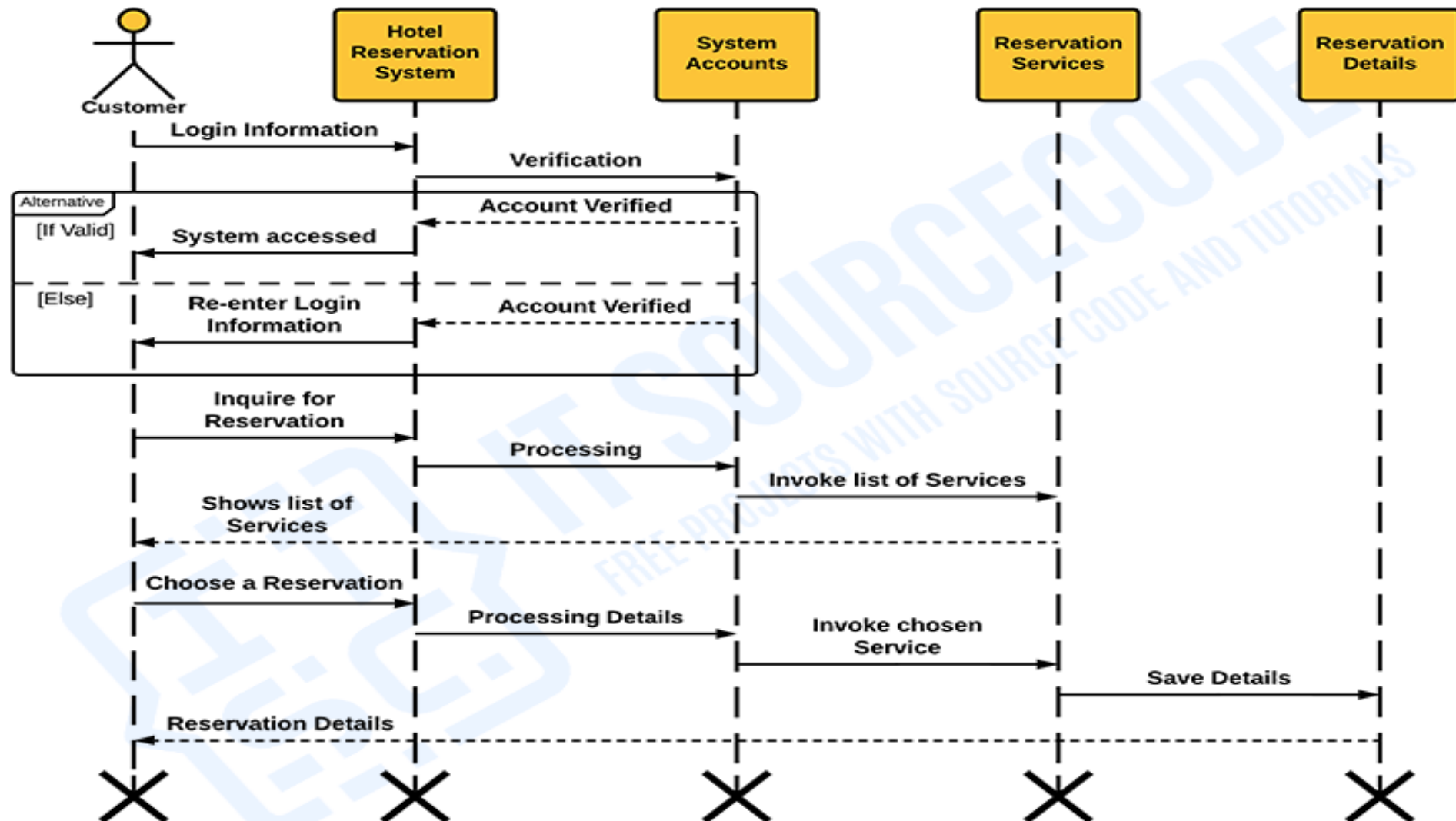
Sequence diagrams Example 2



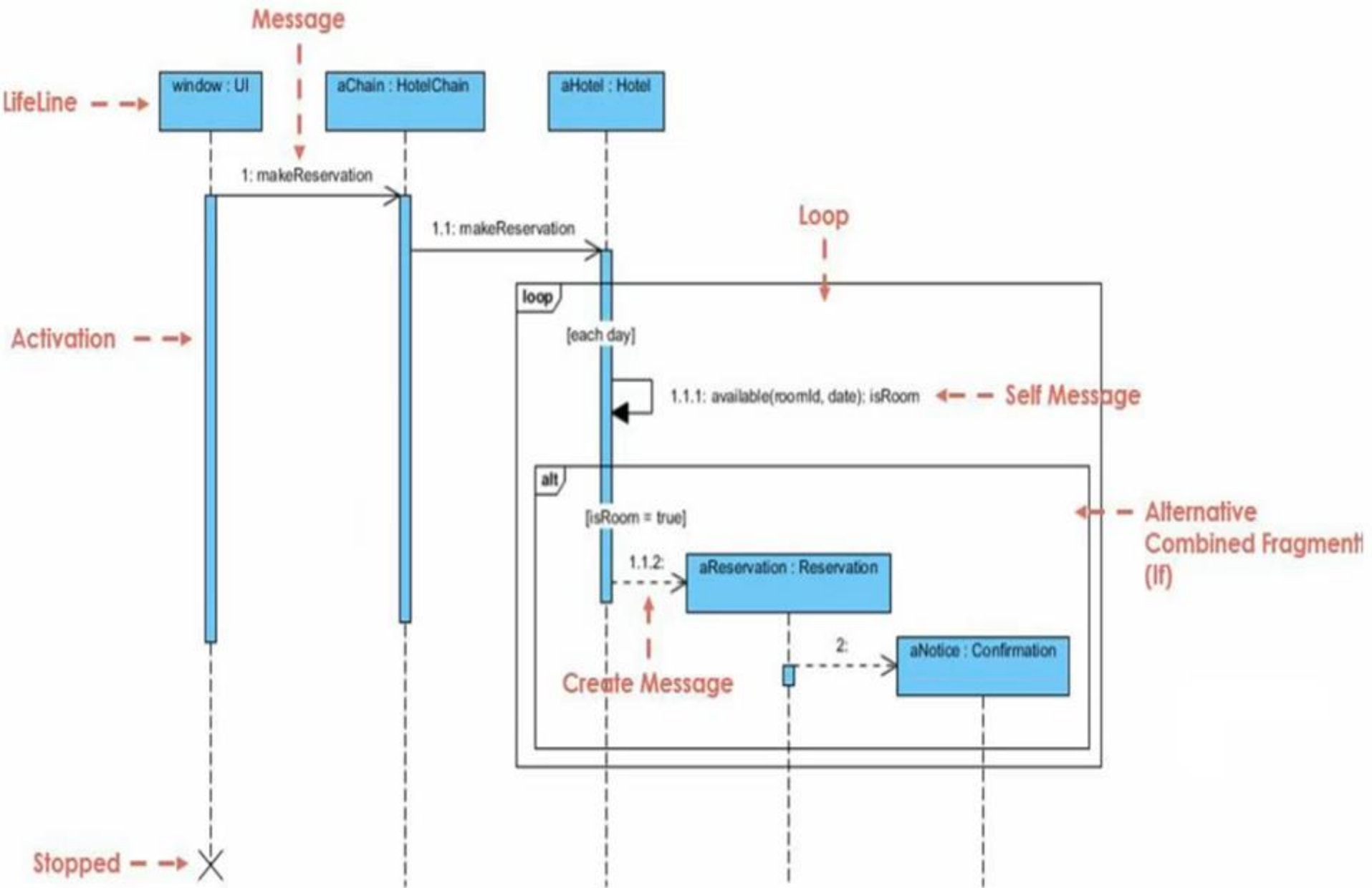
Sequence diagrams Example 3



HOTEL RESERVATION SYSTEM



Sequence diagrams Example 5





Outline

- ❖ UML
 - ❖ Interaction Diagrams
 - ❖ Sequence diagrams
- ❖ Practical Part – on Software Program

Thank You