UML

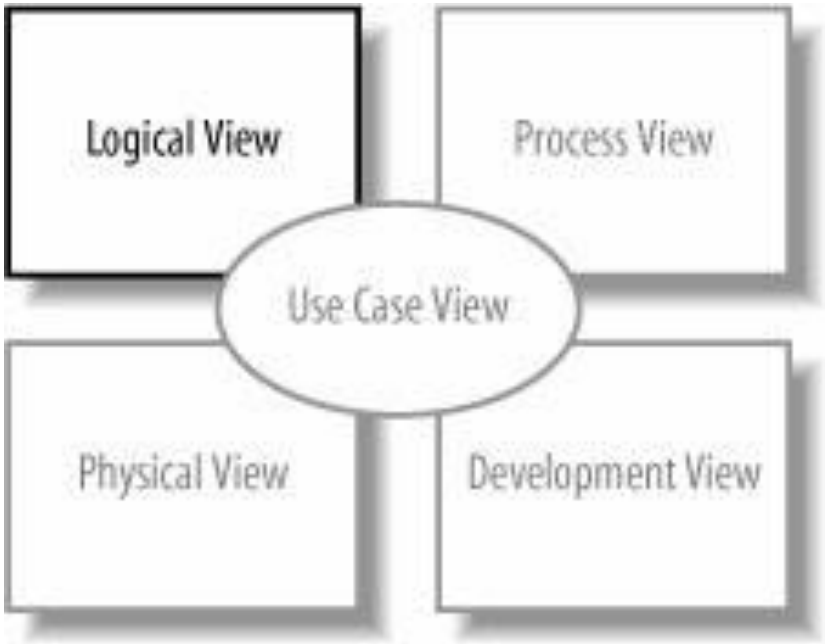
UNIFIED MODELING LANGUAGE

Chapter 6: Sequence Diagram

Table of Content

Modeling Ordered Interaction: Sequence Diagrams

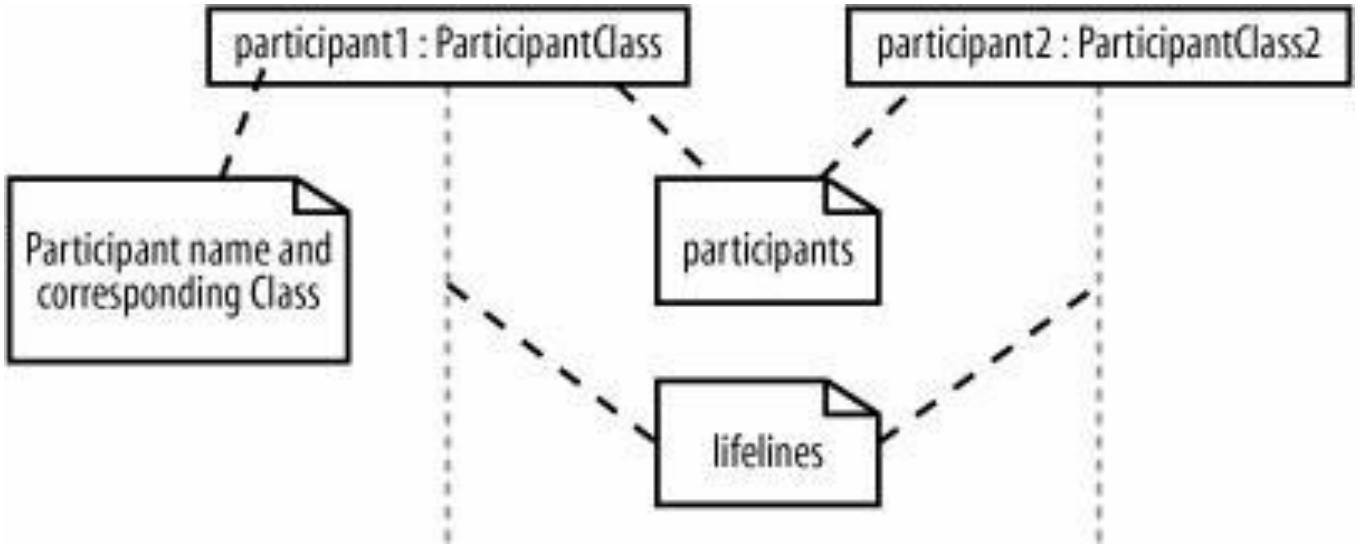
- Sequence diagram model important runtime interactions between the parts that make up your system
- It forms part of the logical view
- Sequence diagrams are all about capturing the ***order*** of interactions between parts of your system



6.1 Participants

- Participants: the parts of your system that interact with each other during the sequence
- Participants are arranged horizontally
- Each participant has a corresponding lifeline running down the page

6.1 Participants



6.1 Participants

- Participant Names

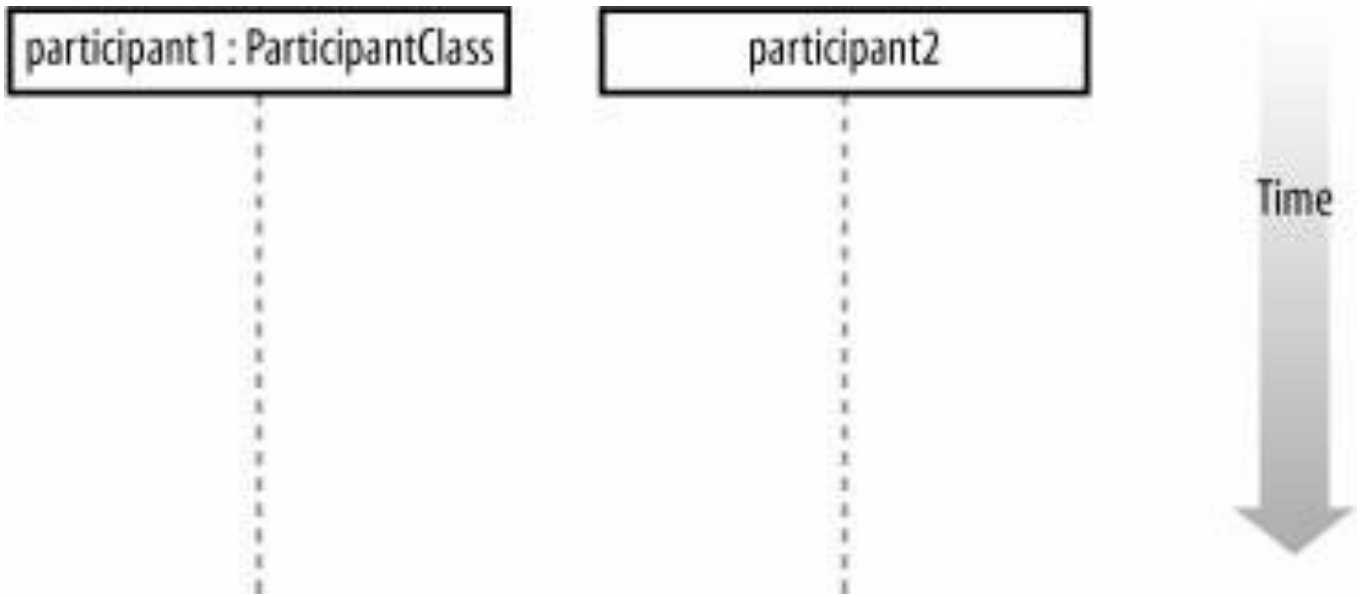
```
name [selector] : class_name
```

- Some examples:
  - admin
  - : ContentManagementSystem
  - admin : Administrator
  - eventHandlers[2] : EventHandler

6.2 Time

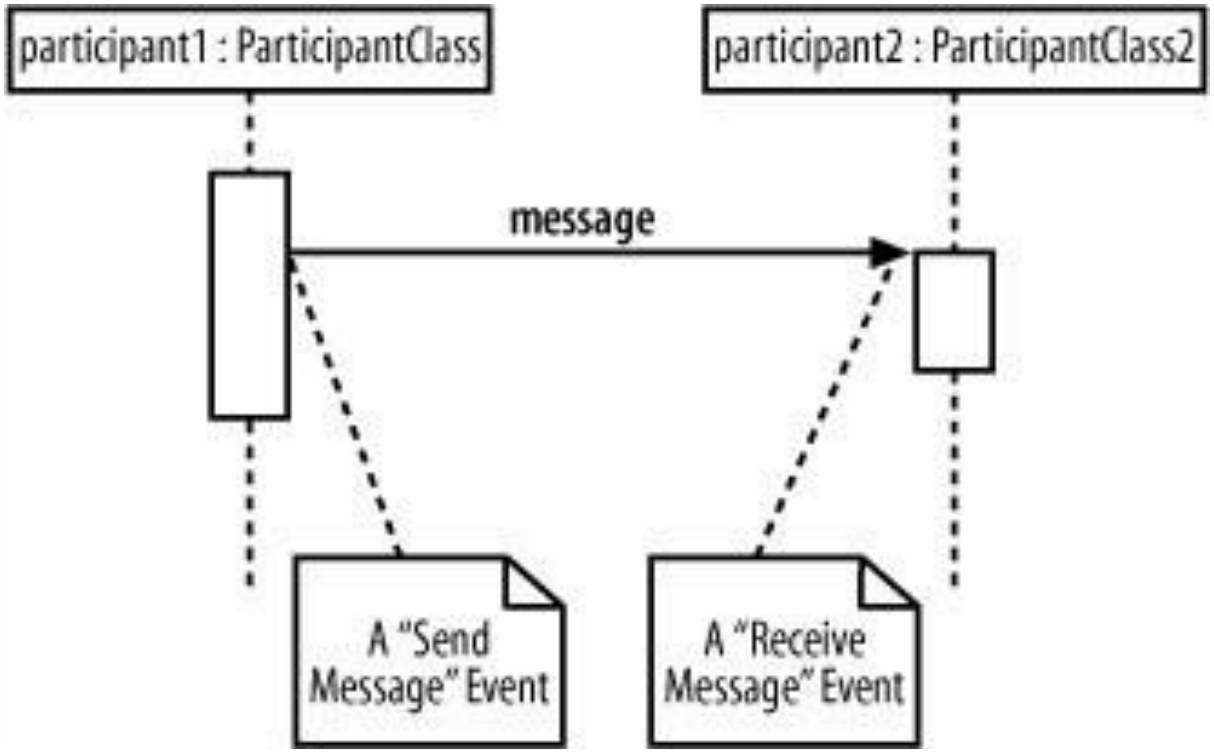
- Time flows top to bottom on the diagram.
- The vertical order of messages shows the sequence in which interactions happen.
- The diagram shows the order of events, not the actual duration.

6.2 Time



6.3 Events and Messages

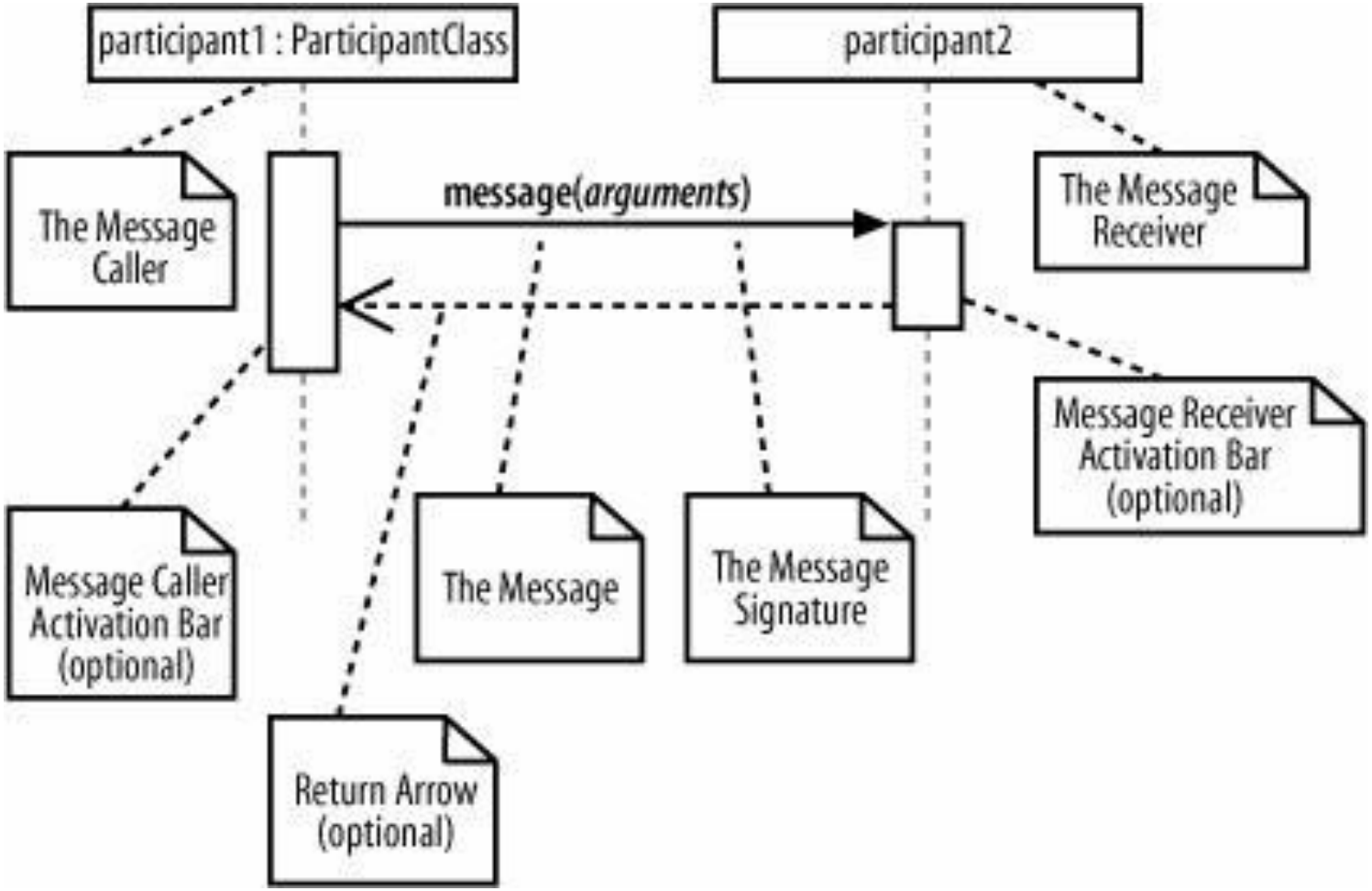
- Event: any point where something happens during the interaction.



6.3 Events and Messages

- Message: what being sent from a participant (the message caller) to another participant (the message receiver) to do something
  - Messages can flow in any direction (left to right or right to left)
- Activation Bar: is used to show that a participant is active

6.3 Events and Messages



6.3 Events and Messages

- Message Signatures
  - The format for a message signature is

```
attribute = signal_or_message_name (arguments) :
  return_type
```

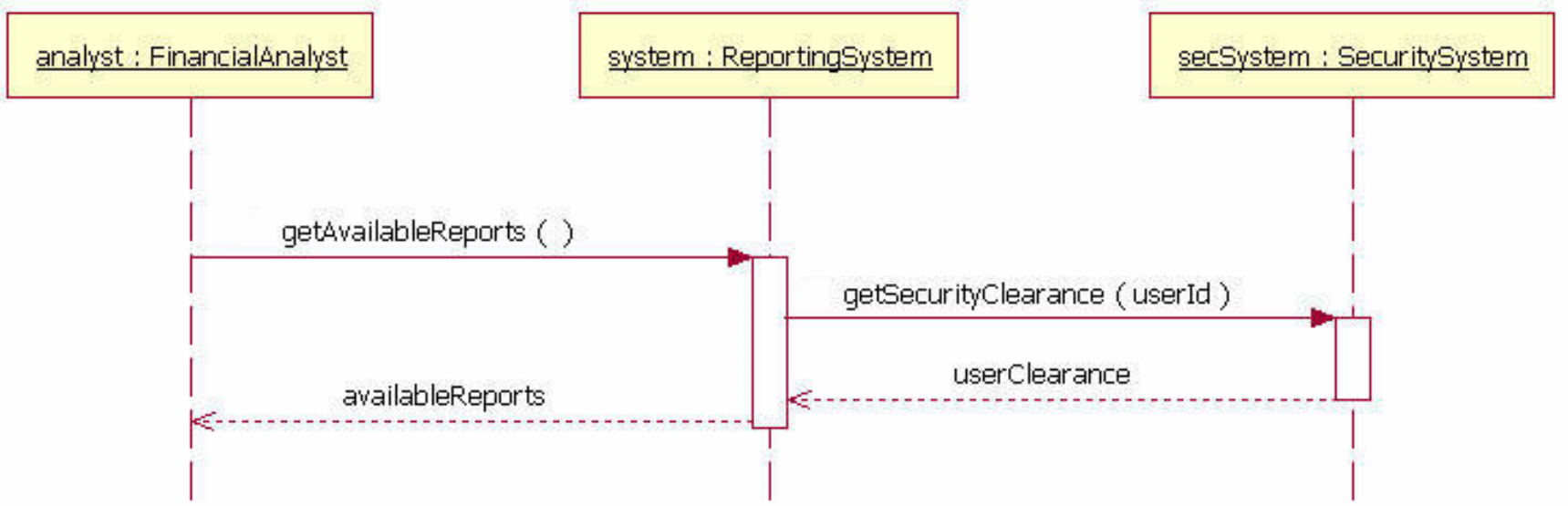
  - The format of an argument is

```
<name>:<class>
```

- Some examples:
  - doSomething()
  - doSomething(arg1 : Class1, arg2 : Class2)
  - doSomething() : Class3
  - myVar = doSomething() : Class3

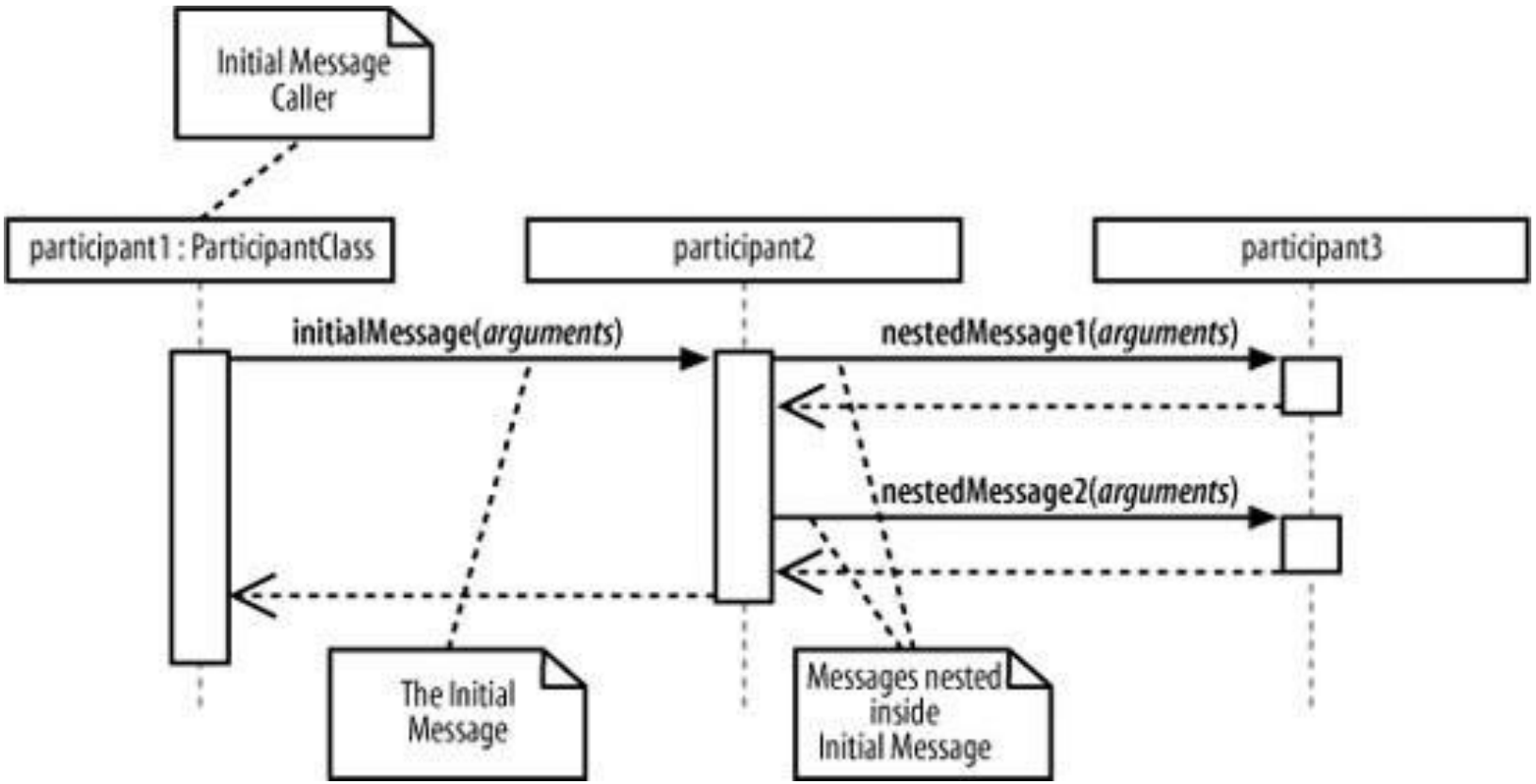
6.3 Events and Messages

- Return messages are optional in sequence diagrams.
- The return value from the operation is placed above this dotted line.
- Including return messages can make a sequence diagram easier to read, especially when a value is returned.



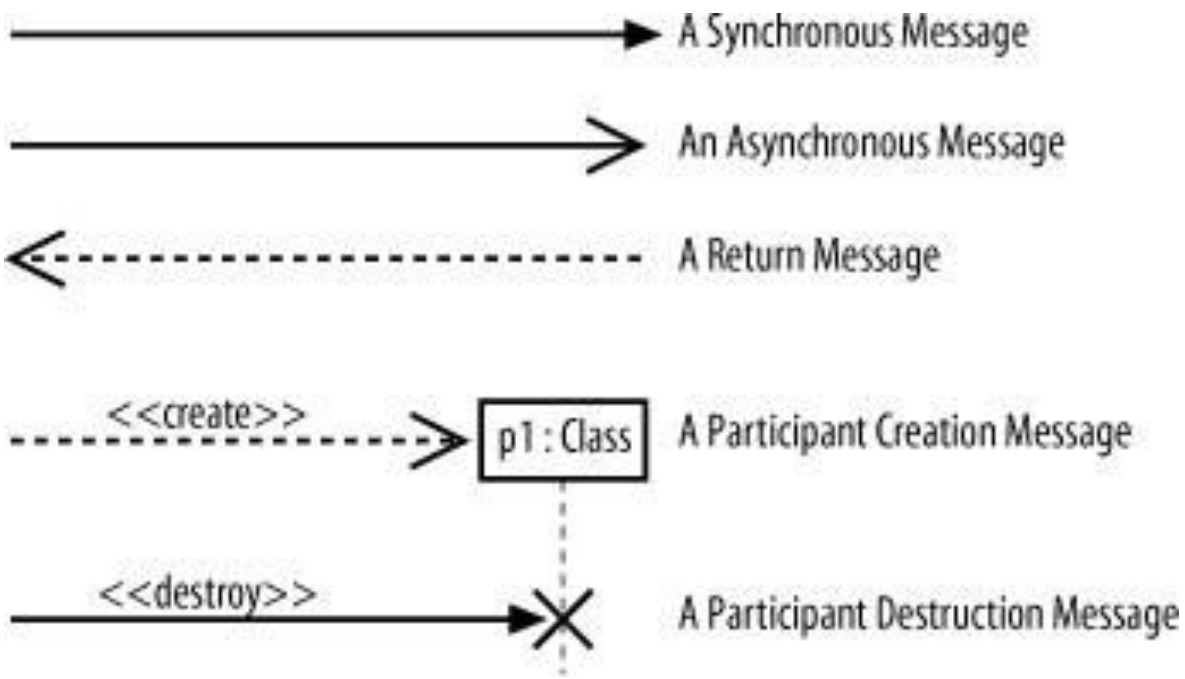
6.3 Events and Messages

- Nested Messages: when a message from one participant results in one or more messages being sent by the receiving participant



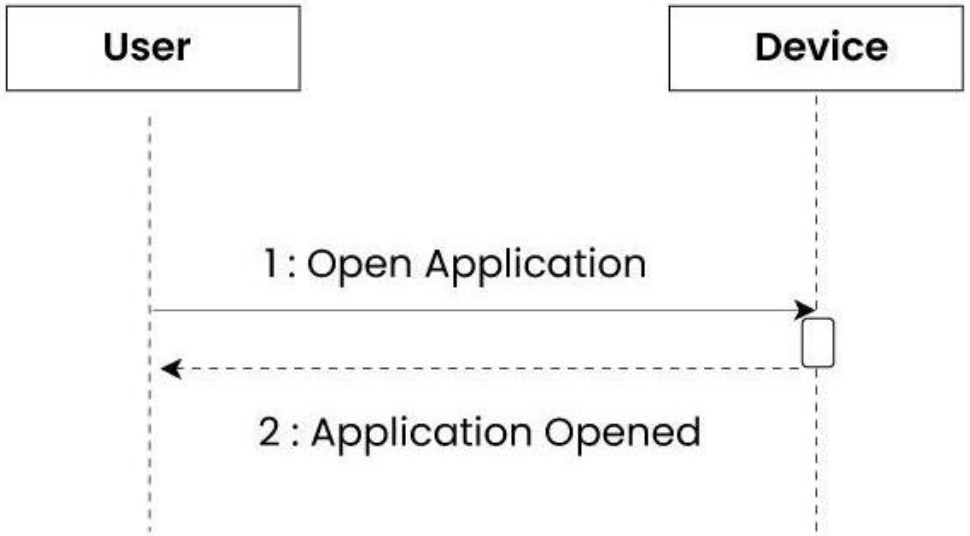
6.4 Message Arrows

- There are 5 main types of message arrow



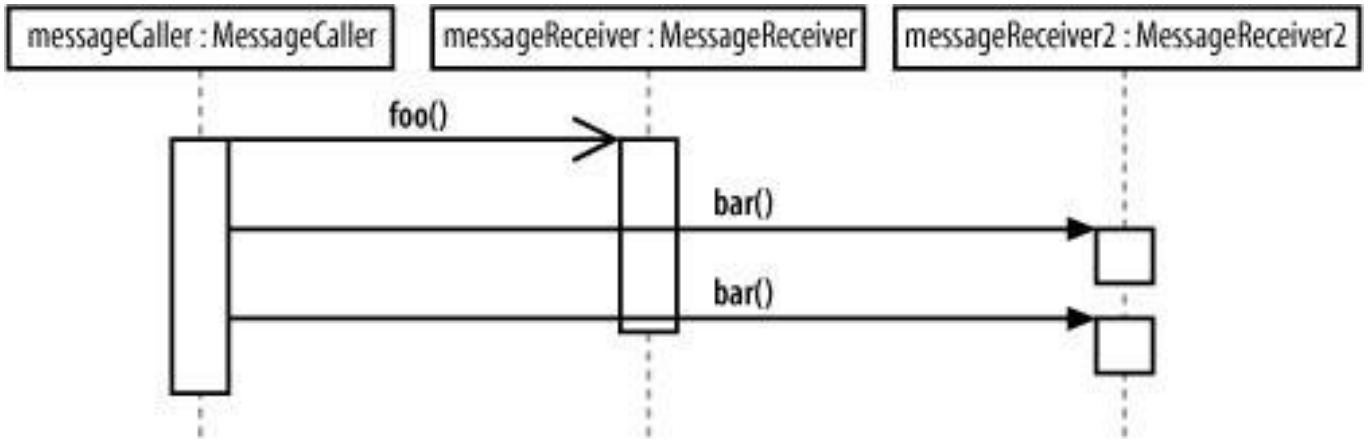
6.4 Message Arrows

- **Synchronous Message:** this type of message is invoked when the Message Caller waits for the Message Receiver to return from the message invocation



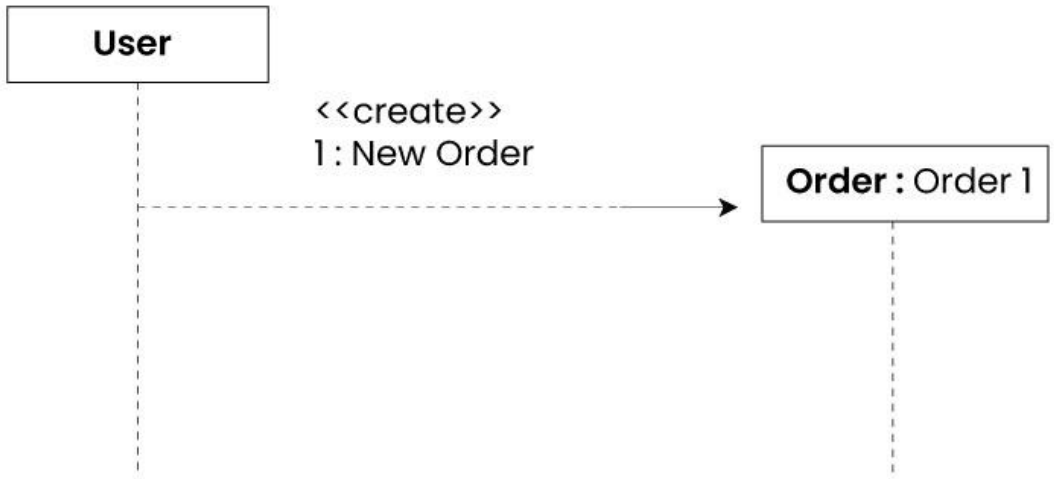
6.4 Message Arrows

- **Asynchronous Message:** this type of message is invoked by a Message Caller on a Message Receiver, but the Message Caller does not wait for the message invocation to return before carrying on with the rest of the interaction's steps.



6.4 Message Arrows

- **Participant Creation Messages:** use a Create message to instantiate a new object in the sequence diagram. There are situations when a particular message call requires the creation of an object



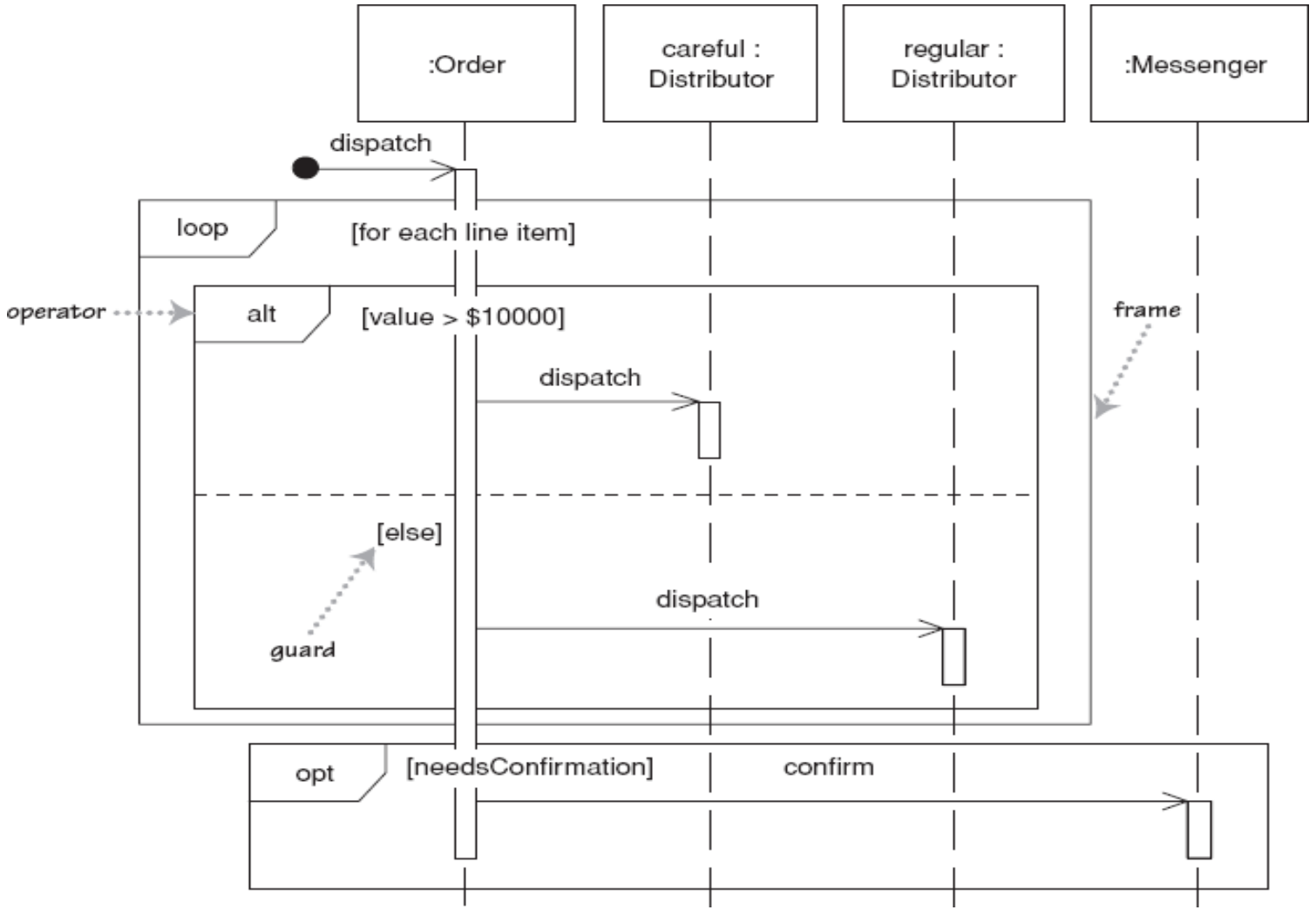
6.4 Message Arrows

- **Participant Destruction Messages:** use a Delete Message to delete an object. When an object is deallocated memory or is destroyed within the system we use the Delete Message symbol. It destroys the occurrence of the object in the system.

6.5 Selection and Loops

- Frame: Box around part of diagram to indicate `if` or `loop`
  - `if -> [opt]` : condition
  - `if/else -> [alt]` : condition, separated by horizontal dashed line
  - `loop -> [loop]` : condition or items to loop over

6.5 Selection and Loops



Exercise

Model the Withdraw Money use case of a Bank ATM system by completing the following:

1. Identify the two main participants involved in the interaction.

3. Draw a System Sequence Diagram (SSD) that includes at least:

- Card insertion
- Password entry
- Withdrawal selection
- Amount entry
- Cash dispensing
- Card ejection

References

- Miles, R. (2006). *Learning UML 2.0*. O'Reilly.
- Chonoles, M. & Schardt, J. (2003). *UML 2 for Dummies*. Wiley Publishing.