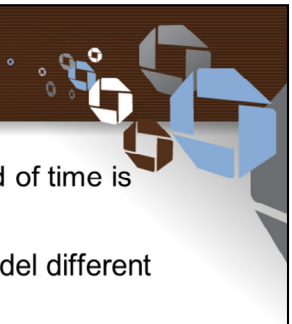
Interaction Diagrams

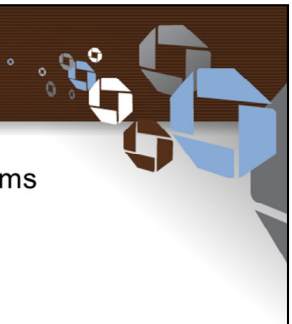


- The interaction between multiple objects over a period of time is visually depicted through interaction diagrams.
- The following interaction diagrams can be used to model different aspects of system-actor and object-object interaction:
  - Sequence diagrams
  - Communication diagrams
  - Timing diagrams
  - Interaction overview diagrams

Interaction Diagrams (Cont'd)

- Sequence and communication diagrams are the most commonly used ones because of the insight they provide into the behavior of the system.
- To draw these two types of interaction diagrams, you need to identify the following components:
  - Participating elements
  - Messages passed between the participants
  - Time sequence or order of flow of messages
  - Organization of the participating elements

Sequence Diagrams



- To model the behavior of a system, sequence diagrams combine:
  - Class diagrams
  - Use cases
- Sequence diagrams are of two types:
  - System sequence diagrams - Model the interaction between the system and external actors.
  - Object sequence diagrams - Model the interaction between various objects involved in a particular process.
  - During domain modeling in the OOA phase, you can create system sequence diagrams for each use case. The object sequence diagrams are essentially part of the solution model.

Sequence Diagrams (Cont'd)

- System sequence diagrams describe the events that occur in the actor-system interaction.
- A system input event (or system event) is an event generated when an actor provides some input to the system. A system operation is invoked in response to the system input event.
- The procedure for creating a system sequence diagram for the basic and interesting alternative scenarios of a use case consists of the following steps:
  - Draw a lifeline representing the system.
  - Draw a lifeline for each actor that interacts with the system for the given scenario.
  - Identify the system input events from the scenario.
  - Identify the system operations.

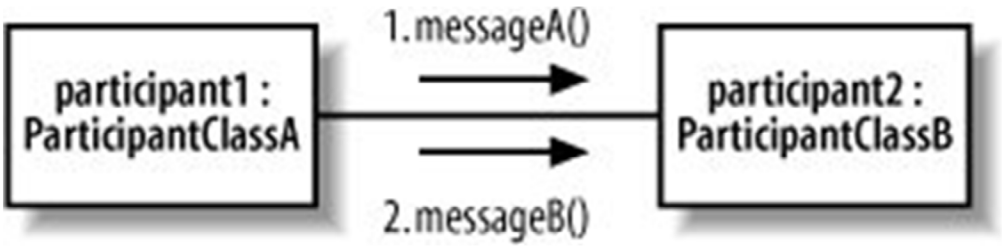
Sequence Diagrams (Cont'd)

- By creating system sequence diagrams, you:
  - Get more clarity on the behavior of entities in the problem domain
  - Can also identify gaps in your understanding

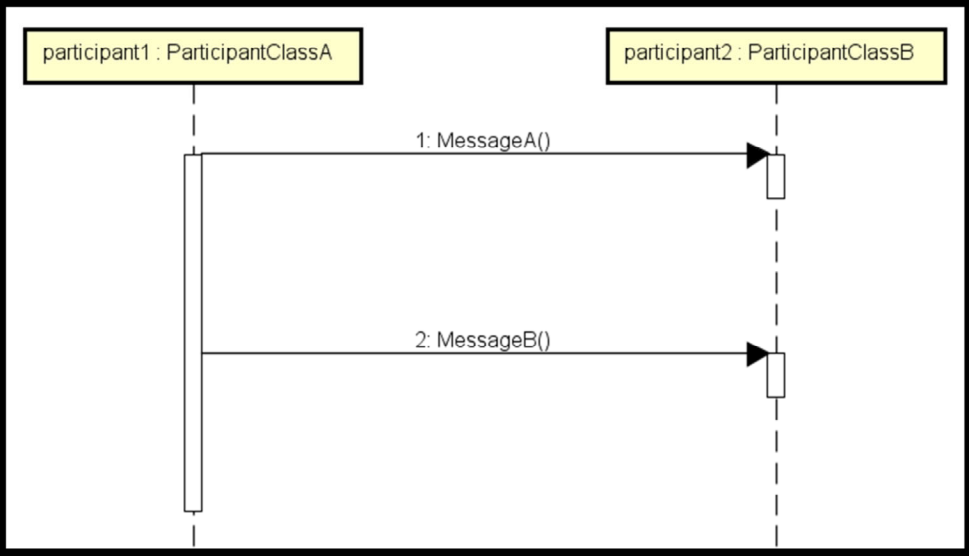
UML

Communication Diagram

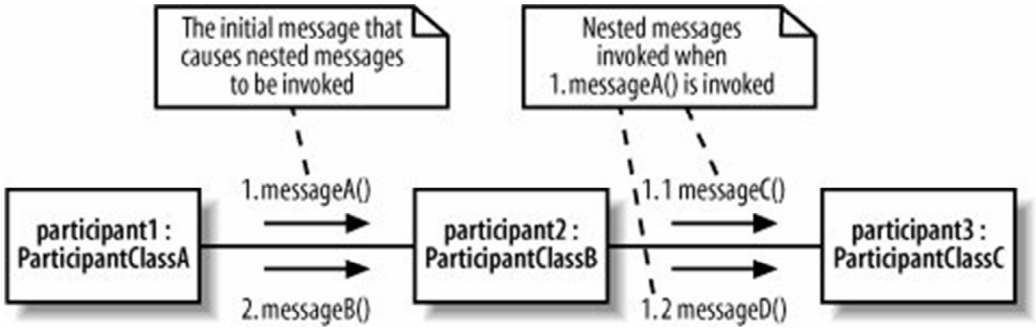
Example


participant1 :
ParticipantClassA
1.messageA()
2.messageB()
participant2 :
ParticipantClassB

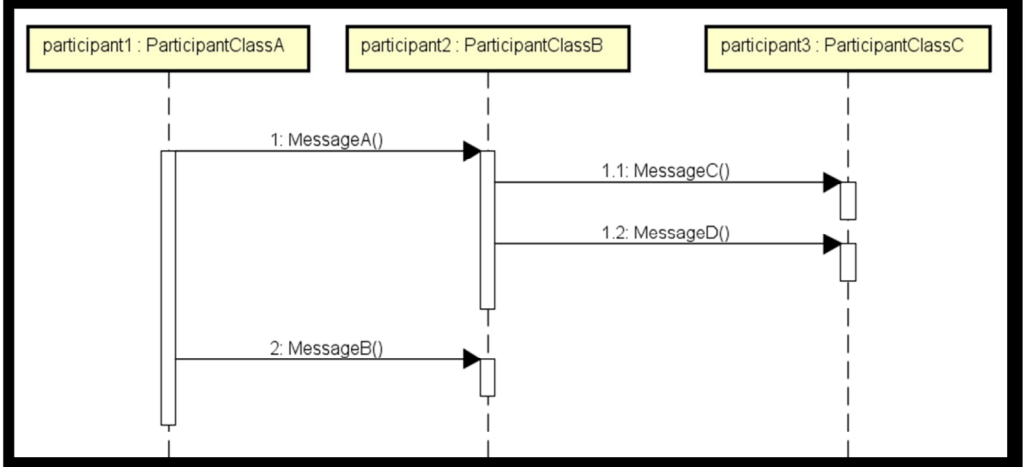
Equivalent Sequence Diagram



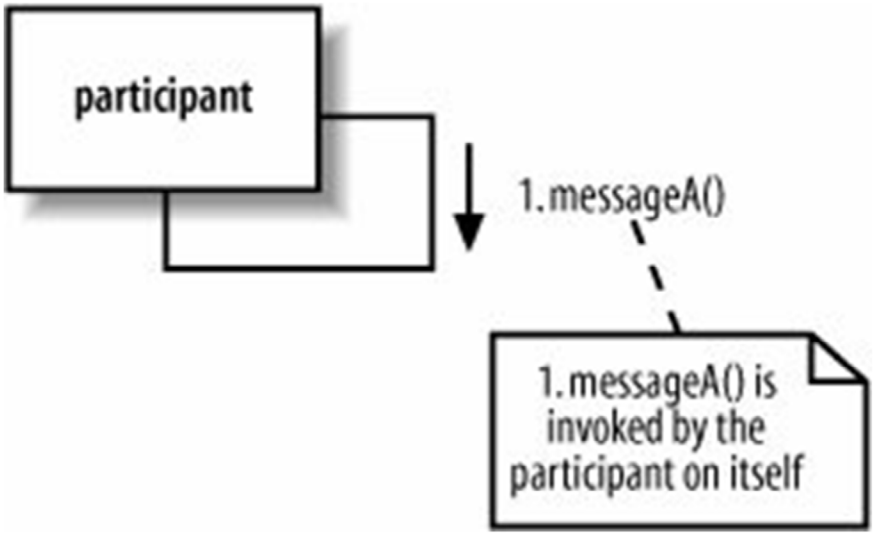
Nested Messages



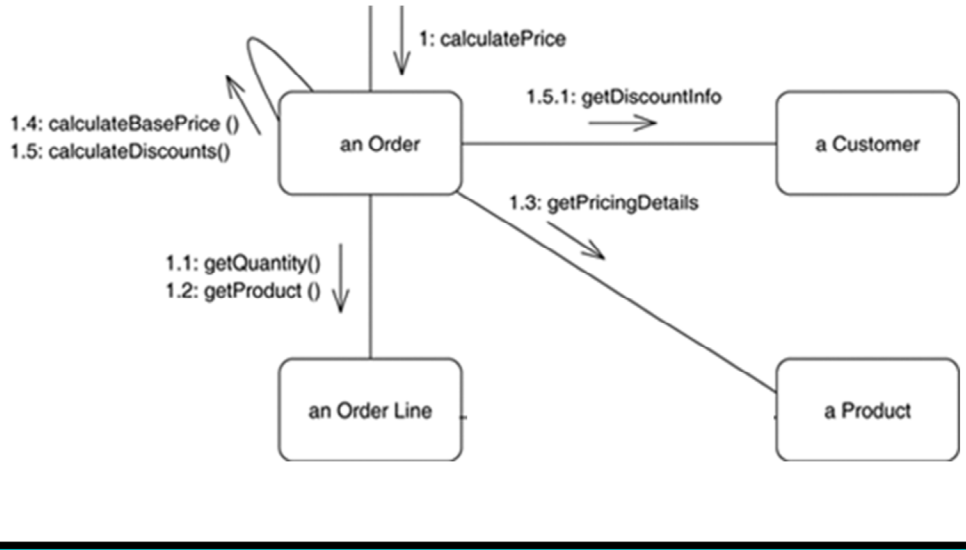
Equivalent Sequence Diagram



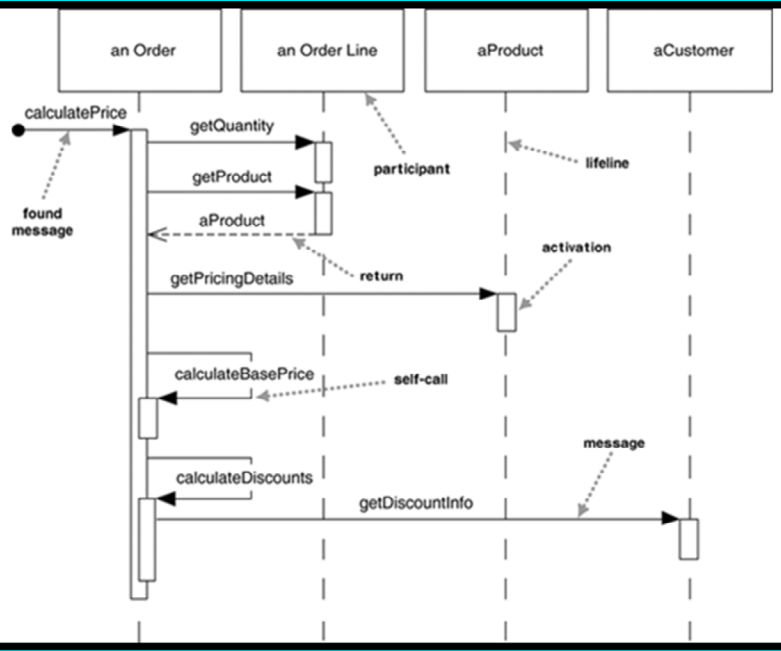
Self call


participant
1.messageA()
1.messageA() is invoked by the participant on itself

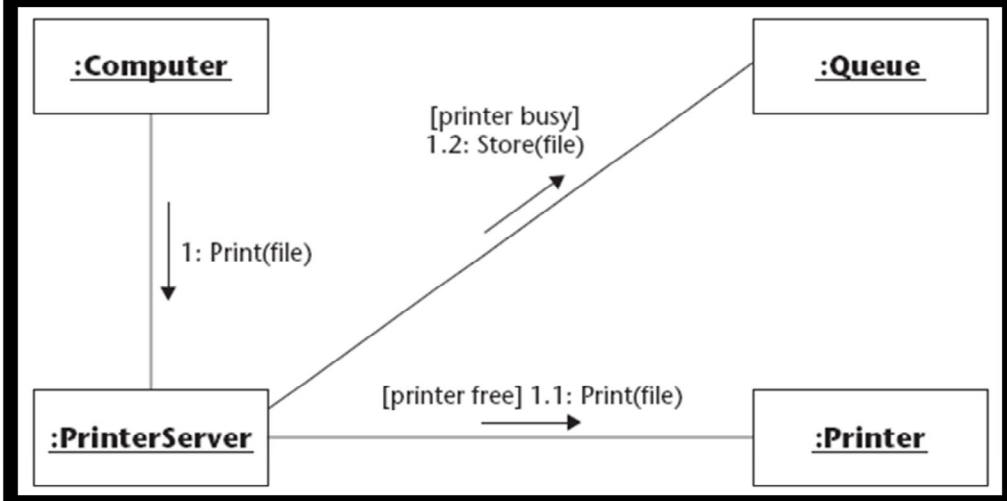
Collaboration Diagram



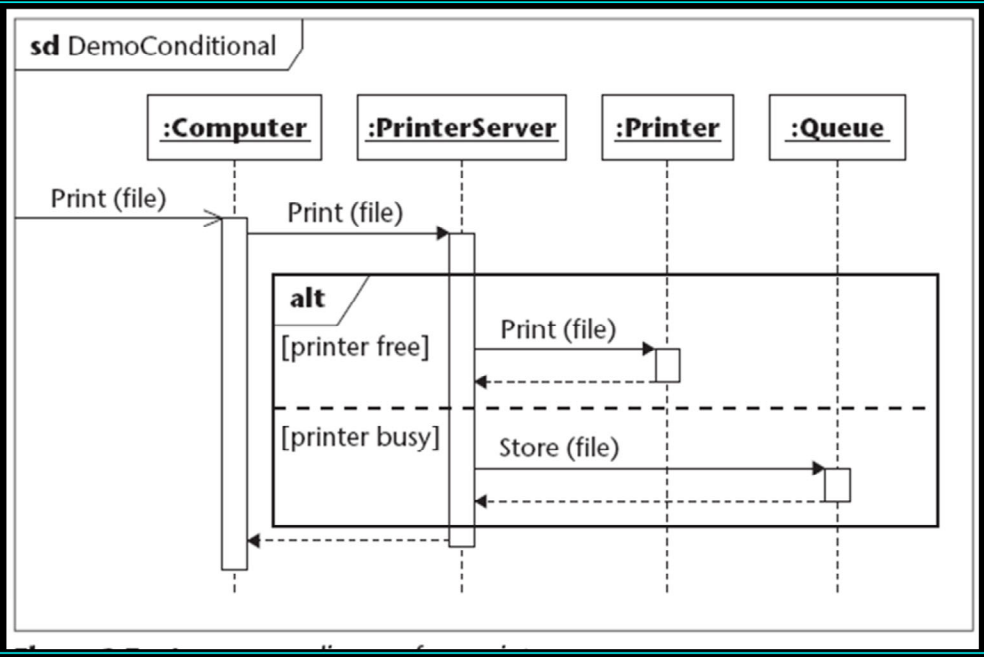
Equivalent Sequence Diagram



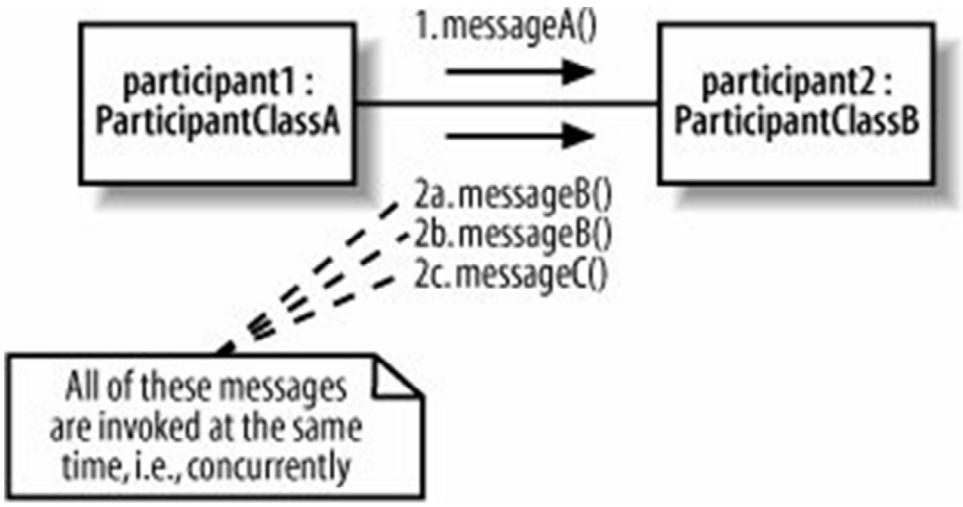
Print Server



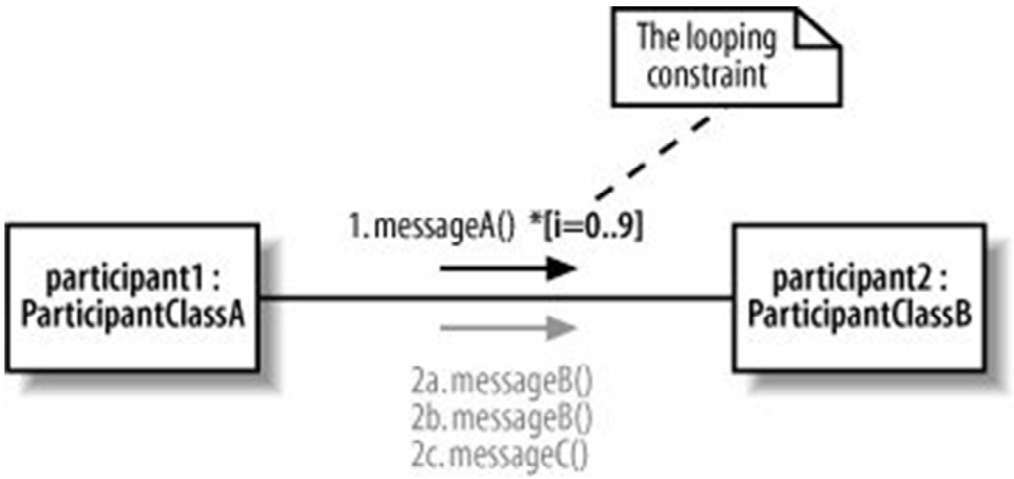
Print Server



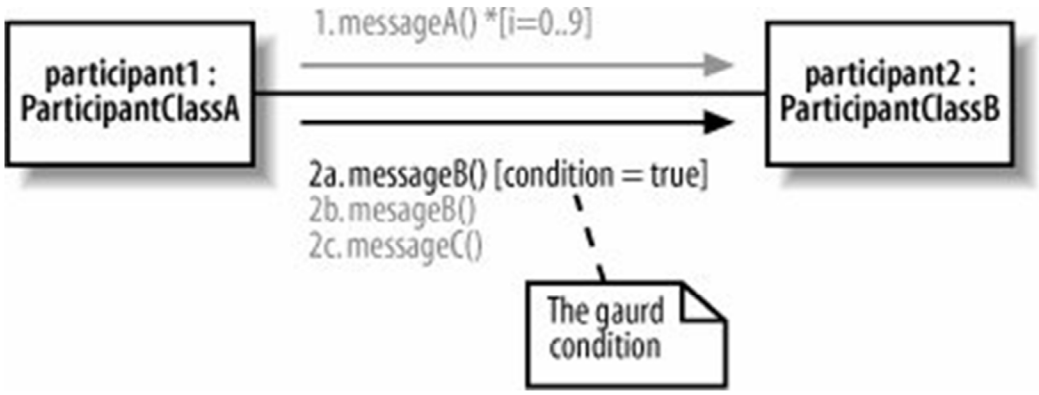
Parallel invocation



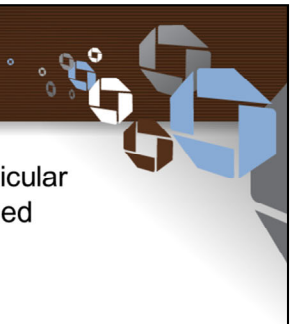
Looping


The looping constraint
1. messageA() *[i=0..9]
participant1 : ParticipantClassA
participant2 : ParticipantClassB
2a. messageB()
2b. messageB()
2c. messageC()

Conditional invocation

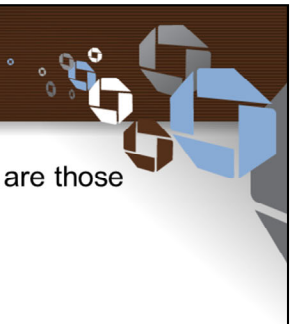


Communication Diagrams



- To model the collaboration of various objects in a particular sequence, you can use communication diagrams (called collaboration diagrams in UML 1.x).
- Communication diagrams show:
  - Objects involved in a sequence
  - Their relationship in a collaboration
  - The messages passed in the collaboration
  - The sequence in which the messages are passed

Communication Diagrams (Cont'd)



- The basic notations used in communication diagrams are those of:
  - Objects
  - Links
  - Messages
- The messages are shown on the links as arrows and a detailed sequence expression.
- The sequence expression consists of:
  - Nested sequence numbers – The sequence numbers follow a logical nesting.
  - Message name – It can consist of the basic name, return type and value and parameters.
  - Recurrence – It can be used to define a guard condition and iteration on the message.