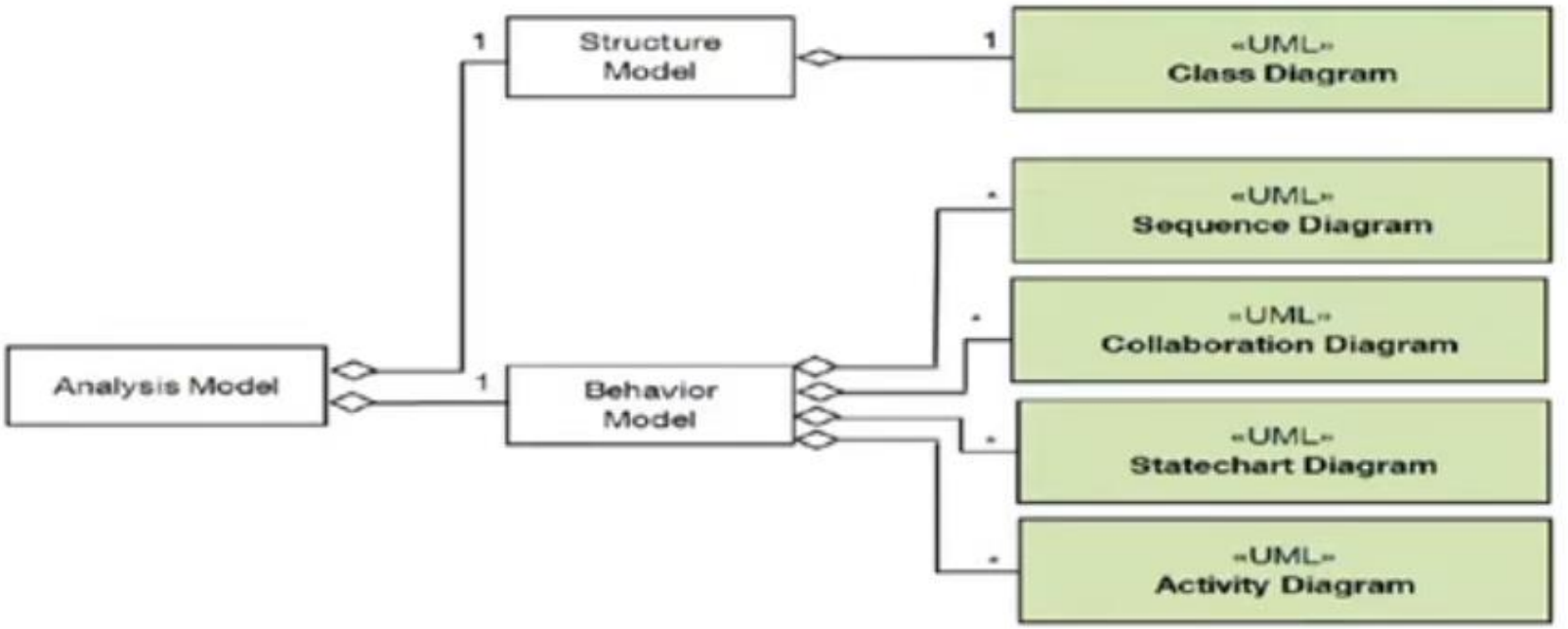
Communication (Collaboration Diagram)

Overview

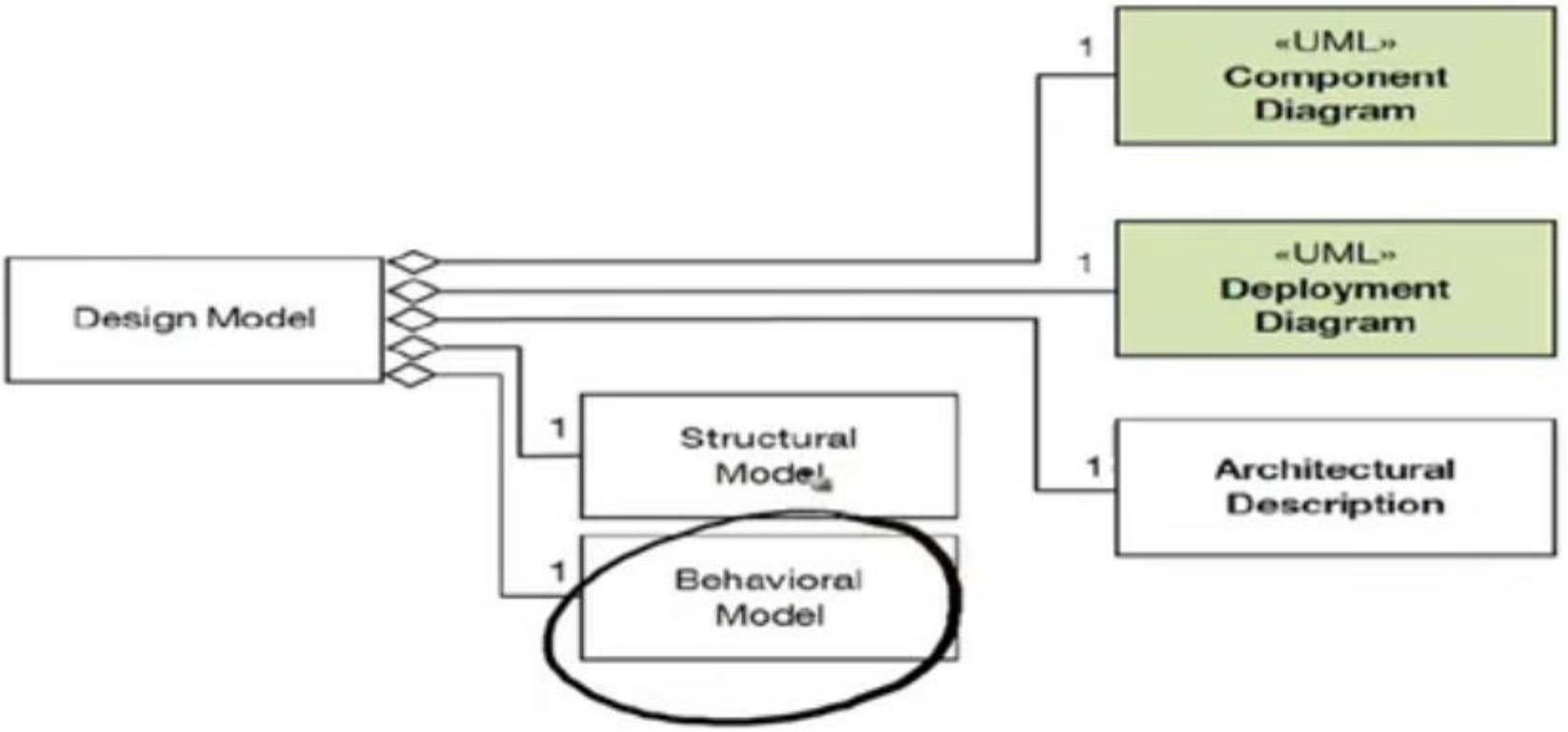
- Collaboration Diagram (Communication Diagram)
- Frames
- Lifelines
- Messages
  - Sequence Expression
  - Sequential Order
  - Concurrent Threads
  - Guards
  - Recurrence & Iteration
- Examples

Communication (Collaboration Diagram)



- In the Analysis Phase the problem domain is analyzed and refined from the Requirements Phase
- The behavior model of the system is hence understood in this phase
- Communication (Collaboration) diagrams is a result of the Analysis Phase

Communication (Collaboration Diagram)



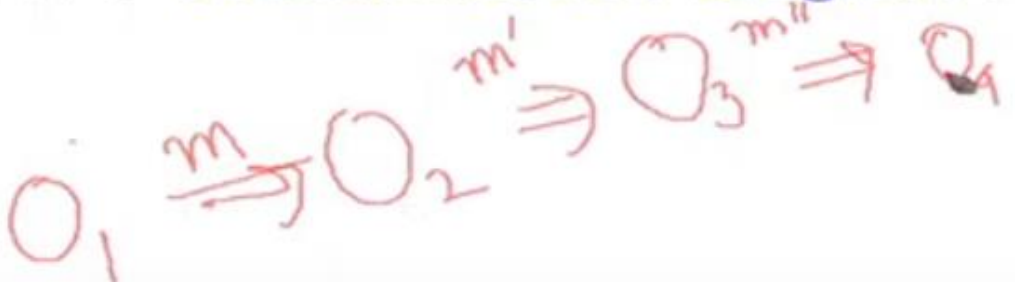
- Communication (Collaboration) diagram is included in the Behavioral Model
- It is further refined in the Design Phase

Communication (Collaboration Diagram)

- Communication Diagram (called Collaboration Diagram in UML 1.x) shows interactions between objects and / or parts (represented as lifelines) using sequenced messages in a free-form arrangement
- Communication Diagram is a UML Behavior Diagram
- Communication Diagram depicts the inter-object behavior of a system, ordered by space
- The major components of a Communication Diagram are:
  - Frames
  - Lifeline
  - Messages

Communication (Collaboration Diagram)

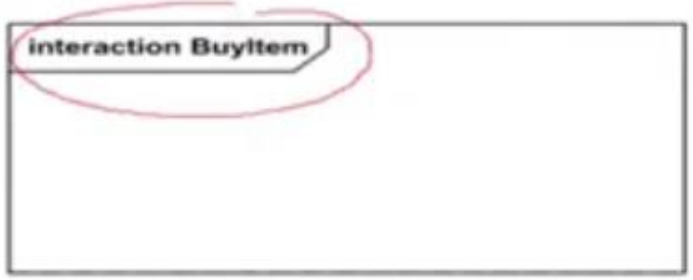
- Communication Diagram (called Collaboration Diagram in UML 1.x) shows interactions between objects and / or parts (represented as lifelines) using sequenced messages in a free-form arrangement
- Communication Diagram is a UML Behavior Diagram
- Communication Diagram depicts the inter-object behavior of a system, ordered by space
- The major components of a Communication Diagram are:
  - Frames
  - Lifeline
  - Messages



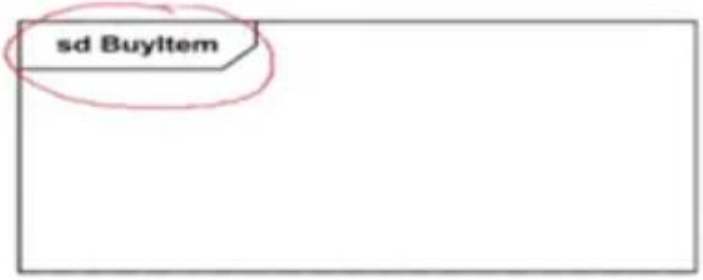
Communication (Collaboration Diagram)

Frames

- Communication diagrams could be shown within a rectangular frame with the name in a compartment in the upper left corner



Interaction Frame for Communication Diagram BuyItem



Sd Frame (short frame) for Communication Diagram BuyItem

Source: *UML 2.5 Diagrams Overview*: http://www.uml-diagrams.org/uml-25-diagrams.html (20-Aug-16)

Communication (Collaboration Diagram)

Lifelines

- Lifeline is a specialization of named element which represents an individual participant in the interaction
- A Lifeline is shown as a rectangle (corresponding to the head in sequence diagrams)
- Lifeline in sequence diagrams does have tail representing the line of life whereas lifeline in communication diagram has no line, just head
- The information identifying a lifeline is depicted as ObjectName[selector]:ClassName

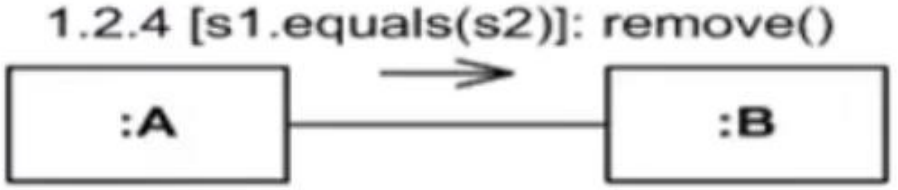
:User	data:Stock	x[k]:X
Anonymous lifeline of class User	Lifeline "data" of class Stock	Lifeline "x" of class X is selected with selector [k]

Source: *UML 2.5 Diagrams Overview*: http://www.uml-diagrams.org/uml-25-diagrams.html (20-Aug-16)

Communication (Collaboration Diagram)

Messages

- Message in Communication Diagram is shown as a line with sequence expression and **arrow** above the line
- The arrow indicates direction of the communication



Instance of class A sends remove() message to instance of B if s1 is equal to s2

Source: *UML 2.5 Diagrams Overview*: http://www.uml-diagrams.org/uml-25-diagrams.html (20-Aug-16)

Communication (Collaboration Diagram)

Sequence Expression in Messages

- The sequence expression is a dot separated list of sequence terms followed by a colon (":") and message name after that:
  sequence-expression ::= sequence-term '.' . . . ':' message-name
- Example: **3b.2.2:m5** : Sequence expression **3b.2.2** and message name **m5**
- Each Sequence term
  sequence-term ::= [integer [name]] [recurrence]
- The **integer** represents the **sequential order** of the message within the next higher level of procedural calling

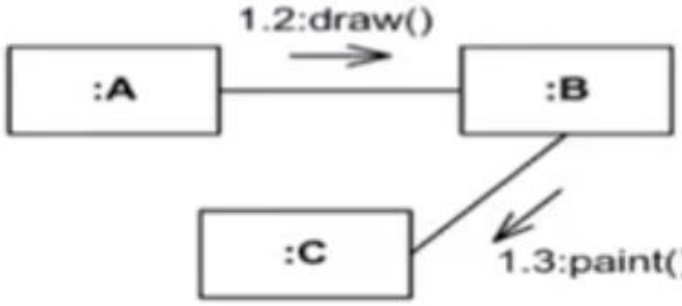
Source: *UML 2.5 Diagrams Overview*: http://www.uml-diagrams.org/uml-25-diagrams.html (20-Aug-16)

Communication (Collaboration Diagram)

Sequence Expression in Messages

Sequential Order

- **Example**:
  - message with sequence 2 follows message with sequence 1
  - 2.1 follows 2
  - 5.3 follows 5.2 within activation 5
  - 1.2.4 follows message 1.2.3 within activation 1.2.



Instance of A sends draw() message to instance of B, and after that B sends paint() to C

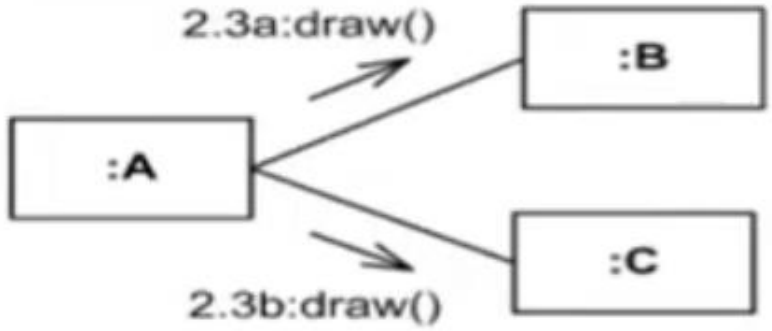
Source: *UML 2.5 Diagrams Overview*: http://www.uml-diagrams.org/uml-25-diagrams.html (20-Aug-16)

Communication (Collaboration Diagram)

Sequence Expression in Messages

Concurrent Thread

- The **name** represents a **concurrent thread** of control
- **Example**:
  - messages 2.3a and 2.3b are concurrent within activation 2.3
  - 1.1 follows 1a and 1b
  - 3a.2.1 and 3b.2.1 follow 3.2



Instance of A sends draw() messages concurrently to instance of B and to instance of C

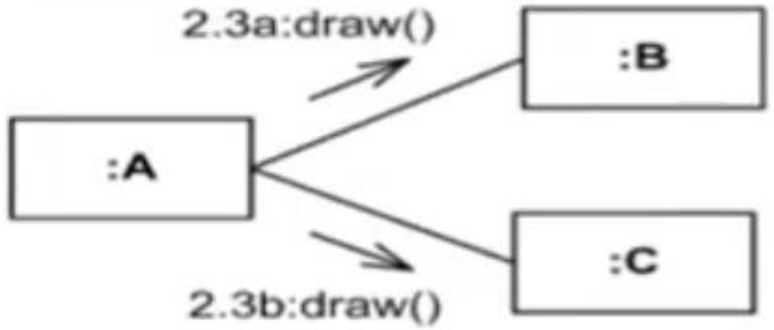
Source: *UML 2.5 Diagrams Overview*: http://www.uml-diagrams.org/uml-25-diagrams.html (20-Aug-16)

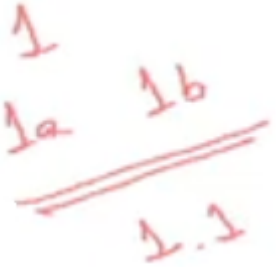
Communication (Collaboration Diagram)

Sequence Expression in Messages

Concurrent Thread

- The **name** represents a **concurrent thread** of control
- **Example**:
  - messages 2.3a and 2.3b are concurrent within activation 2.3
  - 1.1 follows 1a and 1b
  - 3a.2.1 and 3b.2.1 follow 3.2





Instance of A sends draw() messages concurrently to instance of B and to instance of C

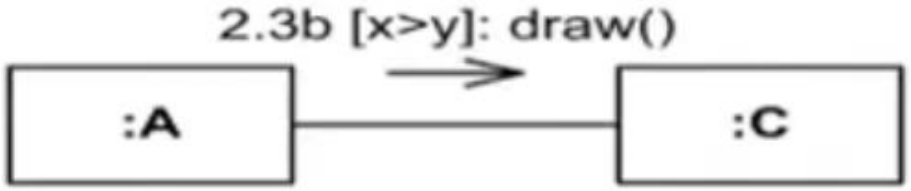
Source: *UML 2.5 Diagrams Overview*: http://www.uml-diagrams.org/uml-25-diagrams.html (20-Aug-16)

Communication (Collaboration Diagram)

Sequence Expression in Messages

Guard

- A **guard** specifies condition for the message to be sent (executed) at the given nesting depth
- **Example**:
  - **2.3b [x>y]: draw()**: message draw() will be executed if x is greater than y
  - **1.1.1 [s1.equals(s2)]: remove()** – message remove() will be executed if s1 equals s2



Instance of class A will send message draw() to the instance of C, if x > y

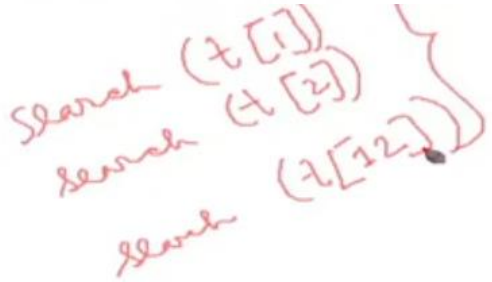
Source: *UML 2.5 Diagrams Overview*: http://www.uml-diagrams.org/uml-25-diagrams.html (20-Aug-16)

Communication (Collaboration Diagram)

Sequence Expression in Messages

Recurrence & Iteration

- The **recurrence** defines conditional or iterative execution of zero or more messages that are executed depending on the specified condition
  recurrence ::= branch | loop , branch ::= '[' guard ']'
- An **iteration** specifies a sequence of messages at the given nesting depth
- **Notation**:
  - * : *Messages Executed Sequentially*
  - *|| : *Messages Executed Concurrently*



Example:

- **4.2c *[i=1..12]: search(t[i])** – search() will be executed 12 times, one after another
- **4.2c *||[i=1..12]: search(t[i])** – 12 search() messages will be sent concurrently
- **2.2 *: notify()** – message notify() will be repeated some unspecified number of times

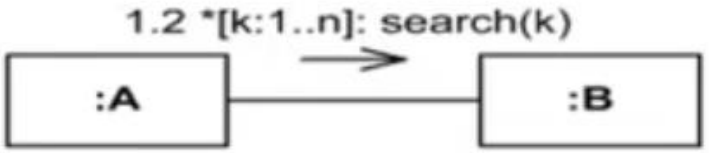
Source: *UML 2.5 Diagrams Overview*: http://www.uml-diagrams.org/uml-25-diagrams.html (20-Aug-16)

Communication (Collaboration Diagram)

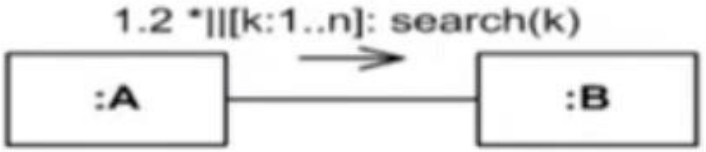
Sequence Expression in Messages

Recurrence & Iteration (Examples)

Example:



Instance of class A will send search() message to instance of B n times, one by one

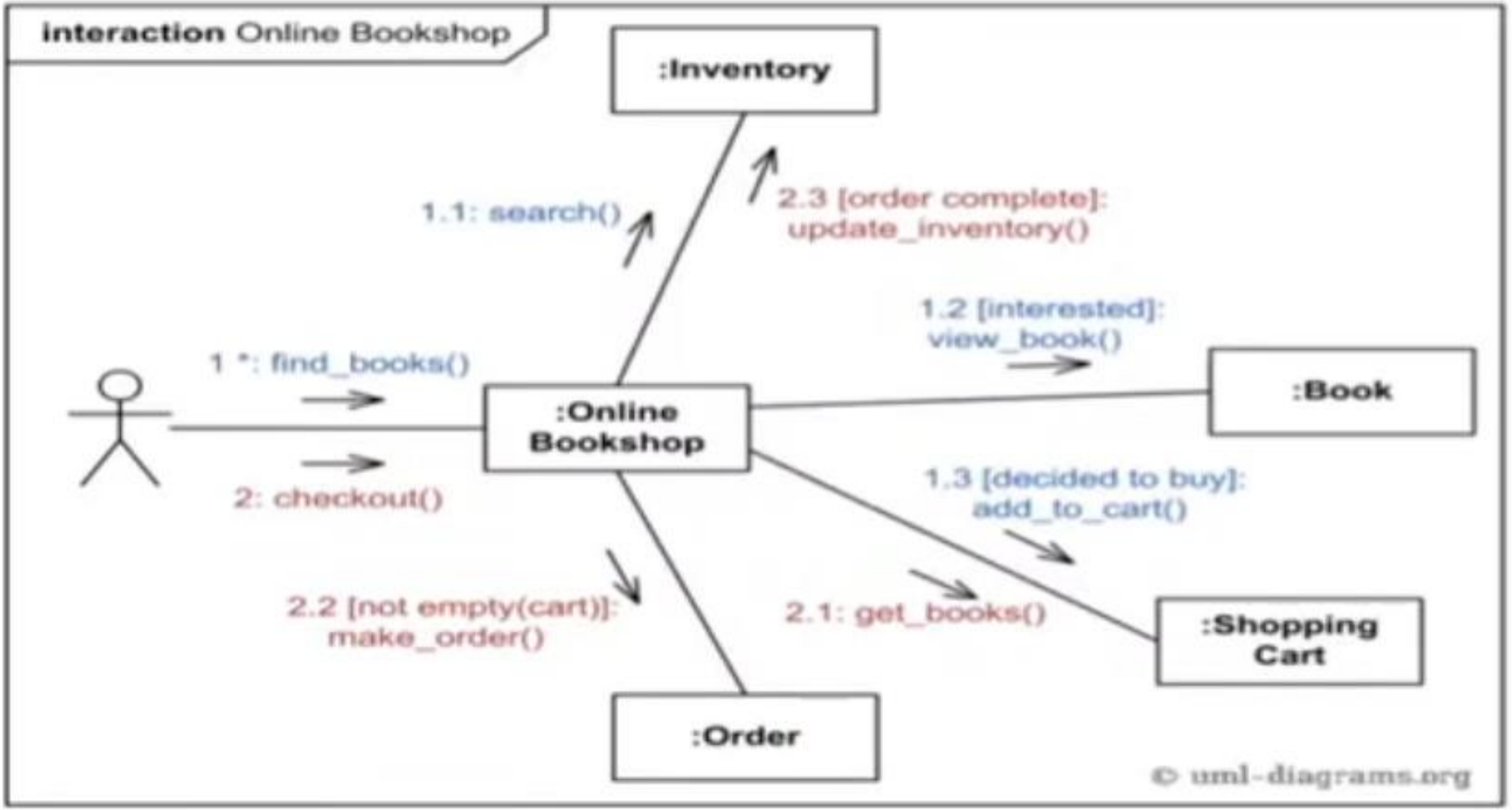


Instance of class A will send n concurrent search() messages to instance of B

Source: *UML 2.5 Diagrams Overview*: http://www.uml-diagrams.org/uml-25-diagrams.html (20-Aug-16)

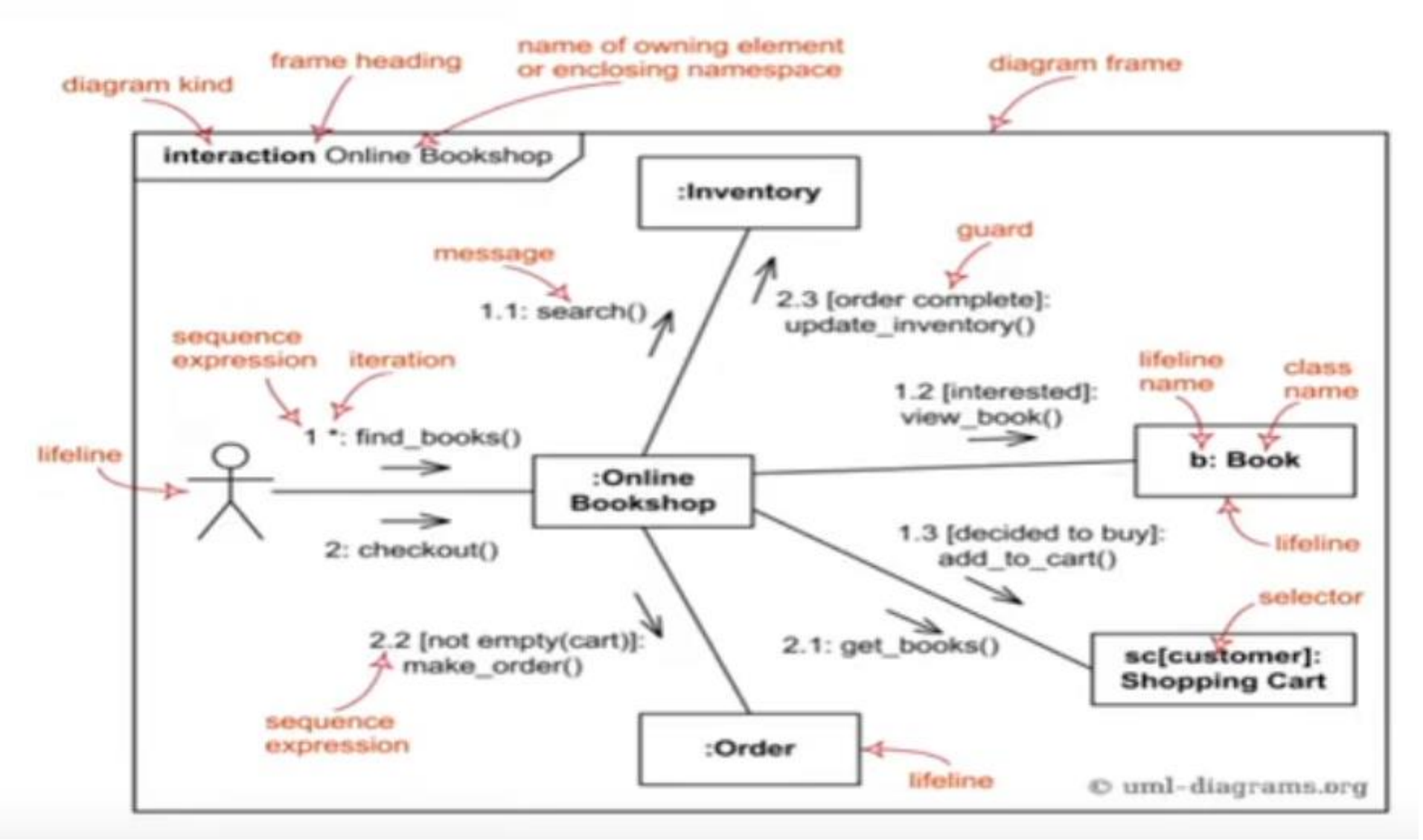
Communication (Collaboration Diagram)

Online Book Shop



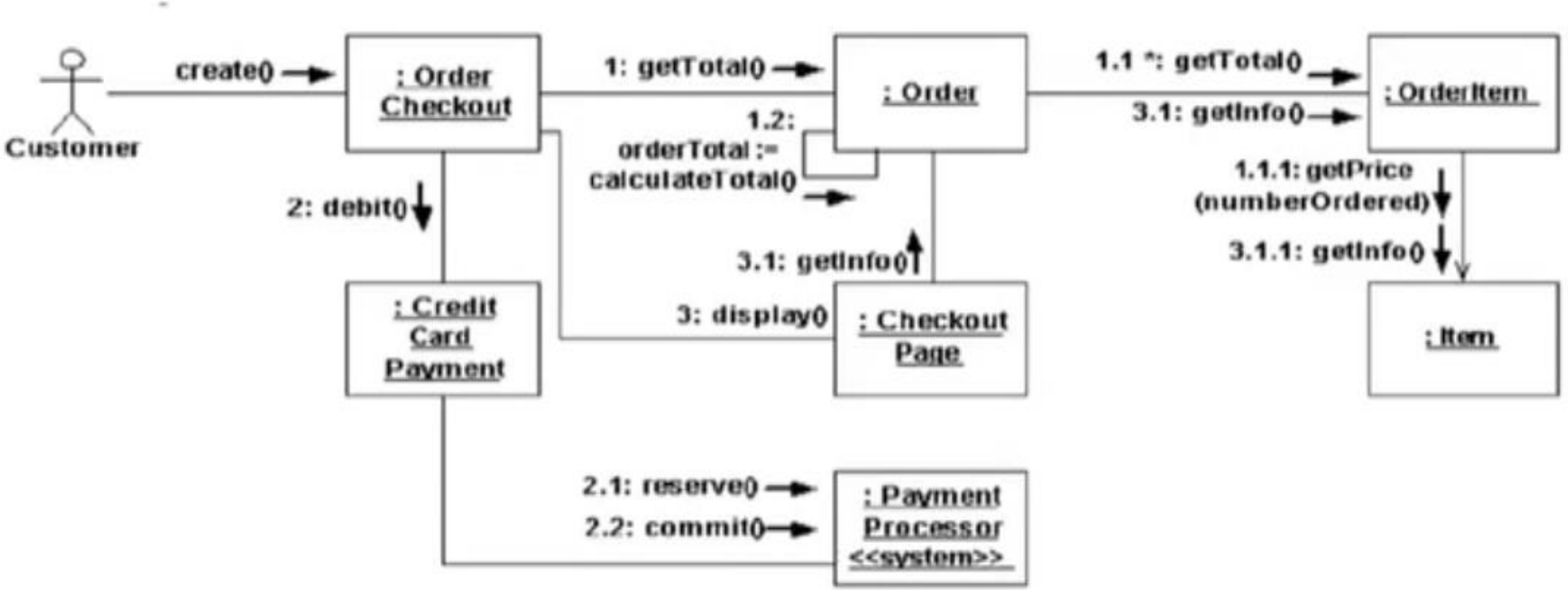
Communication (Collaboration Diagram)

Online Book Shop (Annotated)



Communication (Collaboration Diagram)

Order Management



Source: *http://agilemodeling.com/style/collaborationDiagram.htm (20-Aug-16)*

Communication (Collaboration Diagram) Summary

- Collaboration Diagram (Communication Diagram)
- Frames
- Lifelines
- Messages
  - Sequence Expression
  - Sequential Order
  - Concurrent Threads
  - Guards
  - Recurrence & Iteration
- Examples

- Communication Diagrams are introduced
- Various components of Communication Diagrams like Frames, Lifeline, Messages are discussed
- Examples are illustrated

Reference

Source: NPTEL - Object-Oriented Analysis and Design, IIT Kharagpur Prof. Partha Pratim Das Prof. Samiran Chattopadhyay Prof. Kausik Datta

https://nptel.ac.in/courses/106105153