

# Basic Communication diagram Notation

①

The notations are

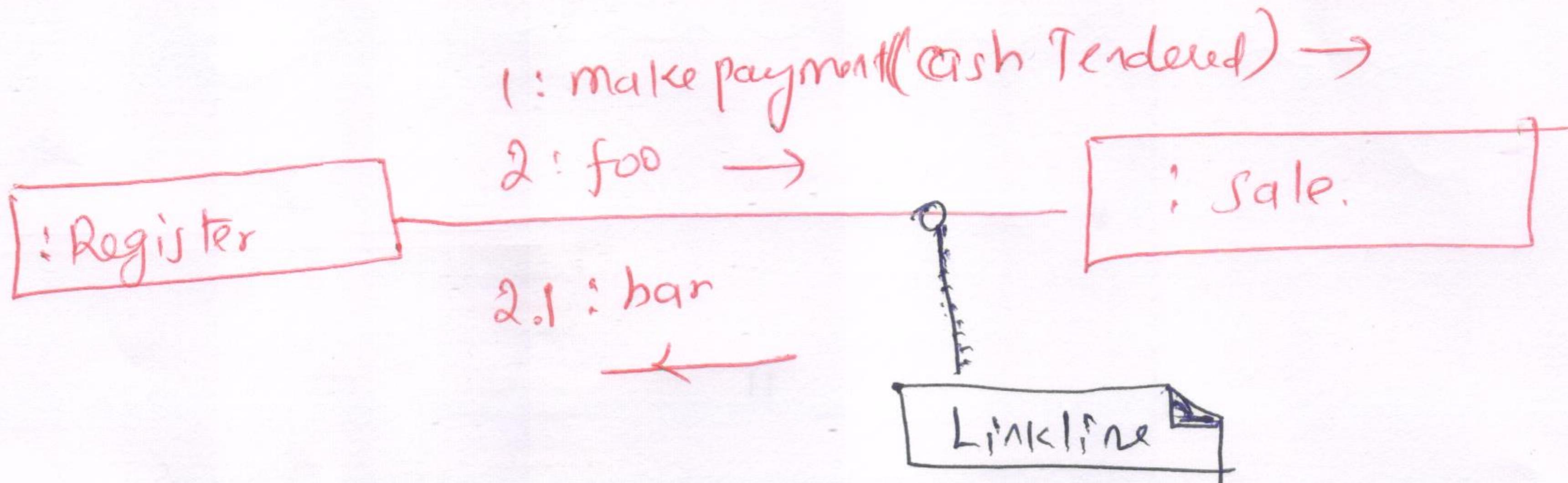
- (i) Links
- (ii) Messages
- (iii) Message to 'self' or 'this'
- (iv) Creation of instances
- (v) Message number sequencing
- (vi) conditional messages
- (vii) Mutually Exclusive Conditional paths
- (viii) Iteration or Looping
- (ix) Iteration over a collection
- (x) Messages to a classes to invoke static(class) methods
- (xi) polymorphic messages and cases
- (xii) Asynchronous and Synchronous calls.

## Links:

- A link is connection path b/w two objects.
- It indicates some form of navigation and visibility b/w the objects is possible.
- a link is an instance of association

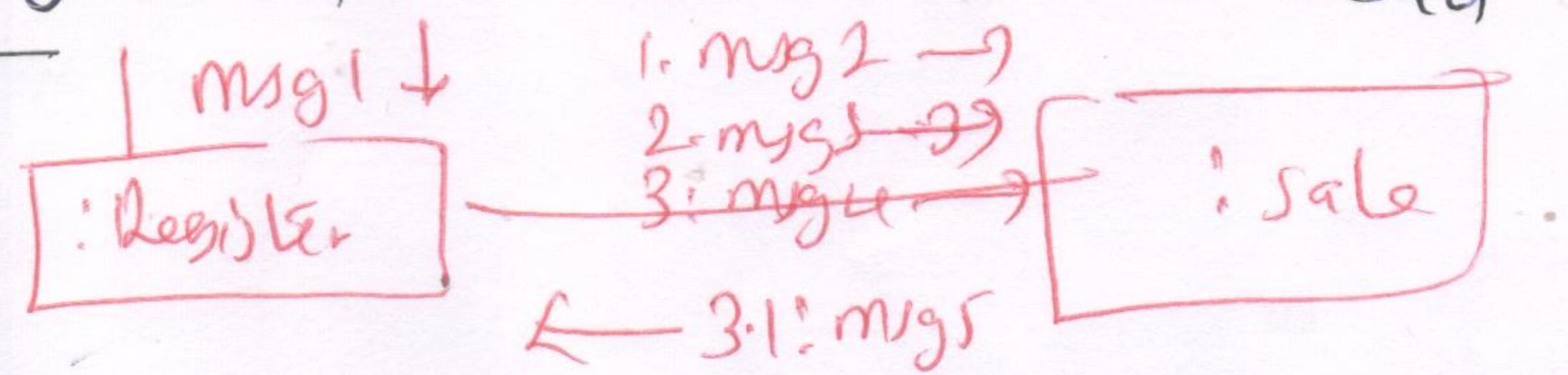
## Example: Linklines

(2)



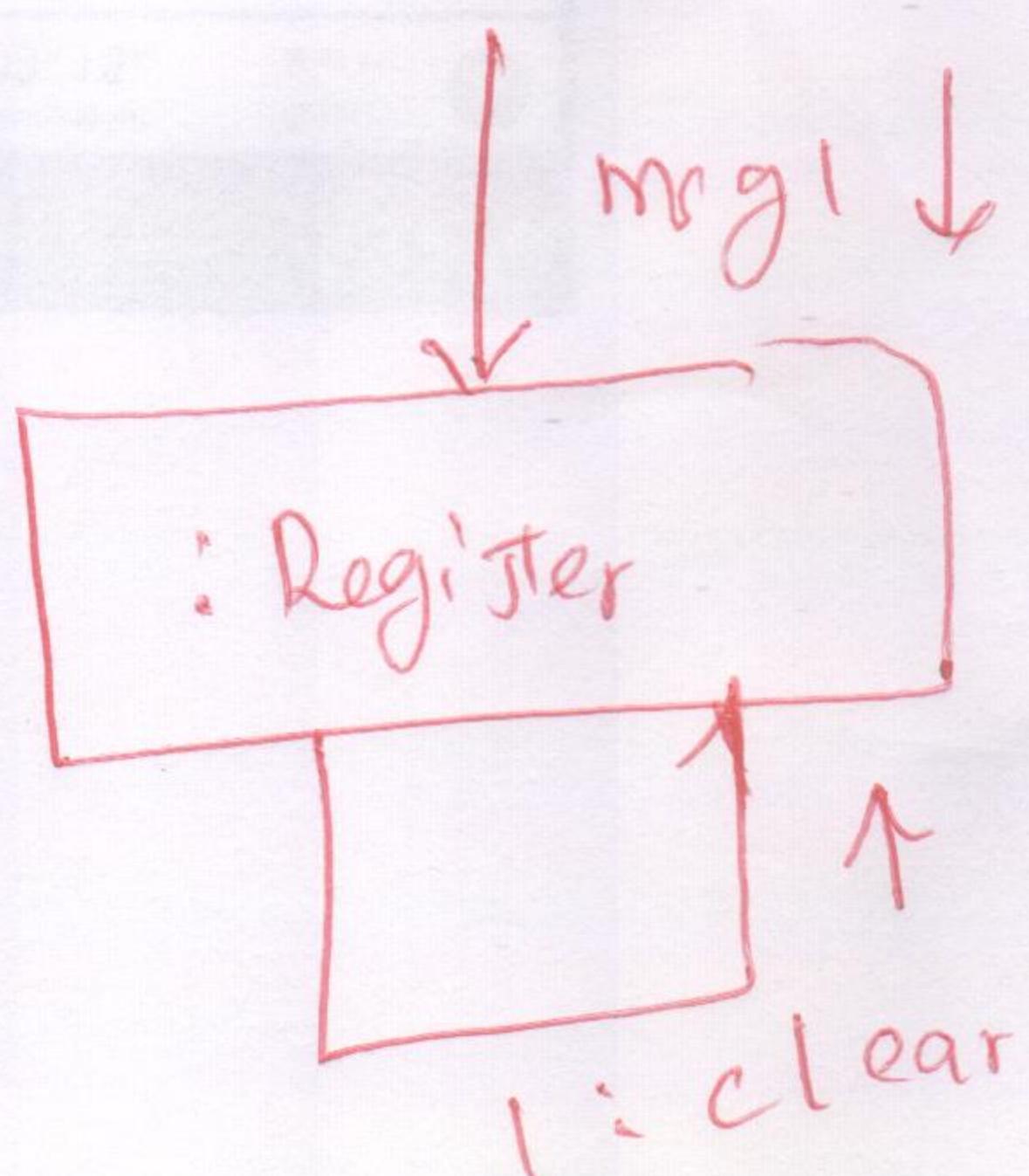
## Messages

- \* Each message between objects is represented with a message expression and small arrow indicating the direction of the message,
- \* A sequence number is added to show the sequential order of messages in the current thread of control.



## Message to "self" or "This"

- \* A message can be sent from an object to itself.
- \* This is illustrated by a link to itself, with messages flowing along the link.



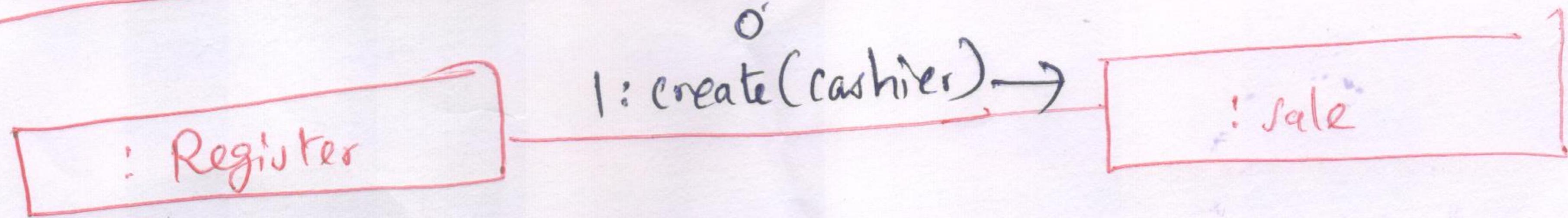
(3)

## Creation of instances

- Any message can be used to create an instance, but in UML, a message named **create** is used for this purpose.
- If another message name is used, the message may be annotated with a **UML stereotype**, **LL Create ??**.
- The create message may include parameters indicating the passing of initial values.
- Furthermore, the **UML Tagged Value {new}** may be optionally added to the lifeline box to highlight the creation.

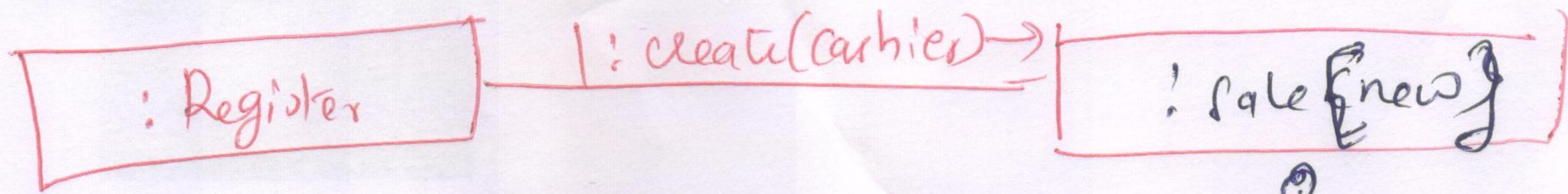
Three ways to show creation in communication

Way 1

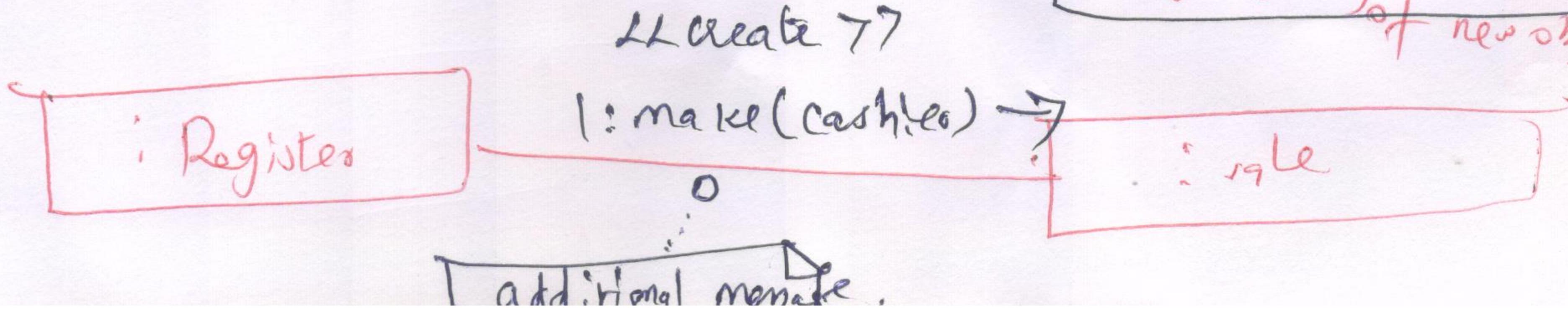


① Create message with optional initializing parameters

Way 2



Way 3



highlighting the creation of new object

! additional message.

(4)

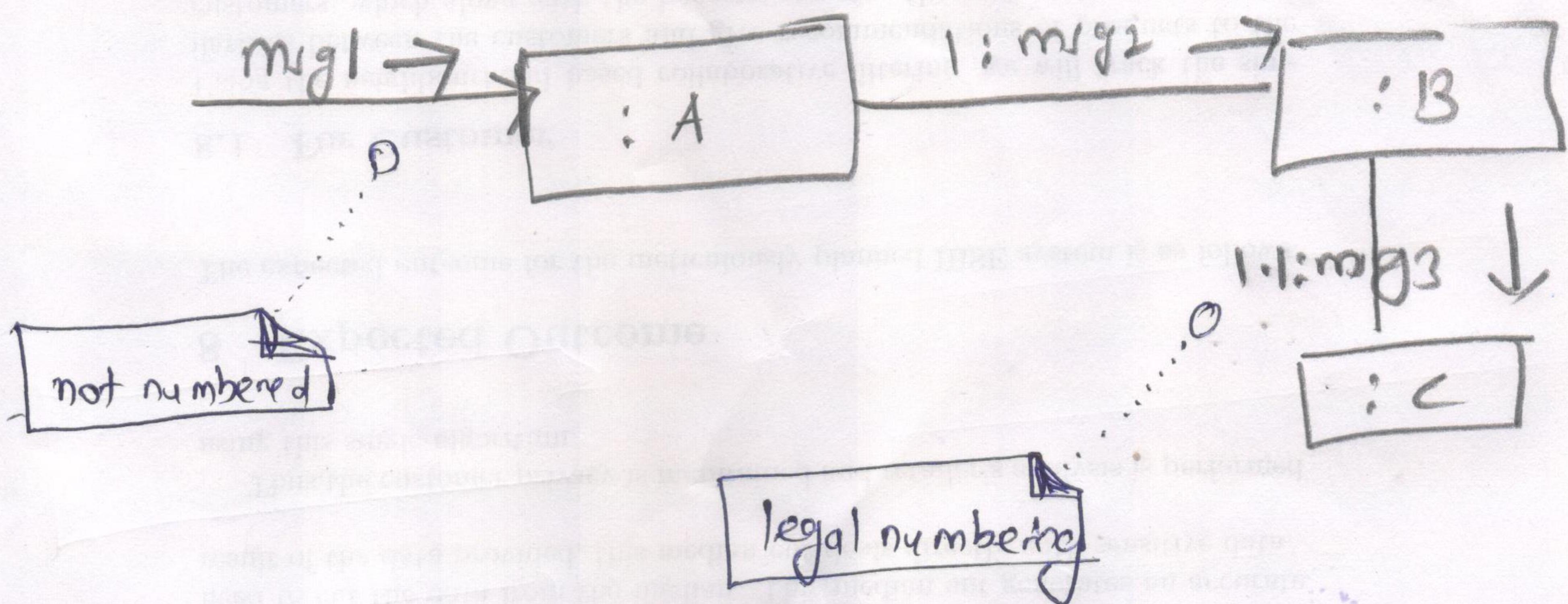
## Message Number Sequencing

The order of messages is illustrated with sequence numbers

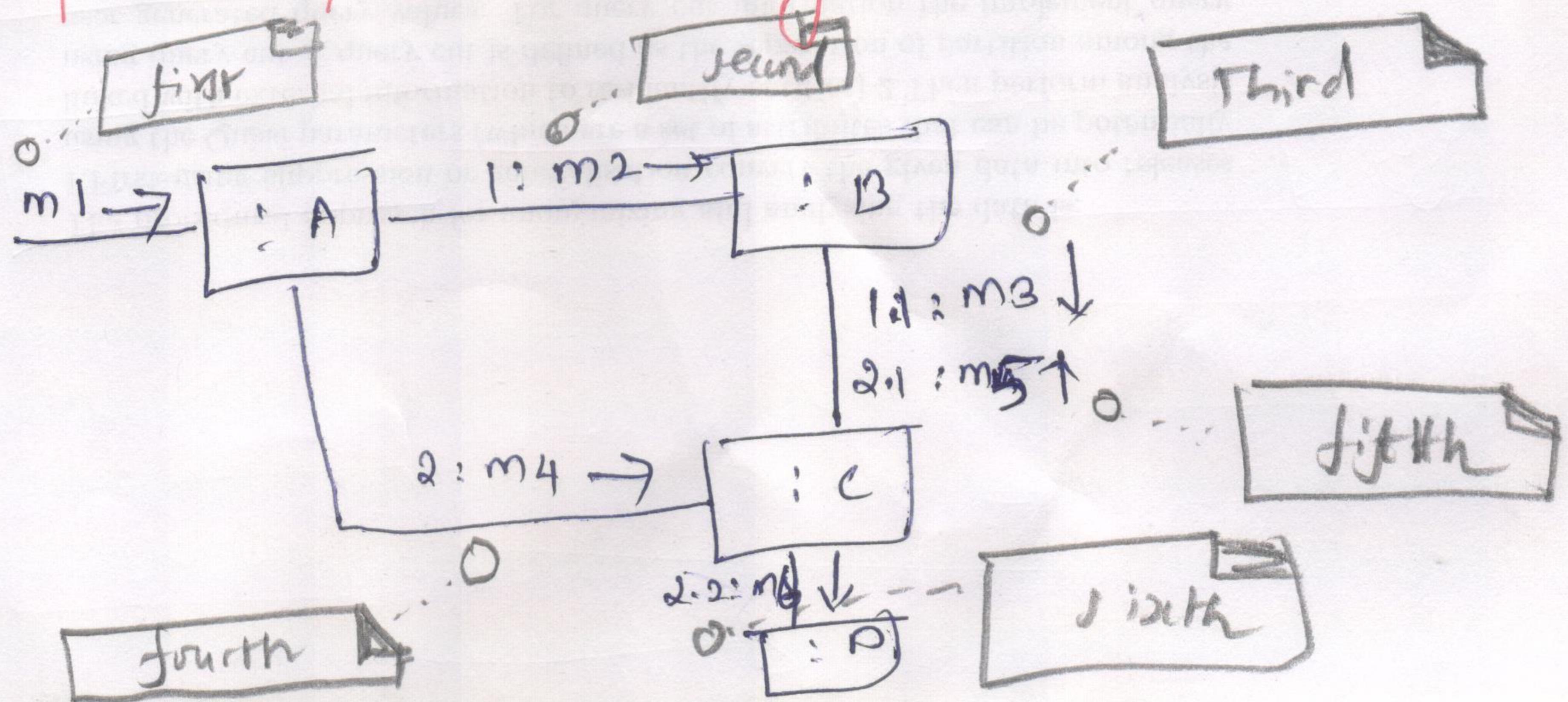
① The first message is not numbered

② The order and nesting of subsequent messages is shown with a legal numbering scheme in which nested messages have a number appended to them.

Fig: sequence numbering



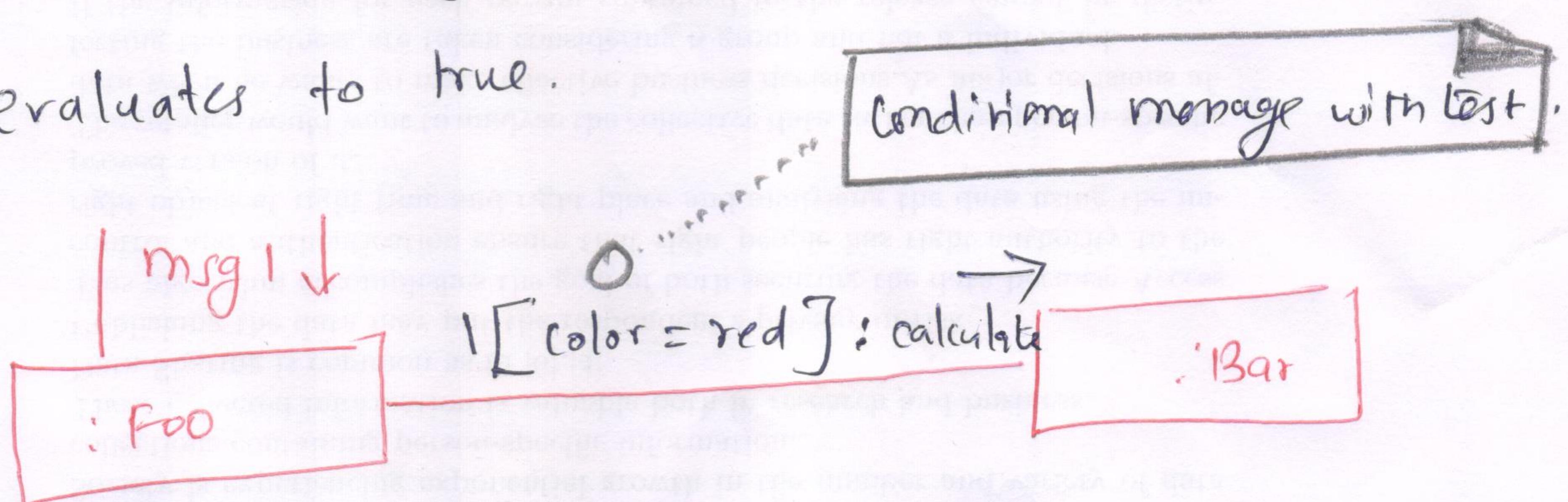
### Complex Sequence numbering



⑤

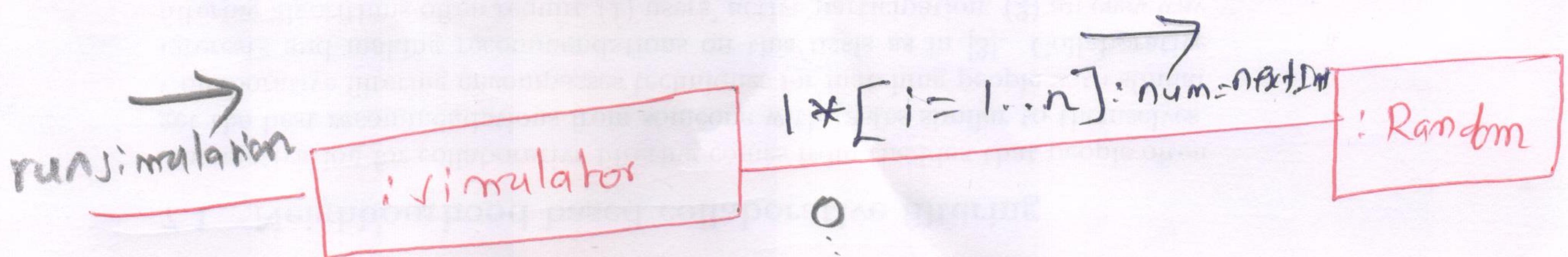
## Conditional messages

- Conditional message is shown by sequence number with a conditional message clause in square brackets, similar to an iteration clause.
- The message is only sent if the clause evaluates to true.



## Mutually Exclusive Conditional Paths

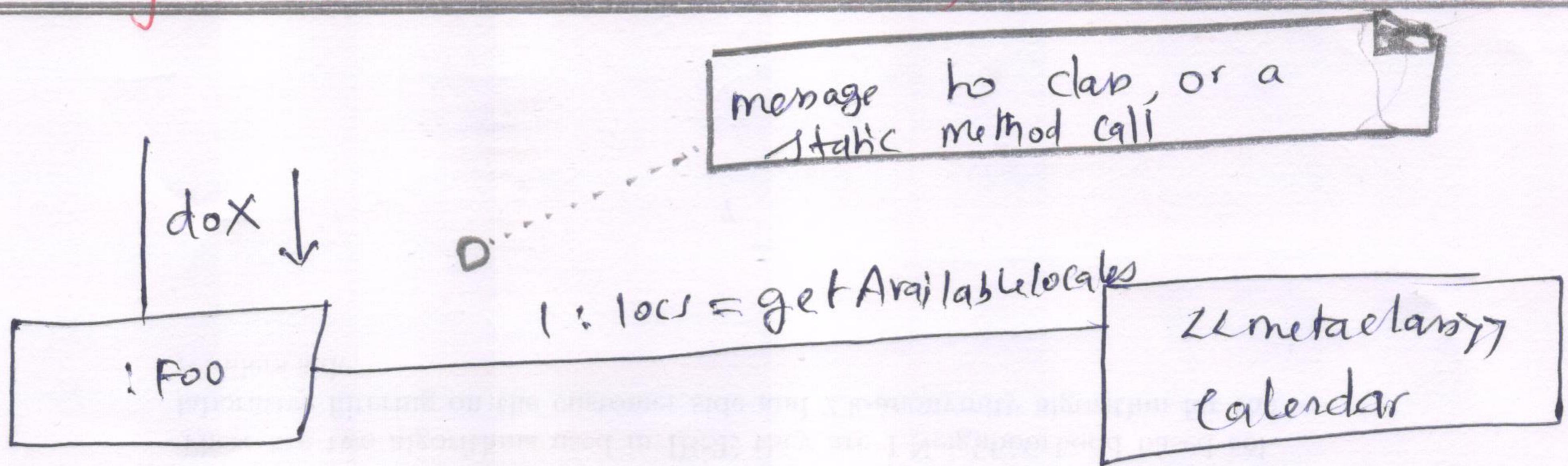
- Fig states that either 1a or 1b could execute after msg1. Both are sequence number 1 since either could be the first internal message.
- 1b.1 is nested message within 1b.



Iteration is indicated with a \* and an optional iteration clause following the sequence number.

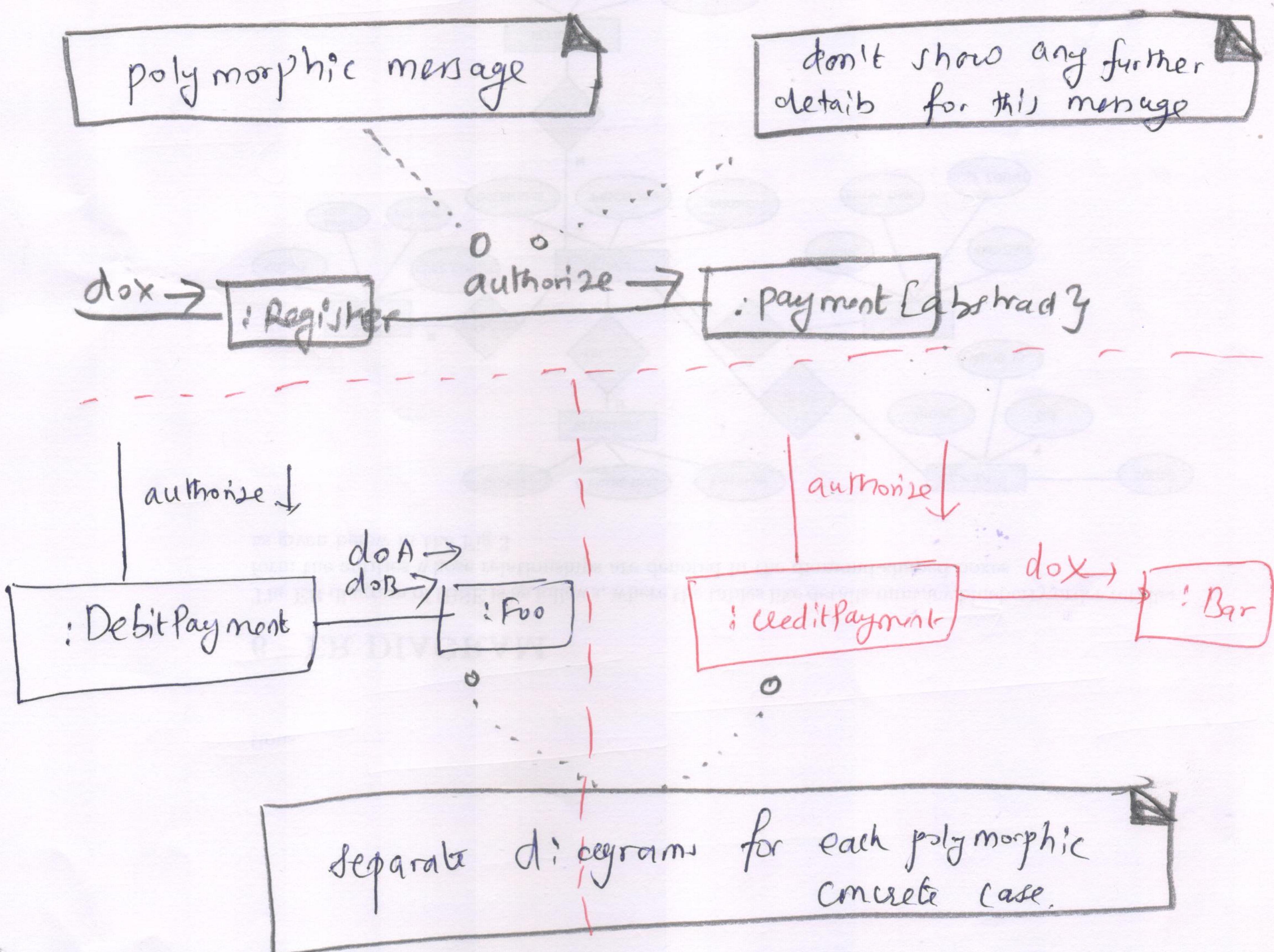
(6)

## Messages to a classes to invoke static (class) methods



## Poly morphic Messages and cases

- As in sequence diagram Case multiple communication diagrams can be used to show each <sup>concrete</sup> poly morphic case



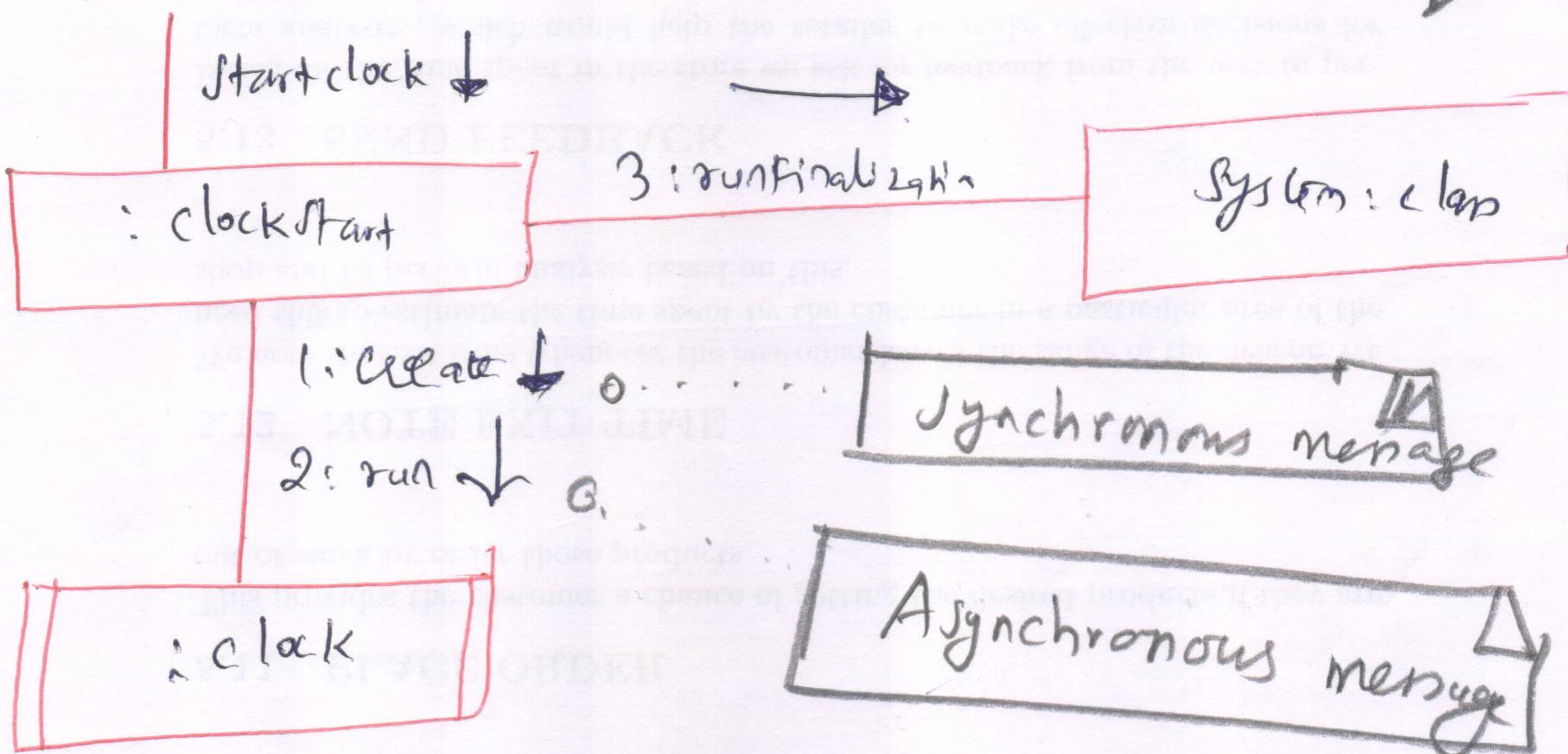
⑦

## Asynchronous and Synchronous Calls

→ Asynchronous calls are shown with stick arrow



→ Synchronous calls are shown with filled arrow



## 2.2. DIGITAL RECORDING ADVICE