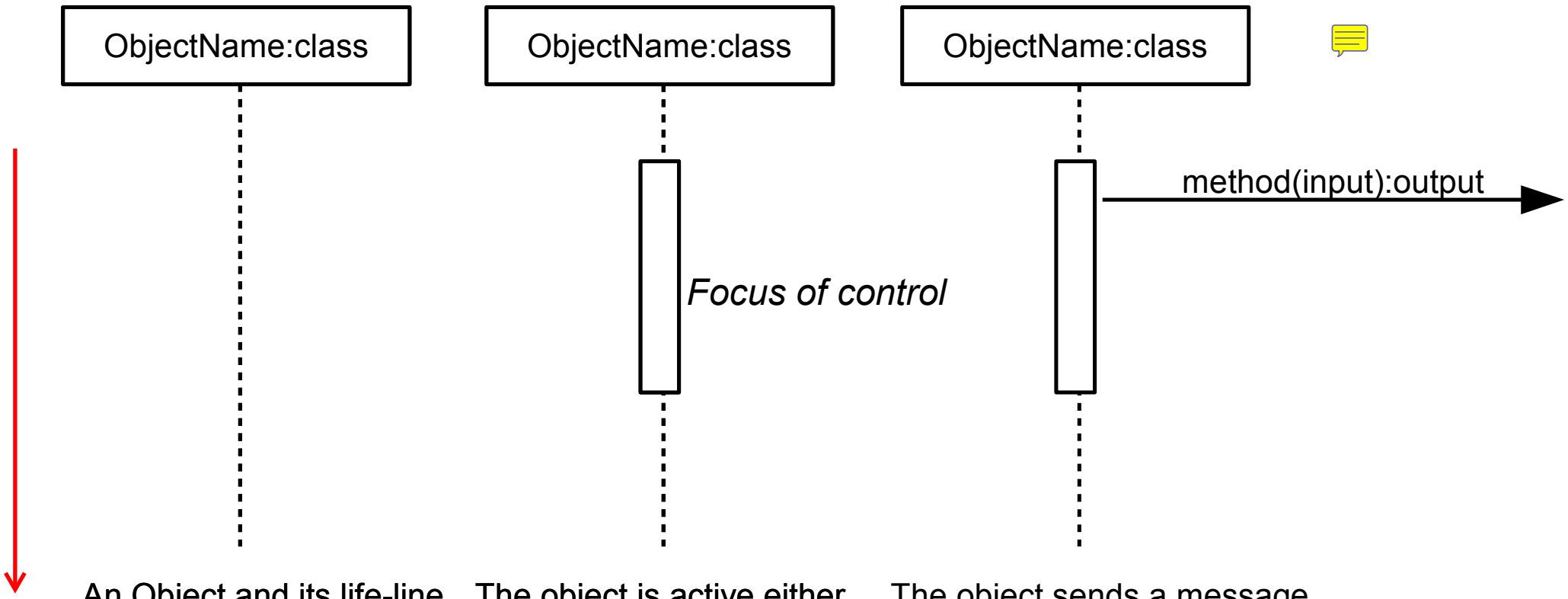
Sequence Diagrams

Modeling behavior

Motivation

- Definition: shows how processes operate with one another and in what order
- Used in: technical specification on the level of class-objects
  - To describe the realization of a use case
- Have less details than code
- Can be created by non-developers
- Provide parallel inspection of objects (visualization)

How they work



An Object and its life-line (instantiated, consumes memory)

The object is active either Processing or waiting for something

The object sends a message And receives a return value

The „message“ is a methods that is called or just a message
Of the output is written behind the call, a return arrow can be spared

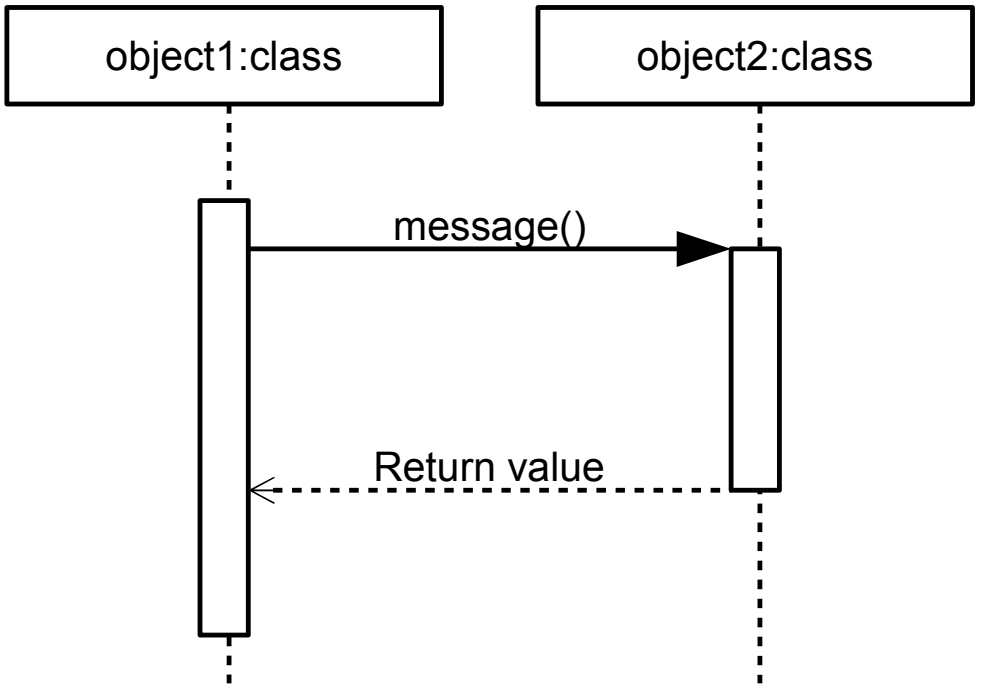
Modeling Tasks

- Model a behavior and present it
  - A packet is shipped from Mannheim to New York
    - Delivery using postal office, regional repository, airport
  - A round of texal holdem poker
    - Cards are issued, Bets were set & showdown
  - Order a product from a webshop
    - Search,checkout, declare shipping details

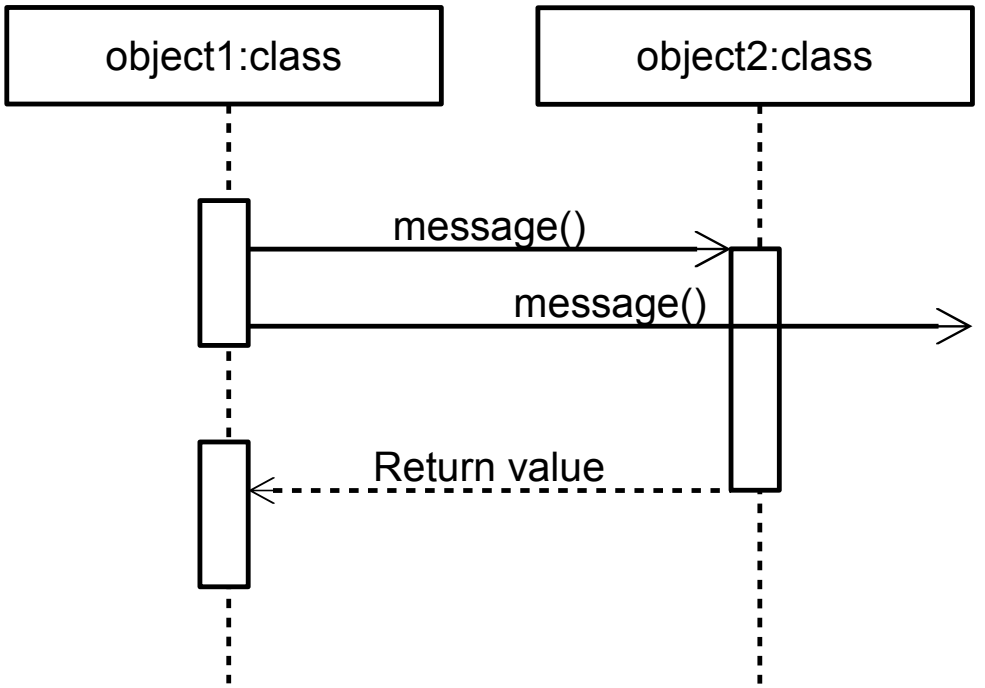
Benefits

- Helps to analyze communication infrastructure
  - Memory waste
  - Failure tolerance
  - System stability
  - Ease of use

Asynchronous VS Synchronous



Synchronous call: the called objects cannot deliver immediately, object1 "blocks" until object2 returns

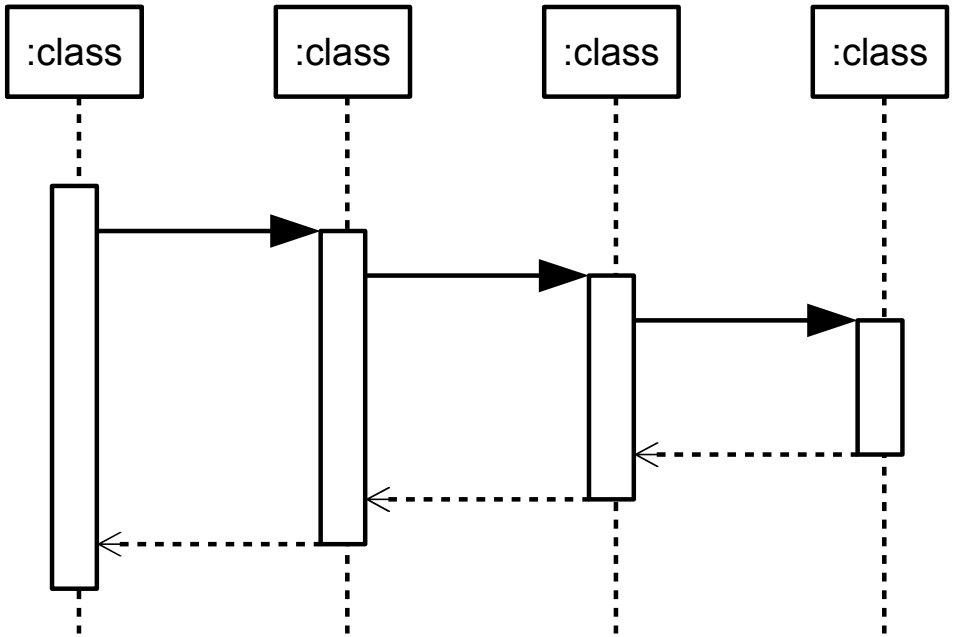


Asynchronous call: the called object is not waiting for the return value, but does something Else or goes "sleeping" until the answer returns

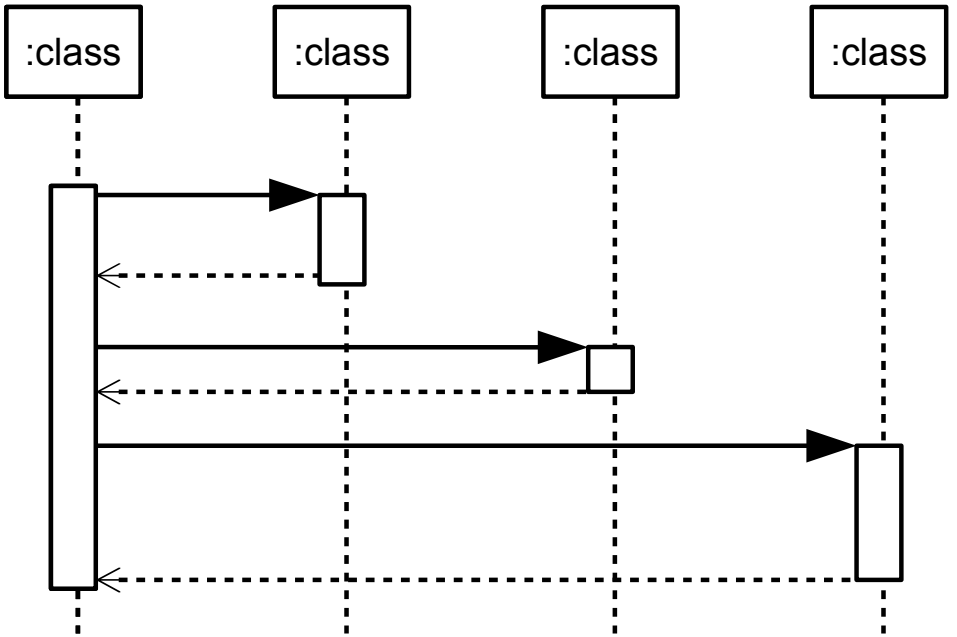
one single action between send and receive

multiple actions between send & receive

Communication Architecture

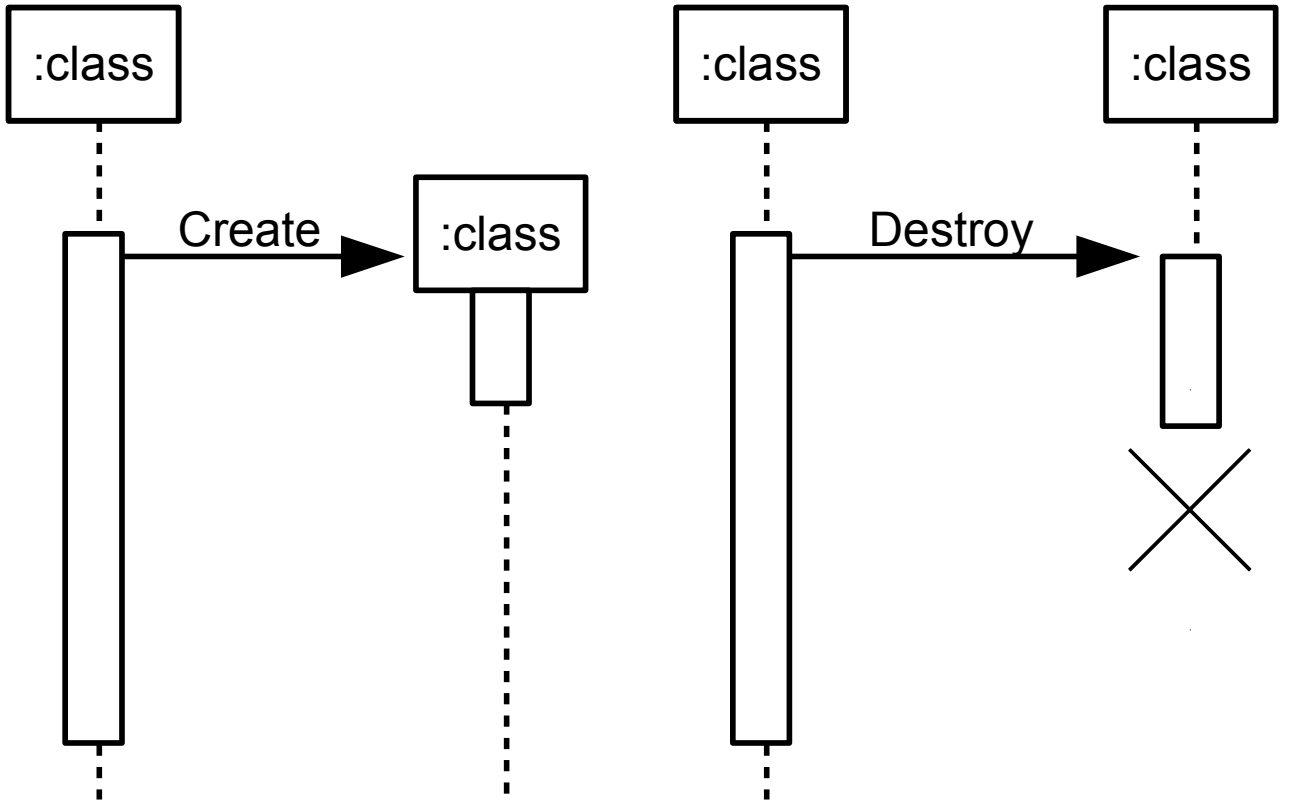


Centralized / cascading



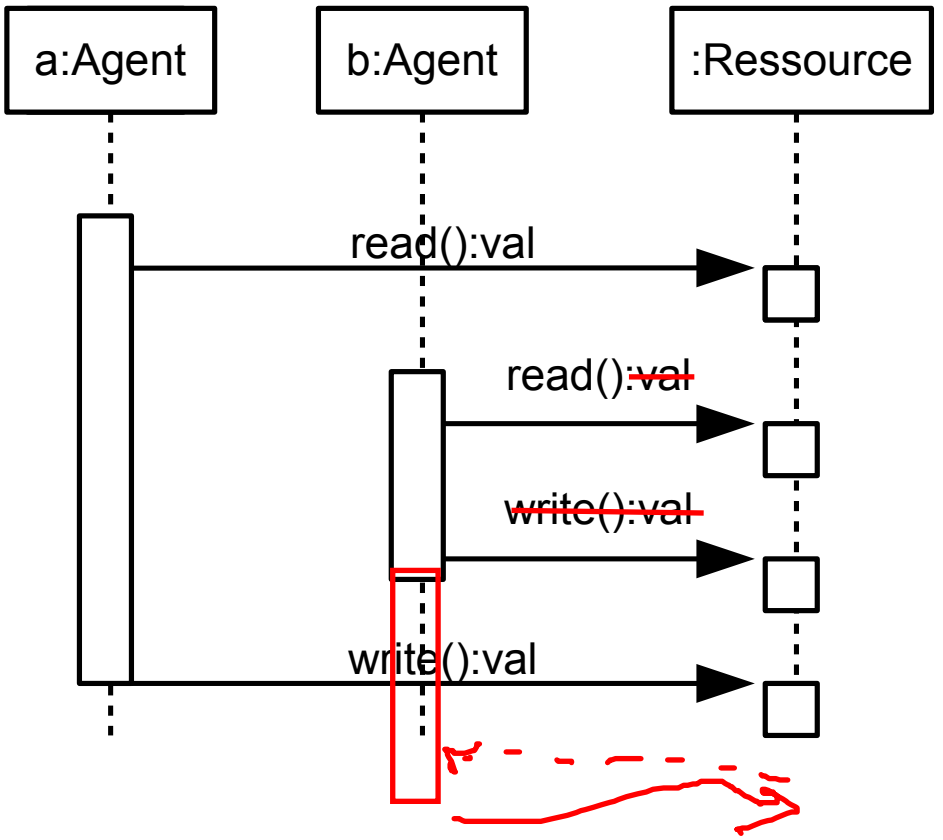
Distributed, client/ Server

Creation & Destruction



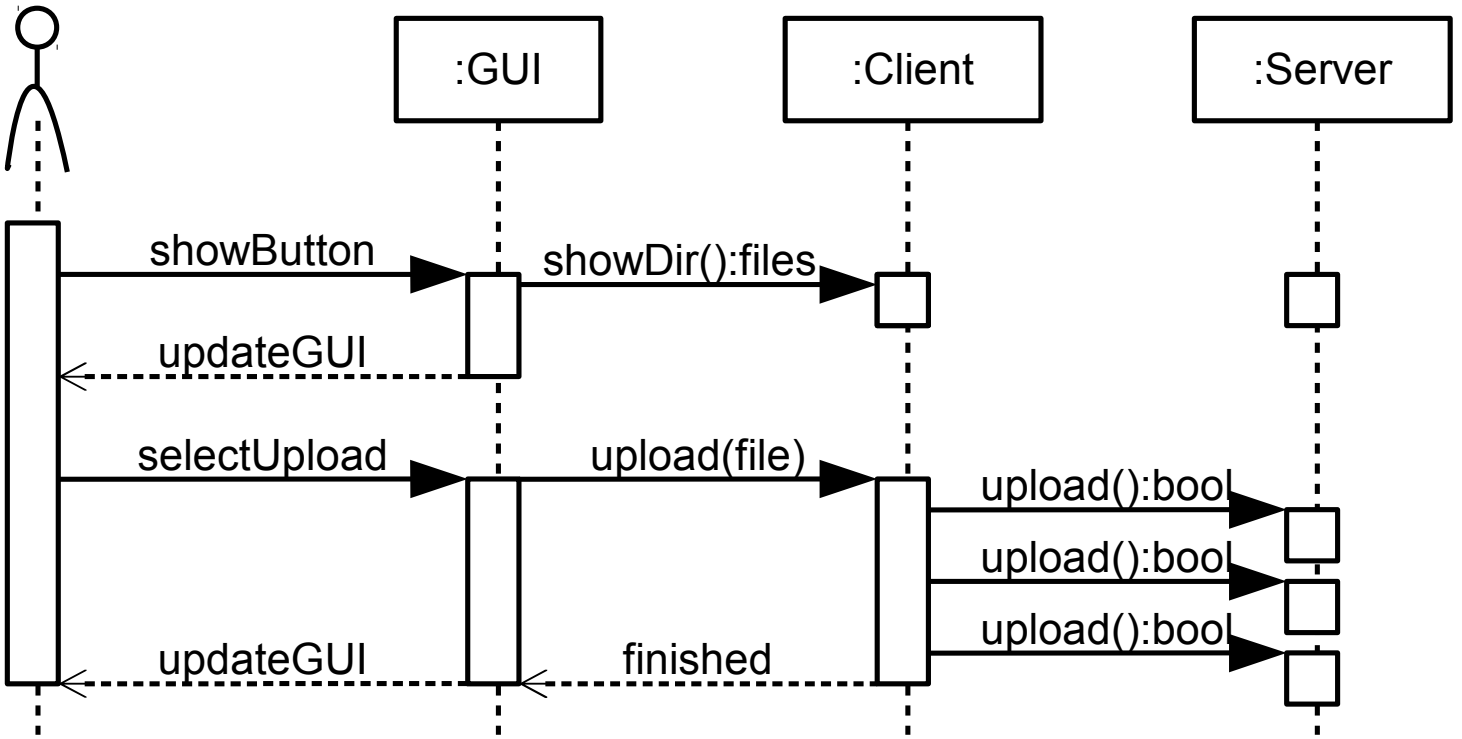
Task shared ressource

- What is problematic with the left communication
- Suggest solutions to this problem as sequence diagram



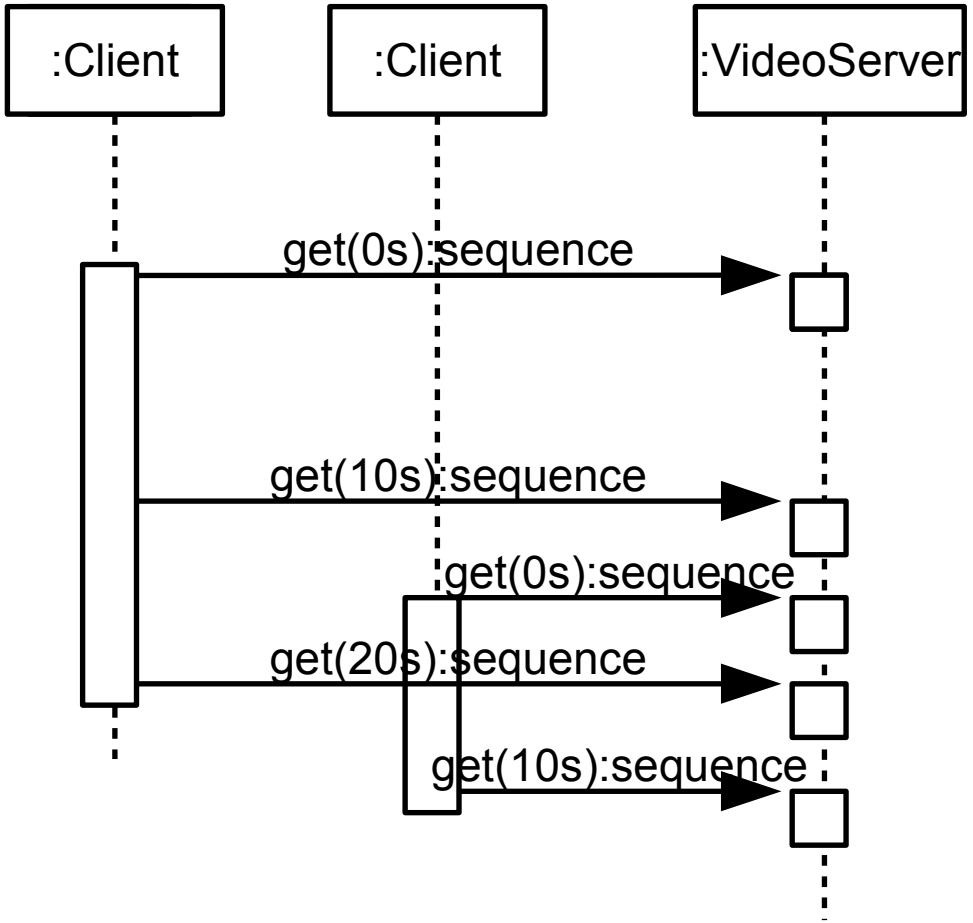
Task GUI Client-Server

- What is problematic with this approach
- Again make suggestions to improve usability / functionality



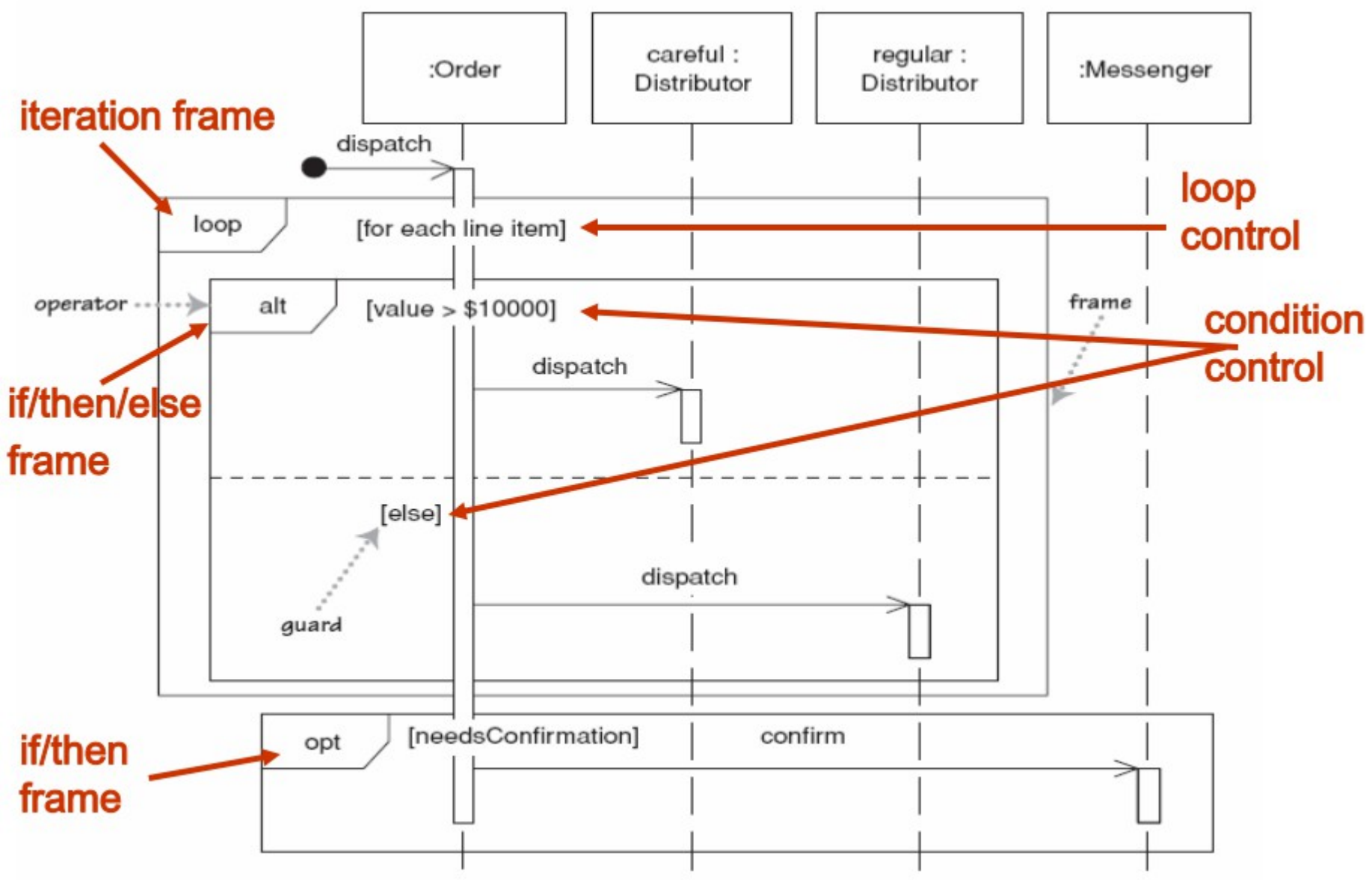
Task Videostreaming

- Is that approach good for
  - Rights control (DRM)
  - Distribution speed & broadcasting
  - Reliability and availability
- In case its not, make suggestions



get(posotion_seconds) returns a sequence of 10 seconds from passed time pointer

Conditionals & loops



http://www.cs.washington.edu/education/courses/cse403/11sp/lectures/lecture09-uml2.pdf by Michael Ernst

Task

Create a Sequence diagram of your game's most challenging Use Case

- Identify required class-objects
- Model the interaction between those objects and describe their "communcation"