



Modellare il comportamento



Diagramma di sequenza

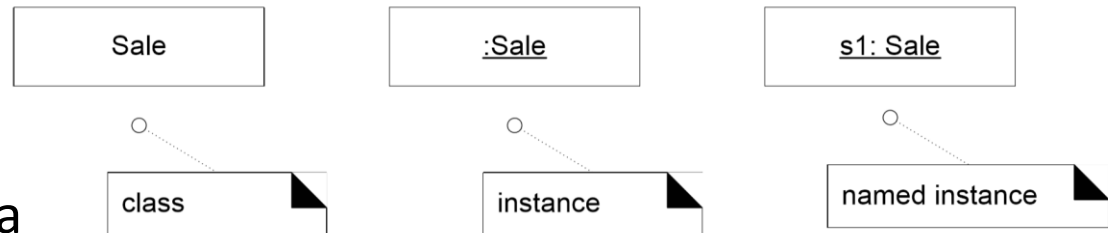
- Mostra l'interazione tra oggetti con enfasi sulla sequenza dei messaggi
- Documenta, dal punto di vista del software, il comportamento di un gruppo di oggetti che realizzano un singolo scenario di utilizzo del sistema

Partecipanti in un diagramma di sequenza



- Partecipante

- Oggetto rappresentato come un rettangolo in cima a una linea tratteggiata



- Linea di vita

- Linea tratteggiata
- Rappresenta lo scorrere del tempo



L'oggetto qui è anonimo

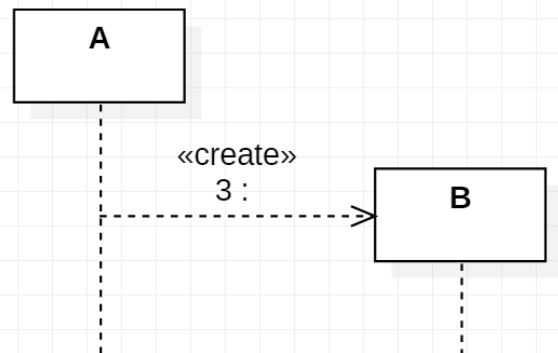
Oggetto partecipante

Linea di vita

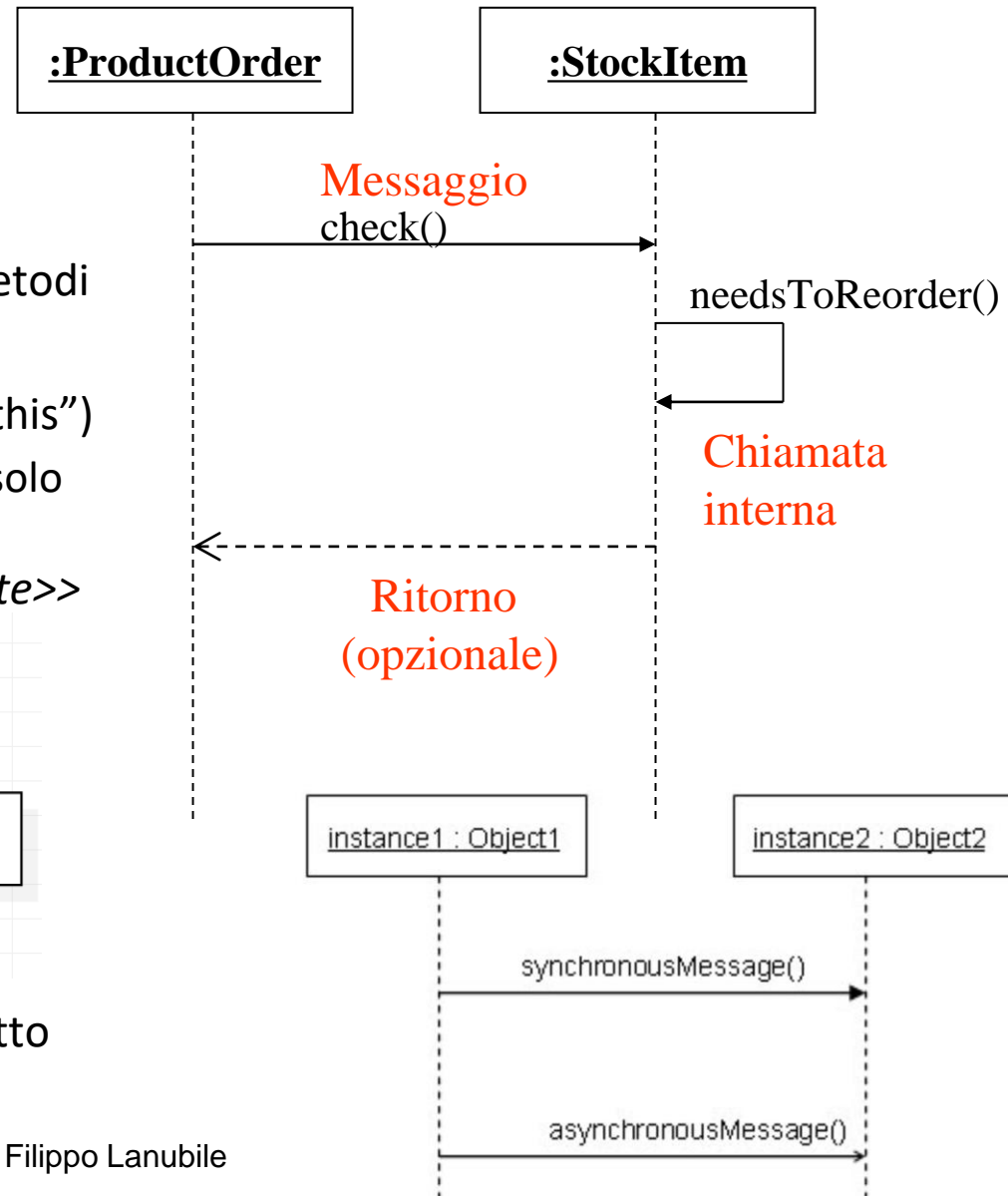
Messaggi in un diagramma di sequenza



- Messaggi
 - Comunicazione tra oggetti
 - Realizzate come chiamate di metodi
- Tipi speciali di messaggi
 - Chiamata interna (all'oggetto "this")
 - Ritorno da chiamata (si mostra solo se aggiunge informazione)
 - Creazione di un oggetto `<<create>>`



- Distruzione esplicita di un oggetto `<<destroy>>`

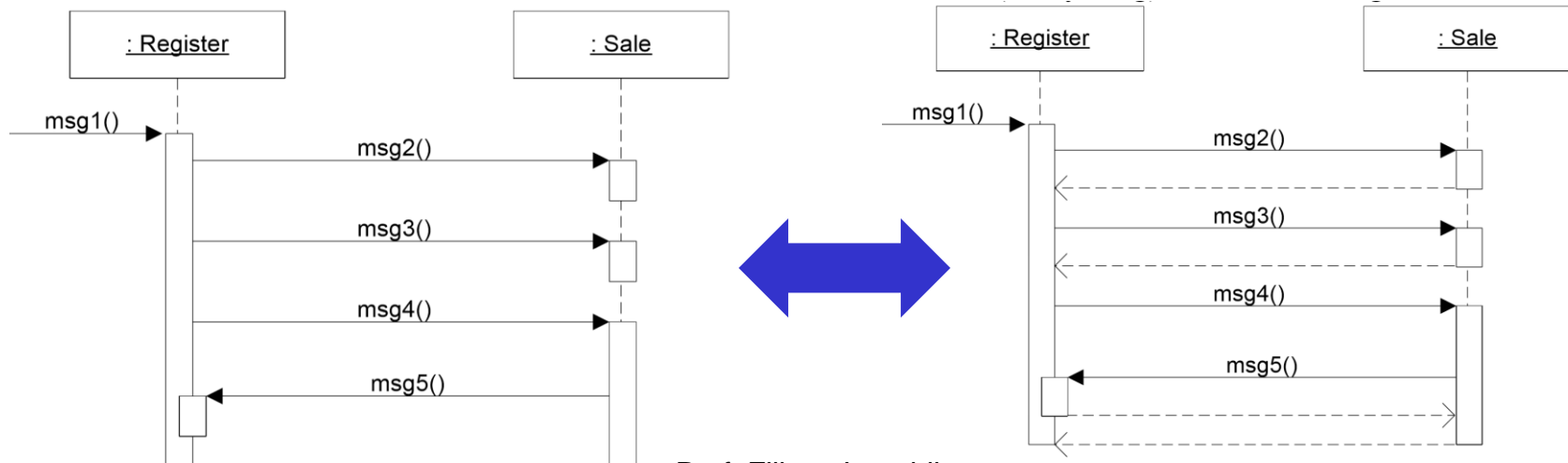
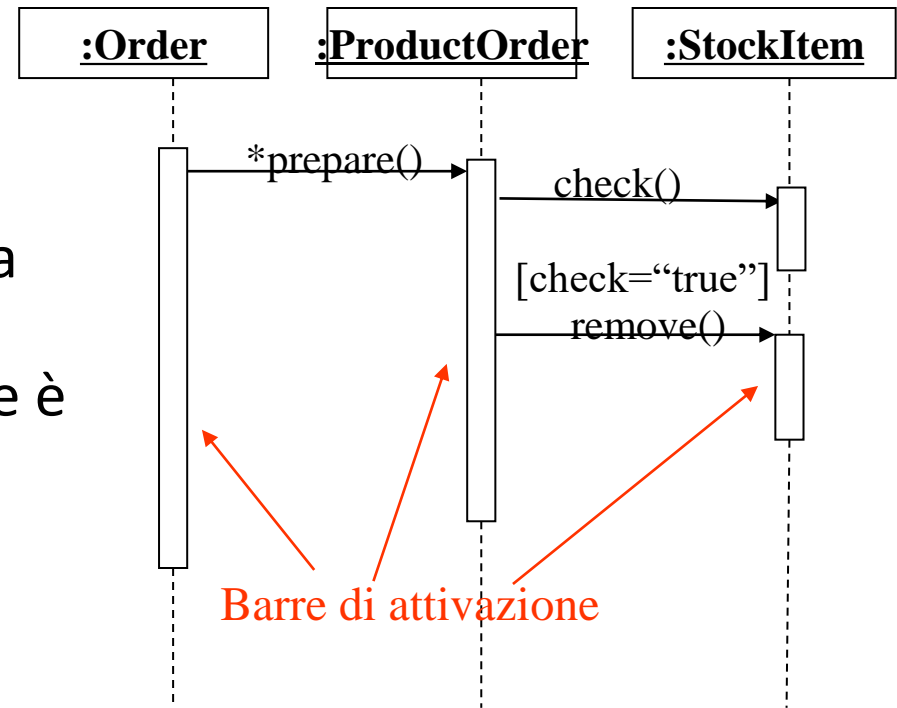


Barre di attivazione in un diagramma di sequenza

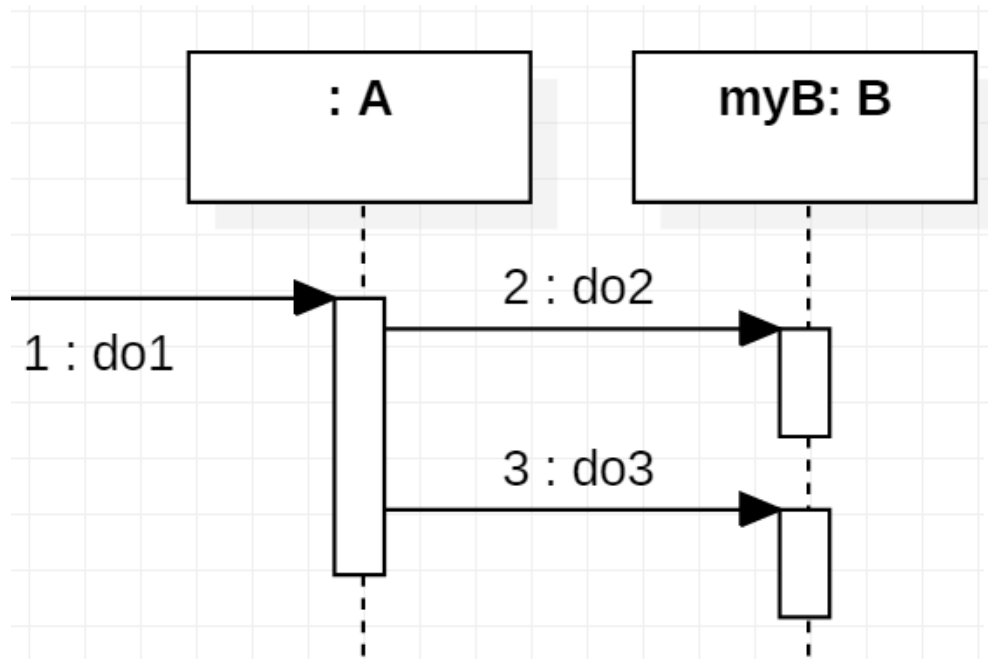


Barra di attivazione

- Rettangolo sottile posto sulla linea di vita di un oggetto
- Indica quando il partecipante è attivo nell'interazione
- Rende ridondante l'uso di linee di restituzione del controllo (return line)



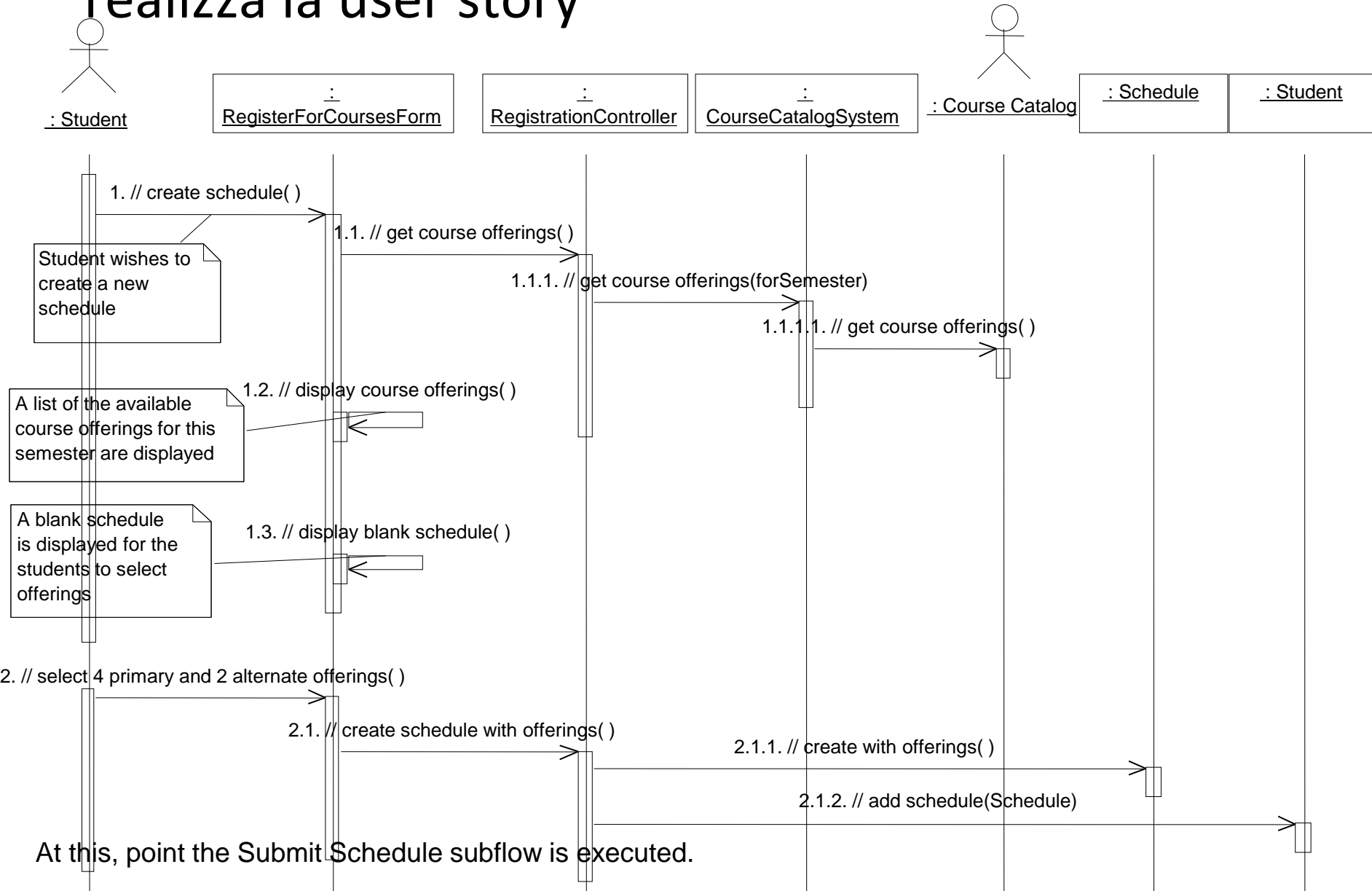
Esempio di diagramma di sequenza e codice Java corrispondente



```
public class A {
    private B myB = ...;

    public void do1() {
        myB.do2();
        myB.do3();
    }
}
```

Esempio di diagramma di sequenza che realizza la user story

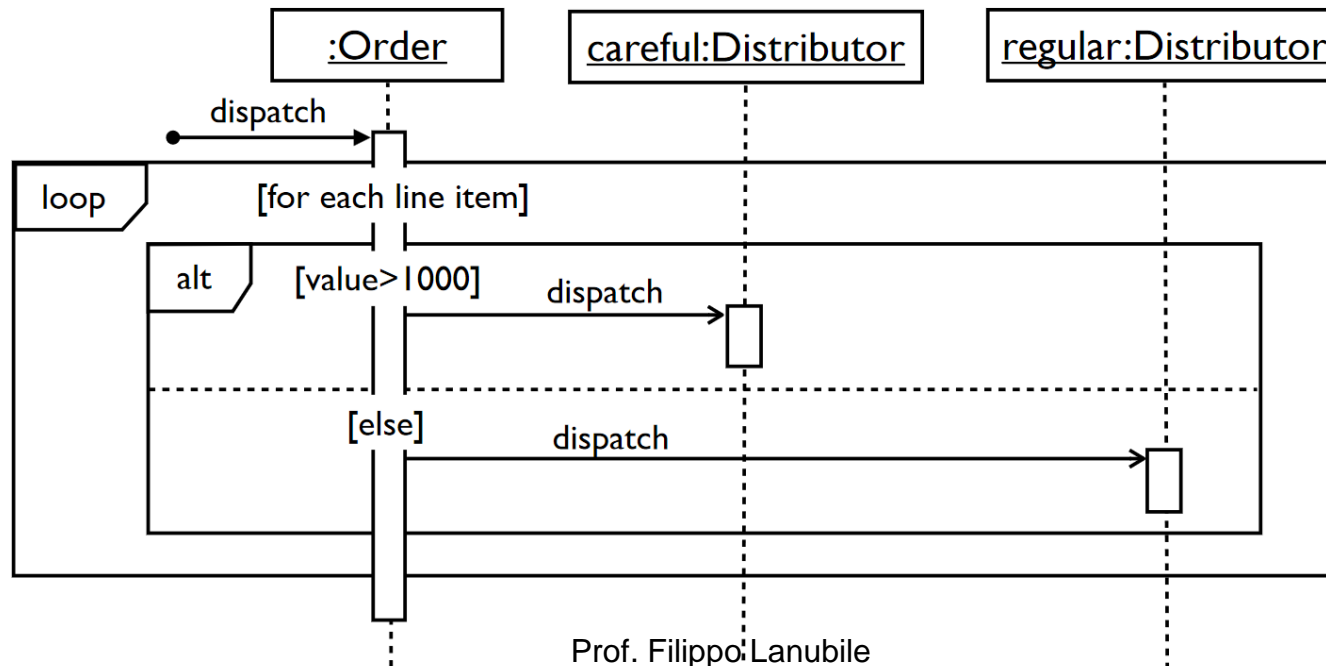




Strutture di controllo in un diagramma di sequenza: **UML 2**

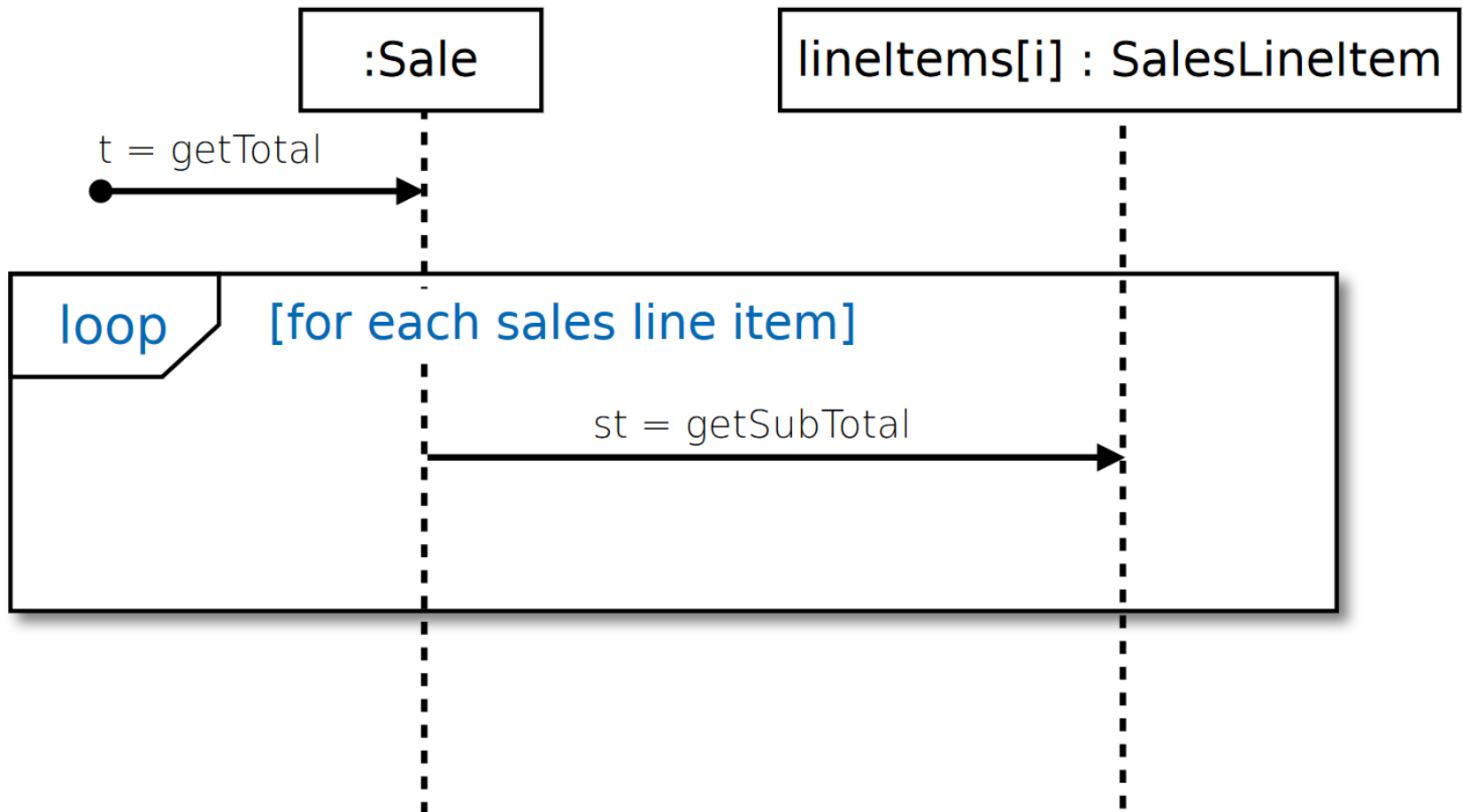
Frame di interazione: una cornice intorno a una parte del diagramma di sequenza

- if → (**opt**) [condition]
- if/else → (**alt**) [condition], separati da una linea tratteggiata orizzontale
- loop → (**loop**) [condizione o elementi su cui iterare]



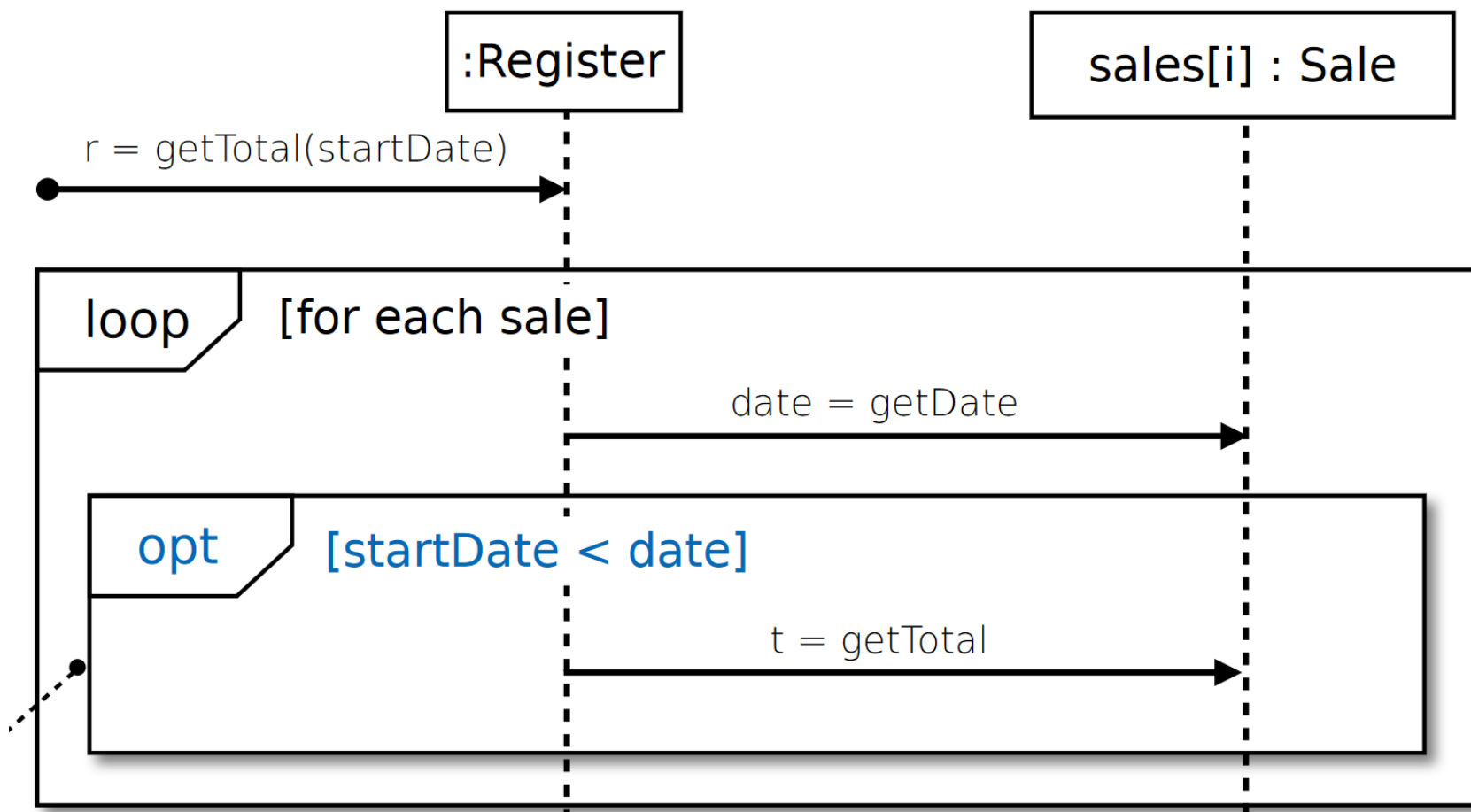


Esempio loop



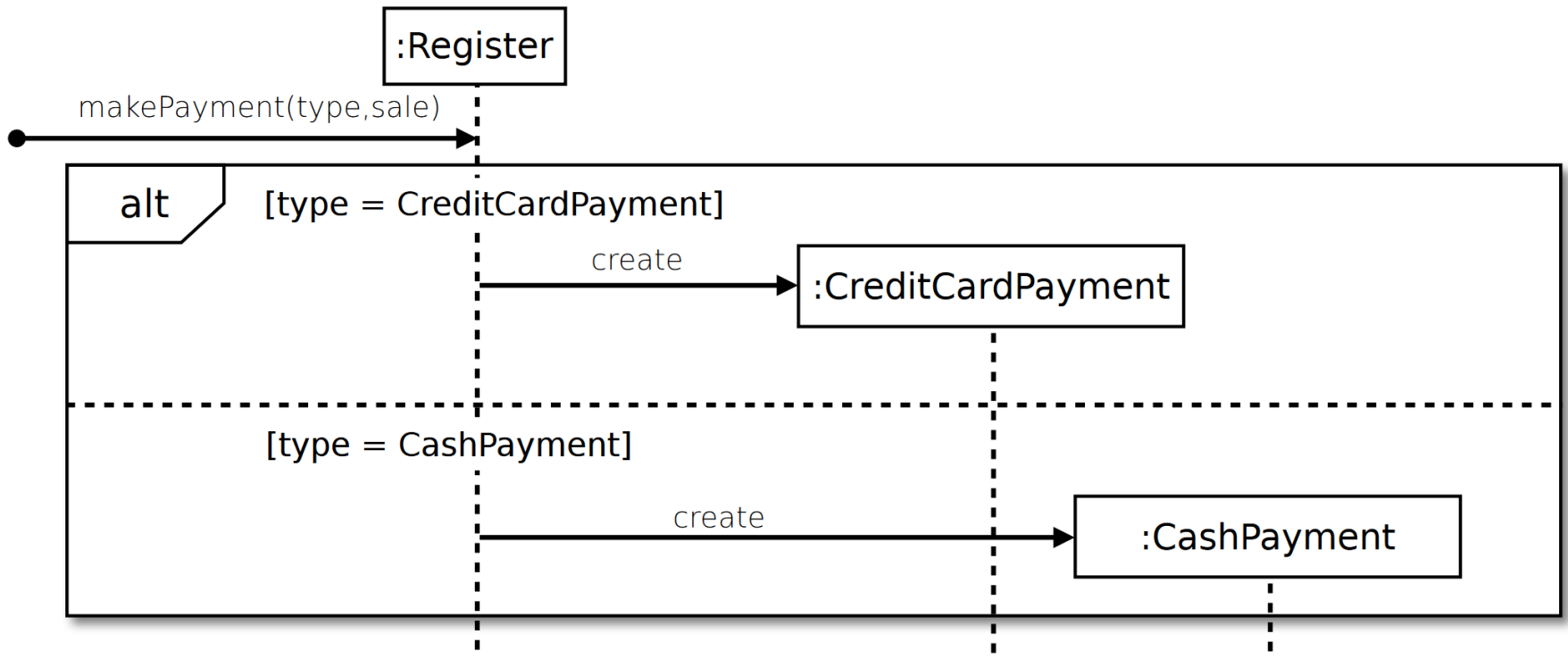


Esempio **opt**





Esempio alt





Il caso della biblioteca

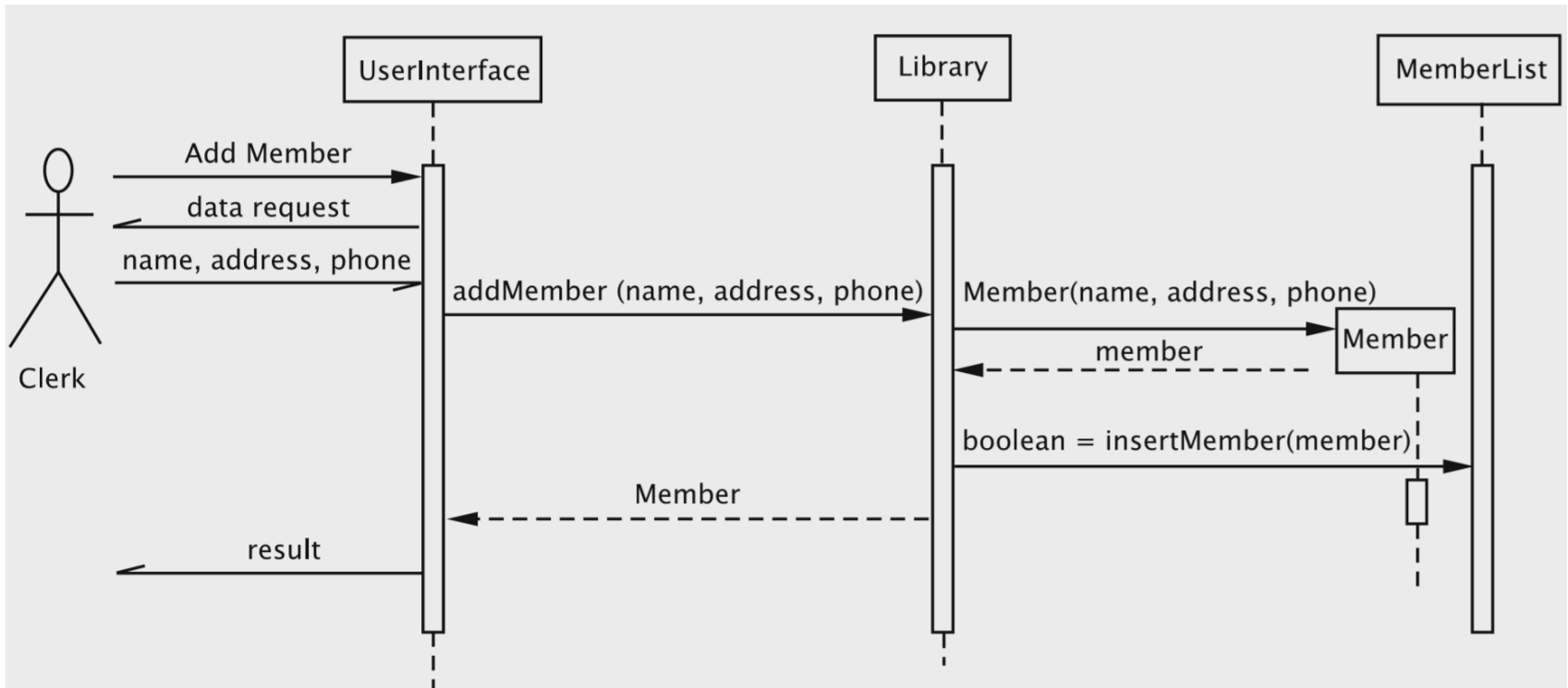
User stories

- Adding a new member
- Adding books
- Issuing books
- Returning books
- Removing books
- Placing a hold

Tratto da *Object-Oriented Analysis, Design and Implementation*,
chapter 7 Design and Implementation

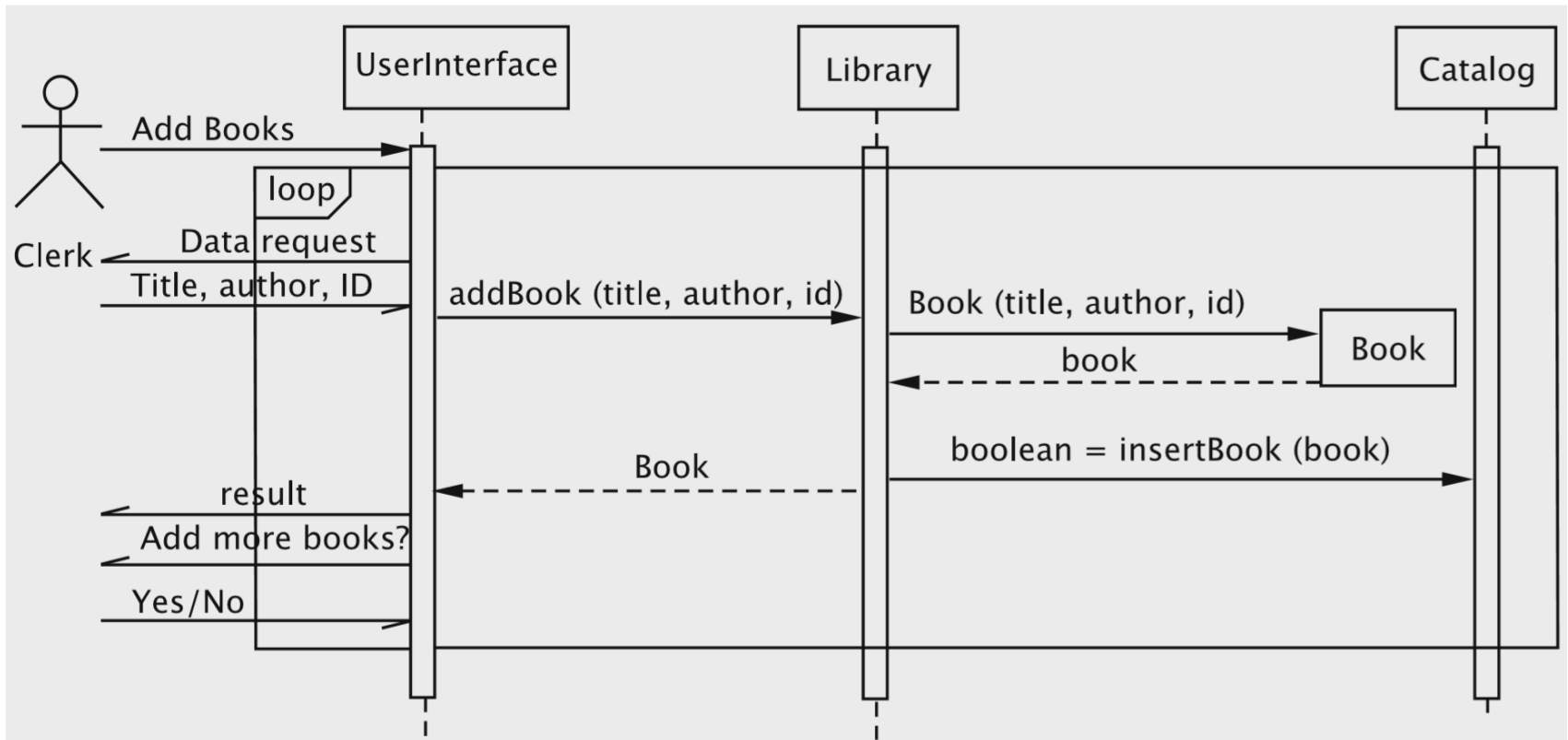


Adding a new member



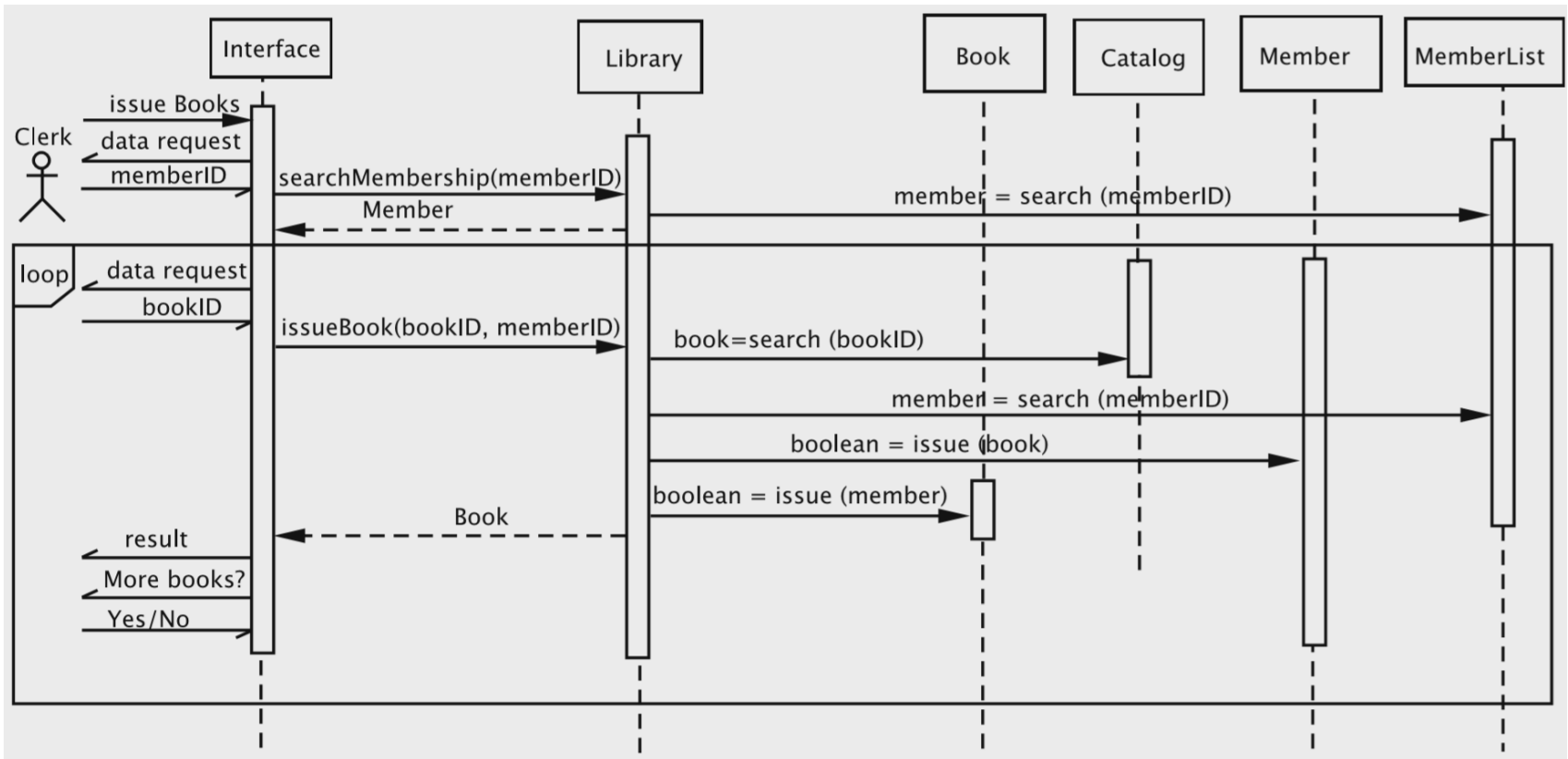


Adding books



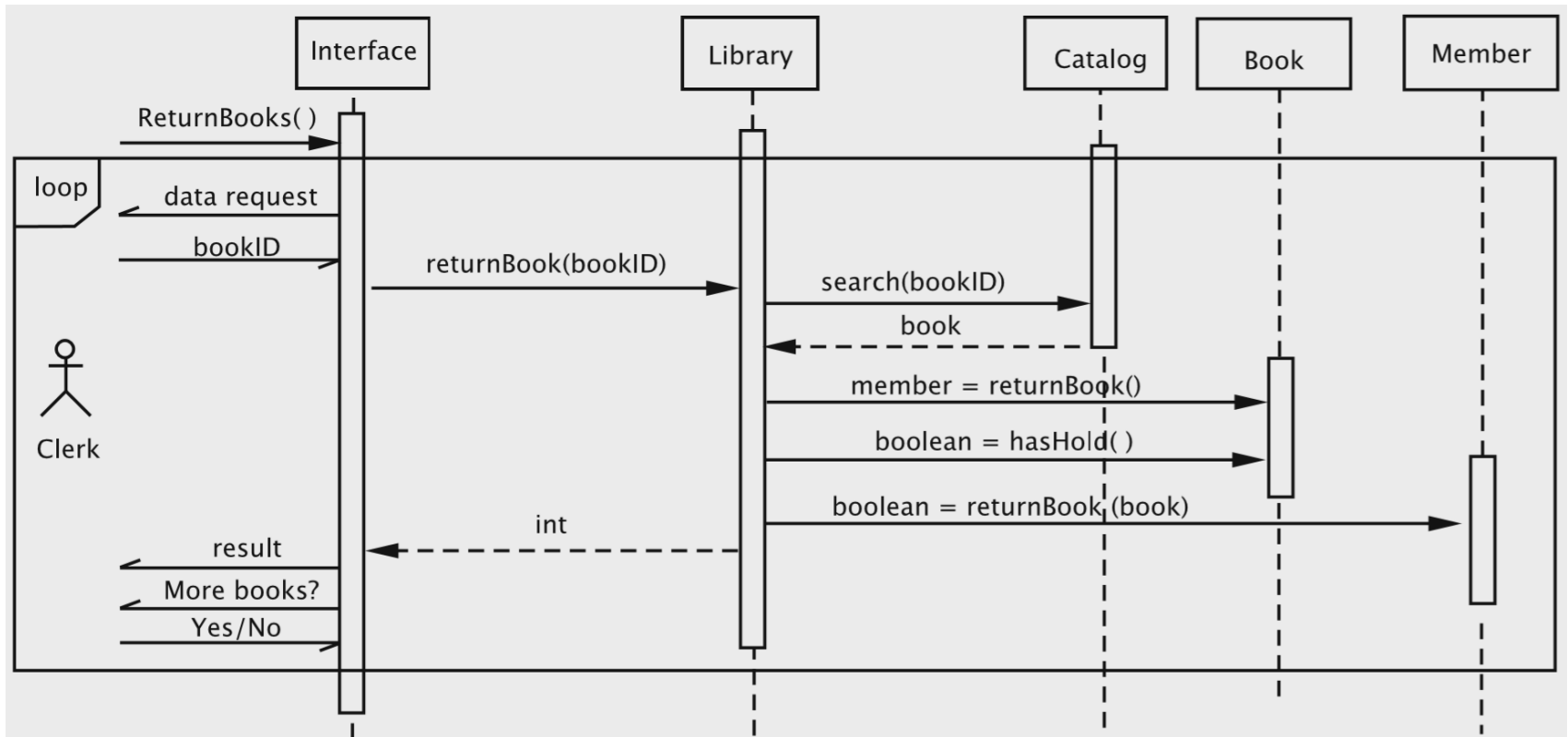


Issuing books



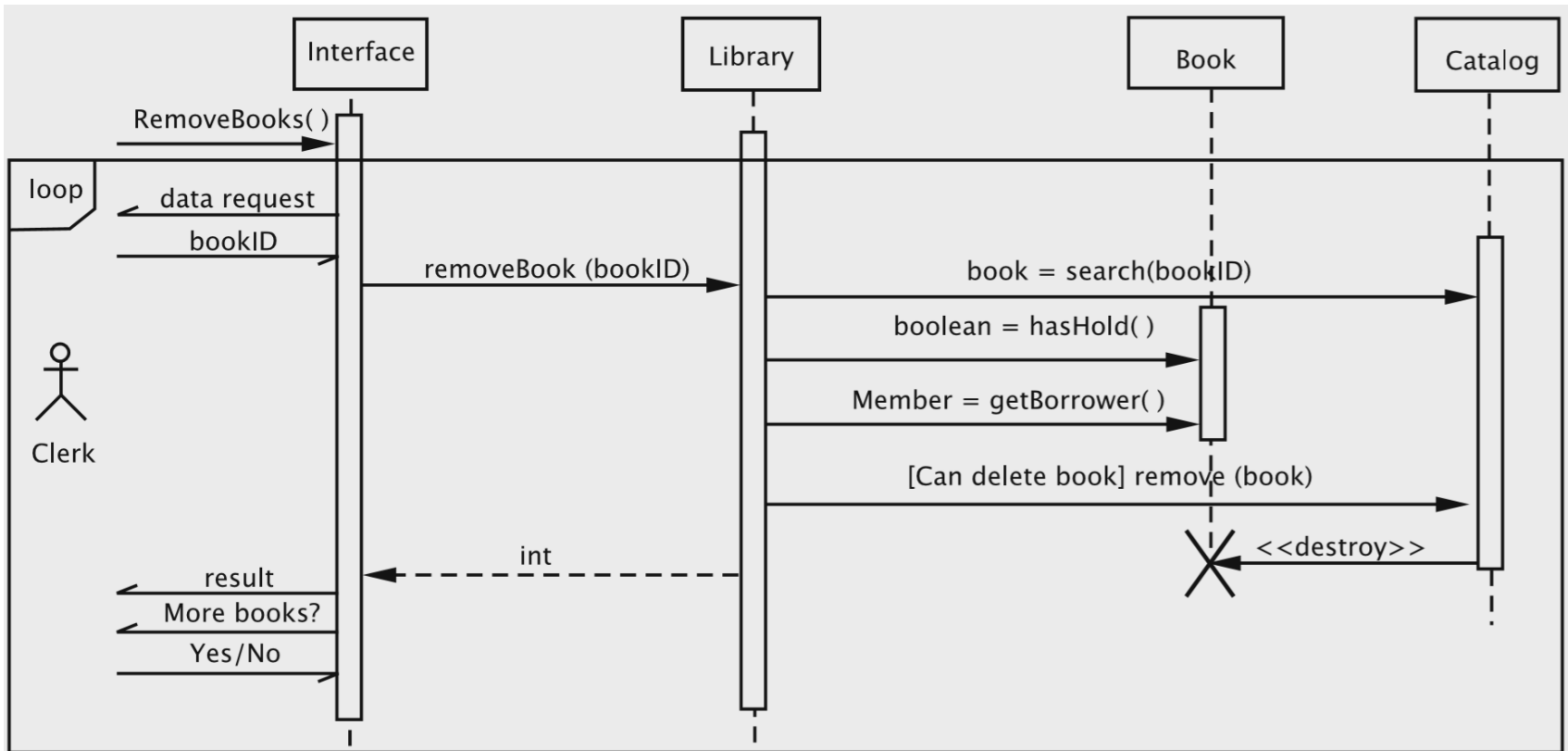


Returning books





Removing books





Placing a hold

