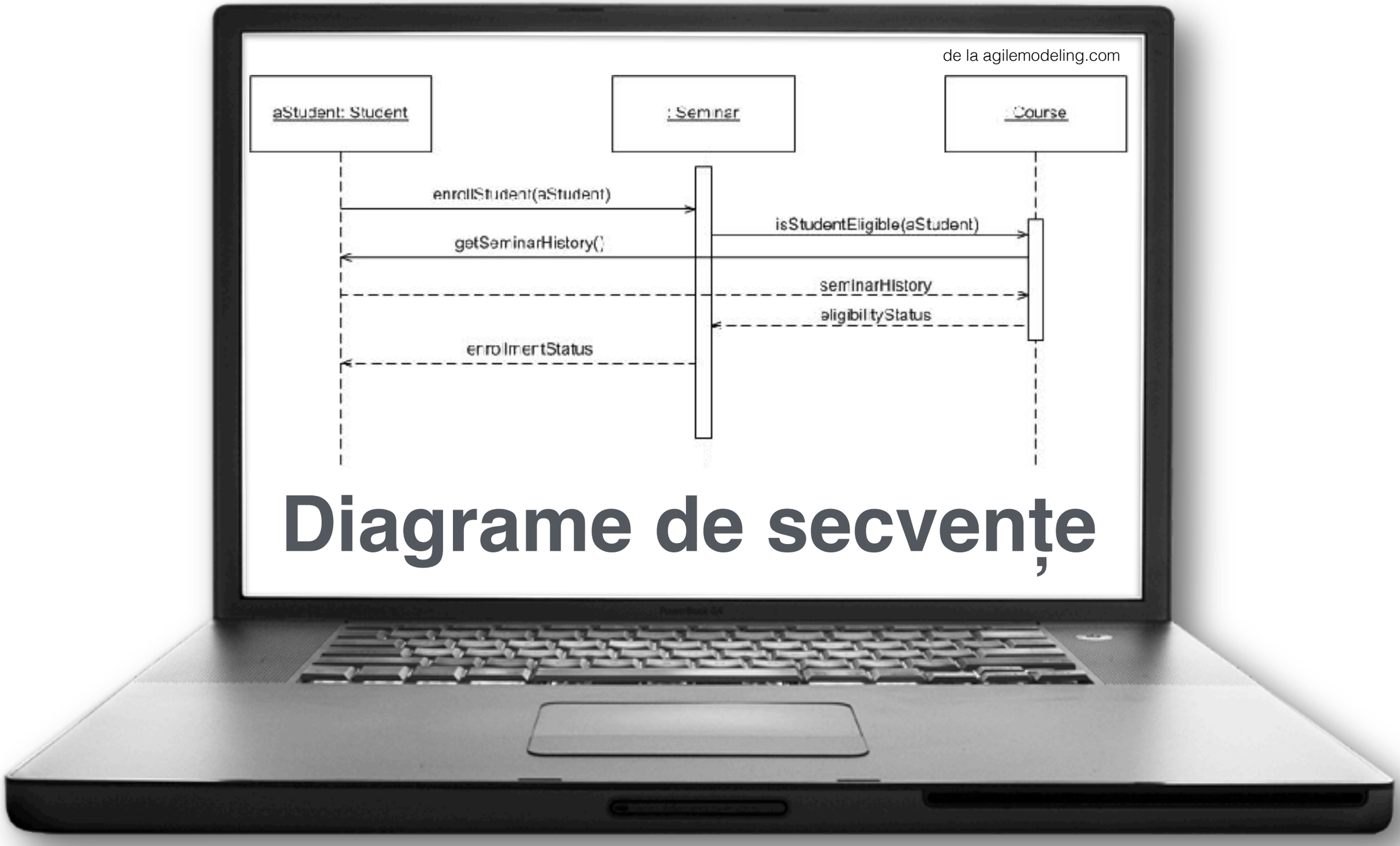
Universitatea din București

Metode de dezvoltare software

Diagrame UML de secvență

15.05.2017

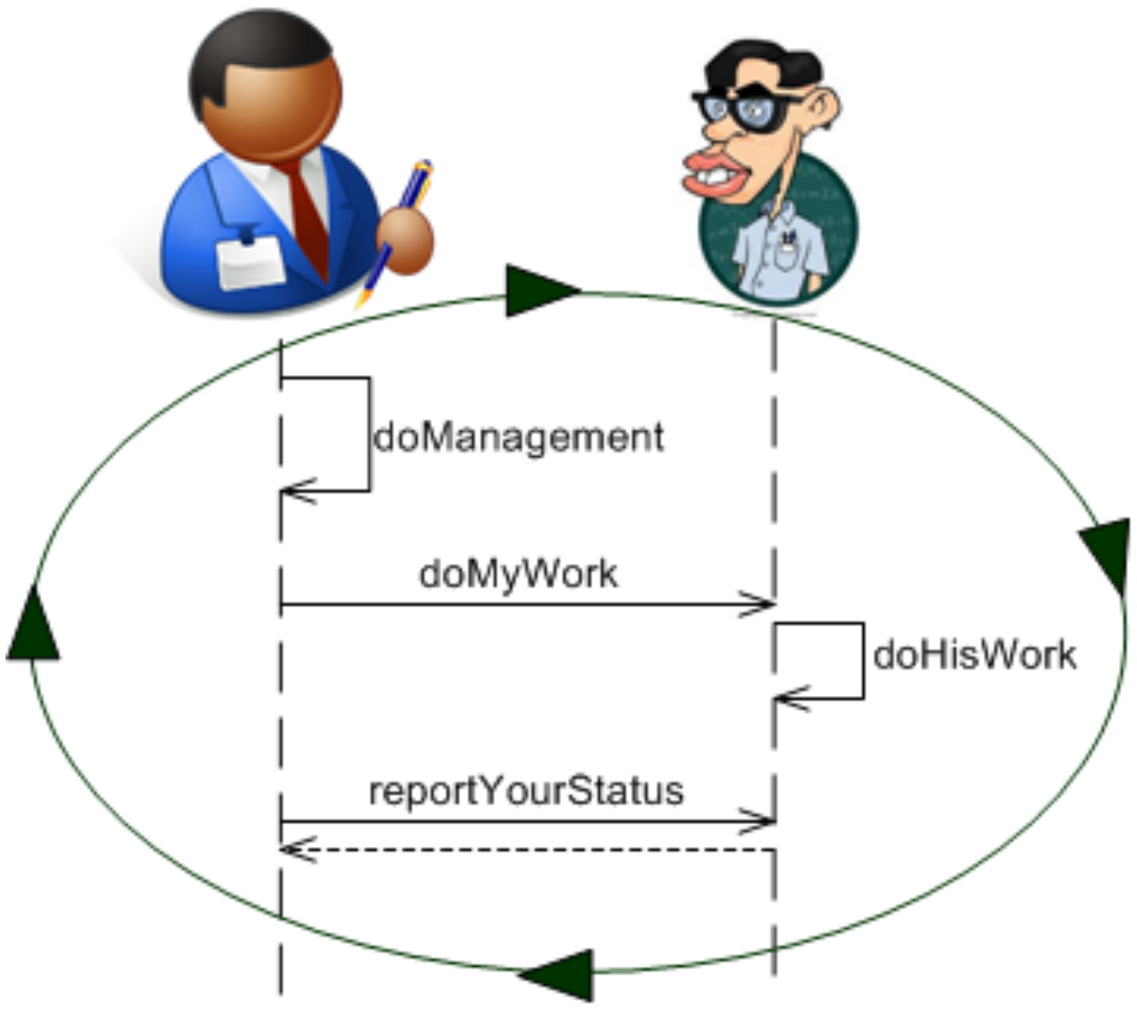
Alin Ștefănescu



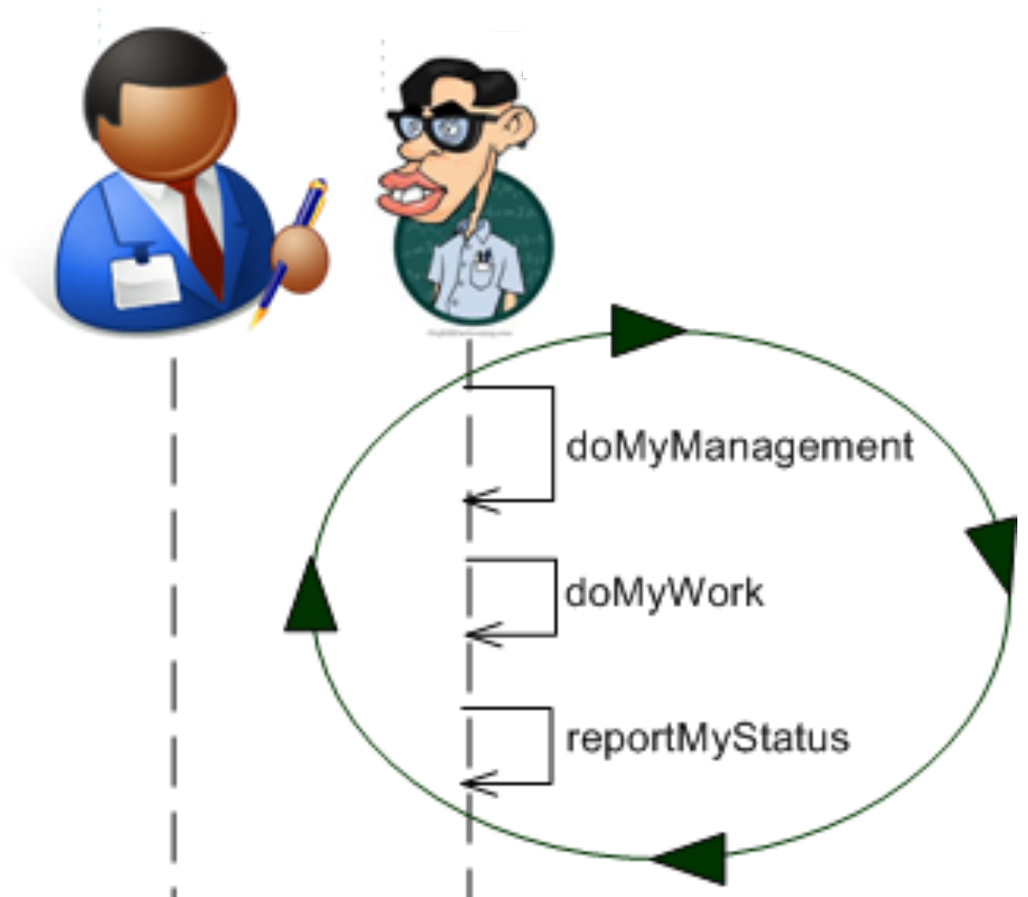
Cu exemple din “UML basics: sequence diagrams” de Donald Bell (IBM)

Diagrame de secvență… în practică

Manager

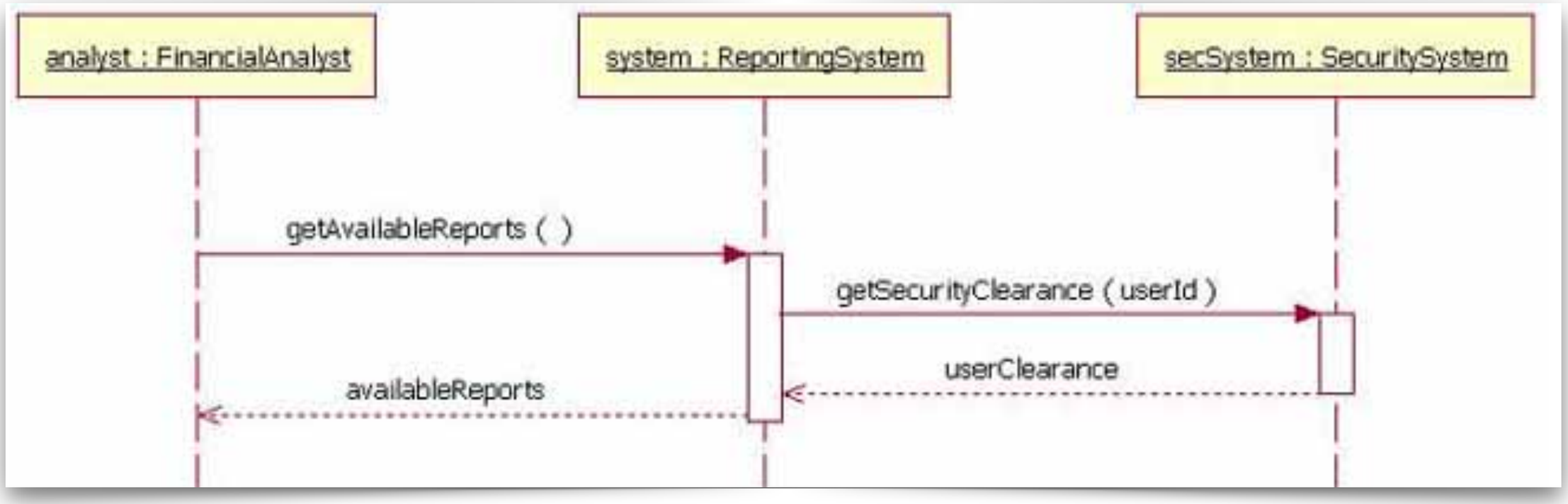


Șef (autohton)

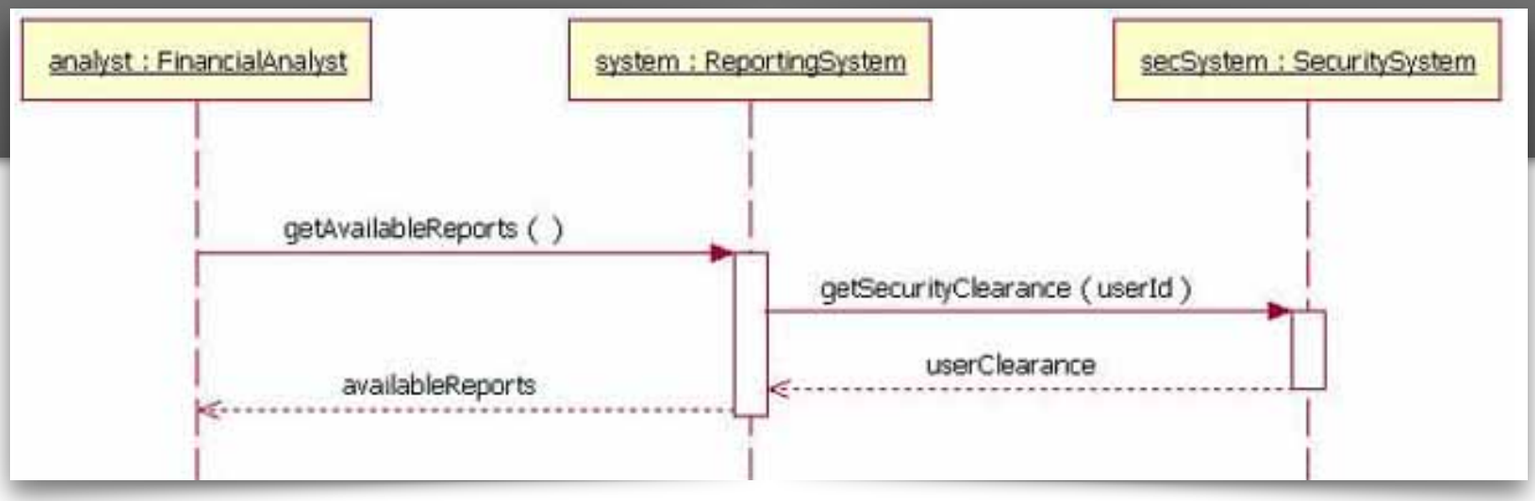


Diagrame de secvențe (sequence diagrams)

- tipul de diagramă UML care pune în evidență transmiterea de mesaje (sau apeluri de metode) de-a lungul timpului
- foarte asemănătoare cu MSC (Message Sequence Charts) standardizate separat prin ITU Z.120. Diferența principală este faptul că diagramele de secvență sunt orientate spre paradigma OO. Vezi și `http://en.wikipedia.org/wiki/Message_Sequence_Chart#Comparison_to_UML`

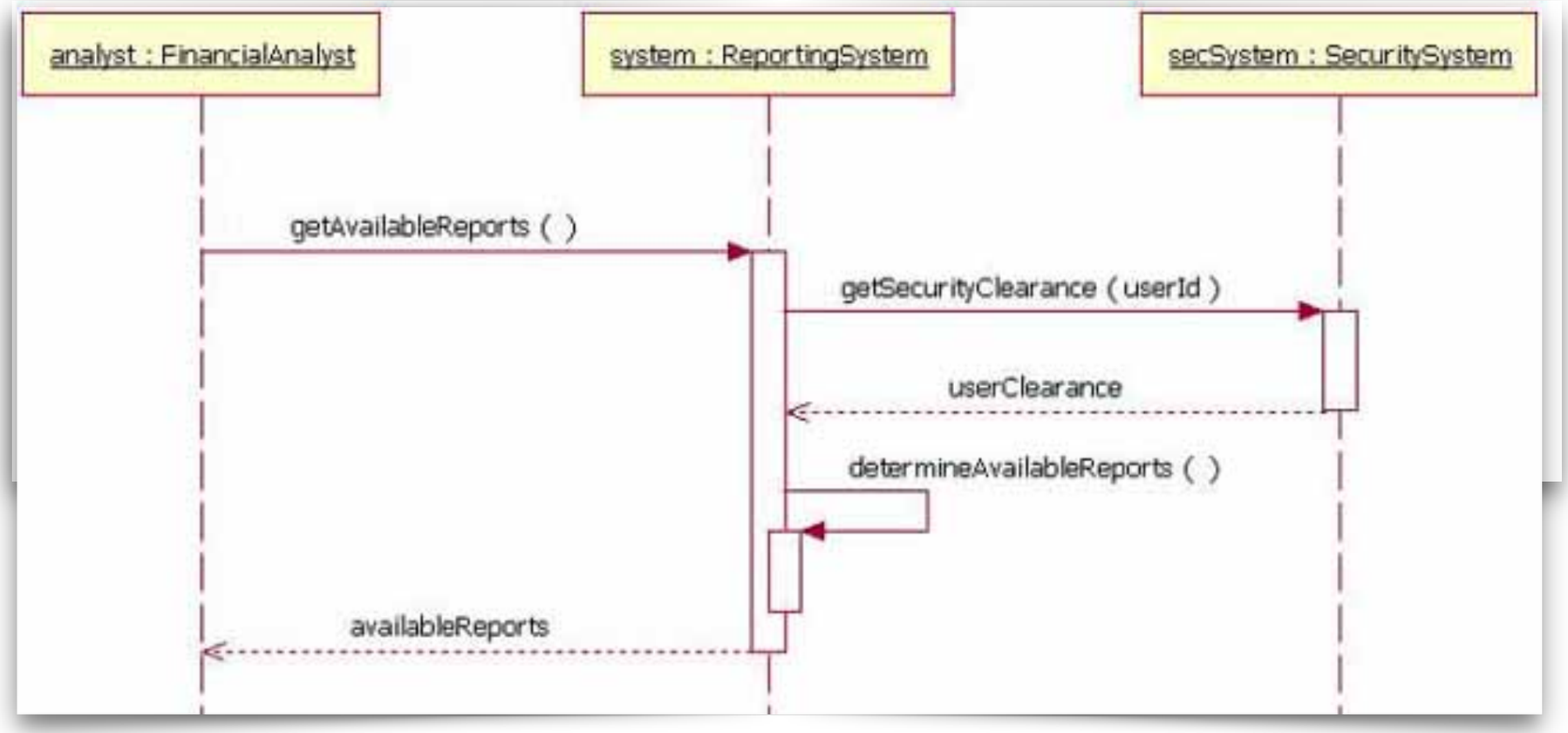


Elemente de bază



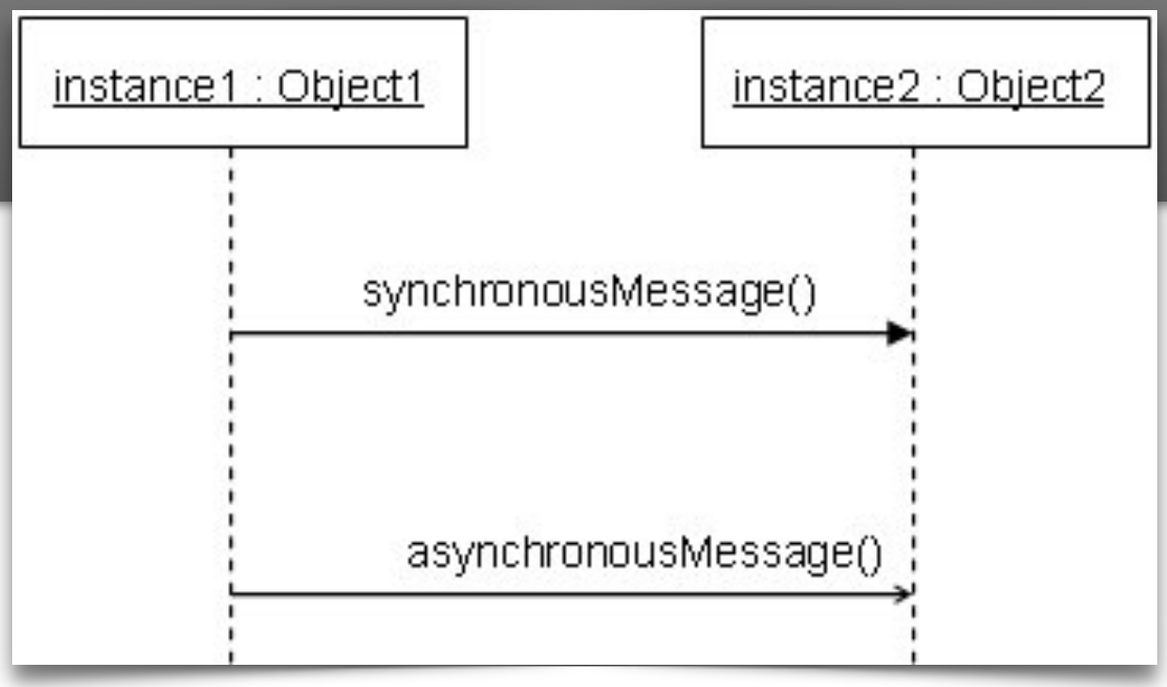
- Obiectele şi actorii sunt reprezentaţi la capătul de sus al unor linii punctate, care reprezintă linia de viaţă a obiectelor.
- Scurgerea timpului este reprezentată în cadrul diagramei de sus în jos.
- Un **mesaj se reprezintă printr-o săgeată** de la linia de viaţă a obiectului care trimite mesajul la linia de viaţă a celui care-l primeşte.
- **Timpul cât un obiect este activat** este reprezentat printr-un dreptunghi subţire care acoperă linia sa de viaţă.
- Opţional, pot fi reprezentate răspunsurile la mesaje printr-o linie punctată, dar acest lucru nu este necesar.

Mesaj intern



- Un mesaj poate fi intern (adică un mesaj nu e neapărat între două obiecte diferite)

Tipuri principale de mesaje

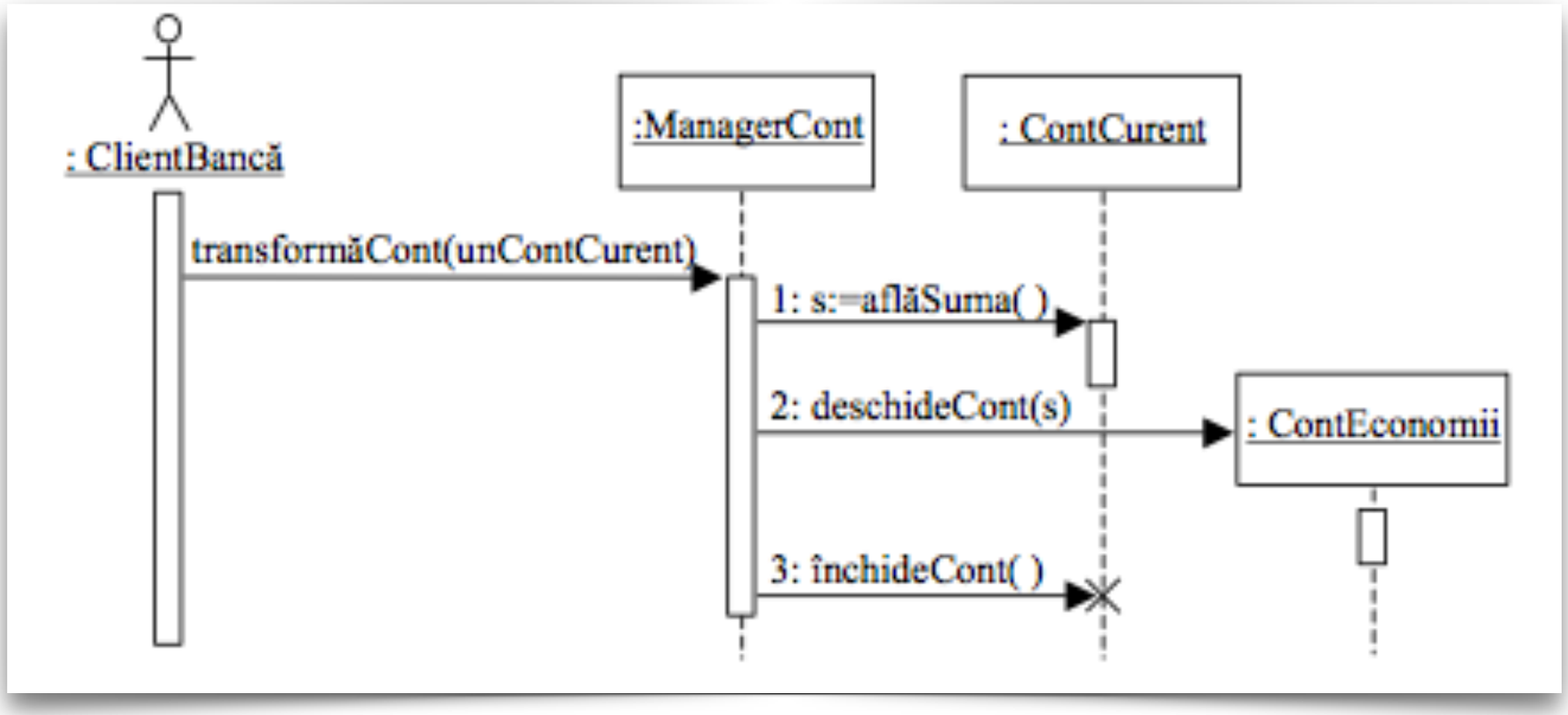


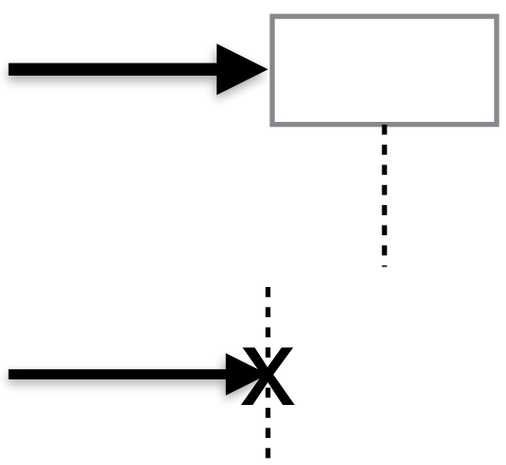
→ **mesaj sincron** (sau apel de metodă). Obiectul pierde controlul până primește răspuns

⇠ **mesaj de răspuns**: răspunsuri la mesajele sincrone; reprezentarea lor este opţională.

⟶ **mesaj asincron:** nu așteaptă răspuns, cel care trimite mesajul rămânând activ (poate trimite alte mesaje).

Creare și distrugere

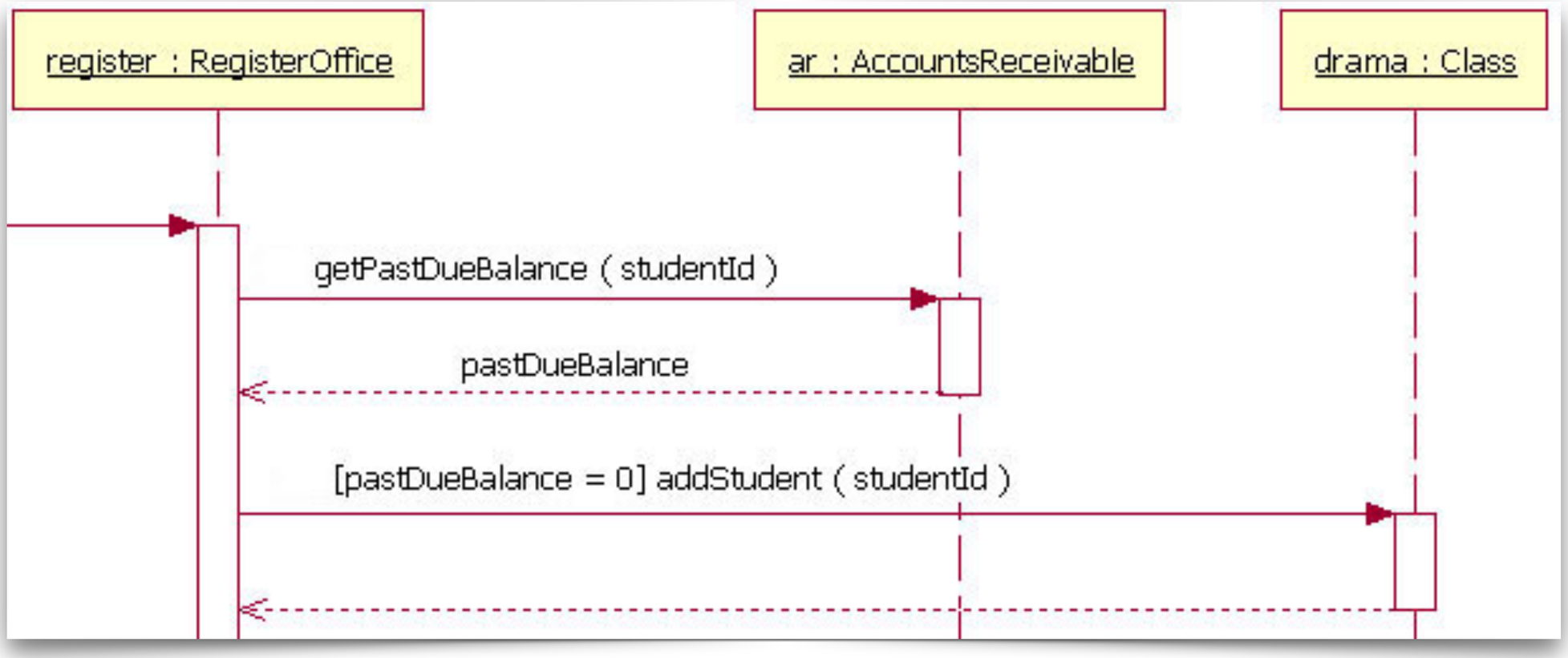




Un obiect este creat prin plasarea acestuia în josul paginii, poziţia sa corespunzând cu momentul în care a fost creat obiectul.

Distrugerea unui obiect este reprezentată printr-un X.

Gărzi



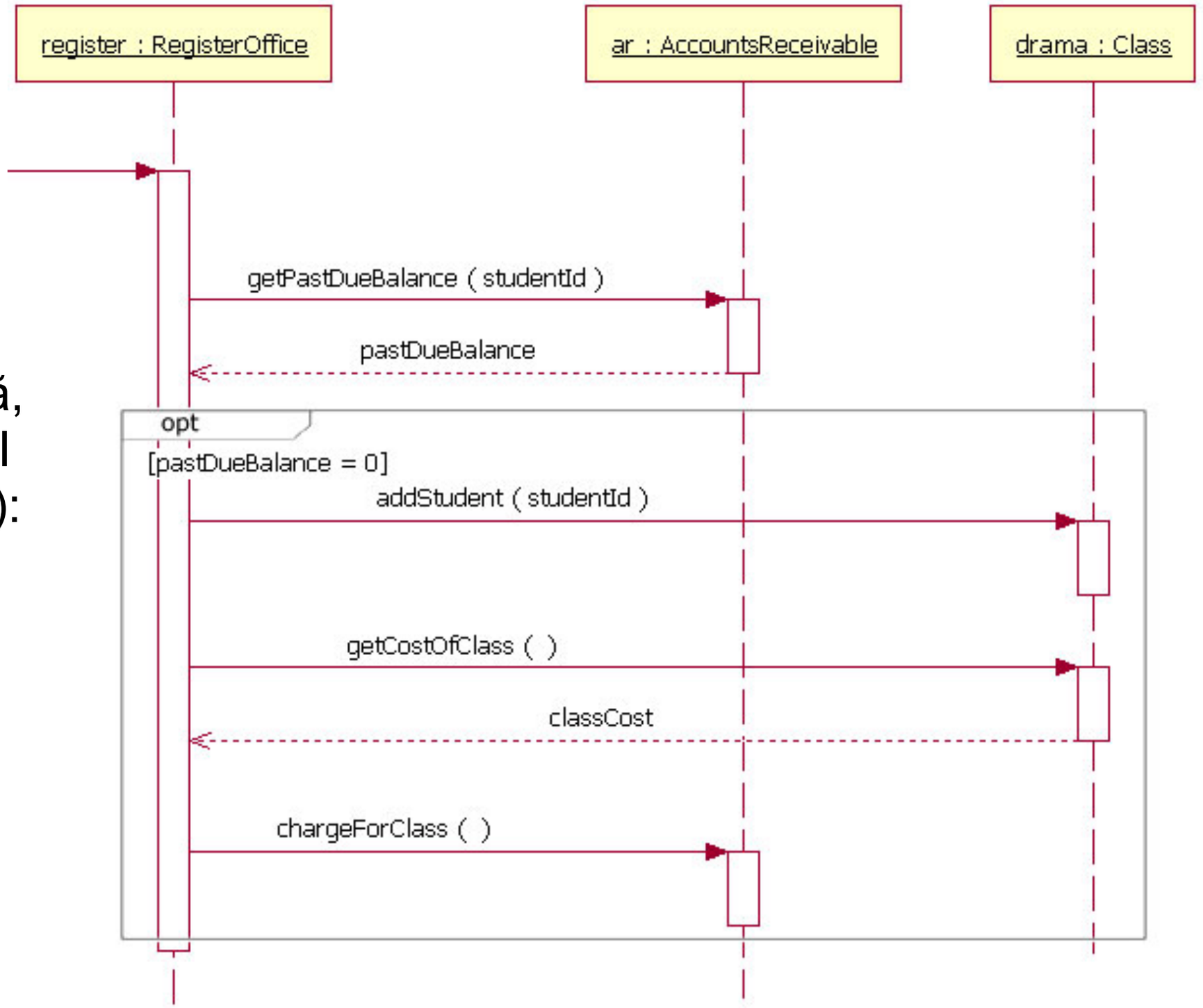
[pastDueBalance=0] decide dacă mesajul addStudent() este transmis sau nu

Fragment [opt]

Atunci când există mai multe mesaje dependente de o gardă, se folosește fragmentul (combination fragment):

opt
[gardă]
…

(Similar cu
`if … then …`
din programare)



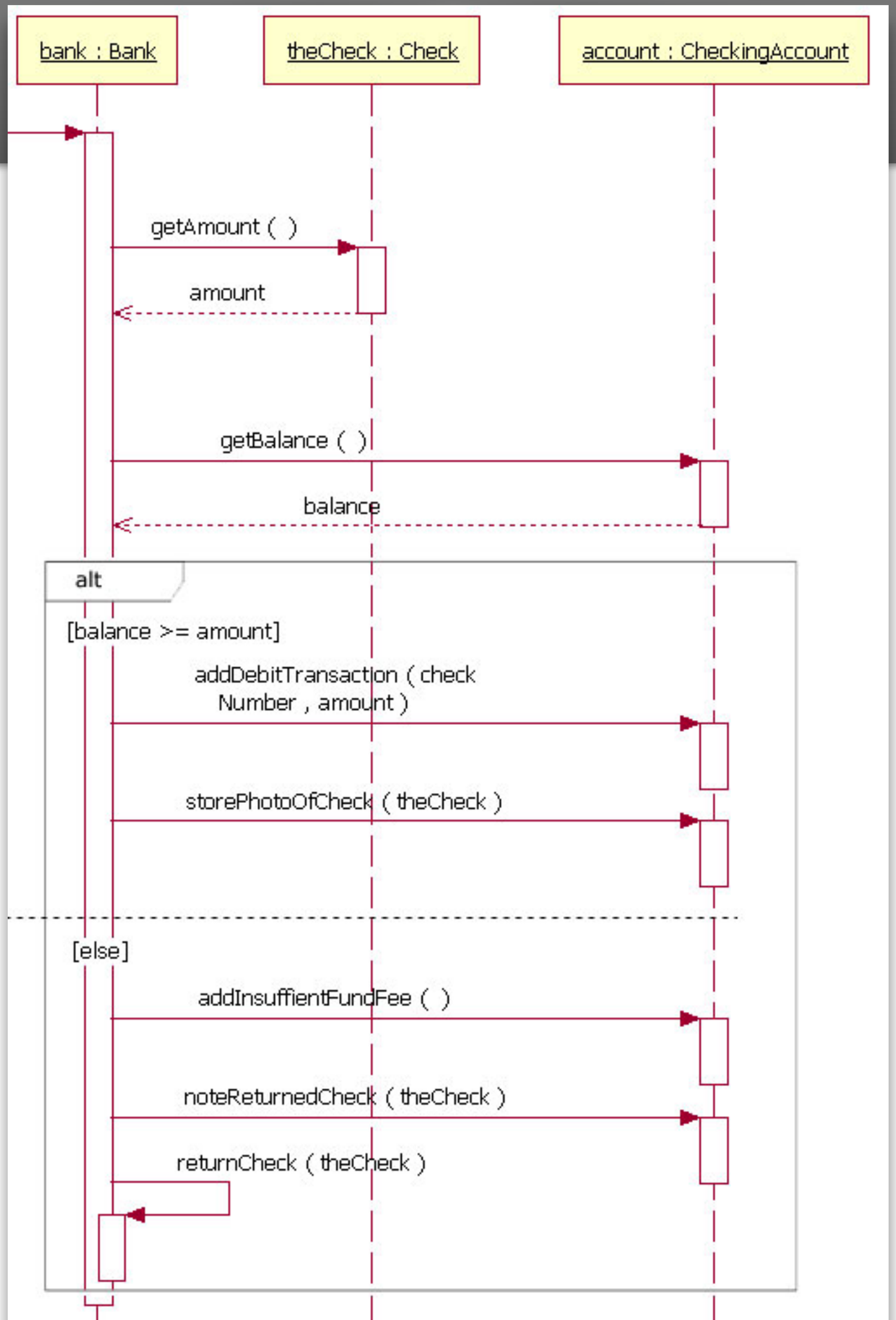
Fragment [alt]

Similar cu

`if … then … else …`

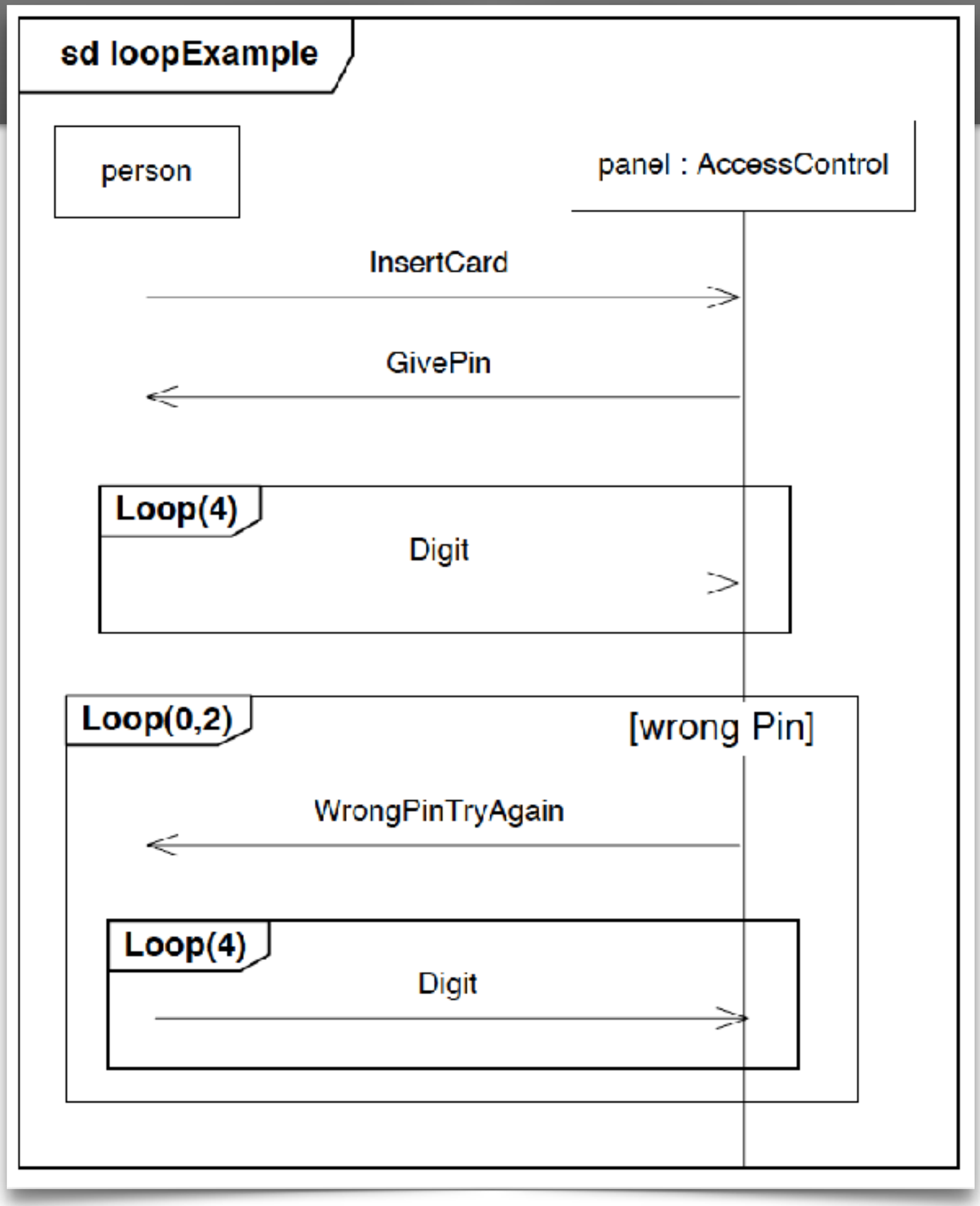
există în diagramele de secvențe

alt
[gardă]
…
[else]
…



Fragment [loop]

Structurile repetitive sunt introduse prin cuvântul cheie **[loop]** urmat de o un număr care spune de câte ori se execută grupul respectiv sau de o gardă.



Și multe altele…

- Există multe alte concepte disponibile în diagramele de secvențe, acestea permițând modelarea unor scenarii complexe

- De exemplu:
  - [neg]: secvențe de mesaje invalide (care nu trebuie să aibă loc)
  - [par]: paralelism
  - [ref] includerea diagramelor unele în altele
  - aspecte temporale: cuantificarea trecerii timpului între mesaje
  - invarianți
  - etc.