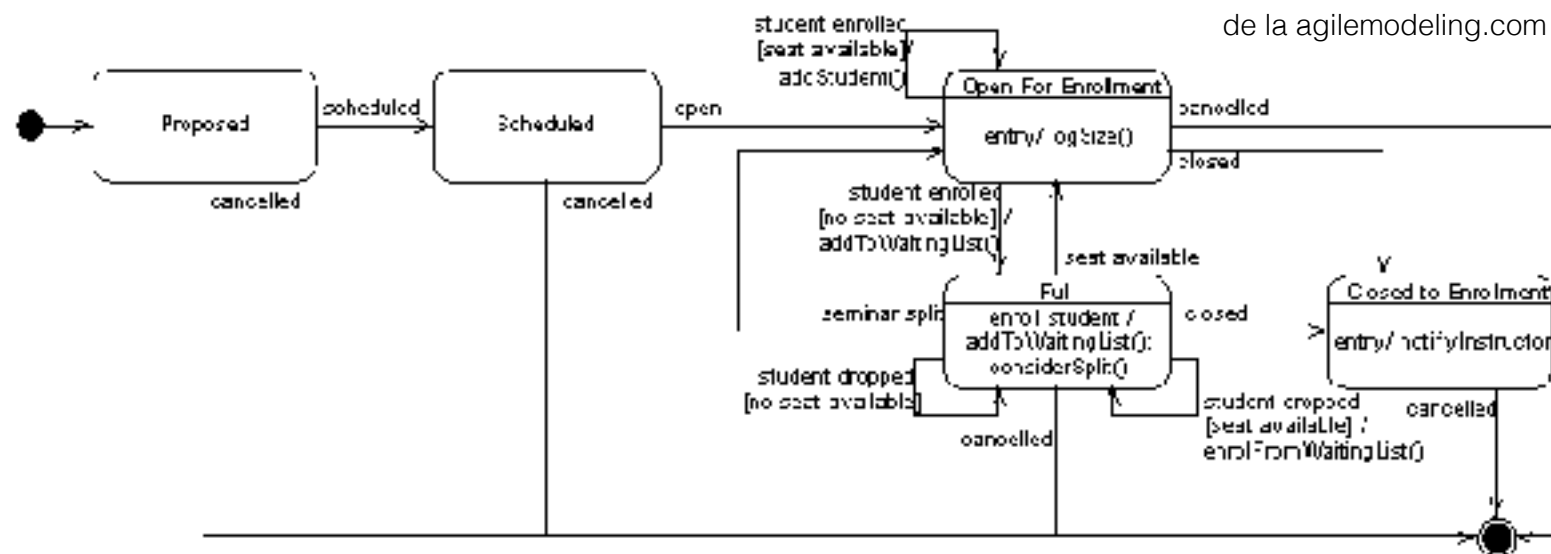


Metode de dezvoltare software

Diagrame UML de stări

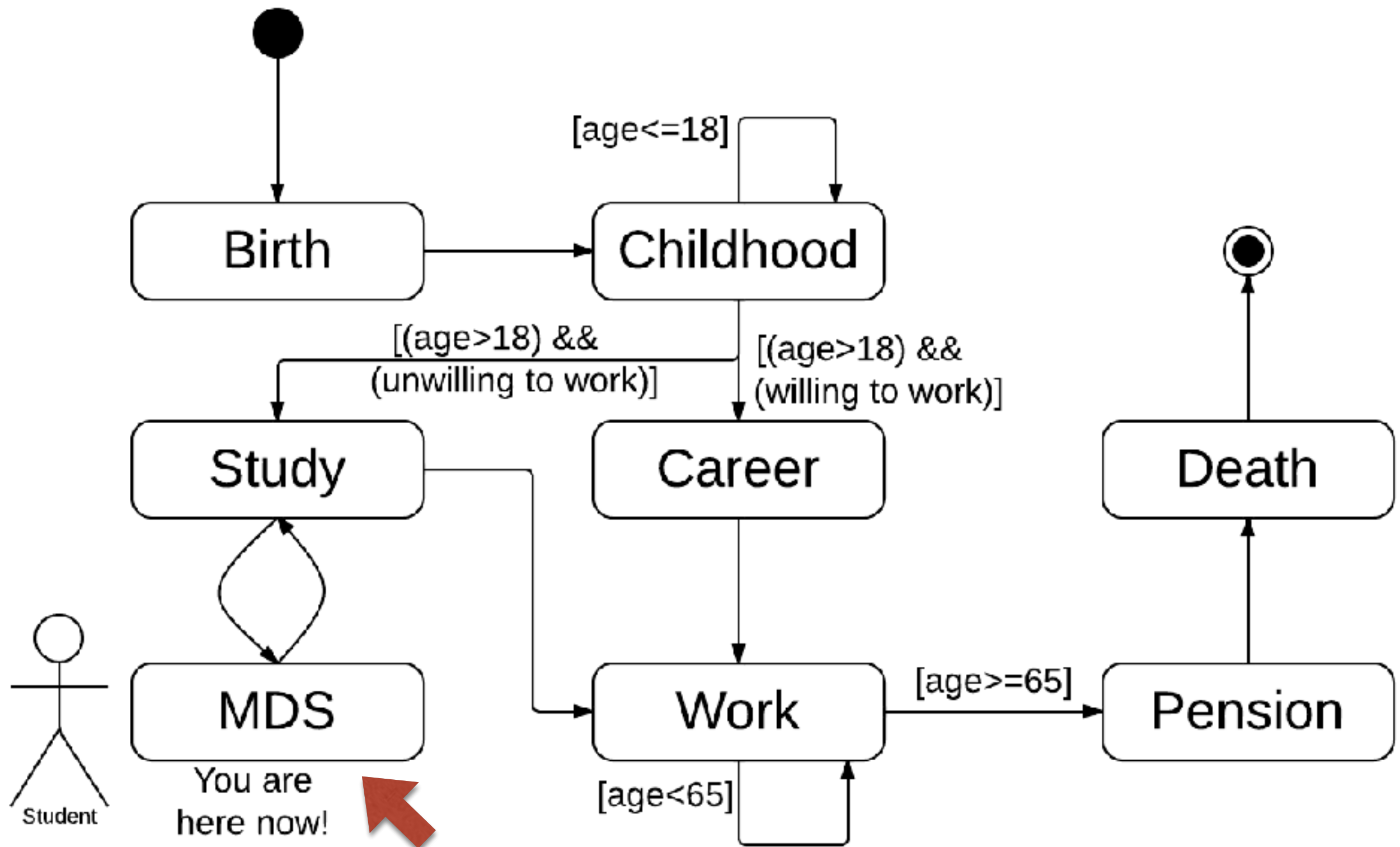
19.03.2018

Alin Ștefănescu



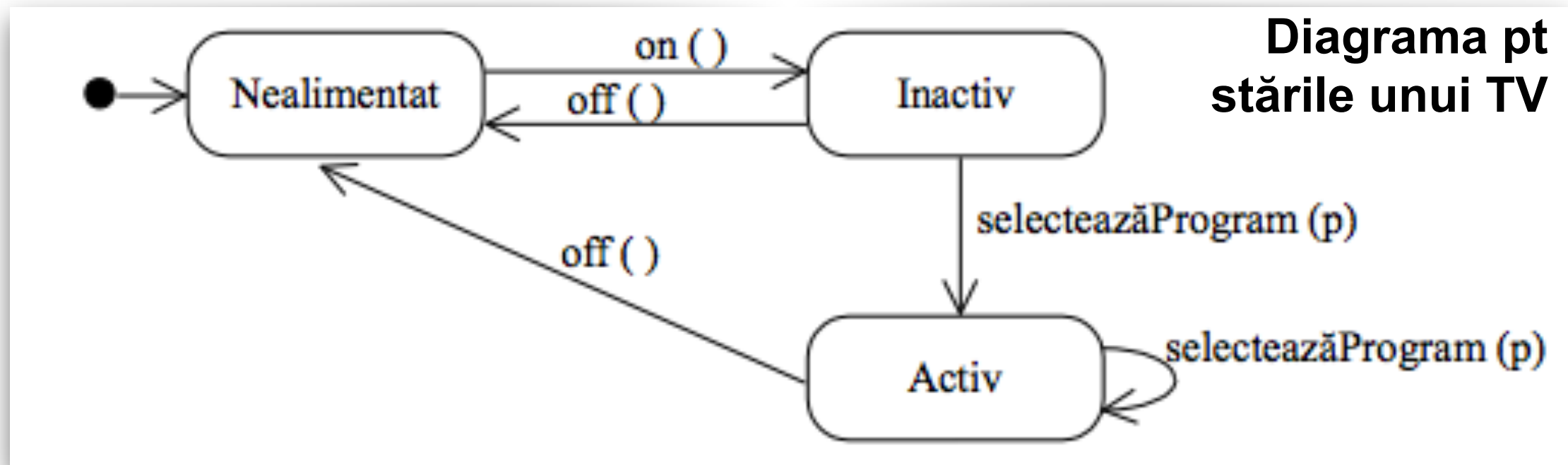
Diagrame de stări

Diagrame de stări... în viața reală

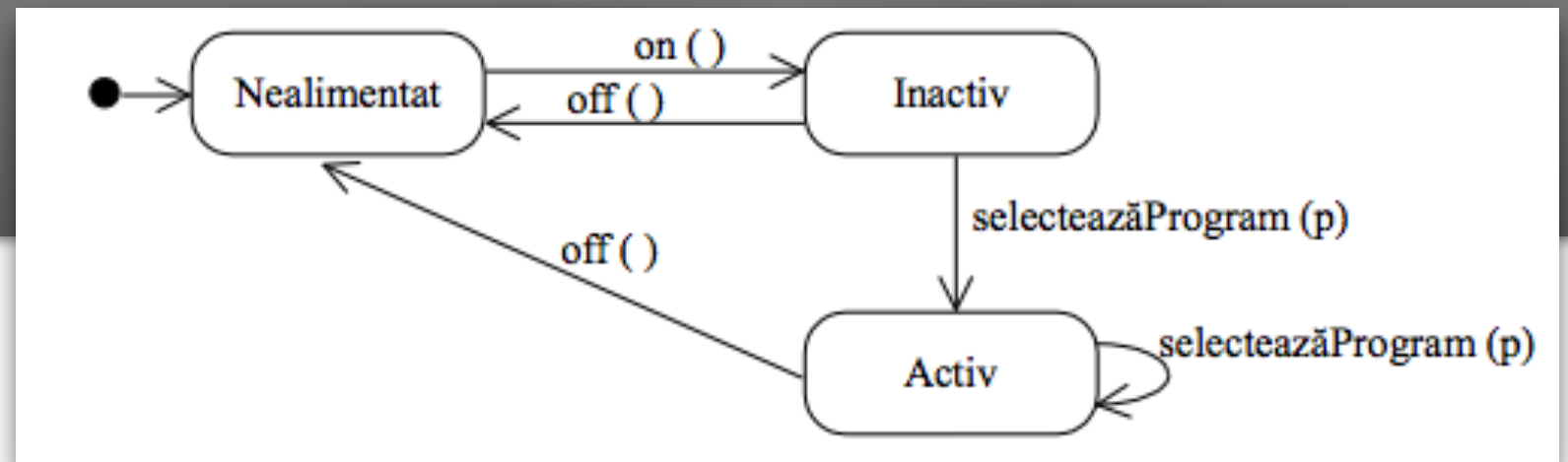


Mașini de stări (State machines)

- Obiectele din aceeași clasă pot reacționa diferit la primirea unui mesaj, acest lucru depinzând de **starea** lor, adică de valorile atributelor obiectelor.
- **Diagramele de stare** (numite și *mașini de stare* sau *statecharts*) descriu dependența dintre starea unui obiect și mesajele pe care le primește sau alte evenimente recepționate.



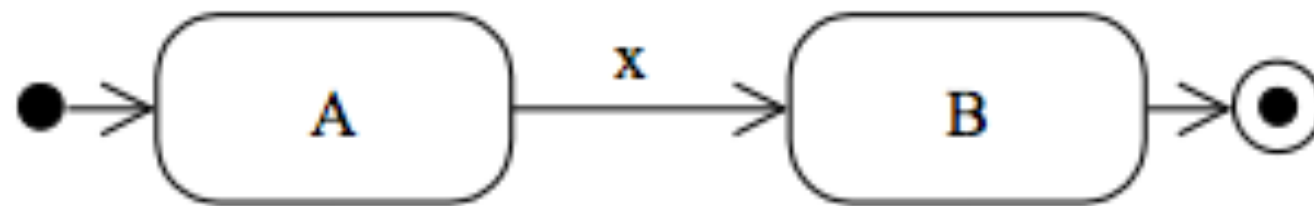
Elemente



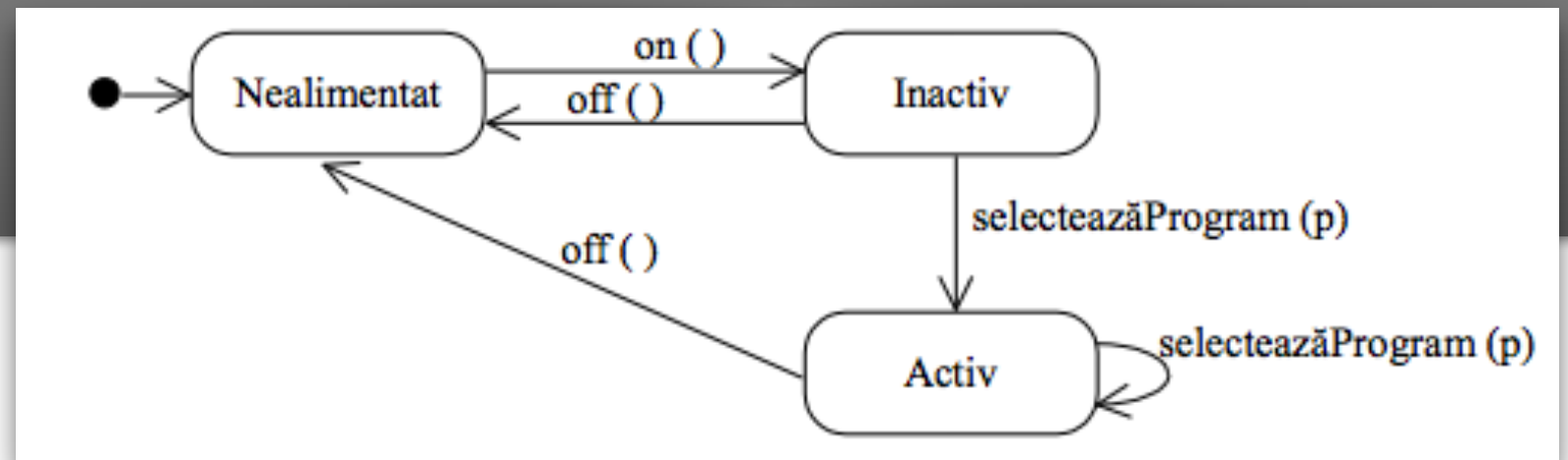
- **stări**, reprezentate prin dreptunghiuri cu colțuri rotunjite
- **tranziții** între stări, reprezentate prin săgeți
- **evenimente** care declanșează tranzițiile dintre stări
- cel mai des întâlnite evenimente sunt mesajele primite de către obiect.

Începutul și sfârșitul

- **semnul de început**, reprezentat printr-un disc negru din care pornește o săgeată (fără etichetă) spre **starea inițială** a sistemului.
- pot exista de asemenea și **semne de sfârșit**, reprezentate printr-un disc negru cu un cerc exterior, în care sosesc săgeți din **stările finale** ale sistemului. Acestea corespund situațiilor în care obiectul ajunge la sfârșitul vieții sale și este distrus.



Stări



- O stare este o mulțime de configurații ale obiectului care se comportă la fel la apariția unui eveniment.
- O stare poate fi identificată prin constrângeri aplicate atributelor obiectului.
- Exemplu: diagrama de stare pentru televizor

Atribute (ale unei clase Televizor)

`esteAlimentat : Boolean`

`numarProgram : Integer`

Constrângeri care definesc stările:

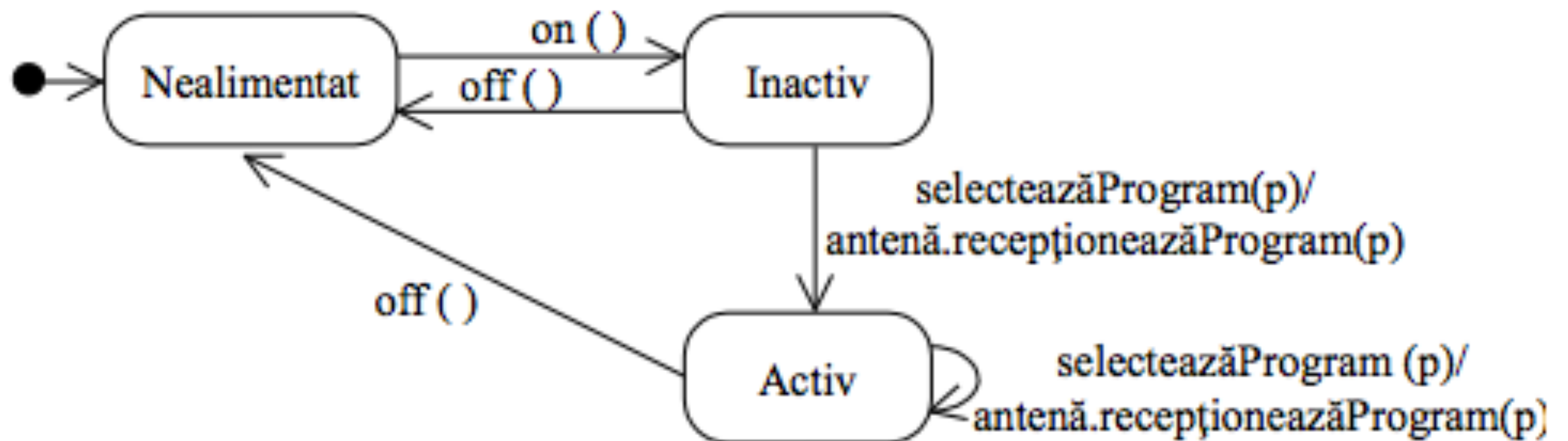
Nealimentat: `{not esteAlimentat}`

Inactiv: `{esteAlimentat and numarProgram = 0}`

Activ: `{esteAlimentat and numarProgram > 0}`

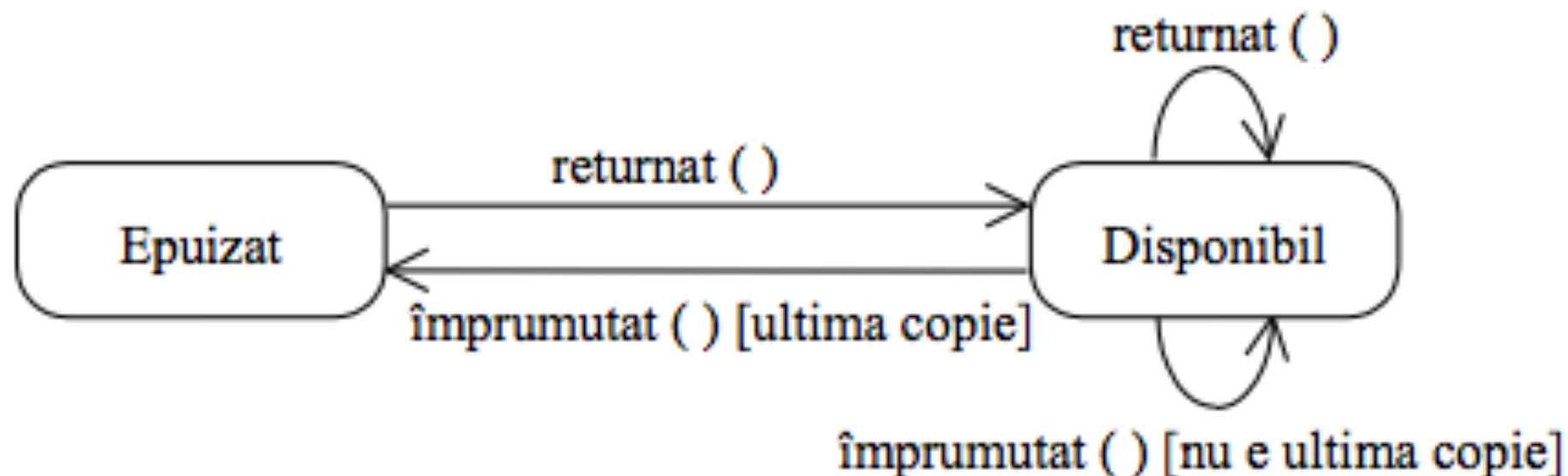
Evenimente și acțiuni

- un **eveniment** este ceva care se produce asupra unui obiect, precum primirea unui mesaj.
- o **acțiune** reprezintă ceva care poate fi făcut de către obiect, precum transmiterea unui mesaj.
- reprezentare pe tranziții: **eveniment/acțiune**
 - `selecteazaProgram(p) / antena.recepționeazăProgram(p)`



Gărzi

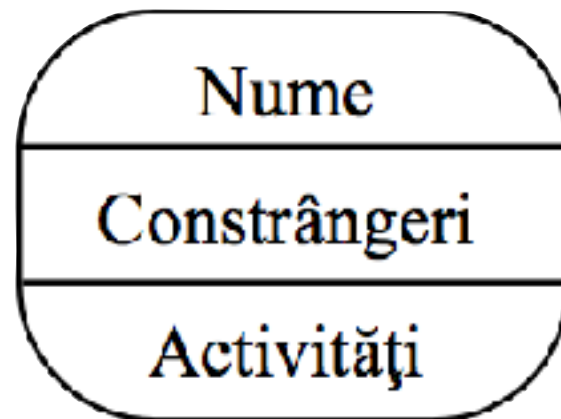
- pe lângă evenimentele care declanșează o tranziție (schimbarea stărilor), există și **gărzi**, care pot condiționa execuția tranzițiilor.
- Astfel, în anumite situații, un eveniment declanșează o tranziție **numai dacă** atributele obiectului îndeplinesc o anumită **condiție** suplimentară (gardă).
- reprezentare pe tranziții: **eveniment** [**gardă**] / **acțiune**



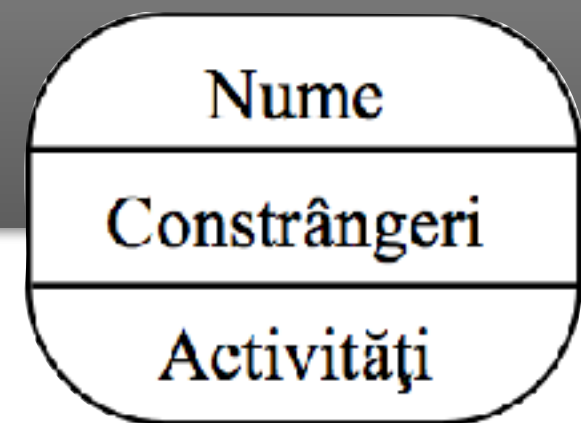
Activitățile stărilor

- evenimentele și acțiunile pot fi asociate și stărilor, în acest caz ele fiind folosite pentru a specifica activitățile desfășurate în timpul în care obiectul se află în starea respectivă.

Stare

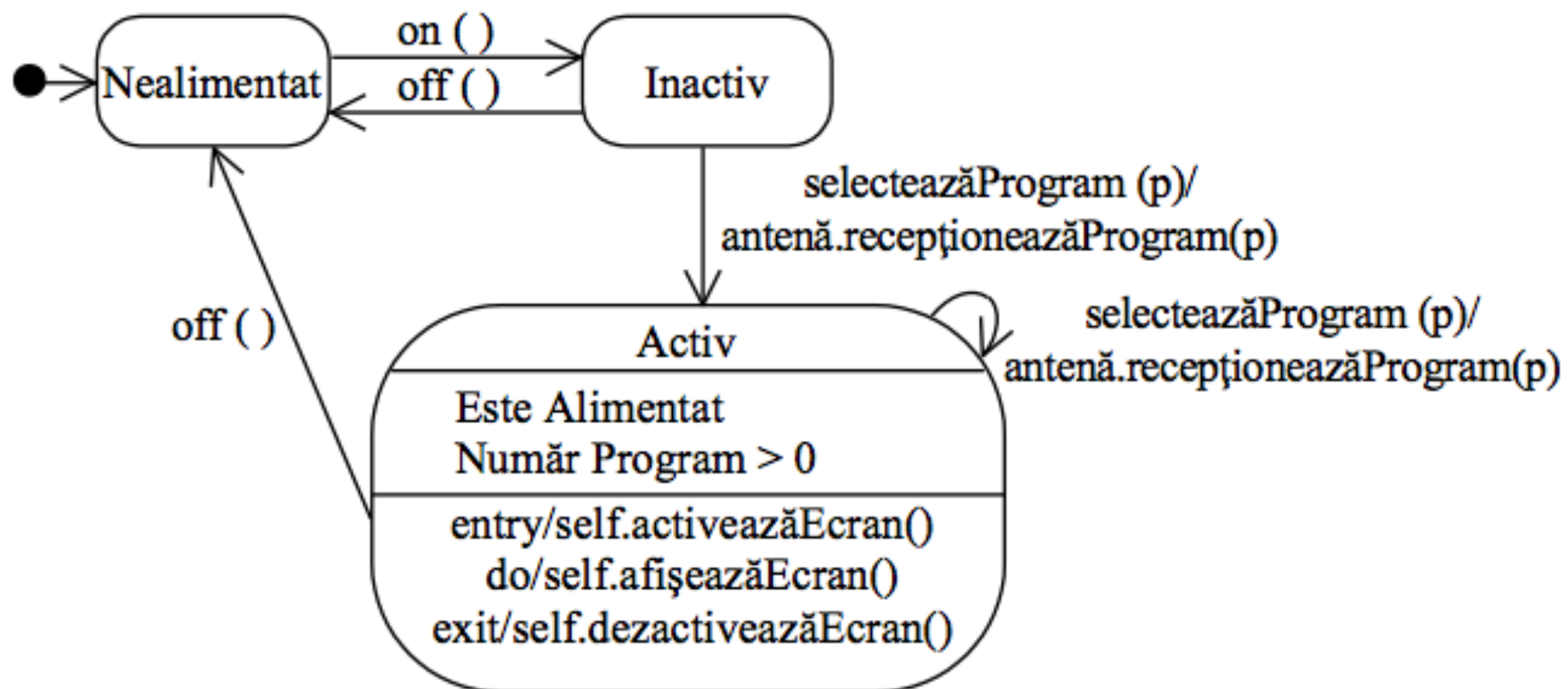


Activitățile stărilor



■ Cel mai frecvent utilizate evenimente asociate stărilor sunt:

- **entry** : activități produse când obiectul **intră** în starea respectivă;
- **exit** : activități produse când obiectul **iese** din starea respectivă;
- **do** : activități produse **în timp ce** obiectul se află în starea respectivă.

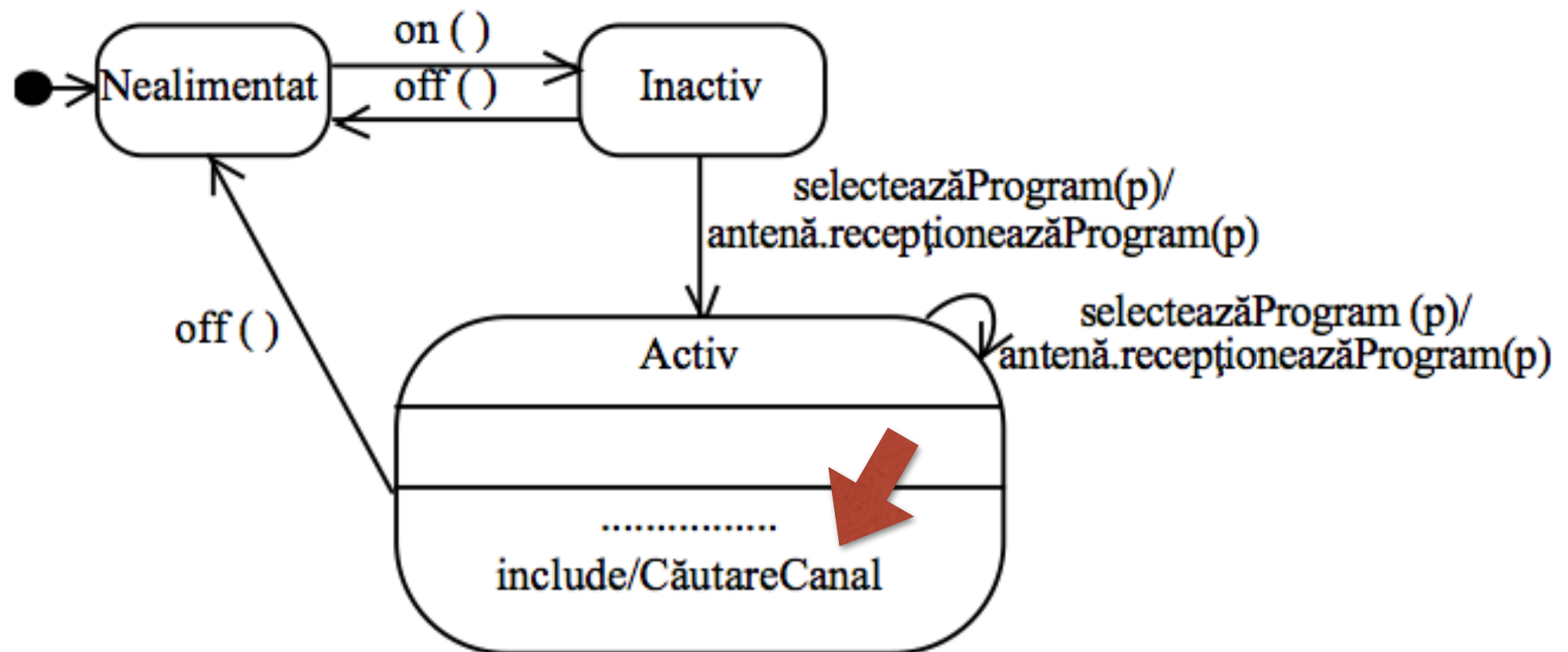
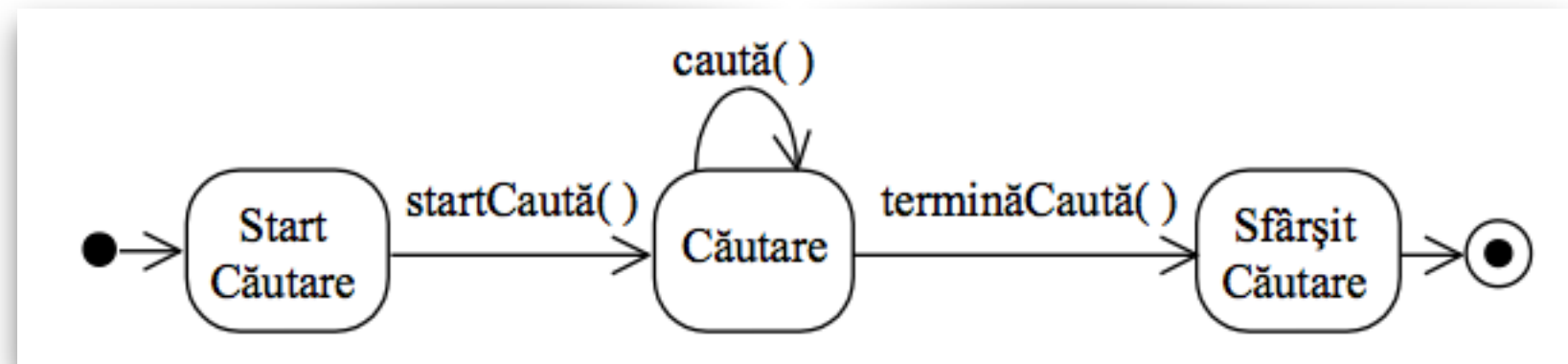


Stări compuse

- O stare `S` poate conține substări care detaliază comportamentul sistemului în starea `S`.
În acest caz, spunem că `S` este o **stare compusă**.
- **Exemplu:** situația căutării unui canal de televiziune se face în timp ce televizorul este activ și poate fi reprezentată ca o diagramă de stare inclusă `CăutareCanal`.
- Astfel, starea `Activ` va deveni compusă, incluzând subcomportamentul de căutare. Pentru aceasta se folosește notația `include/CăutareCanal`.

Stări compuse: substări

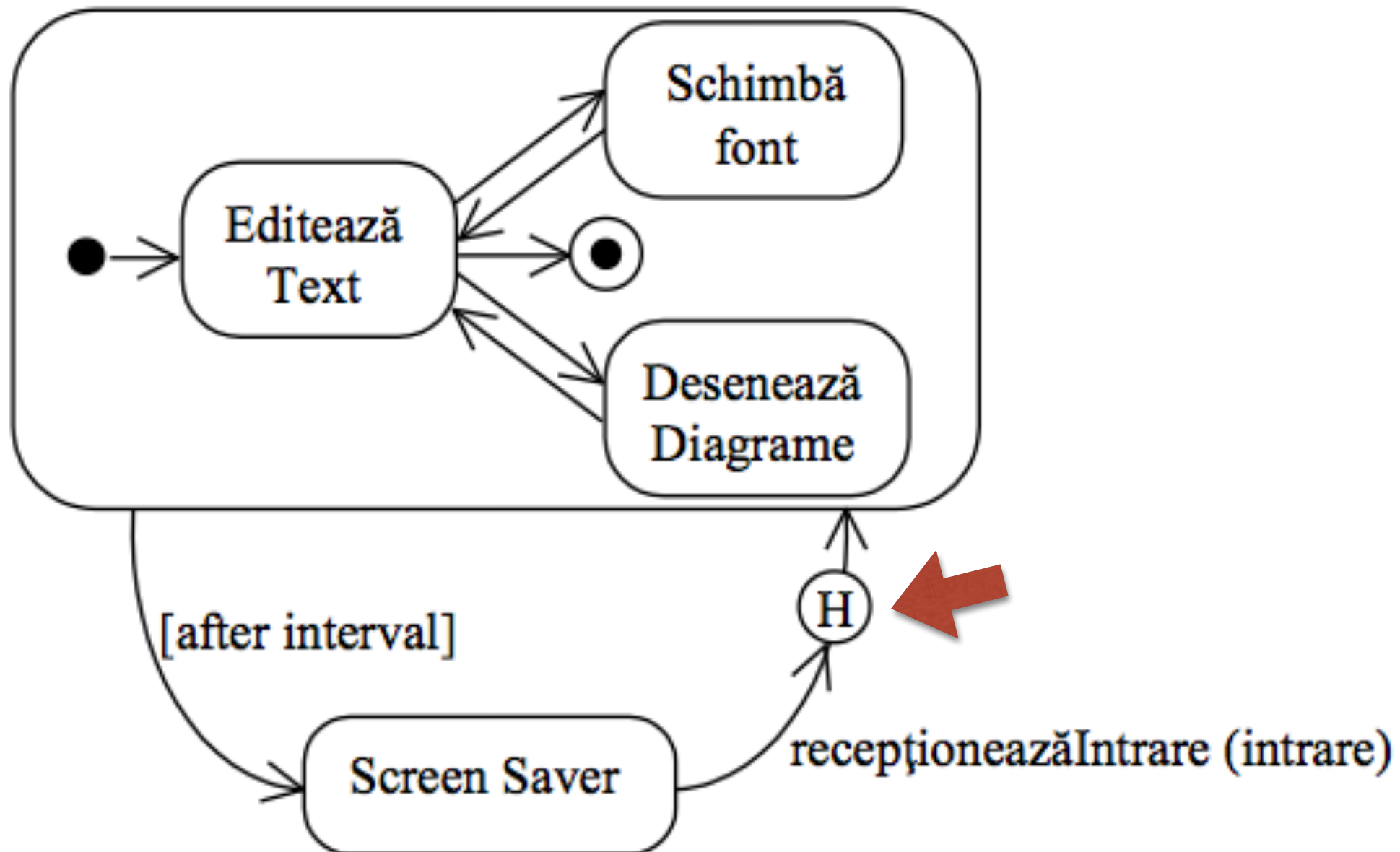
CăutareCanal:



Stări compuse: stări “istoric”

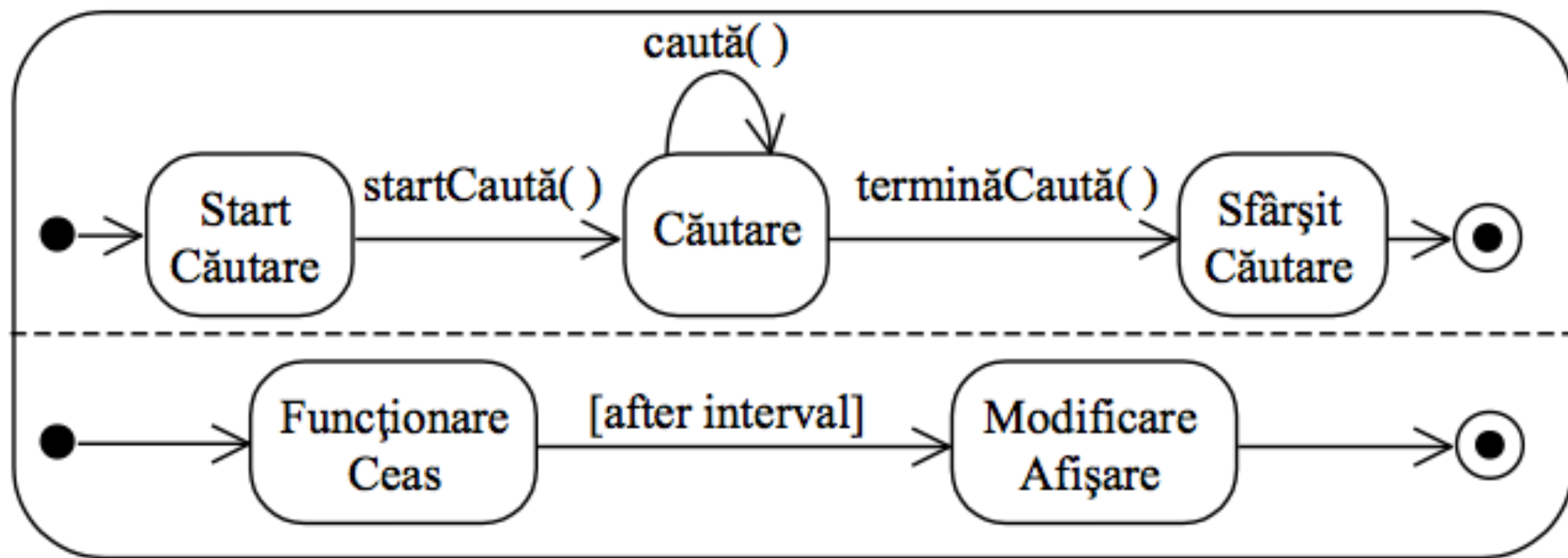
- În general, când o tranziție intră într-o stare compusă, submașina corespunzătoare acesteia va începe să funcționeze din starea sa inițială.
- Uneori este necesar ca submașina să-și “**reamintească**” starea în care a rămas și să-și reia funcționarea din acea stare.
- Pentru acest lucru se folosește o stare “istoric”, reprezentată printr-un cerc în care apare litera **H**.

Stări “istoric” - exemplu



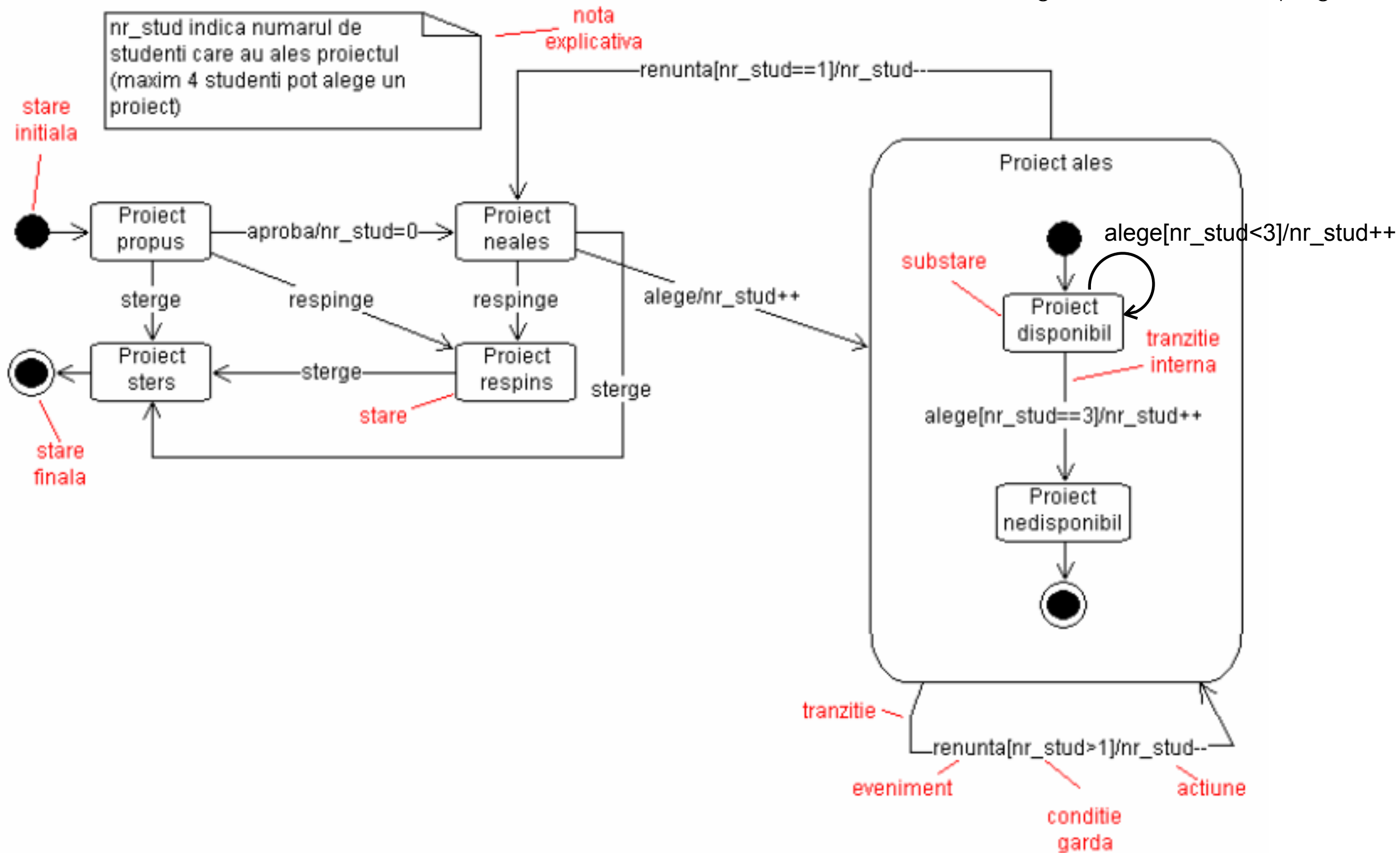
Stări compuse: stări concurente

- Există posibilitatea exprimării **activităților concurente** dintr-o stare.
- **Grafic**: se împarte dreptunghiul corespunzător stării compuse printr-o **linie punctată**, în **regiunile** obținute fiind reprezentate submașinile care vor acționa concurent.
- **Exemplu**: adăugăm la cazul televizorului considerat și situația când acesta conține un ceas, care funcționează în același timp cu televizorul.



Încă un exemplu (proiecte pentru studenți)

de la Florin Ostafi - "Ingineria sistemelor de programe"



Și un alt exemplu (câteva operații într-un website)

