

# UML: Unified Modelling Language

Enric Mieza - 2013

# Què és UML?

- Llenguatge estàndard per especificar, visualitzar, construir i documentar sistemes de software, models de negoci i altres sistemes.
- El seu ús permet unes bones pràctiques de desenvolupament.
- Útil per sistemes complexes.
- Orientació a objectes.
- Representació gràfica.
- Ajuda al treball en grup (comunicació, exploració de dissenys, validació, etc.)

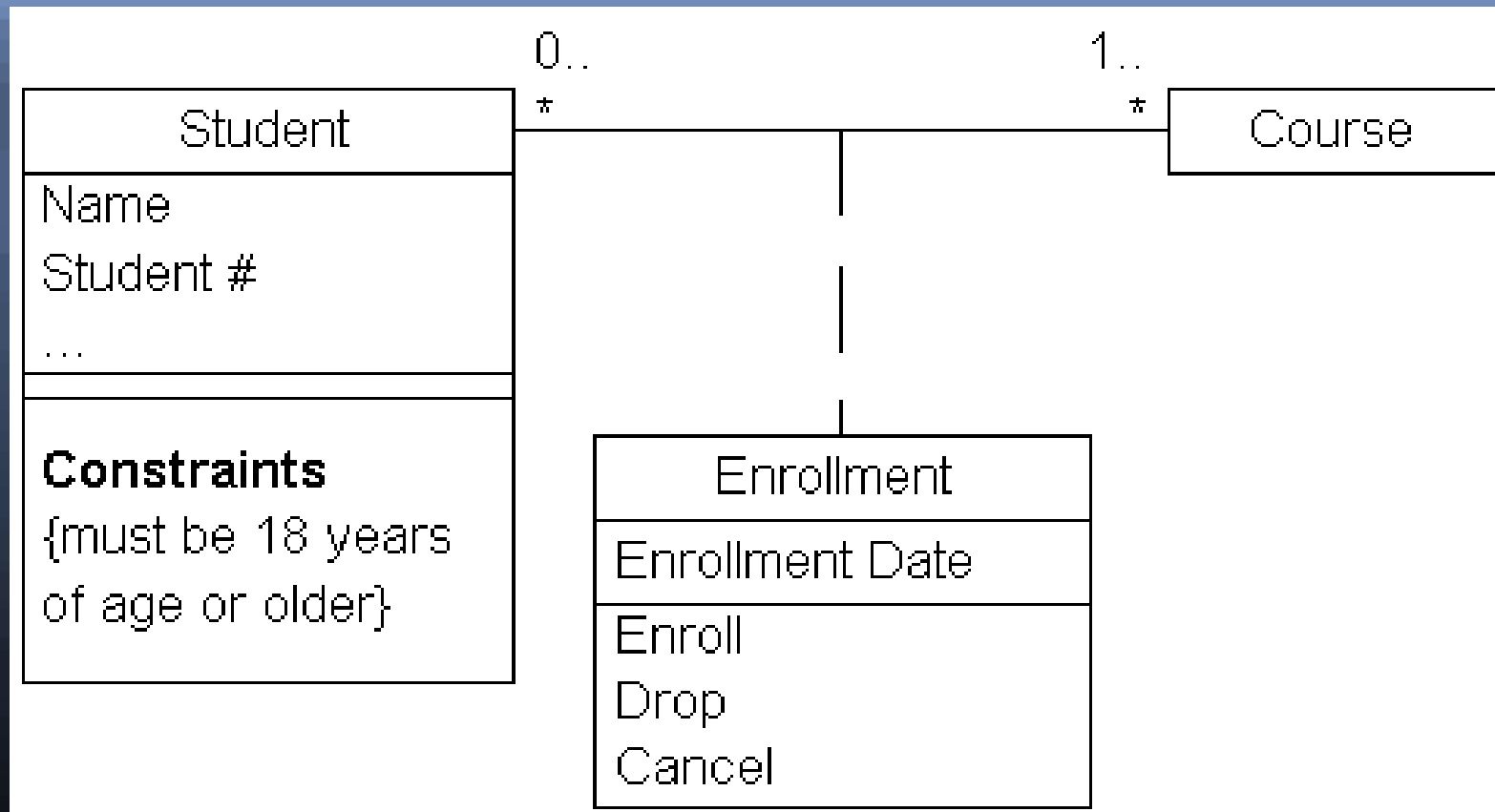
# Tipus de diagrama UML

- Estructural
  - **Classes**
  - Components (llibreries)
  - **Desplegament**
  - Objectes (instàncies)
- Comportamental (behavioral)
  - **Activitat (flux)**
  - **Estat**
  - **Casos d'ús**
  - Interacció
    - **Seqüència**
    - Comunicació
    - Timing

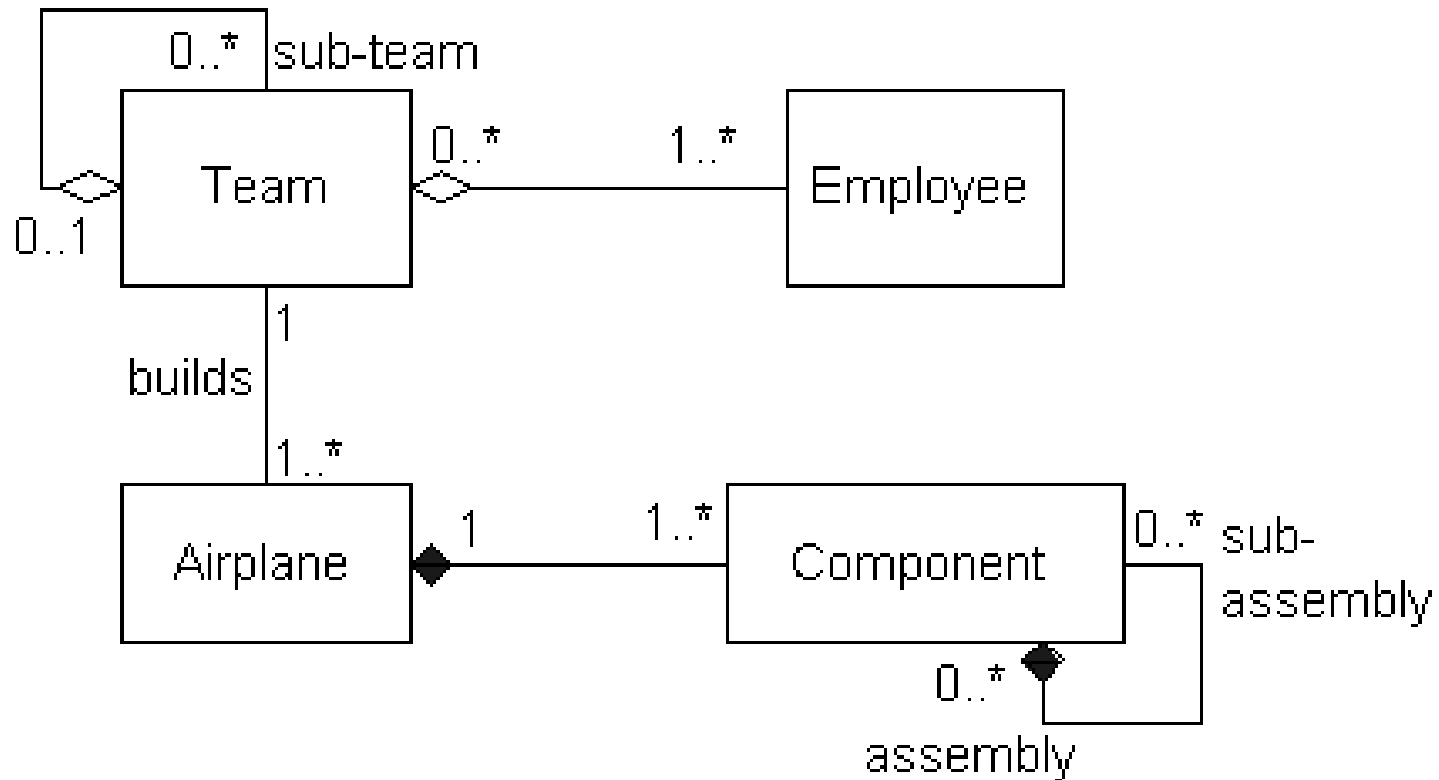
# Diagrama de classes

- Mostra les classes (tipus d'objecte)
  - Atributs (+ públics, - privats)
  - Mètodes (operacions)
- Orientat a objectes.
- Facilita la especificació i implementació.
- Relacions:
  - Agregació: relació més laxa o opcional.
  - Composició: relació més forta de dependència.
- Herència (fletxa buida).

# Diagrama de classes 1



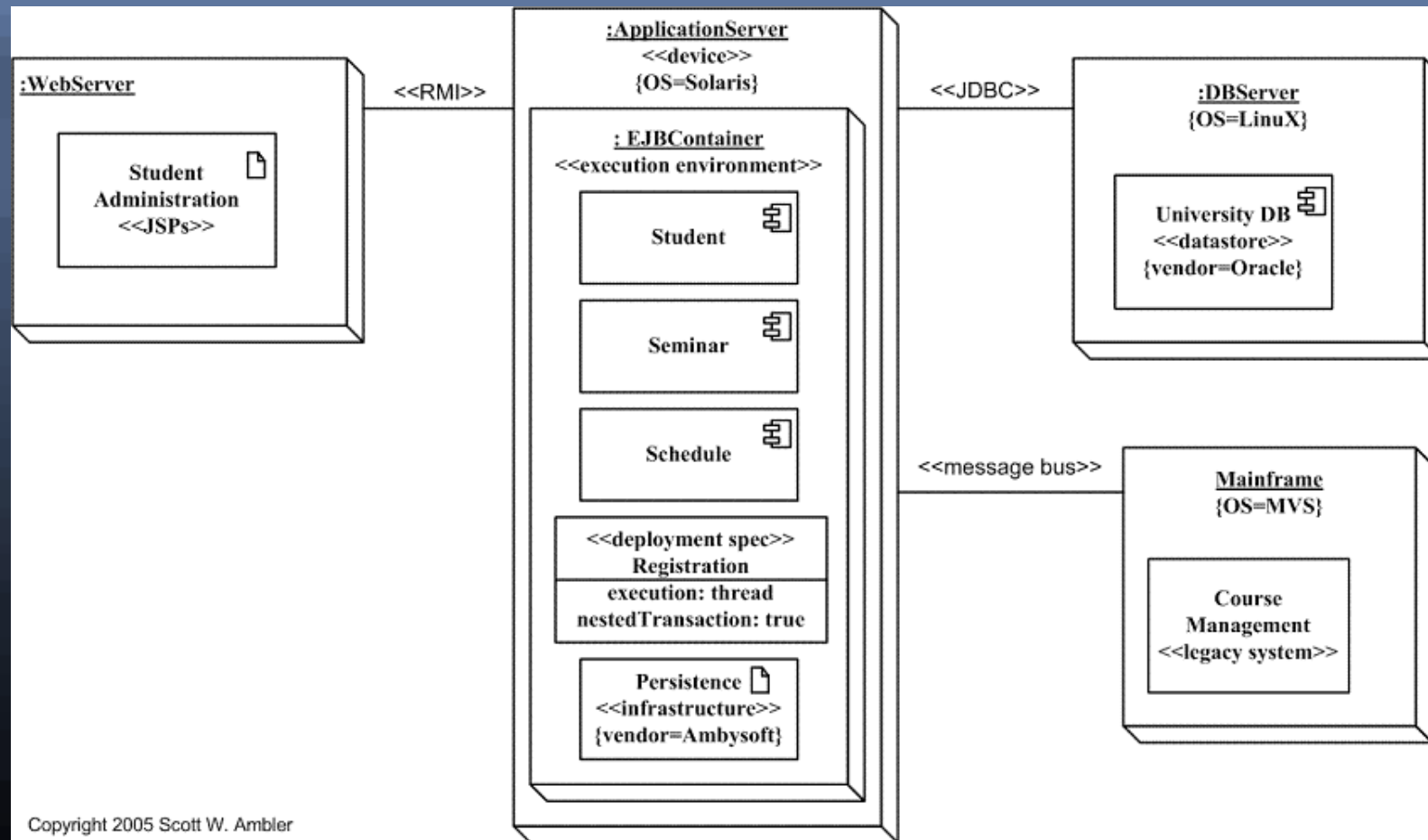
# Diagrama de classes 2



# Diagrama de desplegament

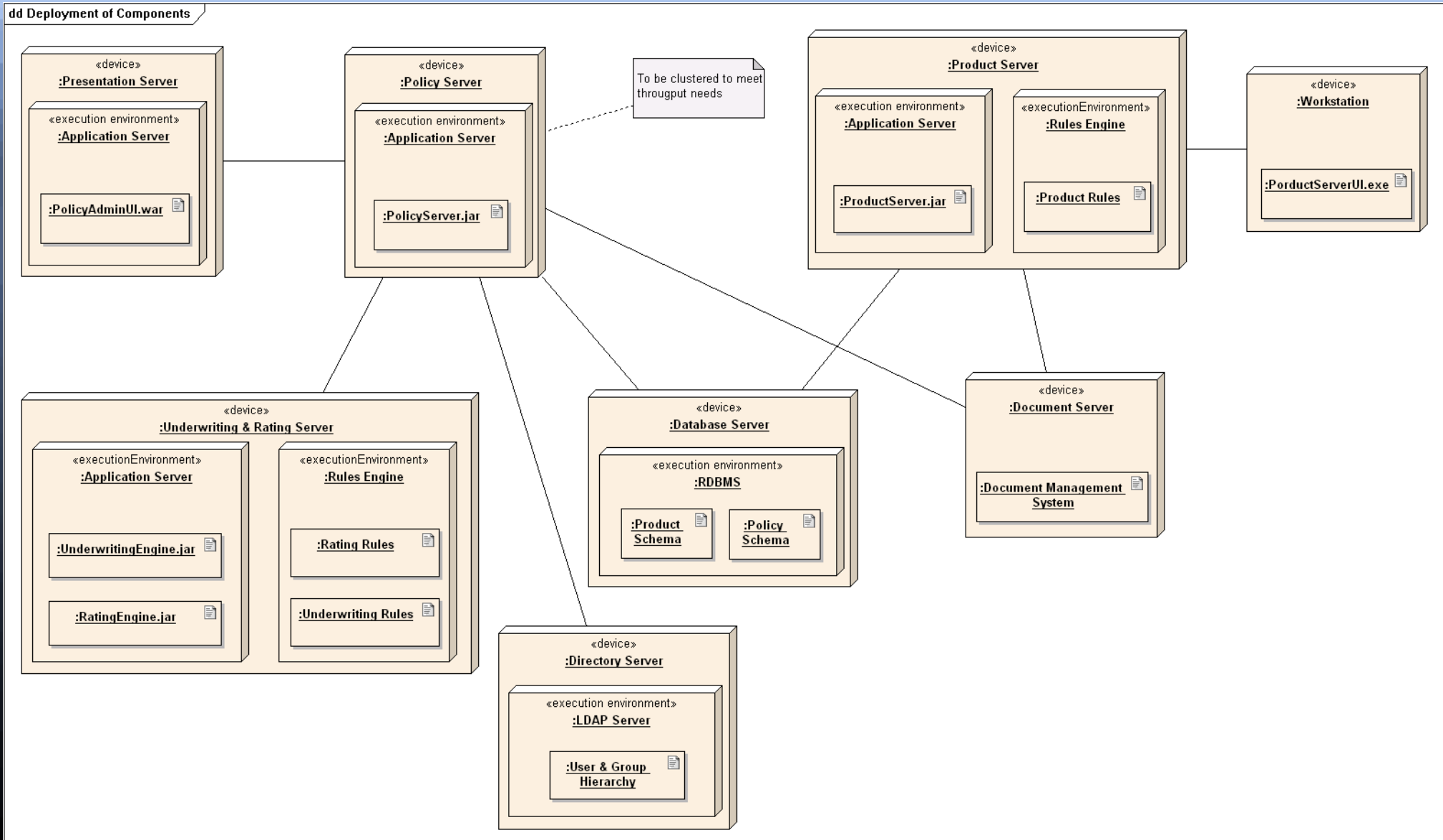
- Mostra una visió estàtica del hardware i/o serveis implicats en sistema de software.
- S'estructura en *nodes*.
- Cada *node* pot córrer una aplicació o servei.
- Es mostren les relacions entre els elements.

# Diagrama de desplegament 1

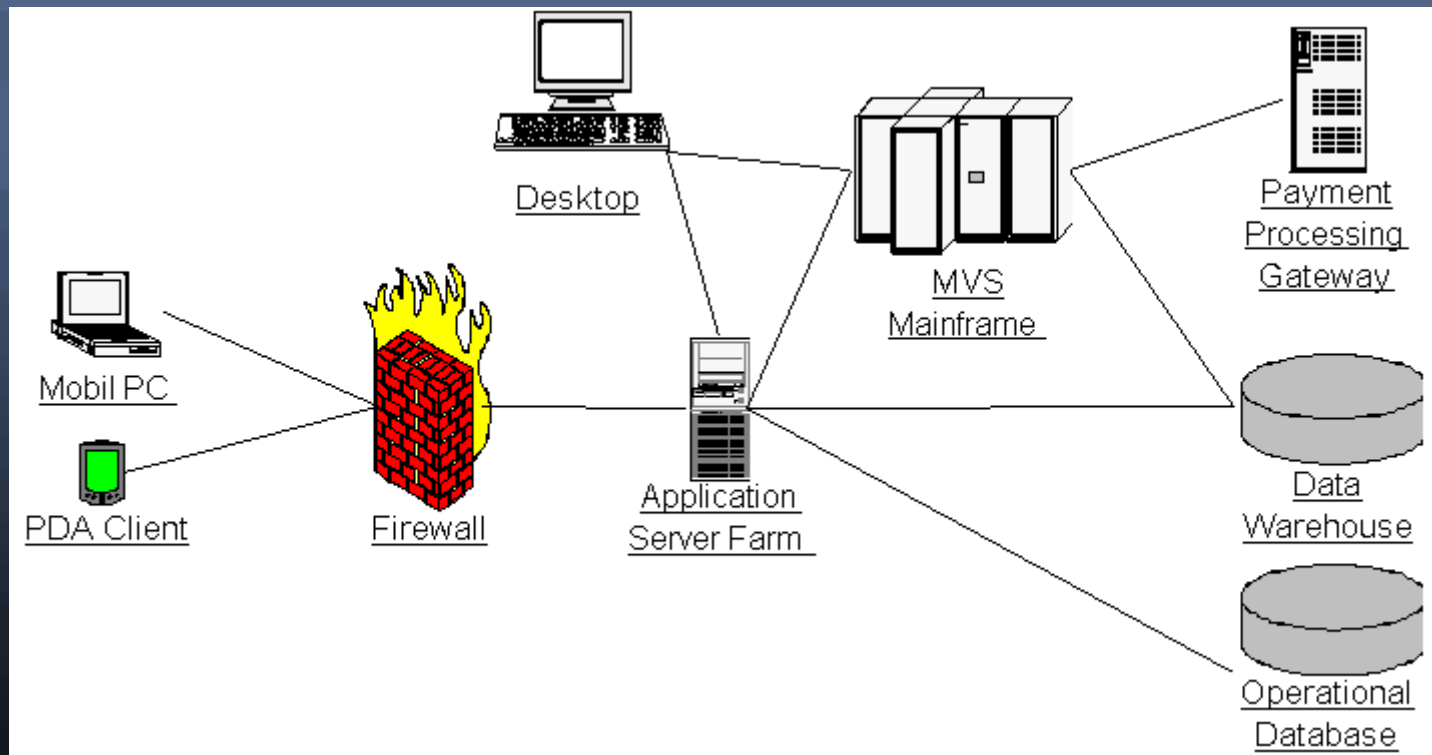




# Diagrama de desplegament 2



# Diagrama de desplegament 3



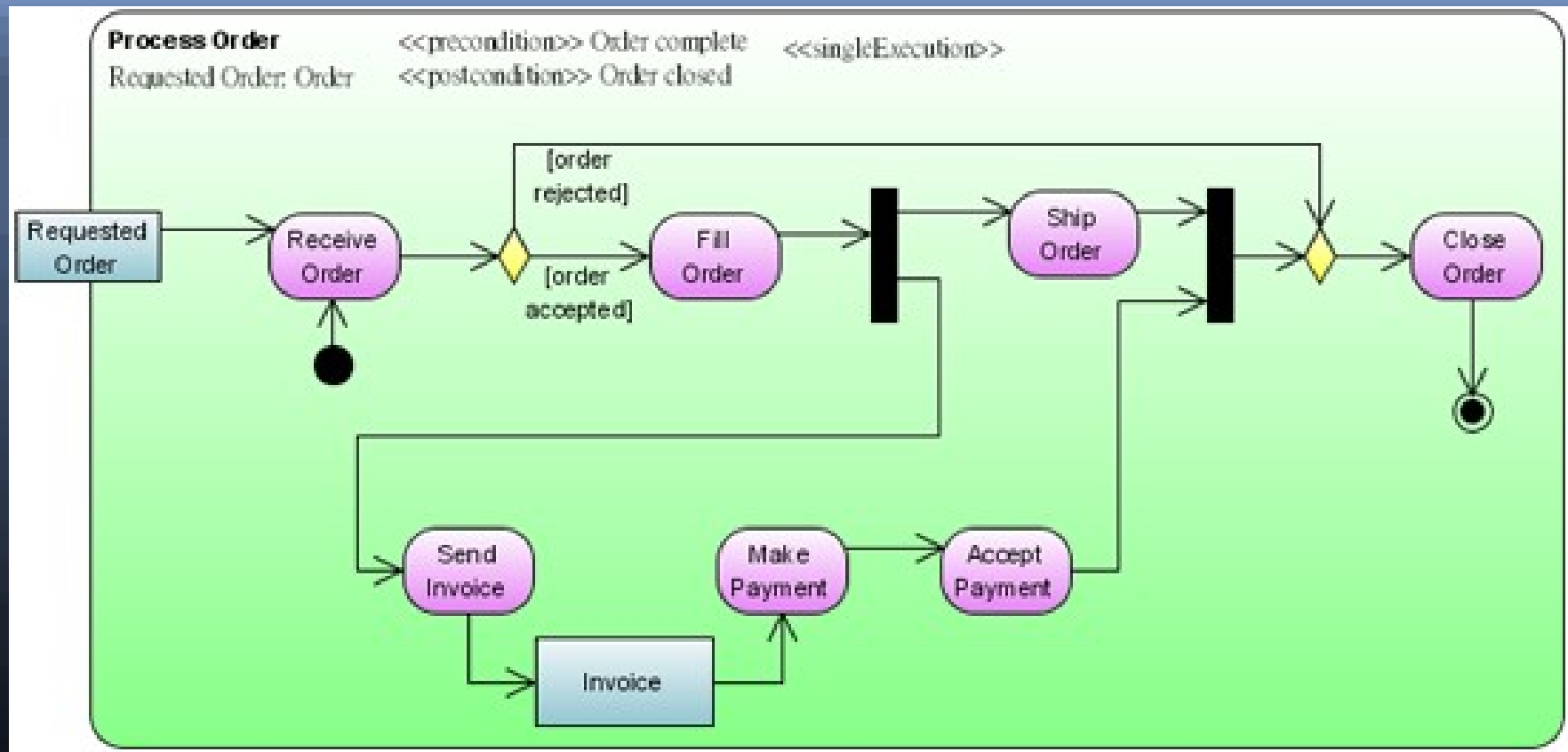
# Tipus de diagrama UML

- Estructural
  - Classes
  - Components (llibreries)
  - Desplegament
  - Objectes (instàncies)
- Comportamental (behavioral)
  - Activitat (flux)
  - Estat
  - Casos d'ús
  - Interacció
    - Sequència
    - Comunicació
    - Timing

# Diagrama d'activitat

- Descriu el comportament del flux del procés (informàtic o empresarial, etc.)
- Equivalent al clàssic diagrama de flux però versió orientada a objectes.
- [http://en.wikipedia.org/wiki/Activity\\_diagram](http://en.wikipedia.org/wiki/Activity_diagram)
- Elements:
  - Punt inici
  - Punt final
  - Node
  - Acció
  - Decisió (condicional)
  - Fork (multiprocés)
  - Join

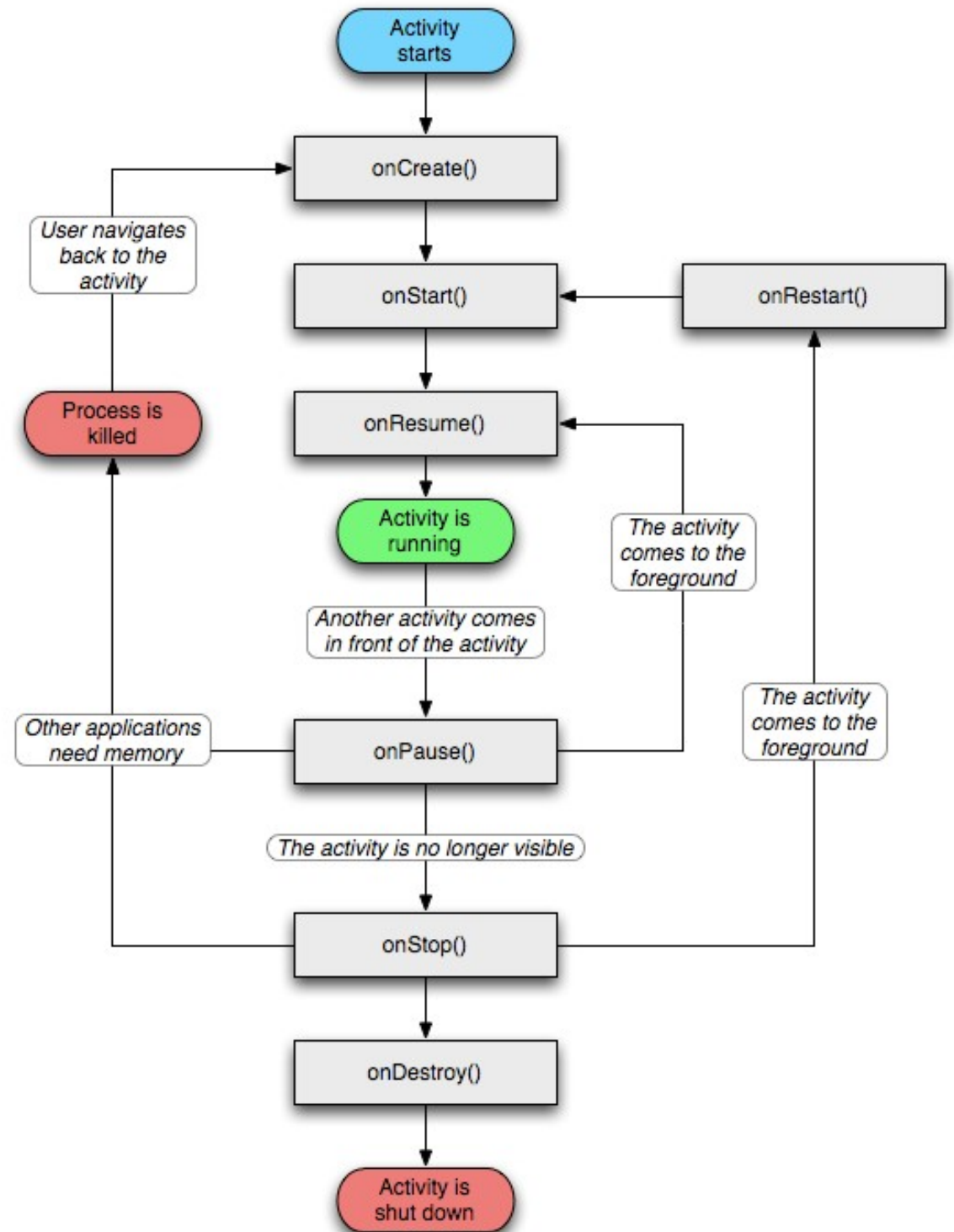
# Diagrama d'activitat 1



# Diagrama d'activitat 2:

## Android “Activity” (app)\*

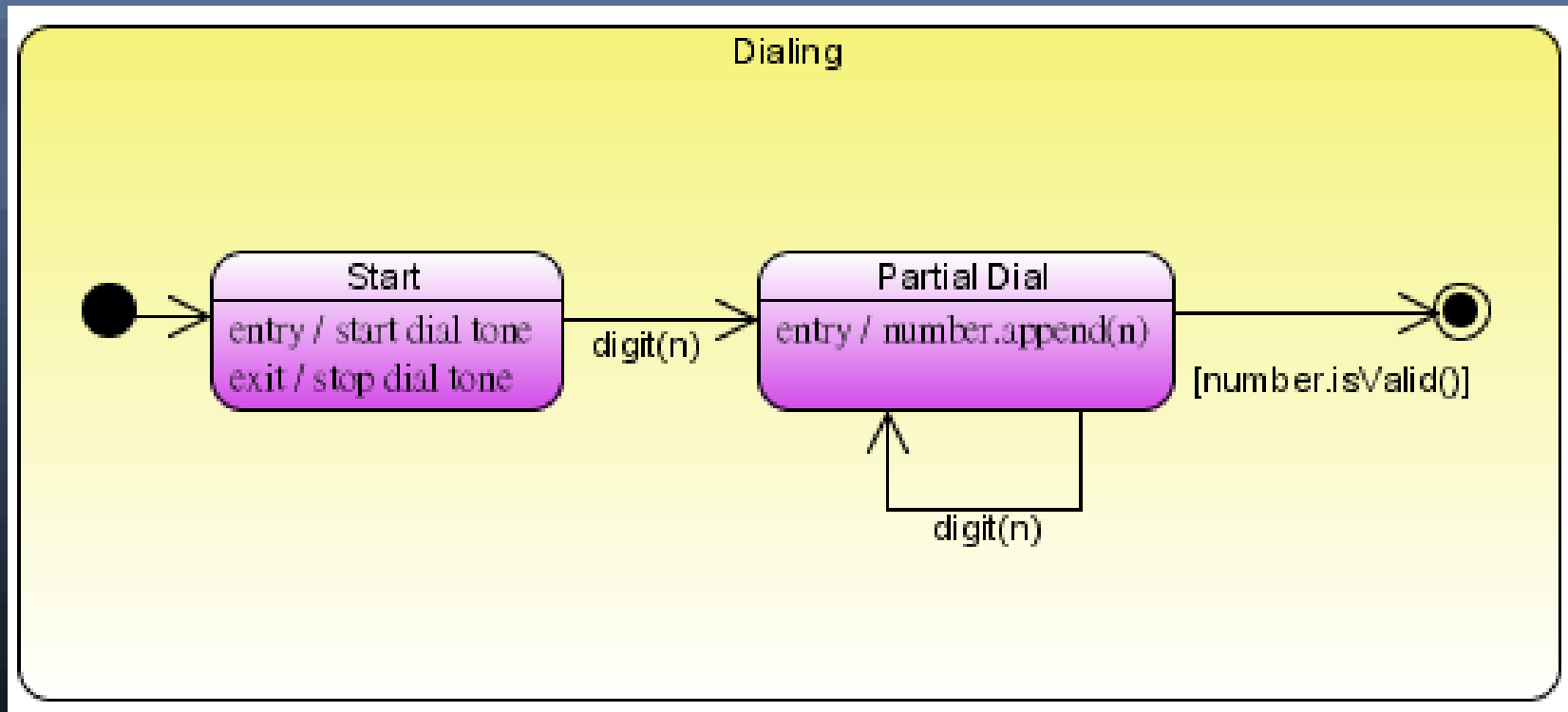
\* És un exemple híbrid poc estàndard, però interessant.



# Diagrama d'estats

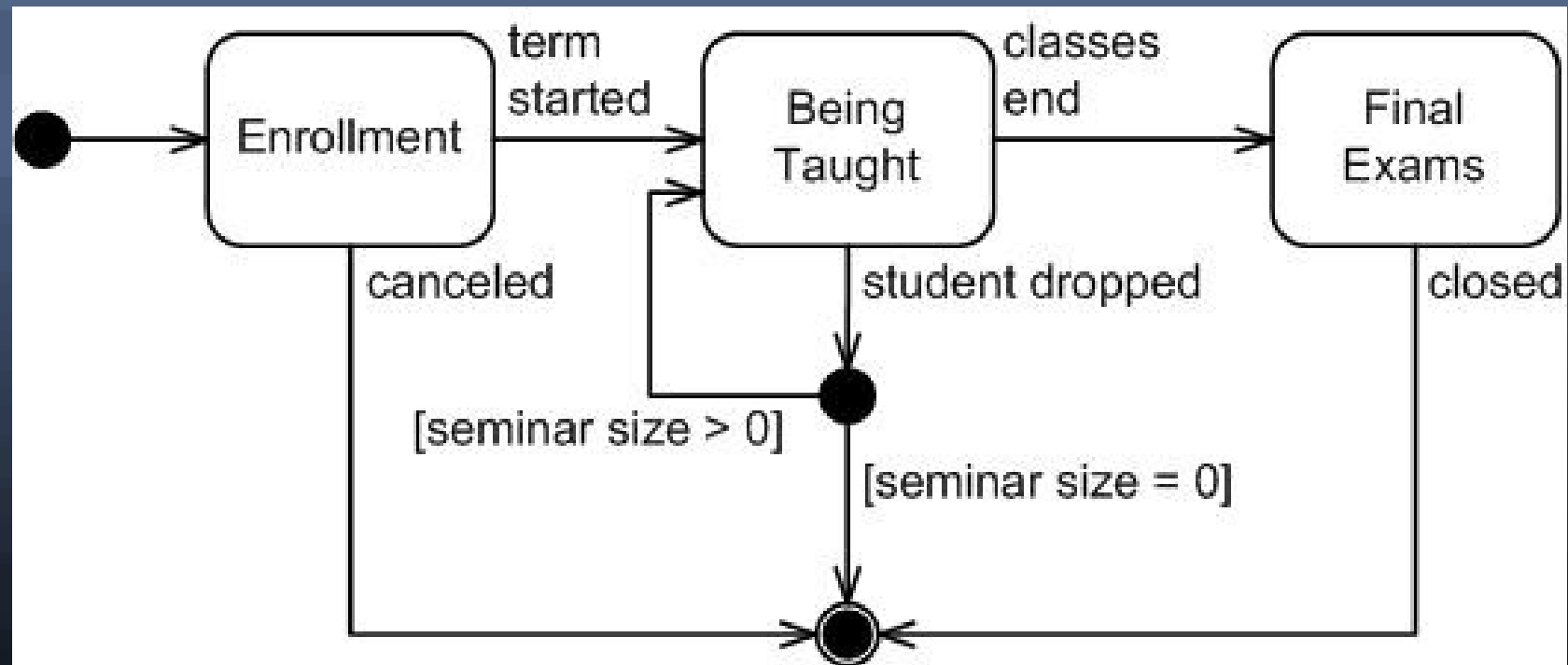
- Màquina d'estats (analogia amb *hardware*).
- Mostra els diferents estats del procés o sistema.
- Transicions: els events canvien l'estat del sistema.

# Diagrama d'estats 1





# Diagrama d'estats 2

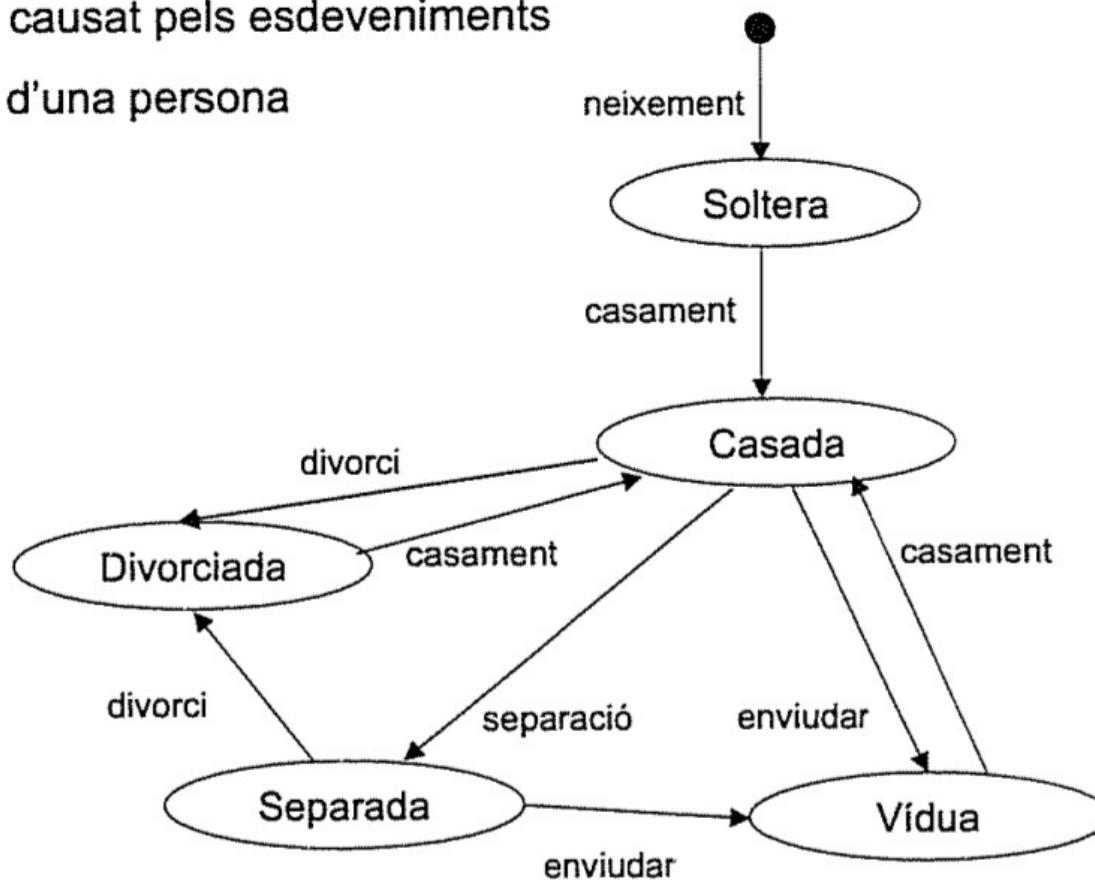


# Diagrama d'estats 3

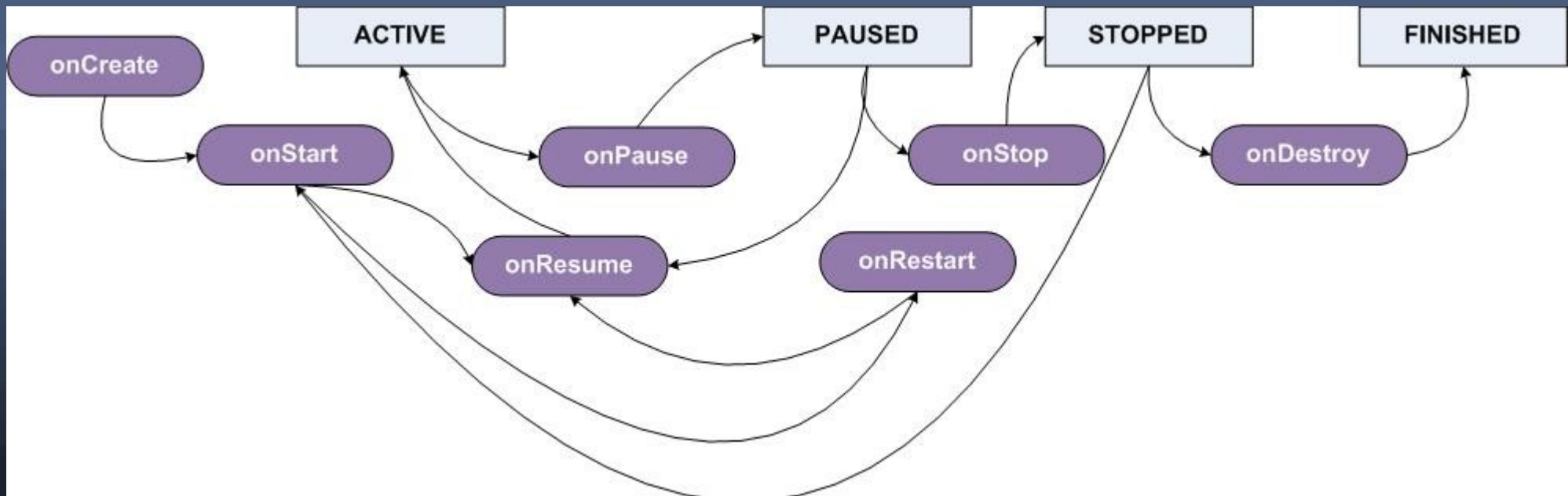
## Descripció del Comportament (III)

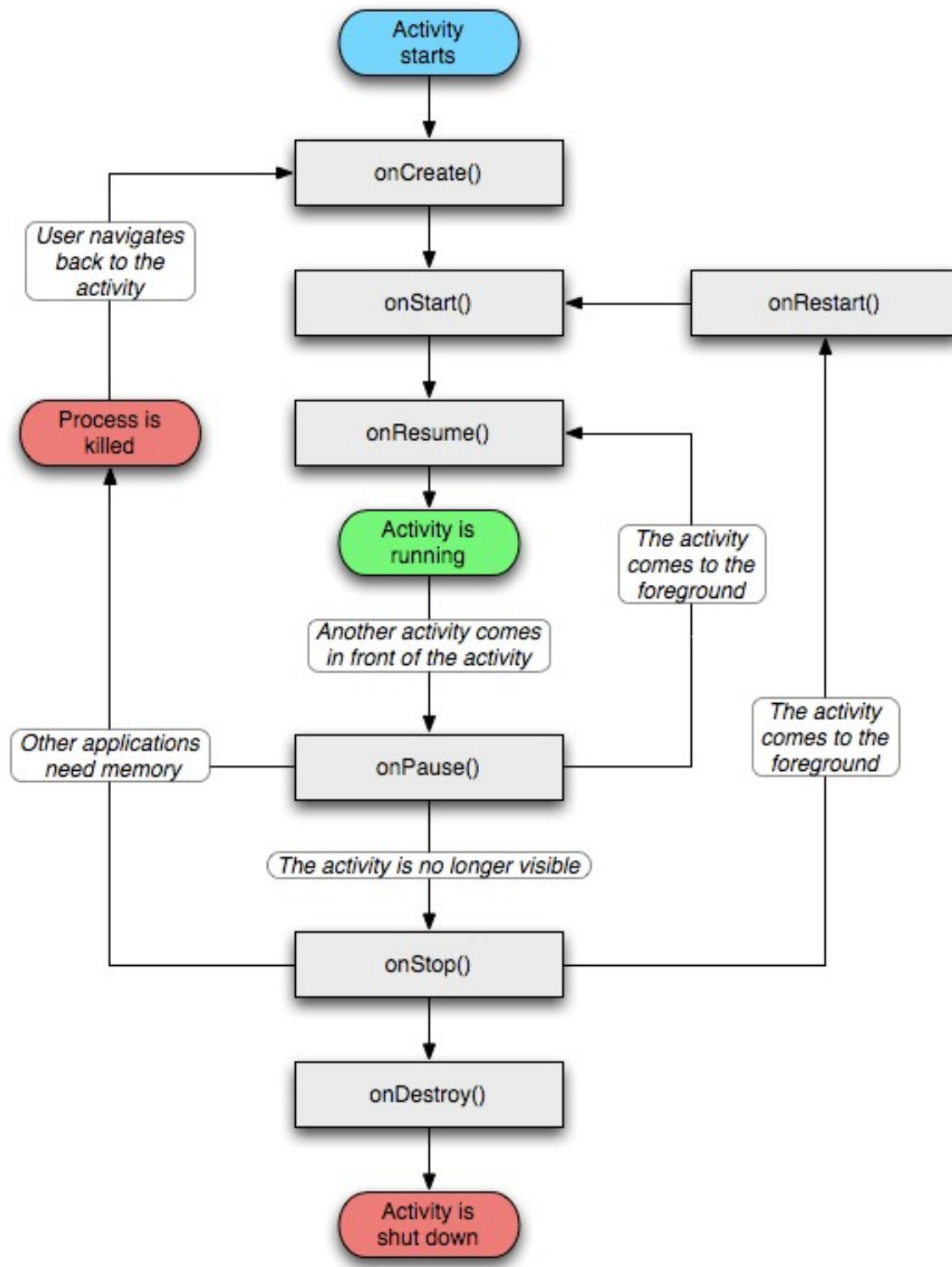
**Diagrama de transició d'estats:** especifica el canvi d'estat d'un objecte causat pels esdeveniments

**Exemple:** estat civil d'una persona



# Diagrama d'estats 4: Android “Activity” (app)





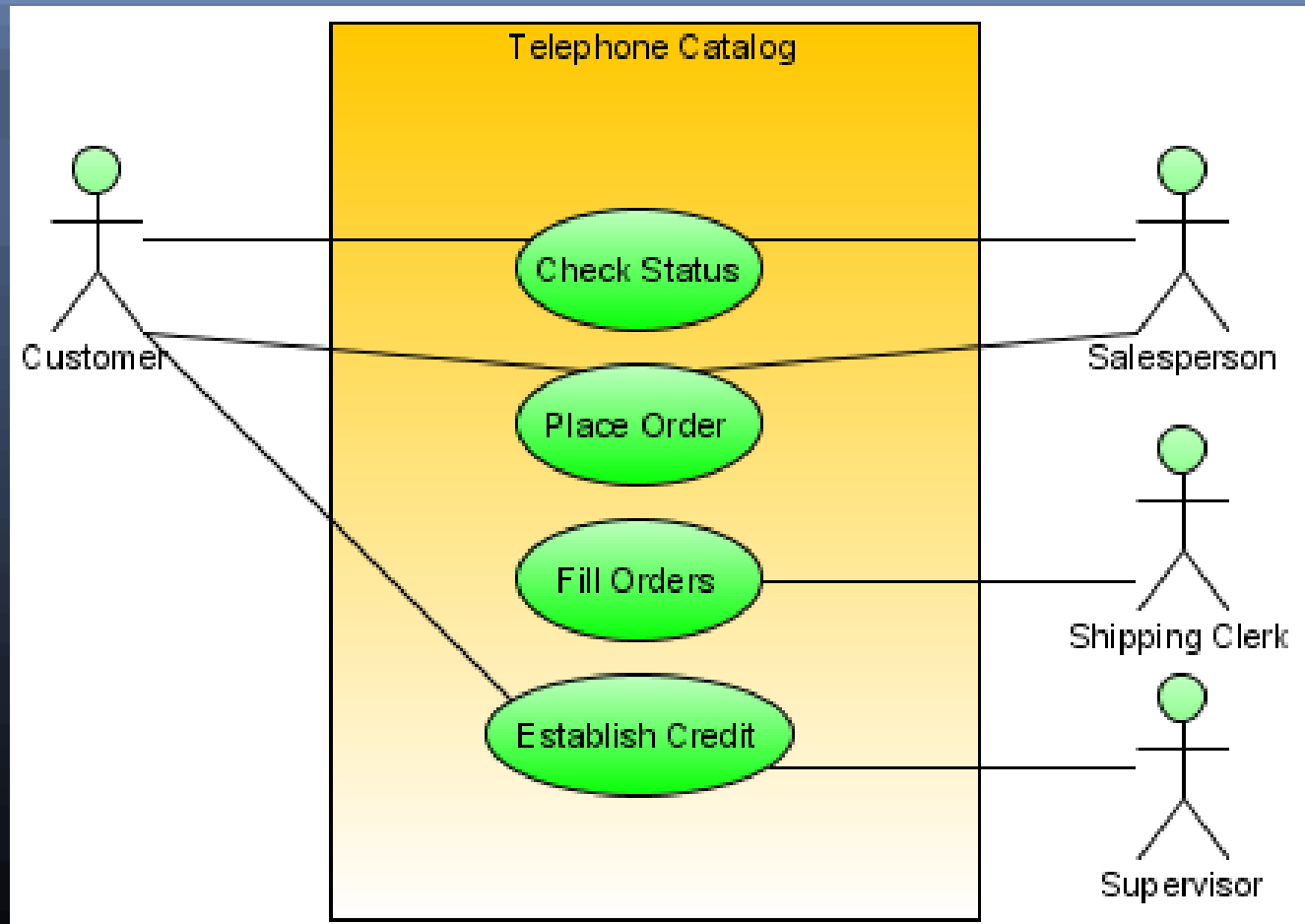
# Diagrama de casos d'ús

- Descriu el comportament “extern”.
- Serveix per establir els requeriments.
- Elements:
  - Actors: persona, organització o sistema que pot interaccionar amb el sistema (genera events).
  - Casos d'ús: accions que tenen un resultat visible o mesurable per l'actor.
  - Associacions: entre actors i casos d'ús.

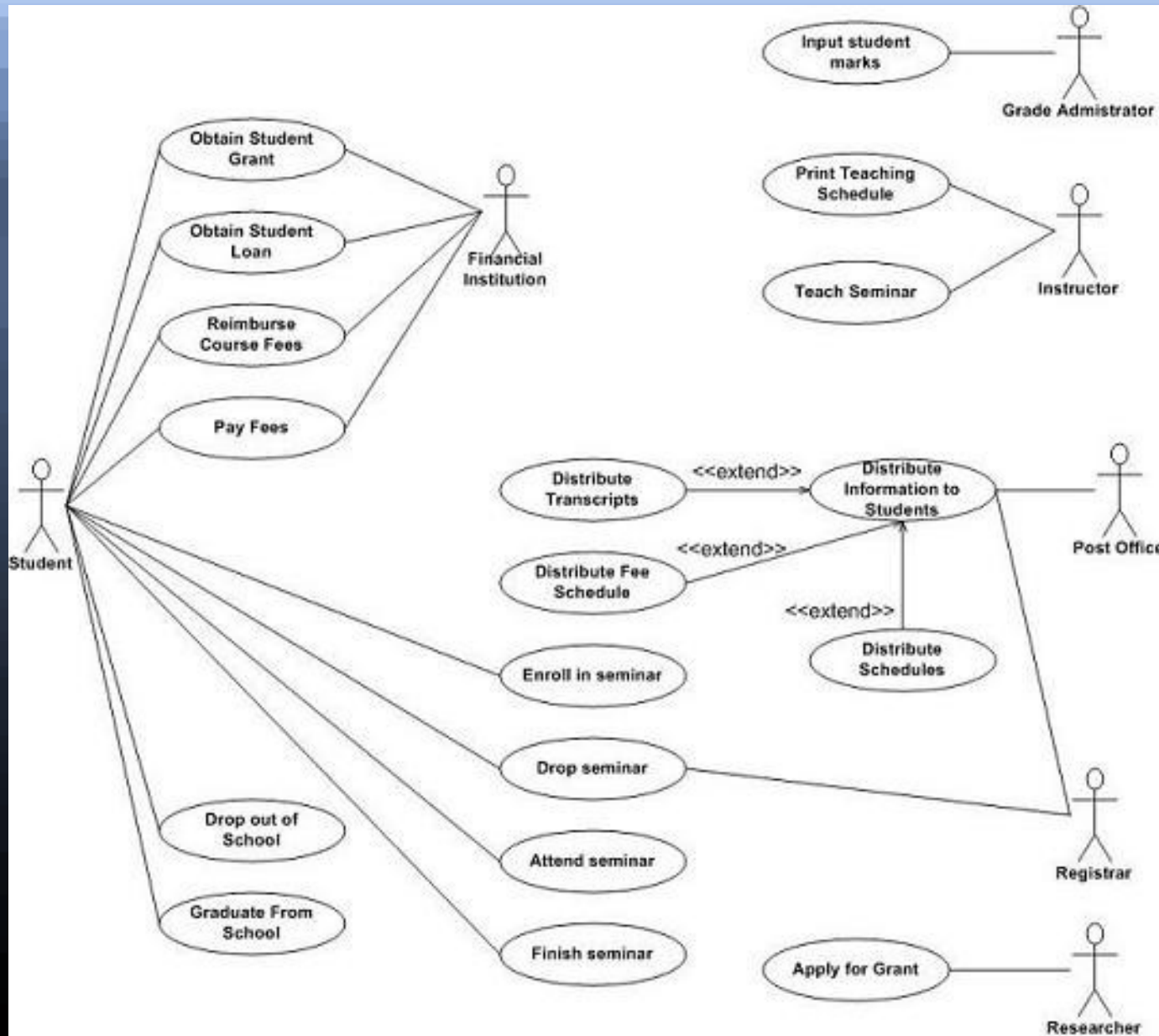
# Casos d'ús 1



# Casos d'ús 2



# Casos d'ús 3

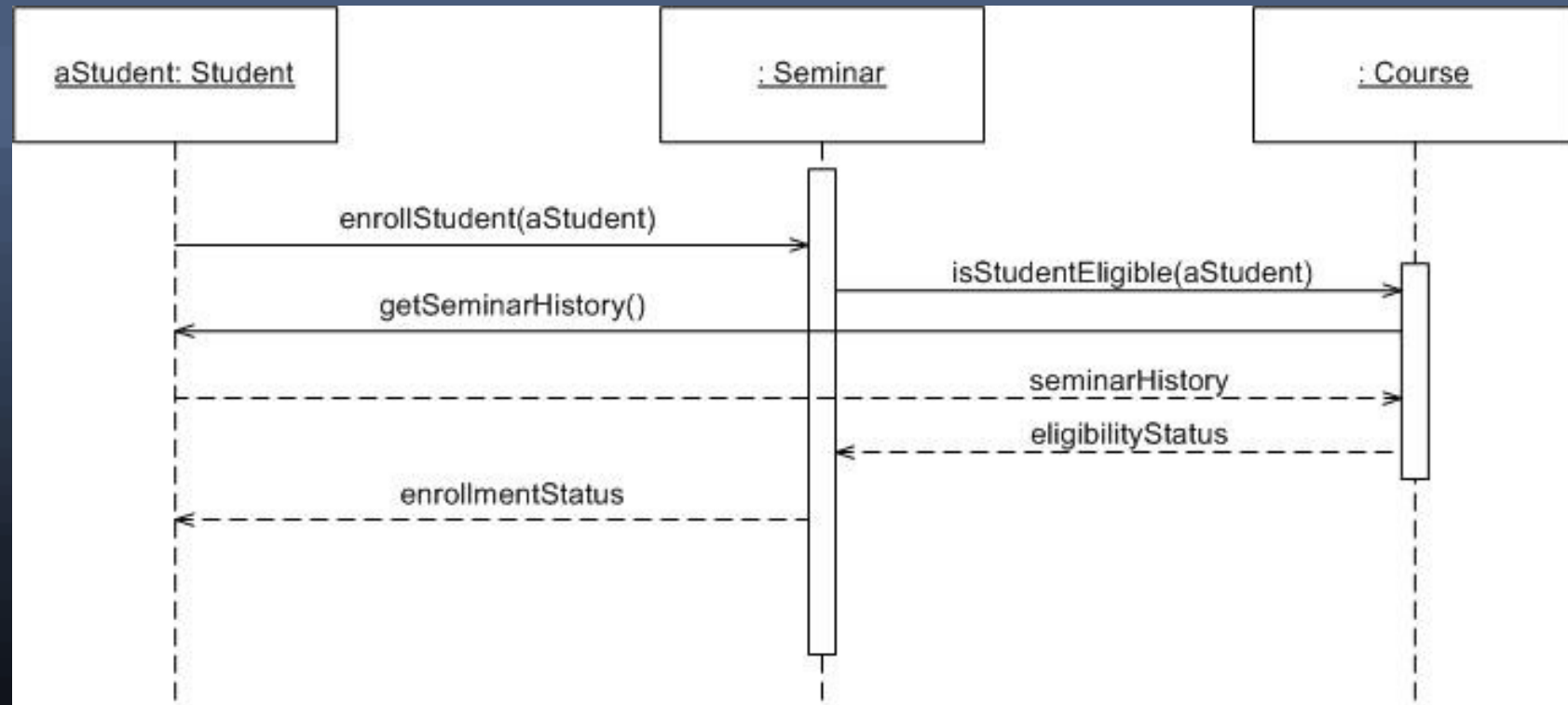




# Diagrama de seqüència

- Modela la col·laboració d'objectes basat en línia de temps o *timeline*.
- Mostra un “escenari” d'un dels casos d'ús concret.
- Elements:
  - Classes i funcions
  - Crides síncrones (fletxa plena): el procés que crida espera a que s'acabi l'operació.
  - Crides assíncrones (fletxa oberta): el procés que crida retorna immediatament. La resposta pot succeir en un temps qualsevol després.

# Diagrama de seqüência 1



# Diagrama de seqüência 2

