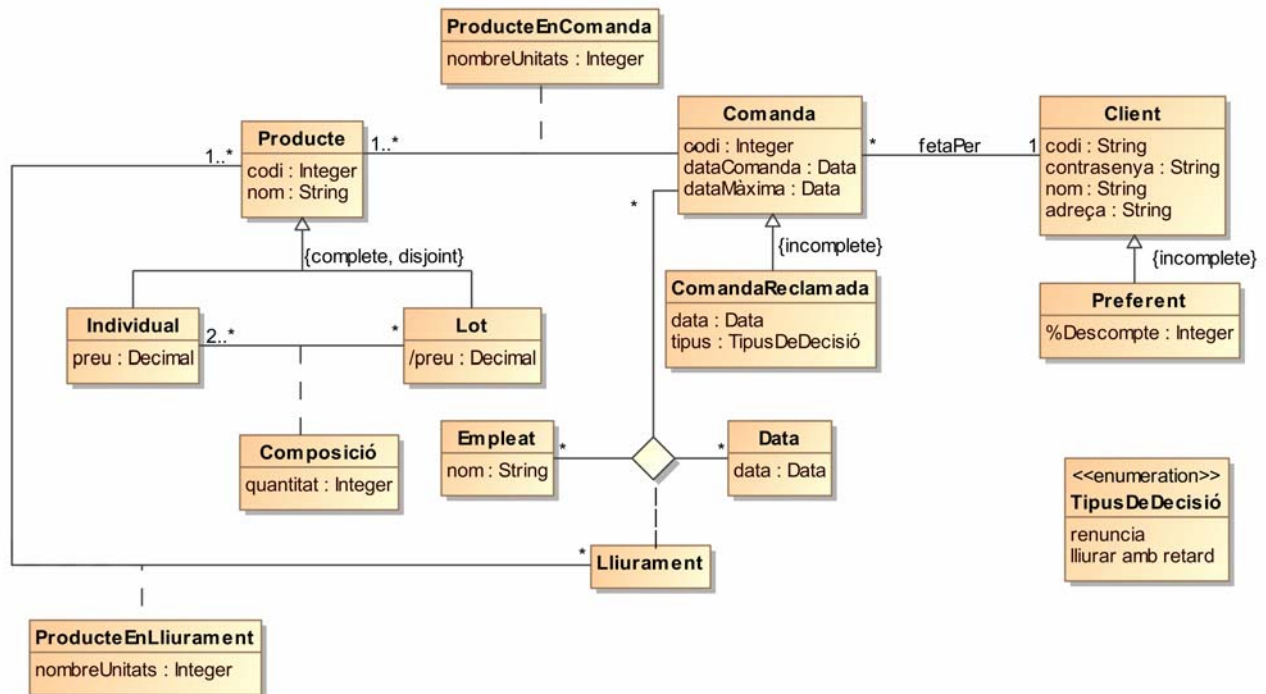


1er Problema – Model Conceptual (3,5 punts)

**Restriccions Textuals:**

1. Claus externes de classes no associatives:
(Producte, codi); (Comanda, codi); (Client, codi i contrasenya); (Data, data); (Empleat, nom)
2. La data màxima d'una comanda ha de ser posterior o igual a la data de la comanda.
3. Un client no pot fer més de 5 comandes en una mateixa data.
4. La data de lliurament d'una comanda ha de ser posterior o igual a la data de la comanda.
5. Els productes d'un lliurament d'una comanda pertanyen a la seva comanda.
6. La suma de les quantitats dels productes lliurats d'una comanda no és superior a la suma de les quantitats dels productes demanats en la comanda.
7. La data de reclamació d'una comanda ha de ser posterior a la data màxima de la comanda.

Informació derivada:

/preu d'un lot és igual a la suma de: $\text{preu} \times \text{quantitat}$ dels productes individuals que componen el lot.

2on Problema – Model del Comportament (3,5 punts)

Crear acte (0,8 punts)

Operació: `nouActe(ta:String, an:Any, ne:String, da:Data, hi:Integer, hf:Integer, acts:Set(String), patrs:Set(String))`

Precondició:

- `TipusActe.allInstances()->exists(t|t.nom=ta)`
- `Any.allInstances()->exists(a|a.any=an)`
- `an > currentyear()`
- `Espai.allInstances()->exists(e|e.nom=ne)`
- `Activitat.allInstances().nom->includesAll(acts)`
- `Patrocinador.allInstances().nom->includesAll(patrs)`
- `not Acte.allInstances()->exists(ac|ac.tipusActe.nom=ta and ac.any.any=an)`

Postcondició: `a.oclIsNew()` and `a.oclIsTypeOf(Acte)` and `a.tipusActe.nom=ta` and `a.any.any=an` and `a.espai.nom=ne` and `a.data=da`, and `a.horaIni=hi` and `a.horaFi=hf` and `a.activitat.nom->includesAll(acts)` and `a.patrociandor.nom->includesAll(patrs)`

Afegir participant (1,2 punts)

Operació: `afegirPart(np:String, ta:String, vol:Boolean):Participació`

Precondició:

- `Acte.allInstances()->exists(a|a.tipusActe.nom=ta and a.any.any=currentyear() and a.data > today() and a.persona.nom->excludes(np))`

Postcondició: `if not Persona.allInstances()@pre ->exists(pr|pr.nom=np) then per.oclIsNew() and per.oclIsTypeOf(Persona) and per.nom=np endif and p.oclIsNew() and p.persona.nom=np and p.acte.tipusActe.nom=ta and p.acte.any.any=currentyear() and if vol then p.oclIsTypeOf(Voluntari) else p.oclIsTypeOf(Assistent) endif`

Sortida: `result = p`

Operació: `afegirTorn(p:Participació, ho:Hora, na:String)`

Precondició:

- `Hora.allInstances()->exists(h|h.hora=ho)`
- `Activitat.allInstances()->exists(a|a.nom=na)`
- `p.oclIsTypeOf(Voluntari)`
- `not Torn.allInstances()->exists(t|t.voluntari=p and t.hora.hora=ho and t.activitat.nom=na)`

Postcondició: `t.oclIsNew()` and `t.oclIsTypeOf(Torn)` and `t.voluntari=p` and `t.hora.hora=ho` and `t.activitat.nom=na`

Consultar persones (0,7 punts)

Operació: `consultaPers(an:Any): Set(TupleType(np:String, infActes:Set(TupleType(ta:String, ne:String))))`

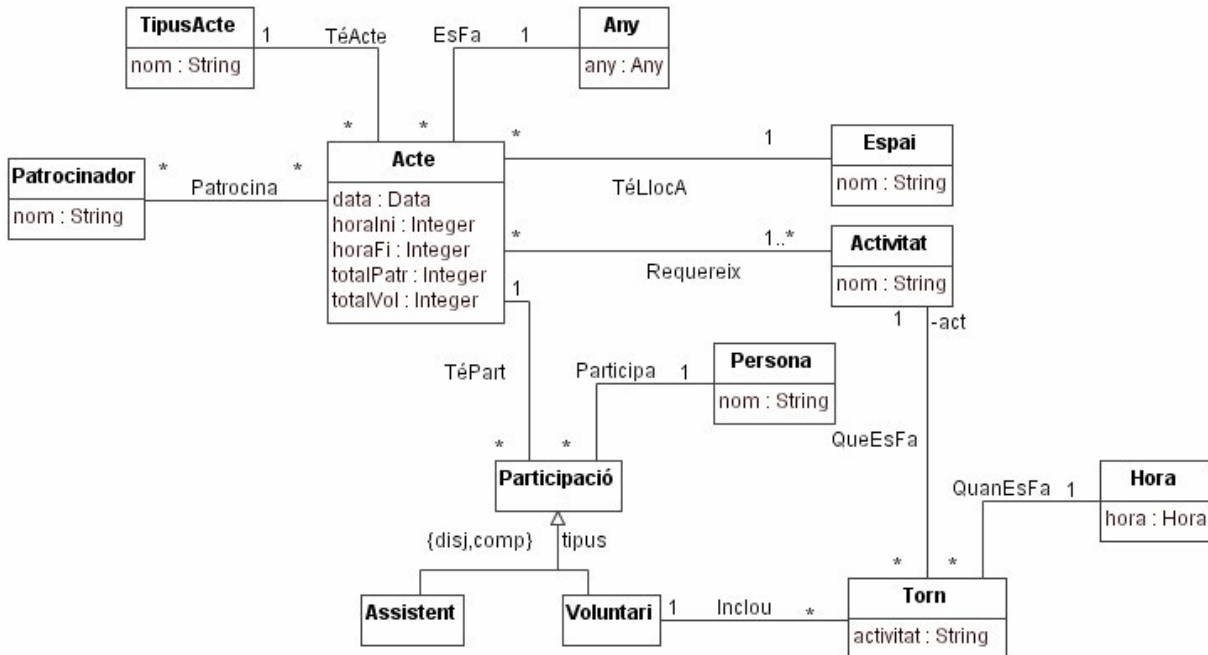
Precondició: `-Any.allInstances()->exists(a|a.any.any=an)`

Sortida:

```
let pers:Set(Persona) = Persona.allInstances()->
select (p|p.participació->
select (v|v.oclIsTypeOf(Voluntari)).acte.any.any->excludes(an)
and p.participació->select(pa|pa.acte.any.any=an)->size()>3)
in
result = pers->collect(p| Tuple{np=p.nom,
```

```
infActes=p.acte->select(ac|ac.any.any=an)-> collect(a|
  Tuple{ta=a.tipusActe.nom, ne=a.espai.nom})})
```

Normalització (0,8 punts)



RI Textuals afegides:

- No hi pot haver dos actes amb el mateix tipus d'acte i any
- No hi pot haver dues participacions amb el mateix acte i persona
- No hi pot haver dos torns amb el mateix voluntari, activitat i hora
- No hi pot haver més de dos torns del mateix voluntari i hora

Normalització del contracte nouActe:

- Afegir a la precondició:
 - L'hora d'inici de l'acte és anterior a la seva hora de fi
 - La data de l'acte pertany a l'any de l'acte
 - A l'espai de l'acte no hi ha cap altre acte simultani
 - 'acts' té al menys una activitat
- Afegir a la postcondició:
 - totalPatr pren per valor el nombre d'elements de 'patrs'
 - totalVol pren per valor 0

Normalització del contracte afegirPart:

- Afegir a la precondició:
 - La persona no participa en cap altre acte que es faci alhora
- Afegir a la postcondició:
 - Si la participació és com a voluntari, totalVol de l'acte s'incrementa en 1

Normalització del contracte afegirTorn:

- Afegir a la precondició:
 - No hi ha ja dos torns del mateix voluntari i hora
 - L'activitat és una de les que requereix l'acte del voluntari
 - L'hora del torn està entre l'hora d'inici i de fi de l'acte corresponent al torn
- Afegir a la postcondició:
 - activitat pren per valor el nom de l'activitat del torn

La resta queda igual.