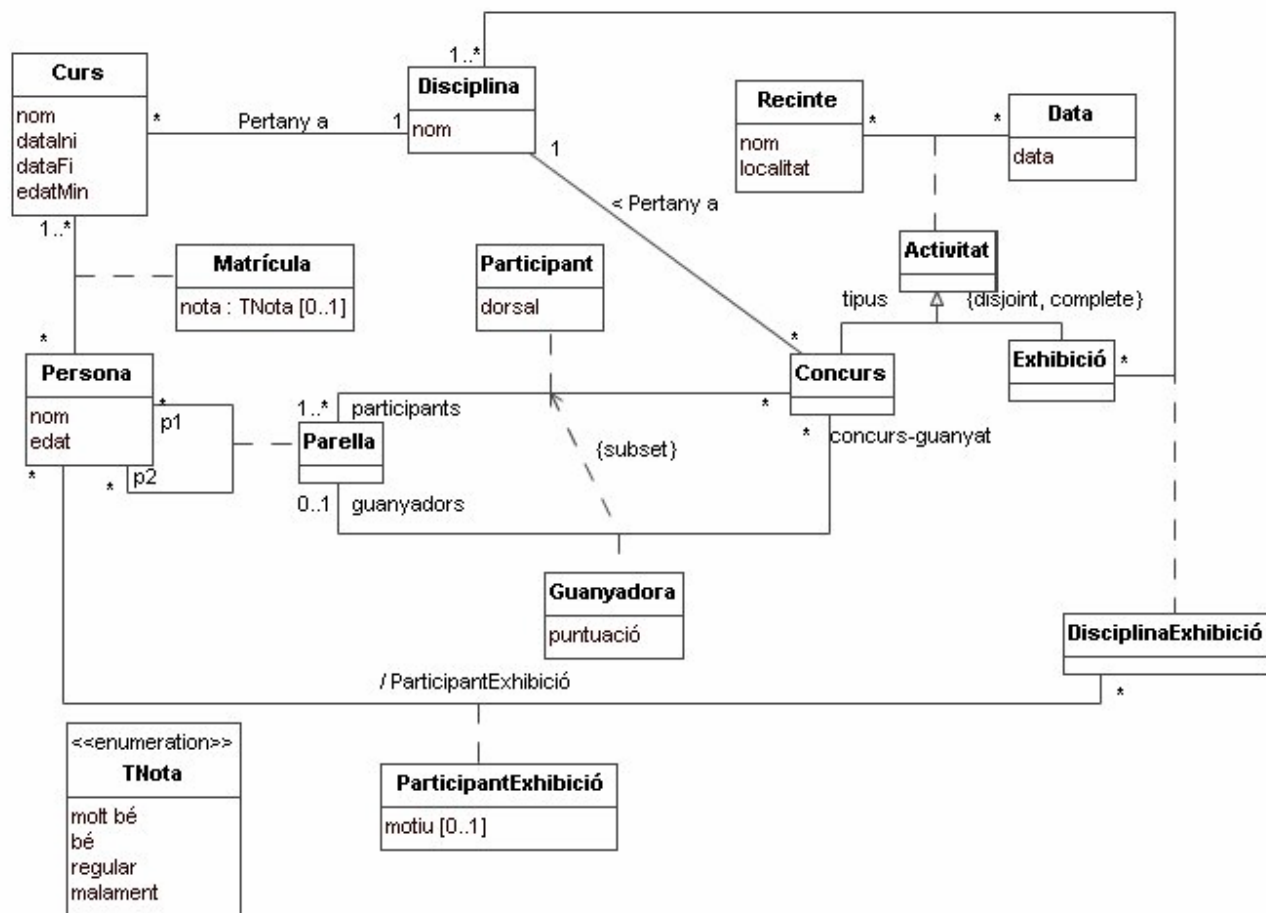


## 1r Problema – Model Conceptual de les Dades (3,5 punts)



## Restriccions textuais

1. Claus externes: (Curs, nom), (Disciplina, nom), (Persona, nom), (Recinte, nom), (Data, data)
2. La data d'inici d'un curs és anterior a la data de final
3. L'edat dels alumnes que es matriculen a un curs ha de ser superior a l'edat mínima del curs
4. Una persona no pot fer dos cursos que coincideixin en alguna data
5. Els membres d'una parella han de ser diferents
6. Una persona no pot participar més d'una vegada al mateix concurs (com a membre de parelles diferents)
7. Els membres de les parelles participants a un concurs han d'haver fet com a mínim un curs de la disciplina del concurs
8. Una persona no pot participar a més d'un concurs i/o exhibició en la mateixa data
9. En un concurs no hi ha dorsals repetits

## Informació derivada

1. Els alumnes que participen a una disciplina d'una exhibició són tots els que han tret un "Molt bé" en algun curs d'aquesta disciplina

---

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

---

### Creació de nova festa (1,2 punts)

- Operació:** `novaFesta(nomLloc:String, data:Data, codiEmp:String, tipusFesta:String): Festa`
- Semàntica:** Crea una nova instància de Festa amb les dades indicades
- Precondició:**
- `Lloc.allInstances()->exists(l1|l1.nom=nomLloc)`
  - `Data.allInstances()->exists(d|d.data=data)`
  - `Empresa.allInstances()->exists(e|e.codi=codiEmp)`
  - `not Festa.allInstances()->exists(fe|fe.lloc.nom=nomLloc and fe.data.data=data)`
- Postcondició:** `f.oclIsNew()` and `f.oclIsTypeOf(Festa)` and `f.lloc.nom=nomLloc` and `f.data.data=data` and `f.empresa.codi=codiEmp` and `f.tipusFesta.nom=tipusFesta`
- Sortida:** `result = f`
- Operació:** `inscriureOrganitzador(festa:Festa, nom:String)`
- Semàntica:** Es crea una associació entre la festa i l'organitzador i es crea una instància d'inscripció.
- Precondició:**
- `Persona.allInstances()->exists(p|p.nom=nom)`
  - `not festa.participant->exists(or|or.nom=nom)`
- Postcondició:** `festa.organitzador.nom->includes(nom)` and `i.oclIsNew()` and `i.oclIsTypeOf(Inscripció)` and `i.festa=festa` and `i.participant.nom=nom`

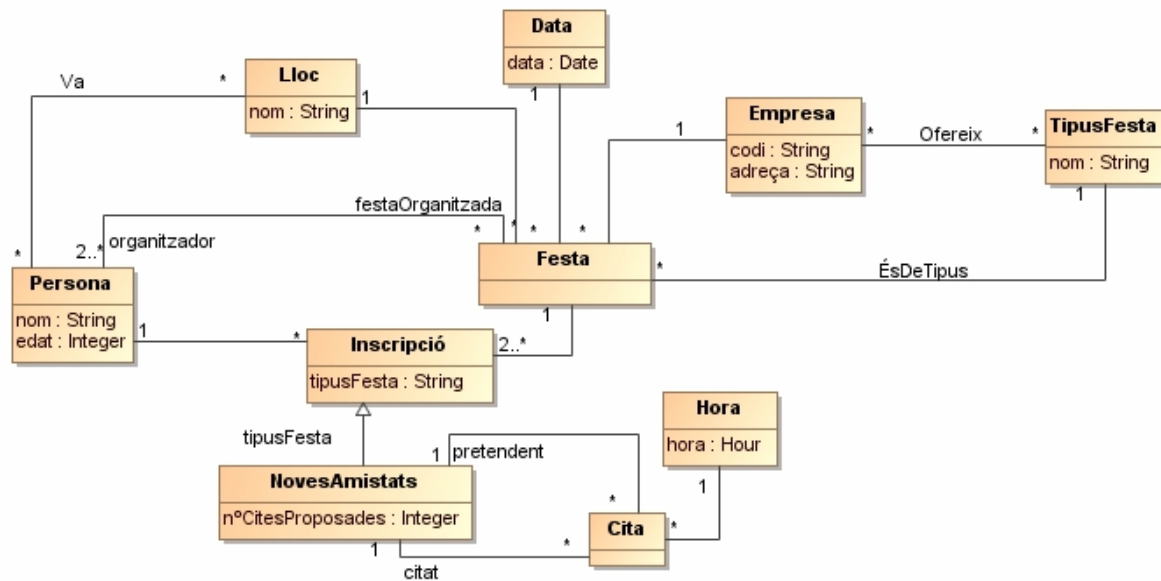
### Cancel·lar cita (0,7 punts)

- Operació:** `cancel·larCita(nomLloc:String, dataFesta:Date, nomPret:String, horaCita:Hour)`
- Semàntica:** Esborra la cita de la que s'han passat les dades com a paràmetres.
- Precondició:**
- `Cita.allInstances()->exists(ci|ci.pretendent.participant.nom=nomPret and ci.pretendent.festa.lloc.nom=nomlloc and ci.pretendent.festa.data.data=dataFesta and ci.hora.hora=horaCita) and dataFesta>today()`
- Postcondició:**
- `Cita.allInstances()->select(ci|ci.pretendent.participant.nom=nomPret and ci.pretendent.festa.lloc.nom=nomlloc and ci.pretendent.festa.data.data=dataFesta and ci.hora.hora=horaCita)->isEmpty()`

### Llistar les festes amb èxit (0,8 punts)

- Operació:** `llistarFestesAmbÈxit(): Set(TupleType(codiEmp:String, lloc:String, dataFesta:Date, orgs:Set(TupleType(nomOrganitzador:String, edat:Integer, numCitesProposades:Integer))))`
- Semàntica:** Retorna un llistat de totes les festes de 'NovesAmistats' amb més de 50 inscrits, descartant organitzadors i els noms, edat i cites dels organitzadors.
- Sortida:**
- ```
let festesAmbÈxit = Festa.allInstances()->select (f|
f.tipusFesta.nom='NovesAmistats' and
(f.participant->size()-f.organitzador->size())>50 in
result = festesAmbÈxit->collect(f| Tuple{codiEmp=f.empresa.codi,
lloc=f.lloc.nom, dataFesta=f.data.data,
orgs=f.organitzador->collect(or|
Tuple{nomOrganitzador=or.nom, edat=or.edat,
numCitesProposades= or.inscripcio
->select(ins|ins.festa=f).oclAsType(NovesAmistats).
n°CitesProposades}}))
```

## NORMALITZACIÓ (0,8 punts)



### RI Textuals afegides:

- No hi pot haver dues festes al mateix lloc i en la mateixa data
- No hi pot haver dues inscripcions de la mateixa persona a la mateixa festa
- Per a qualsevol persona que organitza una festa ha d'existir una inscripció de la persona a la festa

### Normalització contractes

#### Contracte NovaFesta:

afegir a la precondition:

- No pot ser que ja hi hagi una festa en el lloc i la data que se li han passat com a paràmetres
- El paràmetre tipusFesta ha de ser igual al nom d'algun dels tipus de festa oferts per l'empresa amb el codi que s'ha passat com a paràmetre

#### Contracte InscriureOrganitzador:

afegir a la postcondició:

- Crear una instància de l'associació Va entre l'organitzador i el lloc a on es fa la festa
- Assignar a l'atribut tipusFesta de la inscripció creada el nom del tipus de festa de la festa
- Si el tipus de festa és Noves Amistats s'especialitza la Inscripció a NovesAmistats i s'inicialitza l'atribut n°CitesProposades a 0.

La resta queda igual.