

## OO - TRENS.

A la figura de la dreta es mostra el diagrama de classes, ja normalitzat, d'un sistema d'informació per controlar el material mòbil d'una empresa de Ferrocarrils.

### RI textuais:

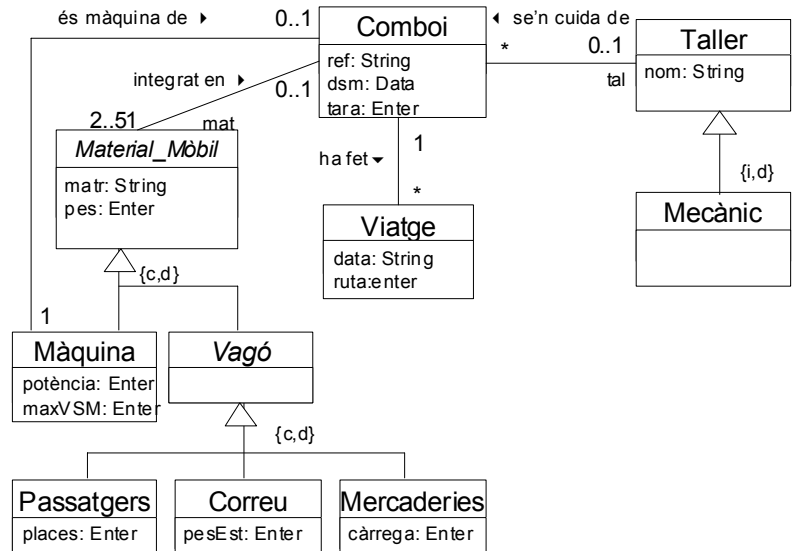
RT1: Claus de les classes: (Material Mòbil, matr), (Comboi, ref), (Taller, nom), (Viatge, data)

RT2: Un comboi té com a mínim un vagó.

RT3: En un comboi, la potència de la màquina no pot ser inferior al **pes total en càrrega** del comboi.

RT4: **Tots els viatges s'han fet després de l'últim manteniment del comboi que ha fet el viatge.**

RT5: L'associació *és\_màquina\_de* està inclosa dins de l'associació *integrat\_en*.



La resta de RI no tenen interès aquí.

El **pes total en càrrega** d'un comboi és la suma del pes en càrrega de la màquina i el dels vagons que integren el comboi. El **pes total en càrrega** de cada tipus de material mòbil es calcula com:

- **Màquina:** És el valor de l'atribut *pes*.
- **Vagó:**
  - **Passatgers:**  $pes + places * 100$ .
  - **Correu:**  $pes + pesEst$ .
  - **Mercaderies:**  $pes + càrrega + tara$ .

L'atribut **dsm** de la classe Comboi és la data en la qual s'ha de fer l'últim manteniment periòdic del comboi al taller al qual està assignat. L'atribut **maxVSM** de la classe Màquina és el nombre màxim de viatges que pot fer una màquina sense manteniment.

Us demanen que dissenyeu tres operacions que tenen com a contracte:

**context** Mecànic::necessitaManteniment(dm: Data): Set (String)

**post** *afectats*: Retorna les referències dels combois assignats al taller mecànic *self* pels quals:

- La data del següent manteniment **dsm** del comboi és superior igual a la del paràmetre **dm**.
- O el nombre de viatges fets des de l'últim manteniment de la màquina del comboi sobrepassa el màxim de viatges que la màquina pot fer sense manteniment.

**context** Comboi::pesTEC(): Enter

**post** *pesTotalEnCarrega*: Retorna el pes total en càrrega del Comboi *self*.

**context** Taller::creaComboi(ncb: String, t: enter, maq: Màquina, vagons[0..50]: Material\_Mòbil, da: Data)

**pre:** *existeixMàquina*: *maq* correspon a una instància de Màquina que no està integrada en cap comboi.

**pre:** *existeixenVagons*: totes les instàncies de *vagons* corresponen a una instància de Material\_Mòbil de tipus Vagó i cap d'ells està integrat en un comboi.

**pre:** *nouComboi*: no existeix cap comboi de referència *ncb*.

**exc** *senseVagons*: el paràmetre *vagons* no té cap element.

**exc** *misteri*: excepció que heu de descobrir analitzant els invariants derivats del diagrama de classes.

**post** *altaComboi*: Es crea la instància de Comboi de referència *ncb*, *dsm=da* i *tara=t*.  
**post** *creaIntegratEn*: Es crea l'associació *integrat en* entre nou Comboi i el Material Mòbil integrat.  
**post** *creaEsMàquinaDe*: Es crea l'associació *es màquina de* entre el nou Comboi i la màquina *maq*.

Si us convé podeu fer ús de l'operació:

**context** Comboi::comparaData(d1: Data, d2: Data): Booleà.

**post** *major*: retorna cert si d1 és major o igual que d2.

**Es demana:**

- a) Dissenyar completament l'operació necessitaManteniment().
- b) Dissenyar completament l'operació pesTC().
- c) Feu una proposta per a la declaració de l'excepció *misteri*.
- d) Dissenyar completament l'operació creaComboi().
- e) Justifiqueu les navegabilitats i acoblaments resultats de l'enunciat i del vostre disseny i inclogueu-les dins del diagrama repetit al full de teoria.

**No cal que assigneu cap operació a les classes en el diagrama de classes.**