

# Tema 3 Model de domini

**Maria Salamó Llorente**

**Disseny de Software**

Enginyeria Informàtica

Facultat de Matemàtiques, Universitat de Barcelona

# Contingut

- 3.1 Model de domini: concepte i unitat
- 3.2 Conceptes
- 3.3 Associacions
- 3.4 Generalització
- 3.5 Atributs

## 3.1 Model de domini: Concepte i unitat

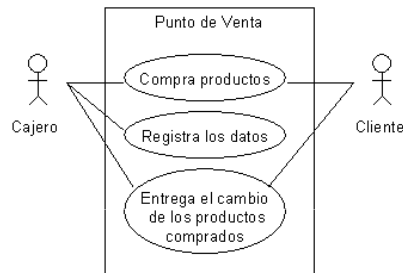


# Introducció

Requisits de l'usuari i del sistema

Models de sistema (Anàlisi)

Diagrames de casos d'ús



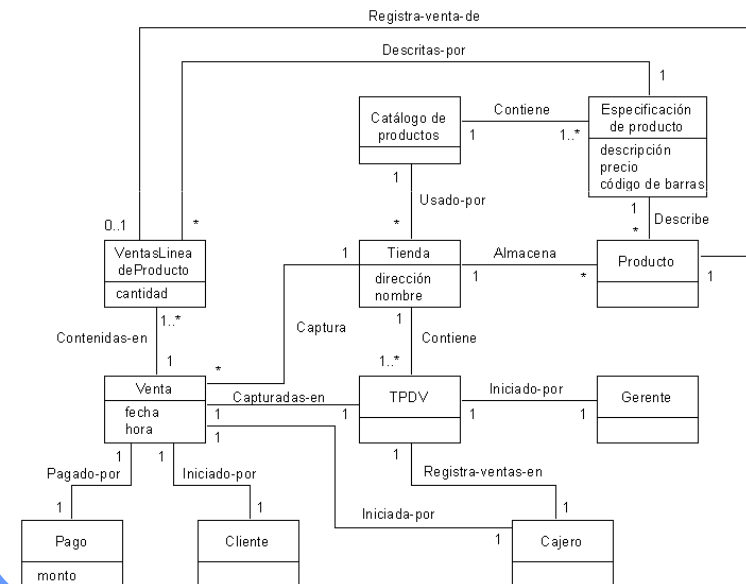
Especificació del UC

## CASO DE USO:

Actor principal
Personal involucrado e intereses
Precondiciones
Postcondiciones
Escenario principal de éxito (Flujo básico)
Extensiones
Requisitos especiales
Lista de tecnología y variaciones de datos (requisitos no funcionales)
Frecuencia
Temas abiertos

Orientació a objectes

Diagrames de classes conceptuals



# Definició model de domini

- El **model de domini** és una representació visual de classes conceptuais o objectes del món real en un domini d'interès
  - També es coneix com: Model conceptual, Model d'objectes del domini o Model d'objectes d'anàlisi
- **L'objectiu** és entendre el domini del problema i el sistema que es vol implementar
- S'utilitzen els casos d'ús de la fase de captura de requisits

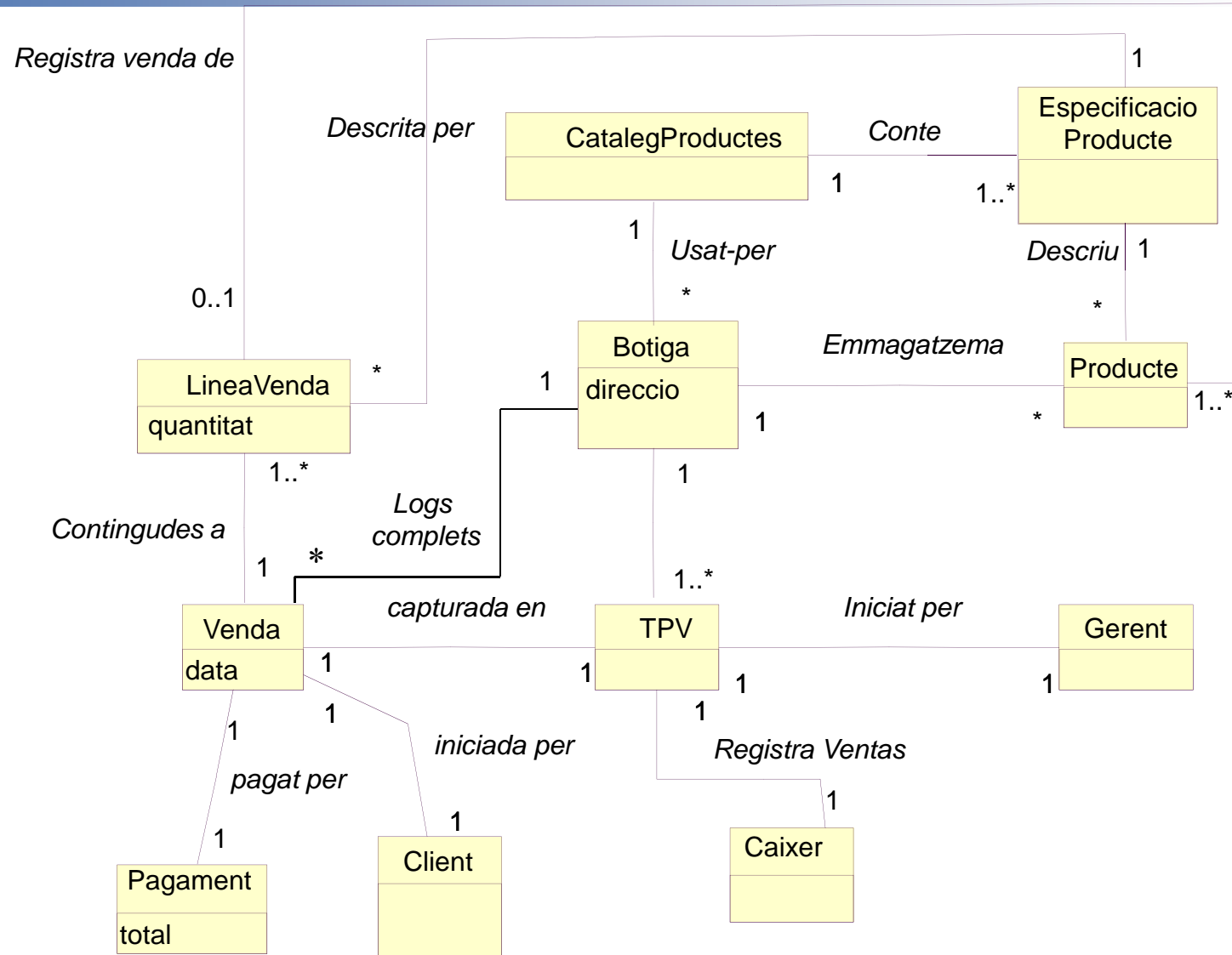
# Què conté un model de domini?

- El **model de domini** s'il·lustra mitjançant un conjunt de **diagrames de classes** en UML en el que **no apareixen operacions**.

Mostra:

- **Classes conceptuais** o objectes del domini
  - **Relacions (associacions) entre classes** conceptuais
  - Atributs de les classes conceptuais
- El model de domini constitueix un **diccionari visual de les abstraccions** rellevants

# Exemple



# Com construir un model de domini

1. **Elaborar una llista** de classes conceptuais candidates usant la tècnica de la llista de categories de classes conceptuais i la identificació de sintagmes nominals en els requisits
2. **Dibuixar-les en un model** de domini
3. **Afegir les associacions** necessàries per emmagatzemar les relacions de les que és necessari guardar memòria
4. **Afegir els atributs** necessaris per satisfer els requisits d'informació



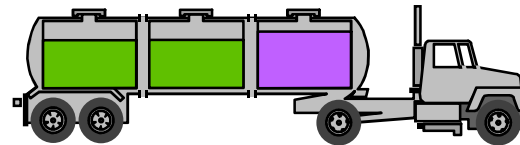
# Consideracions al construir el model de domini

- S'ha d'actuar com un cartògraf
  - Usar els noms existents en el territori
  - Excloure les característiques irrelevantes
  - No afegir res que no estigui allà
- És un error comú oblidar-se classes i descobrir-ho a posteriori
  - És millor sobreespecificar-lo amb una gran quantitat de classes conceptuais molt refinades per subespecificar-lo.
- No associar directament classes amb informació
  - Poden existir classes que només continguin comportament (sense atributs)

## 3.2 Conceptes

# Què és un concepte?

## – Entitat física



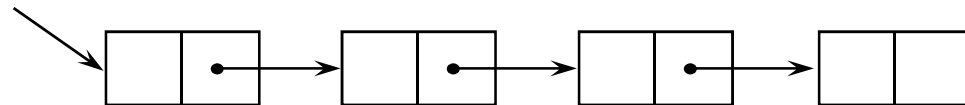
Camió

## – Entitat conceptual



Procés  
químic

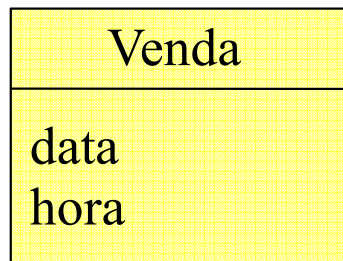
## – Entitat de software



Llista Enllaçada

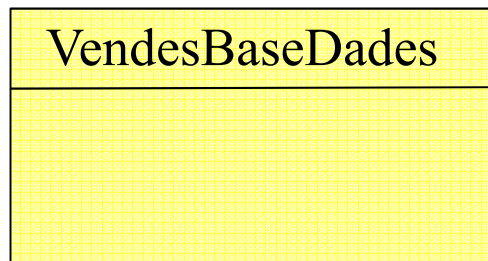
# Conceptes en el model de domini

- Un model de domini no representa components software
- No s'han de representar artefactes software, responsabilitats o mètodes



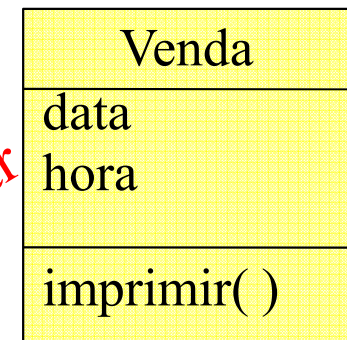
(a) **Concepte**  
de domini

*evitar*



(b) **Artefacte**  
software

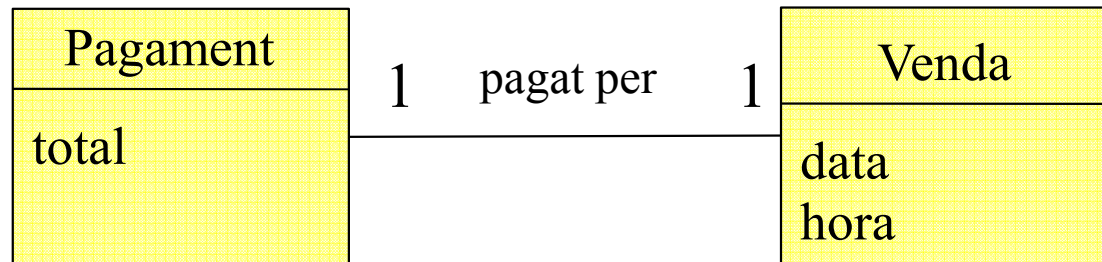
*evitar*



(c) **Classe**  
software

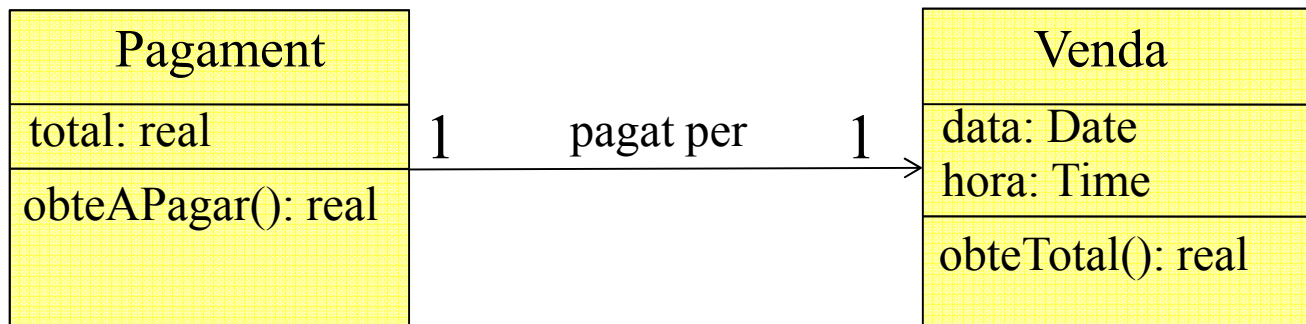
# Perspectives

## Perspectiva conceptual amb el model de domini



Model essencial  
que visualitza  
conceptes del mon  
real

## Perspectiva d'especificació amb el diagrama de classes: Model de disseny



Model específic  
que visualitza  
components  
software

# Salt en la representació

- El salt representacional és la diferència entre el nostre model mental i la seva implementació software
- El model de domini proveeix d'un diccionari visual del vocabulari i els conceptes del domini en el qual ens podem inspirar per nomenar algunes coses en el disseny software
- Des d'un punt de vista pràctic convé un salt representacional reduït

# Classes conceptuals

- Formalment, una classe conceptual es pot considerar en termes de:
  - **Símbol**: paraules o imatges que representen una classe conceptual
  - **Intensió**: la definició de la classe conceptual
  - **Extensió**: El conjunt d'exemples al que la classe conceptual aplica
- Estratègia per identificar-les
  - Usar una llista de categories de classes conceptuals
  - Identificar sintagmes nominals

# Llista de categories de classes (I)

<b>Categoria de classe conceptual</b>	<b>Exemples</b>
Objectes físics o tangibles	Capsa, avió
Especificacions, dissenys o descripcions de les coses	EspecificacioDeProducte DescripcioDeVol
Llocs	Botiga, Aeroport
Transaccions	Venda, Pagament, Reserva
Línies de transacció	LiniaDeVenda
Rols de gent	Caixer, Pilot
Contenidors d'altres coses	Magatzem, paquet, aeroplà
Coses contingudes en un contenidor	Passatger, ítem o article



## Llista de categories de classes (II)

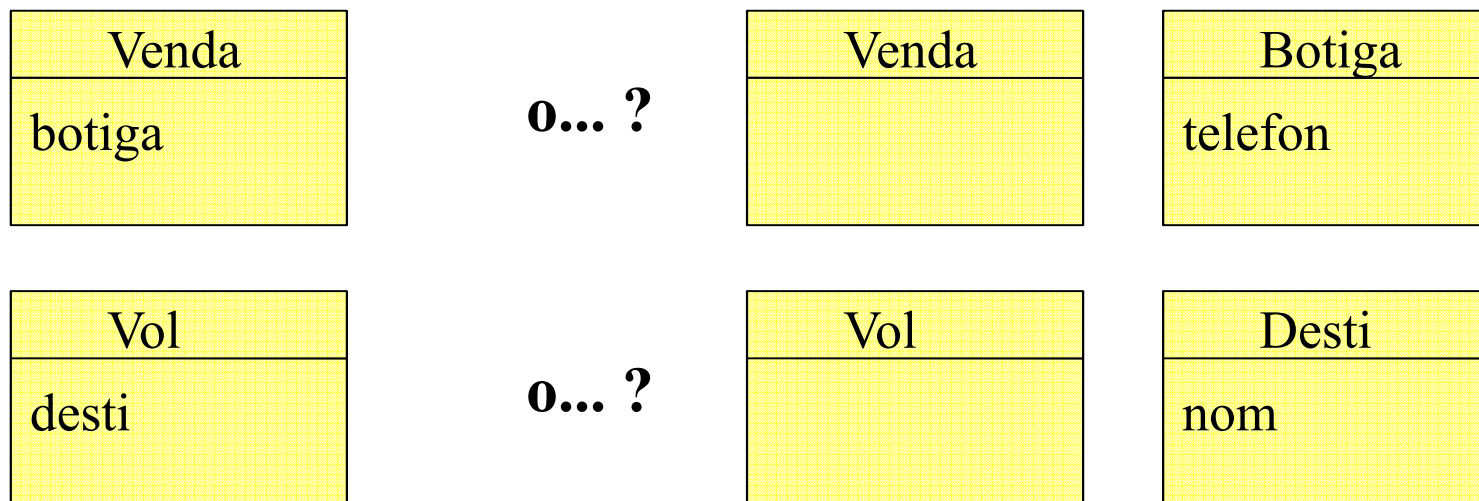
<b>Categoria de classe conceptual</b>	<b>Exemples</b>
Altres sistemes informàtics o electromecànics externs al sistema	SistemaDAutoritzacioDeCredits ControlDeTraficAeri
Noms de conceptes abstractes	Ànsia, Aerofòbia
Organitzacions	DepartamentDeVendes, CompanyiaAeria
Esdeveniments	Venda, Pagament, Reunió, Vol, Accident, Aterratge
Regles i polítiques	PoliticaDevolucions, PoliticaDeCancelacions

## Llista de categories de classes (III)

<b>Categoria de classe conceptual</b>	<b>Exemples</b>
Catàlegs	CatalegDeProductes CatalegDePeces
Registres de finances, treball, contractes, assumptes legals	Rebut, ContracteDeTreball, RegistreDeManteniment
Instruments i serveis financers	LiniaDeCredit, Stock
Manuals, documents, llibres, fulls de referència	LlistaDiariaDeCanvisDePreu, ManualDeReparacions

# Diferenciar entre concepte i atribut

- L'error més comú al construir un model de domini és representar alguna cosa com un atribut quan hauria de ser un concepte
- ① **Si per un ens X no pensem en ell com un número o un text en el món real, llavors X hauria de ser una classe conceptual i no un atribut**



# Exercici: UB Cultura

El centro cultural *UBCultura* se dedica al préstamo de dos tipos de materiales de préstamo: discos y libros. Para los dos se guarda información general, como su código identificativo, el título y el autor. En el caso de los libros, almacenamos también su número de páginas, y para los discos el nombre de la discográfica.

Al centro cultural acuden una serie de clientes (de los que se guarda su DNI y nombre), que realizan una serie de peticiones de discos o libros (como mucho hasta 5 peticiones). Para cada petición se guarda la fecha de inicio y fin del préstamo.

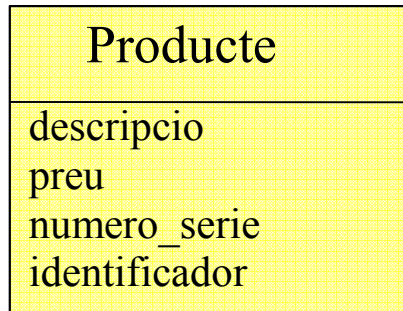
**Objetivo:** Realizar una abstracción en detalle de las clases conceptuales que estimes apropiadas para el problema planteado anteriormente.

# Solució proposada a classe

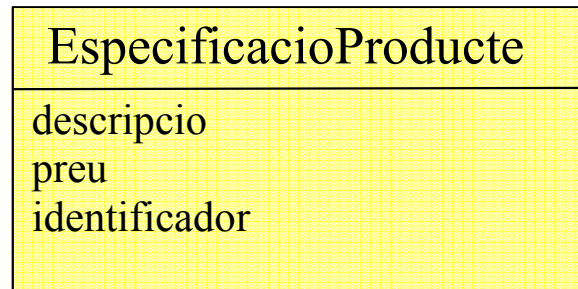
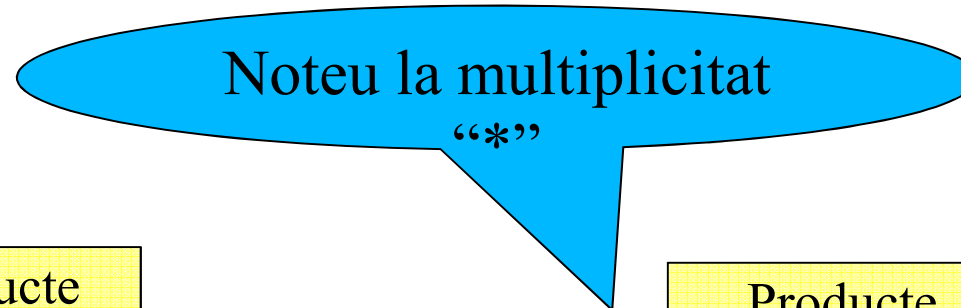
# Classes conceptuais d'especificació o descripció

- Una classe **EspecificacióDeX** descriu un X
- S'ha d'afegir una classe d'especificació quan:
  - Es necessari tenir la descripció d'un ítem o servei amb independència de l'existència d'un exemplar
  - Eliminar instàncies d'un cert tipus resulta una pèrdua d'informació que ha de ser mantinguda
  - Redueix la informació duplicada o redundant

# Classes conceptuais d'especificació



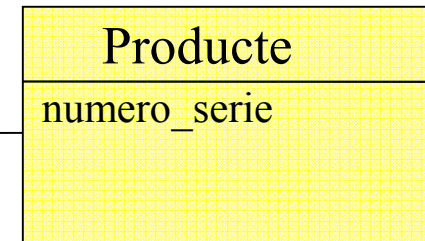
**Pitjor**



1

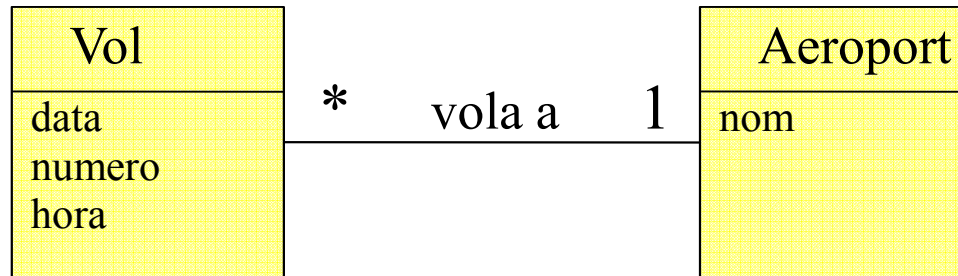
descriu

\*

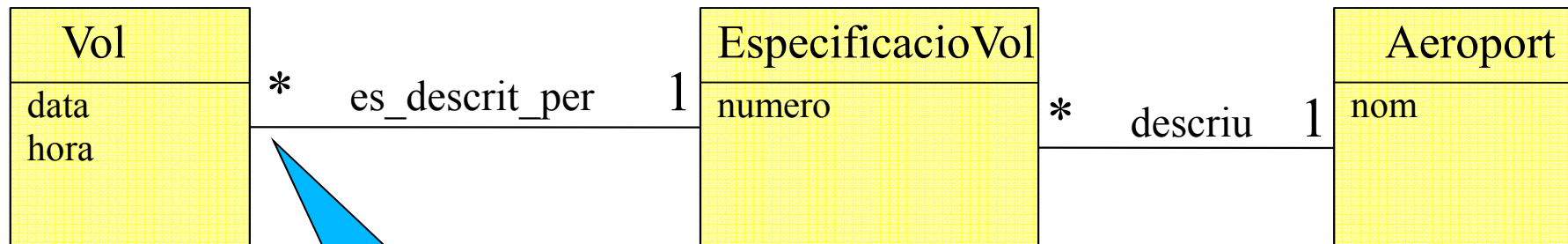


**Millor**

# Classes conceptuals d'especificació



**Pitjor**



Noteu la  
multiplicitat “\*”

**Millor**



# Classes en UML

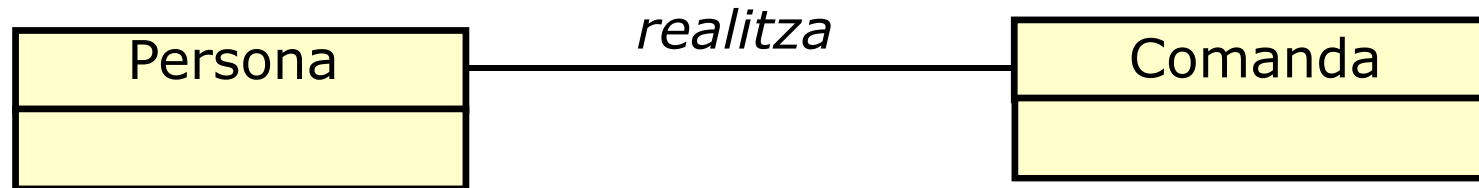
- **Classe conceptual.** Concepte o cosa del món real
- **Classe software o de disseny.** Representa la perspectiva d'especificació o implementació d'un component software
- **Classe d'implementació.** Una classe implementada en un llenguatge orientat a objecte (com Java)
- **Classe.** Terme general representat o bé un concepte del món real o del software

## 3.3 Associacions

# Definició

- Una **associació** és una **relació** entre classes conceptuais (instàncies d'aquests tipus) que indica alguna connexió interessant i significativa
- Existeixen 3 tipus de relacions entre classes:
  - **Associació**
  - **Agregació**
  - **Composició**
- Es representa per una línia entre classes amb un nom d'associació
  - Les associacions són inherentment bidireccionals (no en software o implementació)
  - Existeix una símbol per indicar la direcció de lectura
  - Els extrems contenen expressions de **multiplicitat**

# Associació



- Tota associació té un nom
- S'ha de donar nom a les associacions basant-se en el format:

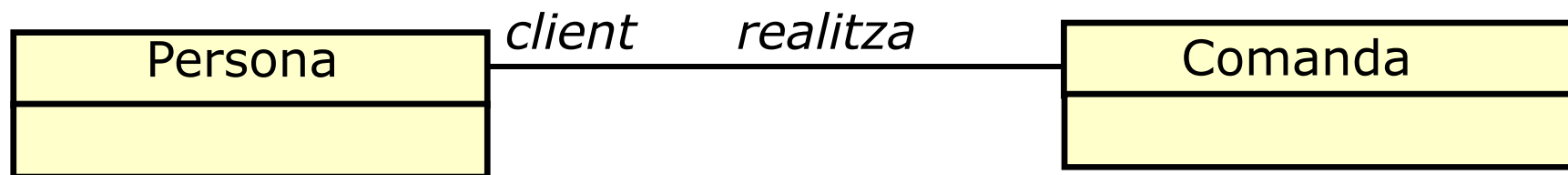
## **NomDeClasse – PerífrasisVerbal – NomDeClasse**

On la perífrasis verbal crea una seqüència que és llegible i té significat en el context del model

- Per defecte es llegeixen d'esquerra a dreta i de dalt a baix

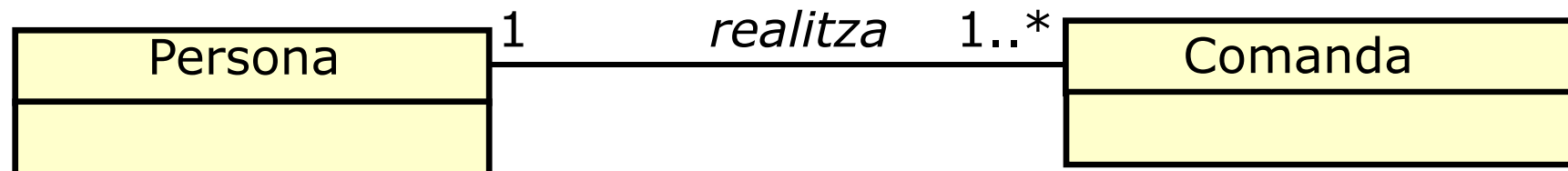
# Rol

- Cada extrem d'una associació s'anomena **rol**
- El rol s'usa per millorar la comprensió del diagrama
- Una o ambdues classes que connecta l'associació pot tenir un rol assignat



# Multiplicitat

- La multiplicitat indica el numero d'instàncies d'una classe relacionada a una instància d'una altra classe
- El valor de la multiplicitat comunica quantes instàncies es permeten associar amb una altra en un moment concret en el temps, no al llarg d'un interval de temps

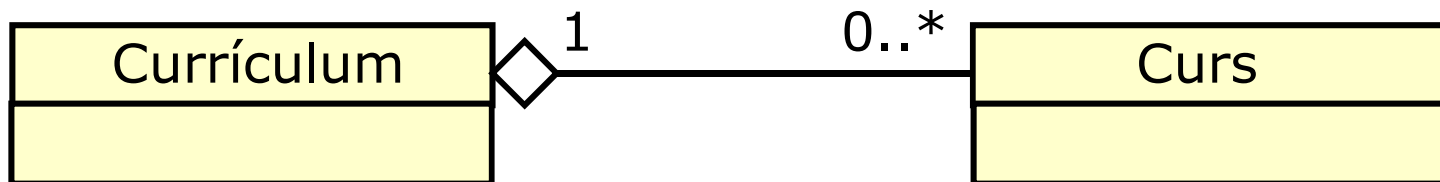


# Indicadors de multiplicitat

— *	T	Zero o més (molts)
— 1..*	T	Un o més
— 1.. 40	T	D'un a 40
— 5	T	Exactament 5
— 3, 5, 8	T	Exactament 3, 5 o 8

# Agregació

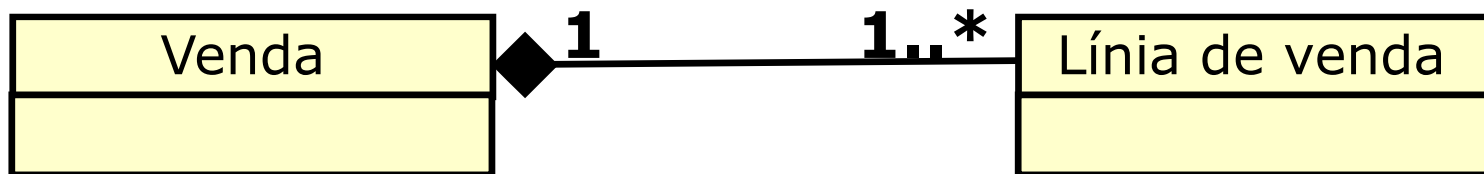
- L'agregació és una associació especialitzada en la qual un tot es relaciona amb les seves parts.





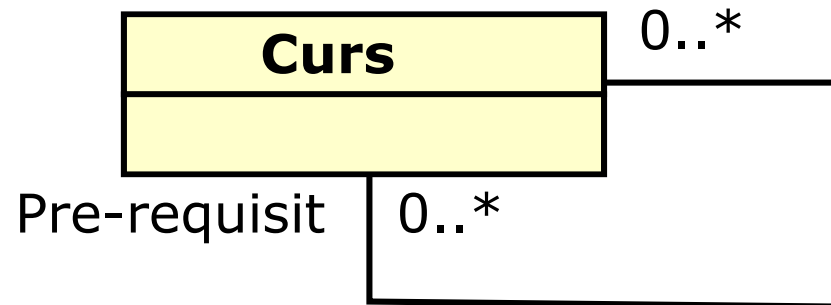
# Composició

- La composició és una associació especialitzada d'una agregació
- Les parts es creen i es destrueixen amb el tot



# Associació reflexiva

- En una associació reflexiva, els objectes d'una mateixa classe estan relacionats
  - Indica que els objectes d'una mateixa classe col·laboren entre ells per aconseguir una responsabilitat

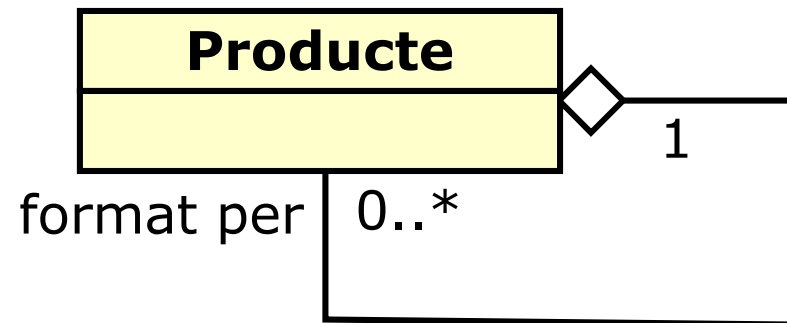


**Un curs pot tenir molts pre-requisits**

**Un curs pot ser pre-requisit per molts altres cursos**

# Agregació reflexiva

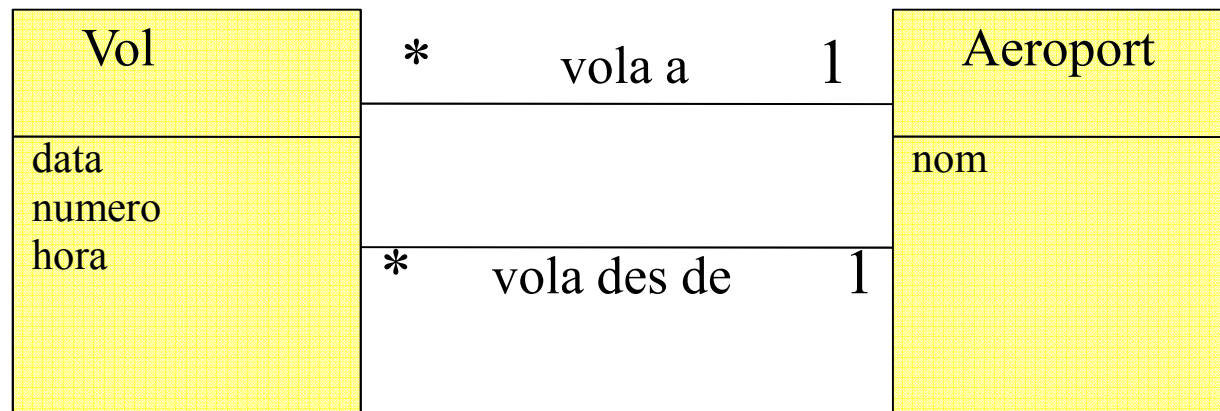
- En una agregació reflexiva els objectes d'una mateixa classe poden ser part del tot



**Un producte pot estar compostat de molts altres productes**

# Associacions múltiples

- Dos classes poden tenir múltiples associacions entre elles



## Recomanacions

- Centrar-se en aquelles associacions per les que el coneixement de la relació es necessita **preservar** per algun temps
- És més important identificar **classes conceptuais** que identificar associacions
- Masses associacions tendeixen a confondre enlloc d'aclarir
- S'ha d'evitar mostrar associacions redundants o derivables

# Llista associacions fonamentals

<b>Categoria</b>	<b>Exemples</b>
A és físicament una part de B	Ala-Avió, Mà-Cos
A és lògicament una part de B	LíniaDeFactura-Factura, Capítol-Llibre
A està físicament continguda en B	Passatger-Avió, Producte-Embalatge
A està lògicament continguda en B	DescripcióDeProducte-Cataleg, Vol-HorariDeVols
A és conegut / registrat/ reportat/ capturat en B	Factura-LlibreDeFactures

# Llista associacions comuns (I)

<b>Categoria</b>	<b>Exemples</b>
A és una descripció de B	Producte-DescripcioDeProducte, Vol-DescripcioDeVol
A és un element de línia d'una transacció de B	Factura-LiniaDeFactura, Comanda-LiniaDeComanda
A és un membre de B	Pilot-LiniaAeria, Professor-Departament
A és una subunitat organitzacional de B	Departament-Universitat, Delegació-Empresa
A usa o gestiona B	Pilot-Avió, Professor-Aula
A es comunica amb B	Client-Caixaer, Alumne-Professor

# Llista associacions comuns (II)

## Llista d'associacions comuns (III)

<b>Categoria</b>	<b>Exemples</b>
A està relacionat amb una transacció B	Client-Pagament, Passatger-Tiquet
A és una transacció relacionada amb un altra transacció B	Pagament-Venda, Reserva-Cancelació
A és el següent de / està junt a B	LiniaDeFactura-LiniaDeFactura, Ciutat-Ciutat
A és l'amo de B	LiniaAeria-Avió
A és un esdeveniment relacionat amb B	Venda-Client, Sortida-Vol



# Associacions i implementació

- Durant el modelat de domini:
  - una associació no es refereix a flux de dades, variables d'instàncies o connexions d'objectes en una solució software
  - una associació es refereix a una relació en sentit conceptual.
- Algunes associacions del model de domini no seran necessàries per la implementació
- El model de domini només ha de mantenir-se al dia (com qualsevol altre artefacte) si això té sentit en el projecte

# Exercici: UB Cultura

El centro cultural *UBCultura* se dedica al préstamo de dos tipos de materiales de préstamo: discos y libros. Para los dos se guarda información general, como su código identificativo, el título y el autor. En el caso de los libros, almacenamos también su número de páginas, y para los discos el nombre de la discográfica.

Al centro cultural acuden una serie de clientes (de los que se guarda su DNI y nombre), que realizan una serie de peticiones de discos o libros (como mucho hasta 5 peticiones). Para cada petición se guarda la fecha de inicio y fin del préstamo.

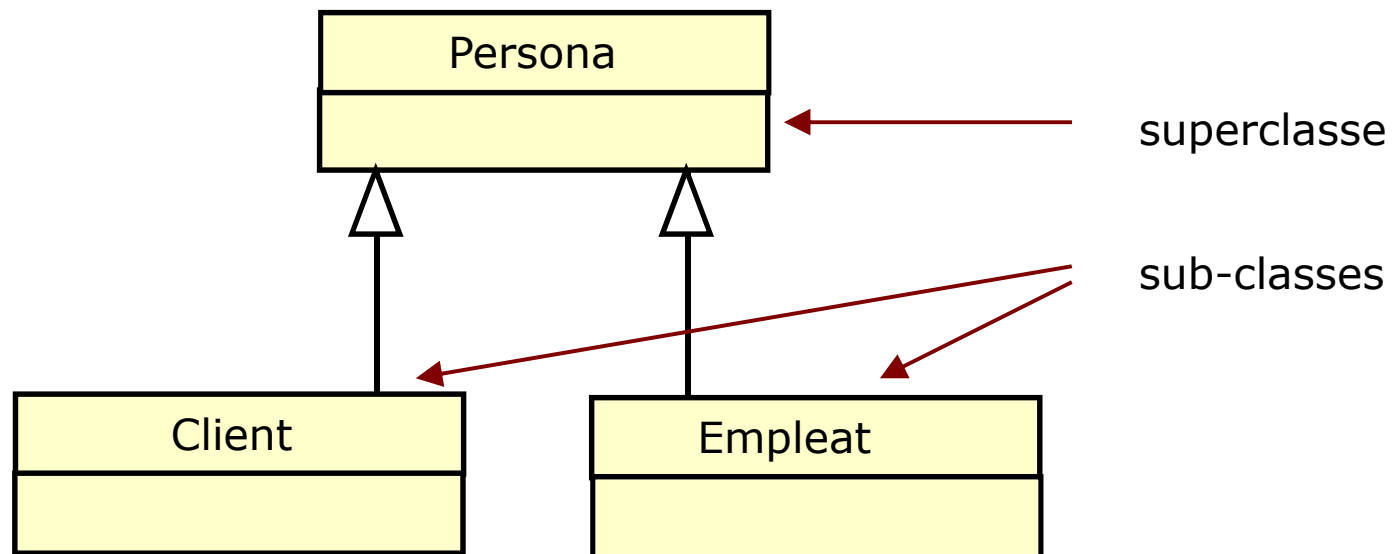
**Objetivo:** Definir las asociaciones entre clases conceptuales que estimes apropiadas para el problema planteado anteriormente.

# Solució proposada a classe

## 3.4 Generalització

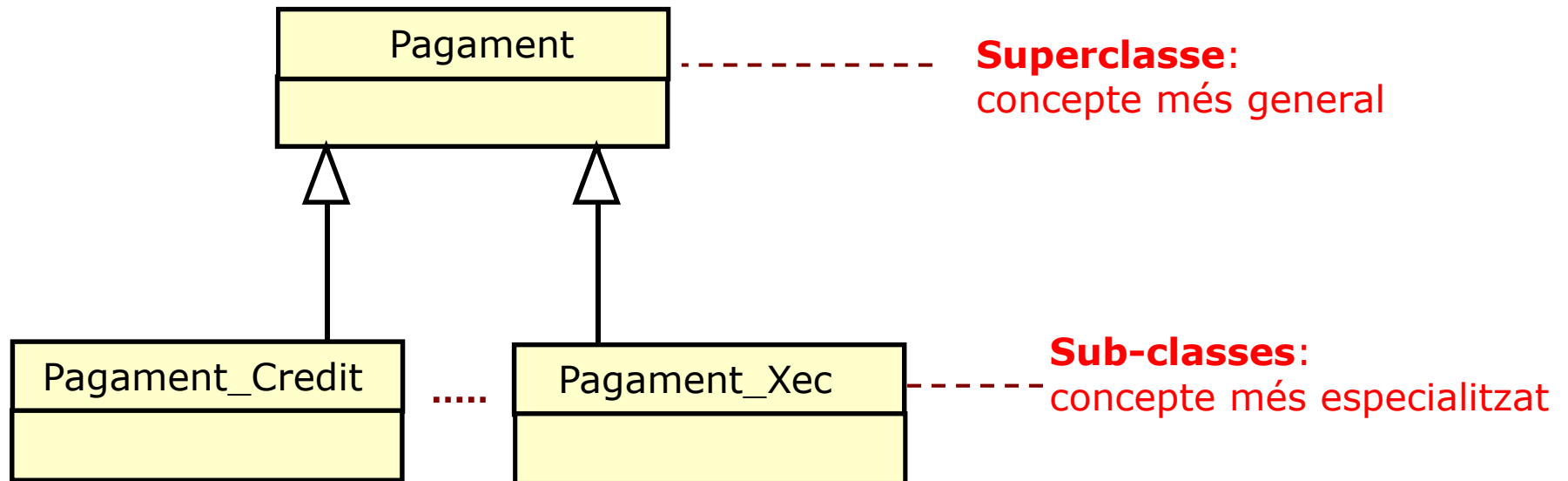
# Definició

- L'herència es representa a través d'una **associació de generalització**
- **Generalització**: És l'activitat d'identificar factors comuns entre conceptes
  - Es defineixen relacions entre superclasse i sub-classe



# Representació generalització

- L'associació de generalització representa una jerarquia de classes.
  - Jerarquia de generalització /especialització



# Regles

## Regles necessàries per la definició d'una subclasse

- **Regla del 100%:** El 100% de la definició d'una classe conceptual ha de ser aplicable a les seves sub-classes.
  - És a dir, per la subclasse tenen sentit el 100% dels **atributs** i les **associacions** de la superclasse
- **Regla És-un:** Tots els membres d'una subclasse han de ser membres de la superclasse.
  - *Sub-classe és una superclasse*

# Quan crear una subclasse

1. Si la subclasse té atributs addicionals interessants
2. Si la subclasse té associacions addicionals interessants
3. Si s'ha d'operar, manegar, reaccionar o manipular de forma diferent i interessant el concepte de la nova subclasse del de la superclasse o altres subclasses
4. Si el concepte de la subclasse representa un ens animat que respon de forma diferent i interessant que la superclasse o altres subclasses



# Quan crear una superclasse

1. Les subclasses conceptuais potencials representen subconceptes d'un concepte similar
2. Les subclasses compleixen amb la regla del 100% i la regla es-un/a
3. Totes les subclasses comparteixen un atribut del que es pot treure "factor comú" i incloure'l en la superclasse
4. Totes les subclasses comparteixen una associació de la que es pot treure "factor comú" i incloure-la en la superclasse

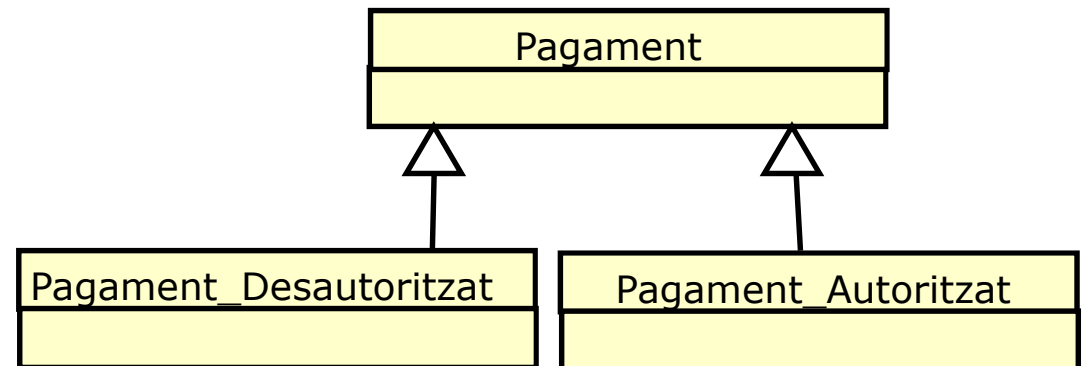
# Classes conceptuais abstractes

- Si tots els membres d'una classe C han de ser també membres d'una subclasse, llavors la classe C és una classe conceptual abstracta
- Exemple:
  - Superclasse abstracta:  
*Triangle* ← Escalens, Equilàter, Isòsceles
  - Superclasse no abstracta:  
Temàtica ← Misteri, Policíac
- En UML s'indica el nom d'una classe conceptual abstracta en cursiva

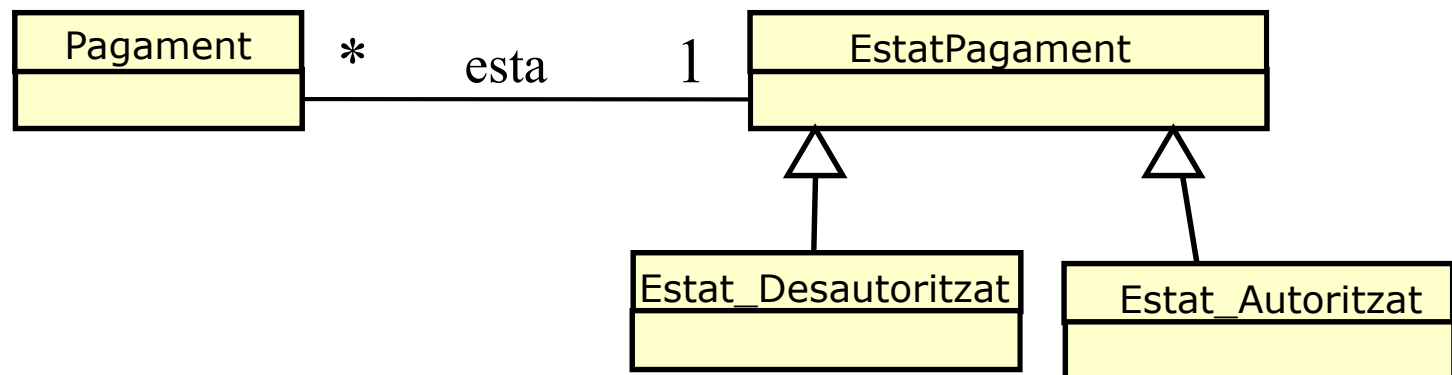
# Modelant estats

- No s'han de modelar els estats d'un concepte X com subclasses de X. Enlloc d'això:
  - S'ha de definir una jerarquia d'estats i associar el concepte amb el seu estat, o bé
  - No s'han de mostrar els estats en un model de domini

**No és útil perquè les subclasses canvien l'estat intern de la superclasse**



**Millor**



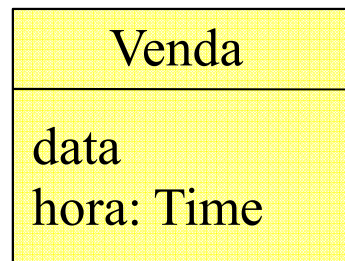
## 3.5 Atributs

# Definició

- Un **atribut** és un valor de dades lògic d'un objecte
- En un model de domini s'han d'incloure:
  - Els atributs per als que els requisits suggereixen o impliquen una necessitat de recordar informació

# Atributs en UML

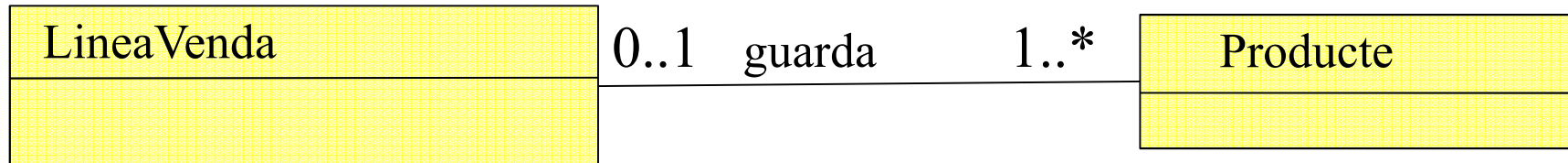
- Els atributs d'un model de domini han d'ésser preferiblement simples
- Opcionalment es pot mostrar el tipus
- Els **tipus de dades** més **comuns** són:
  - booleà, Date, numero (int o real), cadena de caràcters (string), Time
- Altres tipus de dades inclouen:
  - Adreça, color, geometries (punt, rectangle), numero de telèfon, Numero de seguretat social, Codi de producte universal, codi postal, tipus enumerats



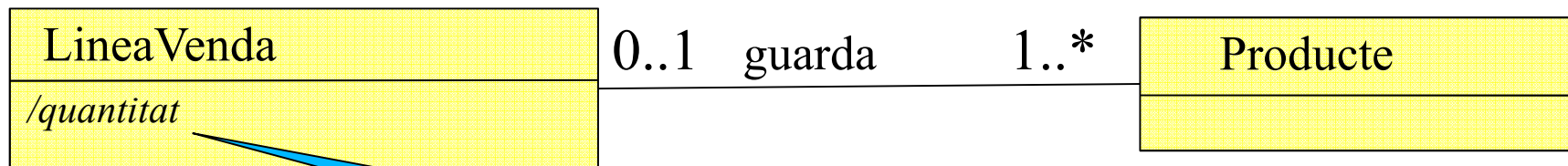
Atributs de la classe  
venda

# Atributs derivats

*Un **atribut derivat** és aquell que pot ser inferit a partir d'una altra informació del model*



-----

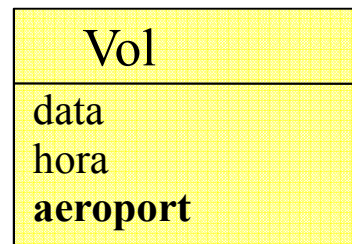


Atribut derivat a partir de la multiplicitat

# Atributs, associacions i classes

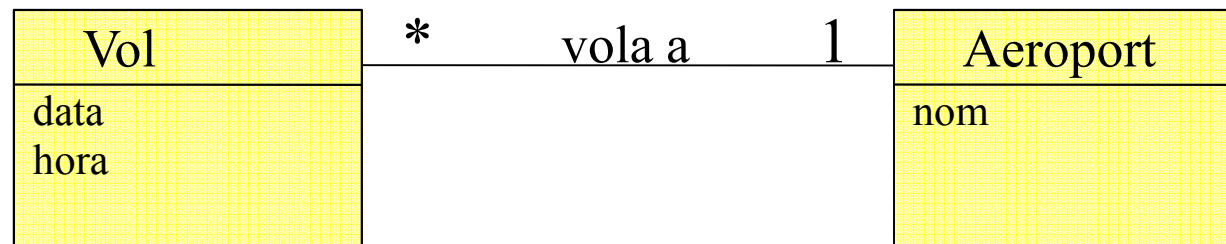
- Les classes conceptuals han de relacionar-se mitjançant associacions, no mitjançant atributs
- Si el valor de l'atribut és un concepte complex del domini s'ha de representar mitjançant una classe conceptual i no com un atribut

**Pitjor**



aeroport no és un atribut “simple”

**Millor**





# Tipus de dades UML

- Els atributs han de ser tipus de dades UML: conjunts de valors per als quals la identitat única no té sentit
- Exemple:  

```
String s = new String("hola");  
String s2 = new String ("hola");
```

s és conceptualment equivalent a s2. String és un tipus de dades UML
- Exemple contrari:  

```
Persona p1 = new Persona("Peter Jackson");  
Persona p2 = new Persona("Peter Jackson");
```

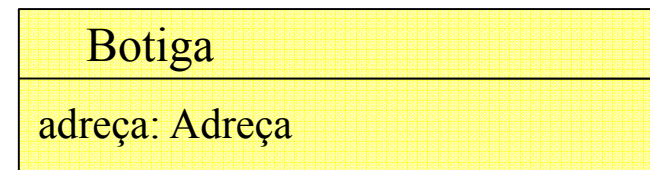
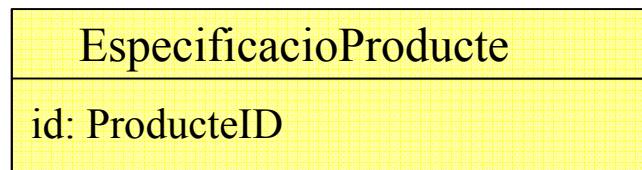
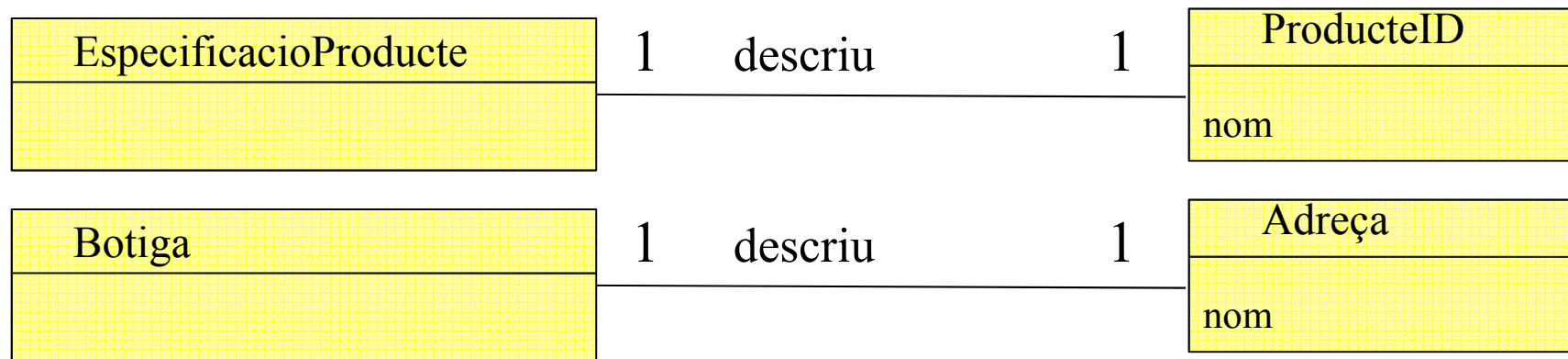
p1 no és conceptualment equivalent a p2.  
Persona és una classe

# Tipus de dades no primitius

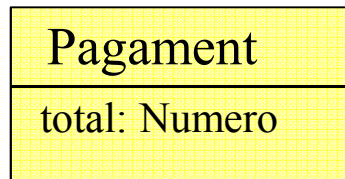
- Un tipus de dada que pot ser considerat com primitiu. S'ha de representar com una classe no primitiva si:
  - **Es compona de diferents seccions:** Telèfon, NomDePersona
  - **Té operacions associades amb ell** (p.e. Validació): Numero de seguretatSocial, NumerodeTargeta
  - **Té altres atributs:** PreuPromocional té data d'inici i data final
  - **És una quantitat** com una unitat: Diners

# Representació en UML de classes de tipus de dades

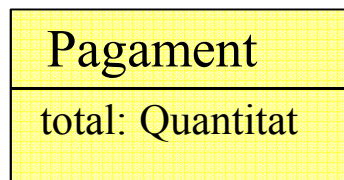
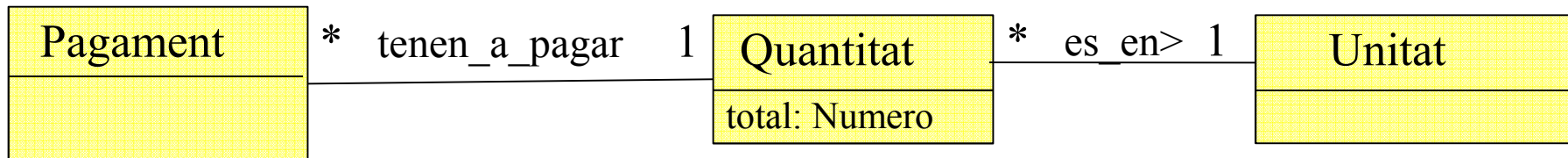
- Seleccionar l'opció que faciliti la comunicació



# Modelant quantitats i unitats



**No és útil**



**Millor qualsevol d'aquestes dues opcions**

# Exercici: UB Cultura

El centro cultural *UBCultura* se dedica al préstamo de dos tipos de materiales de préstamo: discos y libros. Para los dos se guarda información general, como su código identificativo, el título y el autor. En el caso de los libros, almacenamos también su número de páginas, y para los discos el nombre de la discográfica.

Al centro cultural acuden una serie de clientes (de los que se guarda su DNI y nombre), que realizan una serie de peticiones de discos o libros (como mucho hasta 5 peticiones). Para cada petición se guarda la fecha de inicio y fin del préstamo.

**Objetivo:** Completar las clases conceptuales con los atributos que estimes apropiados para el problema planteado anteriormente.

# Solució completa feta a classe

# Exercicis per començar a practicar

Teniu un dossier d'exercicis  
extres al campus virtual

# Exercici1: Profesores

Un alumno asiste cursos. Los cursos están impartidos por un único profesor. El alumno no puede repetir el mismo curso, pero puede asistir a más de un curso. El profesor puede impartir diferentes cursos y repetir un mismo curso en varias ocasiones. Para que un curso se imparta debe haber un mínimo de 10 alumnos y un máximo de 50. Como registro del curso se guarda la fecha de comienzo, la fecha de finalización y la nota del alumno.



# Solució Exercici 1

## Exercici 2: Matriculas

Una persona viene caracterizada por su dni, nombre, dirección y estado civil, y ésta puede convertirse en estudiante al darse de alta como tal en la universidad. Como estudiante podrá matricularse de las asignaturas que se imparten en la universidad, que tendrán un código, un nombre, un profesor responsable y un curso asignado. Una vez matriculado, el estudiante podrá recibir una beca, y en su nueva condición de becario tendrá asignado un nuevo código y se conocerá el importe de la misma; al finalizar el curso, la condición de becario acabará. Una vez el estudiante se matricula, tanto si recibe beca como si no, deberá examinarse de las asignaturas en las que se encuentra matriculado hasta que finalice el curso y vuelva a matricularse de nuevo, o bien deje la universidad y con ello deje de ser estudiante. Además, convendrá tener una serie de aplicaciones tales como dar de alta a nuevas personas y asignaturas, llevar a cabo la matriculación de estudiantes en asignaturas, registrar las notas obtenidas por los estudiantes al examinarse de cualquier asignatura en la que están matriculados y una serie de listados tales como los alumnos matriculados en una asignatura, las asignaturas en las que se ha matriculado un alumno y el listado de notas por asignatura (actas).

# Solució Exercici 2

## Exercici 3: El Centre Excursionista

El *Centre Excursionista de Folgeroles del Vallès* nos solicita informatizar su aplicación de información de excursiones organizadas.

El centro gestiona un conjunto de excursiones para sus socios. Cada excursión tiene un lugar de origen y un lugar de destino. Para cada uno de ellos se desea almacenar la posición  $x$  en kilómetros, la posición  $y$  en kilómetros y la altitud. Cada lugar pertenece a una cierta zona geográfica y en cada zona geográfica existen una serie de especies (animales y vegetales) autóctonas de la zona.

En cada excursión existe la posibilidad de realizar una serie de actividades (visitas a museos, bicicleta de montaña, natación, etc). Especialmente relevantes son las actividades de riesgo (barranquismo, descenso en kayak, etc). Todas las actividades de riesgo deberán estar controladas por un monitor.

La aplicación a diseñar deberá asumir que la introducción de los datos de las excursiones ya se ha realizado y centrarse en gestionar el acceso a la información disponible (diferentes consultas que permitan encontrar la excursión ideal para un cierto socio), así como añadir y eliminar socios de cada excursión. En cuanto a las consultas, inicialmente solo será necesario una consulta para obtener las excursiones en las que en la zona de su lugar de destino se pueda observar una cierta especie.

# Solució Exercici 3