

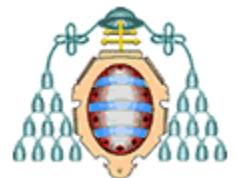
Técnica: Diagramas de Clases

Ingeniería del Software

José García Fanjul (<http://www.di.uniovi.es/~fanjul/>)
Grupo de Investigación en Ingeniería del Software
<http://giis.uniovi.es/>



Curso 2022-2023





Contenidos

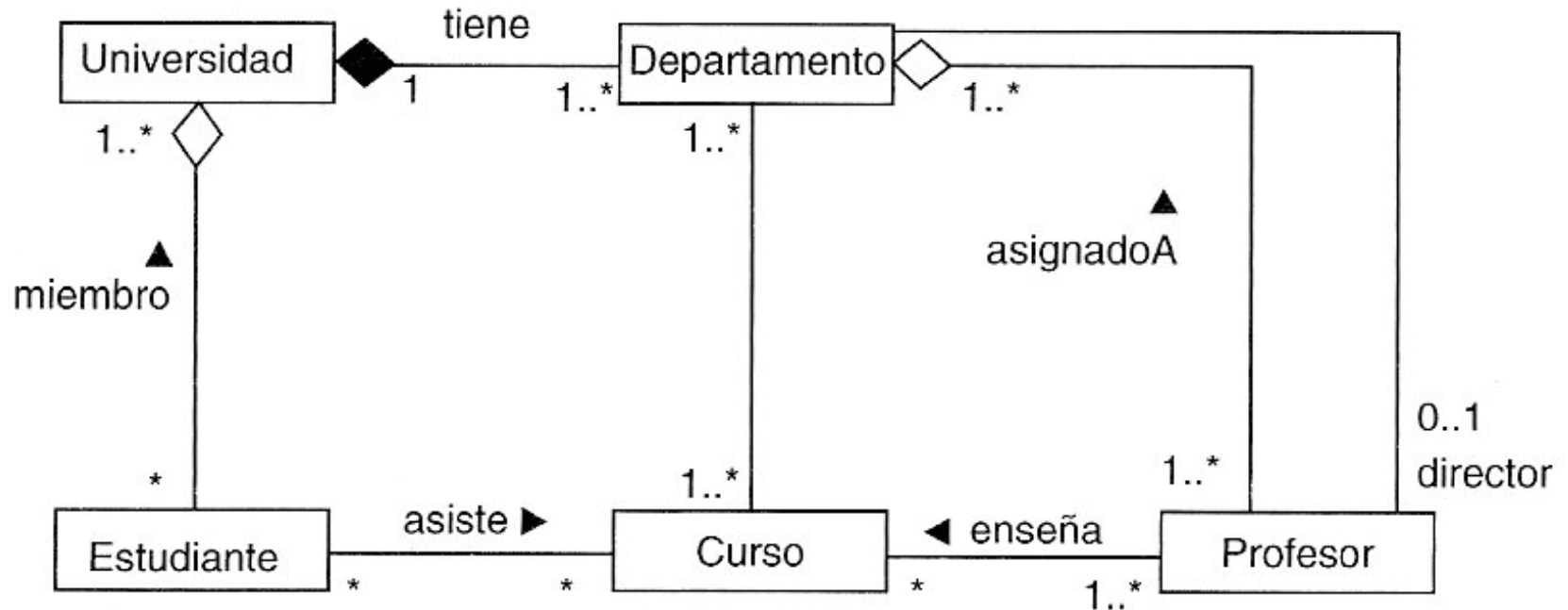
- Introducción.
- Elementos de un diagrama de clases.
- Conclusiones.

Introducción

- Técnica gráfica para modelar la **estructura estática** de un sistema.
- Especifican **qué clases hay y cómo están relacionadas**, pero no cómo interactúan para alcanzar comportamientos particulares.
- Se usan durante las siguientes fases del desarrollo:
 - Análisis.
 - Diseño.
 - Construcción.
- Importante:
 - En el análisis deben utilizarse desde un punto de vista **conceptual o esencial**, obteniendo las clases del dominio del problema (objetos de negocio).



Introducción: Ejemplo.



Extraído de Booch, Grady; Jacobson, Ivar; Rumbaugh, James (1999): *El lenguaje Unificado de Modelado*, Addison-Wesley.

¿Puede haber cursos sin profesor?



Contenidos

- Introducción.
- **Elementos de un diagrama de clases.**
- Conclusiones.

Aclaración: diagramas de clases para análisis de software



- En el manual de usuario de UML se especifican multitud de elementos y adornos para los diagramas de clases.
- El objetivo de estas transparencias es proporcionar ejemplos para usar **diagramas de clases en los procesos de análisis** por lo que sólo se explicarán aquellos elementos y adornos útiles en ese contexto.
- En la asignatura los diagramas de clases son la técnica que permite analizar el dominio de información que debe tratar un sistema de información.
- **Un diagrama de clases en este contexto expresa requisitos** en forma de las clases que existen en el dominio del problema, sus atributos y las asociaciones entre ellas.
- **El modelo de clases o modelo de dominio será el producto que se utilice para diseñar posteriormente la base de datos del sistema.**

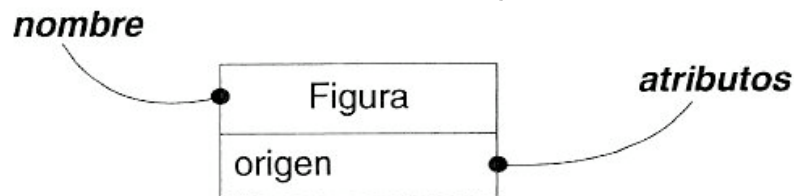
Elementos de un diagrama de clases

- **Clases.** Describen un conjunto de objetos con propiedades y comportamientos comunes.
- **Relaciones.** Enlaces entre los distintos elementos de los diagramas.

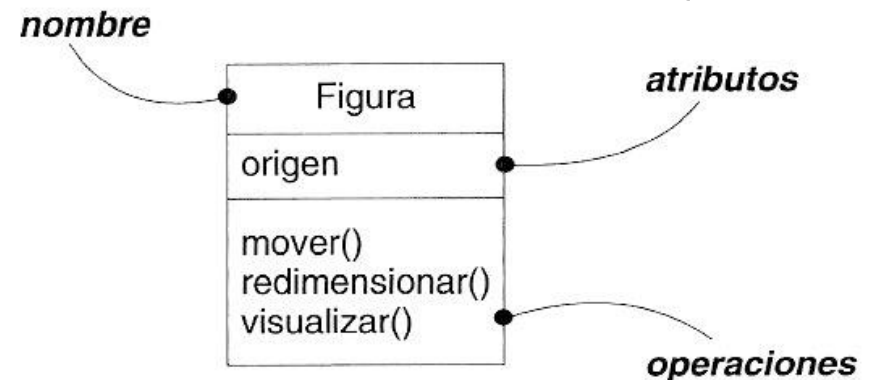
Clases

- Es una **descripción de un conjunto de objetos** que comparten los mismos atributos, relaciones y semántica.
- En los procesos de análisis no identificaremos operaciones de las clases.
- Notación en UML:

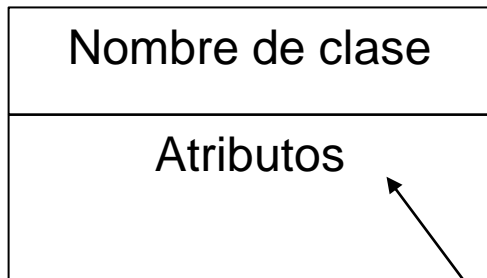
En la asignatura (análisis de software)



Con operaciones (diseño y construcción de software)



Notación UML de las clases



En los diagramas de clases de análisis aparece una lista de atributos.

En la especificación debe haber un diccionario de datos (ver transparencias sobre ASI) con una descripción detallada de atributos.
P.ej. Nombre + Tipo + Descripción + Valor defecto (si hay) + Unidades (si hay) + Ejemplos

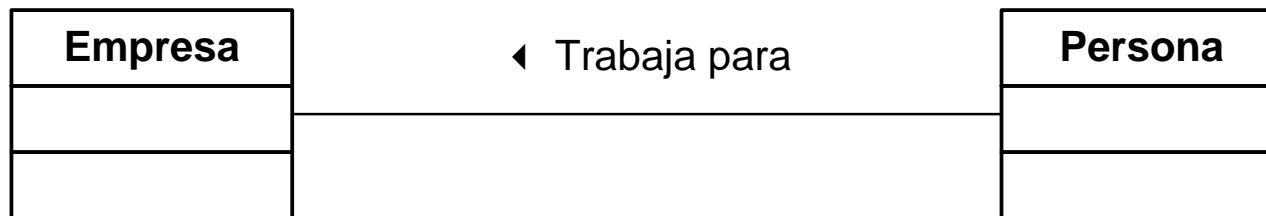
¿Qué es un crotal?

Relaciones

- Las relaciones indican **qué clases colaboran con otras.**
- Dos tipos:
 - Asociaciones: relaciones estructurales.
 - Generalizaciones: herencias.

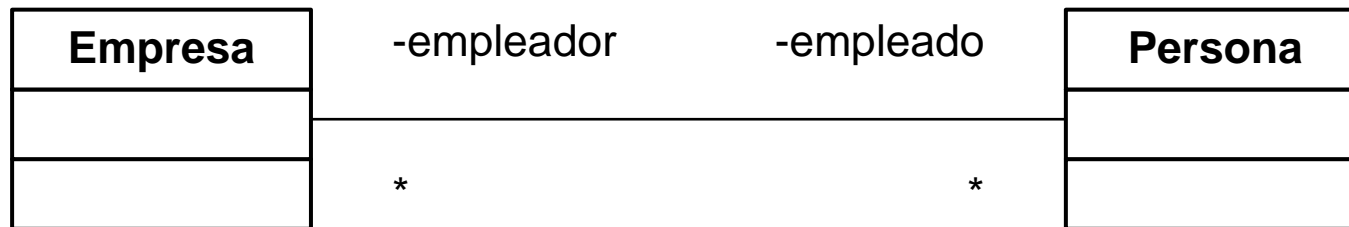
Asociaciones: nombres

- Las asociaciones pueden tener un **nombre** que las describa.



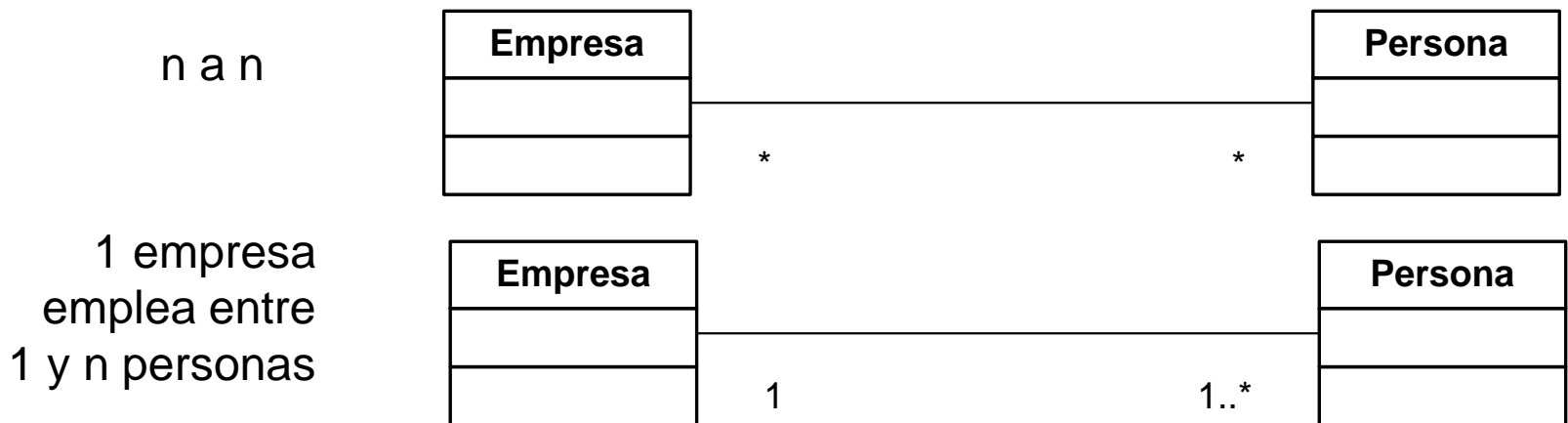
Asociaciones: roles

- También se puede especificar el **rol** que la clase juega en la relación.



Asociaciones: multiplicidad

- La multiplicidad indica el **número de objetos** involucrados en una asociación.



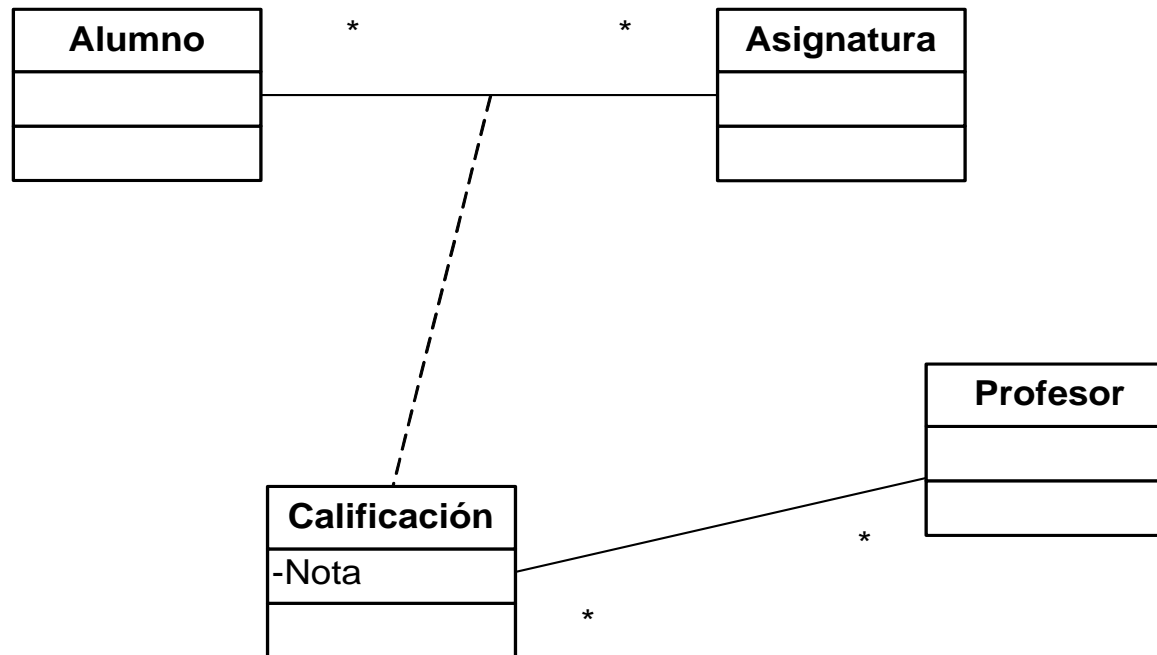
Asociaciones: multiplicidad (y 2)

■ Tipos de multiplicidad habituales:

- * = Cero, uno ó n.
- 0,1 = Cero o uno.
- 1..* = Uno o más.
- 1 = Exactamente uno (también podría ser otro número).
- 1..5 = Entre uno y cinco.

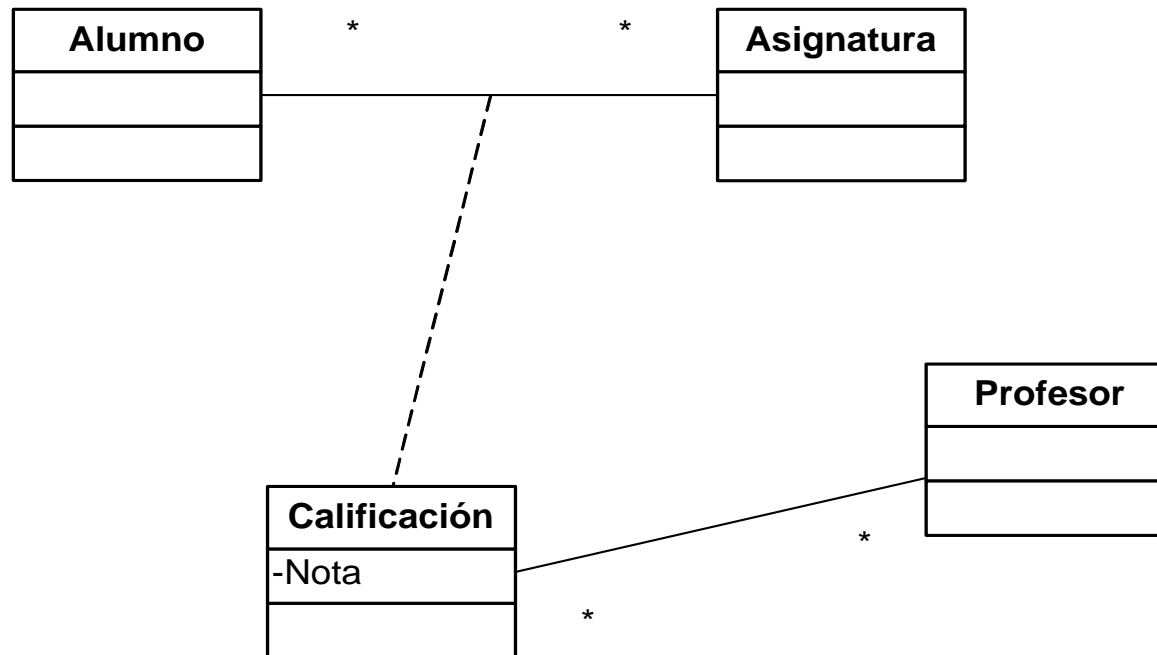
Clases de asociación

- Las clases de asociación permiten añadir atributos a las relaciones estructurales.
- Por ejemplo:



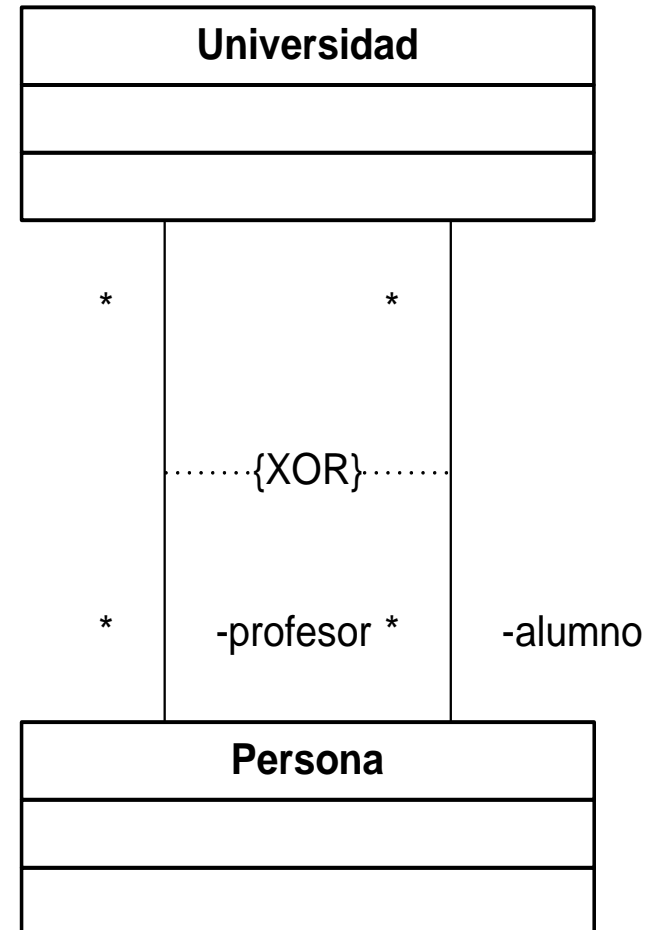
Clases de asociación (II)

- Según éste diagrama, sólo habría UNA calificación para cada alumno en una asignatura.
- ¿Cómo se puede modelar que hay más de una?



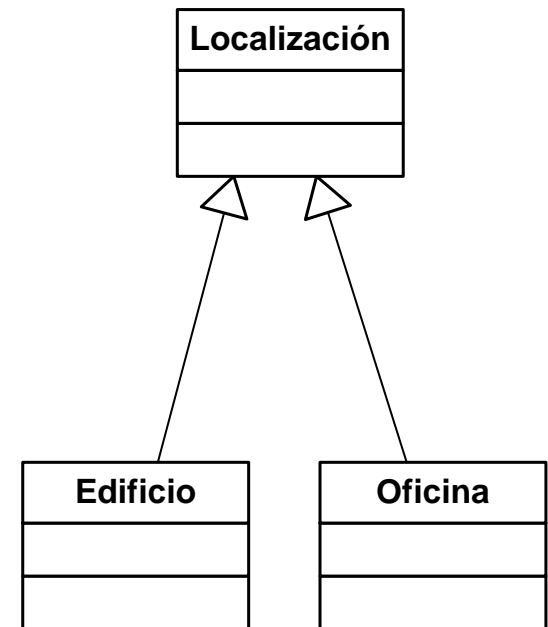
Restricciones

- Permiten establecer **limitaciones o condiciones** entre los elementos del diagrama.
- Por ejemplo, para modelar que una persona no puede ser, simultáneamente, profesor y estudiante en una Universidad:



Relaciones de generalización

- Indica una relación de “**tipo/subtipo**”.
- Permite a una clase (hijo o subclase) heredar los atributos de otra (padre o superclase).
- Se debe poder leer como “es un”. En el ejemplo de la derecha, en el contexto del sistema modelado, “un edificio es una localización”.





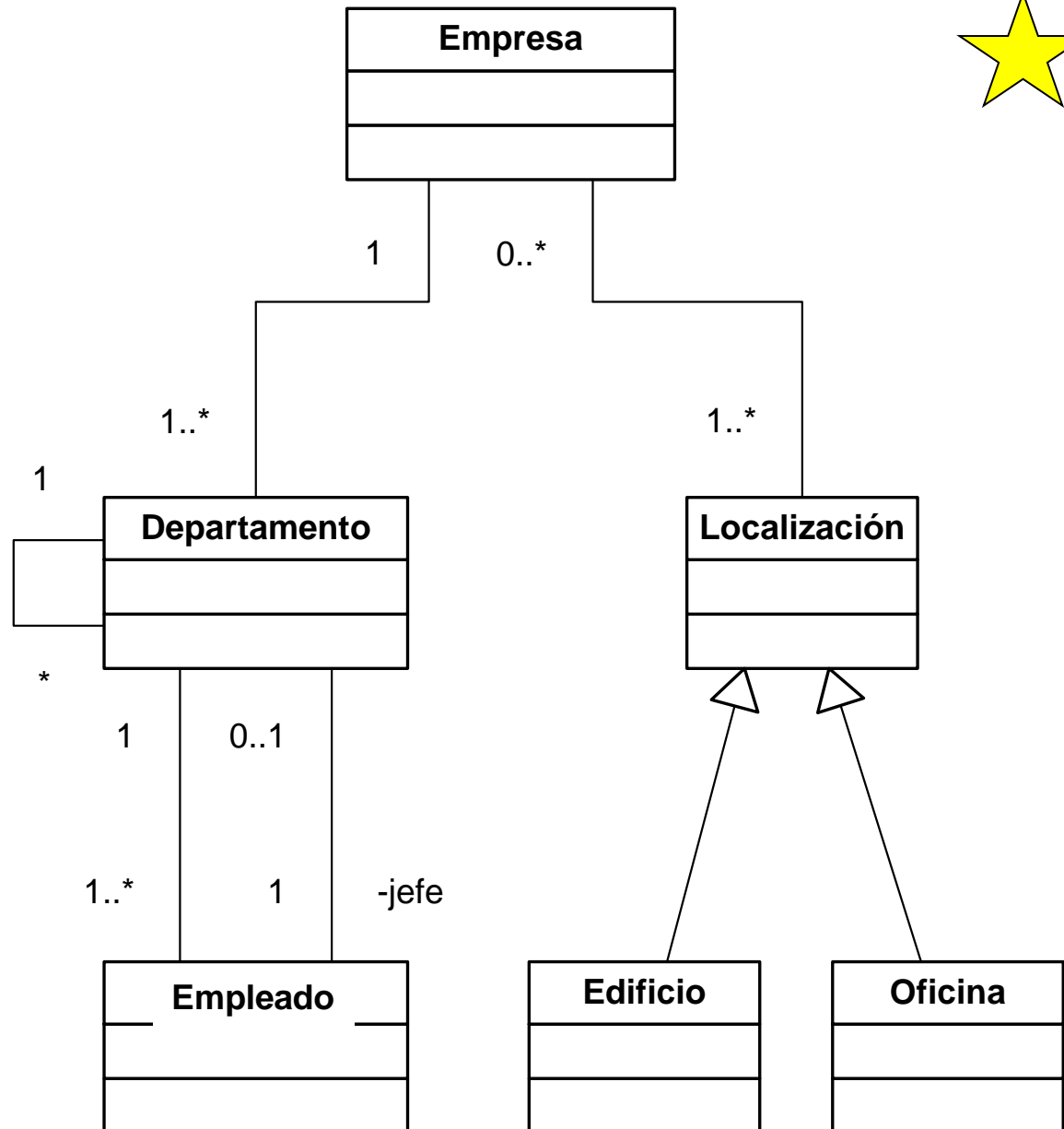
Según este modelo ¿qué requisitos de los siguientes se deben cumplir en el software?

¿debe el software permitir registrar en qué edificio trabaja un empleado?

¿puede haber departamentos sin jefe?

¿puede un empleado ser jefe de dos departamentos? ¿puede serlo a la vez? ¿y consecutivamente?

¿qué significa la relación recursiva del departamento?



**Posible diseño de BD**

Fase posterior, detalle de departamentos y personas con claves propias + ajenas para todas las asociaciones.

OJO: en diagramas de clases no incluimos claves si no forman parte del dominio (en general en el análisis)

IdDpto	NomDpto	DptoSuperior	Jefe
001	Comercial		10100001
101	Comercial LATAM	001	00100028

IdEmp	NomEmp	Dpto
10100001	Julia Álvarez	001
00100028	Antonio Díaz	101
00100101	Erika Subiabre	101



Contenidos

- Introducción.
- Elementos de un diagrama de clases.
- **Conclusiones.**

Conclusiones



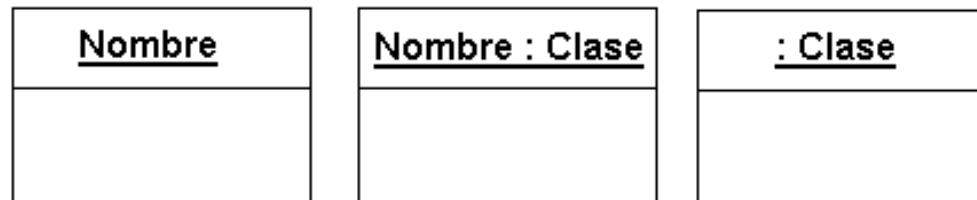
- Los diagramas de clases modelan la **estructura estática** de un sistema.
- Especifican **qué clases hay y cómo están relacionadas**, pero no cómo interactúan para alcanzar comportamientos particulares.
- En el análisis deben utilizarse desde un punto de vista **conceptual o esencial**, obteniendo las clases del dominio del problema (objetos de negocio).
- En UML se definen multitud de elementos y adornos para los diagramas de clases.
- En estas transparencias se han presentado **los más relevantes para los procesos de análisis**.

ANEXOS

- Diagramas de objetos.
- Paquetes.

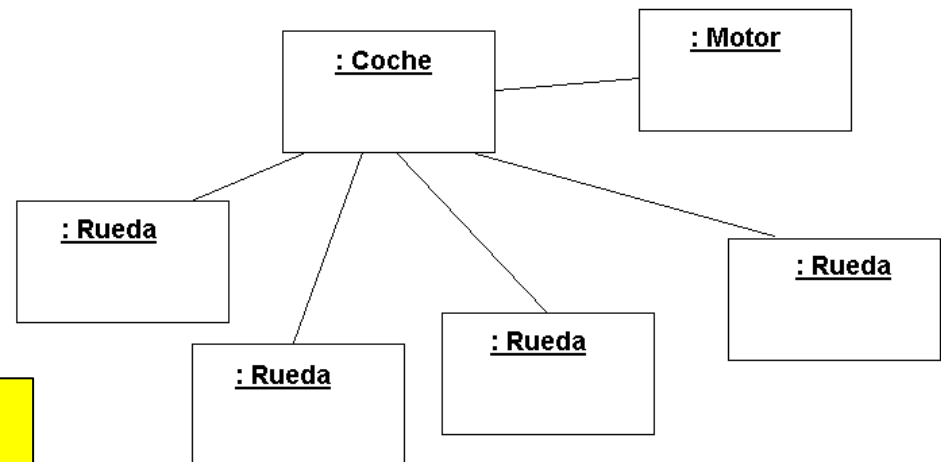
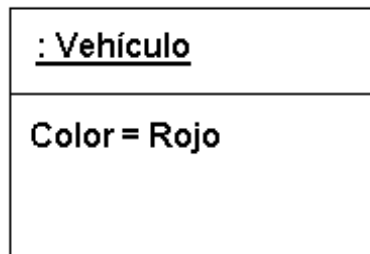
Diagramas de objetos: Definición

- Un diagrama de objetos muestra **un caso particular** de un diagrama de clases.
- Los elementos que aparecerán serán **objetos concretos** obtenidos a partir de la instanciación de las clases a las que pertenecen y **uniones** obtenidas de instanciar las asociaciones.



Diagramas de objetos (atributos y uniones)

- En el caso de aparecer atributos estos tendrán un valor asignado.
- Las uniones surgen de la instanciación de las relaciones, y por tanto 'unen' objetos concretos.
- El nombre de las uniones, roles y agregación, entre otros, pueden representarse también en un diagrama de objetos.



Ejercicio: hacer diagrama de objetos para ejemplo de transparencia 20.

Paquetes: Definición

- Permiten **dividir un modelo en partes** manejables mediante la agrupación de clases u otros paquetes.
- Son útiles, en el análisis, si debe abordarse el modelado de un sistema complejo. En concreto para:
 - Repartir el trabajo entre los miembros del EDS.
 - Definir subsistemas.
 - Ocultar detalles irrelevantes al realizar modelos parciales del sistema.
- En los procesos de diseño y construcción, los paquetes se utilizan para el desarrollo o uso de un componente de forma más o menos independiente.

Paquetes: Notación

- Se representan mediante un icono con forma de **carpeta**.
- Dentro del paquete se pueden representar los elementos incluidos.
- Dado que es una agrupación de elementos, éstos pueden ser de cualquier tipo. P.ej. otros paquetes.

