

UT4-1-UML-Apuntes. Diagramas de clase-Relaciones.

Relaciones

En un diagrama de clases, todas las clases están vinculadas entre sí mediante relaciones

Estas relaciones ayudan al usuario a comprender a fondo la conexión entre diferentes entidades.

Una relación identifica un **vínculo**. Este vínculo puede ser:

- Entre dos o más clases
- Una clase hacía sí misma: **reflexiva**

Las relaciones se representan con una **línea** que une las clases, esta línea variará dependiendo del tipo de relación

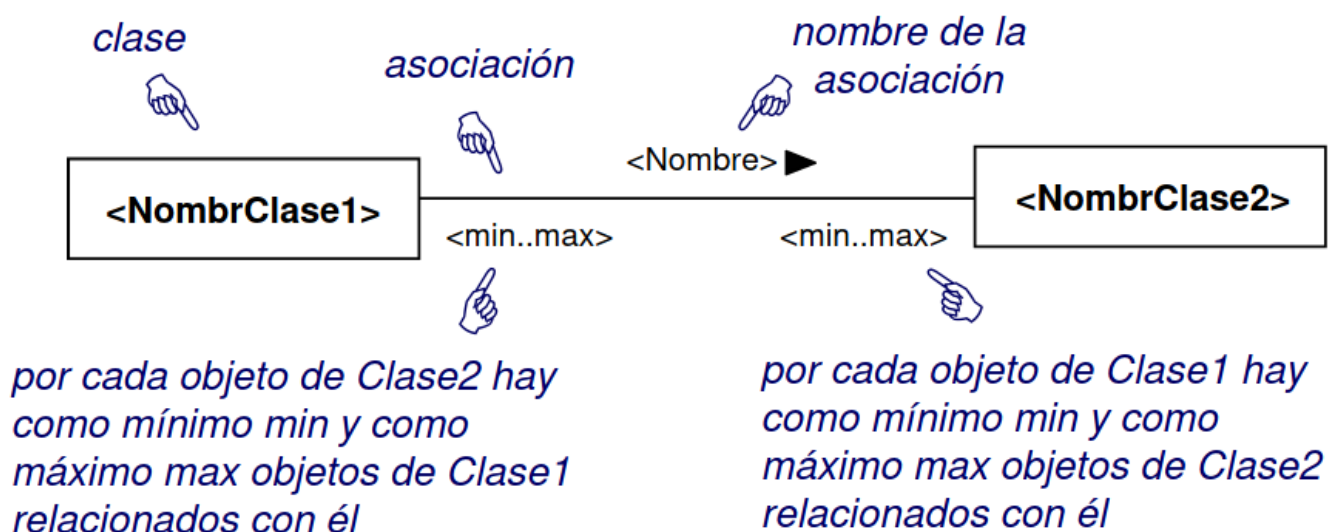
Propiedades

Las relaciones en el diagrama de clases tienen varias propiedades, que dependiendo la profundidad que se quiera dar al diagrama se representarán o no. Estas propiedades son las siguientes:

Multiplicidad

Número de elementos de una clase que participan en una relación.

Para especificar la multiplicidad de una asociación hay que indicar la multiplicidad mínima y la multiplicidad máxima (mínima..máxima).

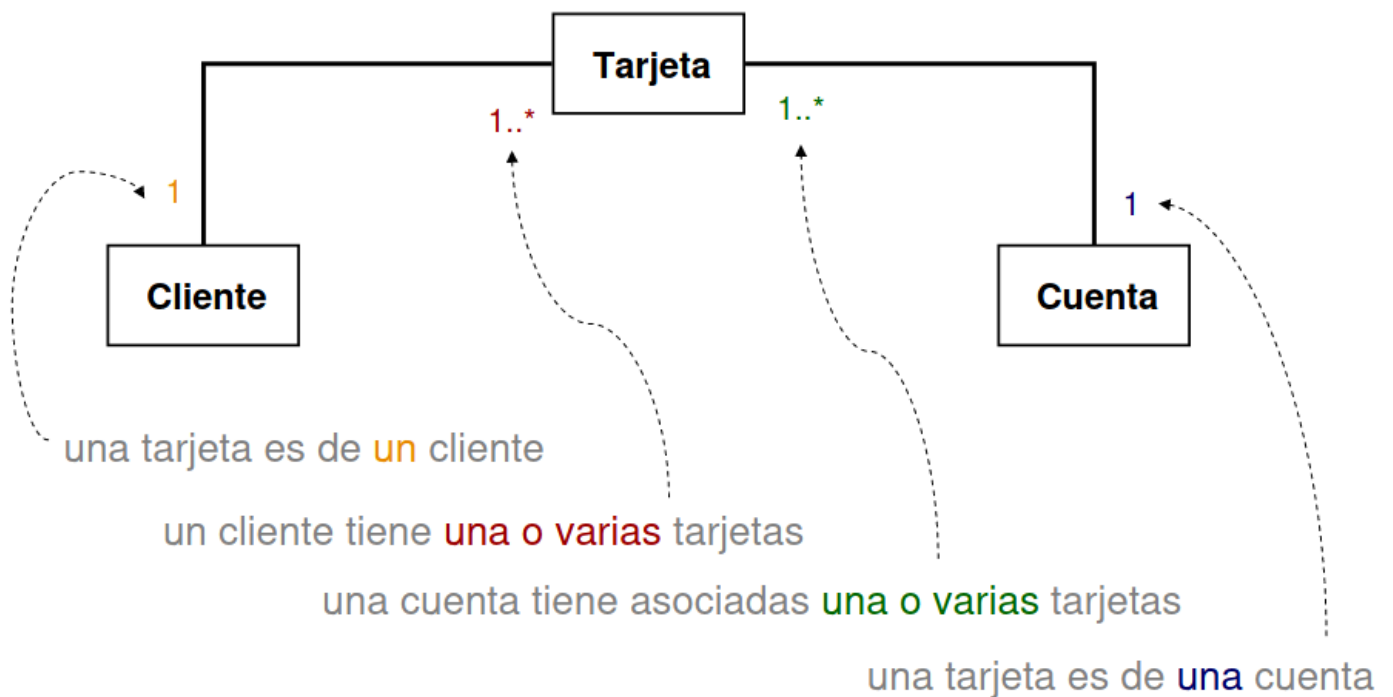


Los tipos de multiplicidad son los siguientes:

Multiplicidad	Significado
1	Uno y sólo uno
0..1	Cero o uno
N..M	Desde N hasta M
*	Cero o varios
0..*	Cero o varios
1..*	Uno o varios (al menos uno)

- Cuando la multiplicidad mínima es 0, la relación es opcional
- Una multiplicidad mínima mayor o igual que 1 establece una relación obligatoria.

Ejemplo:

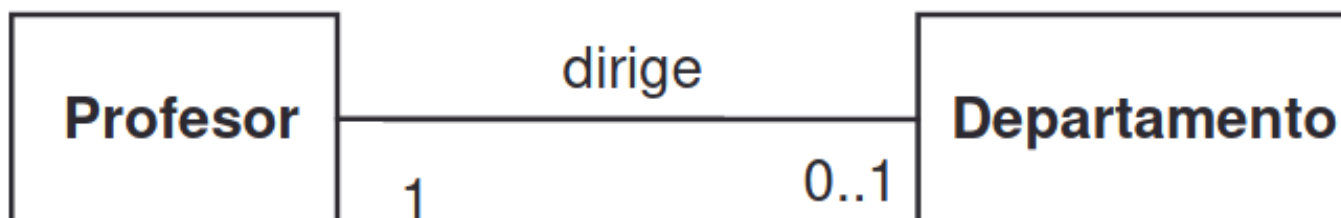


Nombre de la relación

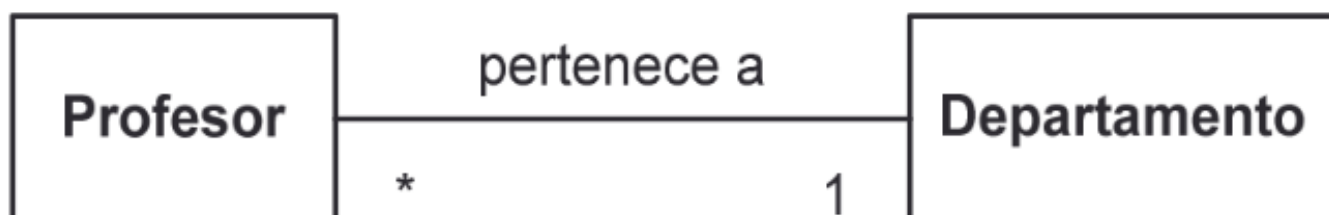
En ocasiones se escribe una indicación de la relación que ayuda a entender la relación que tienen dos clases.

Suelen utilizarse verbos

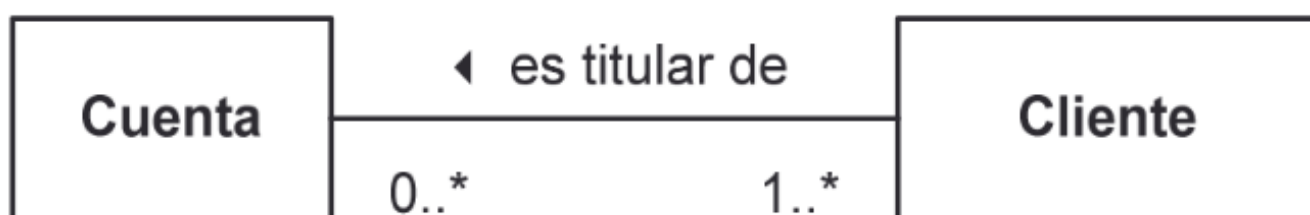
Ejemplos:



- Todo departamento tiene un director
- Un profesor puede dirigir un departamento



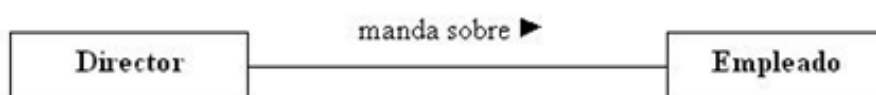
- A un departamento pueden pertenecer varios profesores.
- Todo profesor pertenece a un departamento.



- Un cliente puede o no ser titular de una cuenta.
- Una cuenta ha de tener un titular como mínimo

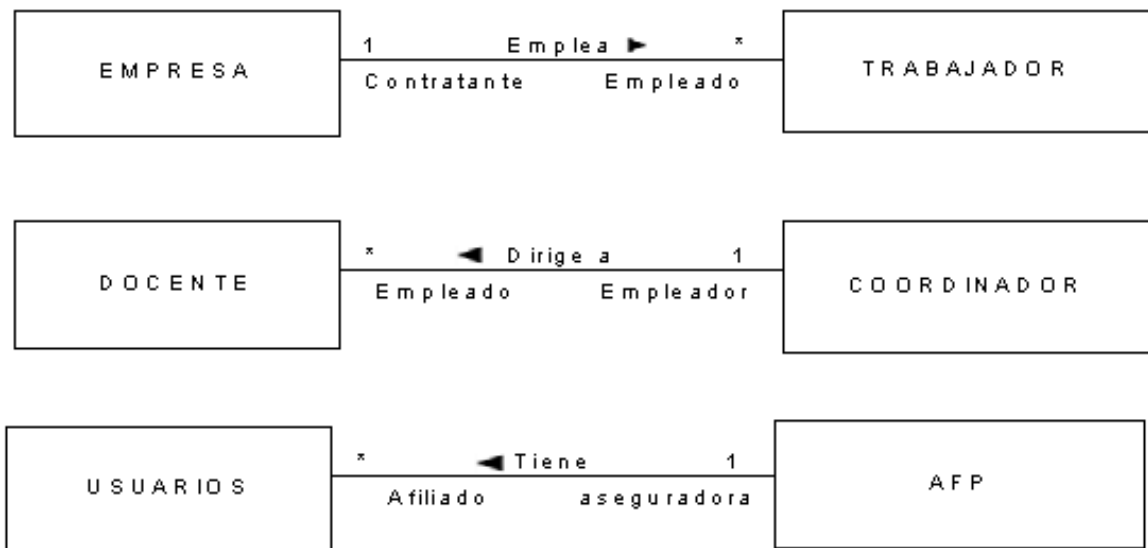
Dirección en el nombre de la relación

Para que no exista ambigüedad en el significado del nombre de una relación se le puede dar una dirección al nombre por medio de una flecha (un pequeño triángulo negro sólido) que apunte en la dirección que se pretende que el nombre sea leído.



Roles

Para indicar el papel que juega una clase en una asociación se puede especificar un nombre de rol. Se representa en el extremo de la asociación junto a la clase que desempeña dicho rol.



Tipos de relaciones

1. Asociaciones

Una asociación es una relación estructural entre elementos del mismo nivel. Ambos elementos están relacionados de una forma directa y es posible navegar de uno a otro.

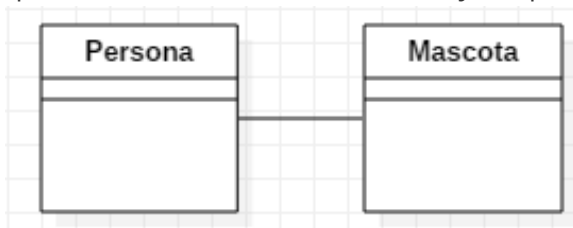
Se representa con una simple línea continua que une las clases que están incluidas en la asociación.

Se utilizan, por tanto, cuando se quiere definir una relación estructural en el modelo:

- pedidos que contienen productos
- empleados que trabajan en departamentos
- estudiantes que asisten a cursos

En todos esos casos es necesario poder navegar entre los elementos relacionados:

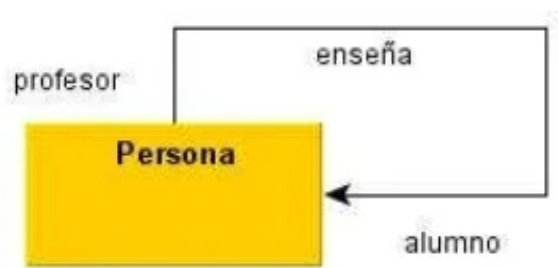
- qué productos tiene un pedido y un producto en que pedidos se incluye.
- qué estudiantes asisten a un curso y en qué cursos está matriculado un estudiante



Asociaciones reflexivas (Involutivas)

En ocasiones una clase es una asociación consigo misma, cuando una clase puede desempeñar diversos roles o papeles.

Una persona por ejemplo puede ser Profesor o alumno, para representar estas asociaciones se realiza de la siguiente manera.



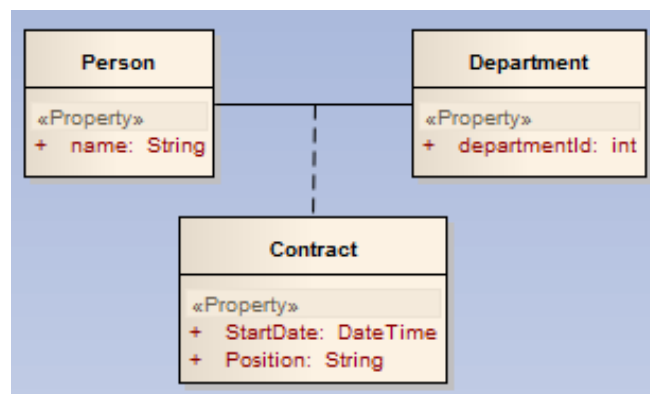
Clases asociación

En una asociación entre dos clases la propia asociación puede tener propiedades.

En este caso se usa una clase asociación que contiene atributos y métodos que no pertenecen a las clases que relaciona.

Se representa con un símbolo de clase unido por línea discontinua a una asociación.

Por ejemplo, entre un persona y un departamento puede haber una clase asociación **contrato** con atributos como el cargo** y la fecha de contratación**. Dichos atributos no pertenecen ni a la persona ni al departamento:



2. Composición



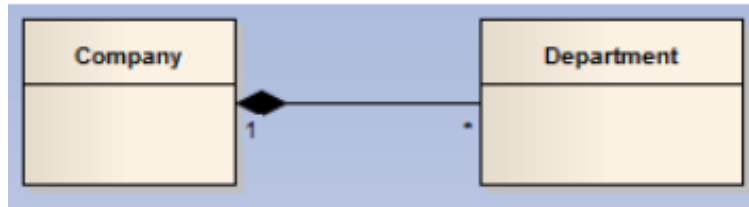
Se considera cómo un tipo de **asociación** en la que hay una fuerte relación de dependencia entre el elemento contenedor y las partes.

La clase **contenida** sólo existe si existe la clase **contenedor**

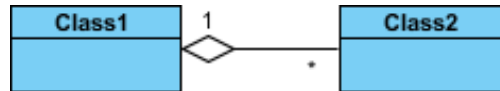
Expresa una relación **pertenece a**, pero es más fuerte que en el caso de la agregación: las partes se destruyen (dejan de tener sentido) cuando se destruye el todo.

Se representa con un rombo relleno pegado al elemento contenedor

Un ejemplo es la asociación entre una empresa y sus departamentos: éstos no podrían existir de forma independiente, no tendría sentido.



3. Agregaciones



Es un tipo especial de asociación.

Expresa una relación de "**parte de**" o "**tiene una**"

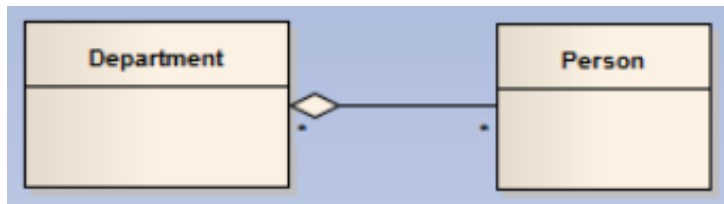
- Class2 es **parte de** Class1 o
- Class1 **tiene una** Class2

Se considera cómo otro tipo especial de **asociación** en la una clase representa una parte de otra de forma que las partes pueden existir por si mismas.

Se le llama también **composición débil**

Se representa con un rombo hueco.

Un ejemplo es la asociación entre **departamento** y **empleados**: éstos existen de forma independiente y pueden pasar de un departamento a otro a lo largo del tiempo.



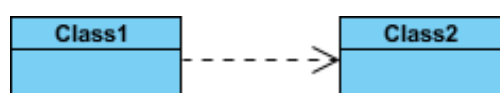
4. Dependencia

Características:

- Se utiliza si la relación entre dos clases es tal que si los cambios en la definición de una pueden causar cambios en la otra (pero no al revés).
 - Clase1 depende de Clase2
- Una relación de dependencia expresa una relación de **uso**.
 - Un cambio en una clase en particular puede afectar a otras clases que la **usan**
 - Se emplea esta relación cuando es necesario indicar que una cosa usa otra.
- Se representa mediante una línea discontinua con una flecha abierta

Este tipo de relación se utilizan para expresar que un elemento **utiliza** a otro, ya sea invocando sus métodos o accediendo a sus propiedades.

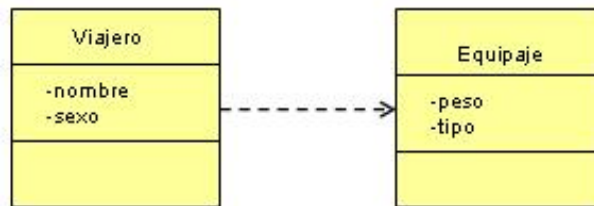
Clase1 depende de Clase2



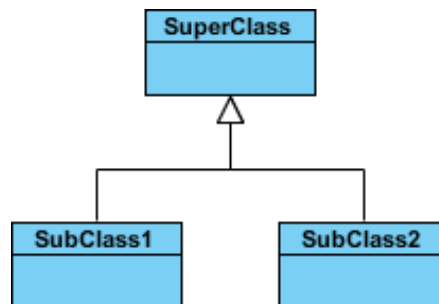
Ejemplo:

El viajero utiliza equipaje. Un equipaje no existe si no hay un viajero que lo utilice.

El equipaje pierde sentido si no hay un viajero que lo posea.



5. Herencia (o generalización)



- Expresa una relación "**es-un**".
- Si la clase es abstracta su nombre se muestra en cursiva.
 - SubClass1 y SubClass2 son **especializaciones** de Super Class.
- Se representan con una **línea continua** con una **punta de flecha hueca** que apunta desde la clase secundaria a la principal

Recursos

- [Asociaciones - David Liñan](#)
- [UML - Tipos de relaciones - ITCA](#)

tags: **ets** **ut4** **uml** **diagramas** **clase** **relaciones** **apuntes**