



Modelado estático usando el Lenguaje de Modelado Unificado (UML)

Dra. Maricela Bravo

Clases

Nombre de
la clase

Atributos

Operaciones

Una *clase* es la descripción de un conjunto de objetos que comparten los mismos

- atributos,
- operaciones,
- relaciones y
- semántica.

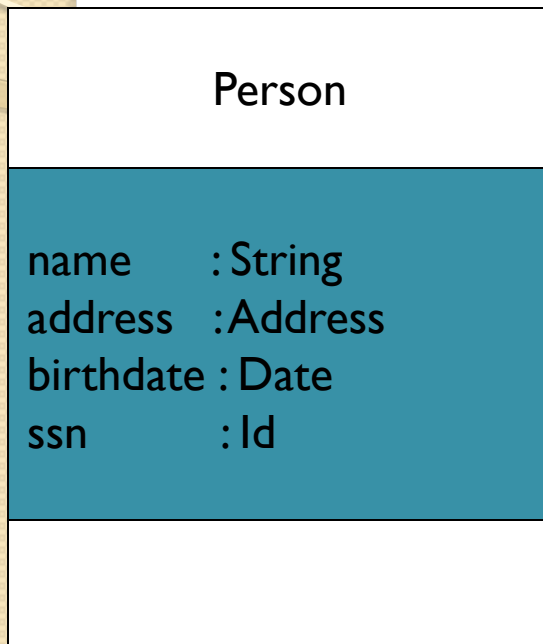
Graficamente una clase se representa con un rectángulo, incluyendo su nombre, atributos, y operaciones.

Nombre de la clase

Nombre de la clase
Atributos
Operaciones

El nombre de la clase es la única etiqueta requerida en la representación gráfica de una clase. Siempre aparece en la parte superior.

Atributos



Un *atributo* es el nombre de una propiedad que describe el objeto modelado. Los atributos aparecen en el segundo compartimento del diagrama de clases .

Atributos

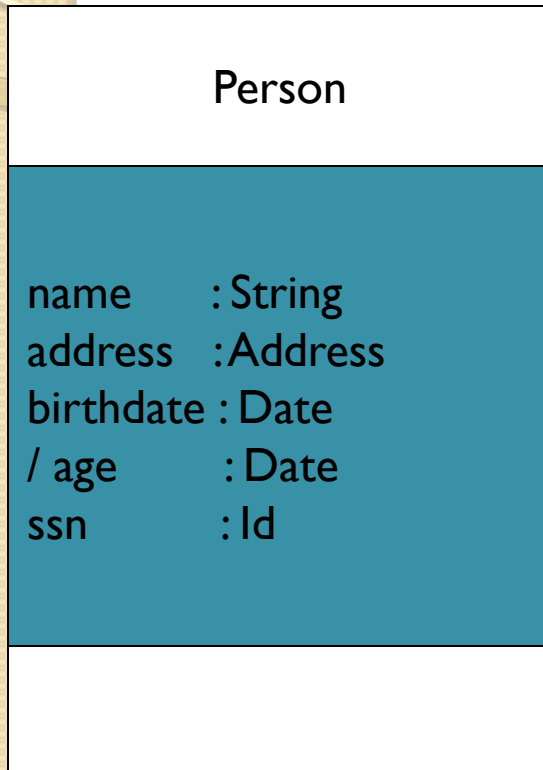
Los atributos se escriben de la sig. forma:

attributeName :Type

Un *atributo derivado* es aquel que se puede componer de otros atributos, pero en realidad no existe.

Por ejemplo, la edad de una persona puede computarse a partir de su fecha de nacimiento.

/ age : Date



Atributos

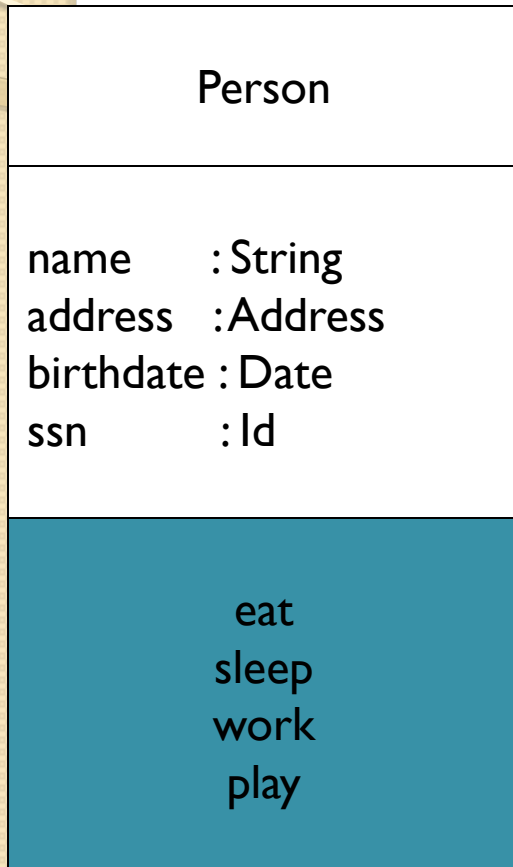
Person

+ name : String
address : Address
birthdate : Date
/ age : Date
- ssn : Id

Los atributos pueden ser:

+ public
protected
- private
/ derived

Operaciones



Las *operaciones* describen el comportamiento de la clase.

Operaciones

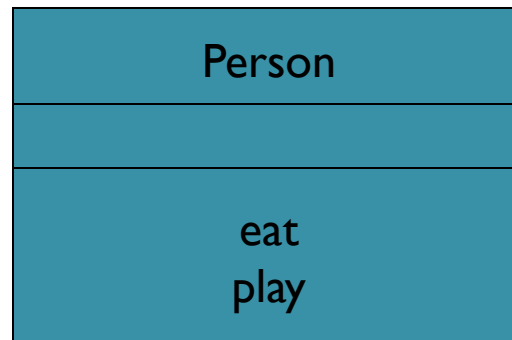
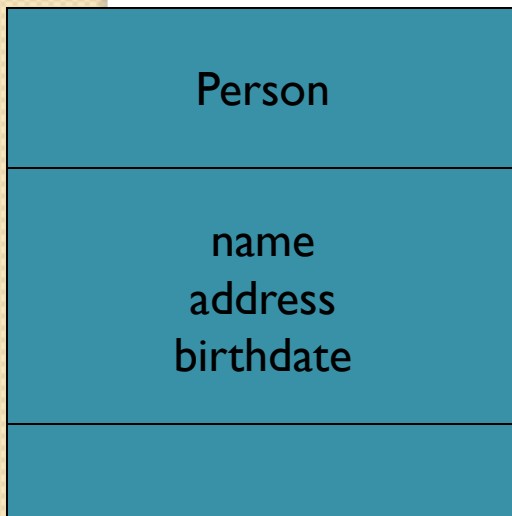
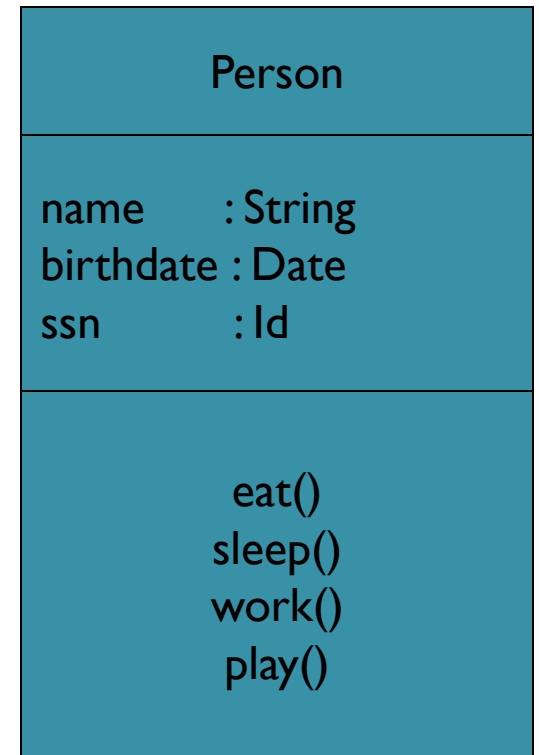
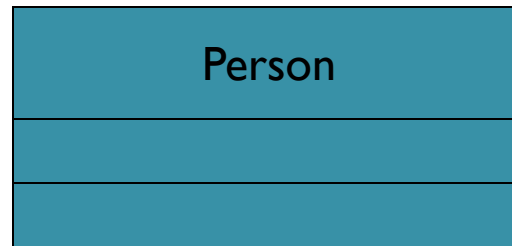
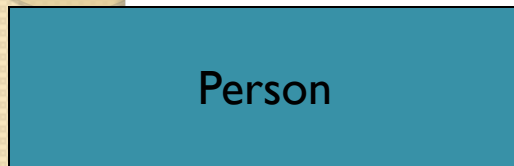
PhoneBook

```
newEntry (n : Name, a : Address, p : PhoneNumber, d : Description)  
getPhone ( n : Name, a : Address) : PhoneNumber
```

Es posible especificar una operación usando su plantilla: nombre, tipo de dato, nombre y tipo de los parámetros.

Diagrama de Clases

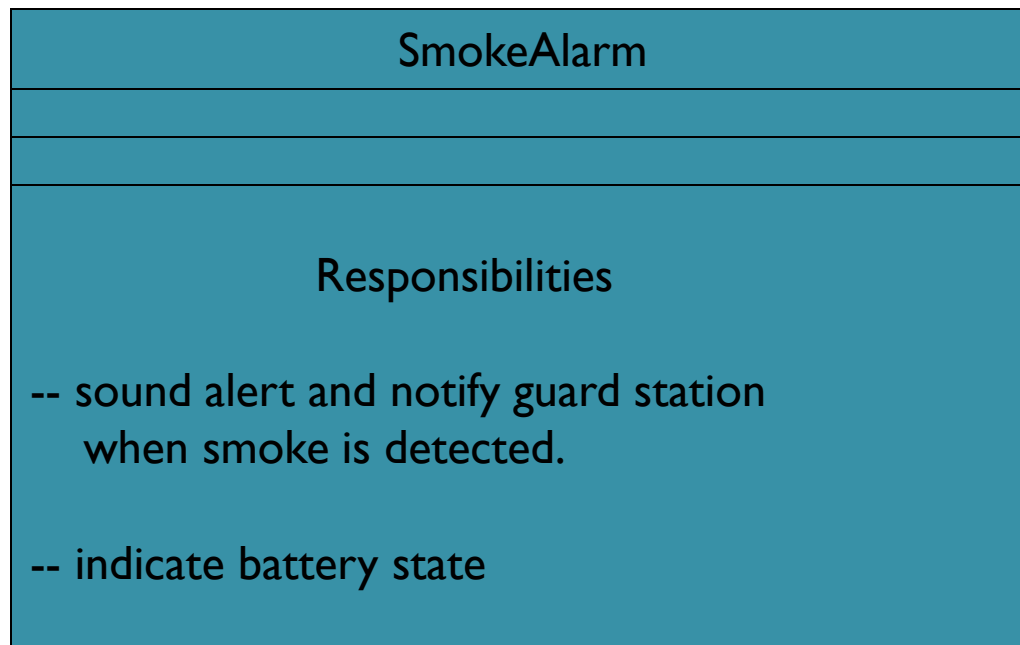
Cuando se dibuja una clase no es necesario mostrar los atributos y operaciones.



Responsabilidades de la clase

Una clase puede incluir responsabilidades.

Una responsabilidad es un contrato u obligación de la clase para realizar un servicio particular.



Relaciones entre clases

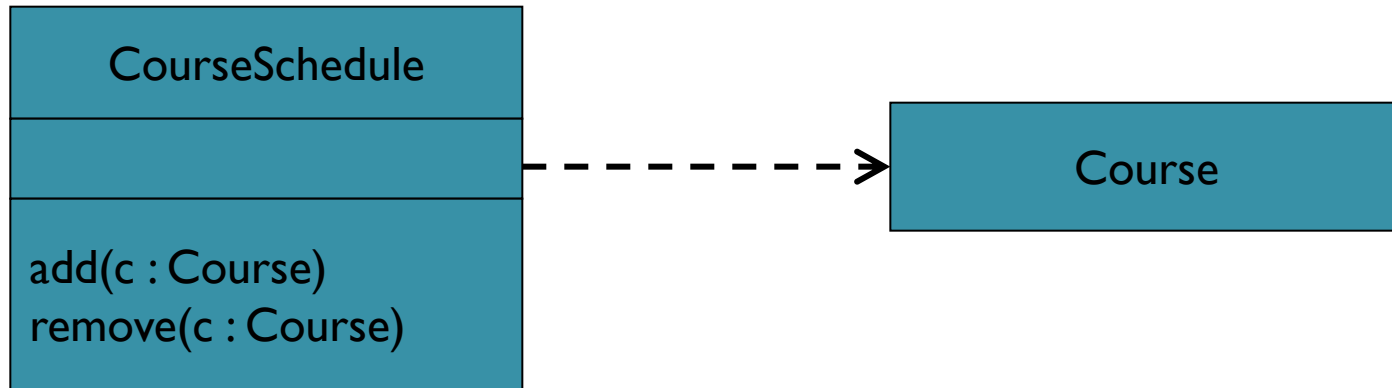
En UML las interconexiones entre objetos son modeladas como relaciones.

Existen tres clases de relaciones en UML:

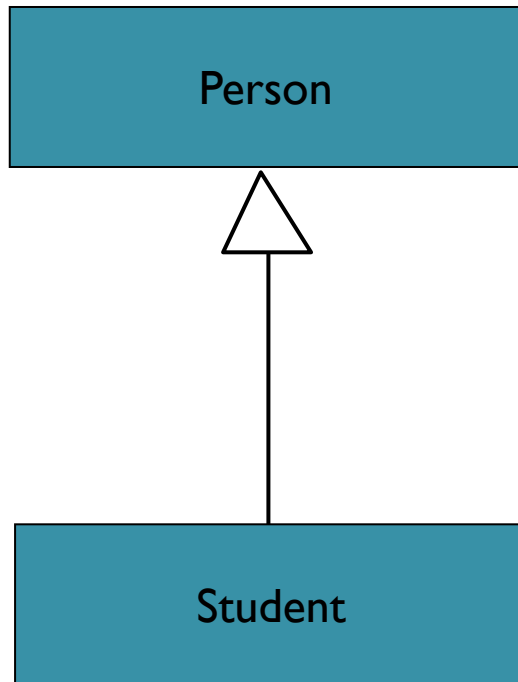
- Dependencia
- Generalización
- Asociación

Relación de Dependencia

Una relación de dependencia indica una relación semántica entre dos o más elementos.



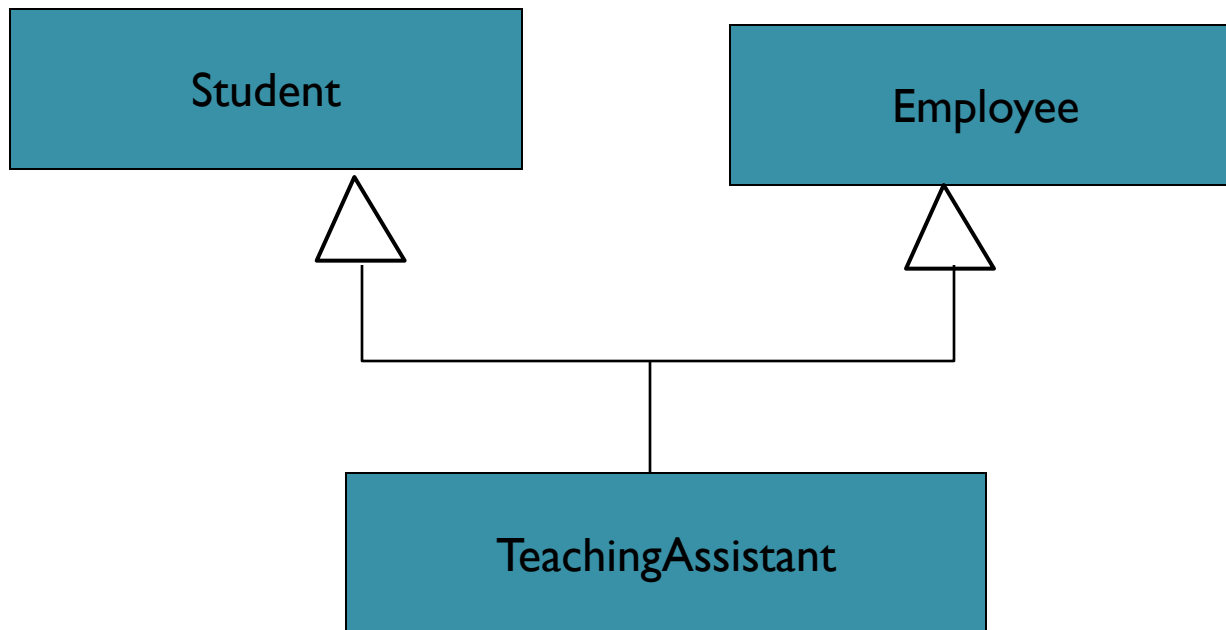
Relación de Generalización



La generalización conecta una subclase con su superclase. Denota herencia de los atributos y el comportamiento de la superclase a la subclase.

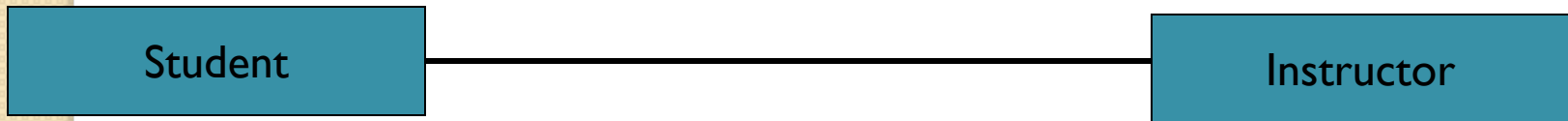
Relación de Generalización

UML permite que una clase herede de múltiples superclases, pero algunos lenguajes de programación no (Java).



Relaciones de asociación

Cuando dos clases de un modelo requieren un intercambio de mensajes se utiliza una asociación.



Relaciones de asociación

La *multiplicidad* de una asociación se refiere a una cardinalidad en la relación.



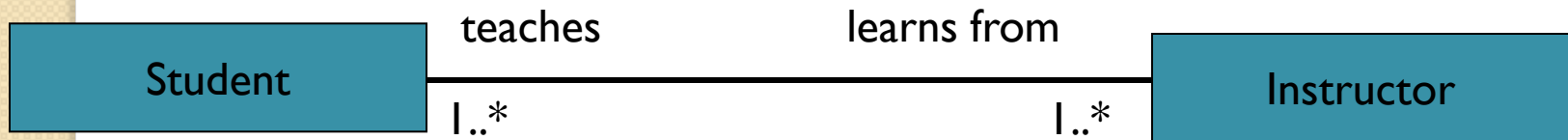
Relaciones de asociación

La *multiplicidad* de una asociación se refiere a una cardinalidad en la relación.



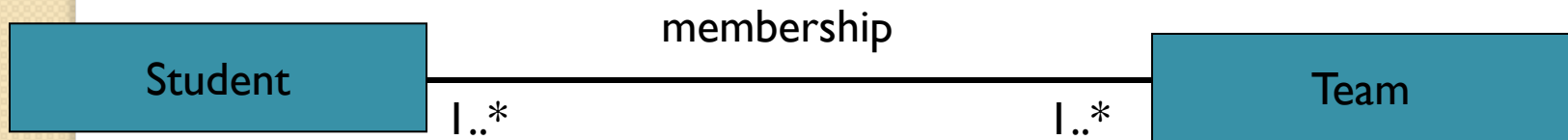
Relaciones de asociación

También es posible indicar el comportamiento de un objeto en una asociación usando roles.

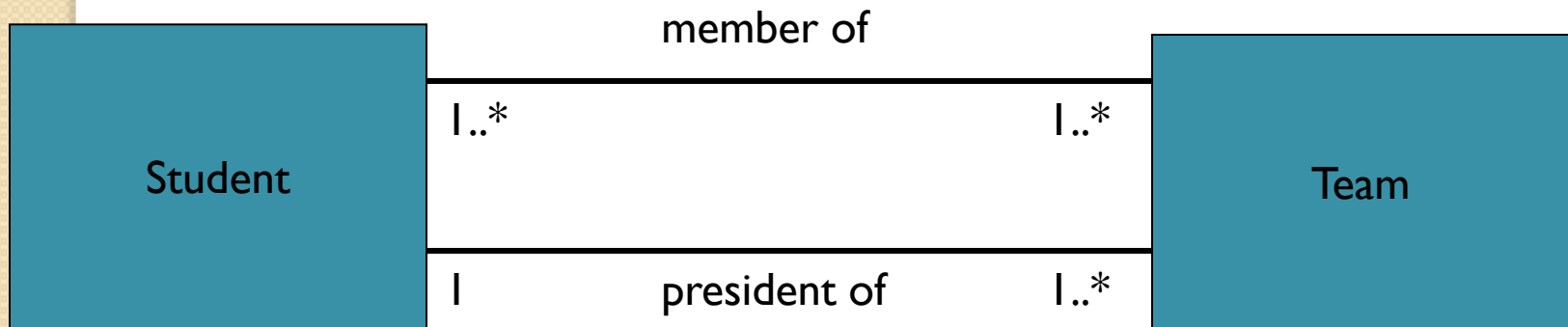


Relaciones de asociación

También es posible nombrar a la asociación.

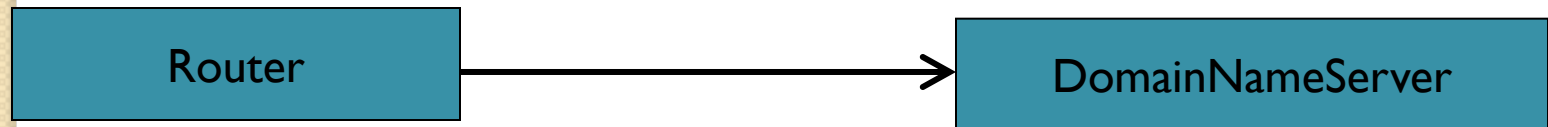


Asociaciones duales



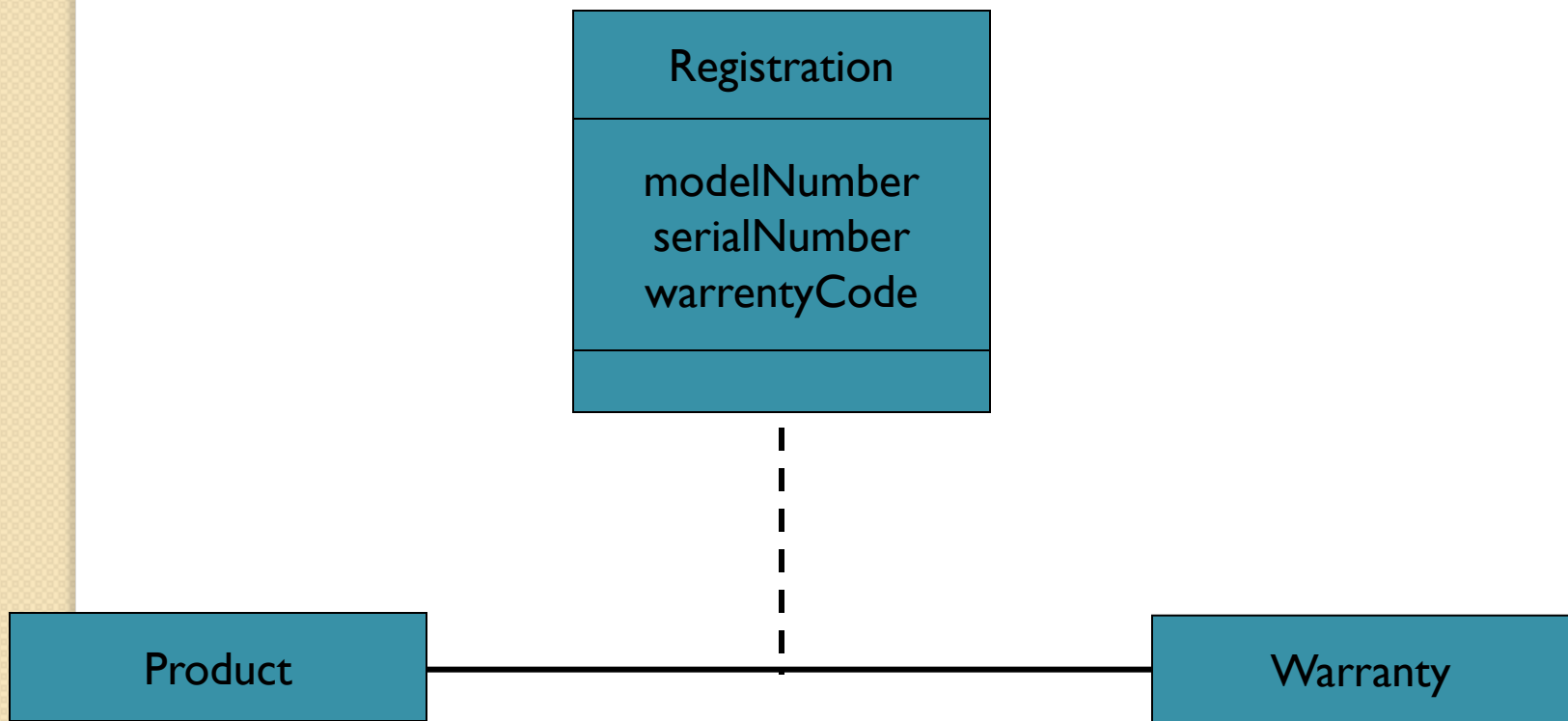
Navegabilidad de la asociación

Se puede restringir la relación de asociación definiendo la navegabilidad de la asociación.



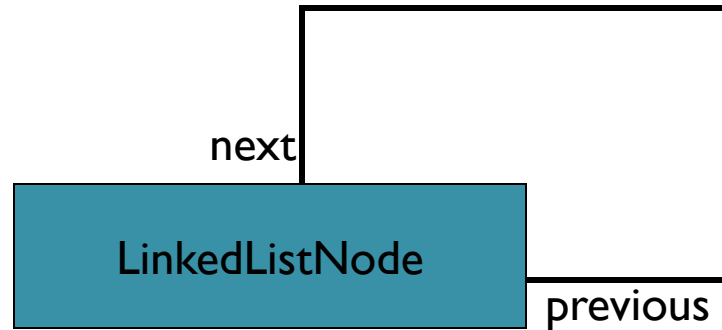
Relaciones de asociación

Las asociaciones también pueden ser objetos, llamados clases de enlace o clases de asociación.



Relaciones de asociación

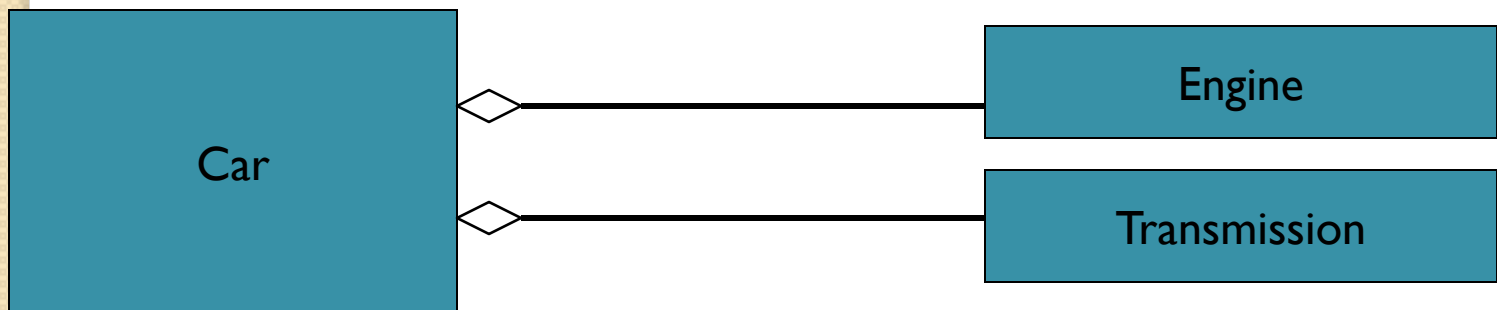
Una clase puede tener una asociación con ella misma
(*self association*).



Relaciones de asociación

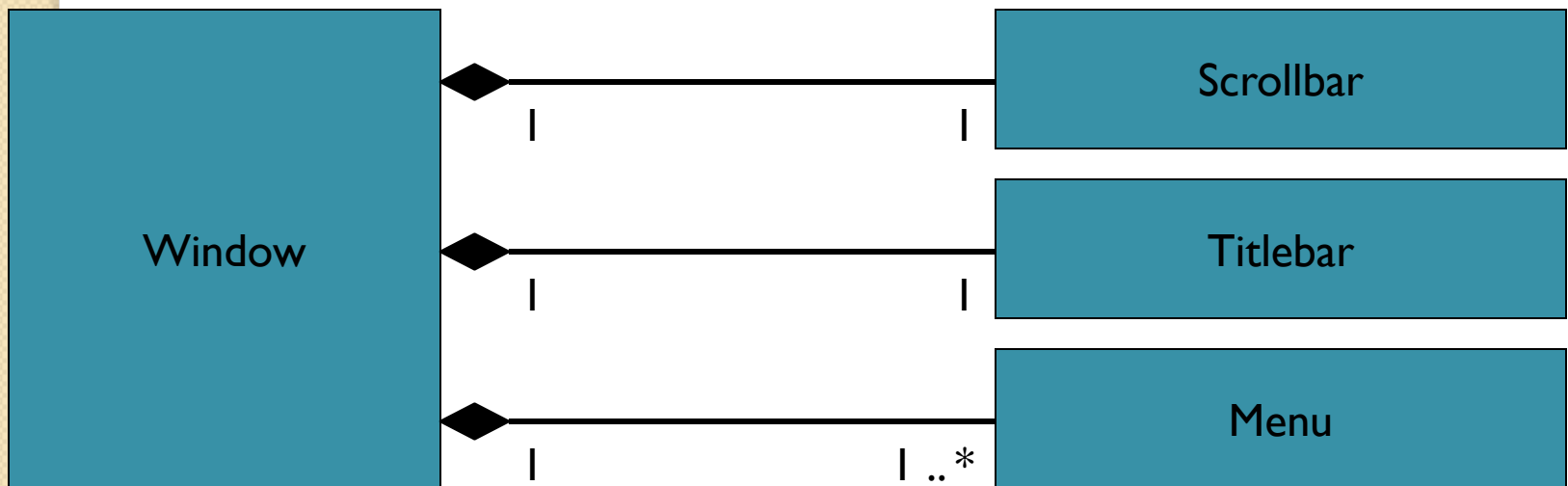
Es posible modelar objetos que contienen otros objetos a través de asociaciones especiales llamadas *aggregations* y *compositions*.

Una *aggregation* especifica una relación tipo todo-parte entre un *aggregate* (todo) y una parte constituyente, donde la parte puede existir independientemente del *aggregate*.

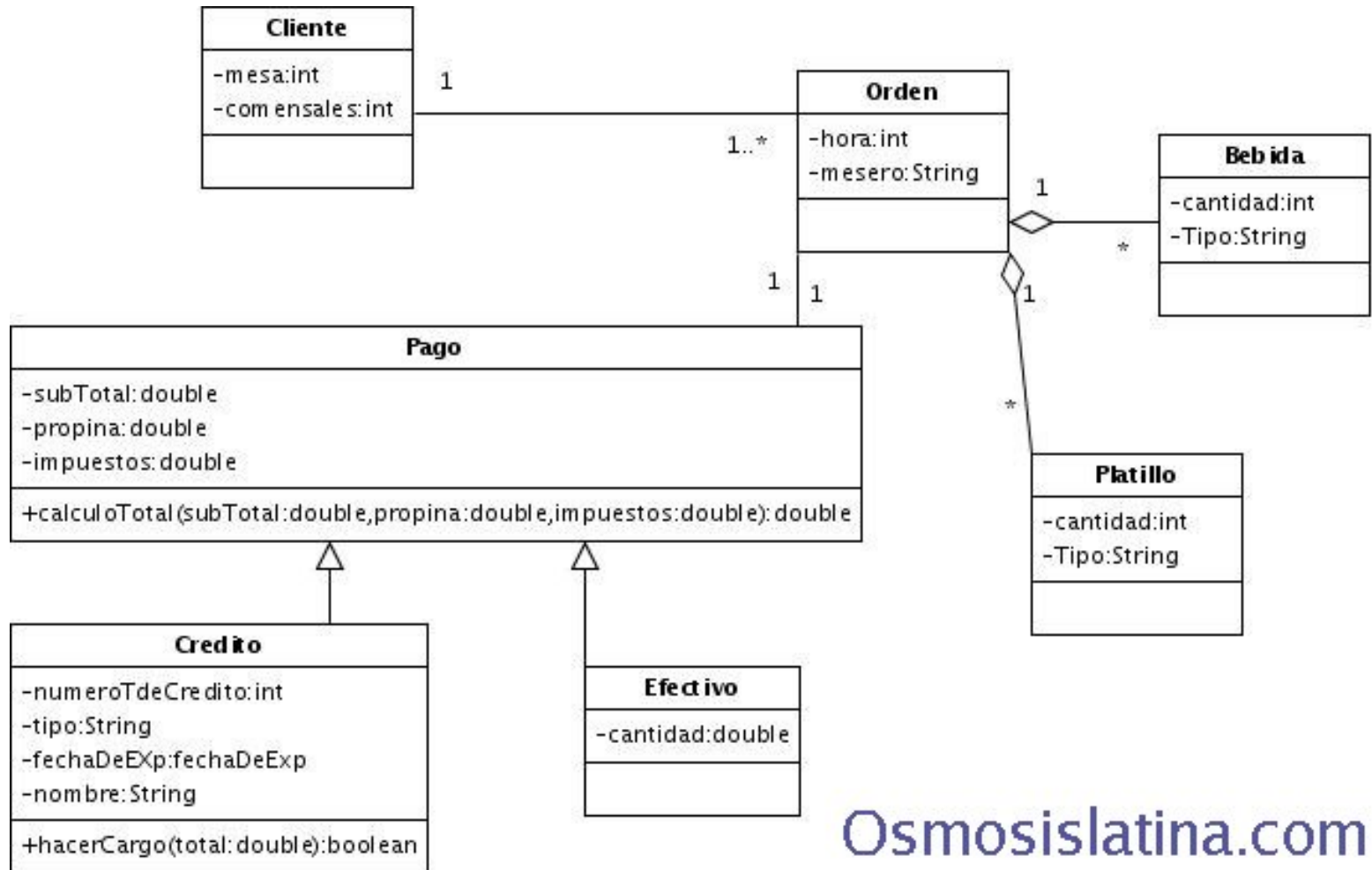


Relaciones de asociación

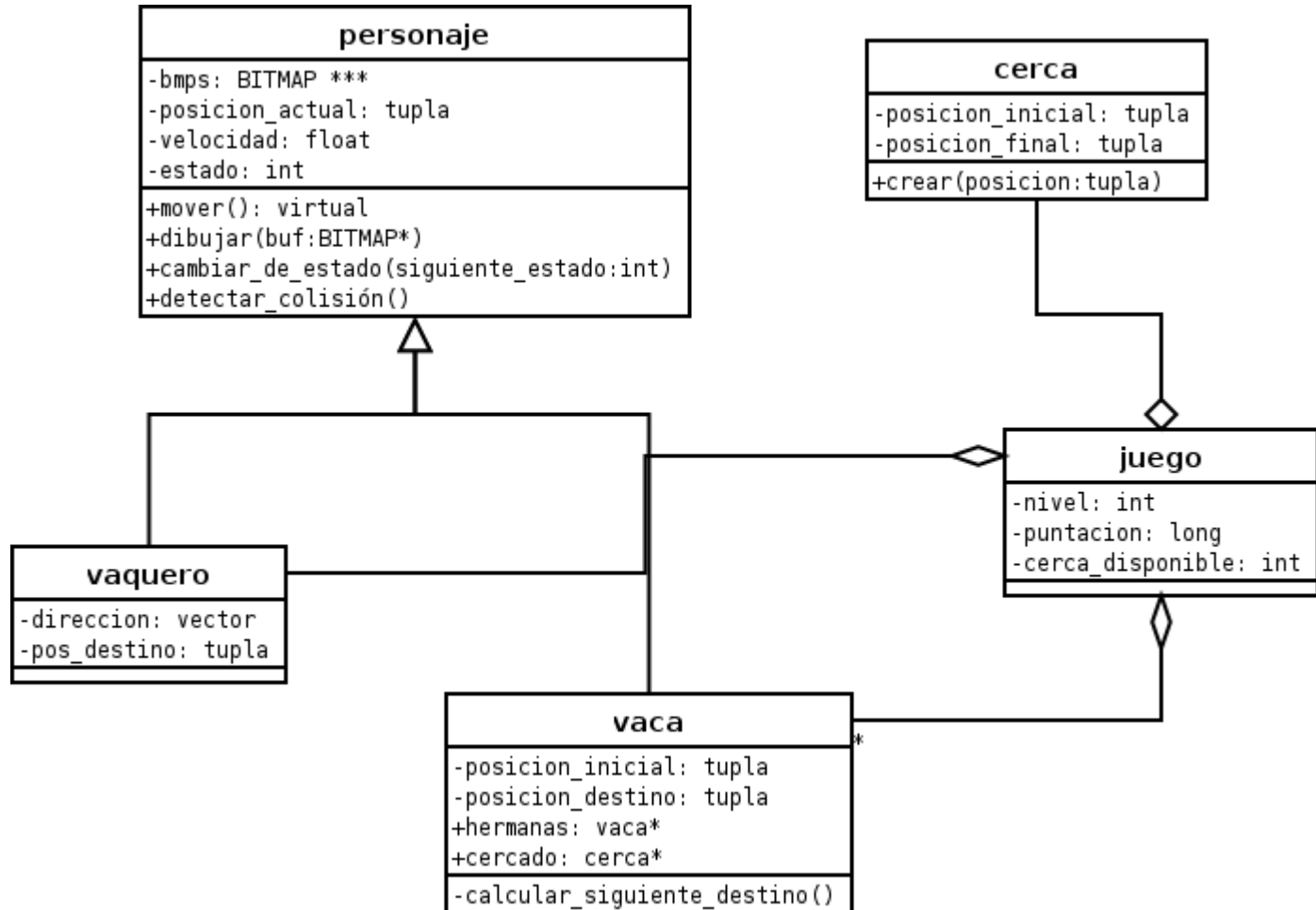
Una composición indica una pertenencia fuerte entre las partes y el todo.



Ejemplos



Ejemplos



Ejemplos

