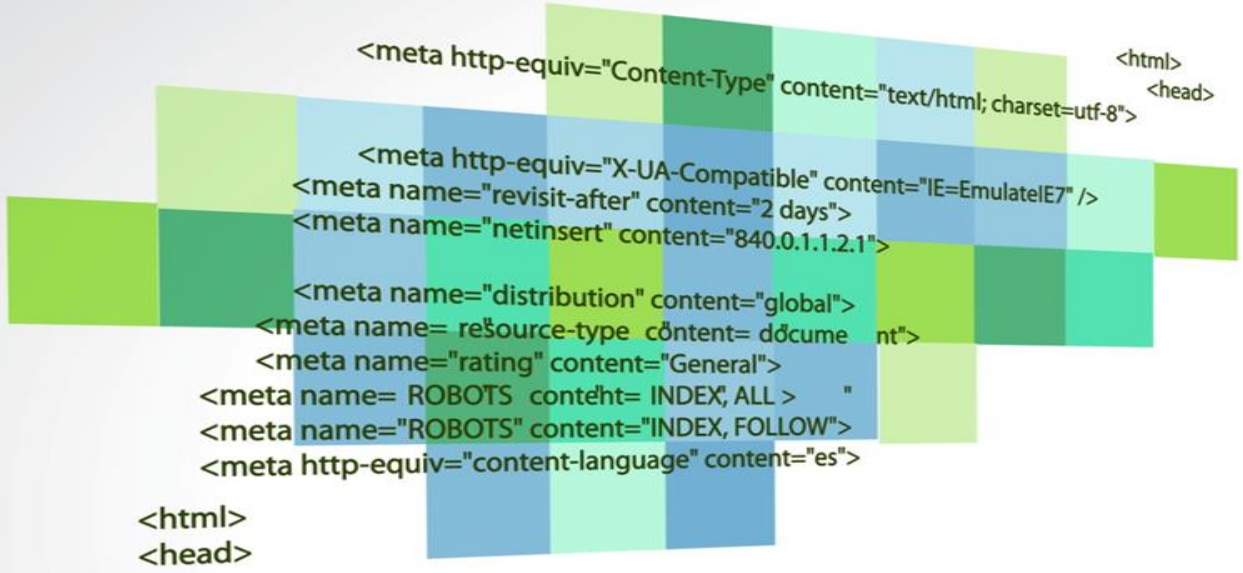


PROGRAMACIÓN ORIENTADA A OBJETOS

Relaciones entre Clases

Relaciones entre Clases

UNIDAD 2

LOGRO DE LA UNIDAD 2

- Al finalizar la unidad el alumno identifica las relaciones entre clases de un sistema.

Agenda

1. Relación
2. Tipos de Relación
   - 2.1 Asociación
   - 2.2 Agregación / Composición
   - 2.3 Generalización / Especialización
3. Conclusiones

1. RELACION

- Una relación es una conexión semántica entre elementos de un modelo.

2. Tipos de Relación

- Las relaciones entre clases que existen son:
  - 2.1 Asociación
  - 2.2 Agregación / Composición
  - 2.3 Generalización / Especialización

2.1 Asociación

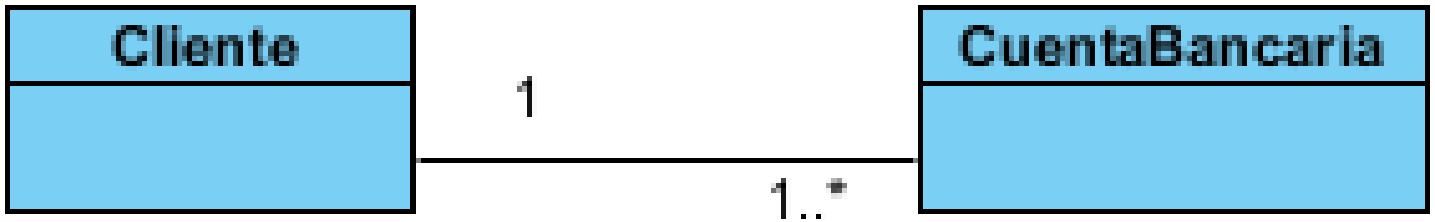
- Es una relación entre clases. Implica una dependencia semántica.
- Es cuando un objeto de una clase requiere un objeto de otra clase para hacer su trabajo. “Para cada X hay un “Y”.
- Se representa por medio de una línea continua entre dos clases.

Asociación – Diagrama UML

- **Aquí, cada cliente vive en una dirección y dirección es utilizada por un solo cliente (es decir, un objeto Cliente está asociado sólo a un objeto Dirección).**

2.1.1 Multiplicidad (cardinalidad)

- La multiplicidad es el número de instancias que tiene una clase en relación con otra clase



Asociación uno a muchos entre clases que representa un cliente y sus cuentas bancarias

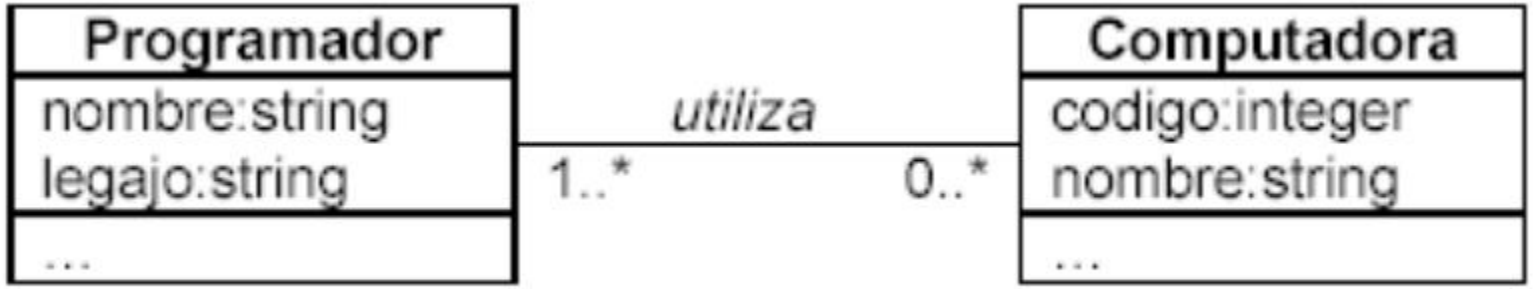
2.1.3 Multiplicidad

La multiplicidad puede especificarse con un solo entero o con un rango **n..m** donde n es el limite inferior y m es el limite superior. Se puede utilizar un asterisco para denotar que no existe un límite superior.

0..1	Cero o una instancia
*0..** ó *	Cero o más instancias
1	Exactamente una instancia
1..*	Una o más instancias

Las asociaciones pueden clasificarse de acuerdo a su multiplicidad, uno a uno, uno a muchos y muchos a muchos.

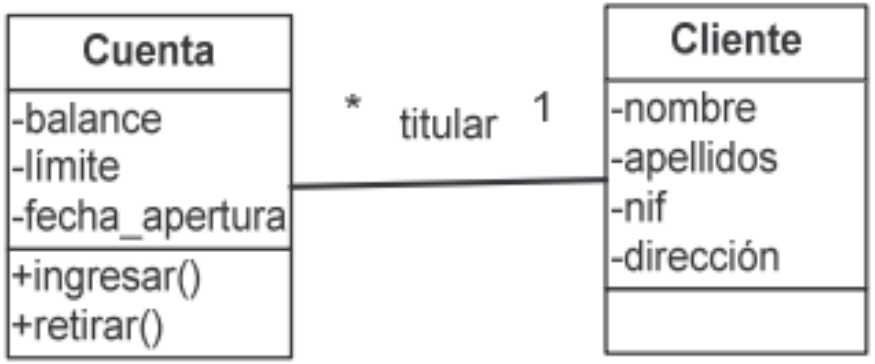
2.1.3 Multiplicidad



Este diagrama indica que cada programador tendrá varias computadoras (posiblemente ninguna), y que cada computadora será usada por al menos un programador.

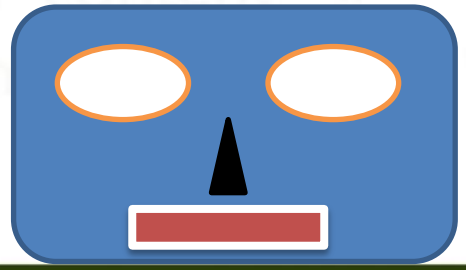
Ejemplos

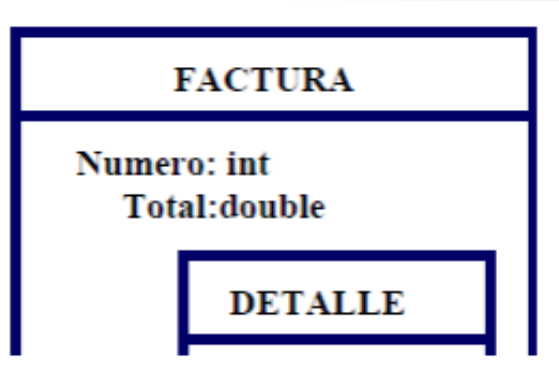

Cliente
OrdenCompra
1
0..*


Cuenta
-balance
-límite
-fecha_apertura
+ingresar()
+retirar()
*
titular
1
Cliente
-nombre
-apellidos
-nif
-dirección

2.2 Relaciones de Composición / Agregación

- Son formas especiales de relación donde una clase está compuesta de otra clase.
- En tal forma que un atributo de una clase es un objeto de otra clase.
- Se les conoce como relación TODO-PARTE.

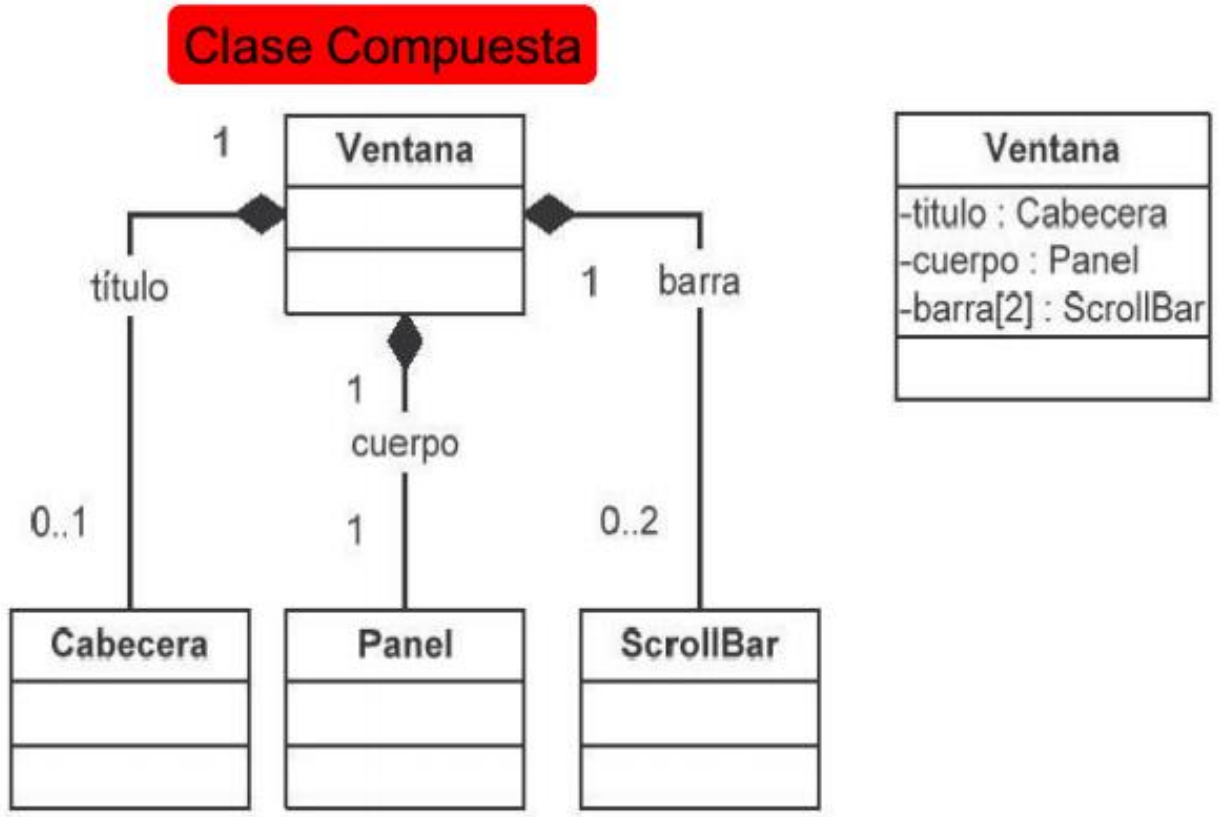




2.2.1 Composición

- Si la relación es fuerte, tal que objetos de la clase PARTE son dependiente de la existencia de la clase TODO entonces la relación es **composición**. La clase **TODO**, tiene la responsabilidad de la creación y destrucción de objetos de sus componentes

2.2.2 Composición



2.2.3 Agregación

- Al contrario, si la existencia de objetos de la clase **PARTE** es **independiente** de la existencia de objetos de la clase **TODO**, entonces la relación es **agregación**.

2.2.4 Representación

Agregación / Composición

- Su representación gráfica es la siguiente:

3. Conclusiones

- Las relaciones entre clases nos permiten entender mejor como es que ciertas clases utilizan o colaboran con otras para lograr un objetivo del Sistema.
- Muchas veces el significado de la relación entre clases dependen exclusivamente del negocio, no hay que inventarlo o aplicar solo nuestro sentido común para idnetificarlos.
- La representación de las relaciones entre clases se realizan utilizando la especificación UML.