Principios de la Tecnología de Objetos

Paradigmas de la Orientación a Objetos

Copyright

Paradigmas de la O.O.

- Los paradigmas de la Orientación a Objetos son:
  - Abstracción.
  - Encapsulación.
  - Ocultamiento.
  - Herencia.
  - Polimorfismo.
- Cualquier lenguaje O.O. debe implementar estos conceptos.

Abstracción

- Consiste en la generalización conceptual de los atributos y comportamiento de un determinado conjunto de objetos.
- La clave de la programación O.O. está en abstraer los métodos y los datos comunes a un conjunto de objetos y almacenarlos en una clase.
- Hay que centrarse en lo que es y lo que hace un objeto, antes de decidir cómo debería ser implementado.

Encapsulación y ocultamiento

- Consiste en separar el aspecto externo del objeto, al cual pueden acceder otros objetos, del aspecto interno del mismo, que es inaccesible para los demás.
- Permite tratar a un objeto como una caja negra.
- Permite que se modifique la implementación interna de un objeto sin afectar a los clientes que lo utilizan.

Relaciones

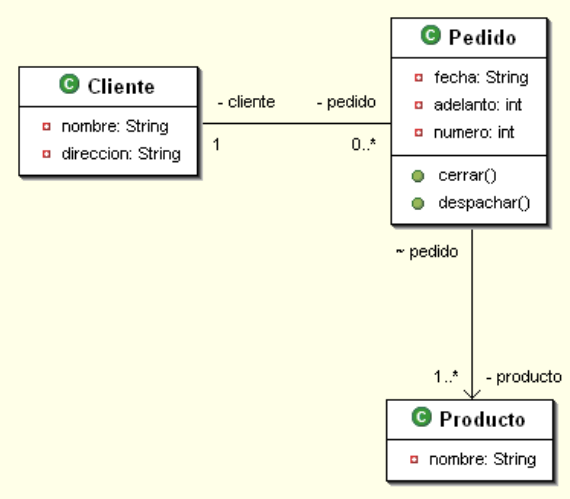
- Las clases no existen aisladas sino que tienen dependencias entre ellas.
- Los distintos tipos de relaciones son:
  - Asociación.
  - Agregación y Composición.
  - Herencia.
  - Relaciones dinámicas: Mensajes.

Relación de asociación

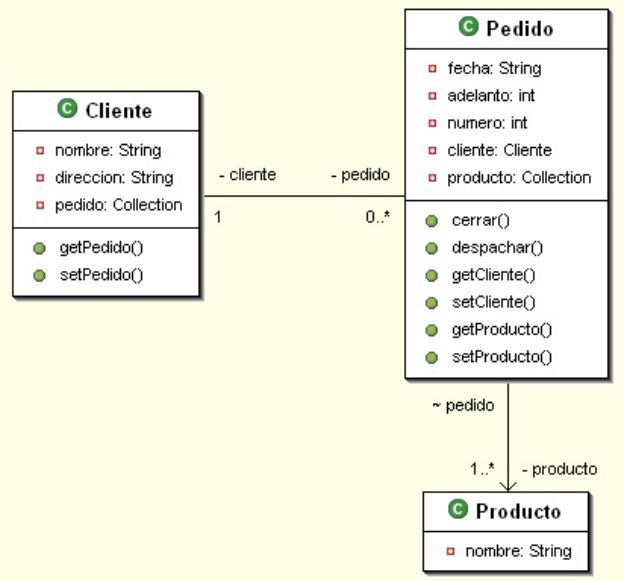
- Representa la dependencia más general entre clases.
- Representa una dependencia semántica entre dos clases.
- Por defecto es bidireccional, aunque se puede restringir a una sola dirección.
- Tiene multiplicidad. Es la propiedad que expresa el número de instancias de cada clase que participa en la relación:

(0..1, 1, 0..*, 1..*)

Ejemplo



Ejemplo



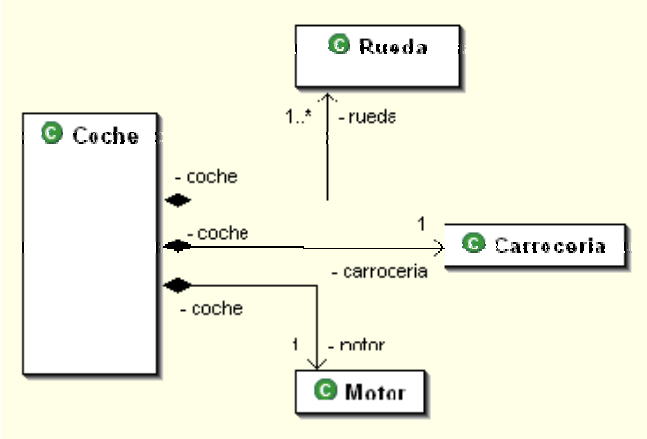
Relación de agregación

- Es una forma particular de asociación que expresa un acoplamiento mas fuerte entre objetos.
- Indica que los objetos de una clase contienen o están formados por objetos de otras clases.
- No siempre precisa contención física.
- Por tanto, un objeto que representa el ‘todo’, está asociado con un conjunto de objetos que representan sus componentes.

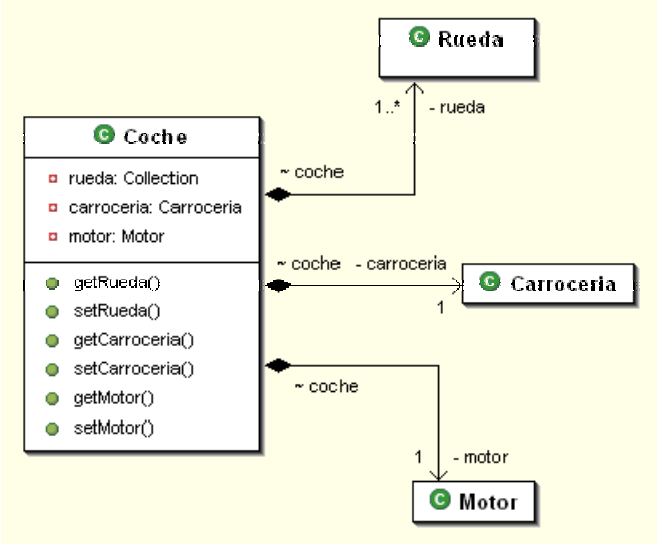
Relación de composición

- Se trata de una relación de agregación fuerte.
- Un objeto no puede existir si no existen los objetos de los que está compuesto.

Ejemplo



Ejemplo



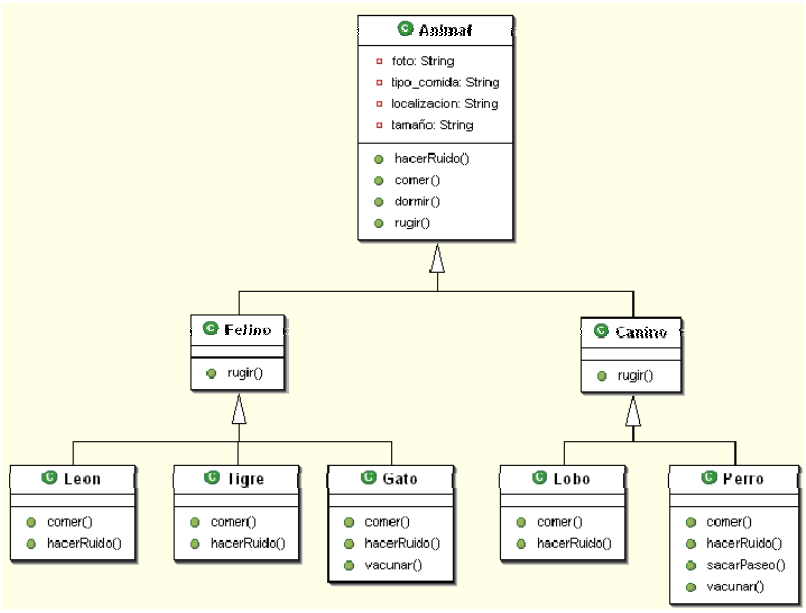
Relación de herencia

- Se basa en la existencia de relaciones de generalización/especialización entre clases.
- Las clases se disponen en una jerarquía, donde una clase hereda todos los atributos y operaciones de las clases superiores en la jerarquía.
- Una clase puede tener sus propios atributos y operaciones adicionales a lo heredado.
- Una clase puede modificar los atributos y operaciones heredadas.

Relación de herencia

- Las clases por encima en la jerarquía a una clase dada, se denominan superclases.
- Las clases por debajo en la jerarquía a una clase dada, se denominan subclases.
- Una clase puede ser superclase y subclase al mismo tiempo.
- Tipos de herencia:
  - Simple.
  - Múltiple (no soportada por todos los lenguajes O.O.)

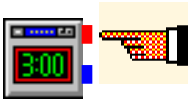
Ejemplo

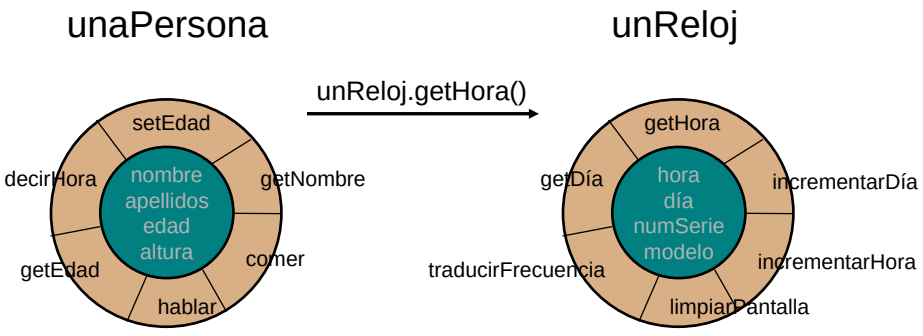


Relación dinámica: mensaje

- Un mensaje es un comando o petición que se le envía a otro objeto.
- Requiere el conocimiento previo del interfaz del objeto receptor.
- Que es dinámica significa que se observa en ejecución, no en el diseño.

Ejemplo

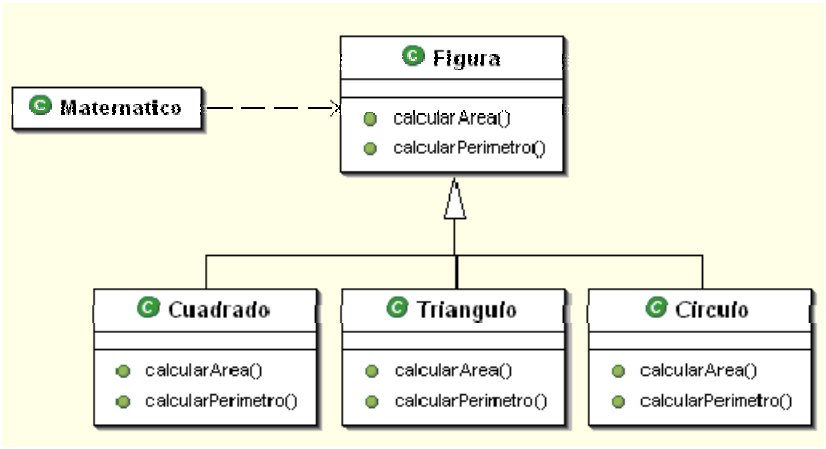




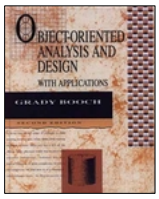
Polimorfismo

- Permite implementar múltiples formas de un mismo método, dependiendo cada una de ellas de la clase sobre la que se realice la implementación.
- Esto posibilita desencadenar operaciones diferentes en respuesta a un mismo mensaje, en función del objeto que lo reciba.

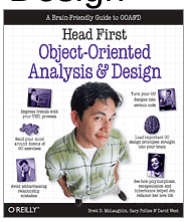
Ejemplo



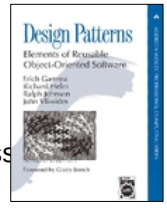
Bibliografía



- Object-Oriented Analysis and Design
  Grady Booch.
  Addison-Wesley.



- Head First Object-Oriented Analysis and Design
  Brett McLaughlin, Gary Pollice, David West
  O'Reilly



- Design patterns
  Erich Gamma, Richard Helm, Ralph Johnson, John M. Vliss
  Addison-Wesley.