



conceptos básicos

del paradigma orientado a objetos

objeto

Instancia de una clase.

digamos, una "cosa"

objeto

representado en UML

<u>p1 : Perro</u>
nombre = 'Firulais' edad = 5
comer() dormir() ladrar()

<u>p2 : Perro</u>
nombre = 'Toby' edad = 4
comer() dormir() ladrar()

instancia

Manifestación concreta
de una abstracción.

por ejemplo, un objeto

abstracción

Características esenciales
de una entidad
que la distinguen
de otros tipos de entidades.

por ejemplo, una clase

clase

Descripción de un conjunto de objetos que comparten los mismos atributos¹, operaciones², relaciones³ y semántica.

1- características estructurales

2- características de comportamiento

3- conexiones semánticas entre elementos

digamos, un “molde” o plantilla para crear “cosas”

clase

representada en UML



atributo

Propiedad con nombre de un clasificador¹
que describe un rango de valores
que pueden obtener
las instancias² de la propiedad.

operación

Implementación de un servicio
que puede ser requerido
a cualquier instancia² del clasificador¹
para que muestre un comportamiento.

1 y 2- atención acá

características

atributos y operaciones

Perro
nombre edad
comer() dormir() ladrar()

<u>p1 : Perro</u>
nombre = 'Firulais' edad = 5
comer() dormir() ladrar()

clasificador

Mecanismo que describe
características¹ estructurales²
y de comportamiento³.

1- propiedades encapsuladas

2- estáticas

3- dinámicas

por ejemplo, una clase

relación

Conexión semántica
entre elementos.

generalización

Relación de
especialización/generalización
en la cual las instancias
del elemento especializado ¹
pueden sustituir a las del general ².

1- el "hijo"

2- el "padre"

generalización *representada en UML*

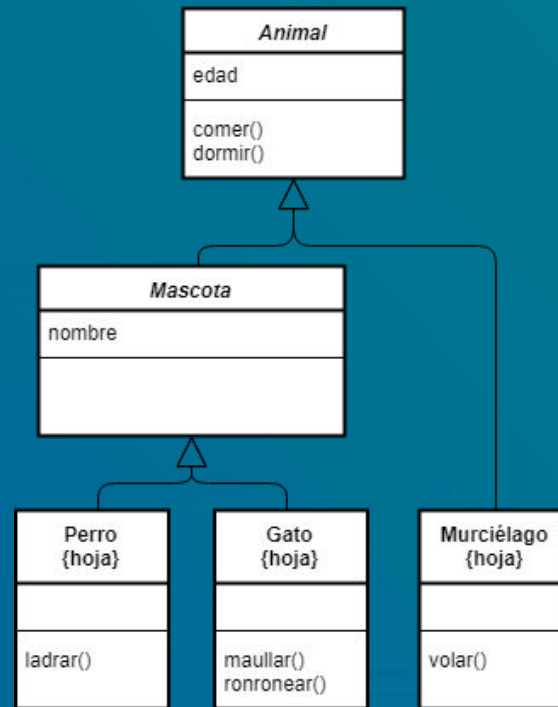


herencia

Mecanismo por el que elementos más específicos incorporan la estructura y comportamiento de elementos más generales.

puede ser simple (sólo un “padre”) o múltiple (más de uno)

herencia *entre clases*

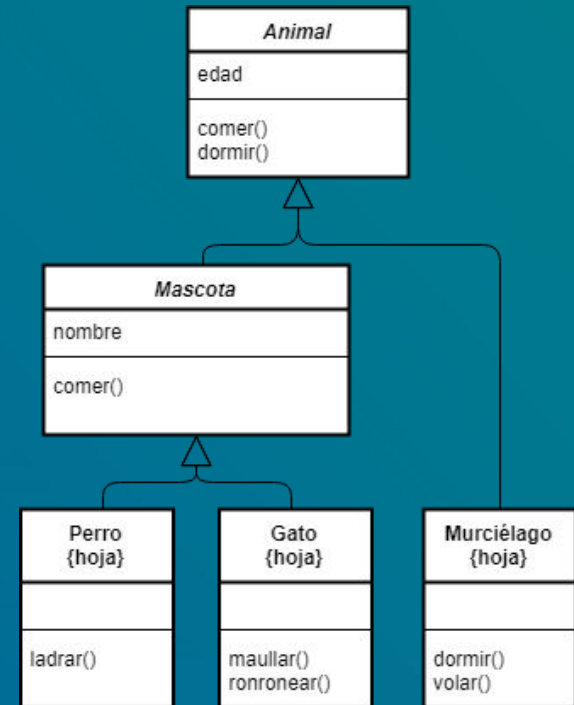


polimorfismo

Mecanismo por el que
una operación puede tener
diferentes métodos asociados
en clases conectadas
por generalizaciones.

*una implementación de una operación en un “hijo”
redefine la implementación de la misma operación en el “padre”*

polimorfismo *entre clases*



dependencia

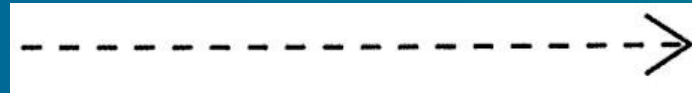
Relación semántica entre elementos
en la cual un cambio a uno¹
puede afectar a la semántica del otro².

1- el independiente

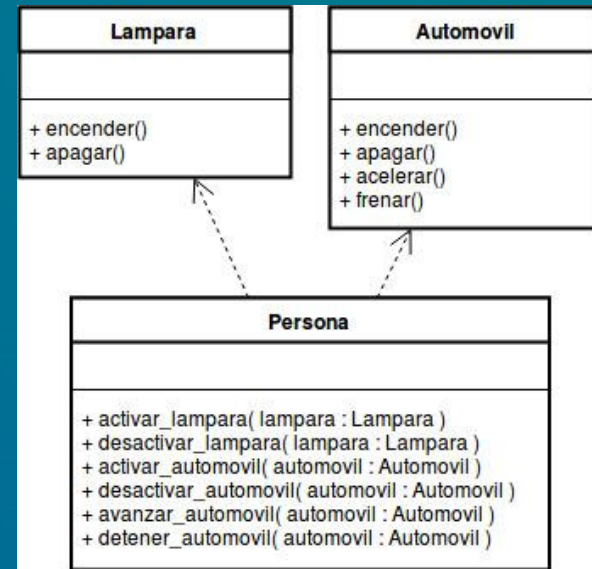
2- el dependiente

dependencia

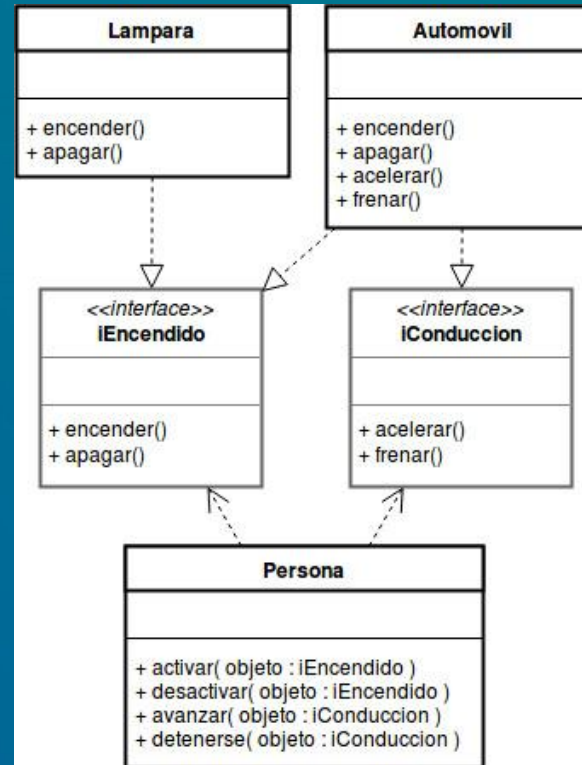
representada en UML



dependencia *entre clases*



dependencia *entre clases e interfaces*



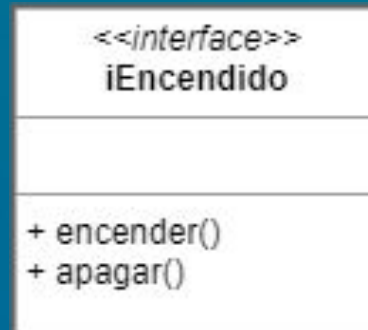
interfaz

Colección de operaciones
que se utiliza
para especificar un servicio
de una clase o componente.

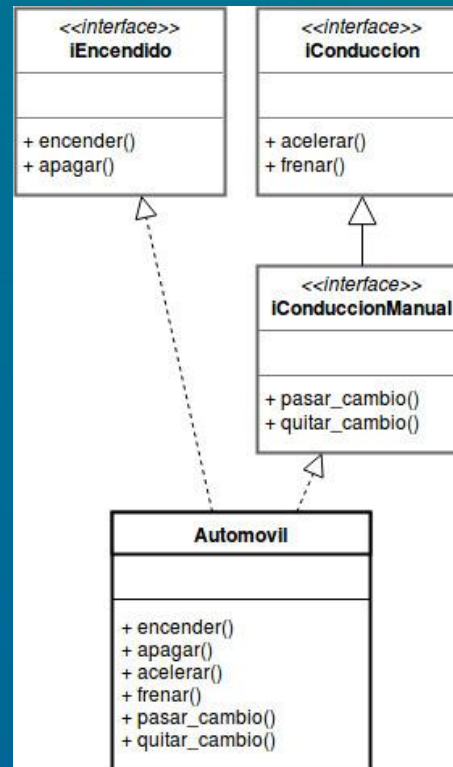
un contrato que se debe cumplir

interfaz

representada en UML



herencia *entre interfaces*



realización

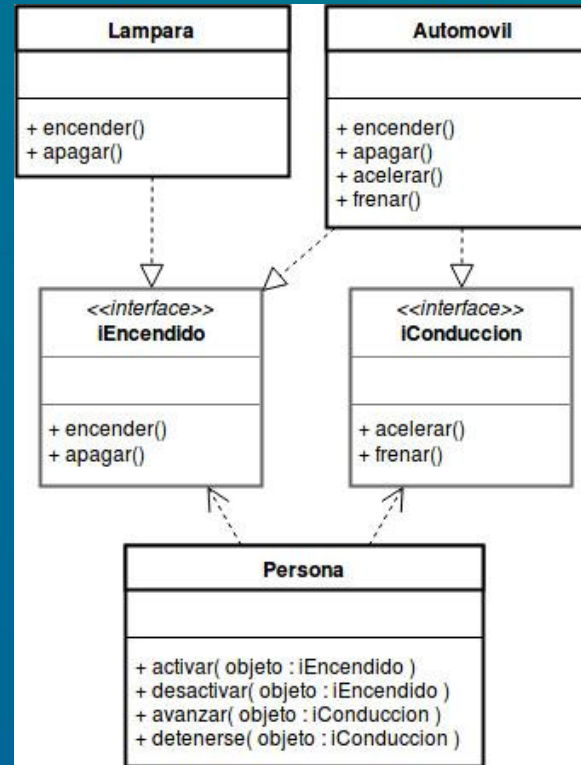
Relación semántica entre clasificadores
en la cual uno especifica un contrato
que el otro se compromete a cumplir.

el que cumple realiza al que especifica

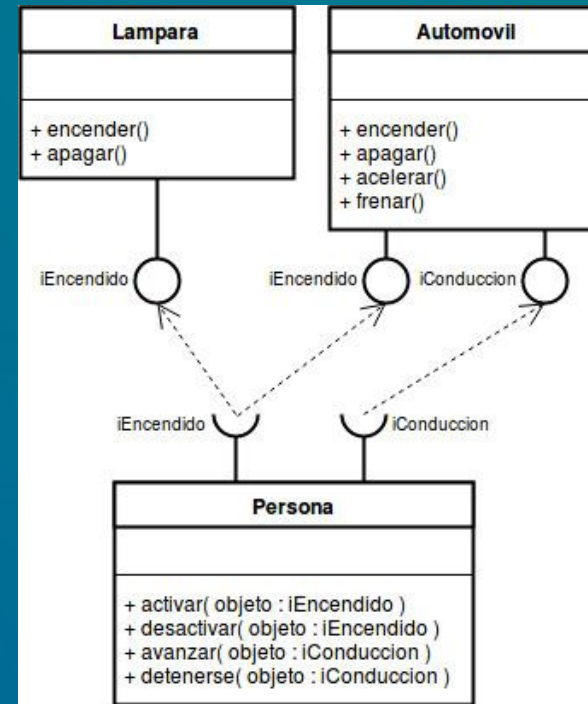
realización *representada en UML*



realización *entre clases e interfaces*



dependencia *entre interfaces*



encapsulamiento

Acto o método de ocultar los
detalles de implantación de un objeto
respecto de su usuario.

para que pueda usarlo sin necesidad de entenderlo

visibilidad

determina desde qué contextos se puede acceder a un elemento

	pública ¹ (public)	protegida (protected)	privada (private)
Desde el clasificador que declara el elemento	✓	✓	✓
Desde un subtipo del clasificador que declara el elemento	✓	✓	✗
Desde una instancia del clasificador que declara el elemento, o una instancia de uno de sus subtipos	✓	✗	✗

1- si no se explicita, la visibilidad predeterminada es pública

visibilidad *en atributos y operaciones*

Ejemplo
+ atributo_publico # atributo_protegido - atributo_privado
+ operacion_publica() # operacion_protegida() - operacion_privada()

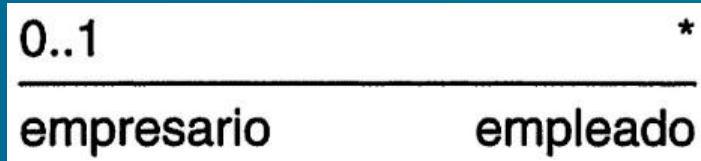
asociación

Relación semántica
entre dos clasificadores
que implica la conexión
entre sus instancias.

*describe un conjunto de enlaces (que son sus instancias)
y tiene multiplicidad*

asociación

representada en UML



asociación *entre clases*



agregación

Forma especial de asociación
que especifica una relación todo-parte
entre el agregado ¹ y un componente ².

1- el todo

2- la parte

agregación *representada en UML*



agregación *entre clases*



composición

Forma especial de agregación
con fuerte pertenencia y
un tiempo de vida coincidente¹
entre las partes y el todo².

*1- las partes pueden crearse después del todo,
pero obligatoriamente mueren con él*

2- compuesto

composición

representada en UML



composición *entre clases*



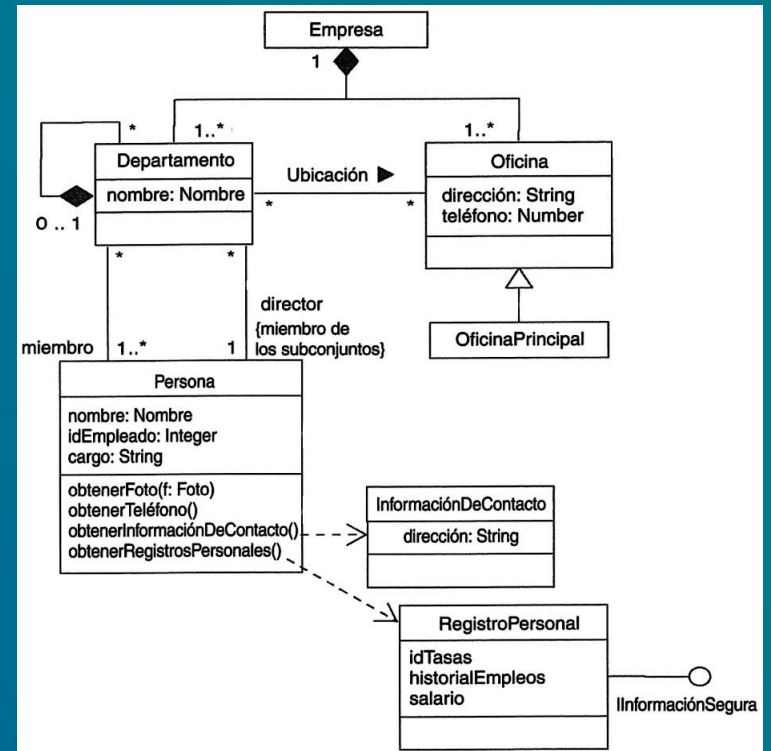
diagrama de clases

diagrama de clases

Muestra un conjunto de clases,
interfaces y colaboraciones,
y las relaciones entre ellas.

cubre la vista de diseño estática de un sistema

diagrama de clases

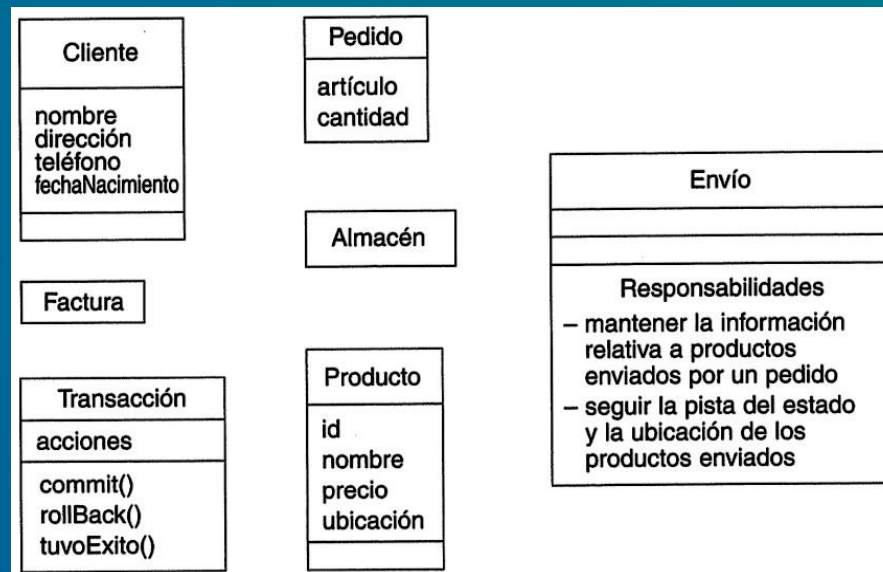


modelado del dominio de un sistema

uso común del diagrama de clases

- identificar las cosas que los usuarios o programadores utilizan para describir el problema o la solución
cada una de ellas será una abstracción
- identificar las responsabilidades de cada abstracción
definiéndolas claramente y repartiéndolas de forma pareja
- proporcionar a cada clase los atributos y operaciones necesarios para cumplir esas responsabilidades

modelado del dominio de un sistema



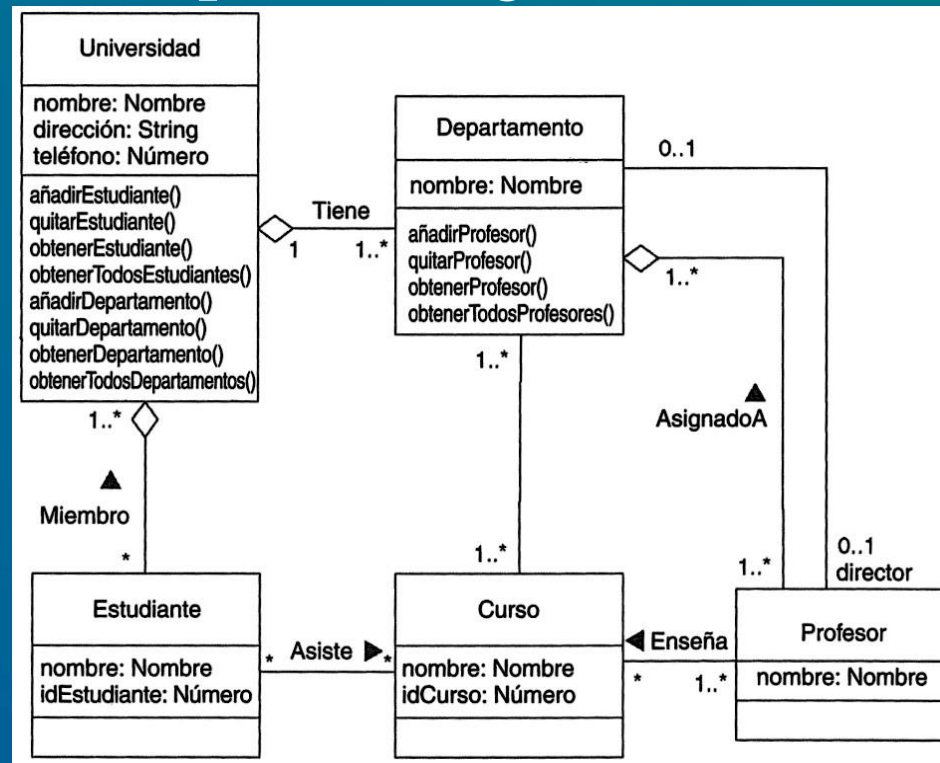
modelado de esquema lógico de BD

uso común del diagrama de clases

identificar las clases del modelo que se deben persistir

- crear un diagrama de clases que las contenga
- expandir sus detalles estructurales
especificando atributos, asociaciones, cardinalidad
- buscar patrones comunes
que compliquen el diseño físico de BD
- crear abstracciones intermedias
que simplifiquen la estructura lógica
- expandir las operaciones importantes
para el acceso y la integridad de los datos

modelado de esquema lógico de BD



bibliografía

el lenguaje unificado de modelado: guía del usuario

- Cap. 4: Clases
- Cap. 5: Relaciones
- Cap. 7: Diagramas
- Cap. 8: Diagramas de clases
- Cap. 9: Características avanzadas de las clases
- Cap. 10: Características avanzadas de las relaciones
- Cap. 13: Instancias
- Ap. A: Notación UML