



# Modelamiento de Clases

## Clase 03

Juan Luis Herencia Guerra

[juanluis.herencia@gmail.com](mailto:juanluis.herencia@gmail.com)

# Modelamiento de Paquetes

## Paquete

Sirve para agrupar y organizar elementos como Casos de Uso y Clases.

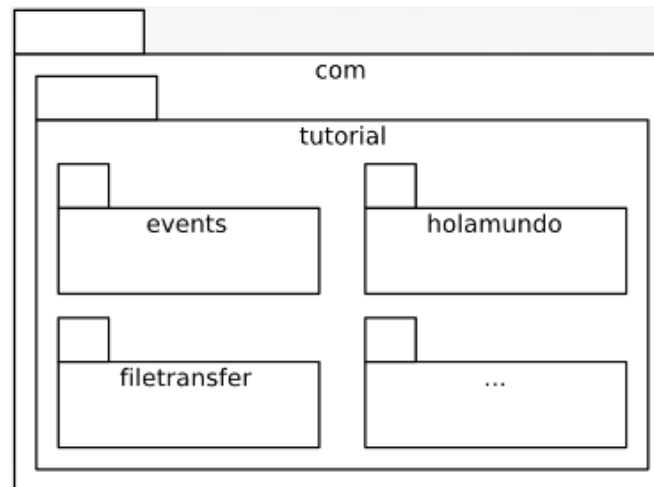
Dos elementos pueden tener el mismo nombre pero al estar en paquetes diferentes estarán listos para su uso.



# Modelamiento de Paquetes

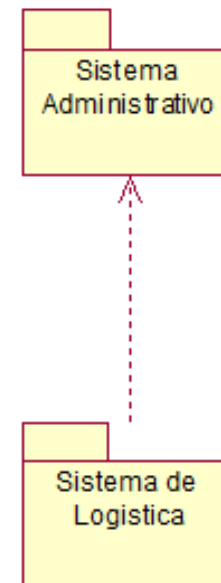
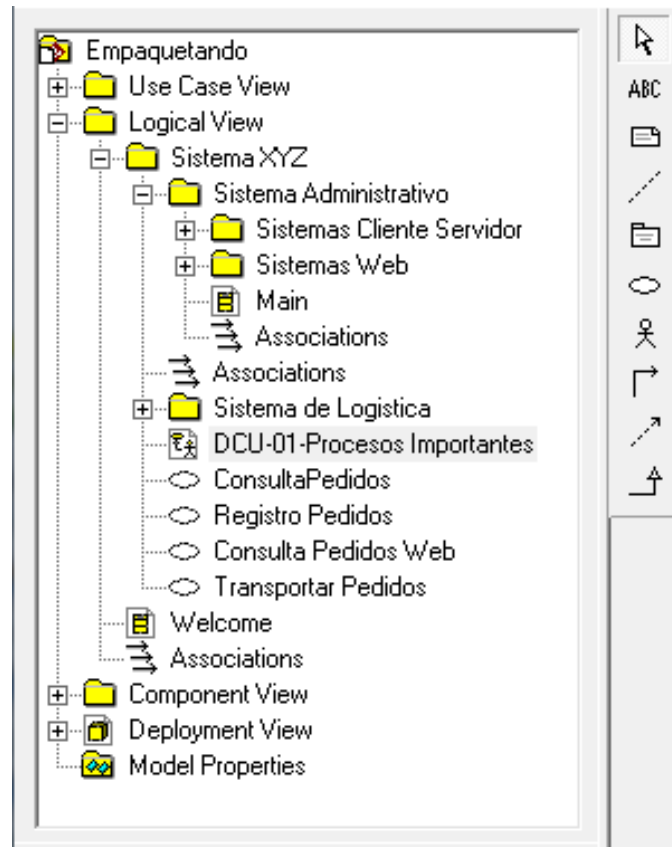
## Paquete

Los paquetes pueden estar anidados unos dentro de otros, y unos paquetes pueden depender de otros. Se puede utilizar paquetes para plantear la arquitectura del sistema a nivel macro.



# Modelamiento de Paquetes

## Paquete



# Modelamiento de Clases

## **Clase**

Es una abstracción, una idea general de algo que existe o nos rodea.

# Modelamiento de Clases

## Clase

La clase es la unidad básica que encapsula toda la información de un Objeto.

Objeto es una manifestación concreta de la abstracción de una clase (un objeto es una instancia de una clase).

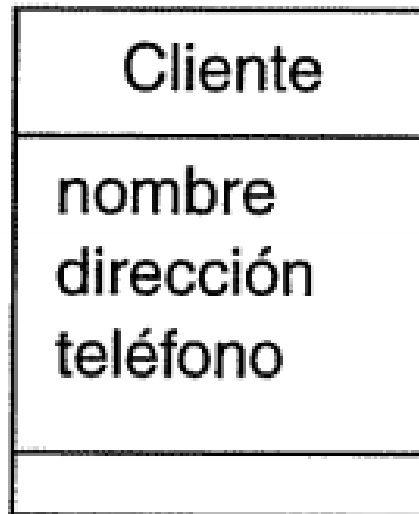
# Modelamiento de Clases

## Clase

Instancia es la especificación de un ente, un cuerpo, una idea, un tema, una cosa, una pieza o un **objeto** que ha proveniendo de la abstracción de una clase.

# Modelamiento de Clases

## Clase



Juan: Cliente

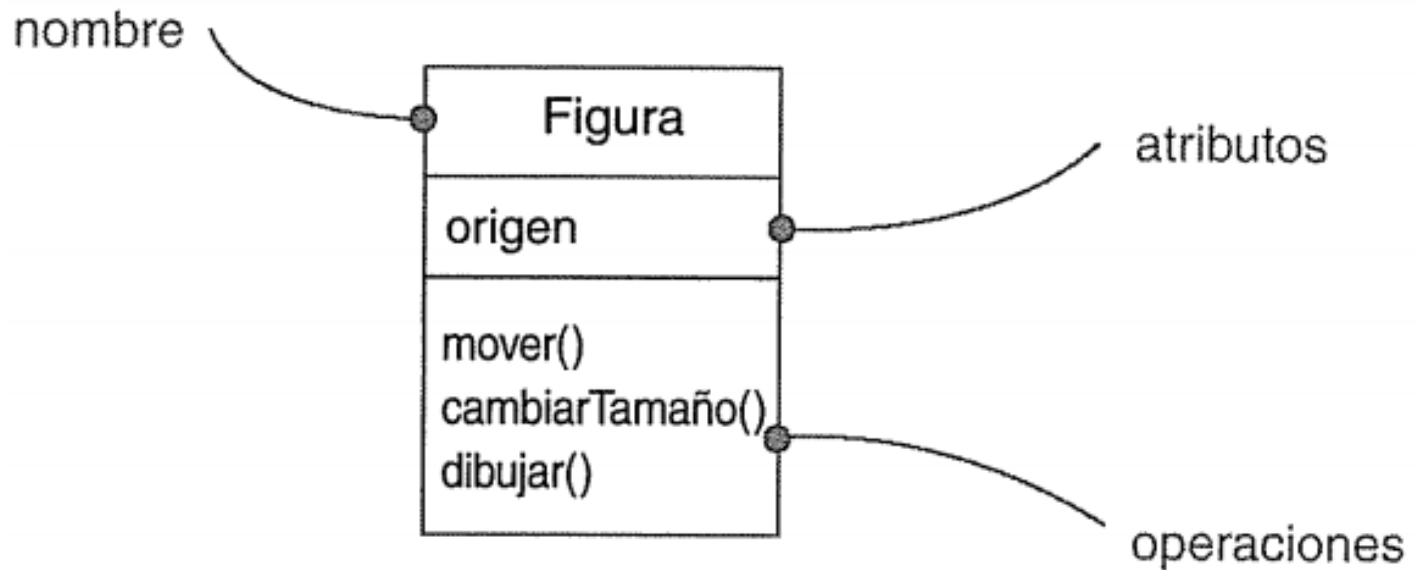
: Cliente

Elisa



# Modelamiento de Clases

## Clase



# Modelamiento de Clases




## Clase

- **Superior:** Contiene el nombre de la Clase
- **Intermedio:** Contiene los atributos (o variables de instancia) que caracterizan a la Clase.
- **Inferior:** Contiene los métodos u operaciones, los cuales son la forma como interactúa el objeto con su entorno.

# Modelamiento de Clases

## Clase

A su vez cada atributo u operación puede tener la visibilidad de público, privado y protegido:

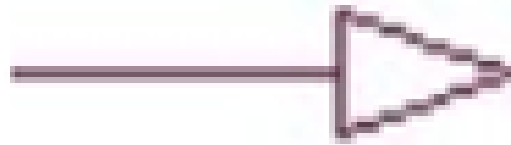
- **Público (public  + )**: Cualquier clase externa con visibilidad hacia la clase dada puede usar su atributo u operación.
- **Protegido (protected  #)**: Cualquier descendiente de la clase puede utilizar su atributo u operación.
- **Privado (private  -)**: Solo la propia clase puede usar el atributo u operación así definido.

# Modelamiento de Clases

## Clase-Relaciones

### Generalización / Especialización

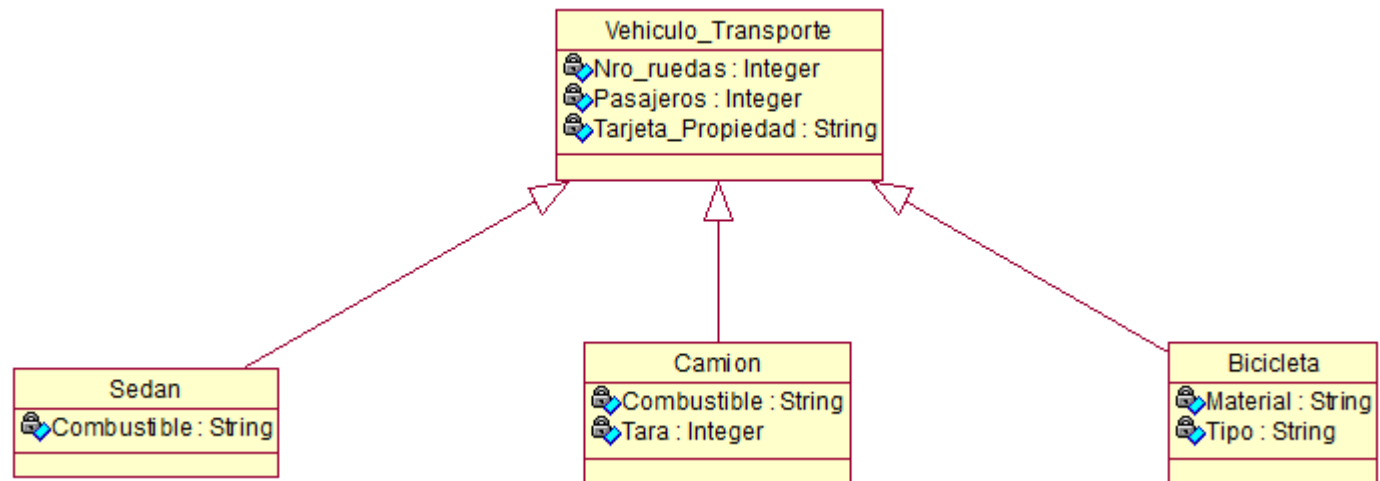
Es la relación entre una clase general (llamada súper-clase o padre) y un tipo más específico de ese elemento (llamado subclase o hijo)



# Modelamiento de Clases

## Clase-Relaciones

### Generalización / Especialización



# Modelamiento de Clases

## Clase-Relaciones

### Asociación

Es una relación estructural que especifica que los objetos pueden conectarse entre ellos como lo requieran.



# Modelamiento de Clases

## Clase-Relaciones

### Asociación



# Modelamiento de Clases

## Clase-Relaciones

### Asociación / Composición

Es cuando se relacionan las componentes de una clase como 'partes' a una clase que hacen el 'todo'. El objeto base se construye a partir de los objetos incluidos.

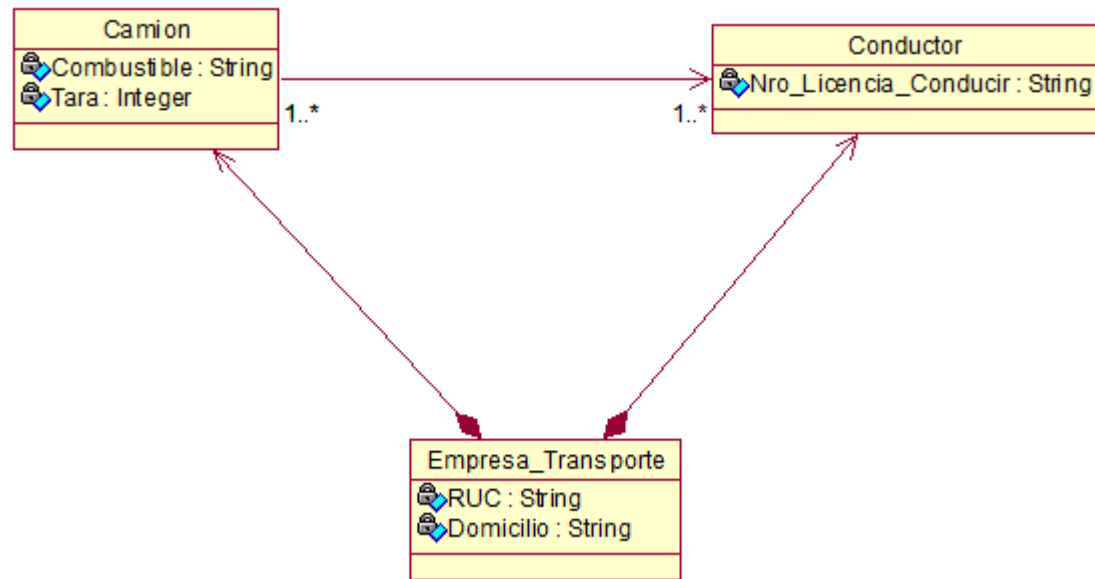




# Modelamiento de Clases

## Clase-Relaciones

### Asociación / Composición

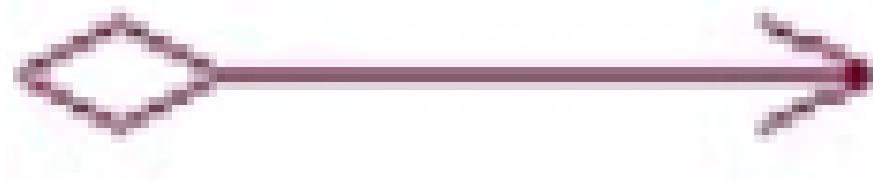


# Modelamiento de Clases

## Clase-Relaciones

### Asociación / Agregación

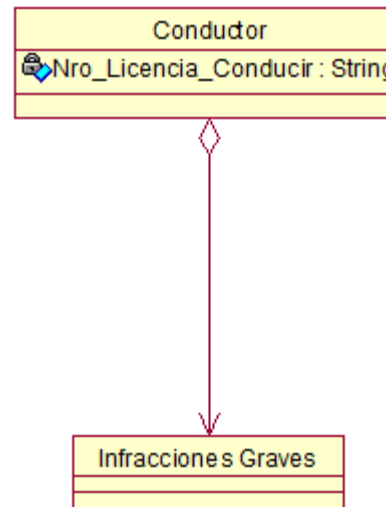
Es una asociación entre clases muy independientes pero una es parte de otra, también se conoce como composición débil.



# Modelamiento de Clases

## Clase-Relaciones

### Asociación / Agregación



# Modelamiento de Clases

## Clase-Relaciones

**Dependencia o Instanciación (uso):** ----->

Representa un tipo de relación muy particular, en la que una clase es instanciada (su instanciación es dependiente de otro objeto/clase). Se denota por una flecha punteada.

El uso más particular de este tipo de relación es para denotar la dependencia que tiene una clase de otra, como por ejemplo una aplicación grafica que instancia una ventana (la creación del Objeto Ventana esta condicionado a la instanciación proveniente desde el objeto Aplicacion):

# Modelamiento de Clases

## Clase-Relaciones



El Conductor depende de su Licencia de Conducir

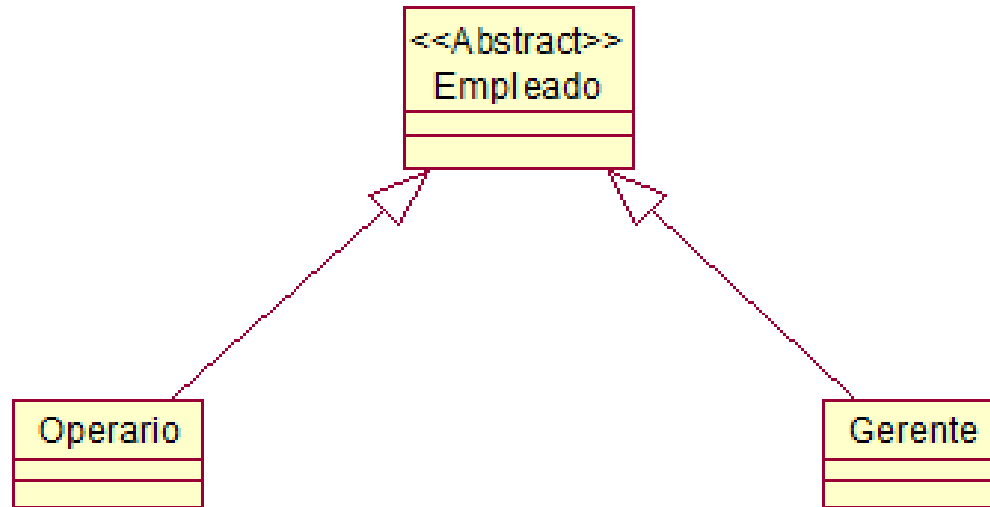
# Modelamiento de Clases

## Clases especiales-Clase Abstracta

Una clase abstracta se denota con el nombre de la clase y de los métodos con letra “*itálica*” (en Rational Rose). Esto indica que la clase definida no puede ser instanciada pues posee métodos abstractos (aún no han sido definidos, es decir, sin implementación). La única forma de utilizarla es definiendo subclases, que implementan los métodos abstractos definidos.

# Modelamiento de Clases

## Clases especiales-Clase Abstracta



# Modelamiento de Clases

## Clases especiales-Clase Interfaz

Es una clase completamente abstracta, sin ninguna implementación.

Su definición es permitir que se herede **solamente** los métodos u operaciones de la clase interfaz a que se extiende.

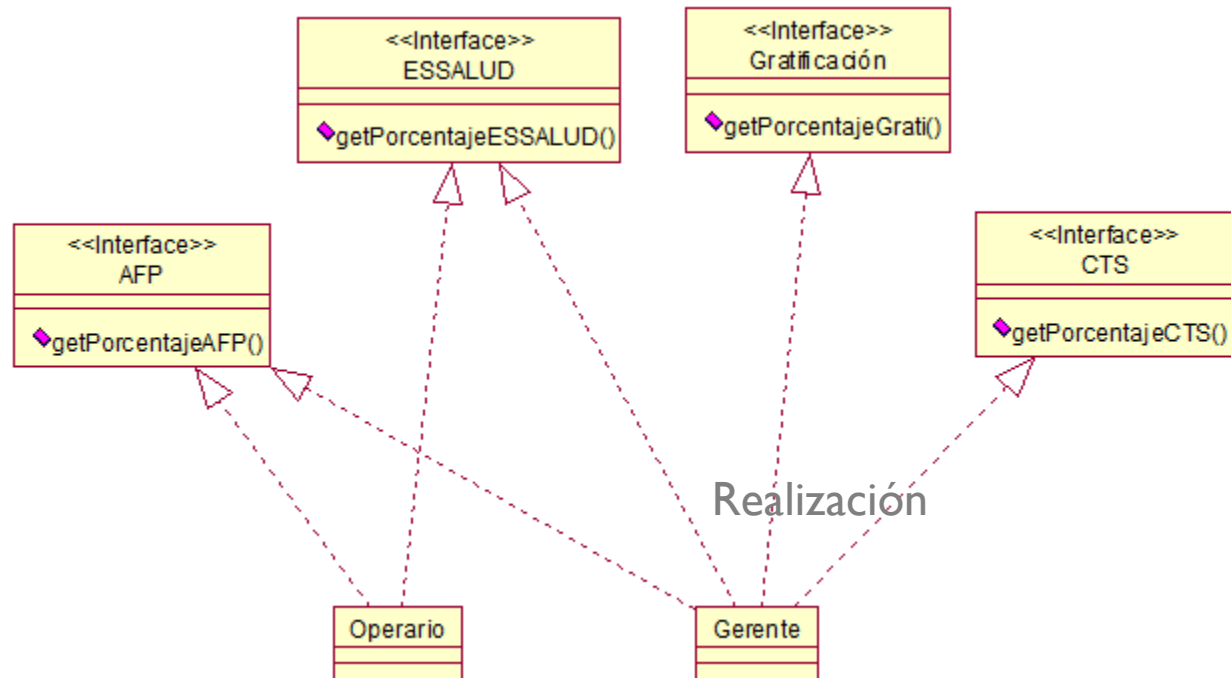
La definición de interfaces es para permitir extensión múltiple (herencia múltiple) de clases que agrupan funciones u operaciones importantes para la clase.

La extensión de estas clases se denomina **Realización**.



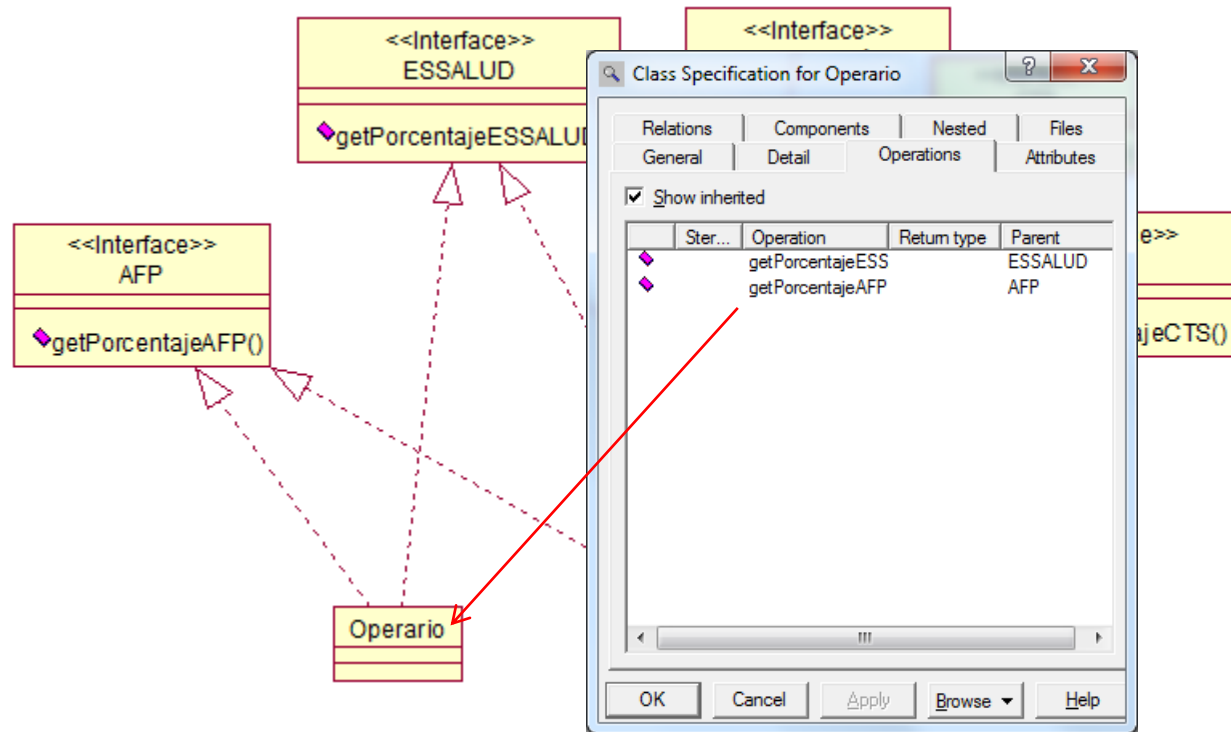
# Modelamiento de Clases

## Clases especiales-Clase Interfaz



# Modelamiento de Clases

## Clases especiales-Clase Interfaz



# Modelamiento de Clases

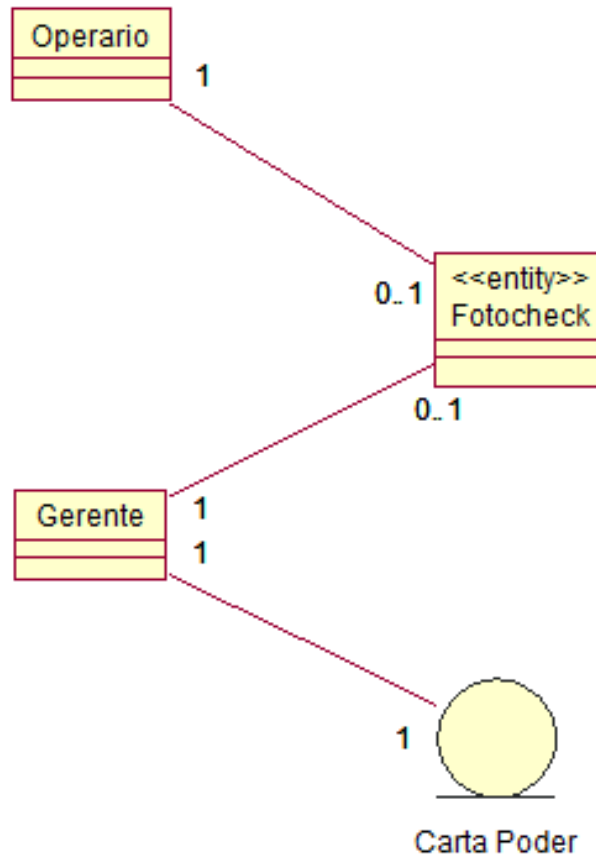
## Clases especiales-Clase Entidad

La **clase entidad** son un conjunto homogéneo de entidades comunes o tangibles que son muy importantes y muy usadas en el sistema. Por ejemplo en un sistema de Infracciones y Sanciones, las clases entidades serían las Papeletas, Licencias de Conducir y las Resoluciones de Sanción; las entidades serían las copias exactas de estos documentos.

# Modelamiento de Clases

## Clases especiales-Clase Entidad

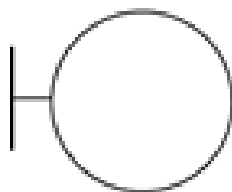
La clase entidad



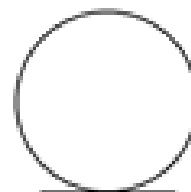
# Modelamiento de Clases

## Clases especiales-Clase Control y Frontera

La **clase Control** y **clase Frontera** son un conjunto homogéneo de entidades comunes que se refieren mayormente a las clases del sistema como controller (control) y web services (Frontera).



Boundary  
object



Entity  
object



Control  
object

# Modelamiento de Clases

## Clases especiales-Clase Control y Frontera

