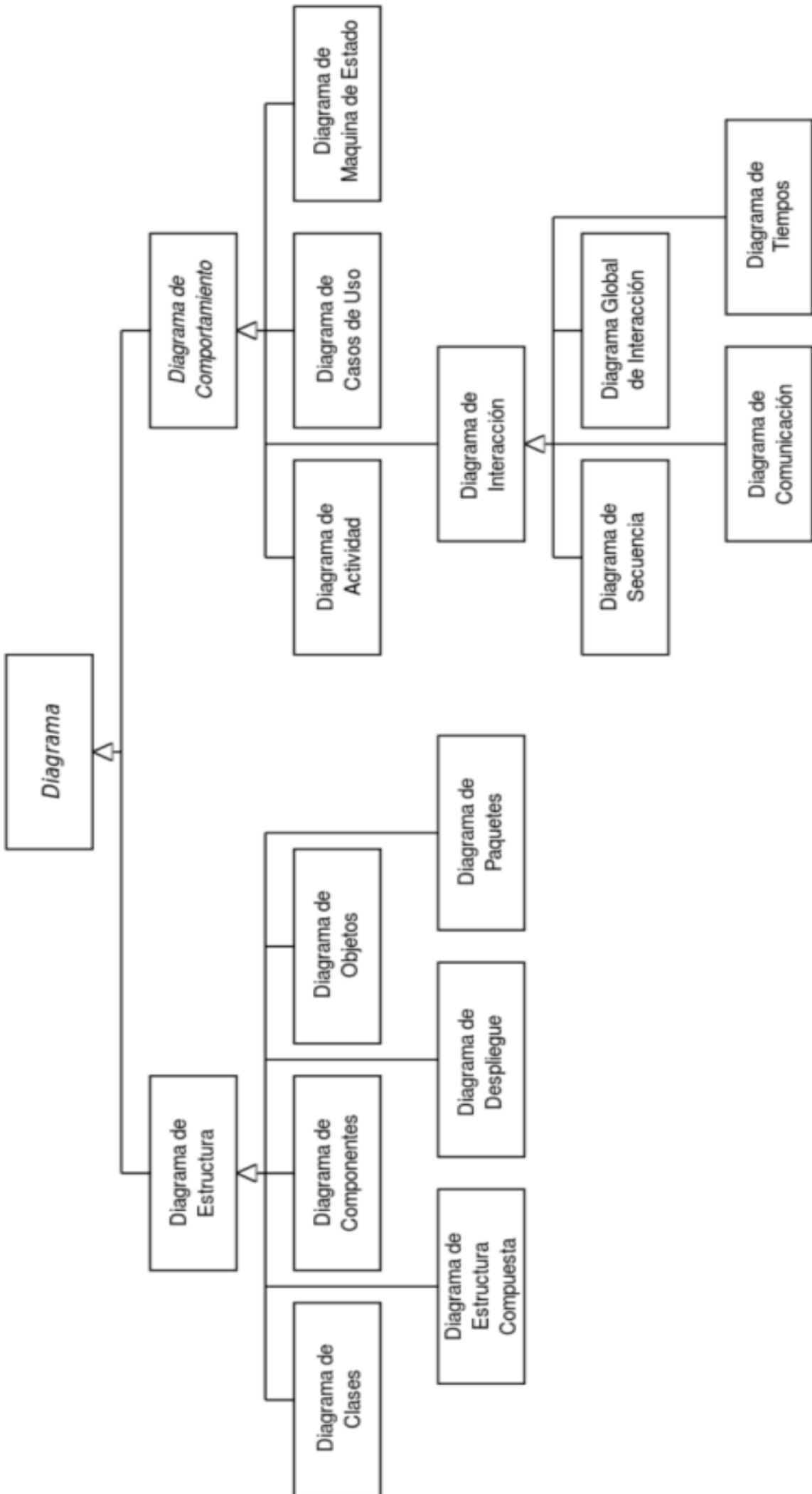


## 4. Ingeniería del Software/Modelos



## 4.1. Ingeniería del Software / Identificar Clases y Atributos

### Identificar Clases y Atributos

#### Identificar clases

Generalmente, los conceptos de interés corresponden con los sustantivos en la descripción del problema.

- Al inicio no ser demasiado selectivo en la elección inicial de clases. Se eliminarán posteriormente las clases incorrectas.
- Identificación de sustantivos en la descripción -> clases
- Listado de conceptos típicos que acompañan al sustantivo -> atributos

#### Identificación de Conceptos

Los conceptos tendrán relevancia (o no) en función del punto de vista de la aplicación.

Ejemplo: ¿Incluir el recibo de sueldo en una liquidación de haberes de empleados?

- No: La información que refleja tiene su origen en otros elementos. No sirve para nada, sólo informa al usuario.
- Si: El recibo tiene entidad propia pues da derecho al usuario a reclamar y acreditar la realización de una operación.

#### Eliminación de Clases

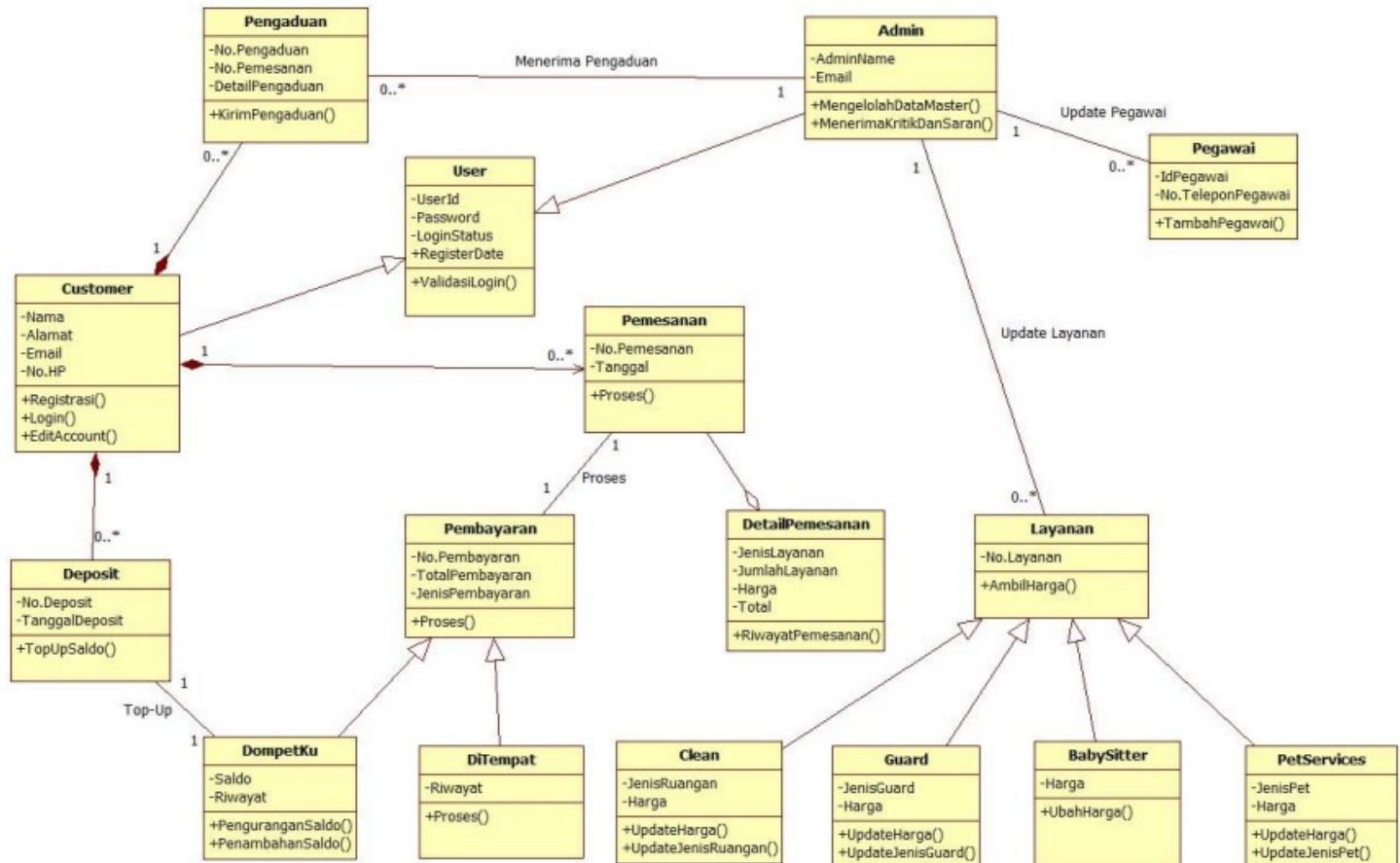
- Redundancia.
- Irrelevancia.
- Ambigüedad.
- Atributos, Operaciones y Roles.
- Estructuras de implementación.

## 4.1.1. Ingeniería del Software / Diagramas de Estructura

### Diagramas de Estructura

#### Diagrama de estructura

- El diagrama de clases es la construcción principal de cualquier solución orientada a objetos.
- Muestra las clases en un sistema, atributos y operaciones de cada clase y la relación entre cada clase.
- En la mayoría de las herramientas de modelado, una clase tiene tres partes, nombre en la parte superior, atributos en el centro y operaciones o métodos en la parte inferior.
- En sistemas grandes con muchas clases relacionadas, las clases se agrupan para crear diagramas de clases. Las diferentes relaciones entre las clases se muestran por diferentes tipos de flechas.

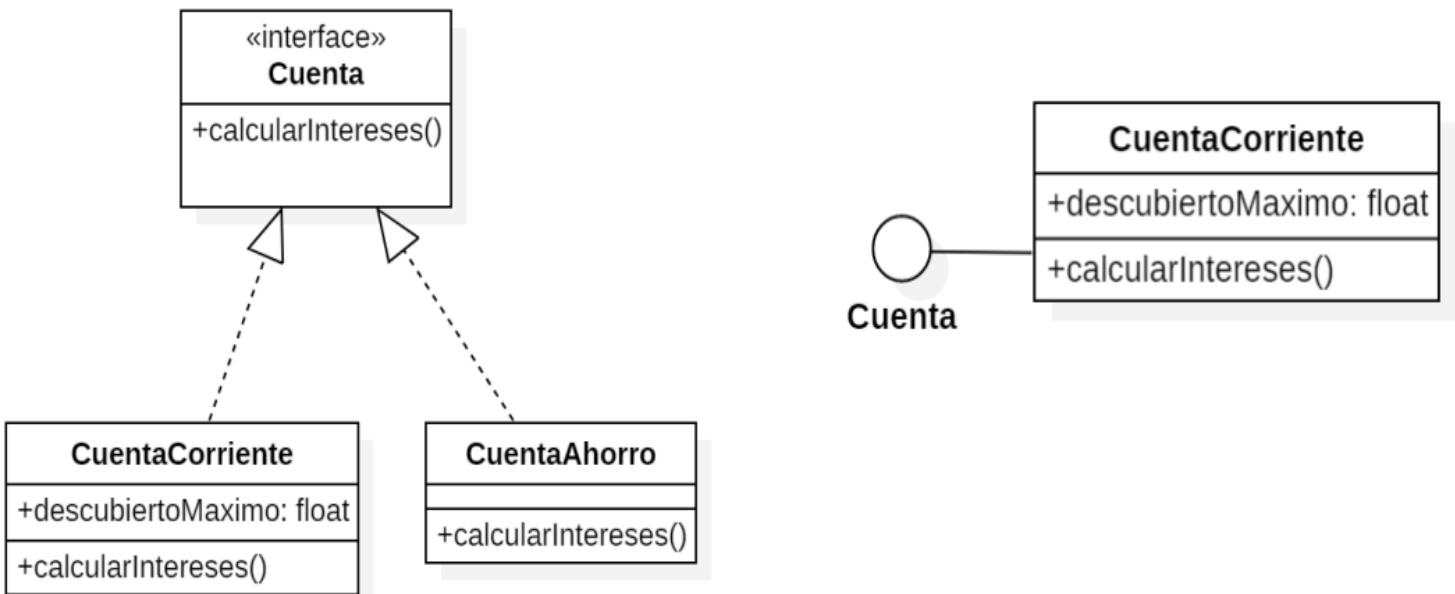


## 4.1.2. Ingeniería del Software / Diagramas de Estructura / Interfaz

### Interfaz

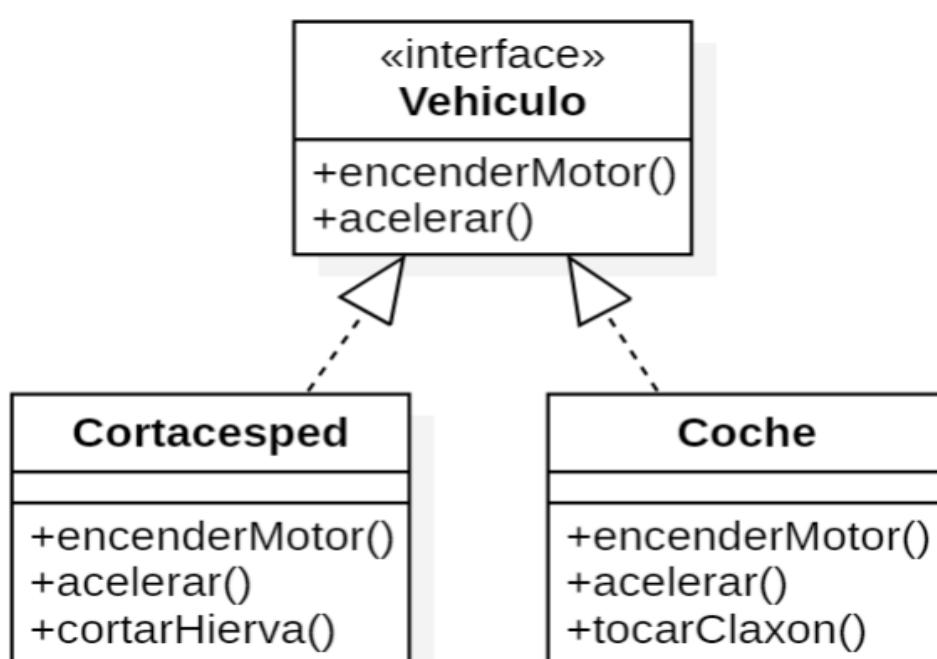
- Una interfaz es una colección de operaciones que especifica los servicios de una clase o componente
- Esas operaciones especifican un contrato que otro clasificador garantiza que cumplirá.
- Conjunto de operaciones que posee un nombre y que caracteriza el comportamiento de un elemento – No poseen implementación – Tienen visibilidad pública
- No contienen atributos
- Pueden tener relaciones de generalización.
- Una o más clases (o componentes) pueden realizar una interfaz
- Una clase puede admitir muchas interfaces

*Ejemplo:*



Clase abstracta, sin atributos ni métodos, con todas las operaciones abstractas

- Diferencia – Mientras que las clases que heredan de una clase abstracta sí están relacionadas, no existe ninguna relación entre diferentes clases que pueden implementar un interfaz.
  - Las implementaciones pueden ser totalmente distintas, las clases pueden no tener nada que ver entre sí, simplemente cumplen con el mismo interfaz



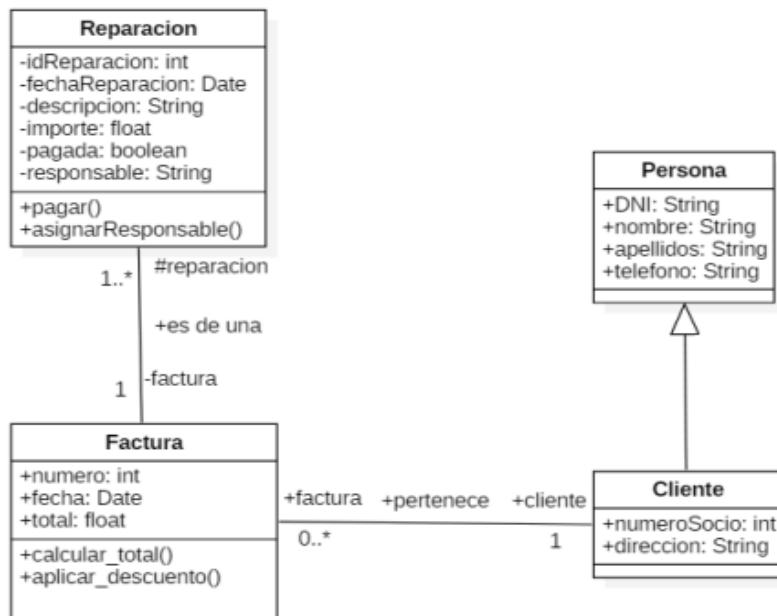
## 4.1.3. Ingeniería del Software / Diagramas de Estructura / Instancias

### Instancias

- Objeto = Instancia de una clase
- Entidad definida por un conjunto de atributos comunes y los servicios u operaciones asociados
- Tiene estado, exhibe algún comportamiento bien definido y posee una identidad única

|   |  |
|---|--|
| <ul style="list-style-type: none"> <li>■ Atributos</li> <li>■ Comportamiento</li> <li>■ Estado</li> <li>■ Identidad</li> <li>■ Responsabilidad</li> </ul> | <ul style="list-style-type: none"> <li>• Balance</li> <li>• Tipo de interés deudor</li> <li>• Interés acreedor</li> <li>• Haber</li> <li>• Debe</li> <li>• Informe</li> <li>• Abrir</li> <li>• Cerrar</li> <li>• Saldo actual ⓘ</li> <li>• Cuenta del Departamento de Informática y Automática</li> <li>• Almacena dinero del Departamento y le proporciona acceso al mismo mediante cheques</li> <li>• Proporciona un interés sobre el saldo siempre que éste se mantenga por encima de una cantidad</li> </ul> |
|---|--|

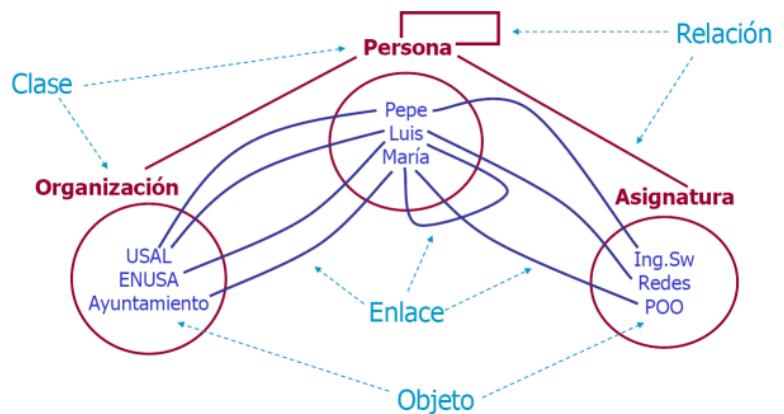
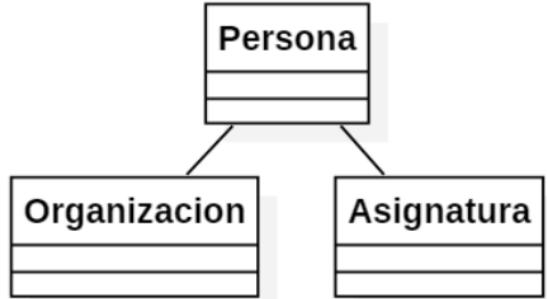
### Objeto Cuenta Bancaria



#### 4.1.4. Ingeniería del Software / Diagramas de Estructura / Relaciones

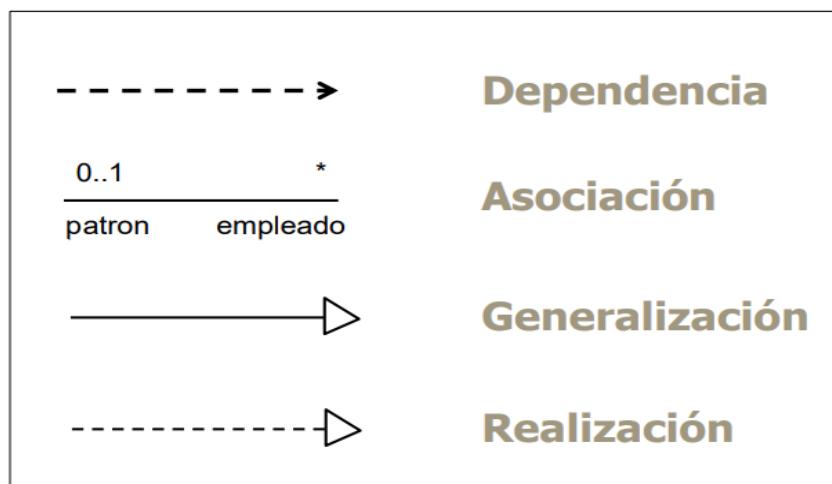
# Relaciones

- Un sistema no tiene componentes independientes
  - Los elementos están conectados y colaboran en el funcionamiento del sistema
  - En el dominio del problema (y de la solución) las clases no están aisladas, sino que se relacionan de diferentes formas – Una persona VIVE EN un apartamento – Una persona TRABAJA EN una empresa – Una rosa ES UN TIPO DE flor – Una rueda ES UNA PARTE DE un coche



Las relaciones expresan las conexiones entre clasificadores

- Tipos de relaciones:



# Referencias

Souto,C, 2010 . Introducción a las Aplicaciones Web