MODELOS DE DISEÑO ESTÁTICO

Introducción

Diagramas estáticos

Diagramas de clases y objetos

Como identificar elementos de los diagramas de clases

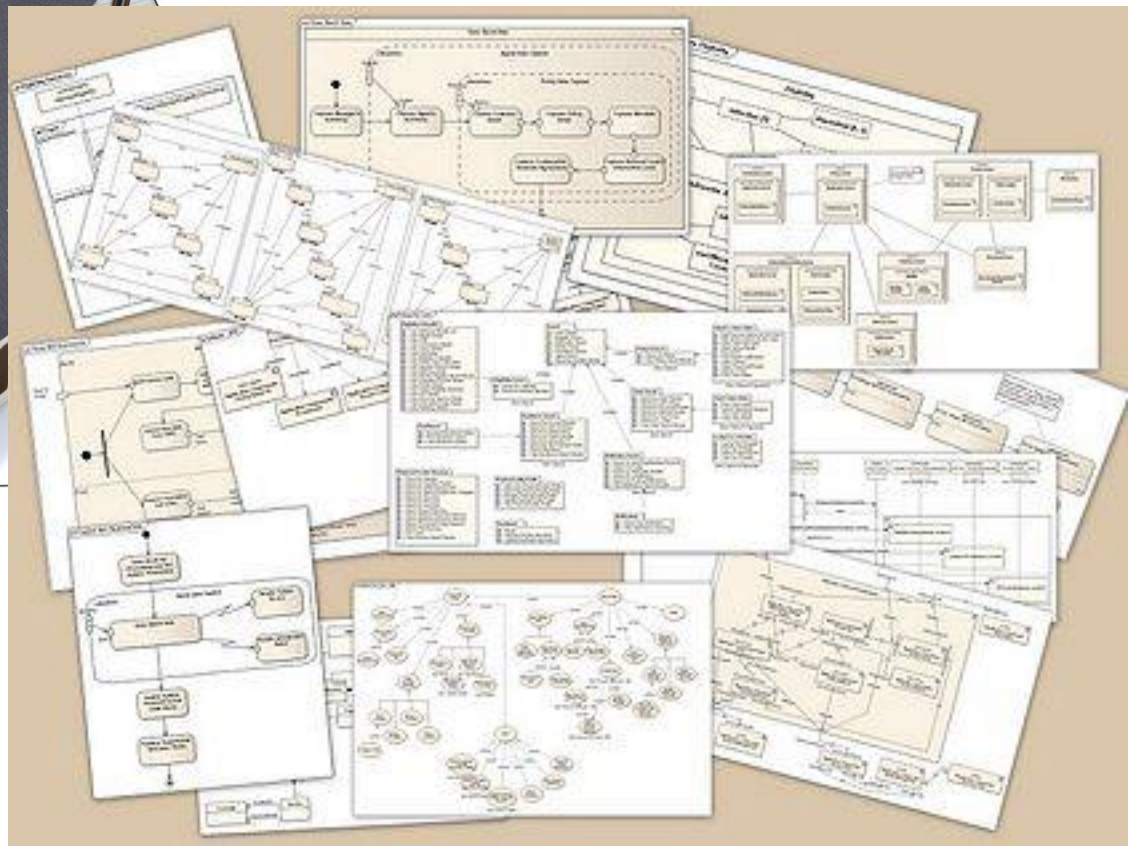
Introducción a la sintaxis UML para estos diagramas

- **Estereotipo**:
  - Un desarrollador de software no puede comunicarse correctamente con “gente normal” en “situaciones normales”
- El modelado puede ayudar a mejorar la comunicación
  - Entre desarrolladores sobre temas técnicos
  - Con los clientes, abstrayendo los términos técnicos
- Construir un sistema de software no es difícil si nos podemos comunicar con nuestros clientes, colegas, gerentes y con las herramientas.
- Cuando los problemas se vuelven más complejos, el riesgo que trae un error de comunicación es grande.

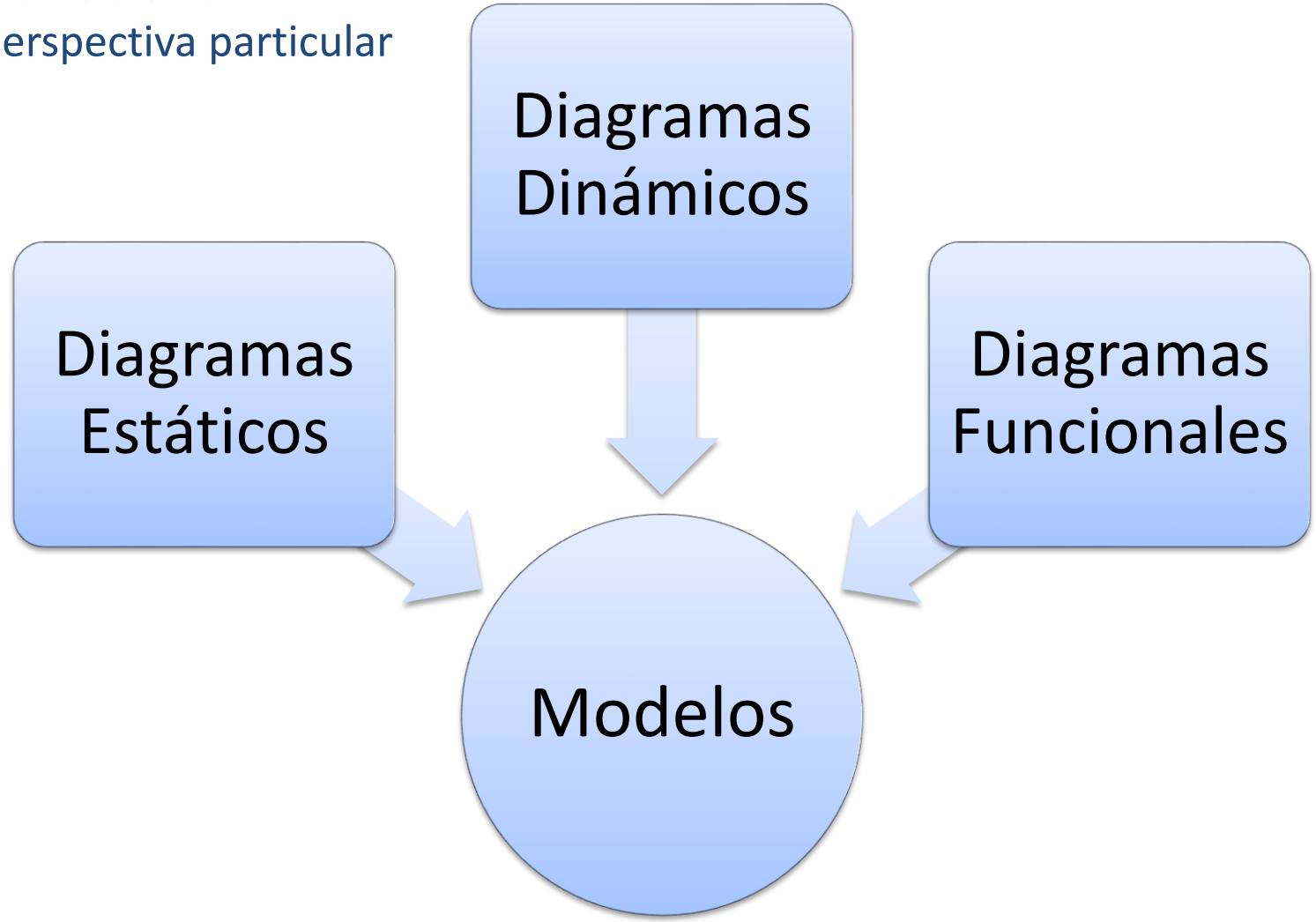
Diseño de una casa

Los diagramas son como los planos para un arquitecto o un ingeniero

Diseño de software



Un modelo es una descripción completa de un sistema desde una perspectiva particular



Diagramas estáticos

- Muestran las características estáticas del sistema.

Nombre	Muestran
de Clases	Entidades del mundo real, elementos de análisis y diseño, o clases de implementación y sus relaciones
de Objetos	Ejemplo específico de objetos y sus relaciones. Las condiciones para un evento o la llamada de una operación.
de Componentes	La organización y las relaciones entre los distintos entregables del sistema
de Estructura Compuesta	Como algo esta hecho. Útil para estructuras complejas o diseño basado en componentes
de Paquetes	La organización de los elementos del modelo y las dependencias entre ellos
de Despliegue	Arquitectura en tiempo de ejecución del sistema. Qué se ejecuta, como y donde.

Diagrama de Clases

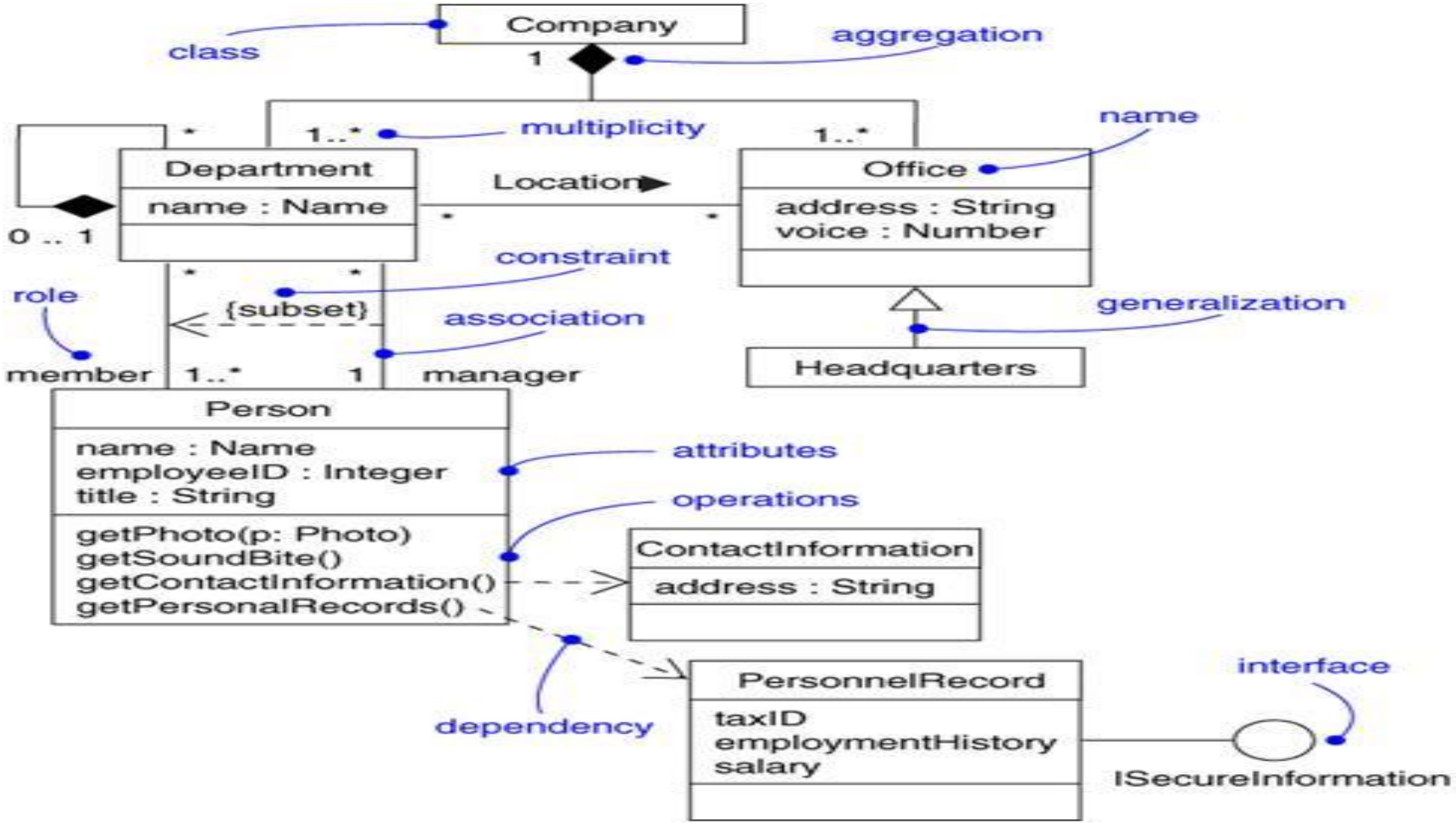
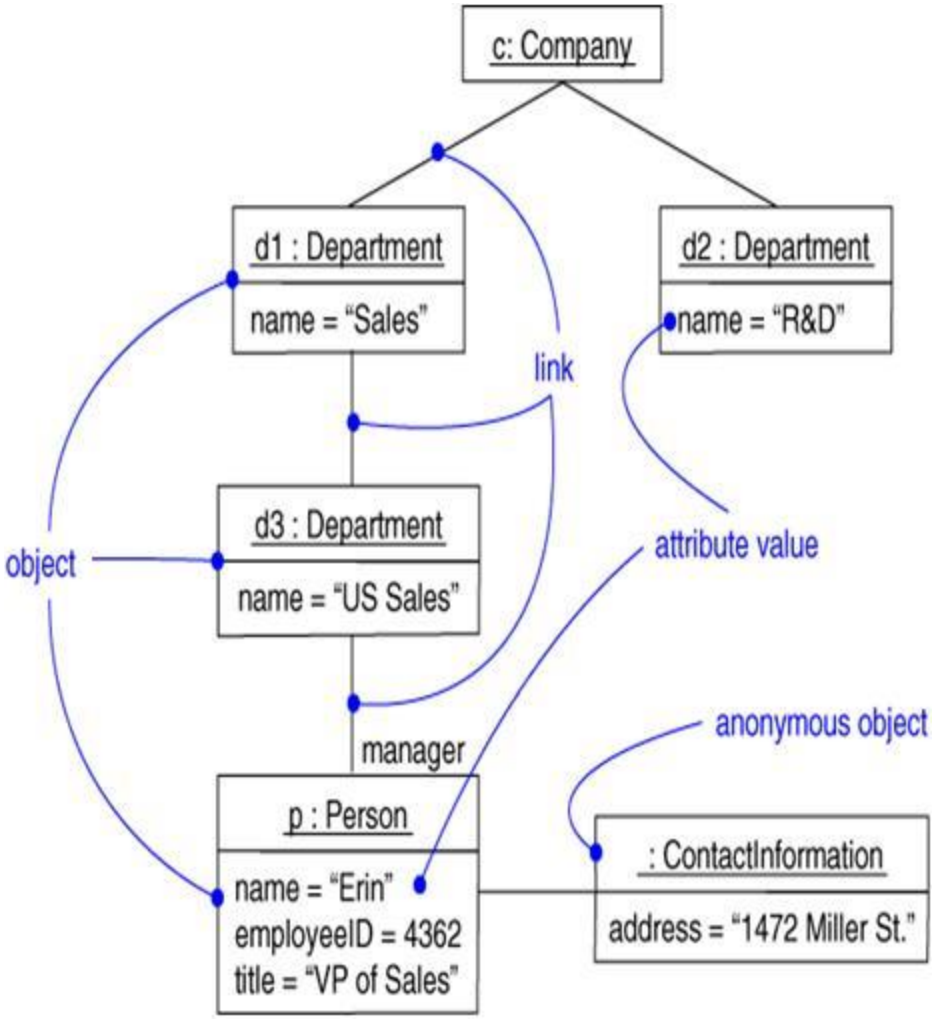


Diagrama de Objetos



Modelando clases y objetos

- Diagramas de clases:
  - Ampliamente usados en proyectos de software
  - Son los más útiles que se pueden producir
    - Las herramientas UML se basan en este diagrama para generar código
- Diagramas de objetos
  - Muestran instancias de clases (objetos) y links .
  - En un modelo complejo, se pueden tener muchas instancias y links aunque el diagrama de clases correspondiente puede ser simple.

Veremos como …

- identificar clases y objetos y darles nombres entendibles, útiles
- obtener los atributos que cada clase tiene
- determinar las operaciones que puede tener una clase y que partes deben ser privadas, públicas, etc
- es la sintaxis básica para modelar diagramas de clases y objetos

Como reconocer …

- **un objeto:** es cualquier entidad que tenga una identidad, estructura y comportamiento. Las entidades en el sistema son objetos de software que interactúan.

- **una clase:** es una familia de objetos que tienen una estructura, comportamiento y significado similares. Cada clase provee un esquema genérico para uno o más objetos.

- Paralelismo entre los objetos de un sistema de software y las entidades del negocio que se quiere modelar

Ejemplo - Introducción

- Construir un sistema para una aplicación bancaria (MiniBank) con un conjunto acotado de operaciones.
- Entidades del negocio traducidos a objetos de software. Ejemplo: cliente, una cuenta para ese cliente, operaciones asociadas a una cuenta.
- Objetos adicionales que forman parte del diseño y de la implementación que no forman parte del mundo real descartados al principio.
- Técnica para identificaar objetos y clases: “Subrayar los Sustantivos (y palabras que relacionan sustantivos)”

Ejemplo MiniBank (1 Paso)

- Crear una cuenta bancaria para un cliente
  - Una cuenta debe tener una identificación que se debe generar automáticamente, también debe tener el balance de la misma.
- Añadir una cantidad de dinero a una cuenta
- Retirar una cantidad de dinero de una cuenta
- Buscar todas las cuentas de un usuario
- Buscar las últimas operaciones o todas las operaciones que se han hecho sobre una cuenta entre dos fechas
  - Identificador de la operación, identificador de la cuenta, fecha, tipo (añadir/retirar), monto

Ejemplo (Técnica - 2 Paso)

- Considerar cada sustantivo en la descripción y ver si cumplen los siguientes criterios:
  - Es una cosa o familia de cosas
  - Es parte del problema a ser resuelto
  - No es parte de los detalles de la implementación
  - No es un evento o ocurrencia
  - No es una propiedad de algo
- Después de subrayar los sustantivos y palabras relacionadas en la descripción, tratar de determinar cuales pueden ser clases u objetos.

CUs Ejemplo (2 Paso)

- Crear una **cuenta bancaria** para un **cliente**
  - Una **cuenta** debe tener una **identificación** que se debe generar automáticamente, también debe tener el **balance** de la misma.
- Añadir una **cantidad de dinero** a una **cuenta**
- Retirar una **cantidad de dinero** de una **cuenta**
- Buscar todas las **cuentas de un usuario**
- Buscar todas las **operaciones** que se han hecho sobre una **cuenta** entre dos **fechas**
  - **Identificador de la operación**, **identificador de la cuenta, fecha, tipo** (añadir/retirar), **monto**

Identificación

Tipo de sustantivo	Ejemplo	Puede ser
Una familia de cosas	Cliente, Cuenta	Clase
Un sustantivo propio	Juán Perez	Objeto
Una propiedad de algo	Balance, tipo de cuenta	Atributo
Un valor o dato	$50, Caja de Ahorro	Valor de atributo
Una condición de algo	Pre-acuerdo, cliente nuevo	Estado
Una ocurrencia, evento o tiempo	Apertura de cuenta, extracción de dinero	Operación
Parte de la implementación	Base de datos, tabla	Descartar en esta etapa

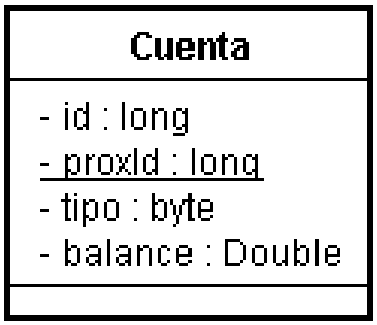
Nombrando clases

Un buen nombre de clase …	Ejemplo
Usar un sustantivo o “noun phrase”	últimas operaciones de mi cuenta
Es singular, no plural	última operación de mi cuenta
Evitar los adjetivos posesivos	última operación de una cuenta
No contener adjetivos irrelevantes	operación de una cuenta
Usar mayúscula al principio	Operación de una cuenta
No dejar espacios entre palabras	OperacionDeUnaCuenta
No contiene artículos, pronombres	OperacionCuenta (AccountOperation) Operacion

Nombrando objetos

- Generalmente, los objetos requieren nombres genéricos
- Se puede decir que los objetos son variables. En un momento son un objeto y en otro pueden ser otro objeto de la misma clase.
- Nombre de la clase precedido por un pronombre, adjetivo, o artículo. Ejemplos:
  - unaCuenta
  - cuentaPrincipal (mainAccount)
  - operacionPrincipal

Identificando atributos



- Para que algo sea una clase, las instancias (objetos) de esa clase deben tener propiedades/atributos que las describan como tales y las diferencien entre ellas.

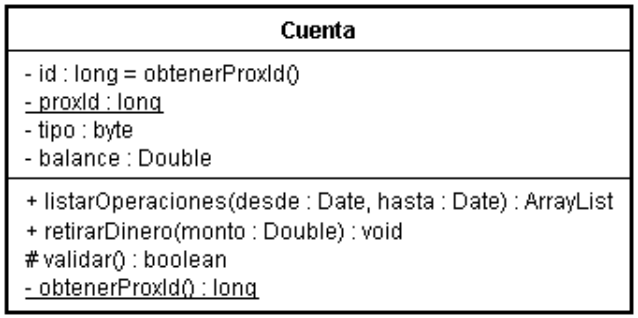
- Se le pueden especificar: tipos, valores por defecto, multiplicidad (dar la idea de muchos valores, por ejemplo varios nros. de teléfono del cliente)

Identificando operaciones

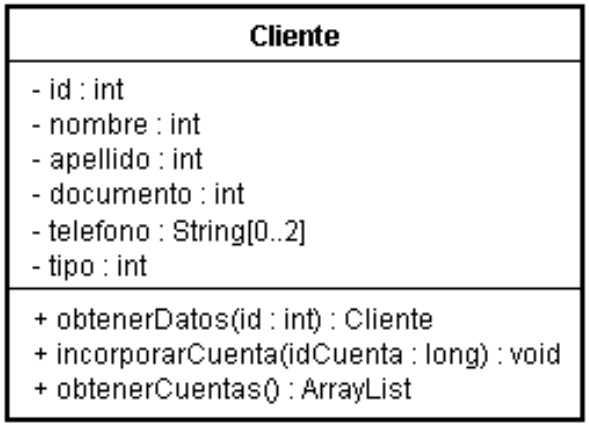
- Así como conocemos las propiedades de las clases, también debemos conocer el comportamiento de las mismas

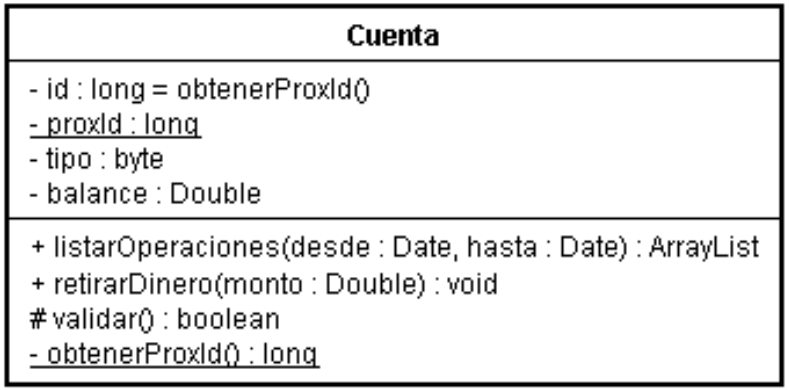
- Considerar que se le puede solicitar a un objeto de una clase que haga – o que puede causar un cambio de su estado.

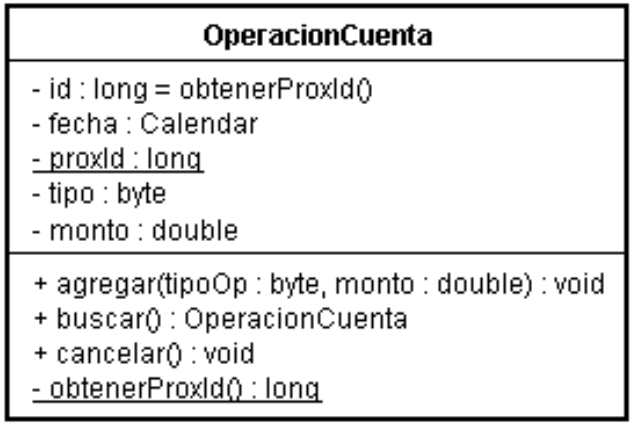
- Por ejemplo:
  - Listar operaciones en la clase Cuenta
  - Retirar dinero de una cuenta



Finalmente … ¿Es todo?



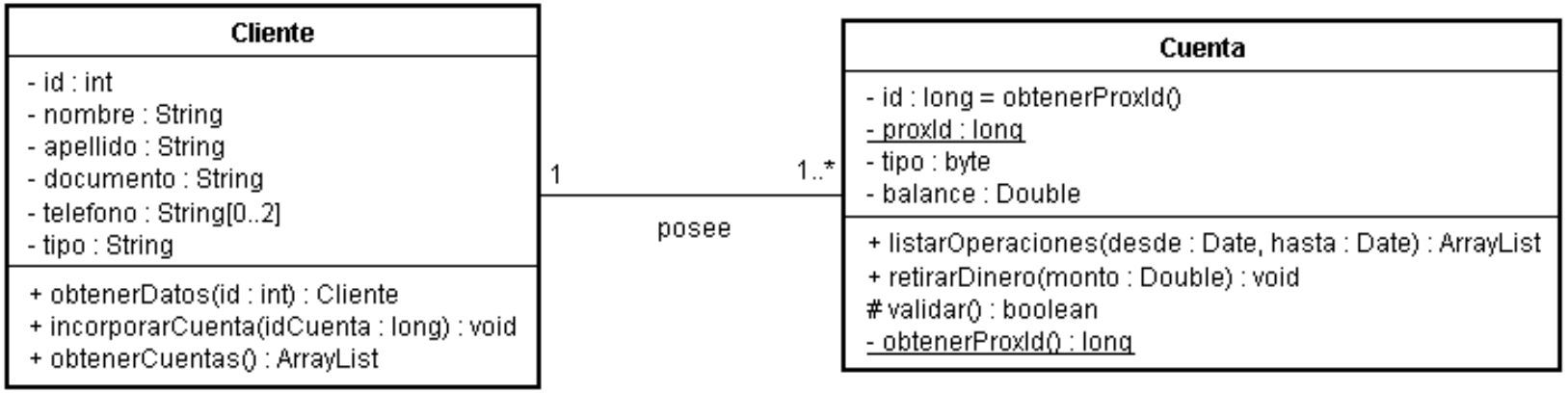




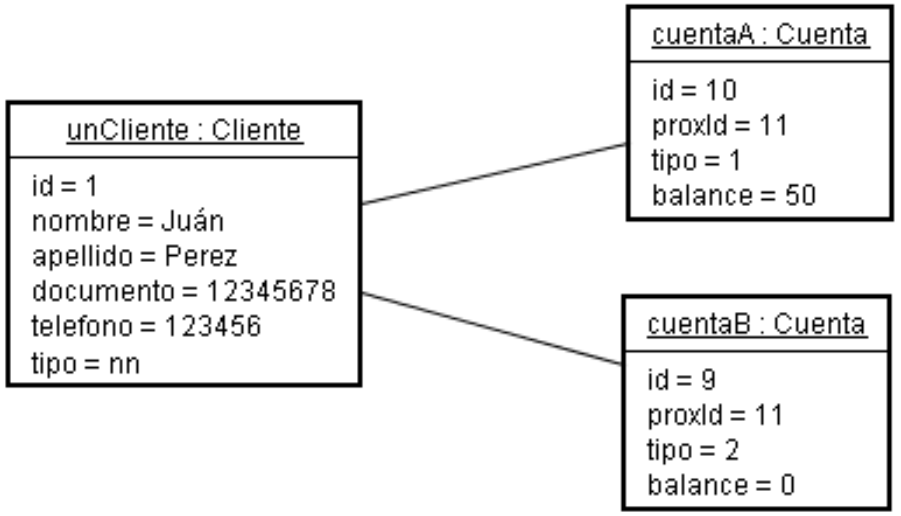
Relaciones estáticas

- Las clases son entidades que se relacionan entre si
- La mayoría interactúa con otras clases, no existen aisladas
- Estas relaciones se definen usando asociaciones en el caso de clases, y links para el caso de objetos
  - Objetos: instancias de clases
  - Links: instancias de asociaciones
- Notación:
  - Asociación: línea simple entre dos clases
  - Link: línea simple entre dos instancias de clases asociadas

Relaciones estáticas



Clases



Objetos

Relaciones estáticas

- Detalles a tener en cuenta
  - Nombre: describe la relación entre las clases.
  - Multiplicidad: especifica cuantas instancias de una clase pueden estar vinculadas a una instancia de la otra clase.
  - Roles: nombra la clase en el extremo de una asociación indicando como dicha clase participa en la asociación.
  - Restricciones: sobre una asociación si el vinculo debe seguir alguna/s regla/s.
  - Navegabilidad: usar flecha de navegabilidad en la asociación cuando una clase puede comunicarse con otra en ese sentido. La punta de flecha indica la dirección permitida de comunicación.