

## Capítulo 8

# Metodología OMT

### 8.1 El Modelo de Objetos

Captura la estructura estática del sistema, mostrando:

- ² objetos

- ² relaciones entre objetos

- ² atributos

- ² operaciones

- ² Es el más importante, puesto que el sistema se construye alrededor de los objetos.

- ² Conceptos, notación y ejemplos.

#### 8.1.1 Objetos y Clases

Concepto, abstracción o cosa con fronteras de...nidas y signi...cado para nuestro problema.

Permite una mejor comprensión del mundo y proporciona la base para una implementación sobre el ordenador.

- No existe una representación exacta.

- Todos los objetos tienen una identidad y son distinguibles.

#### 8.1.2 Clase:

Describe grupos de objetos con propiedades (atributos) similares, comportamiento (operaciones) comunes, relaciones con otros objetos y semántica común.

- Cada objeto sabe cuál es su clase, ya que es una instancia de la misma.

- Elemento esencial para la abstracción y generalización.

#### 8.1.3 Diagrama de Objetos

Notación grá...ca para modelar los objetos, clases y sus relaciones.

- Dos clase de diagrama:

<sup>2</sup> De clases

<sup>2</sup> De objetos (instancias)

### Diagrama de clases

Esquema, patrón o plantilla para describir muchos casos posibles de datos. Describe clases de objetos.

### Diagrama de objetos

Describe cómo se relacionan un grupo particular de objetos entre sí.

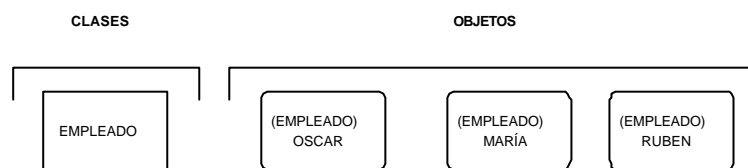
#### 8.1.4 Notación de clases y objetos Atributos

Valor de un dato dentro de un objeto. Cada atributo tiene un valor para cada objeto. El nombre de un atributo es único dentro de una clase.

Debería ser un dato puro, no un objeto (no tiene identidad). Si un objeto necesita otro objeto habrá que modelarlo como asociación.

Además del nombre podemos especificar el Tipo y el Valor por defecto.

Los identificadores de objetos explícitos no se necesitan en el Modelo de Objetos.



Figura~8.1: Notación de Clases y Objetos

#### 8.1.5 Notación de atributos Operaciones y Métodos

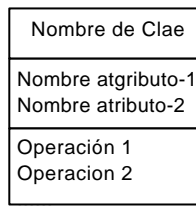
Función o transformación que se aplica a o por los objetos en una clase. Tienen un objeto destino como argumento implícito (el objeto actual).

**Polimorfismo:** la misma operación puede aplicarse a clases diferentes dentro de una jerarquía de herencia.

**Método:** implementación de una operación para una clase. Las operaciones con métodos en varias clases deben compartir la misma signatura (tipos de sus argumentos y valor que devuelven).

#### 8.1.6 Notación de operaciones y métodos

Se describe en el primer cuadrante el Nombre de la Clase, luego, en el segundo, los atributos de dicha clase, finalmente, en el último se detallan las operaciones que intervienen en la Clase. Como lo podemos apreciar en la figura siguiente.



Figura~8.2: Notación de Clase, Atributos y Operaciones

### 8.1.7 Enlaces y Asociaciones

#### Enlace

Conexión física o conceptual entre objetos.

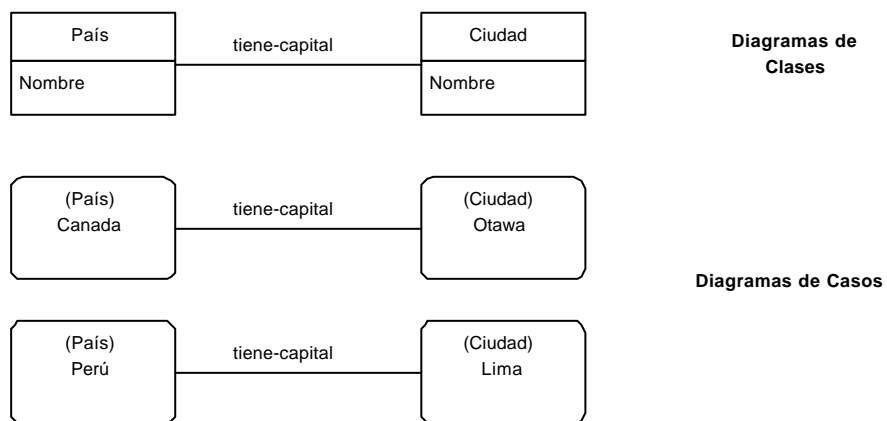
#### Asociación

Grupo de enlaces con la misma estructura y semántica común.

Sentido de una asociación:

- ² Inherentemente son bidireccionales.
- ² directa (forward) e inversa (inverse).
- ² Implementación común en los lenguajes de programación mediante punteros o referencias, aunque en la fase de modelización no es recomendable esta práctica.

Las asociaciones pueden ser binarias, ternarias o de órdenes superiores y los nombres de las asociaciones son opcionales en la notación (se escriben en cursiva).



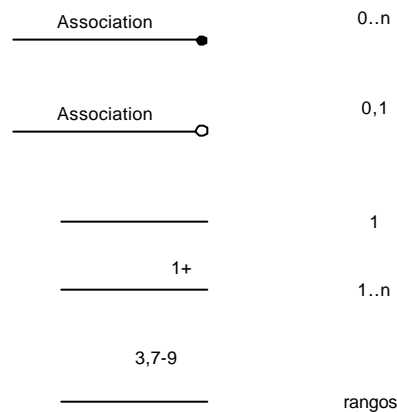
Figura~8.3: Diagrama de Clase y de Casos

### 8.1.8 Multiplicidad

La multiplicidad especifica cuántos objetos de una clase pueden relacionarse con un único objeto de una clase asociada.

Se especifica mediante:

- <sup>2</sup> Un subconjunto (posiblemente infinito) de enteros no negativos: 0..n.
- <sup>2</sup> Uno o varios intervalos no conexo: 1..4, 7.
- <sup>2</sup> No debemos preocuparnos de ajustar la multiplicidad demasiado pronto en nuestro Modelo de Objetos.
- <sup>2</sup> En los Diagrama de Objetos la multiplicidad se especifica mediante símbolos especiales en los extremos de las líneas de las asociaciones.



Figura~8.4: Multiplicidad en OMT

### 8.1.9 Atributos de los enlaces

Las propiedades de los enlaces de una asociación son:

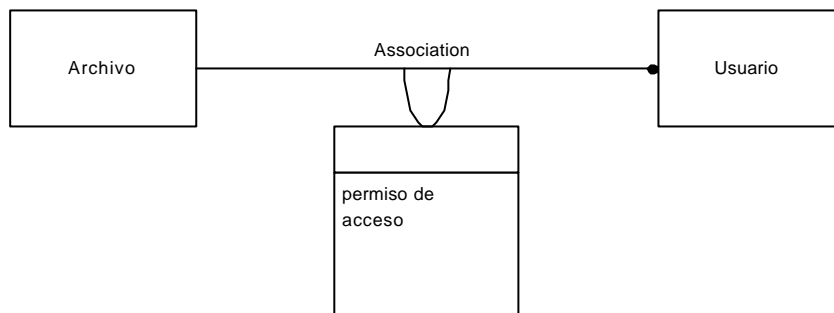
- <sup>2</sup> Opcionales en los enlaces uno-a-uno y uno-a-muchos.
- <sup>2</sup> Obligatorios en los enlaces muchos-a-muchos.

### 8.1.10 Modelando una Asociación como una Clase

Además de los atributos de un enlace, permite añadir un nombre y operaciones.

Son útiles cuando:

- <sup>2</sup> Los enlaces pueden participar en asociaciones con otros objetos.
- <sup>2</sup> Los enlaces están sujetos a operaciones.



Figura~8.5: Atributos en los enlaces

### 8.1.11 Roles

Se lo coloca en uno de los extremos de una asociación. El Nombre de Rol identifica de forma única el ...nal de una asociación. Atributo derivado cuyo valor es un conjunto de objetos relacionados. Deben ser únicos en las asociaciones de una clase.

Suelen aparecer como nombres en los enunciados de los problemas.

En relaciones n-arias se debe colocar un rol en cada extremo.



Figura~8.6: Nombres de Rol para una asociación

Los roles proporcionan una forma de ver las asociaciones como recorrido desde un objeto hacia otro conjunto de objetos. Su uso es opcional. A veces resulta más claro su uso que los nombres de las asociaciones. Normalmente son necesarios en las relaciones entre objetos de la misma clase, o cuando existe más de una asociación entre el mismo par de clases.

### 8.1.12 Ordenación

Se utiliza en la parte de la multiplicidad muchos de una asociación en la que los objetos deben estar ordenados.

### 8.1.13 Cualificadores

Su función específica es describir una asociación calificada, esta relaciona dos clases mediante un calificador. Puede considerarse como un tipo de asociación ternaria.

**Cuali...cador:** atributo especial que puede reducir la multiplicidad de una asociación. Pueden utilizarse en asociaciones uno a muchos o muchos a muchos. Distinguen conjuntos de objetos en el extremo muchos de la asociación.

Mejora la semántica y aumenta la visibilidad de caminos de navegación.

Se representa mediante un cuadro en el extremo más próximo a la clase a la que cuali...ca.

**Ejemplo 8.1** Directorio + nombre de archivo.

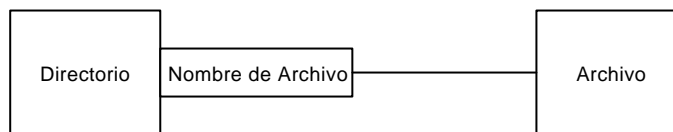


Figura ~8.7: Una asociación cuali...cada

Notación de cuali...cación

Un directorio contiene más de un archivo.

El nombre del directorio más el nombre del archivo en el directorio, identi...can de forma única al archivo:

Una compañía tiene muchos empleados.

El cuali...cador descompone el conjunto total de los empleados en subconjuntos disjuntos (por cargo) de más de un objeto.

#### 8.1.14 Agregación

Relación parte-todo o parte-de, el ensamblaje se relaciona con sus componentes. Veri...ca las propiedades transitiva y antisimétrica. También es común la agregación recursiva.

Cuando tenemos duda en la etapa de modelamiento, mejor es utilizar una asociación ordinaria.

Preguntas útiles para identi...car agregaciones:

¿Se puede utilizar la frase parte de?

¿Se propagan algunas operaciones del todo a las partes de forma automática?

¿Se propagan algunos valores de los atributos del todo a alguna o todas de las partes?

¿Existe una asimetría intrínseca en la asociación, por la cual un objeto está subordinado a otro (la eliminación del todo implica la de las partes)?

Notación de agregación

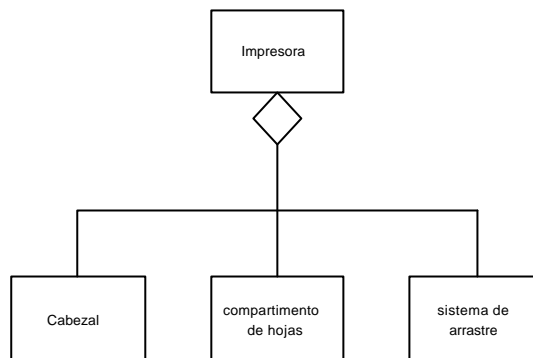
<sup>2</sup> Agregados ...jos: estructura ...ja, con el número y tipos de subpartes prede...nidos..

<sup>2</sup> Agregados variables: tienen un número ...nito de niveles, pero el número de partes varía.

<sup>2</sup> Agregados recursivos: contienen, directa o indirectamente, un caso de la misma clase.

Propagación de las operaciones

Aplicación automática de una operación a una red de objetos cuando se aplica la operación a un objeto inicial. La propagación de una operación a las partes es un buen indicador de agregación.



Figura~8.8: Agregación

### 8.1.15 Generalización y herencia

**Generalización:** relación entre una clase (superclase) y una o más versiones re...nadas de ella (subclases).

Relación se describe con la frase es un.

Las subclases heredan las características, atributos y operaciones de su super clase.

Una instancia de una subclase es una instancia de sus clases antecesoras o ascendientes.

**Distinción entre generalización y herencia:**

**Generación:** relación entre clases.

**Herencia:** mecanismo para compartir características.

**Ascendientes y descendientes:** generalización en múltiples niveles.

**Discriminador:** atributo de tipo enumerado, que indica la propiedad del objeto que se está abstrayendo para una relación de generalización. Solo debería discriminarse una propiedad a la vez.

**Ejemplo 8.2** Fichero. Contenido: de texto, binario, de datos; Acceso: secuencial, directo, indexado; Carácter: ejecutable, no ejecutable.

**Generalización como extensión y restricción**

Una instancia de una clase es una instancia de todas sus clases ascendientes y no puede omitir o suprimir un atributo de sus ascendientes.

Una subclase puede añadir nuevas características: extensión.

Una subclase puede rede...nir una característica de sus ascendientes: restricción.

**Ejemplo 8.3** Limitar los valores en un rango menor (círculo es una elipse de  $a=b$ ).

### 8.1.16 Agregación vs. Generalización.

La agregación relaciona instancias (relación y), la generalización relaciona clases (relación o).

### Redefinición

Una subclase reemplaza una característica de su superclase definiendo una característica con el mismo nombre.

### Motivos para redefinir

Especificar un comportamiento que dependa de la subclase.

Aplicar de forma más rigurosa las especificaciones de una característica. Mejorar el rendimiento.

No se debe reemplazar la signature de una característica.

Debe preservar:

- ² Tipos de los atributos.
- ² Número y tipos de argumentos de funciones.
- ² Tipo del valor devuelto por una operación.

Redefinir operaciones por:

- ² Extensión.
- ² Restricción.
- ² Optimización.
- ² Conveniencia.

Reglas prácticas:

- ² Todas las operaciones de consulta y actualización se heredan por todas las subclases.
- ² Las operaciones de actualización que modifican atributos o asociaciones restringidas, deben mantener la misma restricción.
- ² Las operaciones heredadas se pueden redefinir añadiendo un nuevo comportamiento.

### 8.1.17 Clases abstractas

Las clases abstractas sirven para definir un comportamiento y/o estructura común a un conjunto de subclases. No es instanciable (no existen objetos de esa clase). Sin embargo, sus subclases pueden ser clases concretas (que sí son instanciables). Organizan características comunes de varias clases.

Pueden definir el protocolo para una operación sin proporcionar un método correspondiente (operación abstracta). Una clase concreta no puede contener operaciones abstractas. La naturaleza abstracta de una clase es siempre provisional.

### 8.1.18 Herencia múltiple

Esta característica permite a una clase tener más de una superclase, y heredar las características de todos sus padres.

Las principales ventajas son:

- ² Una mayor potencia a la hora de definir nuevas clases.
- ² Oportunidades adicionales de reutilización.
- ² Acercamiento a la forma natural de pensar.

**Ejemplo 8.4** El triángulo relleno significa que las subclases no son disjuntas.

Las principales desventajas se encuentran en los:

- ² Problemas de implementación (por ejemplo, la herencia repetida de C++).
- ² Clase vínculo (join): clase con más de un padre.
- ² Herencia múltiple accidental.

Un instructor es inherentemente Profesor y Alumno, ¿cómo modelar un Profesor de una Universidad que recibe clases en otra?

### 8.1.19 Metadatos

Son Clases de las Clases. Son datos que describen otros datos: los modelos son metadatos, definición de una clase, p.e.

Los SGBDR también utilizan metatablas para almacenar las definiciones de las tablas (el catálogo).

Introducen confusión pues difuminan la diferencia entre el modelo y el mundo real.

Patrones y Metadatos:

- ² La instanciación relaciona una clase con sus instancias.
- ² La representación explícita del proceso de instanciación puede ser útil si tanto la clase como los objetos han de ser manipulados como objetos.

### 8.1.20 Características de clase

Las clases pueden ser consideradas también como Metaobjetos.

Atributos de clase:

- ² Describen un valor común para todos los objetos de la clase.
- ² Permiten almacenar información que puede usarse como valor por defecto en la instanciación.
- ² Permiten almacenar información sobre las instancias de la clase.

Operaciones de clase:

Operación que tiene lugar sobre atributos de la clase.

Se invoca sobre la clase y no sobre una instancia concreta.

**Ejemplo 8.5** New.

**Ejemplo 8.6** Smalltalk: `x := Dictionary new`

**Ejemplo 8.7** C++: en la declaración del objeto se llama al constructor de la clase

### 8.1.21 Claves candidatas

Una clave candidata es un conjunto mínimo de atributos que identifican un objeto o enlace.

El identificador de objeto (OID) es siempre una clave candidata. Es un concepto lógico muy utilizado en el mundo de los SGBDR.

Se denotan en los diagrama de clases mediante corchetes.

### 8.1.22 Restricciones

Es una relación funcional entre entidades (clase, objeto, atributo, enlace o asociación) dentro del Modelo de Objetos que limita los valores que la entidad puede tomar.

Restricciones sobre enlaces:

<sup>2</sup> Restricciones generales.

Se expresan mediante lenguaje natural o ecuaciones.

Se denota mediante una línea de puntos entre las clases implicadas y con la descripción entre llaves.

<sup>2</sup> Objetos, enlaces y atributos derivados.

Un objeto derivado se define como una función de uno o más objetos.

Es redundante, pero se introduce en el Modelo para facilitar la comprensión.

También existen enlaces y atributos derivados.

### 8.1.23 Homomorfismo

El homomorfismo es un tipo especial de asociación que supone una correspondencia entre dos asociaciones. Aparecen en la práctica en aplicaciones complejas que tratan metadatos.

### 8.1.24 Modelo de Datos vs Modelo de Objetos

Una BD se desarrolla mediante un Modelo de Datos.

1. Se construye el Modelo de Datos sobre el dominio de la aplicación.
2. Se transforma del Modelo de Datos en un Diseño de la BD mediante la aplicación de una serie de transformaciones estándar (normalización).

Un Sistema de Objetos se construye modelando mediante técnicas diferentes, pues las técnicas del Modelo de Datos son bastante limitadas para soportar el Modelo de Objetos.

**Consejos prácticos**

- ² No comenzar construyendo diagrama de clases; primero, es necesario comprender el problema.
- ² Intentar mantener el Modelo sencillo.
- ² Seleccionar con cuidado los nombres.
- ² No introducir punteros o referencias a otros objetos como atributos.
- ² Intentar evitar asociaciones n-arias.
- ² No intentar establecer el grado de multiplicidad perfecto al principio.
- ² No introducir atributos de enlace dentro de la clase.
- ² Utilizar asociaciones cuali...cadas donde sea posible.
- ² Intentar evitar generalizaciones profundamente anidadas.
- ² Intentar asociaciones uno a uno.
- ² No se sorprenda si su modelo requiere una revisión.
- ² Documentar siempre los Modelos de Objetos.

**8.1.25 Resumen del Modelo de Objetos**

El Modelo de Objetos describe la estructura estática de las clases, los objetos y sus relaciones.

Conceptos de clase, atributo, objeto y operación.

Enlaces y asociaciones.

Agregación.

Generalización y herencia.

Características avanzadas: clases abstractas, herencia múltiple, metadatos, claves candidatas y restricciones.

**8.1.26 Diferencias entre el Modelo de Datos y el Modelo de Objetos.****Glosario**

agregación generalización método rede...nición

asociación identidad módulo cuali...cación

atributo herencia multiplicidad rol

clase instancia objeto discriminador

enlace operación atributo de enlace clasi...cación

especialización clase abstracta restricción clase concreta

clase vínculo metadato metaclassa clave candidata

propagación homomor...smo herencia múltiple

### 8.1.27 Cambios introducidos por OMT-2

OMT-2 introduce cambios en el modelo de objetos (compatibilidad con UML)

Notación de Objetos: Objeto como un rectángulo con la siguiente sintaxis

NombreObjeto : nombreClase

Multiplicidad: la multiplicidad muchos añade información al símbolo del círculo relleno, incorporando información textual adicional.

Atributos de la relación: un atributo de la relación es un caso particular de una clase asociación.

Clase asociación: la notación es ahora una línea discontinua que une la clase asociación con la asociación.

Generalización: la representación es aquí una flecha cuyo extremo "toca" la clase más genérica. Puede organizarse en árbol.

Símbolo de cruce: se introduce para evitar ambigüedades.

## 8.2 El Modelo Dinámico

### 8.2.1 Introducción

El Modelo Dinámico, modela los aspectos del sistema que tienen que ver con el tiempo y los cambios.

o modela el Control, aspecto del sistema que describe la secuencia de operaciones que ocurren como respuesta a unos estímulos.

Conceptos principales del Modelo Dinámico:

² Sucesos o Eventos: representan los estímulos externos.

² Estados: representan valores de los objetos.

La notación a utilizar en este libro es la de David Harel, para dibujar los diagrama de estado estructurado, empleando contornos anidados con objeto de mostrar la estructura.

### 8.2.2 Sucesos y Estados

El **Suceso** es estímulo individual de un objeto a otro. Ocurre en un determinado instante y no tiene duración.

El **Estado** son valores de los atributos y enlaces de un objeto en un instante dado.

Se representan con los Diagrama de Estado, que no es otra cosa que un patrón de sucesos, estados y transiciones entre estados para una clase determinada.

El Modelo Dinámico consiste en múltiples diagrama de estado, uno por cada clase con comportamiento dinámico significativo.

La definición de sucesos y estados depende del nivel de abstracción que se utilice.

#### Sucesos

Dos sucesos pueden darse consecutivamente de forma lógica (causa-efecto) o no (concurrentes).

Son un medio de transmisión de información de un objeto a otro, unidireccional.

**Clase de sucesos:** Indica estructura y comportamiento comunes. Pueden contener atributos para expresar la información que manejan.

**Ejemplo 8.8** Pulsación de un botón del ratón (botón, posición) o la salida de un vuelo (avión, número vuelo, ciudad).

### Escenarios y trazas de sucesos

El Escenario es una secuencia de sucesos que tienen lugar durante una ejecución particular del sistema.

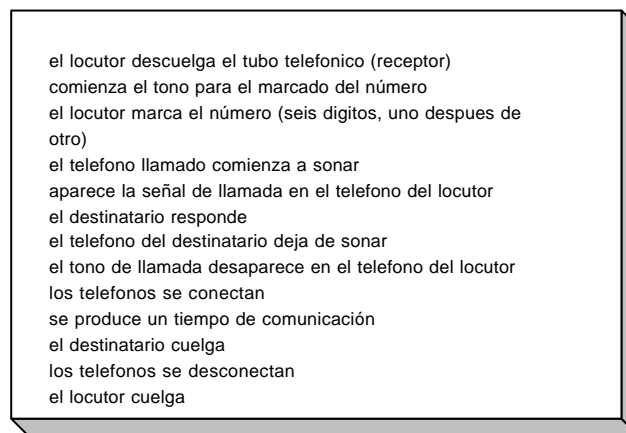


Figura-8.9: Escenario para una llamada telefónica

En los Escenarios y trazas de sucesos pueden enviarse sucesos concurrentes.

La descripción de escenarios es el primer paso del modelo dinámico, al que debe seguir la identificación de los objetos emisores y receptores de sucesos.

El Diagrama de traza de sucesos (DTS): especifican la secuencia de sucesos en el tiempo. En la etapa de diseño hay que representar los DTSs.

### Estados

Abstracción de los valores de los atributos y enlaces de un objeto. Especifica la respuesta de un objeto a los sucesos que llegan (acción/cambio de estado). Representan intervalos de tiempo. En su definición, se ignoran los atributos que no intervienen en el comportamiento del objeto.

### Caracterización de un Estado

Caracterización mediante la indicación de varias pautas que identifican al objeto y que pueden solaparse:

<sup>2</sup> Nombre: alarma sonando

<sup>2</sup> Descripción: la alarma en el reloj suena para indicar la hora seleccionada.

Secuencia de sucesos que producen el estado:

- ² Activar alarma (hora seleccionada)
- ² Cualquier secuencia excluyendo desactivar (alarma)
- ² Hora actual=Hora seleccionada

Condiciones que caracterizan el Estado:

- ² Alarma = sí, Hora selec.  $\leq$  Hora actual  $\leq$  Hora selec. + 20 segundos, y no botón pulsado desde Hora selec.

Sucesos aceptados en el Estado:

- ² Suceso Acción Siguiente Estado

Ejemplo 8.9 Hora actual = Hora selec. + 20 iniciar(alarma) Normal

Ejemplo 8.10 Botón Pulsado (cualquiera) iniciar (alarma) Normal

### Diagrama de Estado

Relaciona Sucesos y Estados. El siguiente Estado depende del Suceso recibido y del Estado actual.

Las transiciones para abandonar un Estado deben corresponder a sucesos diferentes.

Si un Suceso no genera una transición que abandone el Estado, se ignora. Los Estados no de...nen por completo todos los valores del objeto y pueden representar bucles continuos o ejecuciones no repetitivo..

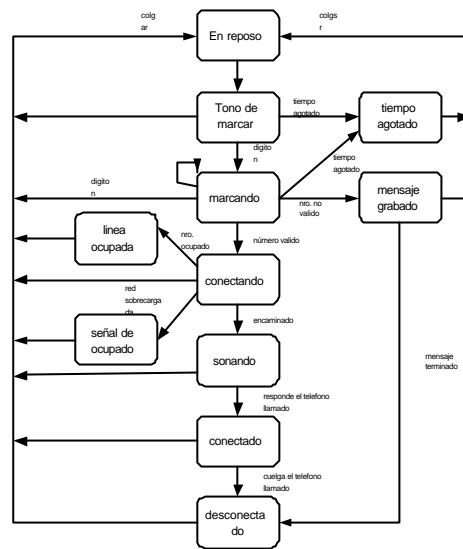
Los diagrama que representan ejecuciones no repetitivo:

- ² Tienen estados iniciales y ...nales
- ² Se utilizan como subrutinas referenciadas desde diversos diagrama de más alto nivel.

### Condiciones

- ² Funciones lógicas sobre los valores de los objetos.
- ² Se indican entre corchetes.
- ² Válidas durante un intervalo de tiempo.
- ² Se pueden utilizar como guardas sobre las transiciones:

se disparan sólo si se cumple la condición.



Figura~8.10: Diagrama de Estado para una línea telefónica

### 8.2.3 Operaciones

Están asociadas a las transiciones entre Estados.

Tipos:

**Acciones:** operación instantánea asociada a un suceso. También pueden representar operaciones de control interno.

**Actividades:** operación con duración en el tiempo, asociada a un Estado.

### 8.2.4 Diagrama de Estado anidados

Los Diagrama de Estado se pueden estructurar de forma similar a la de los objetos.

<sup>2</sup> Generalización: actividades de anidamiento que permiten organizar los Estados y Sucesos de forma jerárquica.

<sup>2</sup> Agregación: permite la descomposición de un Estado en componentes ortogonales, siendo equivalente a la concurrencia de los Estados.

### Anidamiento de Estados

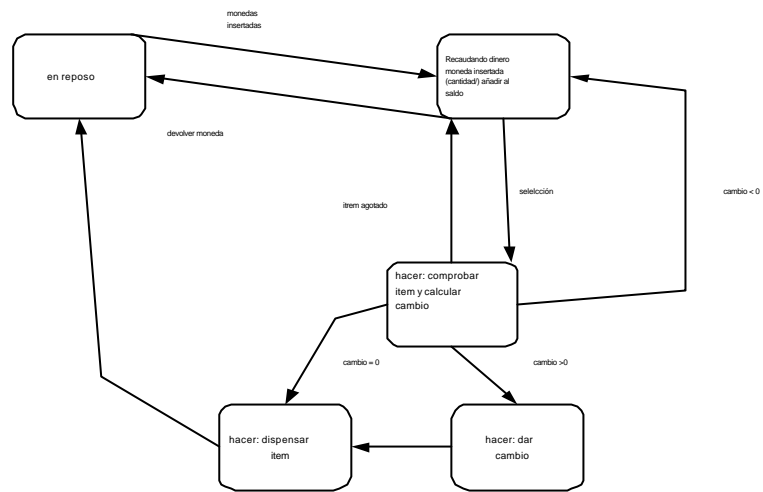
Una actividad en un Estado puede expandirse como un Diagrama de Estado de nivel inferior, en el que cada Estado representa una etapa de la actividad.

Los Sucesos también pueden expandirse en Diagrama de Estados subordinados.

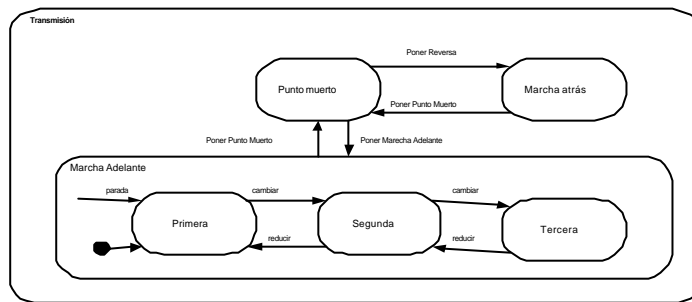
### 8.2.5 Generalización de Estados

En general, los Estados en un Diagrama anidado pueden interactuar con otros Estados.

Los Estados pueden tener subestados que heredan sus transiciones, a menos que las 'reemplacen' por otras equivalentes.



Figura~8.11: Anidamiento de diagramas de estados (máquina expendedora de dinero)



Figura~8.12: Diagrama de estado de la transmisión de un vehículo

### 8.2.6 Generalización de Sucesos

Los Sucesos pueden organizarse en una jerarquía de generalización en la que se heredan los atributos de los Sucesos.

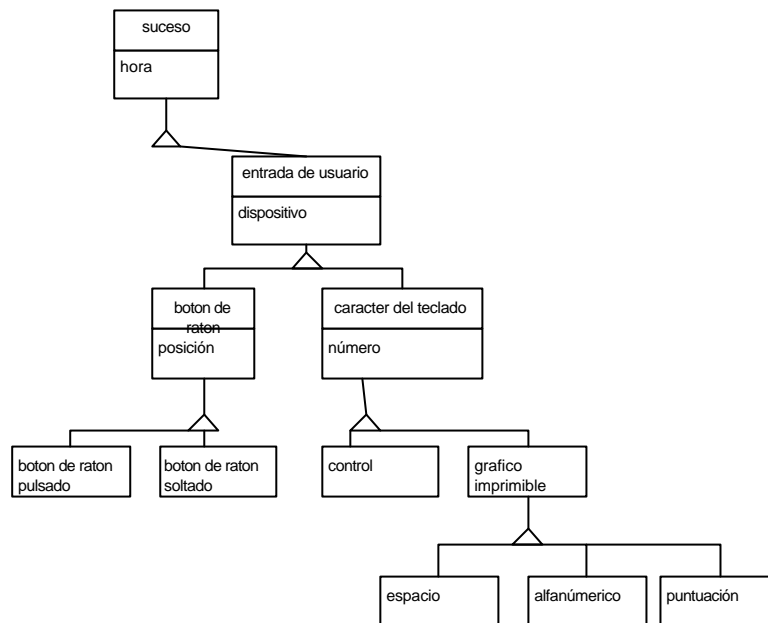
Permiten utilizar distintos niveles de abstracción en diferentes partes del Modelo.

### 8.2.7 Concurrencia

Un Diagrama de Estados para un sistema compuesto es una colección de diagramas, uno por cada componente. La agregación implica Concurrencia. Los Diagramas de Estado de los componentes suelen ser independientes, pero pueden interactuar (las guardas de las transiciones para un objeto en un determinado estado pueden depender de otro objeto).

Concurrencia dentro de un objeto:

Podemos dividir un objeto en subconjuntos de atributos y enlaces, cada uno con su propio diagrama.



Figura~8.13: Jerarquía parcial de sucesos (teclado)

### Acciones de Entrada (Entry) y de Salida (Exit)

Acciones que se ejecutan cuando se entra o se sale de un Estado por cualquier suceso.

Estas acciones se simplifican cuando todas las transiciones impliquen esas mismas acciones para la entrada o la salida.

Orden de ejecución de las acciones:

- 2 Acciones en la transición que llega.
- 2 Acciones de Entrada.
- 2 Actividades.
- 2 Acciones de Salida.
- 2 Acciones en la transición de Salida.

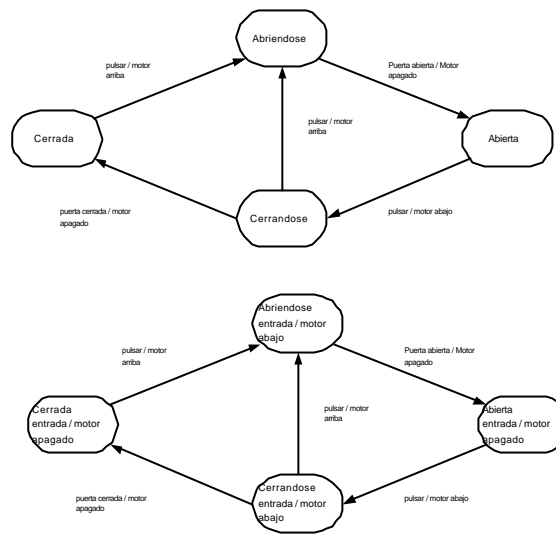
### Acciones internas

Un Suceso puede provocar que se ejecute una acción sin cambiar de Estado.

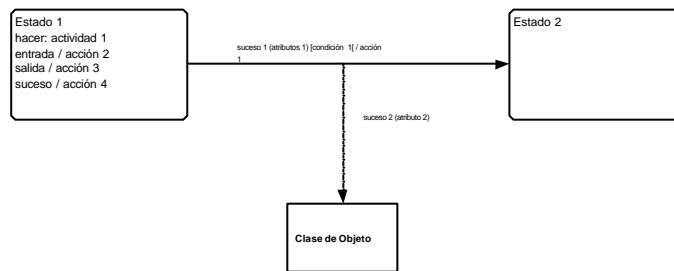
En la notación se refleja como suceso / acción

### Transiciones automáticas y Envío de Sucesos

Transiciones automáticas:



Figura~8.14: Arriba: Acciones en la transiciones. Abajo: Acciones de entrada en los distintos estados



Figura~8.15: Resumen de la notación extendida para Diagramas de Estados

<sup>2</sup> Se disparan siempre que la actividad asociada con el Estado fuente ha terminado, y se cumplen las condiciones de las guardas (transiciones lambda).

Envío de Sucesos:

<sup>2</sup> Los sucesos pueden dirigirse a un solo objeto o a un conjunto de ellos.

<sup>2</sup> La recepción concurrente de Sucesos se denomina condición de carrera. No es un error de diseño pero en general no deseadas.

### Sincronización de Actividades Concurrentes

A veces, un objeto puede realizar actividades de forma concurrente.

Los pasos internos de las actividades no han de estar sincronizados, pero todas ellas han de completarse para poder pasar al siguiente Estado:

<sup>2</sup> División del Control (split).

- <sup>2</sup> Fusión del Control (merge).

### 8.2.8 Relación entre Modelo de Objetos y Modelo Dinámico

El Diagrama de Estados describe toda o una parte del comportamiento de un objeto de una clase. Los Estados son los equivalentes a los valores de los atributos y enlaces de las clases. Los Sucesos se pueden representar como operaciones en el Modelo de Objetos. La estructura del Modelo Dinámico está relacionada y restringida por la del Modelo de Objetos (generalización por restricción).

Un Estado Compuesto es el agregado de más de un subestado concurrente. En el Modelo de Objetos hay tres fuentes de concurrencia:

- <sup>2</sup> Concurrencia entre componentes: cada componente tiene su propio Estado independiente.
- <sup>2</sup> Concurrencia entre partes de un componente: agregación dentro de un objeto.
- <sup>2</sup> Concurrencia del objeto: comportamiento concurrente de un objeto.

El Modelo Dinámico de una clase lo heredan sus subclases.

La jerarquía de Sucesos es independiente de la jerarquía de clases (pueden derivarse a través de diferentes clases de objetos, y son más expresivos que las operaciones).

### 8.2.9 Consejos prácticos

- <sup>2</sup> Construir Diagrama de Estado sólo para las clases con comportamiento dinámico significativo.
- <sup>2</sup> Utilizar escenarios.
- <sup>2</sup> Considerar sólo los atributos relevantes.
- <sup>2</sup> Verificar la consistencia de los diferentes Diagramas de Estado.
- <sup>2</sup> Considerar las necesidades de la aplicación a la hora de decidir el grado de granularidad de los Sucesos y Estados.
- <sup>2</sup> Distinguir entre actividades y acciones.
- <sup>2</sup> Poner acciones de entrada cuando todas las transiciones entrantes generen la misma acción. Ídem para las de Salida.
- <sup>2</sup> Utilizar Estados anidados cuando las mismas transiciones se apliquen a varios Estados.
- <sup>2</sup> Intentar mantener los Diagramas de estados de las subclases independientes de los de las superclases.
- <sup>2</sup> Tener cuidado con las posibles condiciones de carrera en los Diagramas de Estado.

### Resumen del Modelo Dinámico

Representa la información de control, secuencia de sucesos, estados y operaciones que ocurren dentro de un sistema de objetos.

Patrón para los posibles escenarios que se puedan presentar.

**Notación:** compromiso entre simplicidad y expresividad (donde no llegue la notación, lenguaje natural).

**Suceso:** señal de que ha pasado algo.

**Estado:** intervalo entre Sucesos que especi...ca el contexto en el que se interpretan los Sucesos:

Generalizaciones de restricción sobre una clase.

Una subclase hereda los diagramas de estado de su clase base.

**Transición:** respuesta a un suceso, incluido el siguiente estado, acciones posibles y sucesos enviados a otros objetos

Transiciones automáticas: se disparan cuando se cumplen sus condiciones y se han realizado todas las actividades dentro del Estado.

Las transiciones pueden dividir o juntar el flujo de control.

**Acción:** operación instantánea en respuesta a un suceso.

**Acciones de Entrada y Salida:** se pueden asociar con un Estado.

Acciones Internas que representan transiciones que no cambian de Estado.

**Actividad:** secuencia de acciones que requieren un tiempo para realizarse.

Sucesos y Estados pueden expandirse mediante Diagramas de Estados Anidados para mostrar mayor detalle, y pueden organizarse en jerarquías de herencia.

Los objetos son inherentemente concurrentes. Los Diagramas de Estado muestran la concurrencia:

² como agregado de Estados Concurrentes

² como agregado dentro de un objeto

² mediante el comportamiento concurrente de los objetos.

## 8.3 El Modelo Funcional

El modelo funcional describe los cálculos existentes del sistema sin describir ni como ni cuando se calculan. El modelo funciona especi...ca lo que sucede, el modelo dinámico especi...ca cuando sucede y el modelo de objetos especi...ca a que le sucede.

### 8.3.1 Diagrama de Flujo de Datos

El modelo funcional consta de múltiples diagrama de flujo de datos, que especi...can el signi...cado de las operaciones y de las restricciones. Un diagrama de flujos de datos (DFD) muestra las relaciones funcionales entre los valores calculados por un sistema, incluyendo los valores introducidos, los obtenidos, y los almacenes internos de datos. Es un grafo que muestra el flujo de valores de datos desde sus fuentes en los objetos mediante procesos que los transforman, hasta sus destinos en otros objetos. No muestra información de control.

Un diagrama de flujo de datos contiene procesos que transforman datos, flujos de datos que los trasladan, objetos actores que producen y consumen datos, y de almacenes de datos que los almacenan de forma pasiva. La siguiente figura muestra un DFD típico.

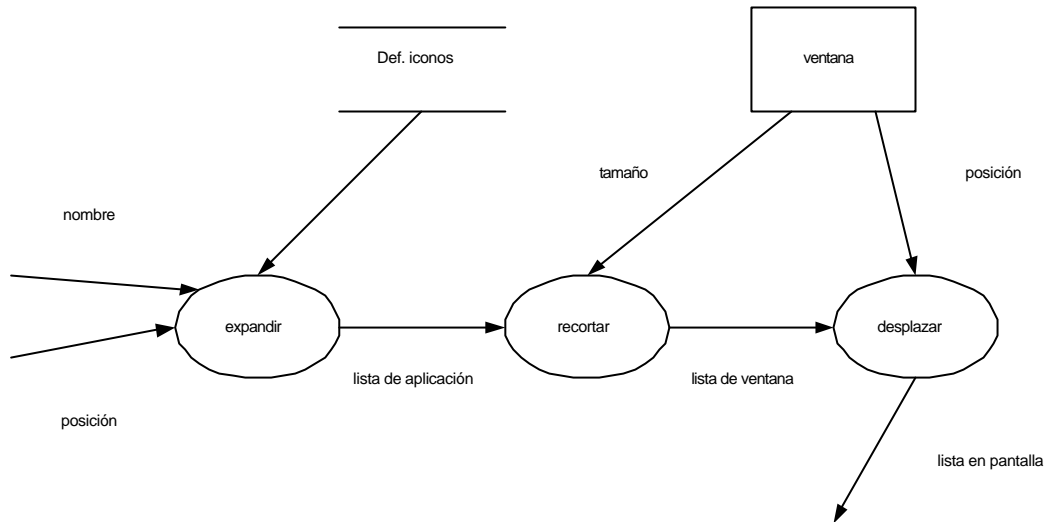


Figura-8.16: Diagramas de flujo de datos típico

### Procesos

Un proceso transforma valores de datos. Los procesos del más bajo nivel son funciones puras, sin efectos laterales. Un grafo completo de flujos de datos es un proceso de alto nivel. Los procesos pueden tener efectos laterales si contienen componentes no funcionales, tales como almacenes de datos u objetos externos.

Los resultados de estos procesos dependen del comportamiento del sistema, según se especi...que en el modelo dinámico.

Los procesos se dibujan en forma de elipse que contienen una descripción de la transformación, normalmente su nombre. Cada proceso tiene un número ...jo de fechas de entrada y salida de datos, cada una de las cuales lleva un valor de un tipo dado.

Las entradas y salidas se pueden rotular, para mostrar su papel en el cálculo, pero es frecuente que baste el tipo de valor asociado al flujo de datos.

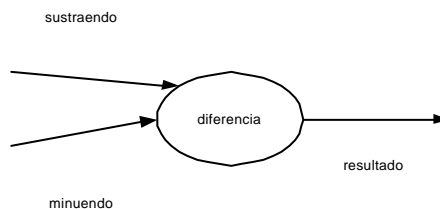


Figura-8.17: Proceso

### Actores

Un actor es un objeto activo que controla el gráfico de flujo de datos produciendo o consumiendo valores. Los actores están asociados a las entradas y salida del gráfico del flujo de datos.

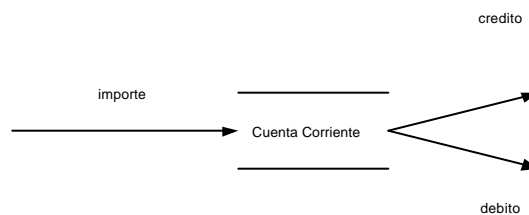
En cierto sentido, los actores yacen en la frontera del gráfico, pero hacen que concluya el flujo de datos como fuentes y sumideros de datos, así que en algunos casos se denominan terminadores.

### Almacenes de datos

Un almacén de datos es un objeto pasivo dentro de un diagrama de flujo de datos que almacena datos para su posterior utilización. Difieren con los actores en que no generan ninguna operación por sí mismos, sino que se limitan a responder a solicitudes de almacenamiento y acceso a datos.

Los almacenes se dibujan en forma de un par de líneas paralelas que contienen el nombre del almacén. Las flechas de entradas indican información u operaciones que modifican los datos almacenados; esto incluye la adición, modificación o el borrado de elementos. Las flechas de salidas indican información que se ha extraído del almacén.

La figura siguiente muestra un almacén típico dentro de un DFD.



Figura~8.18: Almacen

Tanto los actores como los almacenes de datos son objetos. Se diferencian en que su comportamiento y utilización suelen ser distintos, aun cuando en un lenguaje orientado a objetos ambos pudieran ser implementados como objetos. También el almacén puede ser implementado como un fichero y el terminador como un dispositivo externo.

### Diagrama de flujo de datos anidados

Un DFD resulta especialmente útil para mostrar funcionalidad de alto nivel de un sistema, y su descomposición en unidades funcionales más pequeñas. Un proceso se puede expandir en otro diagrama de flujo de datos. Todas las entradas y salidas del proceso lo serán también del nuevo diagrama, también pueden tener almacenes de datos que no se muestren en el diagrama de nivel superior.

### Flujos de control

Un diagrama de flujo de control muestra todas las posibles vías de computación para los valores. No muestra cuales son las vías que se ejecutan ni en que orden. Las decisiones y la secuencias son problemas de control, que forman parte del modelo dinámico. Una decisión afecta a si una o más funciones llegan incluso a ejecutarse, en lugar de proporcionarlos un valor. Aun cuando las funciones no poseen valores de entrada procedentes de estas funciones de decisión, a veces resulta útil incluirlas en este modelo funcional, para que no se olviden, y para que sea posible mostrar sus dependencias de datos. Esto se hace incluyendo flujos de control en el diagrama de flujo de datos.

Un flujo de control es un valor booleano que afecta a si un proceso es o no evaluado. Los flujos de control se muestran en los diagramas con una línea discontinua que va desde un proceso que produce un valor booleano hasta el que se esta controlando.

### Especificaciones de proceso

Toda operación se podrá especificar de diferentes maneras, entre las que se cuentan las siguientes:

- ² Funciones matemáticas, tales como las funciones trigonométricas;
- ² Tablas de valores de entrada y salida para pequeños conjuntos de datos
- ² Ecuaciones que especifican la salida en términos de la entrada
- ² Condiciones previas y posteriores
- ² Tablas de decisión
- ² Pseudocódigo
- ² Lenguaje natural
- ² Diagrama de flujos
- ² Diagrama de árboles
- ² etc.

### Restricciones

Una restricción muestra la relación entre dos objetos al mismo tiempo (tal como la frecuencia y la longitud de onda) o bien entre distintos valores del mismo objeto en instantes diferentes (tal como el numero de acciones de un fondo de pensiones). Las restricciones se pueden expresar como una función total o como una función parcial.

Las restricciones pueden aparecer en todas las clases de modelo. Las restricciones de objetos especifican que algunos objetos dependen entera o parcialmente de otros objetos. Las restricciones dinámicas especifican relaciones entre los estados o sucesos de distintos objetos. Las restricciones funcionales especifican limitaciones aplicables a operaciones.

## 8.4 Preguntas y Ejercicios de Revisión

1. Prepare un diagrama de clase para representar su árbol genealógico.. Desde su tatarabuelo hasta sus nietos.
2. Prepare un diagrama de objetos del ejercicio anterior.
3. Prepare un diagrama de objetos que muestren al menos 10 relaciones entre las siguientes clases de objetos. Incluir asociaciones, agregaciones y generalizaciones. Si es posible utiliza asociaciones cuali...cadas y mostrar la multiplicidad en los diagramas.

Universidad, Campus, Consejo Académico, aulas, alumno, profesor, bar, mantenimiento, laboratorio, mobiliario.

1. Añadir atributos y operaciones al ejercicio anterior.
2. Decidir a que categoría (generalización, agregación o asociación) pertenecen las relaciones siguientes. Puede existir relaciones ternarias.

<sup>2</sup> Todo país tiene una Ciudad que es su Capital.

<sup>2</sup> Los archivos contienen registros.

<sup>2</sup> Los módems y teclado son dispositivos de E/S.

<sup>2</sup> Una ruta conecta dos ciudades.

1. Preparar una porción de un diagrama de objetos para un sistema de préstamo de libros en una biblioteca, sistema que mostrara la fecha en que debe devolverse un libro y las multas por retraso para libros que no se hayan devuelto a tiempo en forma de objetos derivados.
2. Escribir un Escenario para las actividades siguientes:
3. Trasladar una cabra, un lobo, alfalfa a través del río Parana, empleando una canoa. Solo se puede llevar una cosa en cada viaje. Tener en cuenta que si dejamos a la cabra con el lobo, este se comerá a la cabra. Si dejamos a la cabra sola con la alfalfa, esta se comerá la alfalfa.
4. Describa el signi...cado de la ...gura 8.12.

**Ejercicio 1.** Se desea desarrollar un sistema de recogida de encuestas a través de Internet. Cualquier persona desde su navegador puede inscribirse en el sistema a través de un formulario. El formulario recoge los siguientes datos obligatorios del nuevo usuario:

<sup>2</sup> Apellidos y Nombre.

<sup>2</sup> Nombre de usuario deseado para acceder al sistema.

<sup>2</sup> Dirección de correo electrónico.

<sup>2</sup> Temas en los que está interesado (seleccionando uno o más de los que se le ofrecen).

Para cada uno de estos temas, especi...cará un grado de preferencia en una escala de 1 a 10, según el grado de interés que tenga en cada uno de ellos.

Una vez validados los datos, el sistema le devuelve al usuario una palabra clave única, que le permitirá acceder al sistema, junto con el nombre de usuario. Periódicamente, el usuario recibirá mediante correo electrónico un cuestionario sobre uno de los temas que seleccionó en su inscripción. Al recibirlo, deberá rellenarlo y devolverlo al sistema también mediante el correo. A partir de todos los cuestionarios recogidos sobre un tema, el sistema publicará unos resultados estadísticos que podrán consultarse mediante una página Web. Para contestar al cuestionario el usuario marcará una (y solo una) de las cuatro respuestas propuestas a cada pregunta, pero en el futuro se permitirá también hacer preguntas de otros tipos, como selección múltiple y preguntas abiertas, y debemos acomodar nuestro análisis a esta circunstancia.

Realice, utilizando la metodología OMT, un análisis completo del sistema descrito, incluyendo: análisis de requisitos (casos de uso), escenarios, modelo estático (diagrama de clases), modelo dinámico (diagrama de traza de sucesos de los escenarios principales) y modelo funcional.

Tenga en cuenta que el sistema deberá realizar entre otras las siguientes funcionalidades:

Consultas del número de usuarios que han contestado una determinada pregunta.

Consultas del número de usuarios que han contestado alguna encuesta relacionada con un determinado tema.

**Ejercicio 2.** Se desea desarrollar una aplicación de educación asistida por ordenador a través de Internet. La aplicación requiere a cada nuevo alumno la formalización de una inscripción como paso previo a la matriculación. En la inscripción se le solicitan al alumno sus datos personales y académicos así como sus preferencias en cuanto a las áreas de interés de los cursos; para cada área especi...cará un nivel de preferencia, según el grado de interés. De acuerdo con el per...l introducido, el sistema propone al alumno una serie de cursos de forma que el alumno podrá elegir uno en el que matricularse. A partir de este momento, el alumno consulta y estudia los contenidos de los temas del curso. Cada curso está estructurado en varios temas, cada uno de los cuales tiene varias unidades didáctica.. Cada vez que se siente preparado, solicita al sistema un examen sobre el curso. Cada examen consta de un número variable de ejercicios. Cada ejercicio consiste en contestar a una de las preguntas cerradas, seleccionando como mucho una de las respuestas de esa pregunta. Una pregunta cerrada (tipo test) tiene cuatro respuestas posibles y sólo una de ellas es correcta. El alumno puede matricularse en un nuevo curso siempre que haya terminado el anterior. En el futuro, un examen podrá incluir preguntas no sólo de tipo test con respuesta única sino de otros tipos, como selección múltiple o preguntas abiertas, para lo cual se deberá acomodar el análisis a esta coyuntura.

Realice, utilizando la metodología OMT, un análisis completo del sistema descrito, incluyendo: análisis de requisitos (casos de uso), escenarios, modelo estático (diagrama de clases), modelo dinámico (diagrama de traza de sucesos de los escenarios principales) y modelo funcional.

Tenga en cuenta que el sistema deberá realizar entre otras las siguientes funcionalidades:

Consultas del número de alumnos que han aprobado algún examen de un determinado curso.

Cuántos alumnos han acertado una determinada pregunta.

**Ejercicio 3.** En un determinado sistema operativo orientado a objetos, un proceso puede estar en diversos estados dependiendo del planificador. Cuando se crea el proceso, se sitúa en un estado llamado en espera, en el cual se encuentra esperando para entrar en la CPU. El sistema a modelar contempla los siguientes estados para los procesos:

<sup>2</sup> En ejecución (está en la CPU)

<sup>2</sup> Suspendido (se estaba ejecutando y se le ha retirado de la CPU por alguno de los siguientes motivos: ha llegado un proceso con mayor prioridad, está implicado en un interbloqueo o se encuentra esperando la finalización de una operación de E/S)

<sup>2</sup> Terminado (ha finalizado)

Encontrar todas las posibles transiciones entre estados y las actividades de cada estado para el supuesto anterior y modelarlo utilizando la notación de OMT.

## Referencia Bibliográfica

[12] James Rumbaugh. Modelado y diseño orientado a Objetos. Prentice Hall, 1998.

## Parte IV

# Presentación de la Propuesta y Diseño