

# Entornos de Desarrollo

## Tema 6. UML (II)

1º CFGS: Desarrollo de Aplicaciones Web  
IES Severo Ochoa - Elche  
2011/2012



Licencia de Creative Commons.

Entornos de Desarrollo

Tema 6. UML (II)

por: Javier Martín Juan

Esta obra está publicada bajo una licencia Creative Commons Reconocimiento-NoComercial-CompartirIgual 3.0 España con las siguientes condiciones:



Reconocimiento - Debe reconocer los créditos de la obra de la manera especificada por el autor o el licenciador (pero no de una manera que sugiera que tiene su apoyo o apoyan el uso que hace de su obra).



No commercial - No puede utilizar esta obra para fines comerciales



Compartir bajo la misma licencia - Si altera o transforma esta obra, o genera una obra derivada, sólo puede distribuir la obra generada bajo una licencia idéntica a ésta.

Revisión: 8b54b1748861

Última actualización: 11 de diciembre de 2011

Reconocimientos:

- Francisco Aldarias Raya - Plantillas  $\text{\LaTeX}$

## Índice

<b>1. Objetivos</b>	<b>4</b>
<b>2. Diagramas de colaboración</b>	<b>5</b>
Elementos del diagrama . . . . .	5
Ejemplo: diagrama de colaboración . . . . .	5
<b>3. Diagrama de actividades</b>	<b>7</b>
Elementos del diagrama . . . . .	7
Ejemplo: auditorio . . . . .	7
Particiones . . . . .	8
Ejemplo: almacén . . . . .	8
Ejemplo: adivina . . . . .	9
<b>4. Ejercicios propuestos</b>	<b>11</b>
Diagramas de colaboración . . . . .	11
Ejercicio 6.1 . . . . .	11
Ejercicio 6.2 . . . . .	11
Ejercicio 6.3 . . . . .	11
Diagramas de actividades . . . . .	11
Ejercicio 6.4 . . . . .	11
Ejercicio 6.5 . . . . .	11
Ejercicio 6.6 . . . . .	11
<b>5. Bibliografía y documentación adicional</b>	<b>13</b>

## 1. Objetivos

- Realizar diagramas de colaboración
- Realizar diagramas de actividades

## 2. Diagramas de colaboración

Los diagramas de colaboración son un tipo de diagramas de comportamiento que representan la comunicación entre objetos del sistema y el orden en que se envían los mensajes.

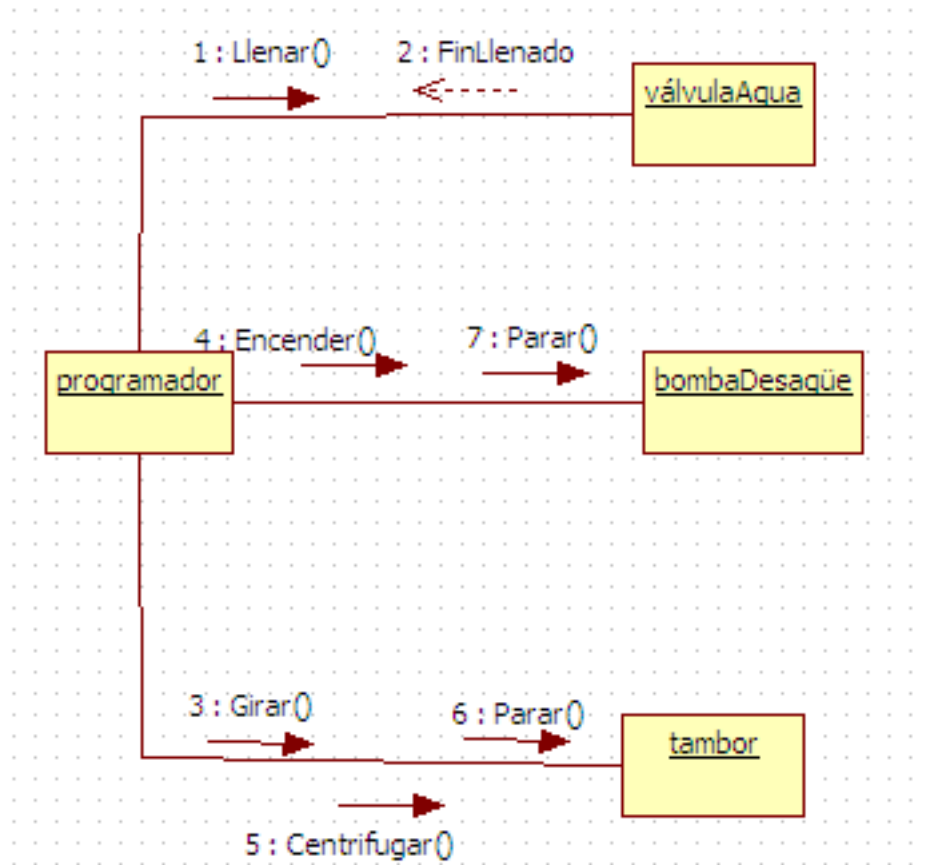
Son parecidos a los diagramas de secuencia, y de hecho se puede pasar de uno a otro fácilmente. La diferencia está en cómo se representa la información. Mientras que los diagramas de secuencia enfatizan el **orden** en que se envían los mensajes, los diagramas de colaboración destacan el **flujo de control** entre los objetos del diagrama.

### Elementos del diagrama

- *Objetos*: Se representan igual que en los diagramas de secuencia, con el nombre del objeto y la clase metidos dentro de una caja.
- *Mensajes*: Cuando dos objetos se envían mensajes entre sí se unen con una línea recta. A lo largo de la línea pondremos los mensajes que se van pasando indicados con una flecha en la dirección adecuada. Al lado de la flecha pondremos el número de orden del mensaje (el mismo que en el diagrama de secuencia) y el nombre del mensaje o el valor de retorno.

### Ejemplo: diagrama de colaboración

El siguiente diagrama de colaboración es equivalente al diagrama de secuencia de la lavadora del tema anterior:



### 3. Diagrama de actividades

Los diagramas de actividad son un tipo de diagrama de comportamiento que representan un flujo de actividades paso a paso y permiten expresar **condiciones**, **iteraciones** y **conurrencia**.

#### Elementos del diagrama

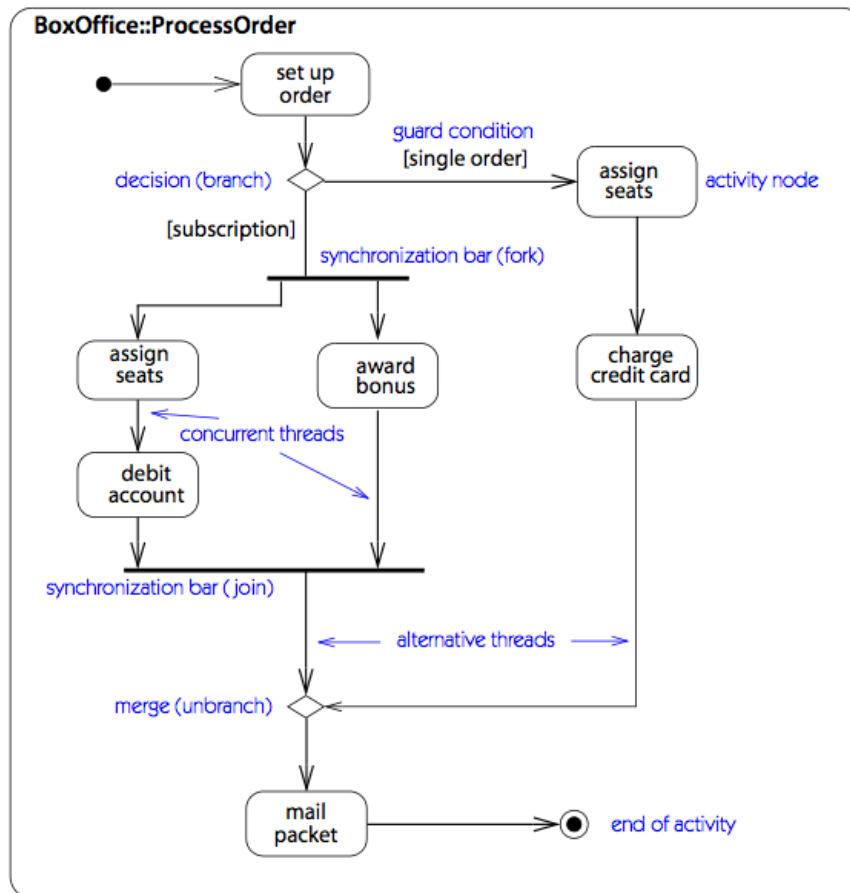
Los diagramas de actividad están formados por una serie de figuras conectadas por flechas. Las figuras más importantes son:

- *Rectángulos redondeados*: representan una actividad
- *Rombos*: representan puntos donde el flujo de control se ramifica en base a alguna condición (*branch*). También se usan para “des-ramificar” o reagrupar (*merge*) flujos previamente ramificados.
- *Barras horizontales*: sirven para sincronizar el inicio o el fin de actividades concurrentes.
- *Círculo negro*: representa el inicio del flujo
- *Un círculo negro rodeado de una circunferencia*: representa el final.
- *Flechas*: las flechas se trazan desde unas figuras a otras para indicar el orden en que se suceden las actividades.

Los diagramas de actividades son parecidos a los diagramas de flujo, con la ventaja de que permiten expresar concurrencia mientras que los diagramas de flujo no.

#### Ejemplo: auditorio

El siguiente diagrama modela el proceso de pedidos de entradas de un auditorio. Las entradas pueden ser de dos tipos: única (*single order*) o abono (*subscription*). En caso de entrada única se asignan los asientos y se cobra por tarjeta de crédito. Si es un abono se concede una bonificación (*bonus*) al tiempo que se asignan los asientos y se carga en cuenta.



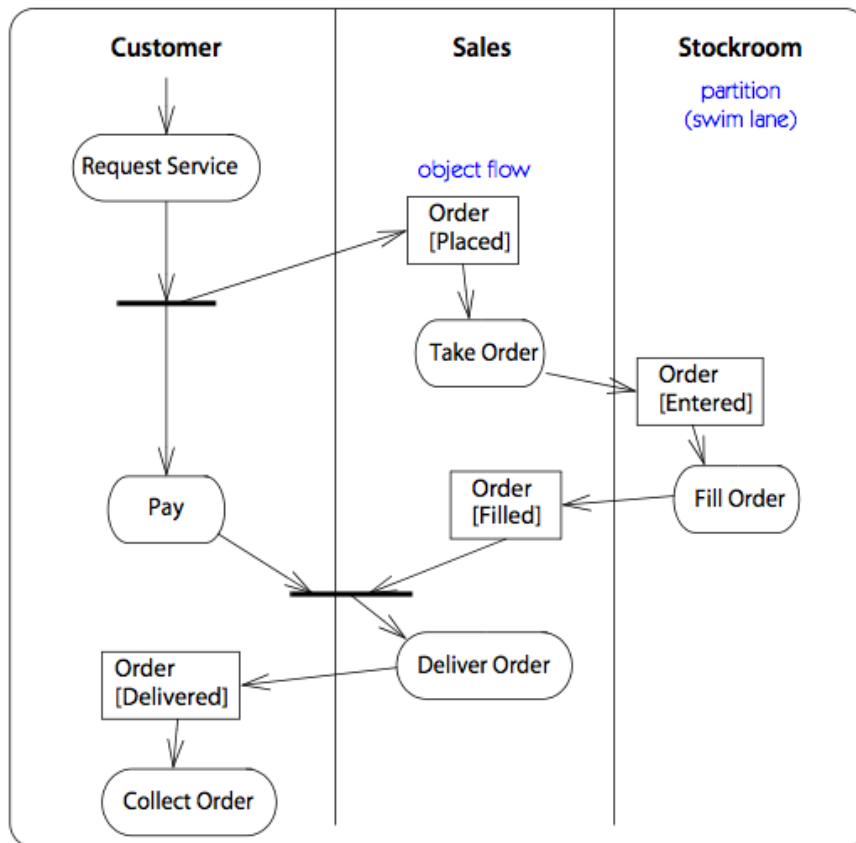
## Particiones

A veces puede ser útil organizar las actividades del modelo según la responsabilidad (“quién las realiza”). Para ello dividimos el diagrama en *particiones* mediante líneas verticales. Por el aspecto del diagrama a las particiones también se les puede llamar *calles* (de una piscina) o *swimlanes*.

## Ejemplo: almacén

El siguiente ejemplo muestra el flujo de actividades en un almacén desde que llega un pedido (*request*) hasta que se recoge (*collect*). Los responsables involucrados son el cliente (*customer*), el departamento de ventas (*sales*) y el almacén (*stockroom*).

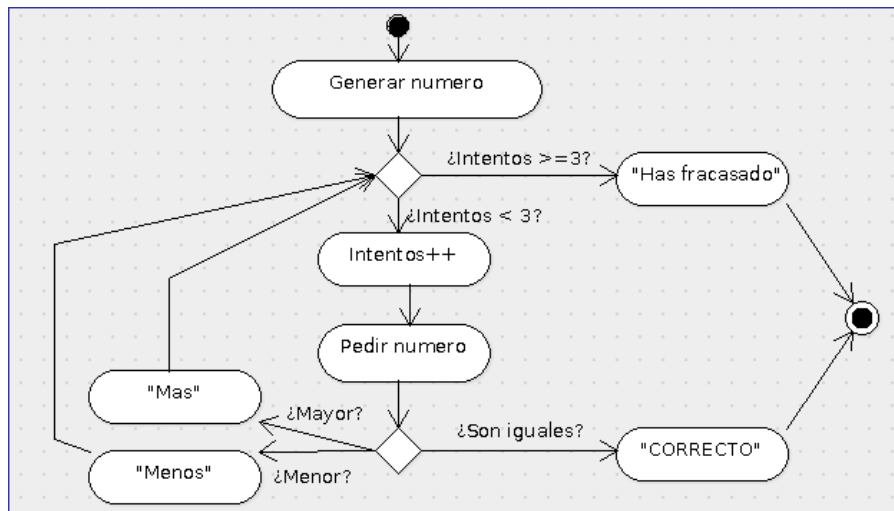




En este diagrama aparece una nueva figura: los *objetos*. Los objetos se dibujan como rectángulos (sin redondear) y representan resultados o valores de entrada a alguna actividad. Cuando representan el resultado se dibuja una flecha desde la actividad al objeto, mientras que si representan los datos de entrada para alguna actividad, la flecha va desde el objeto a la actividad.

### Ejemplo: adivina

El siguiente diagrama modela el juego del tema 1 “Adivina.java”:



## 4. Ejercicios propuestos

### Diagramas de colaboración

#### Ejercicio 6.1

Realiza el diagrama de colaboración equivalente al ejercicio 5.4 del tema anterior.

#### Ejercicio 6.2

Realiza el diagrama de colaboración equivalente al ejercicio 5.5 del tema anterior.

#### Ejercicio 6.3

Realiza el diagrama de colaboración equivalente al ejercicio 5.6 del tema anterior.

### Diagramas de actividades

#### Ejercicio 6.4

Realiza un diagrama de actividades que muestre el proceso de devolución por parte de un cliente de un artículo. El departamento de televentas por Internet se encargará de darle un número de devolución. Una vez obtenido, el cliente envía el artículo defectuoso por mensajería. El almacén es entonces el encargado de la recepción de la mercancía y de la recolocación del artículo. Al mismo tiempo que se recoloca, el departamento de contabilidad actualiza las cuentas. Después de realizarse ambas tareas, el departamento de contabilidad realiza la devolución.

(nota: usar particiones para los distintos responsables)

#### Ejercicio 6.5

Los números primos son aquellos que **sólo** son divisibles (el resto de la división da 0) entre 1 y ellos mismos. Ejemplos de números primos: 2, 3, 5, 7, 11, 13, 17, etc.

Realiza un diagrama de actividades del siguiente algoritmo para decidir si un número N es primo o compuesto: empezaremos dividiendo el número entre 2. Si el resto da 0, el número es compuesto. Si no, probaremos dividiendo por 3, luego por 4, y así sucesivamente hasta llegar hasta N-1. Si ninguna división da como resto 0, entonces diremos que el número es primo.

#### Ejercicio 6.6

Realiza un diagrama de actividades que muestre las acciones de sacar dinero de un cajero:

El cliente del banco inserta la tarjeta en el cajero, la máquina le pide el pin y éste se lo proporciona. Una vez la máquina tiene el PIN lo envía al banco para su verificación. En caso de que el pin sea incorrecto se vuelve a solicitar, y así hasta 3 veces. Si a la tercera oportunidad no se ha introducido el pin correcto, el cajero se traga la tarjeta por seguridad.

Una vez introducido el pin correctamente se solicita al cliente que introduzca la cantidad. El banco comprobará si tiene suficiente saldo disponible. En caso de no tener saldo se expulsa la tarjeta. Si hay

suficiente saldo, el cajero dispensará la cantidad solicitada y paralelamente el banco hará el cargo en cuenta.

Cuando el cliente haya recogido el dinero, se expulsará la tarjeta. El cliente deberá cogerla en menos de 1 minuto. Si tarda más de 1 minuto, el cajero se la tragará por seguridad.

(nota: usar particiones para los distintos responsables)

## 5. Bibliografía y documentación adicional

- UML Reference Manual, 2nd Edition  
Booch, Jacobson, Rumbaugh, Ed. Addison-Wesley
- UML for Java Programmers  
Robert C. Martin, Ed. Prentice Hall
- Learning UML 2.0  
Miles, Hamilton, Ed. O'Reilly
- UML Distilled: A Brief Guide to the Standard Object Modeling Language (3rd Edition)  
Martin Fowler, Ed. Addison-Wesley
- Aprendiendo UML en 24 horas  
Joseph Schmuller, Ed. Prentice Hall
- UML 2.0 - Pocket Reference  
Dan Pilone, Ed. O'Reilly
- Análisis y Diseño Estructurado y Orientado a Objetos de Sistemas Informáticos  
Amescua et al, Ed. McGraw-Hill
- Diseño Orientado a Objetos con UML  
Raúl Alarcón, Grupo Eidos