

## 6.B. Diagramas de interacción.

Sitio: [ALONSO DE AVELLANEDA](#)

Curso: Entornos de desarrollo

Libro: 6.B. Diagramas de interacción.

Imprimido por: Iván Jiménez Utiel

Día: jueves, 2 de abril de 2020, 20:44

# Tabla de contenidos

- [1. Introducción.](#)
- [2. Diagramas de secuencia.](#)
  - [2.1. Representación de objetos, línea de vida y paso de mensajes.](#)
- [3. Diagramas de colaboración.](#)
  - [3.1. Paso de mensajes.](#)
  - [3.2. Ejemplo de un diagrama de colaboración.](#)

# 1. Introducción.

Una vez conocidos los diagramas de casos de uso, se hace necesario buscar la forma de representar como circula la información, los objetos que participan en los casos de uso, los mensajes que envían, y en el momento en que se producen. Disponer de esta información ayudará con posterioridad en el desarrollo de los diagramas de clases.

Los **diagramas de interacción** son vistas del sistema que muestran como grupos de objetos interactúan para un cierto comportamiento. Captan la ejecución de los casos de uso, representando a los actores que participan y los mensajes que se pasan.

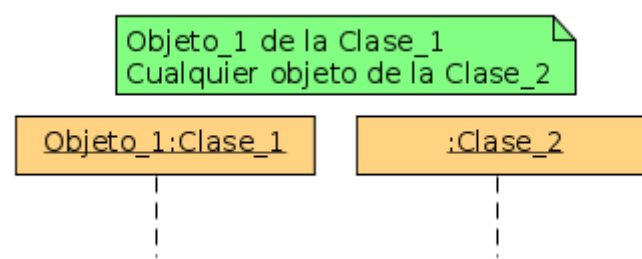
Hay dos tipos de diagramas de interacción: **diagramas de secuencia** y **diagramas de colaboración**.

## 2. Diagramas de secuencia.

En los **diagramas de secuencia**, los objetos/actores que forman parte del escenario de un caso de uso se representan mediante rectángulos distribuidos horizontalmente en la zona superior del diagrama, a los que se asocia una línea temporal vertical de las que salen, en orden, los diferentes mensajes que se pasan entre ellos.

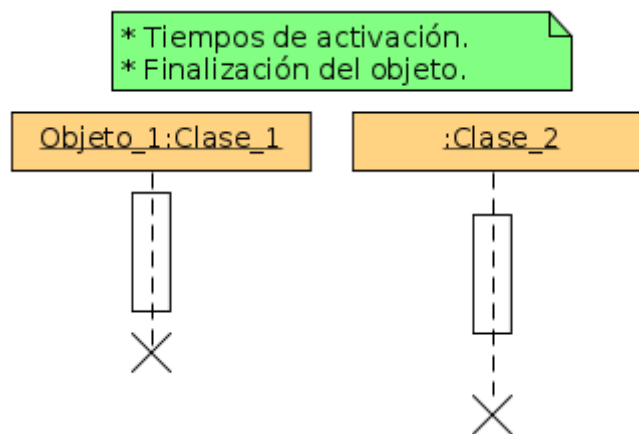
## 2.1. Representación de objetos, línea de vida y paso de mensajes.

En un **diagrama de secuencia**, los **objetos** se dibujan mediante rectángulos y se distribuyen horizontalmente en la parte superior del diagrama. Por cada objeto se identifica su nombre, seguido del símbolo de dos puntos y a continuación el nombre de la clase a la que pertenece. Si no se indica el nombre del objeto, se considera que para el propósito del diagrama es válido cualquier objeto de la clase.



De cada rectángulo sale una línea punteada que representa el paso del tiempo, se denomina **línea de vida**. La línea de vida se prolonga mientras el objeto es relevante en el diagrama, una vez deja de serlo se indica mediante una cruz "X", dejando por tanto de existir a partir de ese momento.

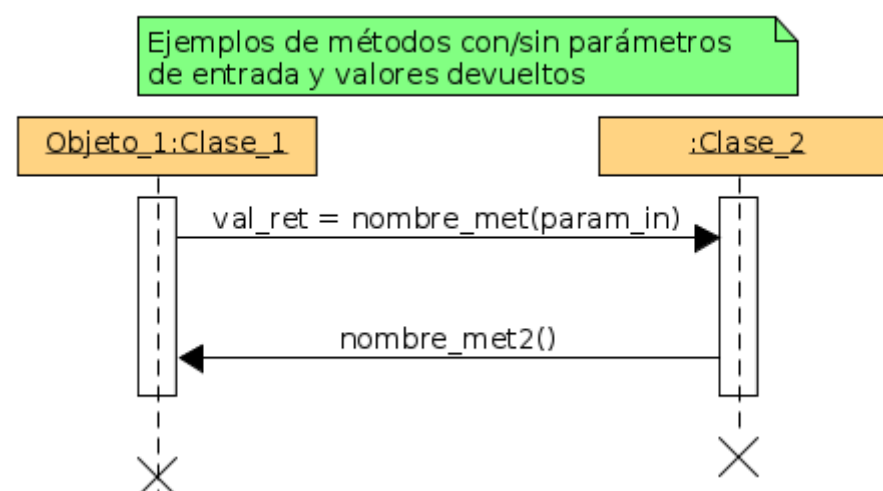
Cuando el objeto toma protagonismo en el intercambio de mensajes, se dice que está **activo** y se indica mediante un recuadro sobre su línea de vida.



Una línea de vida puede estar encabezada por otro tipo de instancias como el sistema o un actor que aparecerán con su propio nombre. Usaremos el sistema para representar solicitudes al mismo, como por ejemplo pulsar un botón para abrir una ventana o una llamada a una subrutina.

### Invocación de métodos.

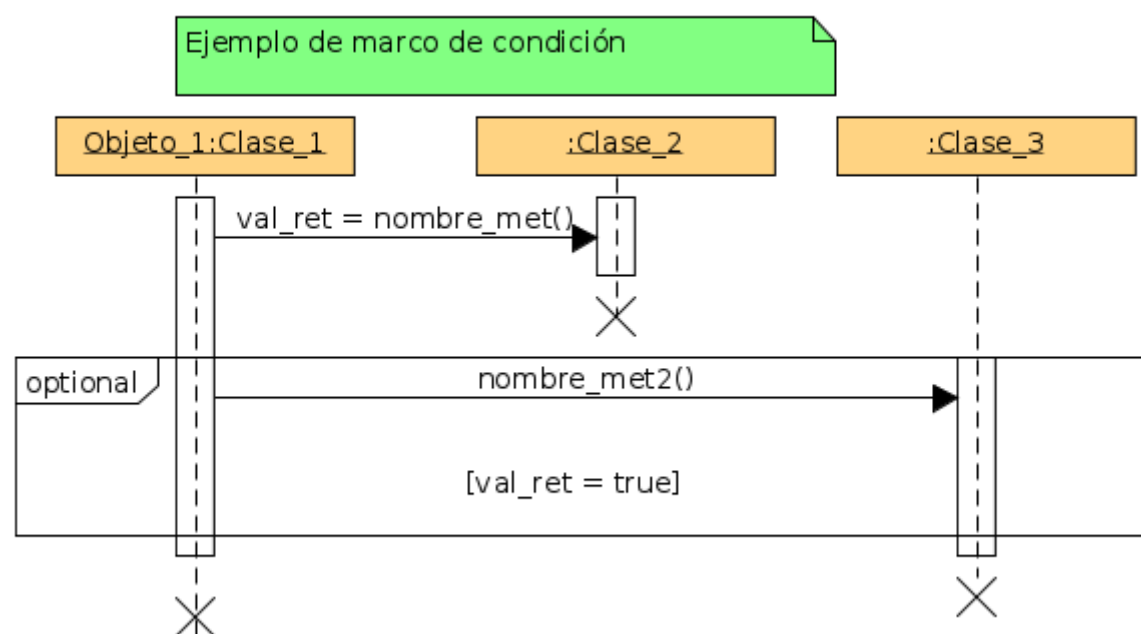
Los **mensajes**, que significan la invocación de métodos, se representan como flechas horizontales que van de una línea de vida a otra, indicando con la flecha la dirección del mensaje. Los mensajes se dibujan desde el objeto que envía el mensaje al que lo recibe, pudiendo ser el mismo objeto emisor y receptor de un mensaje. El orden en el tiempo viene determinado por su posición vertical, un mensaje que se dibuja debajo de otro indica que se envía después, por lo que no se hace necesario un número de secuencia. Los **mensajes** tendrán un **nombre** y pueden incluir **argumentos de entrada**, **valores devueltos** e **información de control** (condición o iteración).



Una notación alternativa para recoger valores devueltos por los métodos es dibujar una línea de puntos finalizada en flecha, que irá desde el objeto destinatario del mensaje al que lo ha generado, acompañado del texto del valor devuelto.

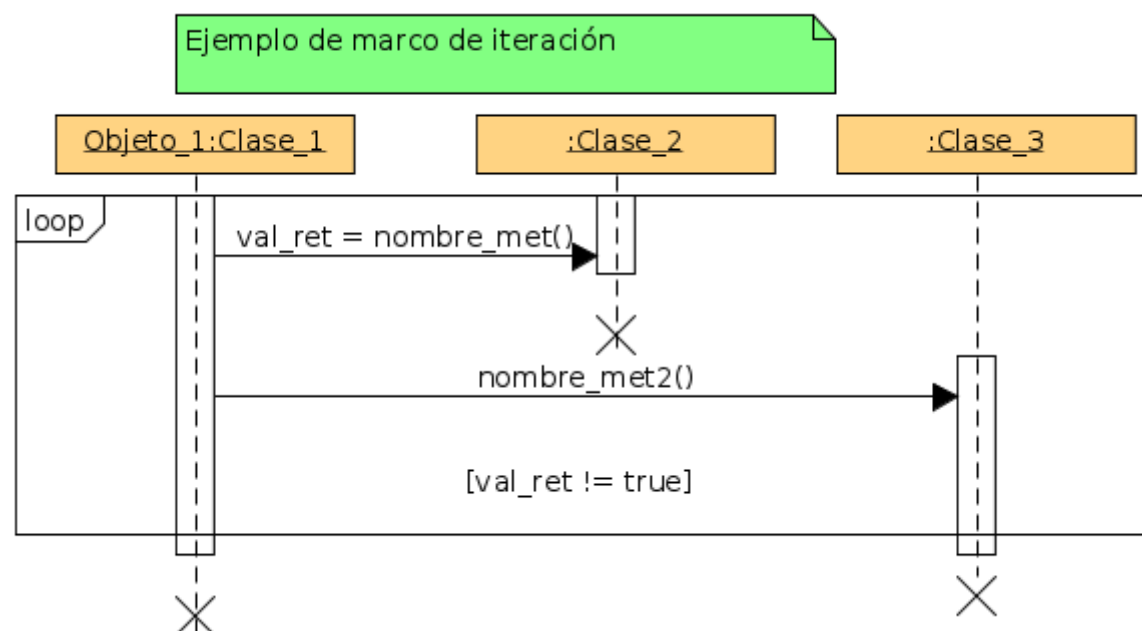
Las secuencias de control; tanto condicionales, como iterativas, se pueden representar usando **marcos**, normalmente se nombra el marco con el tipo de bucle a ejecutar y la condición de parada. También se pueden representar flujos de mensajes condicionales en función de un valor determinado.

La expresión a evaluar para la condición o iteración se representa entre corchetes.



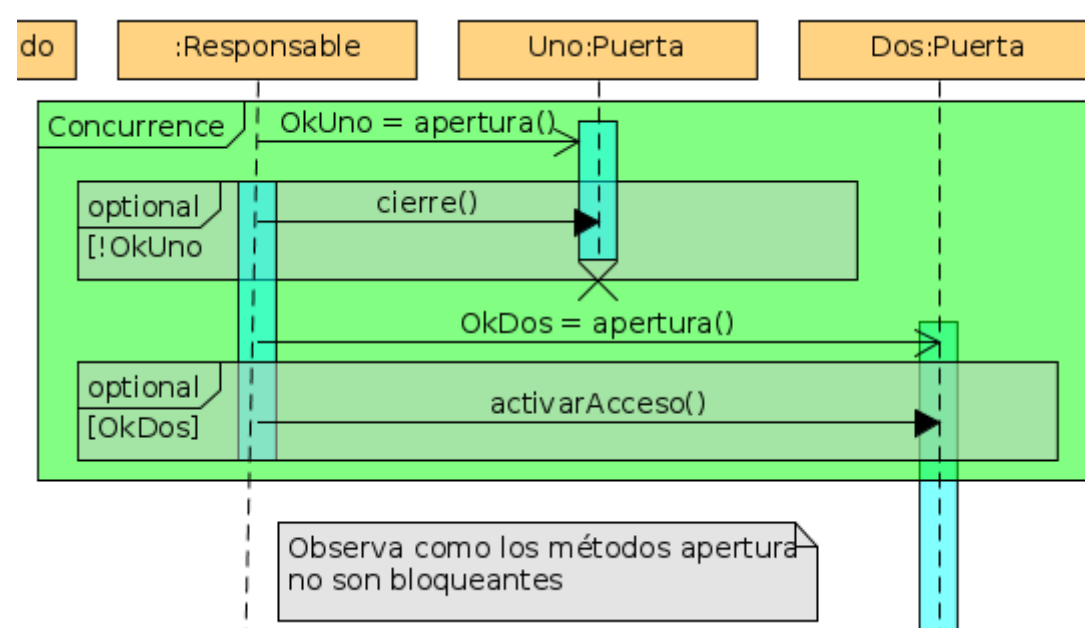
Combinando varios marcos opcionales es posible representar diferentes alternativas en la ejecución de un diagrama de secuencia.

Para el caso de una iteración, tenemos el siguiente ejemplo.



Por defecto los métodos son **bloqueantes**, se entiende que el proceso del diagrama de secuencia completa cada método antes de continuar con el siguiente, es una secuencia de métodos en el tiempo. Pero en ocasiones se producen situaciones en las que se desea mostrar varios procesos en paralelo (**conurrencia**), se puede reflejar mediante el uso de marcos con la etiqueta concurrence.

Junto a los marcos de concurrencia, se hace necesario el uso de **métodos no bloqueantes (asíncronos)**, que permitan en paralelo activar diferentes procesos. La notación utilizada para los métodos asíncronos es una línea finalizada con media cabeza de flecha o en UMLet una línea cuya punta flecha no está rellena.



Por último, indicar que hay situaciones difíciles de reflejar en un diagrama de secuencia, en estas ocasiones el uso de **etiquetas** y **notas** resulta de suma utilidad.

### 3. Diagramas de colaboración.

Al igual que los diagramas de secuencia, **los diagramas de colaboración** muestran una secuencia de ejecución de uno o varios casos de uso. La notación utilizada es muy similar y la principal diferencia radica en el modo de mostrar el orden de mensajes intercambiados entre objetos. Mientras el diagrama de secuencia establece el orden de los mensajes en el tiempo según su posición de arriba-abajo, el diagrama de colaboración lo hace mediante el etiquetado de los mensajes.

Los **diagramas de colaboración permiten una mejor organización visual de los objetos** al no ser obligada su representación en la parte superior del diagrama, en cambio la secuencia temporal suele ser más complicada de seguir.

**Los diagramas de colaboración tienen forma de grafo en el que los nodos son objetos y las aristas son los mensajes que intercambian.**

**UMLet** no dispone de herramientas para la elaboración de diagramas de colaboración directamente. No obstante, no resulta complicado generarlos a partir de los símbolos disponibles para otros diagramas: representación de objetos mediante cajas, paso de mensajes mediante líneas, información de los métodos mediante descripciones textuales; todos ellos disponibles en los diagramas de secuencia de UMLet.

### 3.1. Paso de mensajes.

Para que sea posible el paso de mensajes es necesario que exista una asociación entre los objetos, que se podrá mostrar mediante una línea que los una y una flecha que indique la dirección.

Al igual que sucedía en los diagramas de secuencia, es posible incluir parámetros en los mensajes, valores devueltos, mensajes enviados sólo si se cumple una determinada condición, o mensajes que se ejecutan varias veces (iteraciones).

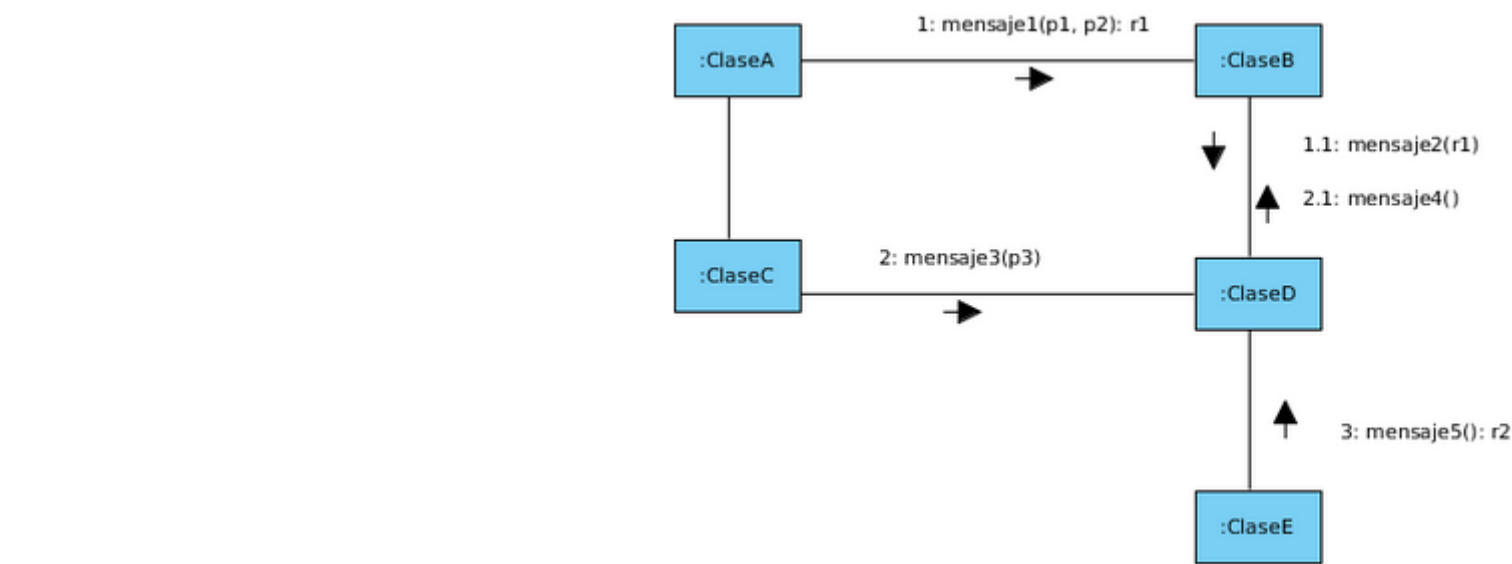
La **sintaxis de un mensaje** es la siguiente:

[Secuencia] [\*] [Condición] {valorDevuelto} : mensaje (argumentos de entrada) o  
[Secuencia] [\*] [Condición] mensaje (argumentos de entrada) : {valorDevuelto}

Donde:

- **Secuencia:** representa el nivel de anidamiento del envío del mensaje dentro de la interacción. Los mensajes se numeran para indicar el orden en el que se envían, y si es necesario se puede indicar anidamiento incluyendo subrangos.
- **\***: indica que el mensaje es iterativo.
- **Condición:** debe cumplirse para que el mensaje pueda ser enviado.
- **ValorDevuelto:** lista de valores devueltos por el mensaje. Estos valores se pueden utilizar como parámetros de otros mensajes.
- **Mensaje:** nombre del mensaje.
- **Argumentos:** parámetros que se pasan al mensaje.

Como se ve en el ejemplo, se puede usar la misma asociación para enviar varios mensajes. Vemos que hay dos mensajes anidados, el 1.1 y el 2.1, se ha usado el nombre de los mensajes para indicar el orden real en el que se envían.



Los mensajes 1, 1.1 y 2 tienen parámetros y los mensajes 1 y 3 devuelven un resultado.



### 3.2. Ejemplo de un diagrama de colaboración.

A continuación se muestra un diagrama de colaboración de ejemplo.

