

# Diagramas de interacción



# Diagramas de interacción

Los diagramas de interacción ilustran cómo interaccionan unos objetos con otros, intercambiando mensajes.

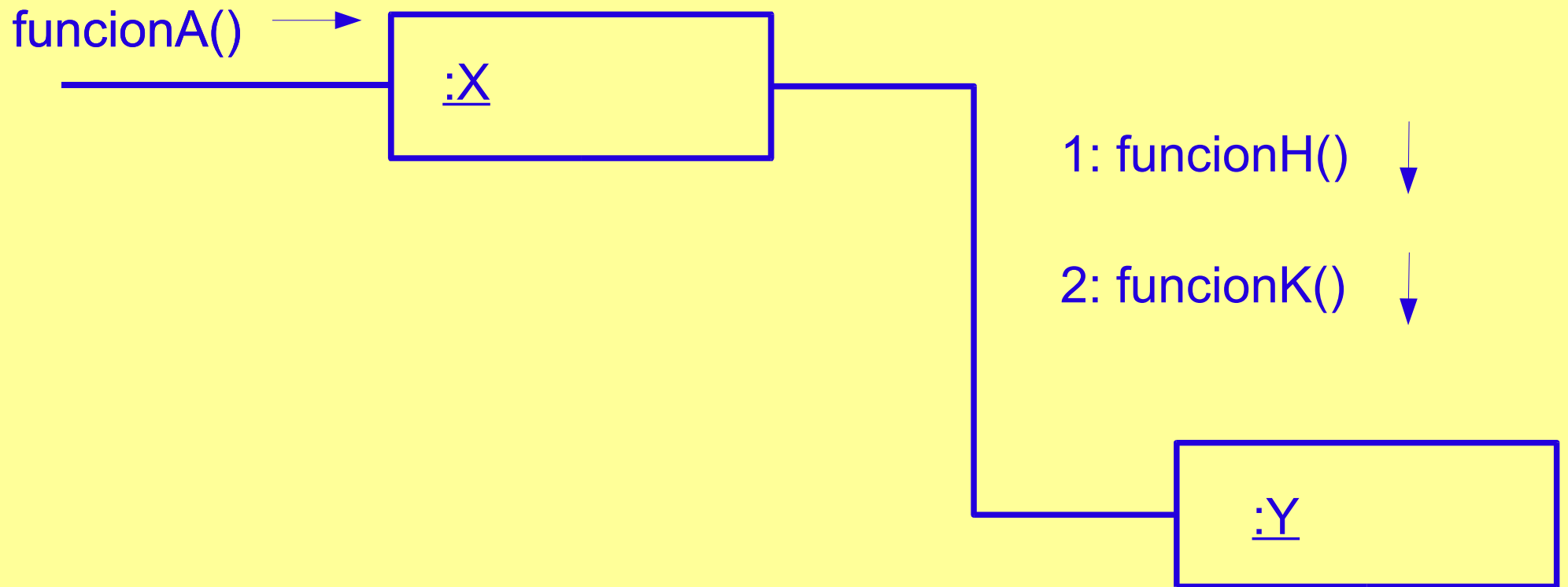
Leer cap. 15 de  
“UML y patrones”  
Craig Larman

# Diagramas de interacción

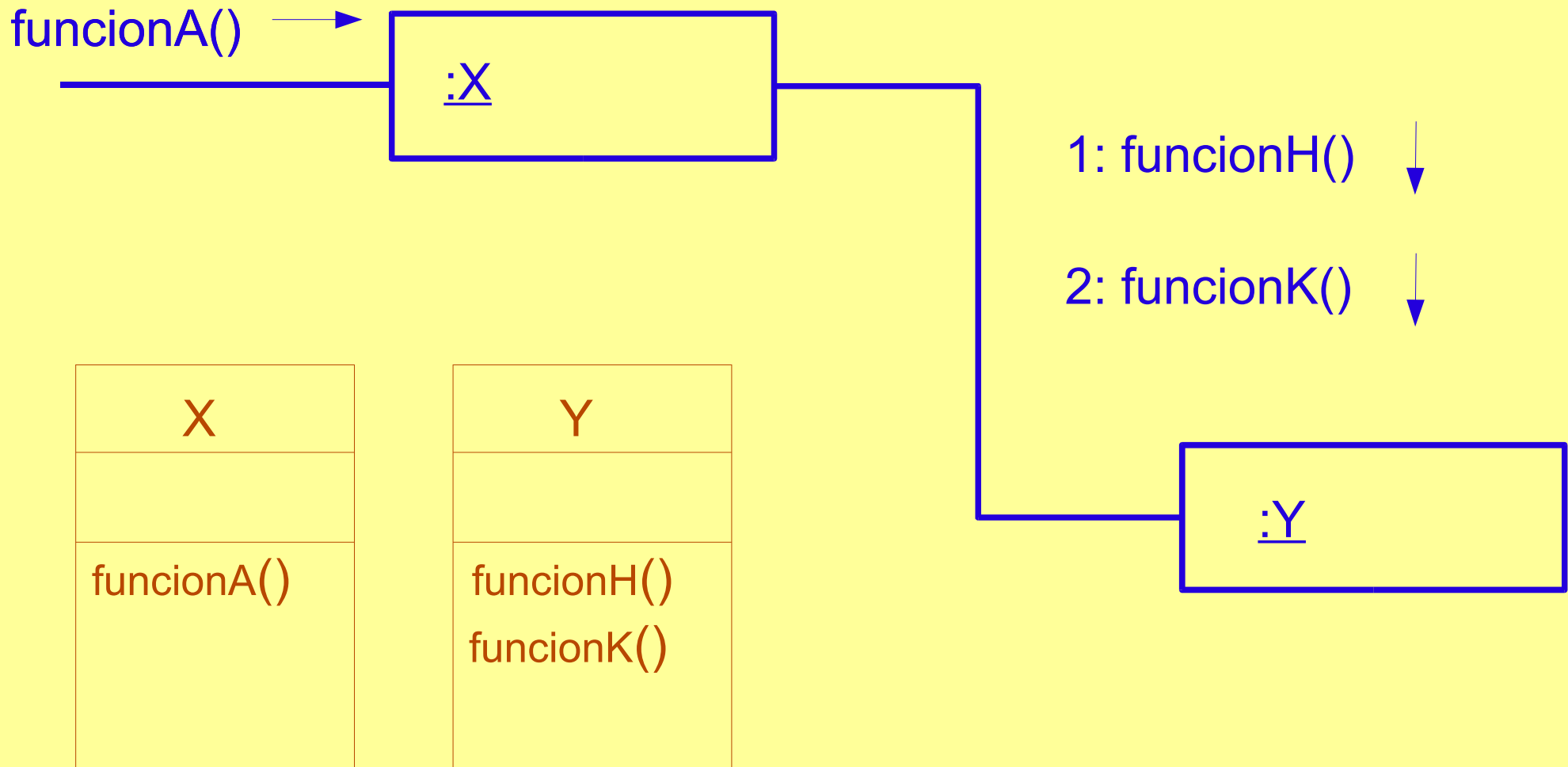
## Tipos

<i>tipo</i>	<i>ventajas</i>	<i>inconvenientes</i>
colaboración	<ul style="list-style-type: none"><li>- Elaboración <i>flexible</i></li><li>- ilustran mejor condicionales, iterativas, concurrentes.</li></ul>	<ul style="list-style-type: none"><li>- Notación compleja</li><li>- ilustran peor secuencias</li></ul>
secuencia	<ul style="list-style-type: none"><li>- Notación simple</li><li>- Ilustran mejor secuencias</li></ul>	<ul style="list-style-type: none"><li>- Elaboración <i>rígida</i></li></ul>

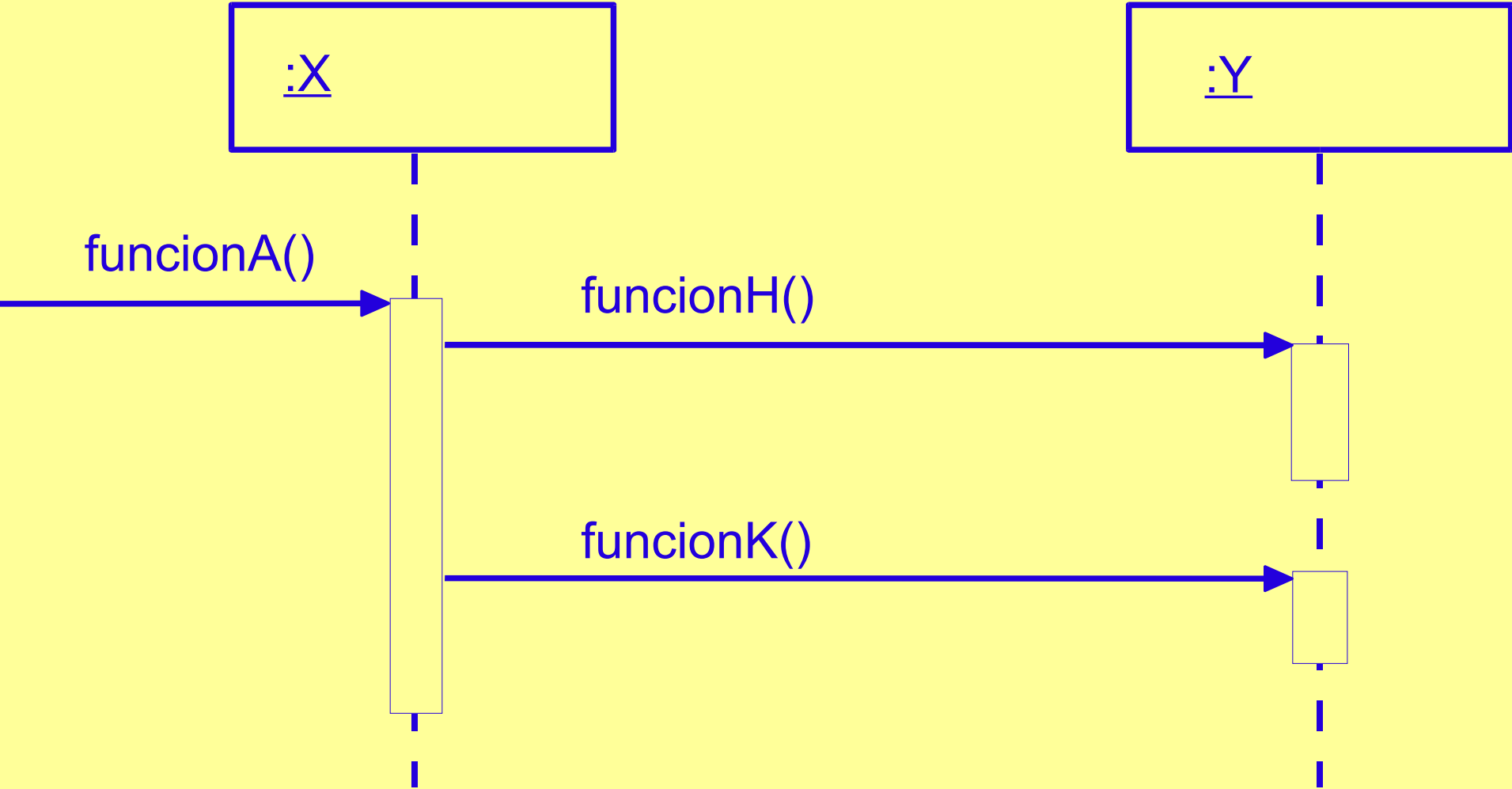
# Diagramas de colaboración



# Diagramas de colaboración



# Diagramas de secuencia



# Diagramas de interacción

## Consideraciones:

- ◆ Los diagramas de interacción son importantes
- ◆ es aconsejable crearlos en colaboración con otros programadores.
- ◆ Elaborarlos implica asignar responsabilidades a los objetos:
  - ◆ ésta no es una tarea fácil
  - ◆ considerar patrones de diseño puede ser útil

# Diagramas de interacción notación

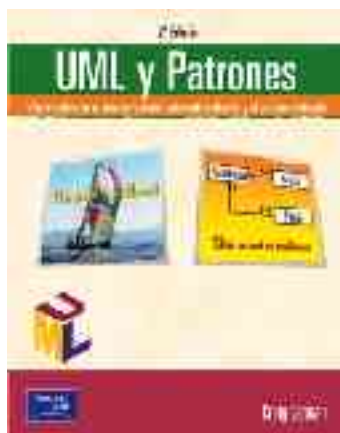
clases e instancias: como en los diagramas de clases

expresión de mensaje

```
return :=  
nombre_mensaje(parametro :tipo_par, ...) :tipo_retorno
```

enlaces: como en los diagramas de clases

Figura 15.6. Líneas de enlaces.



# Diagramas de colaboración notación

## mensajes:

- Contienen expresión de mensaje
- Indicación de sentido: mediante una flecha
- número de secuencia:
  - el primer mensaje (*el que desencadena la colaboración*) no se numera
  - los siguientes: 1 , 2 , 3
  - con anidamiento 1.1, 1.2, ..
- Pueden aparecer mensajes a *this*

Figura 15.7. Mensajes.

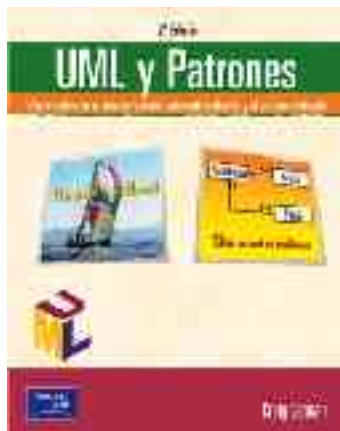
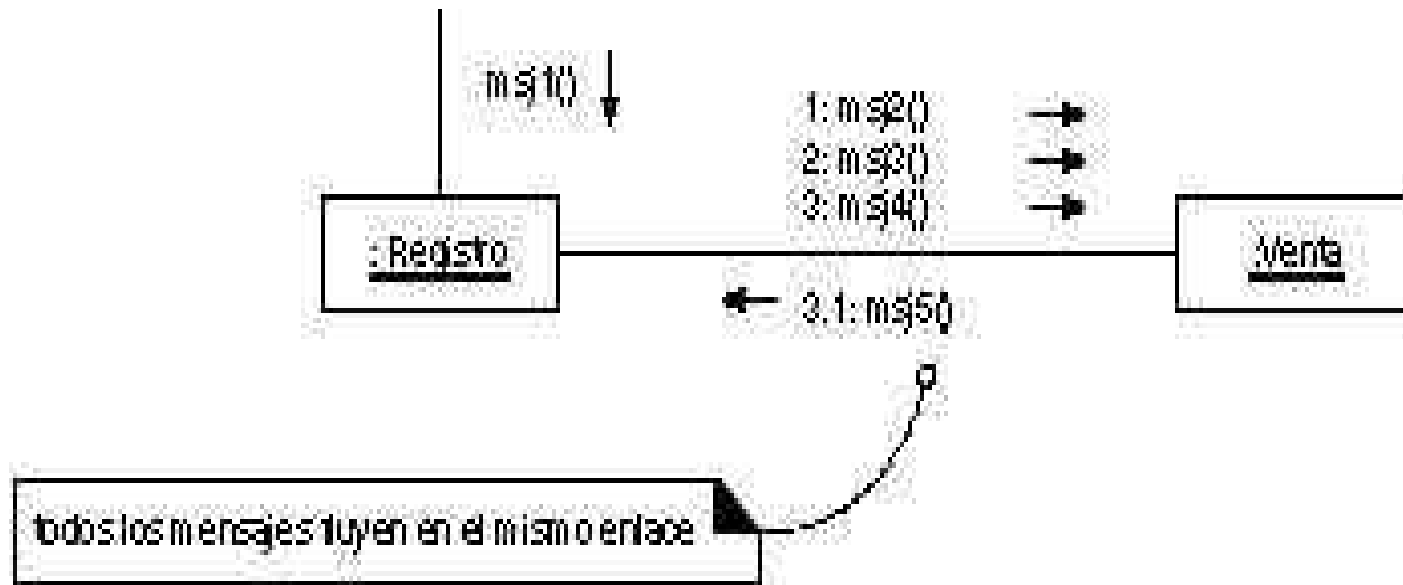


Figura 15.8. Mensajes a “this”.

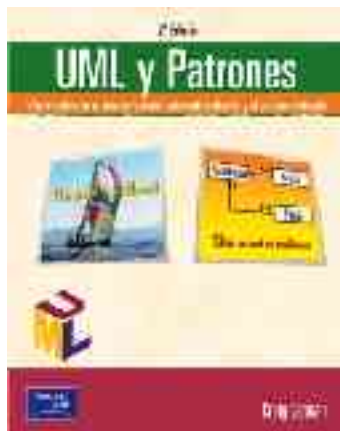


Figura 15.10. Secuencia de numeración.

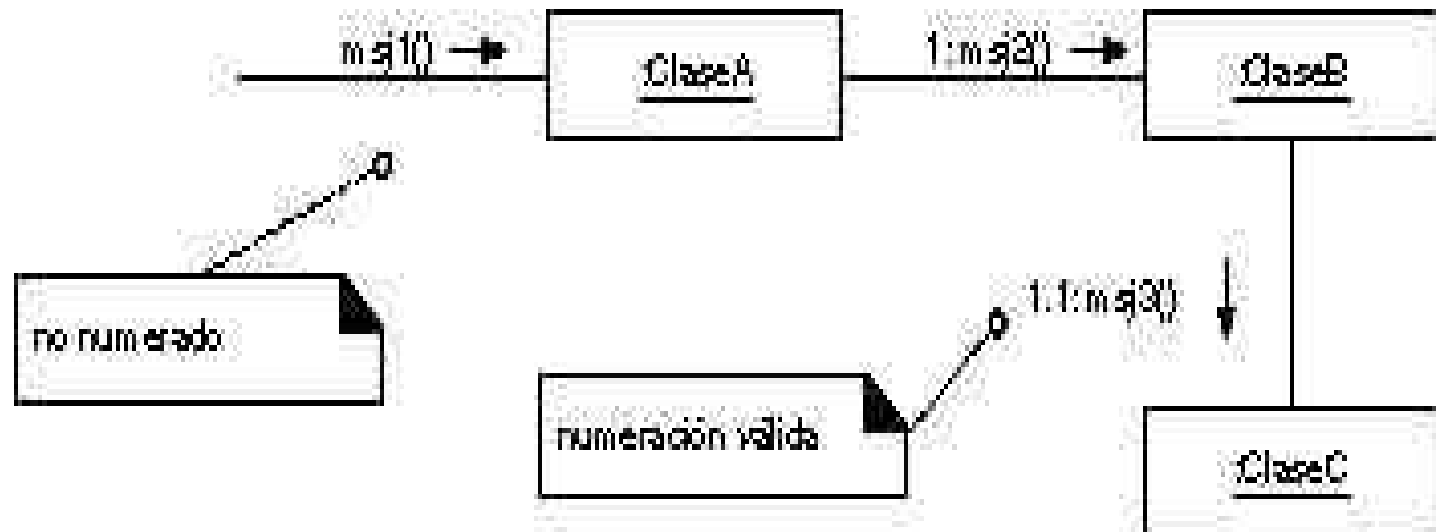
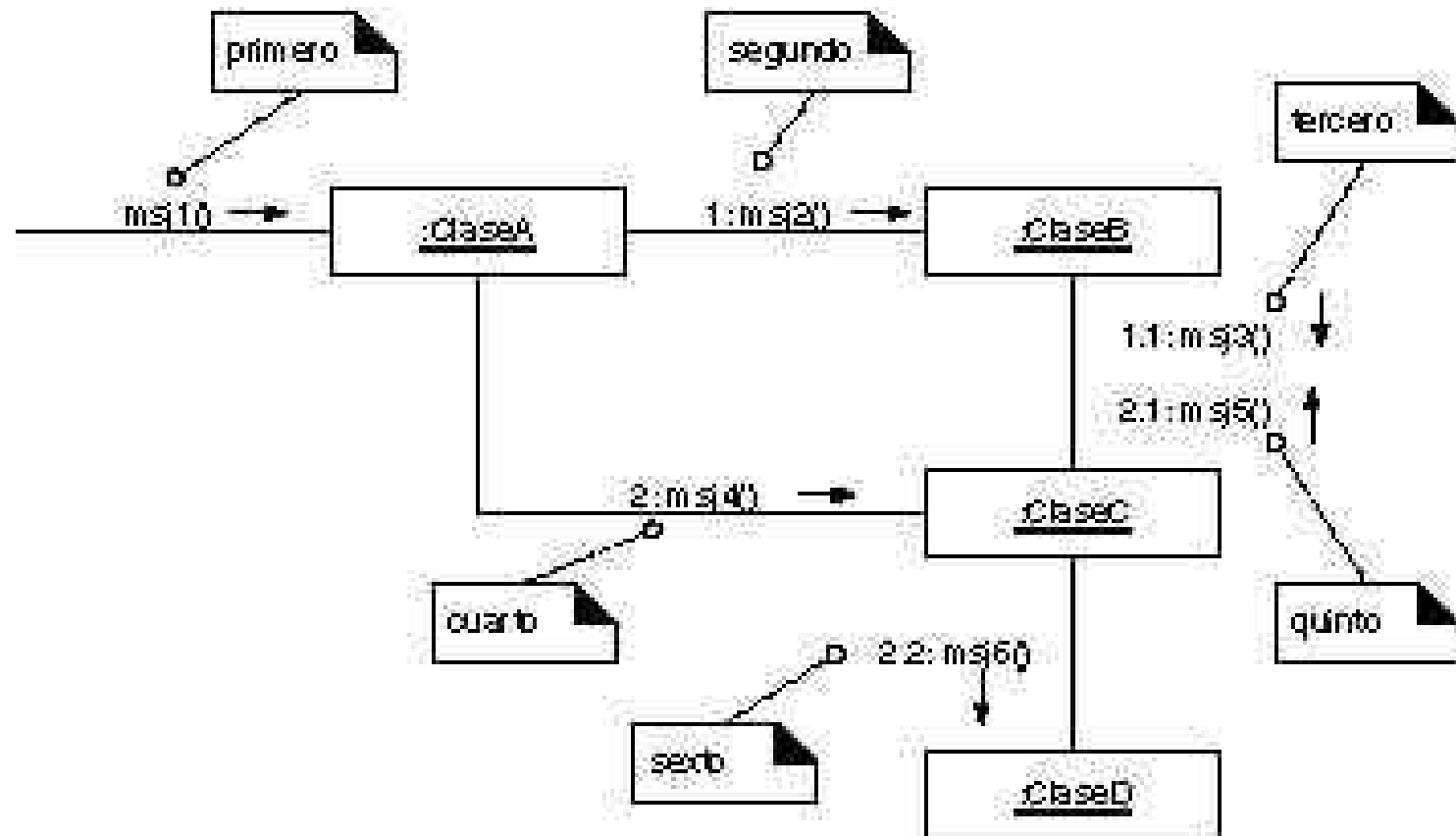


Figura 15.11. Secuencia de numeración compleja.



# Diagramas de colaboración notación

- ◆ mensajes de creación de instancias: *create*
- ◆ mensajes condicionales: anteponer [condicion]
- ◆ caminos condicionales en exclusión mutua: 2a, 2b, 2c, ...
  - ◆ iteración: \* (puede indicarse cláusula de iteración )
    - ◆ iteraciones sobre colecciones
      - ◆ mensajes a clases

Figura 15.9. Creación de instancias.

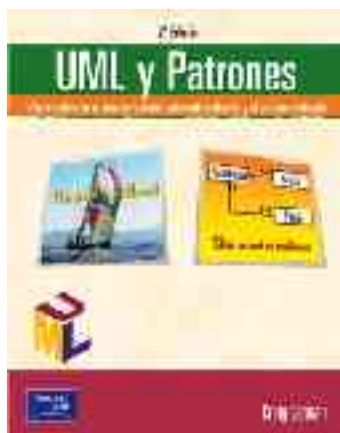
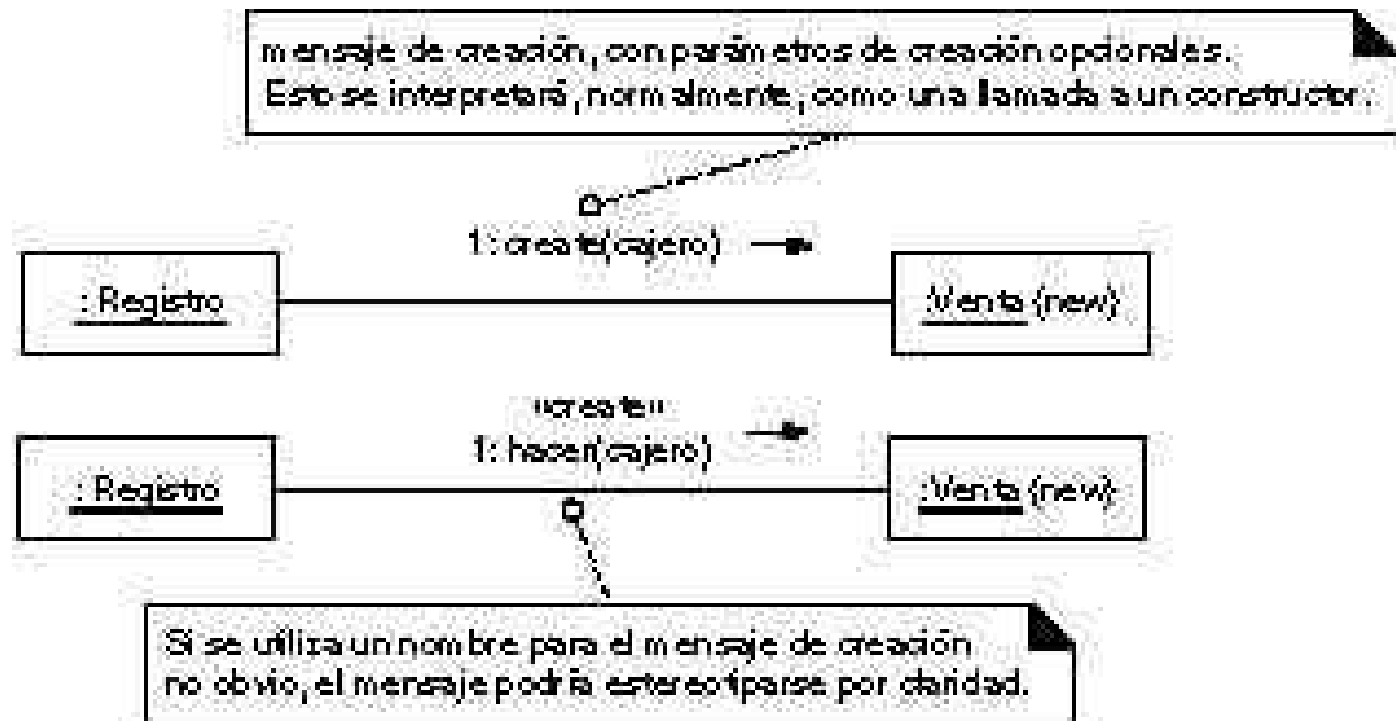


Figura 15.12. Mensaje condicional.

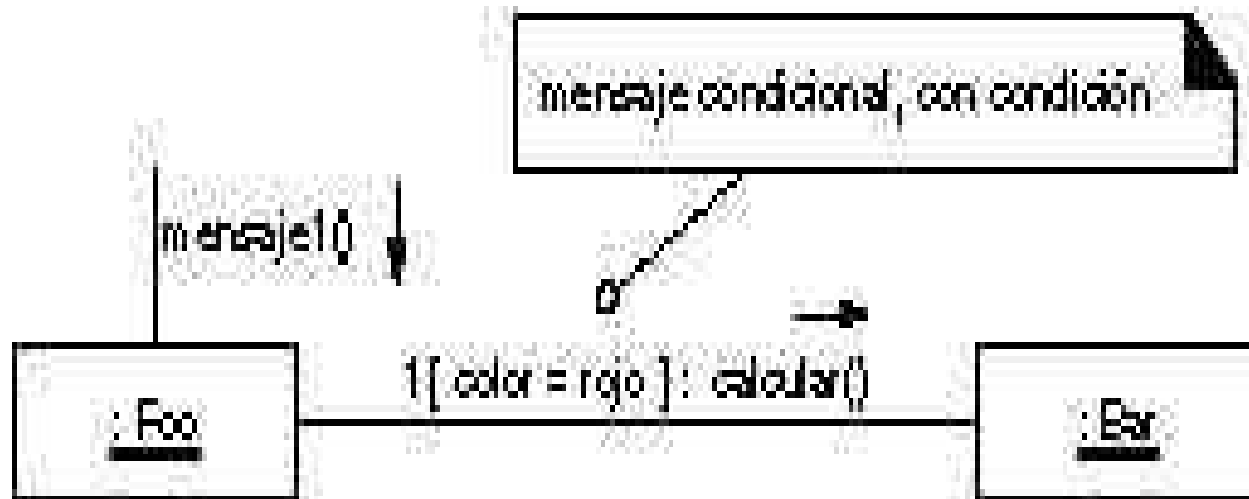


Figura 15.13. Mensajes mutuamente exclusivos.

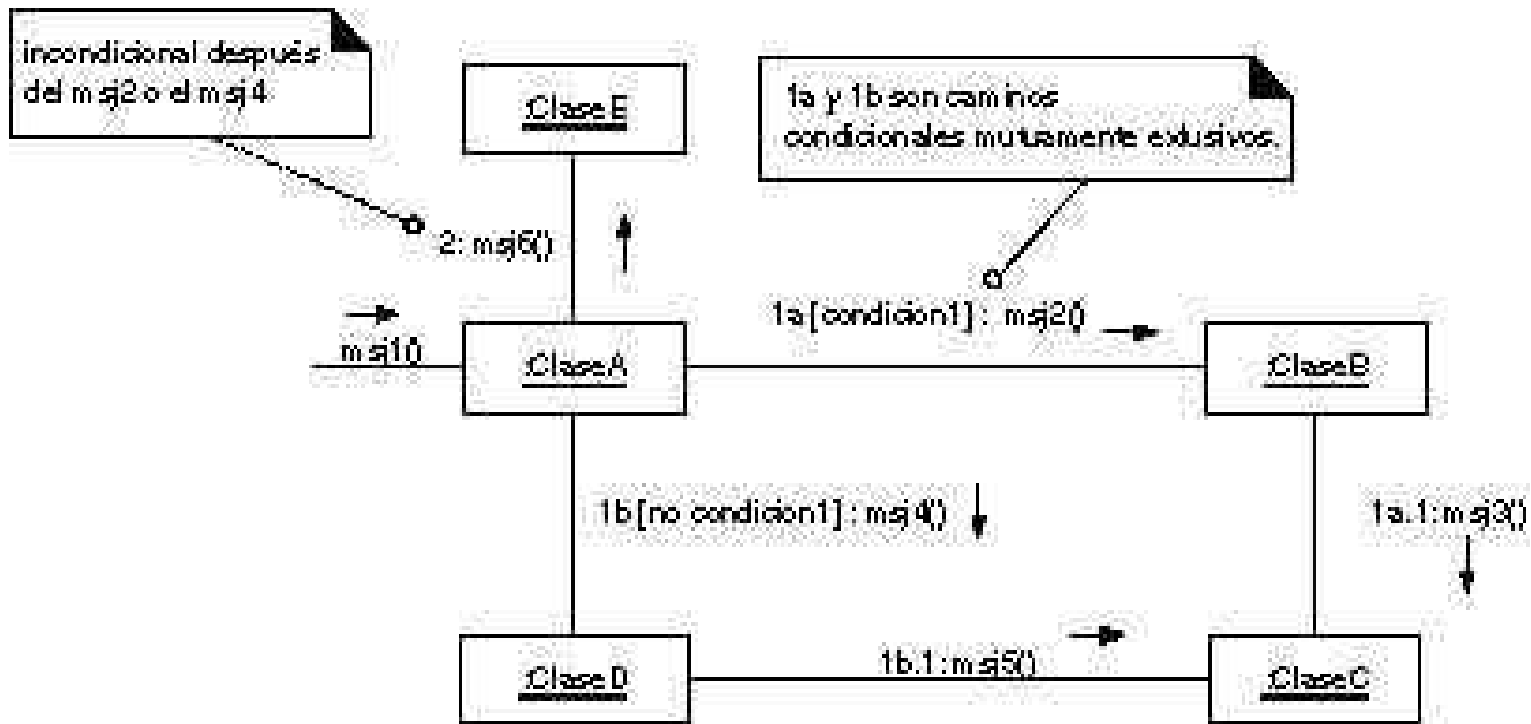


Figura 15.14. Iteración.

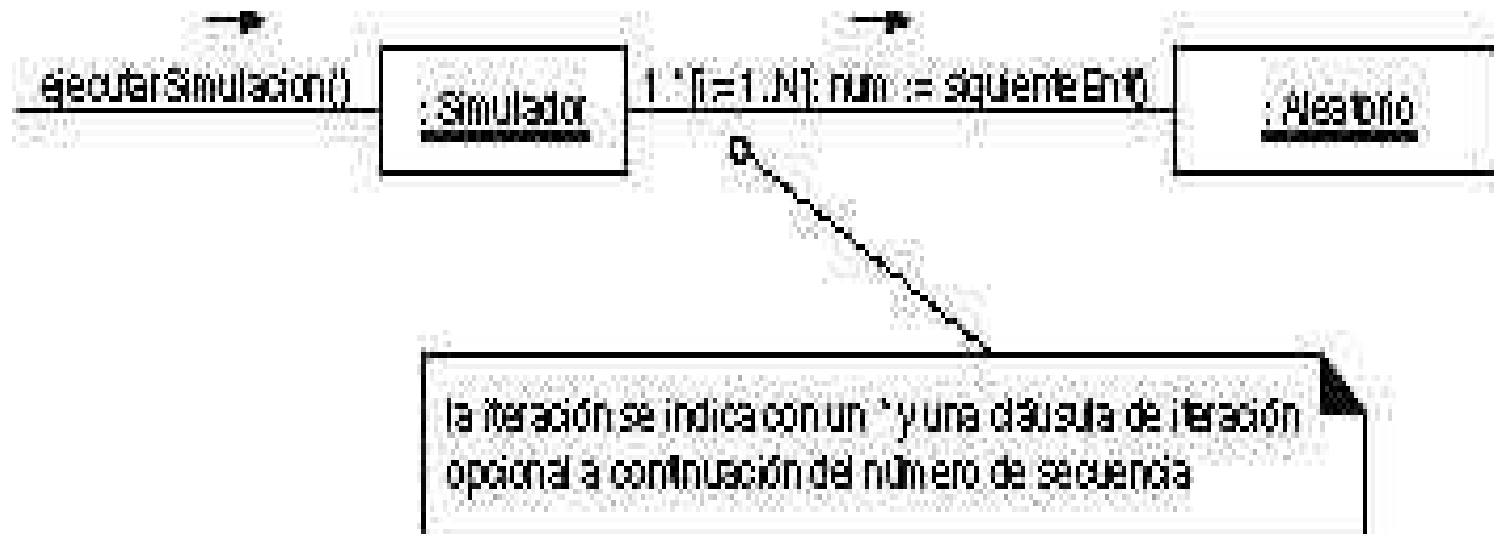
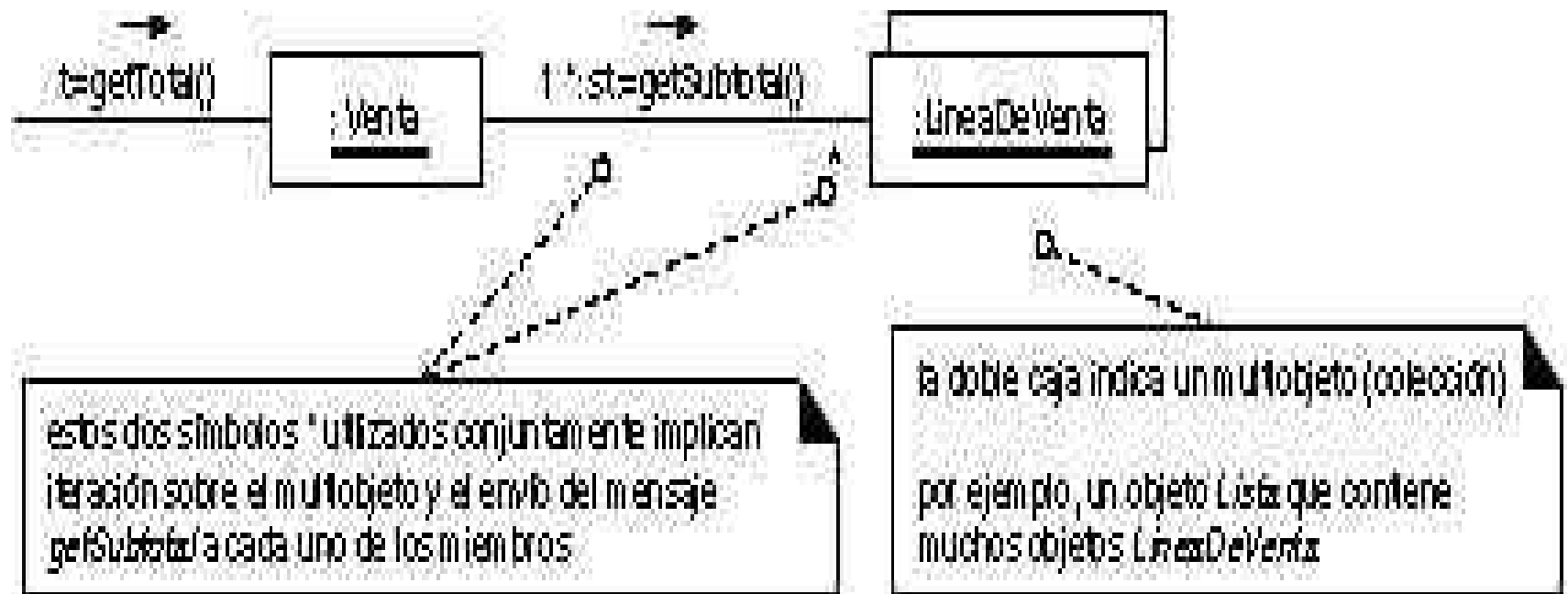


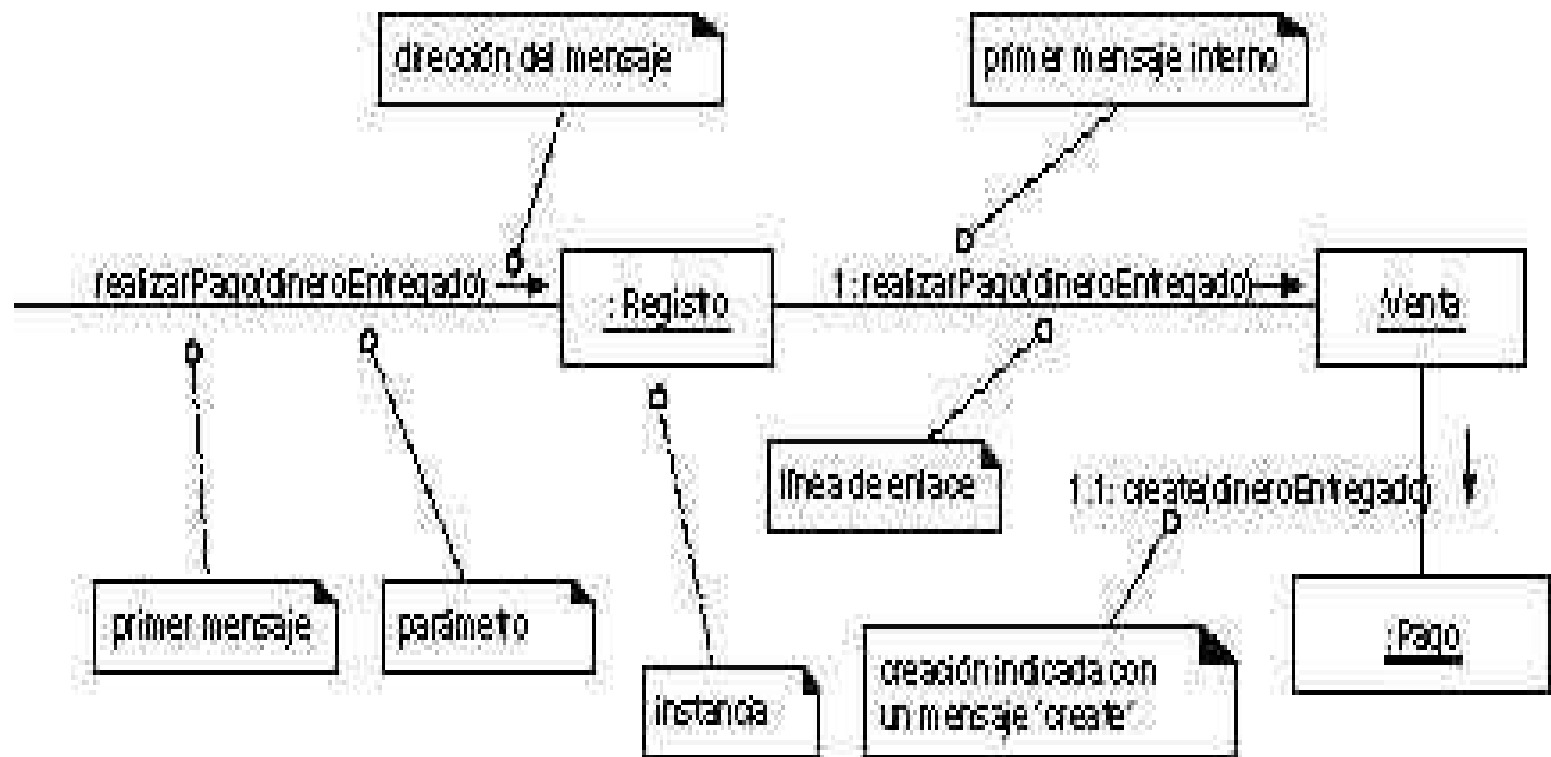
Figura 15.15. Iteración sobre un multiobjeto.



**Figura 15.16.** Mensaje a un objeto clase (invocación de un método estático).



Figura 15.3. Diagrama de colaboración.



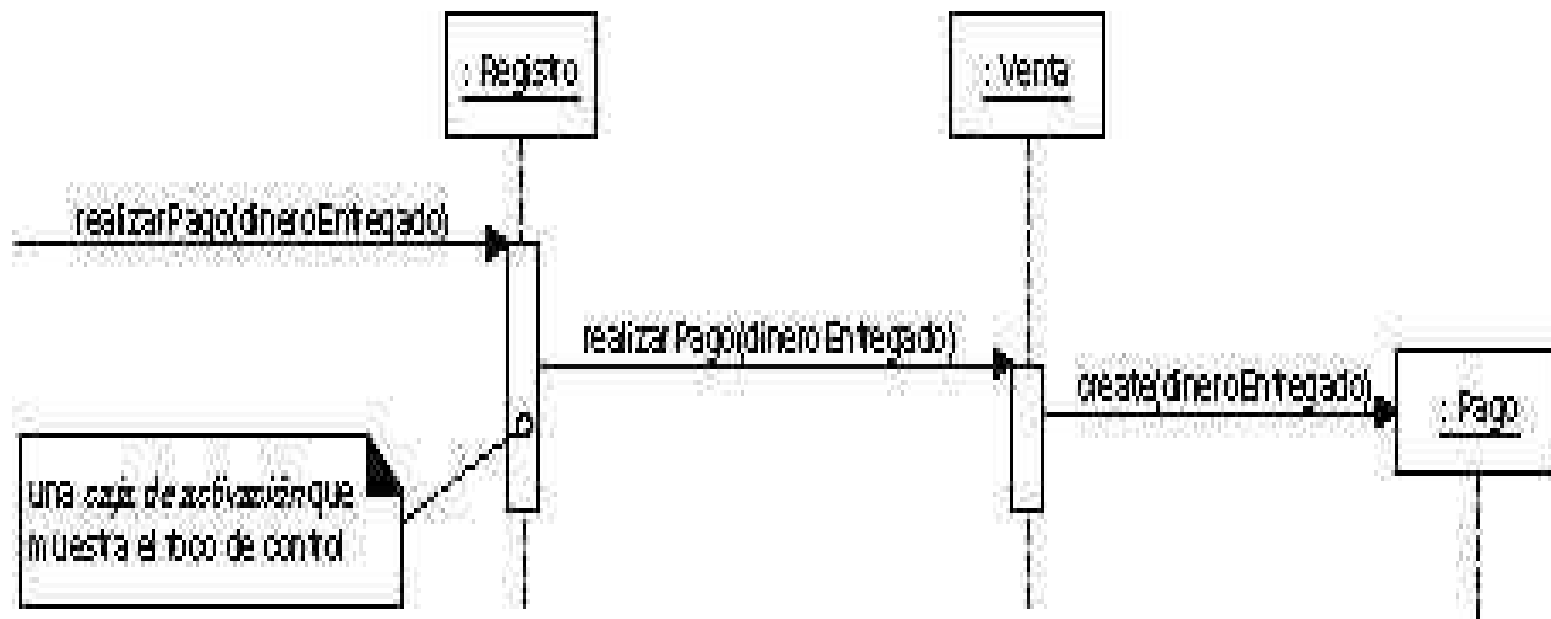
# Diagramas de secuencia notación

♦ enlaces: no aparecen

♦ mensajes:

- ♦ una flecha entre las líneas verticales de los objetos correspondientes.
- ♦ La posición de estas líneas (arriba o abajo) indica orden temporal.

Figura 15.4. Diagrama de secuencia.



# Diagramas de secuencia notación

- ◆ Focos de control y cajas de activación
- ◆ Pueden aparecer mensajes a this
- ◆ creación de instancias
- ◆ líneas de vida
- ◆ destrucción de objetos
- ◆ mensajes condicionales
- ◆ mensajes condicionales en exclusión mútua
- ◆ mensajes a clases

Figura 15.17. Mensajes y focos de control con cajas de activación.

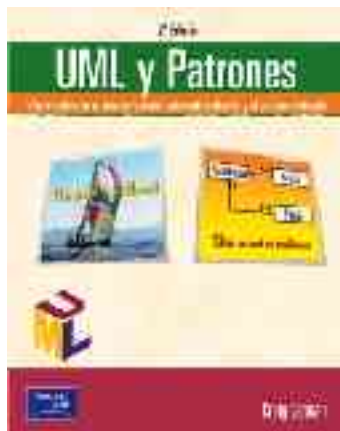
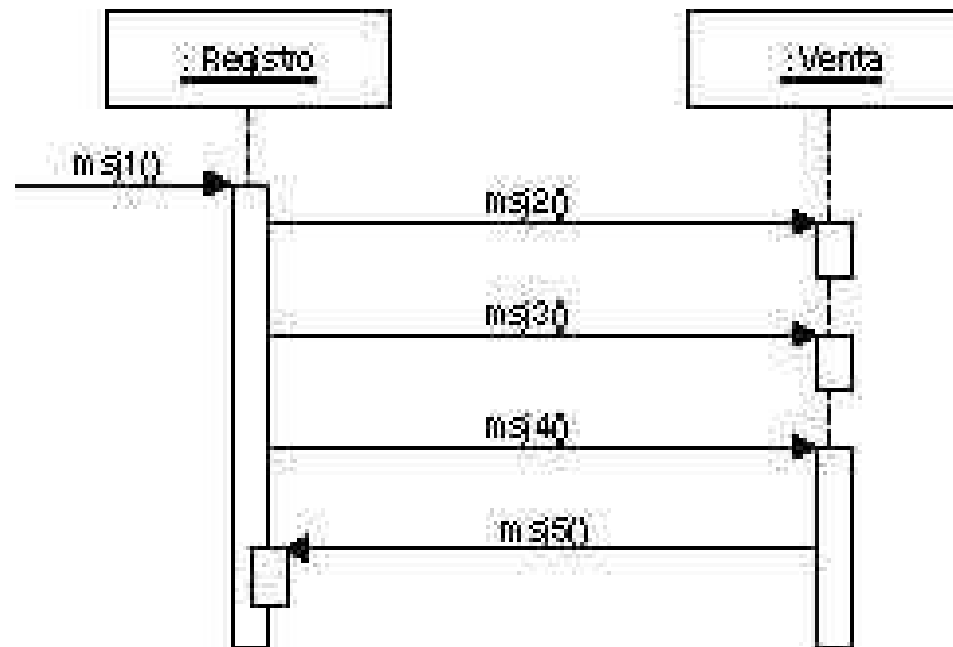
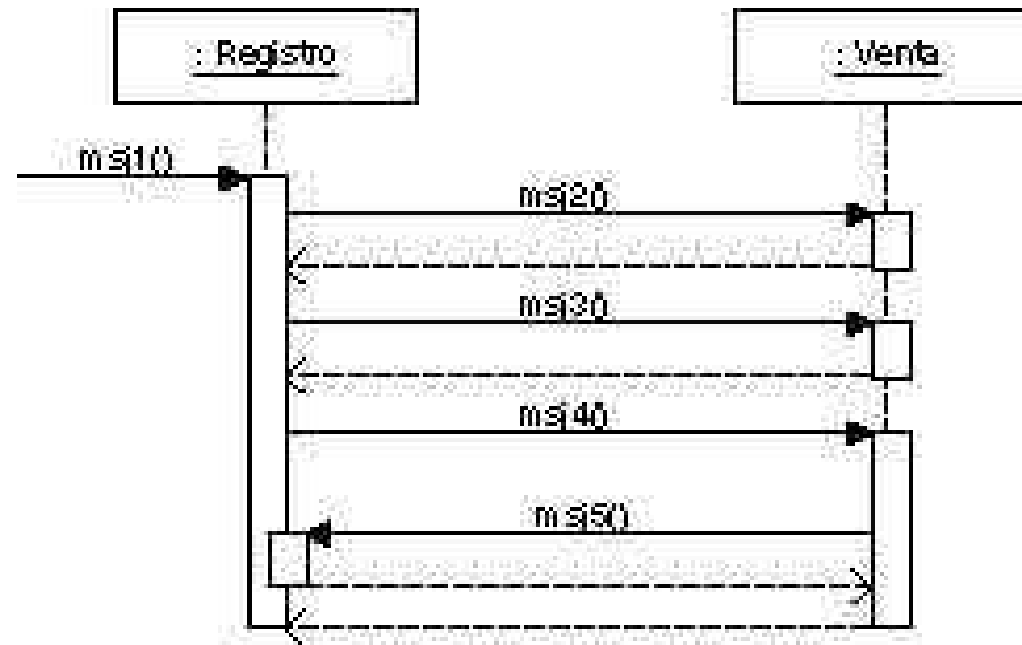


Figura 15.18. Representación de retornos.



**Figura 15.19.** Mensajes a “this”.

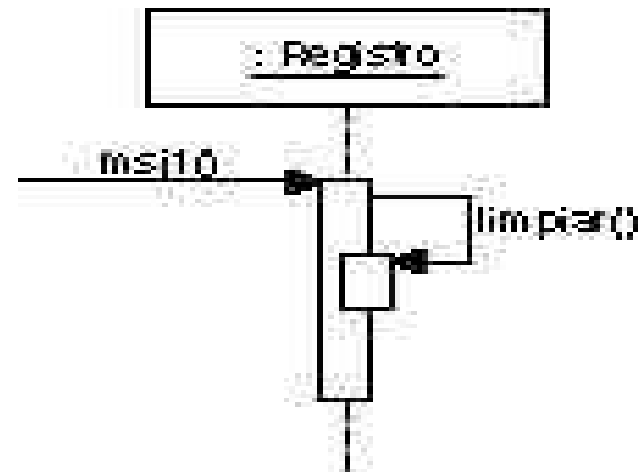


Figura 15.20. Creación de instancias y línea de vida de los objetos.

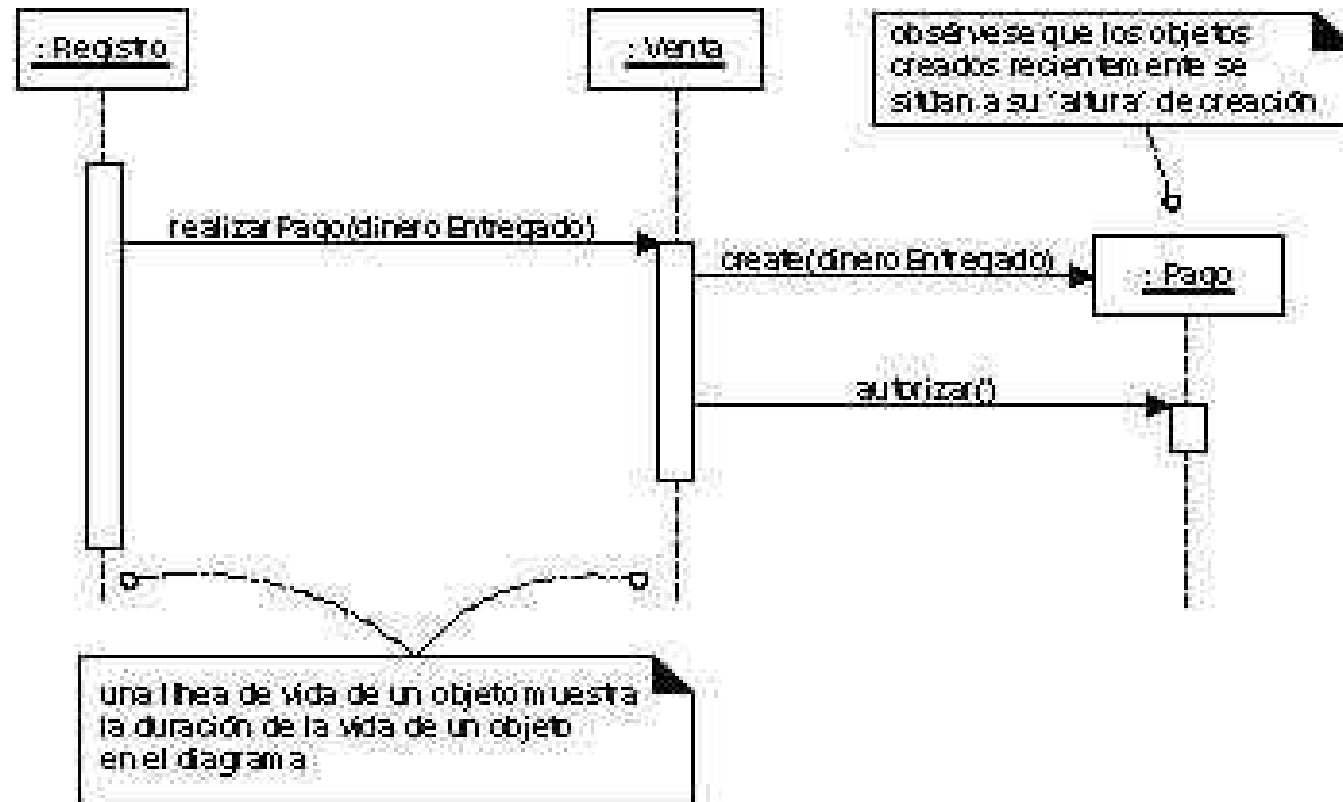
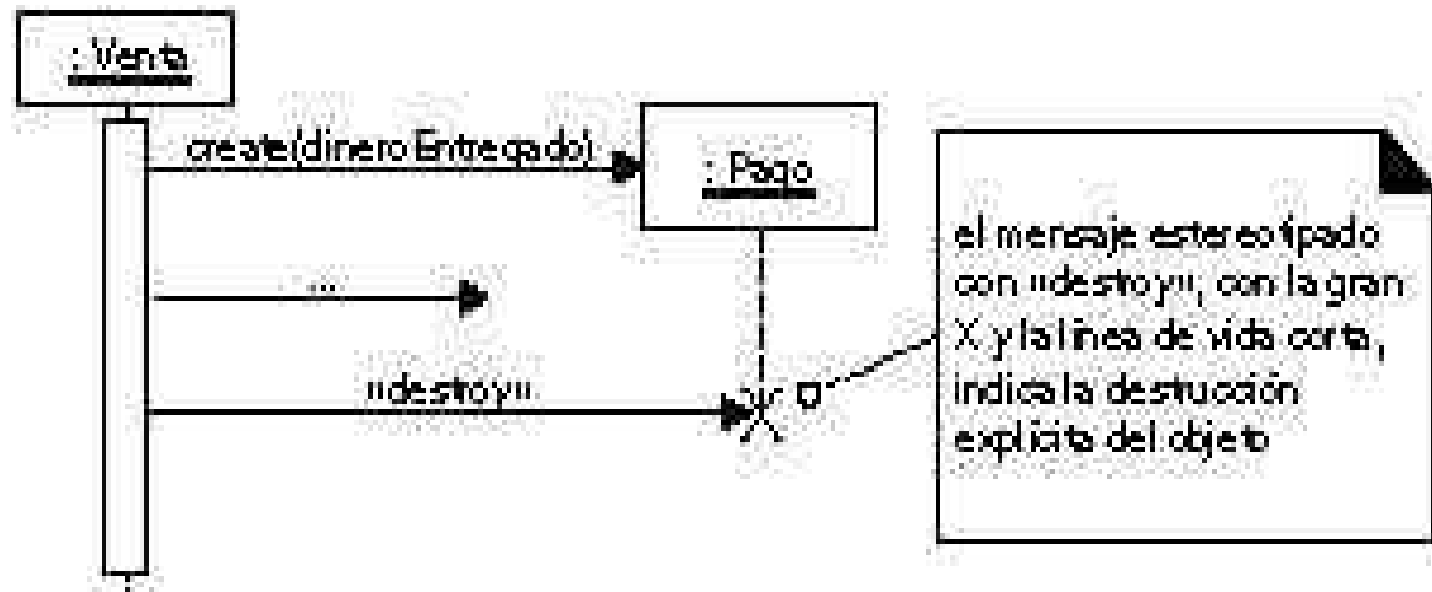
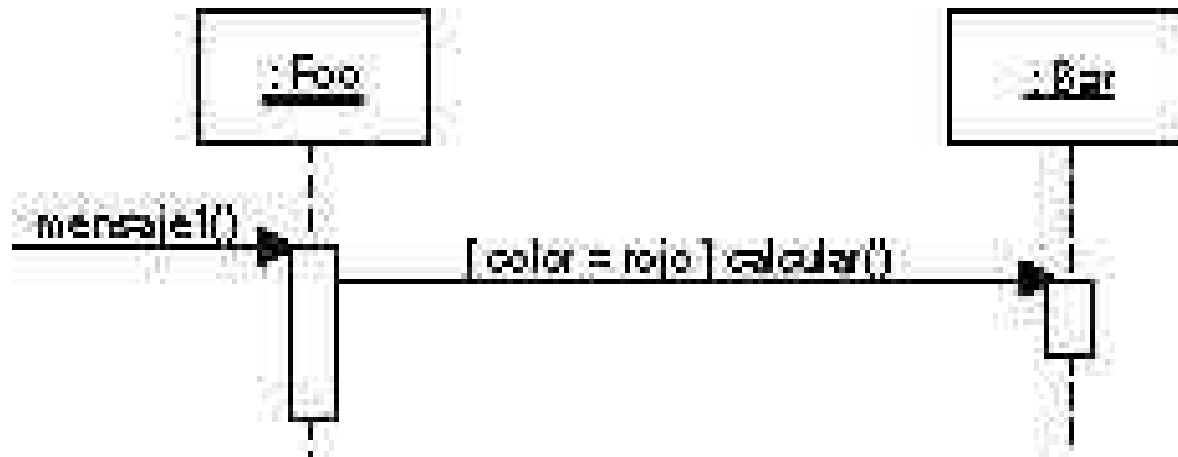


Figura 15.21. Destrucción de objetos.



**Figura 15.22.** Un mensaje condicional.



**Figura 15.23.** Mensajes condicionales mutuamente exclusivos.

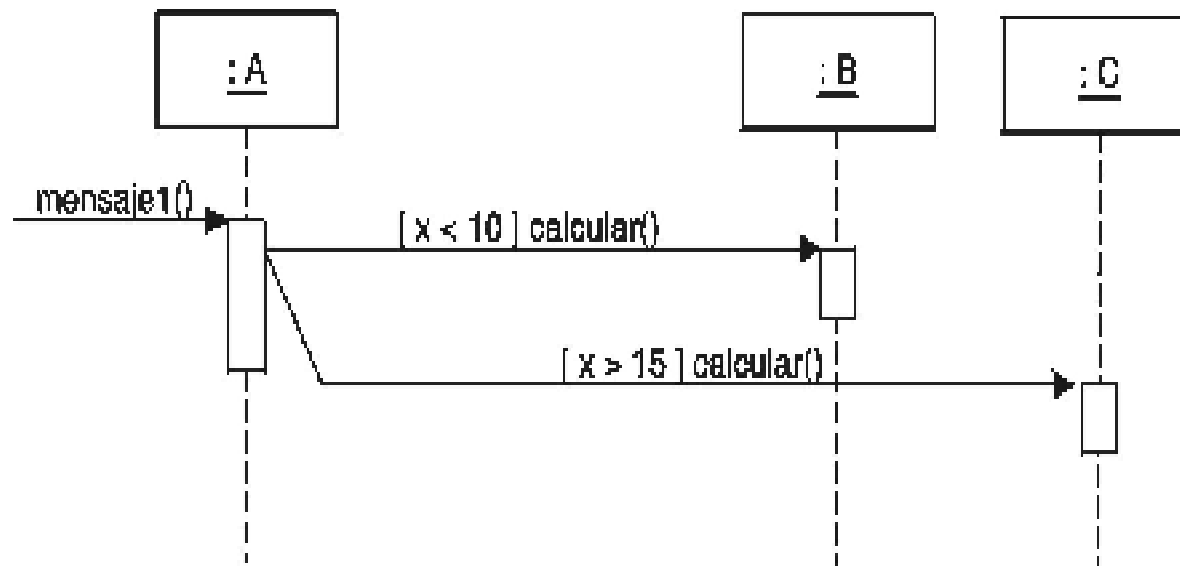


Figura 15.24. Iteración para un mensaje.

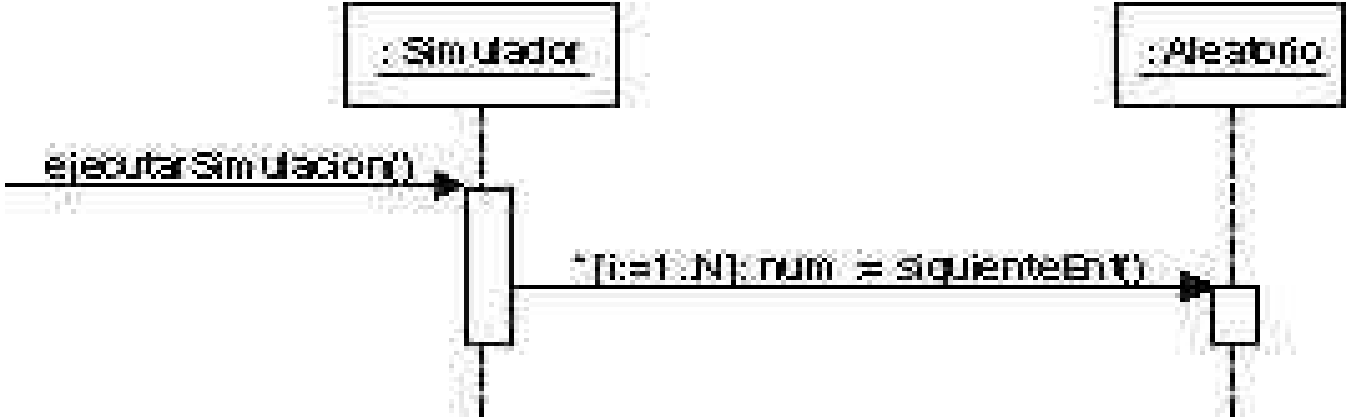


Figura 15.25. Iteración sobre una secuencia de mensajes.

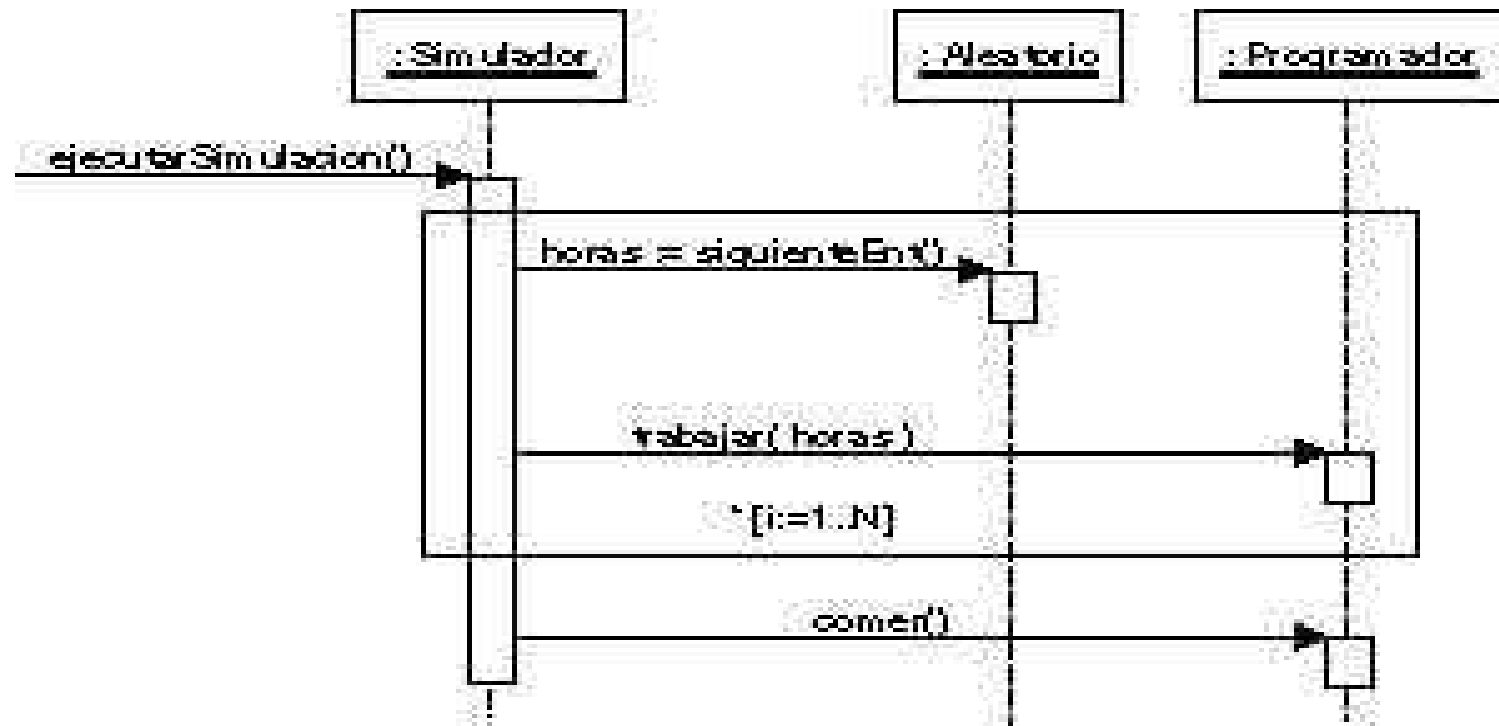


Figura 15.26. Iteración sobre un multiobjeto.

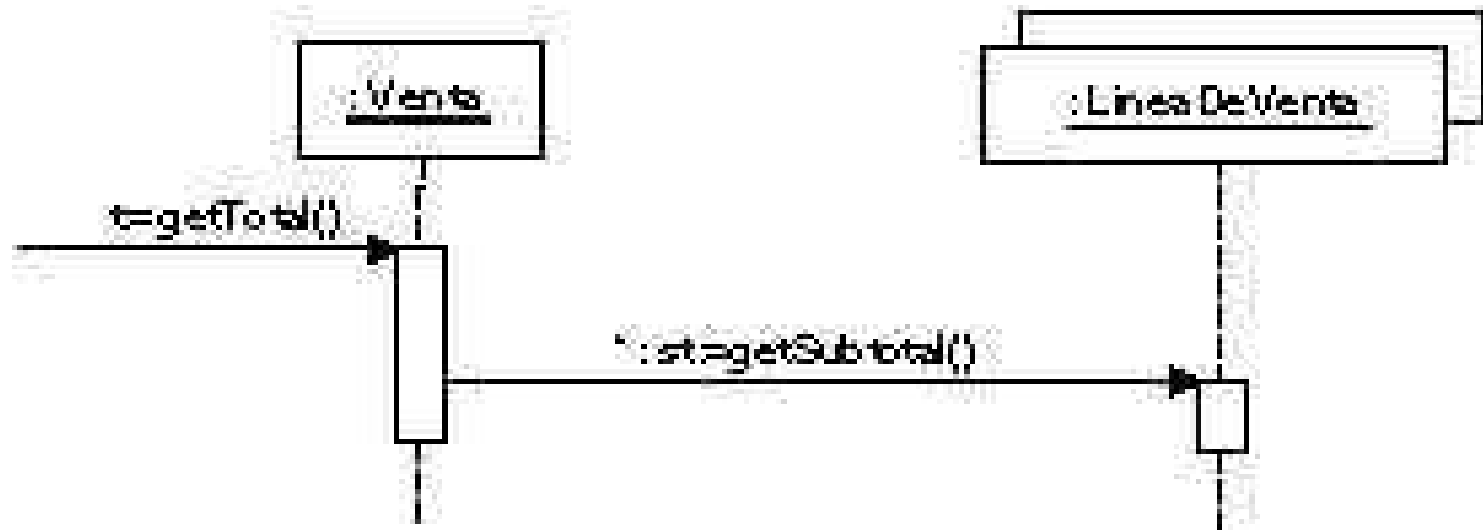


Figura 15.27. Invocación a un método de clase o estático.

