

An antique globe with a brass frame and a map of the world is positioned on the left side of the slide. The globe is tilted and has a brass ring around its equator.

Tema 6 UML – Lenguaje de Modelado Unificado

Elaboración de diagramas del comportamiento



1. Introducción

- Los **diagramas de comportamiento** muestran las características de comportamiento de un sistema o proceso de negocio. Incluyen los siguientes tipos:

DIAGRAMA	RESUMEN
Diagrama de casos de uso	Describe el comportamiento del sistema desde el punto de vista de un usuario/sistema que interactúa con él
Diagrama de actividad	Parecido a los diagramas de flujo, muestra los pasos, puntos de decisión y bifurcaciones. Son útiles para modelar el flujo en un caso de uso o entre casos de uso
Diagrama de estado	Muestra el conjunto de estados por los cuales pasa un objeto y cómo se producen las transiciones de un estado a otro
Diagrama de secuencia	Muestra cómo interactúan unos objetos con otros
Diagrama de comunicación	Muestra las interacciones entre los elementos en tiempo de ejecución
Diagrama de tiempos	Definen el comportamiento de diferentes objetos dentro de una escala de tiempo
Diagrama de vista de interacción	Muestra la cooperación entre otros diagramas de interacción



2. Diagramas de Casos de Uso

- Los **diagramas de Casos de Uso** describen lo que hace un sistema, enfatizando el **qué** en vez del cómo.
- **Describen las funcionalidades del sistema** a partir de las interacciones del usuario.
Es decir, describen un uso del sistema y cómo este interactúa con el usuario.
- Se emplean para visualizar el comportamiento del sistema.



2. Diagramas de Casos de Uso

Elementos del diagrama de casos de uso

- Los actores.
- Los casos de uso.
- Las relaciones.
- Puede aparecer un rectángulo que muestre los límites del sistema.

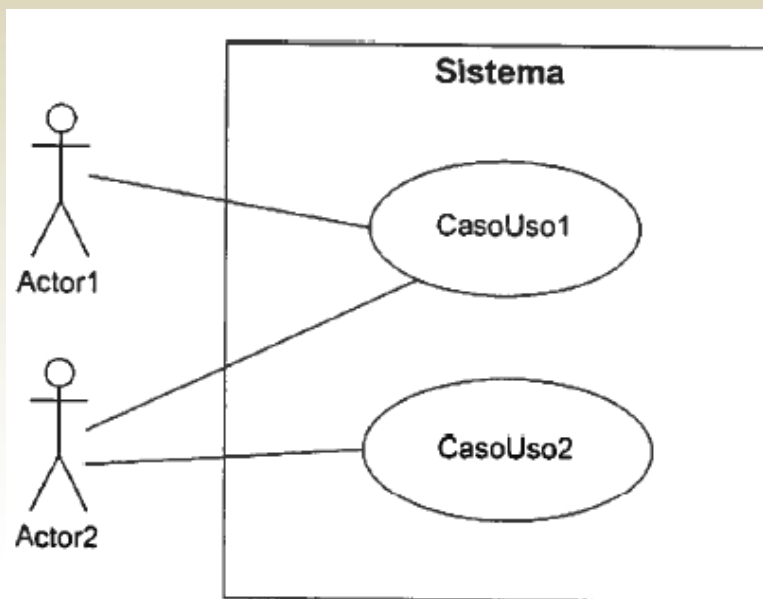


Figura 6.1. Diagrama de caso de uso.



2. Diagramas de Casos de Uso

- **Ejemplo 1.** Caso de uso que representa a dos actores interactuando con una máquina de refrescos. El cliente que compra una bebida y el reponedor que abastece de bebidas a la máquina.

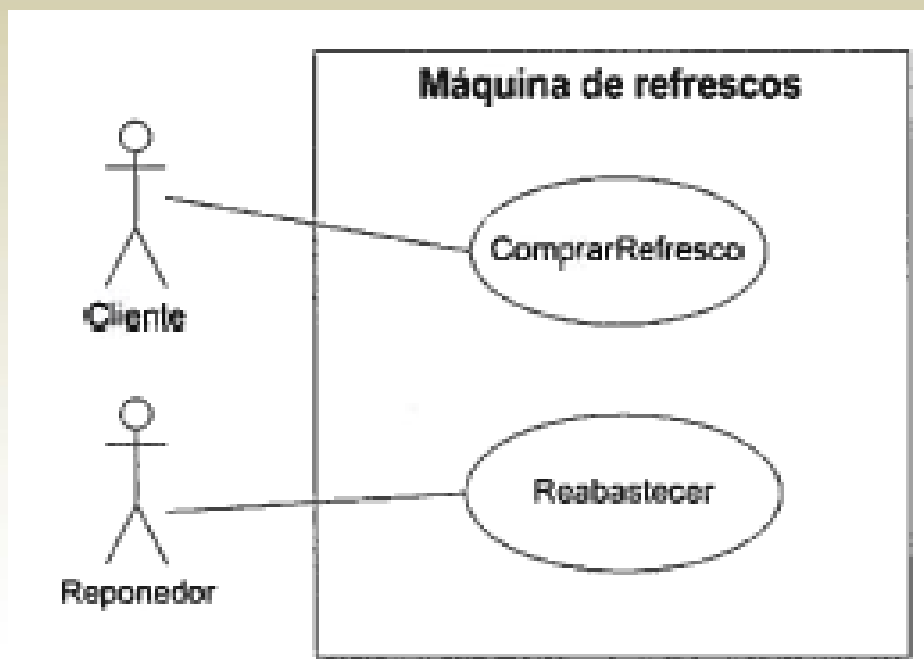


Figura 6.2. Ejemplo 1.



2. Diagramas de Casos de Uso

- **Ejemplo 2.** Caso de uso que representa a dos actores interactuando con un sistema de venta de productos. El cliente interactúa con el sistema para buscar y comprar productos. El administrativo interactúa con el sistema para insertar nuevos productos.

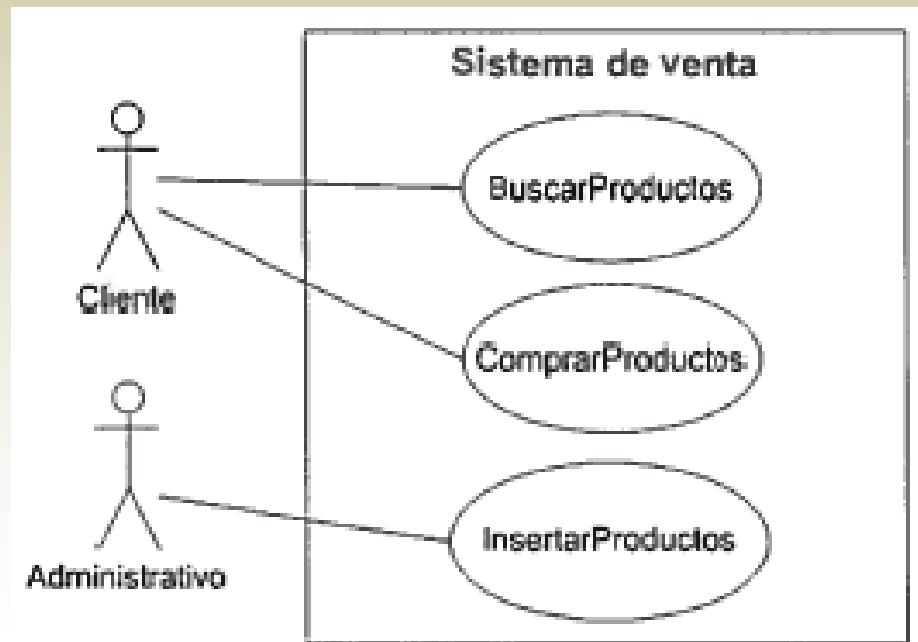


Figura 6.3. Ejemplo 2.



2. Diagramas de Casos de Uso

6.2.2 Identificar Actores

- **Entidad externa** que interactúa con el sistema.
- Entidades distintas a los usuarios de sistema.
- En algunos casos, representan cierta función que un usuario va a realizar en el sistema. Rol
- La misma persona o elemento puede interpretar varios roles

Personas

Sistema



Nombre del Actor

Componente de Software

Organización



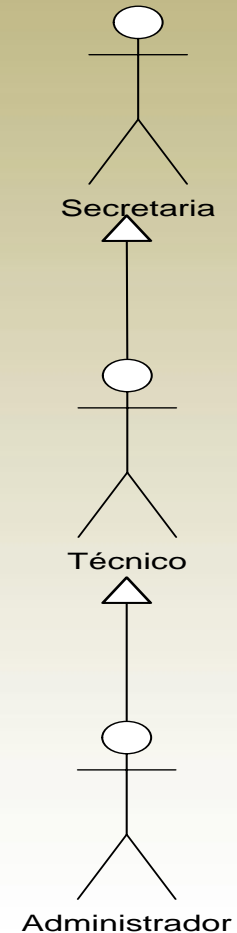
2. Diagramas de Casos de Uso

Actores

- **Relaciones entre actores:**

- **Generalización:**

Cuando diferentes actores realizan roles similares, pueden heredar de un actor común.

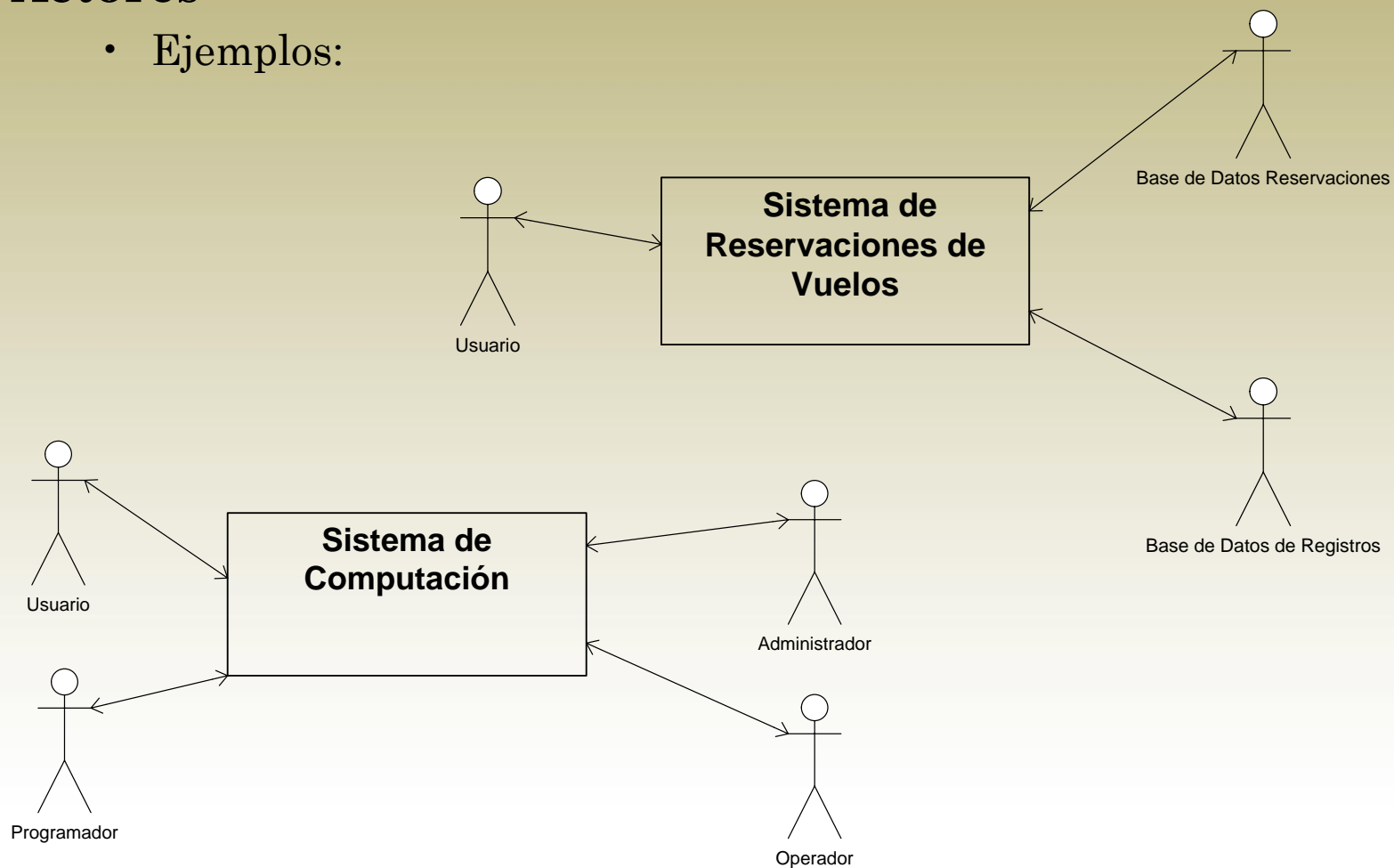




2. Diagramas de Casos de Uso

Actores

- Ejemplos:





2. Diagramas de Casos de Uso

Casos de Uso

- Un caso de uso define una **funcionalidad** del sistema.
- Es iniciado por u actor.
- Cada caso de uso constituye un **flujo de eventos**, que especifican la interacción que toma lugar entre el actor y el sistema.
- Cada caso de uso **produce** un **resultado** observable y válido para el actor involucrado en la secuencia de acciones.

Nombre del Caso de Uso

Verbos

Son Acciones

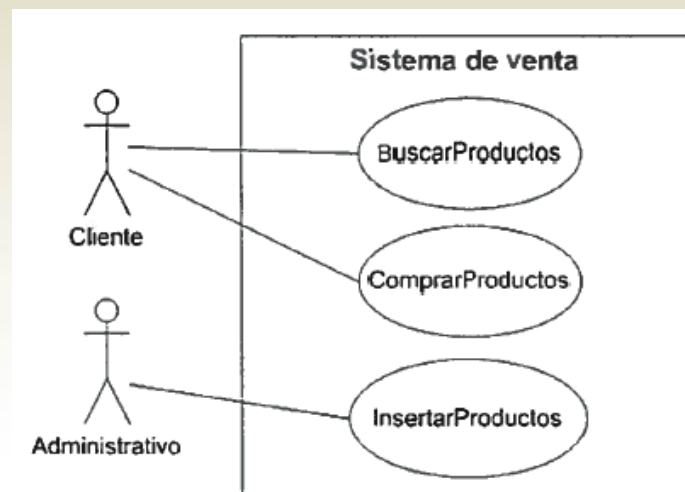


Figura 6.3. Ejemplo 2.



2. Diagramas de Casos de Uso

Relaciones entre Casos de Uso:

RELACIÓN	FUNCIÓN	NOTACIÓN
Asociación	Es la línea de comunicación entre un actor y un caso de uso en el que participa	_____
Extensión «extend»	Permite que un caso de uso extienda su comportamiento con uno o más fragmentos de comportamiento de otro. Se utiliza para especificar que el comportamiento de un caso de uso es diferente dependiendo de ciertas circunstancias. Se representa mediante una flecha discontinua con la palabra «extend». La relación apunta al caso de uso que se extenderá	— — — —>
Generalización de casos de uso	Es como la generalización entre clases. El caso de uso hijo hereda el comportamiento y el significado del caso de uso padre. Se representa mediante una flecha continua con una cabeza triangular en el caso de uso padre	—————>
Inclusión «include» o «uses»	Permite que un caso de uso base incluya el comportamiento de otro caso de uso. Se representa mediante una flecha discontinua con la palabra «include». La relación apunta al caso de uso a ser incluido	— — — —>



2. Diagramas de Casos de Uso

Relaciones entre Casos de Uso:

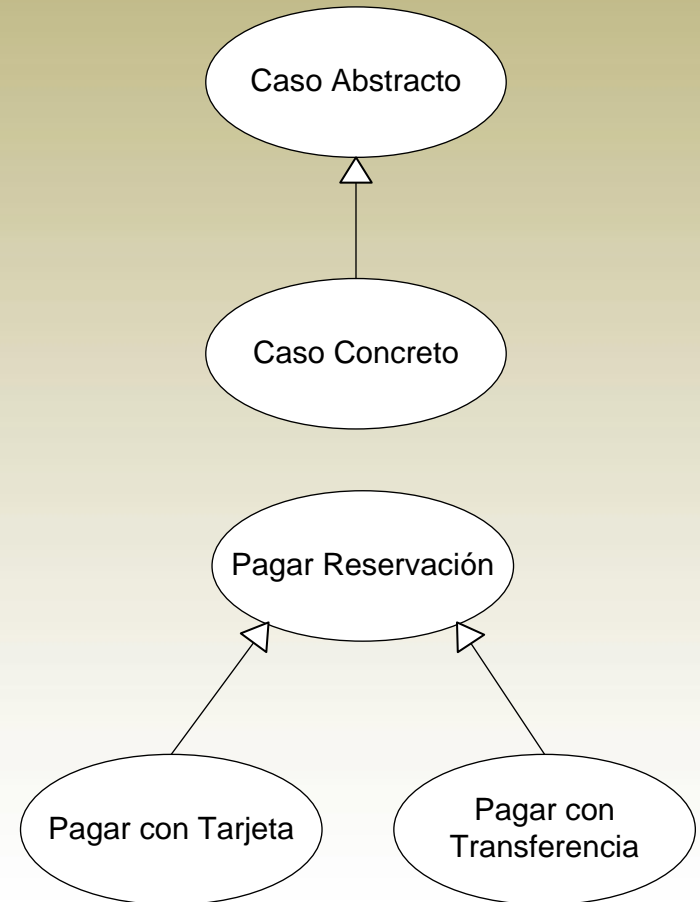
RELACIÓN	FUNCIÓN	NOTACIÓN
Asociación	Es la línea de comunicación entre un actor y un caso de uso en el que participa	_____
Extensión «extend»	Permite que un caso de uso extienda su comportamiento con uno o más fragmentos de comportamiento de otro. Se utiliza para especificar que el comportamiento de un caso de uso es diferente dependiendo de ciertas circunstancias. Se representa mediante una flecha discontinua con la palabra «extend». La relación apunta al caso de uso que se extenderá	— — — —>
Generalización de casos de uso	Es como la generalización entre clases. El caso de uso hijo hereda el comportamiento y el significado del caso de uso padre. Se representa mediante una flecha continua con una cabeza triangular en el caso de uso padre	—————>
Inclusión «include» o «uses»	Permite que un caso de uso base incluya el comportamiento de otro caso de uso. Se representa mediante una flecha discontinua con la palabra «include». La relación apunta al caso de uso a ser incluido	— — — —>



2. Diagramas de Casos de Uso

1. Generalización:

- Relación que define la especialización de un caso de uso.
- Los casos de uso **abstractos** describirán las partes similares y no podrán ser instanciados independientemente.
- Los casos de uso **concretos** describirán el comportamiento específico.

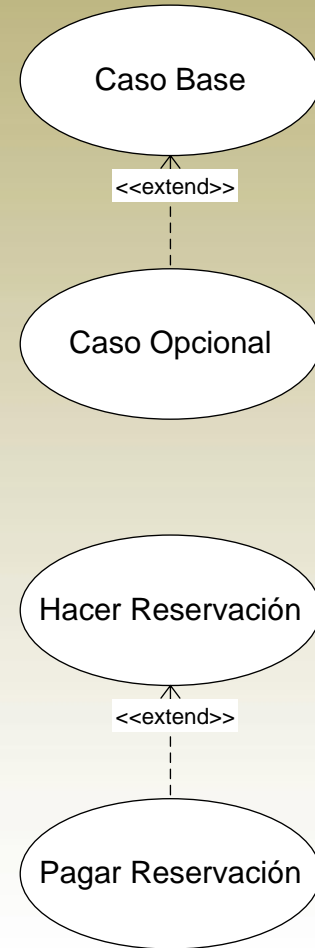




2. Diagramas de Casos de Uso

2. *Extensión*: (<<extend>>)

- Especifica como un caso de uso puede insertarse en otro para extender la funcionalidad de un caso de uso base.
- *El Caso Opcional es una extensión del Caso Base:*
Una instancia del caso de uso **Base** *puede incluir* el comportamiento especificado por el Caso **Opcional**.

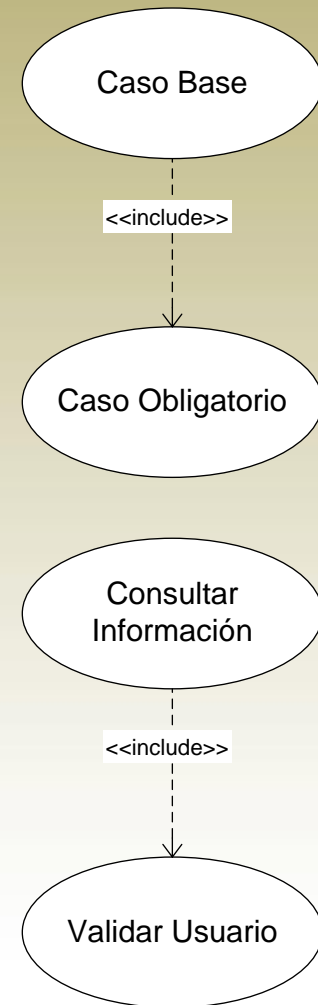




2. Diagramas de Casos de Uso

3. *Inclusión*: (<<include>>)

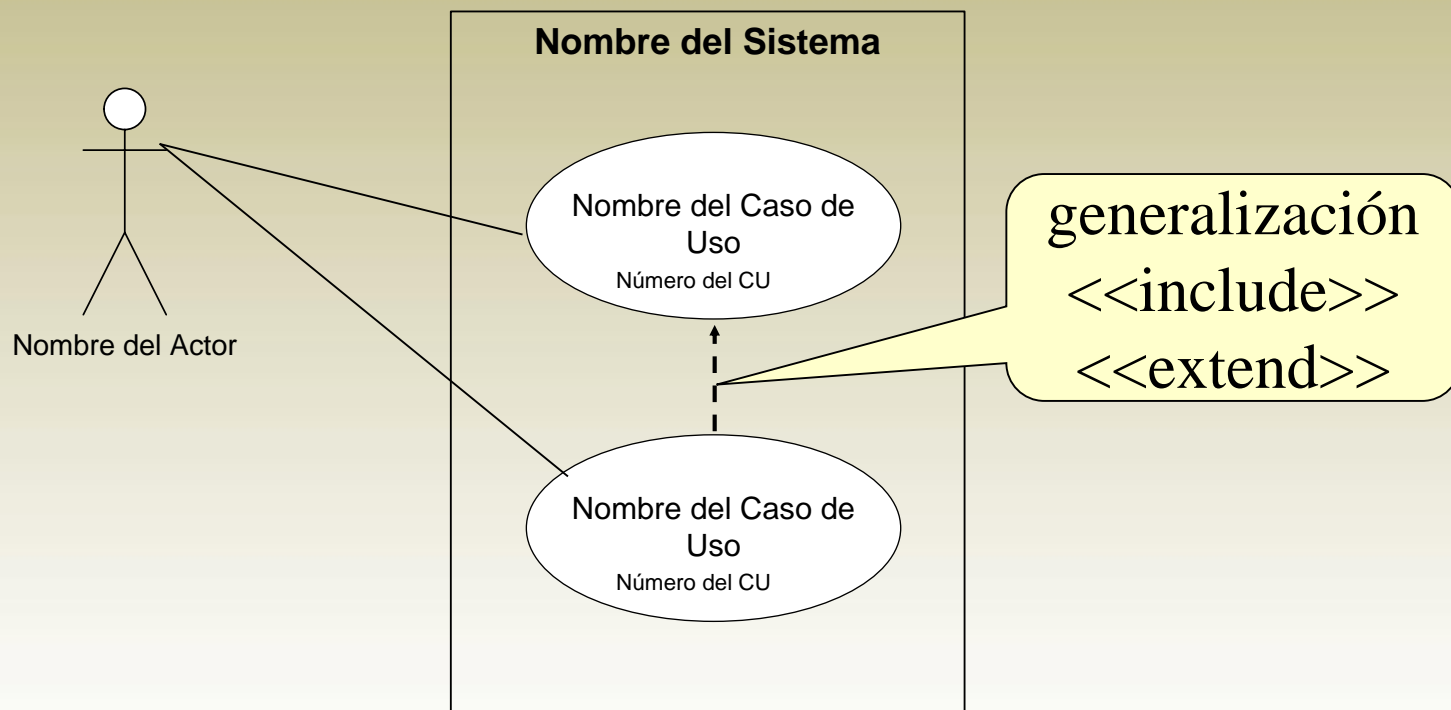
- La inclusión define como un caso de uso es parte obligatoria de un caso de uso base.
- *Un Caso Base incluye un Caso Obligatorio:*
Una instancia de un caso base **siempre** incluye el comportamiento especificado por un caso de uso obligatorio.





2. Diagramas de Casos de Uso

- Notación

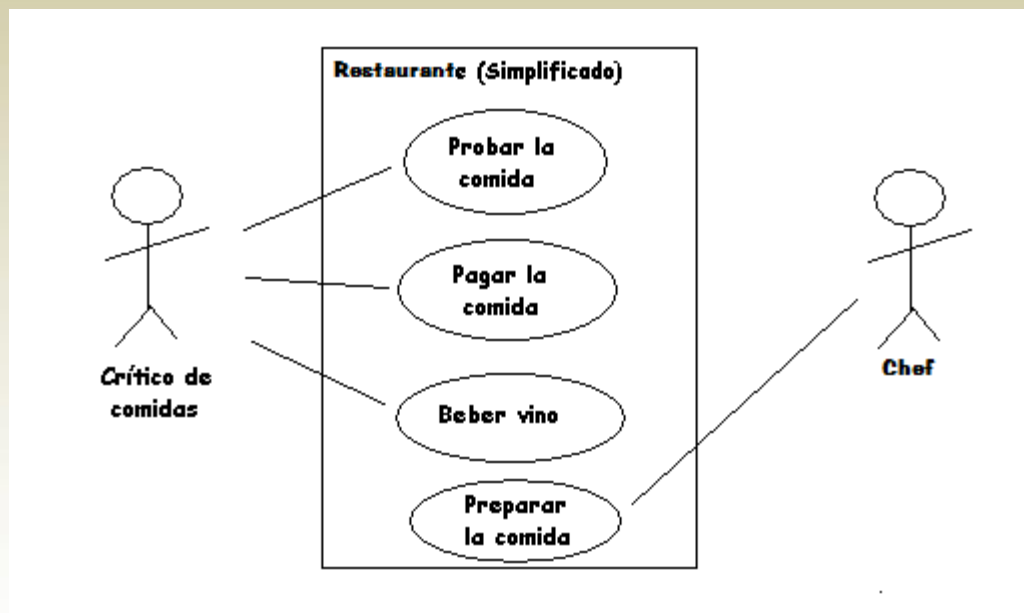




2. Diagramas de Casos de Uso

Caso de Uso

- Ejemplo:





2. Diagramas de Casos de Uso

Caso de Uso

- **Ejemplo 7.** En el modelo de un sistema de viajes aéreos se tienen dos casos de uso que tienen unos pasos comunes que se incluyen en otro caso de uso.



Figura 6.6. Un caso de uso incluido.

Ejercicio 6.2 Pag. 271



2. Diagramas de Casos de Uso

Caso de Uso

- **Ejemplo 8.** En el siguiente ejemplo se amplía el caso de uso Realizar Reserva para incluir la situación de que se puedan reservar asientos en primera clase.



Ejercicio 6.3 Pag. 272



2. Diagramas de Casos de Uso

Especificación de un Caso de Uso: (Documentación)

- Nombre del Caso de Uso.
- Actores.
- Propósito.
- Precondiciones.
- Flujo de Eventos Principal.
- Sub Flujos.
- Excepciones.
- Postcondiciones.



2. Diagramas de Casos de Uso

Nombre: BuscarProductos

ID: CU-1

Descripción:

El cliente solicita consultar productos que cumplan una serie de criterios de búsqueda. El sistema muestra los datos de los productos solicitados.

Actores: Cliente.

Precondiciones:

Se requiere que el cliente esté registrado en el sistema.

Curso normal del caso de uso:

1. El cliente selecciona buscar productos.
2. El sistema solicita los criterios de búsqueda.
3. El cliente introduce los criterios solicitados.
4. El sistema busca productos que cumplan los criterios introducidos por el cliente.
5. El sistema encuentra productos y se los muestra al cliente.

Postcondiciones:

El sistema muestra los productos encontrados.

Alternativa 1:

5. El sistema no encuentra productos.
6. El sistema informa al cliente de que no hay productos que cumplan el criterio de búsqueda.
7. Fin del caso de uso.



2. Diagramas de Casos de Uso

Nombre: BuscarProductos
ID: CU-1
Descripción: El cliente solicita consultar productos que cumplan una serie de criterios de búsqueda. El sistema muestra los datos de los productos solicitados.
Actores: Cliente.
Precondiciones: Se requiere que el cliente esté registrado en el sistema.
Curso normal del caso de uso: <ol style="list-style-type: none">1. El cliente selecciona buscar productos.2. El sistema solicita los criterios de búsqueda.3. El cliente introduce los criterios solicitados.4. El sistema busca productos que cumplan los criterios introducidos por el cliente.5. <i>SI</i> el sistema encuentra algún producto<ol style="list-style-type: none">5.1 <i>Para</i> cada producto encontrado<ol style="list-style-type: none">5.1.1. El sistema muestra sus datos.6. <i>SI NO</i><ol style="list-style-type: none">6.1 El sistema comunica al cliente que no ha encontrado productos.7. El cliente repite los pasos 2 a 6 hasta finalizar las búsquedas.
Postcondiciones: Ninguna.
Alternativas: Ninguna



3. Herramientas para elaborar Diagramas de Casos de Uso

- Eclipse
- ArgoUml
- WhiteStar

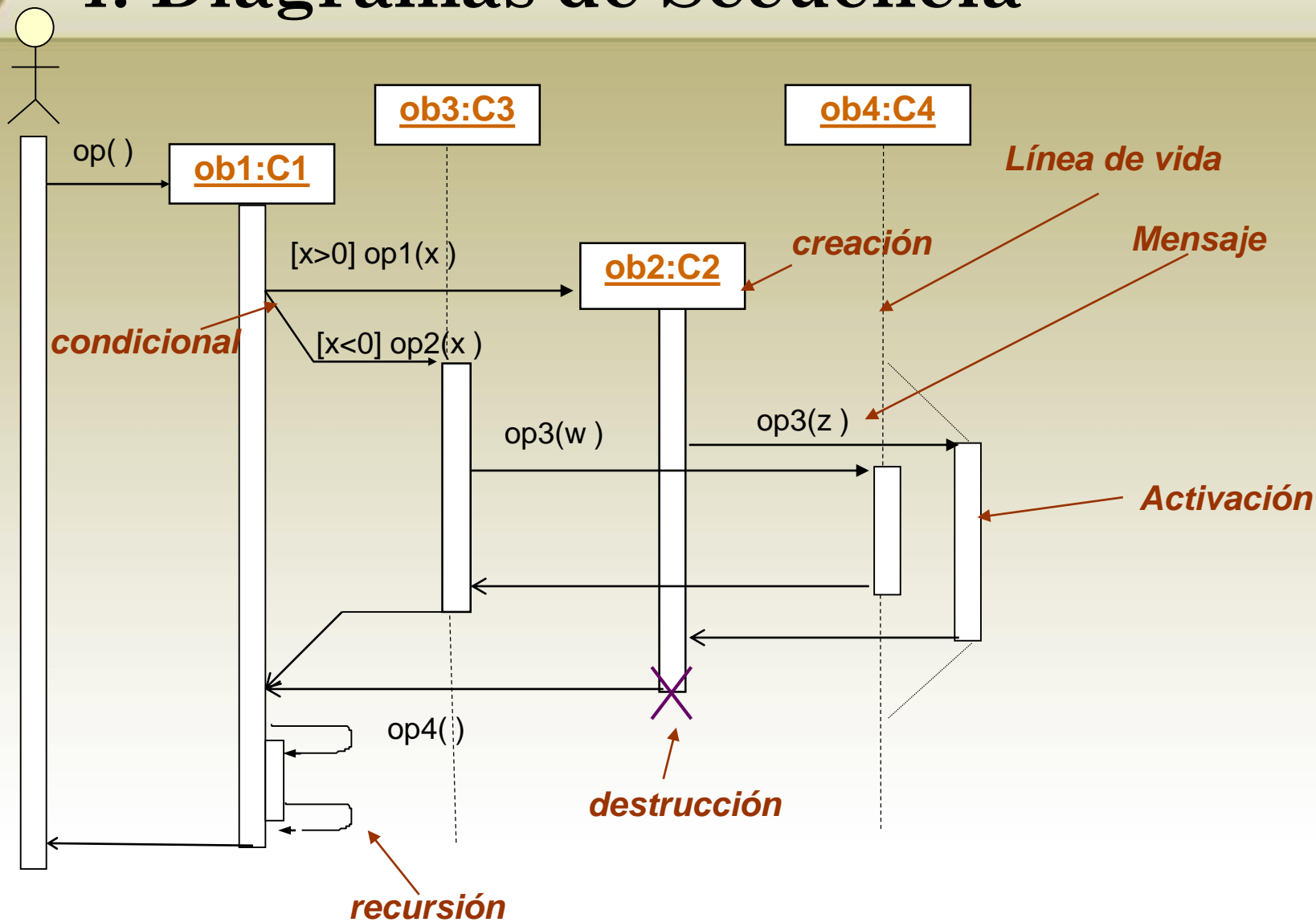
ACTIVIDAD 6.6 Realiza en ArgoUML el diagrama de casos de uso para el sistema de venta y el de gestión de notas.



4. Diagramas de Secuencia

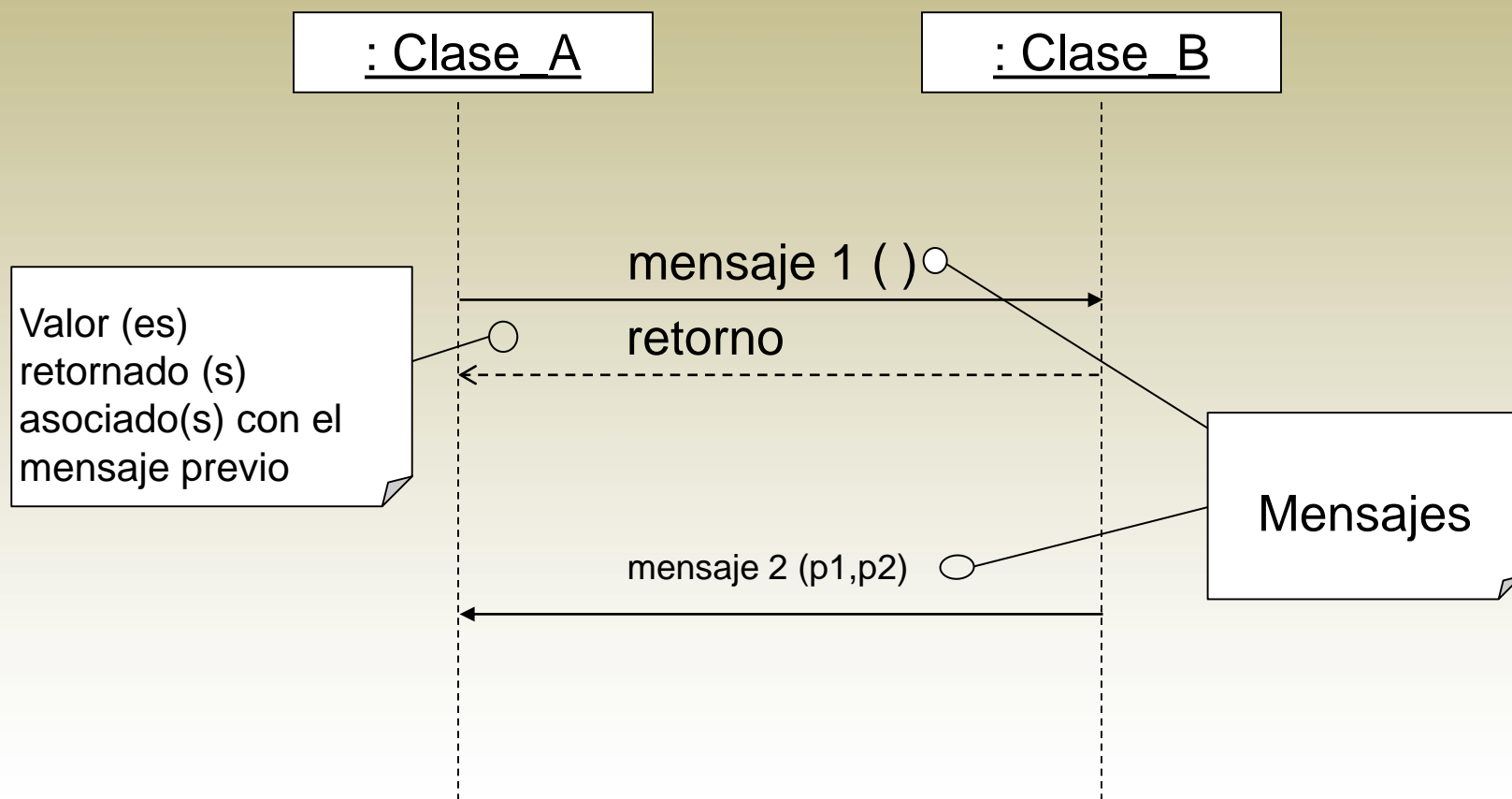
- Los diagramas de secuencia describen como **colaboran y se comunican (interacción)** los objetos del sistema.
- Muestra los objetos que participan en una interacción, el intercambio de mensajes y su ordenamiento en el tiempo.
- Un diagrama de secuencia es una representación que muestra, para un escenario de un caso de uso, los *eventos* que generan los actores, su *orden* y posibles eventos internos en el sistema

4. Diagramas de Secuencia





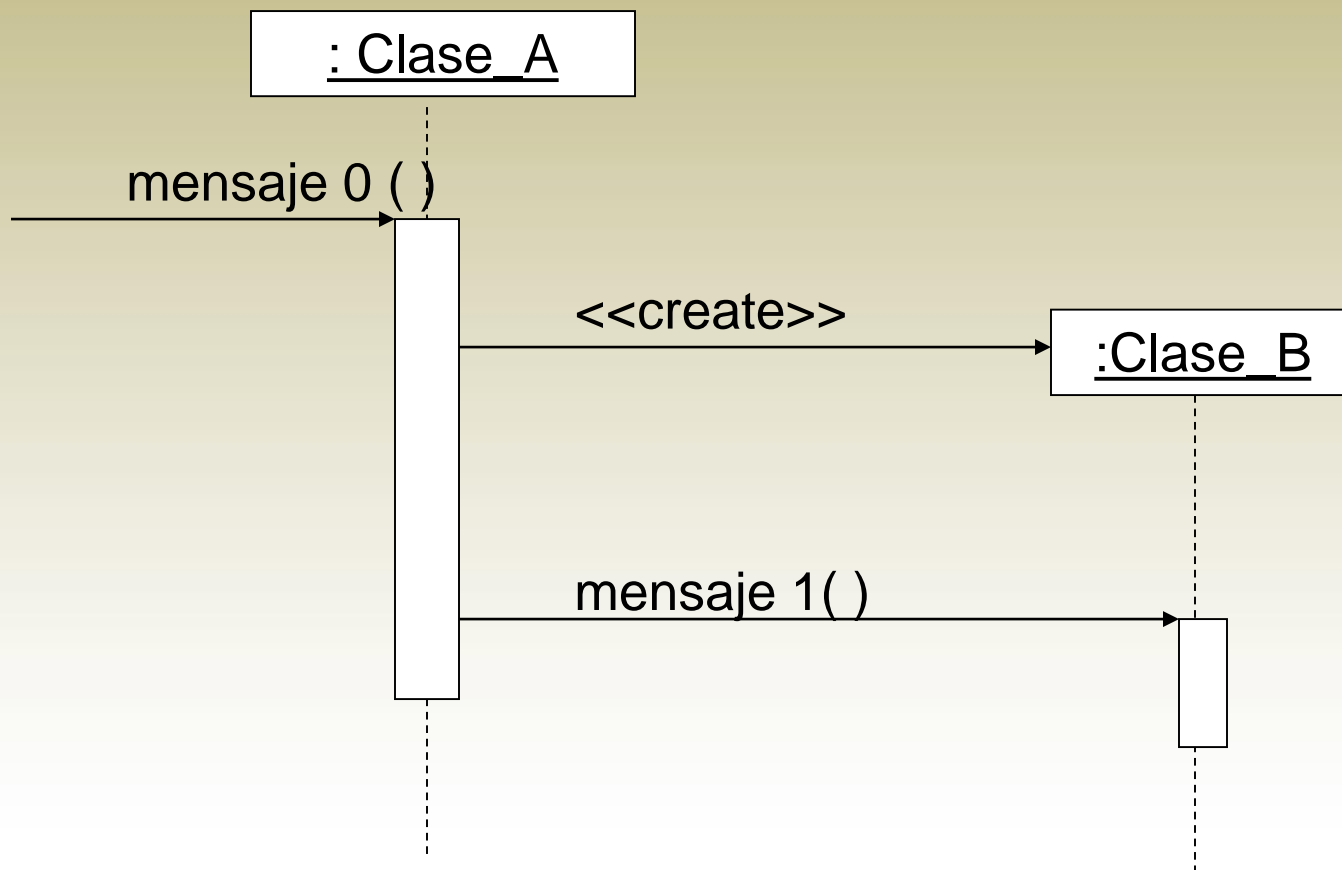
4. Diagramas de Secuencia





4. Diagramas de Secuencia

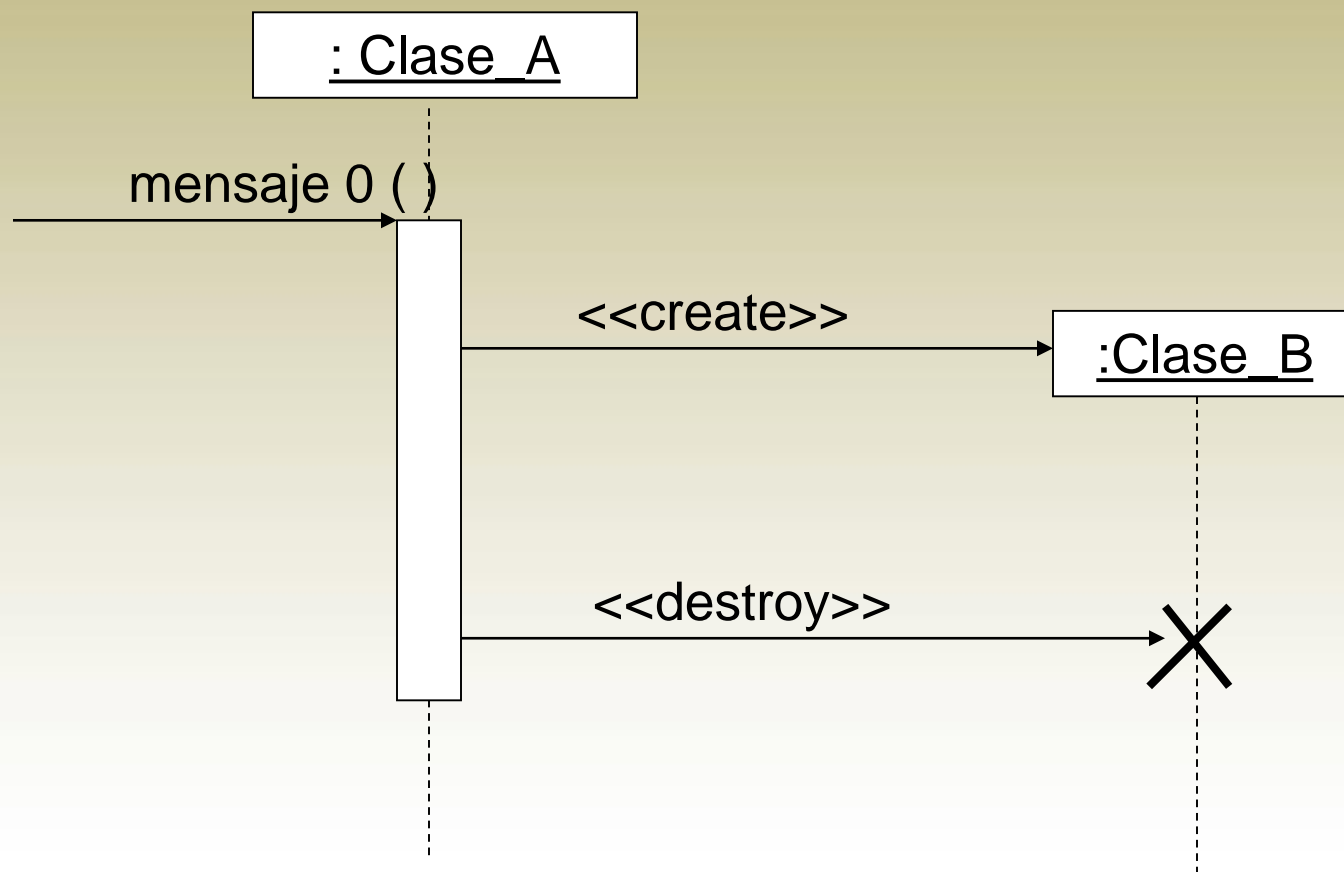
- Crear Objetos: **<<create>>**





4. Diagramas de Secuencia

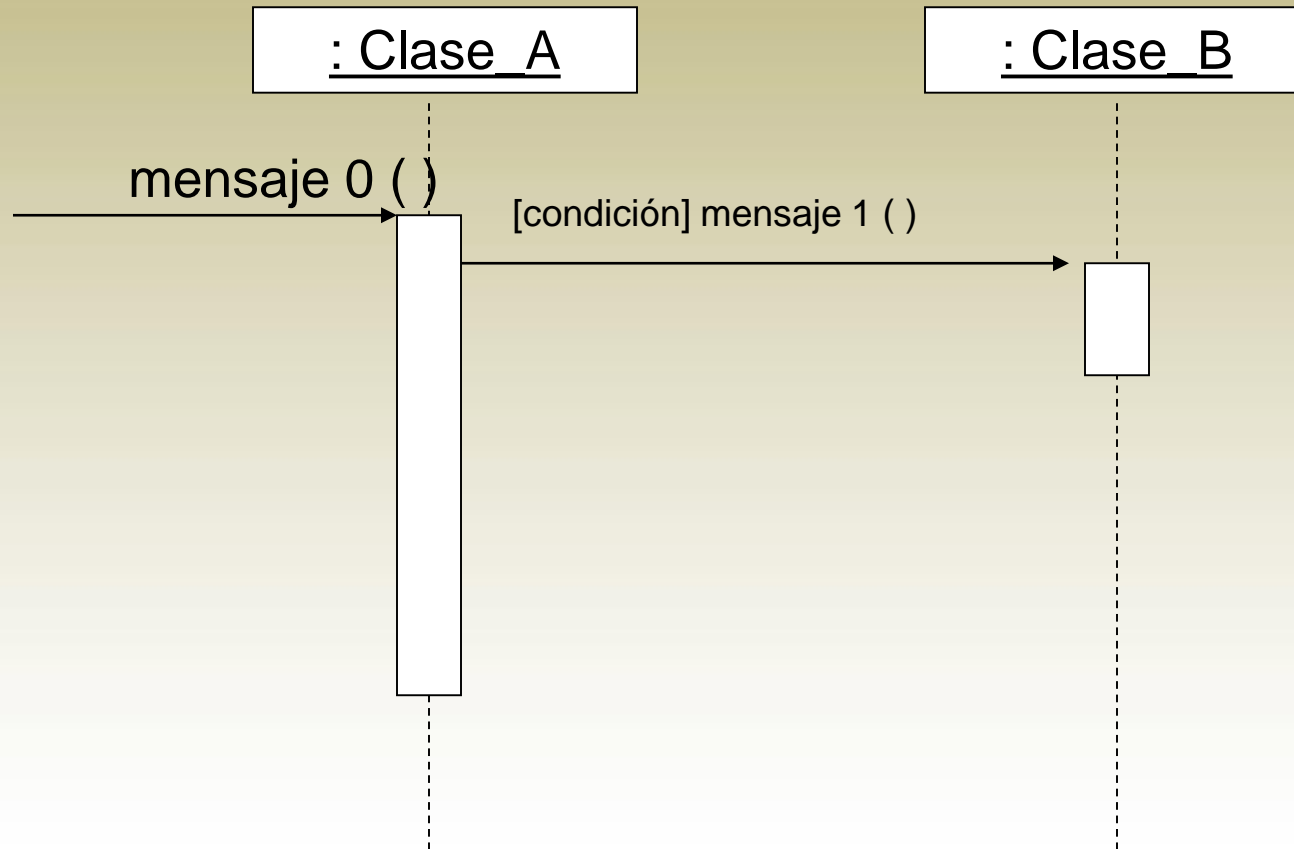
- Destruir Objetos: **<<destroy>>**





4. Diagramas de Secuencia

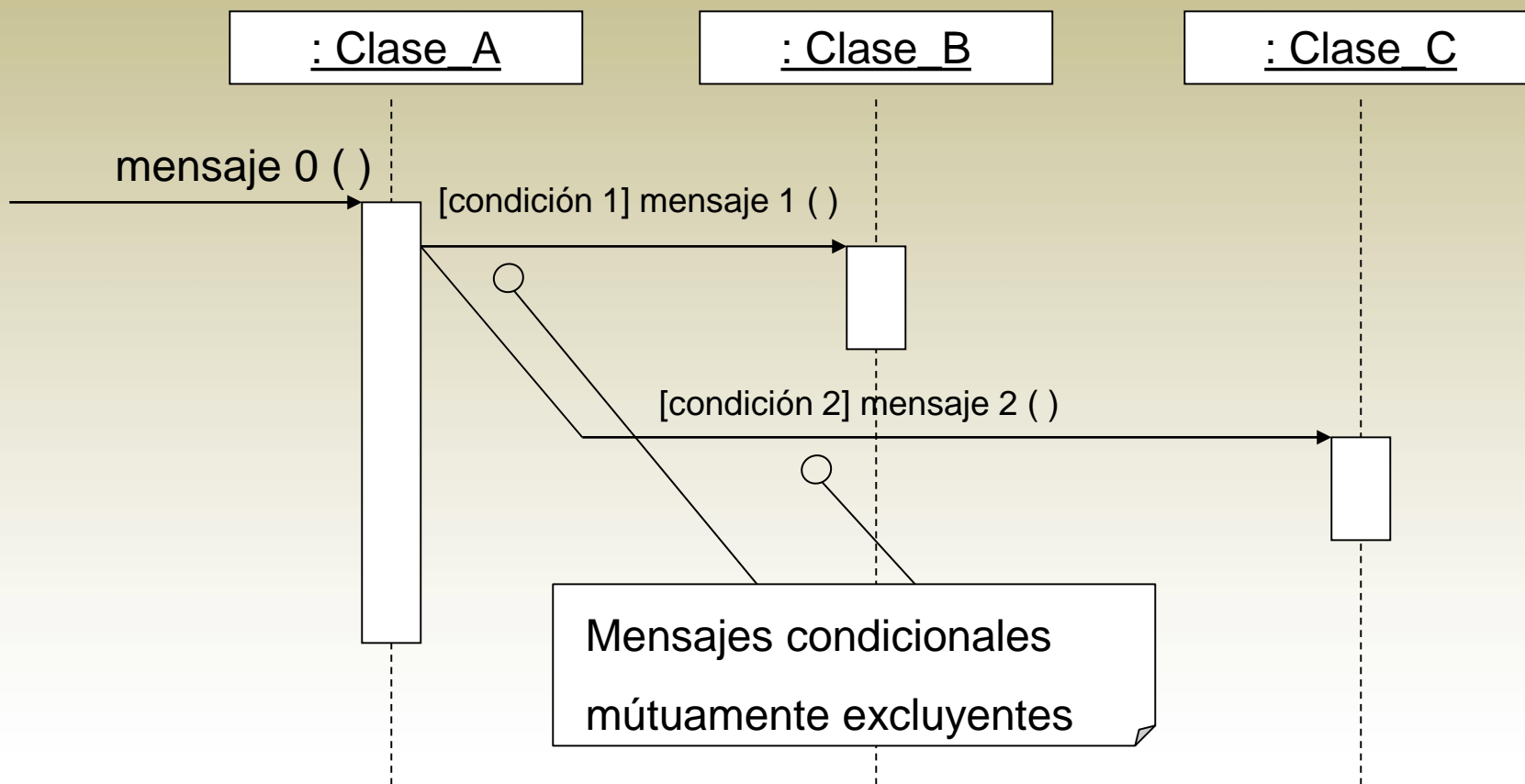
- Mensajes Condicionales





4. Diagramas de Secuencia

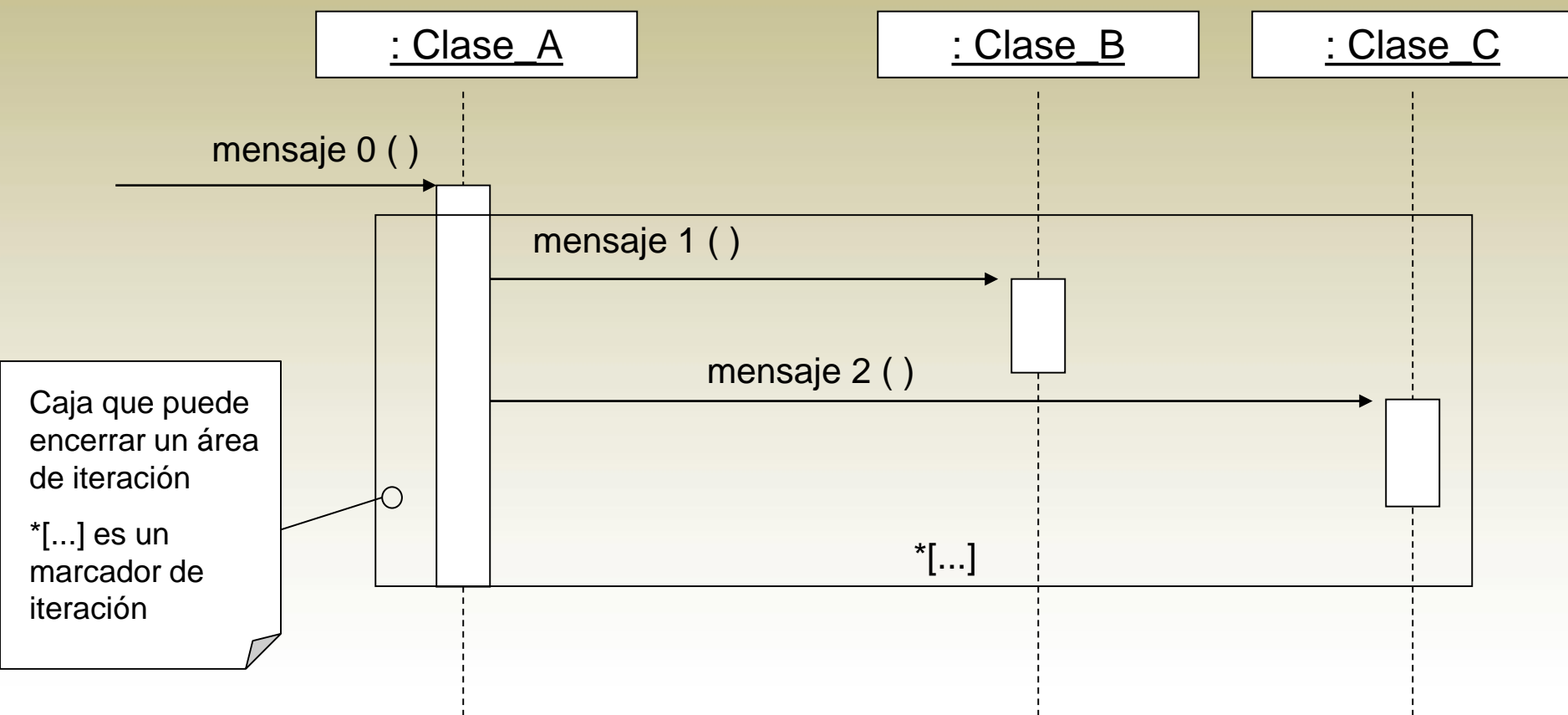
- Mensajes Condicionales Excluyentes





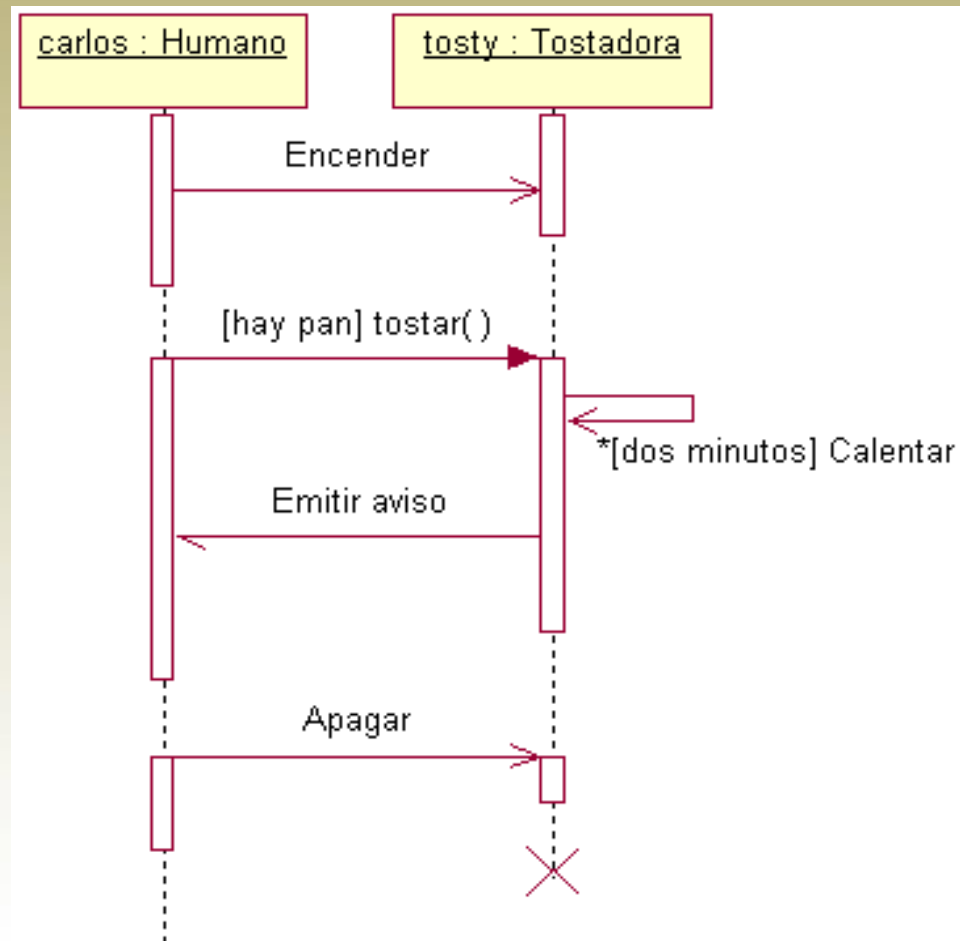
4. Diagramas de Secuencia

- Caja de Iteración:





4. Diagramas de Secuencia



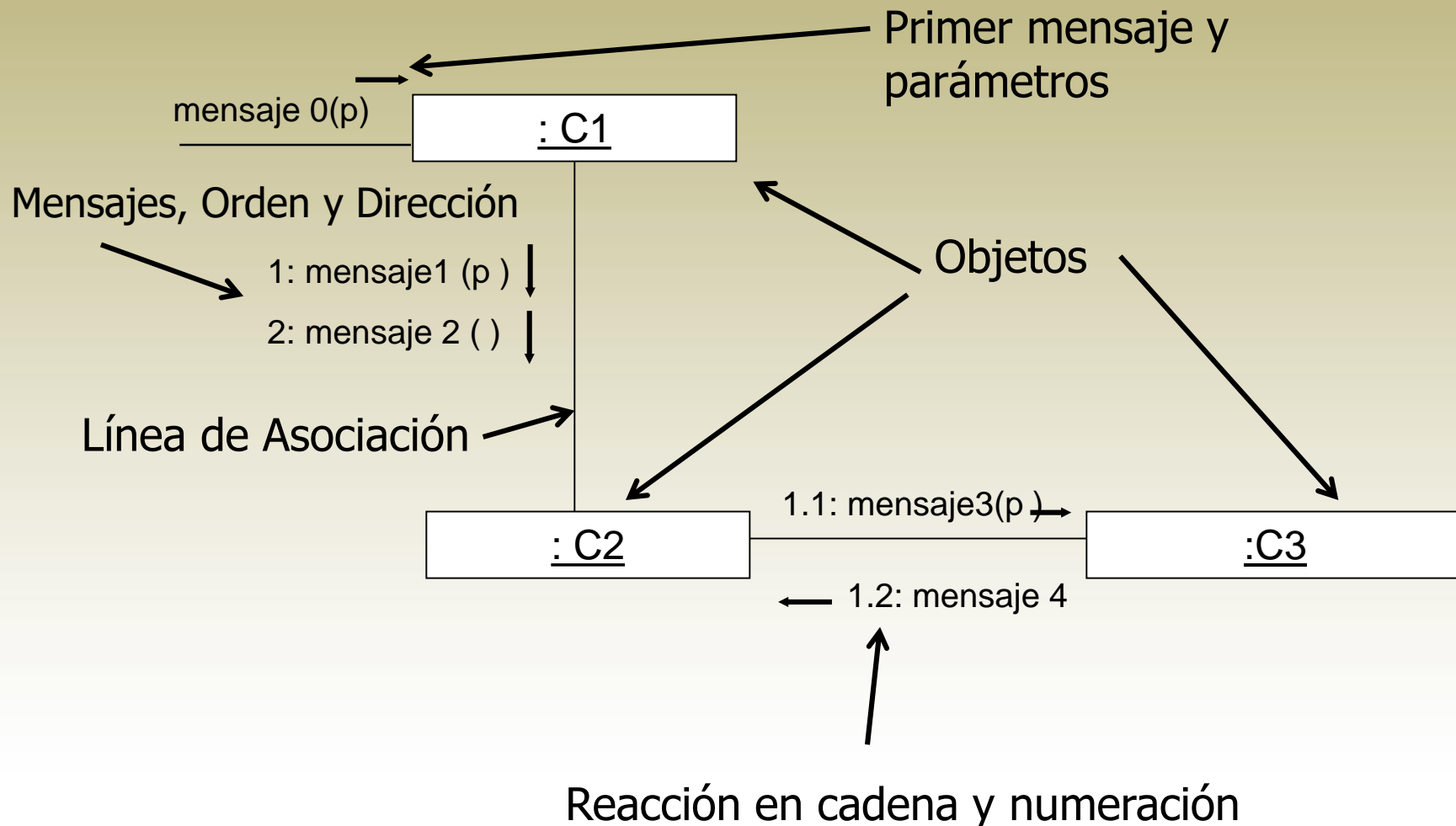


5. Diagramas de Colaboración

- Un Diagrama de Colaboración describe la **interacción** entre los objetos, numerando la secuencia de mensajes.



5. Diagramas de Colaboración





5. Diagramas de Colaboración

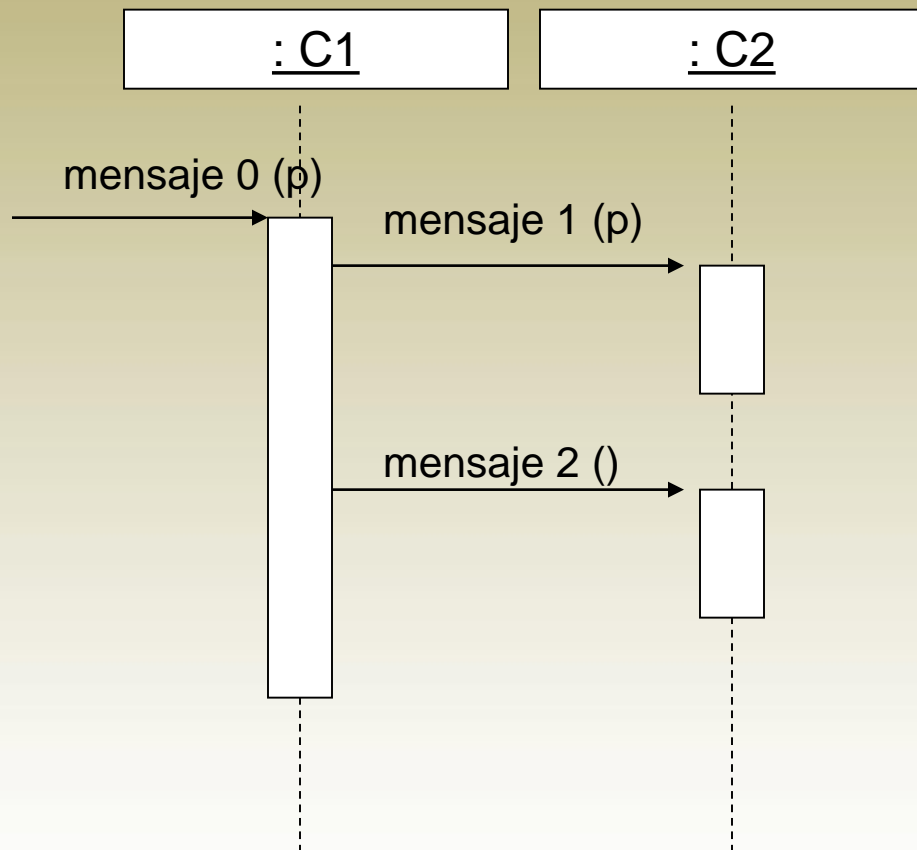


Diagrama de Secuencia

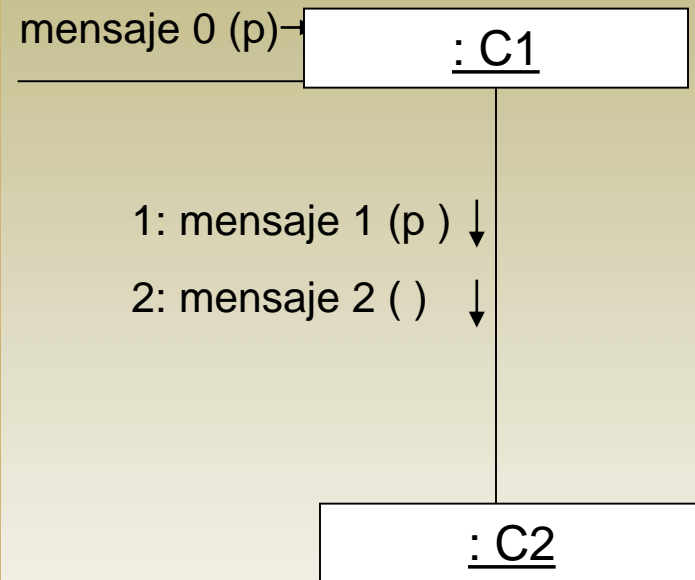
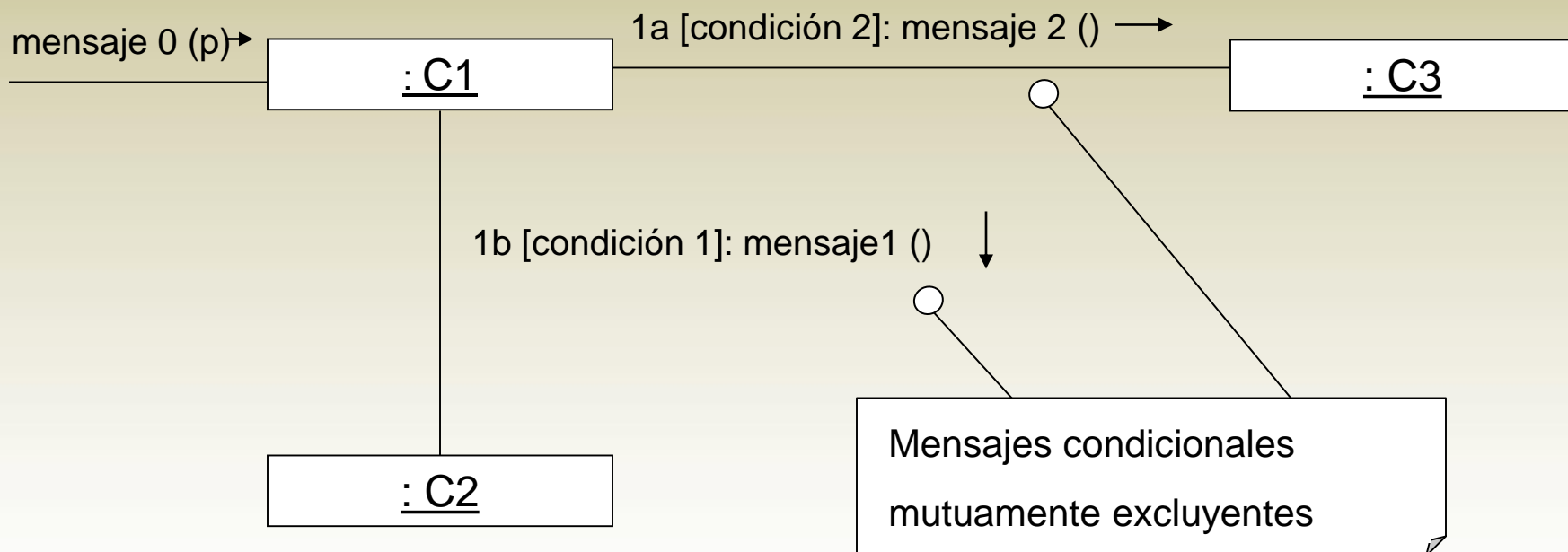


Diagrama de Colaboración



5. Diagramas de Colaboración

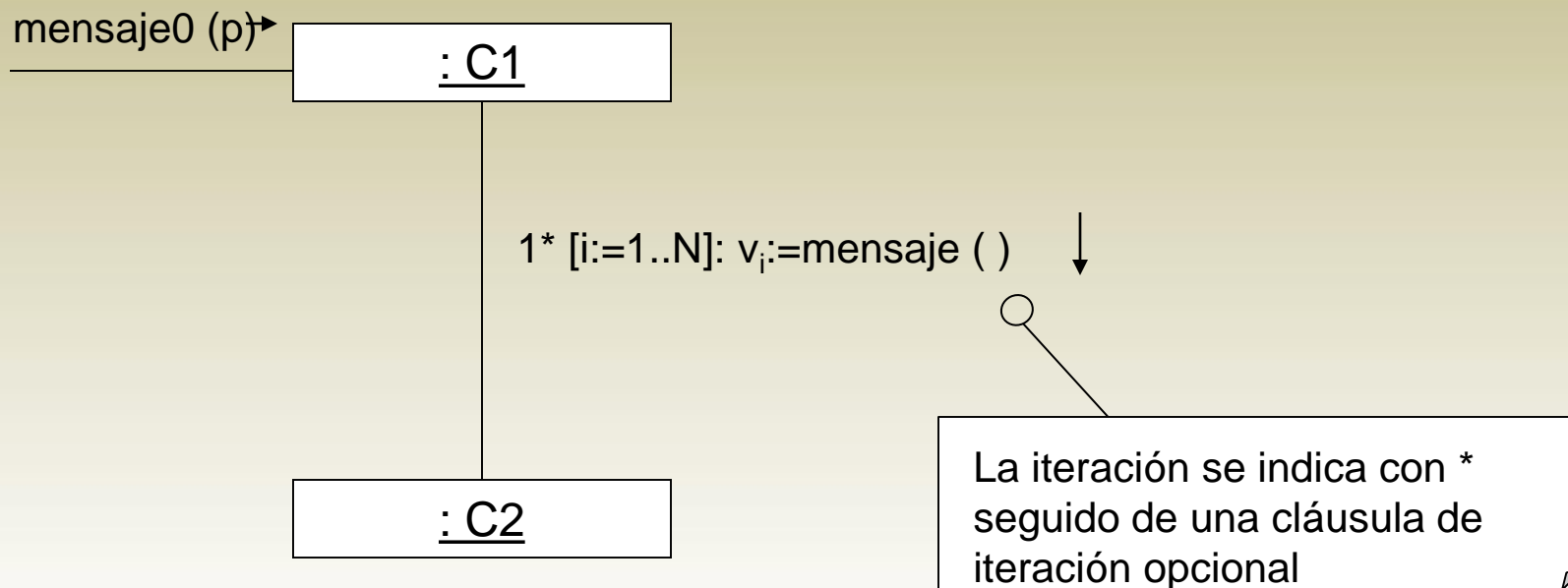
- Mensajes Excluyentes





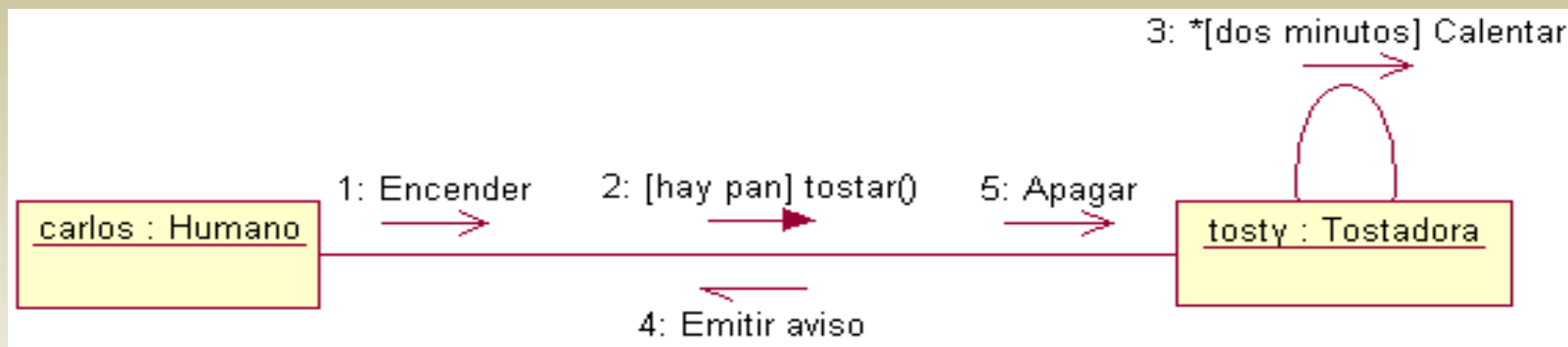
5. Diagramas de Colaboración

- Iteración





5. Diagramas de Colaboración





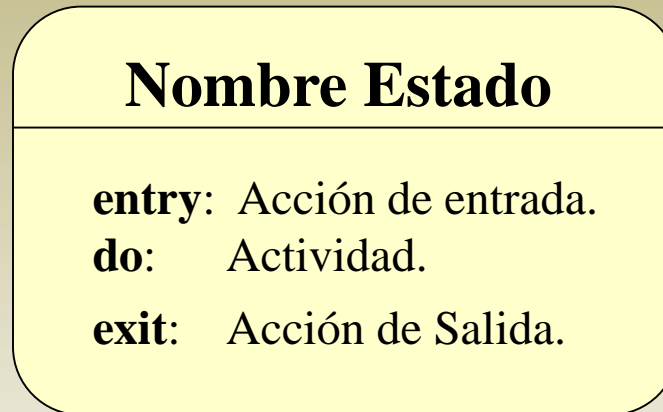
6. Diagramas de Estado

- Los diagramas de estado muestran los diferentes estados de un objeto durante su vida, y los estímulos que provocan los cambios de estado en un objeto.
- Los diagramas de estado ven a los objetos como *máquinas de estado* o autómatas finitos que pueden estar en un conjunto de estados finitos y que pueden cambiar su estado a través de un estímulo perteneciente a un conjunto finito.



6. Diagramas de Estados

- Estado:



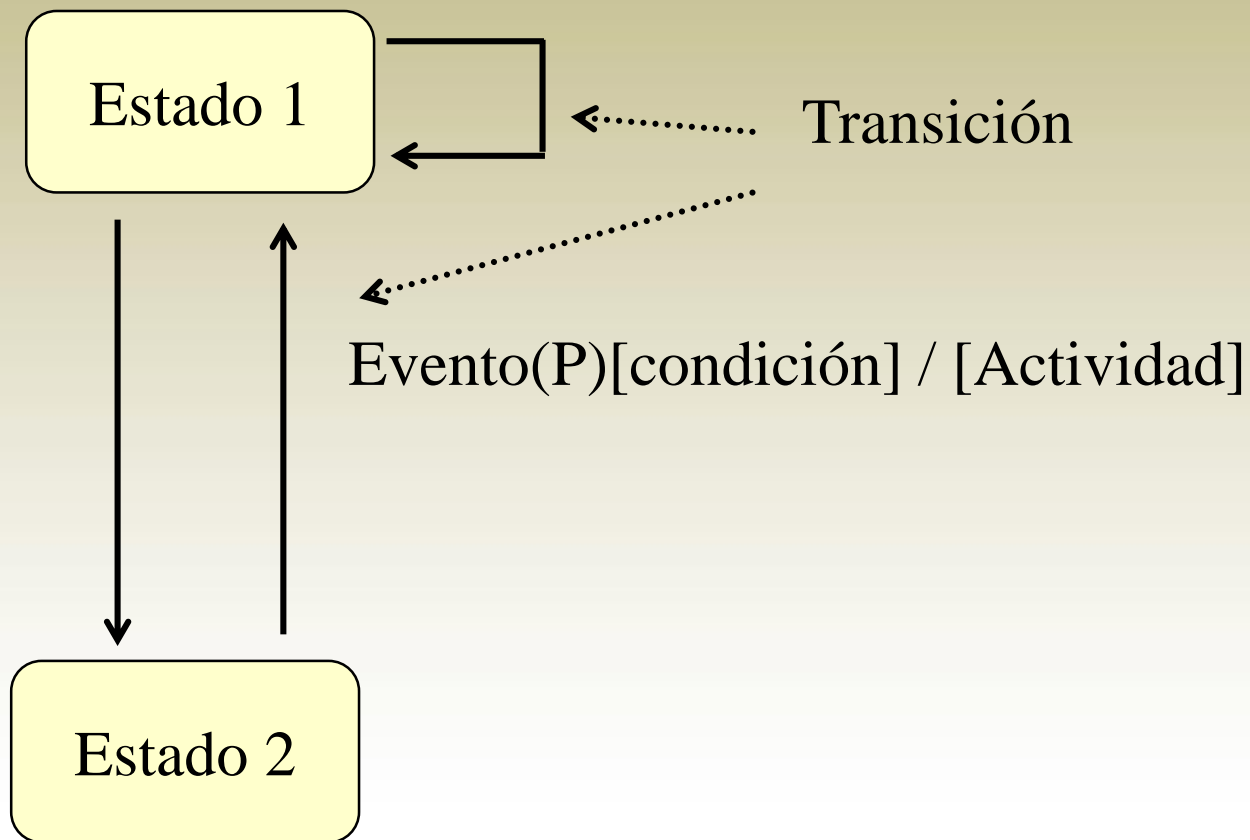
- Estado Inicial ●

- Estado Final ●



6. Diagramas de Estados

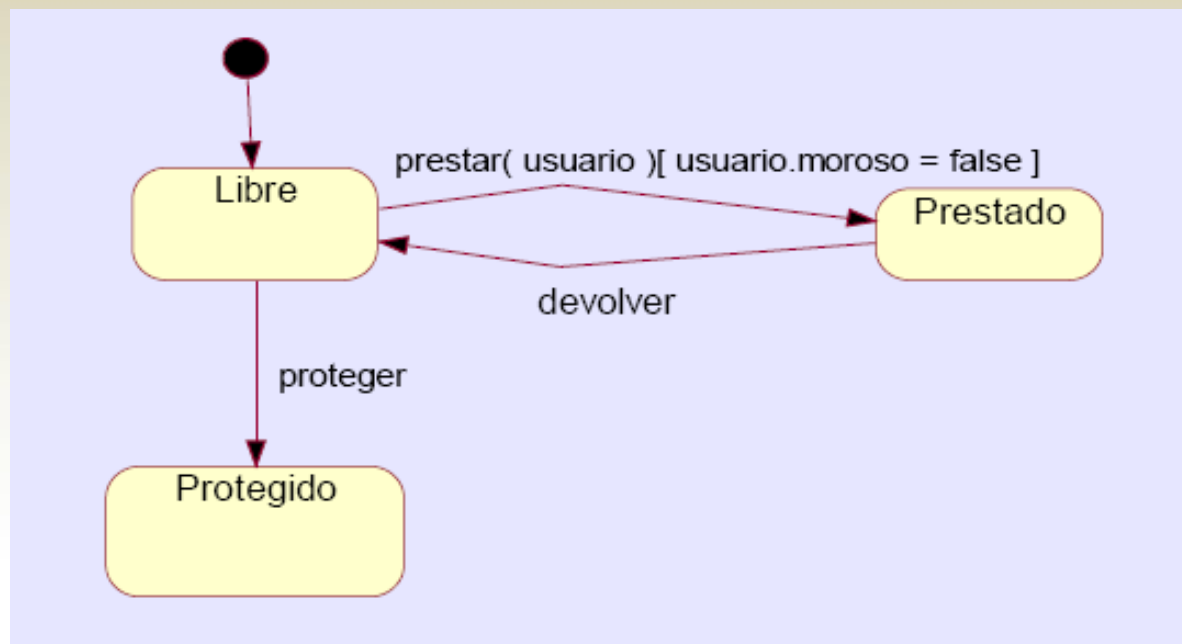
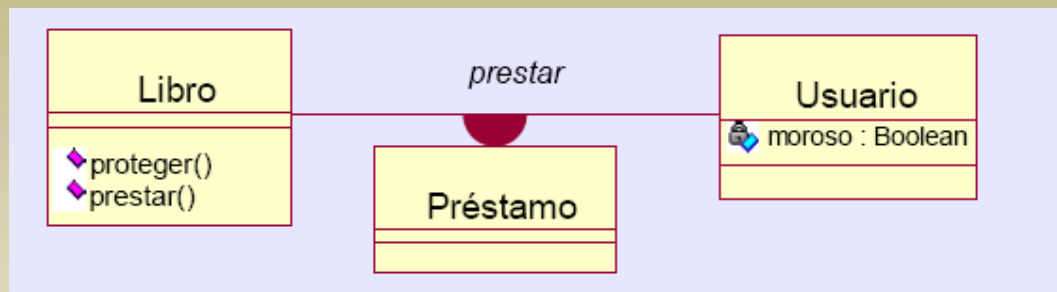
- Transición:





6. Diagramas de Estados

- Diagrama de Estados de un Libro:





7. Diagramas de actividades

- El **diagrama de actividades** define la lógica de los procedimientos, los procesos del negocio y flujos de trabajo del sistema.
- Un diagrama de actividades demuestra la *serie de actividades* que deben ser realizadas en un caso de uso, así como las distintas rutas que pueden irse desencadenando en el caso de uso.



7. Diagramas de Actividades

- **Actividad:**

Representa una acción que será realizada por el sistema.

Nombre de la Actividad

- Actividad *Inicial* ●

- Actividad *Final* ●

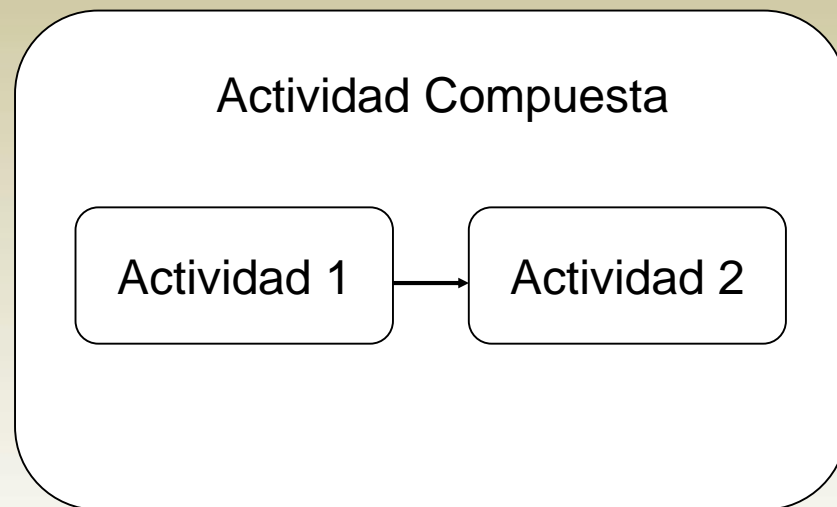
Verificar Password del Usuario



7. Diagramas de Actividades

- Sub Actividad:

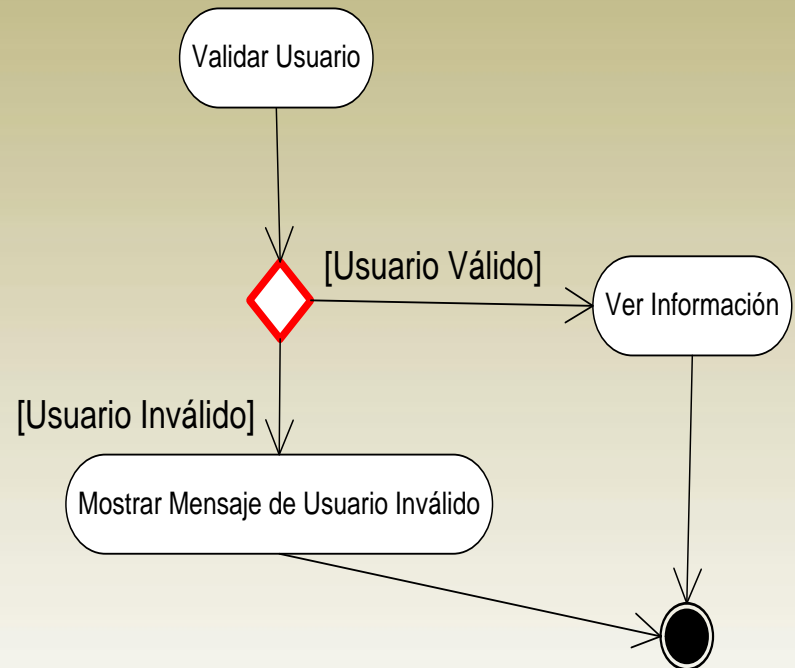
Una acción puede ser descompuesta en varias actividades.





7. Diagramas de Actividades

- **Ramificación (Branch):**
 - Una ramificación surge cuando existe la posibilidad que ocurra más de una transición (resultado) al terminar determinada actividad.
 - Este elemento es representado a través de un rombo.

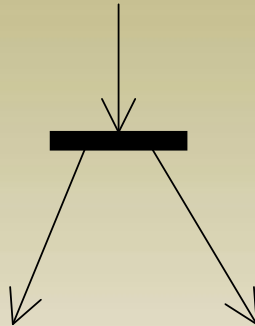




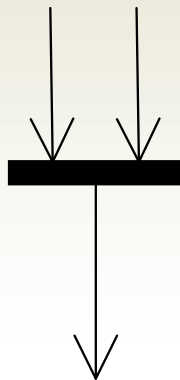
7. Diagramas de Actividades

- **Especificaciones Join**

- *División*



- *Unión*





7. Diagramas de Actividades

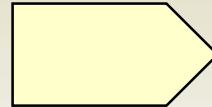
- **Señales**

Algunas acciones responden a señales

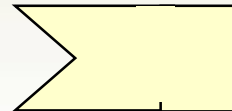
- Señales de tiempo



- Envío de señal



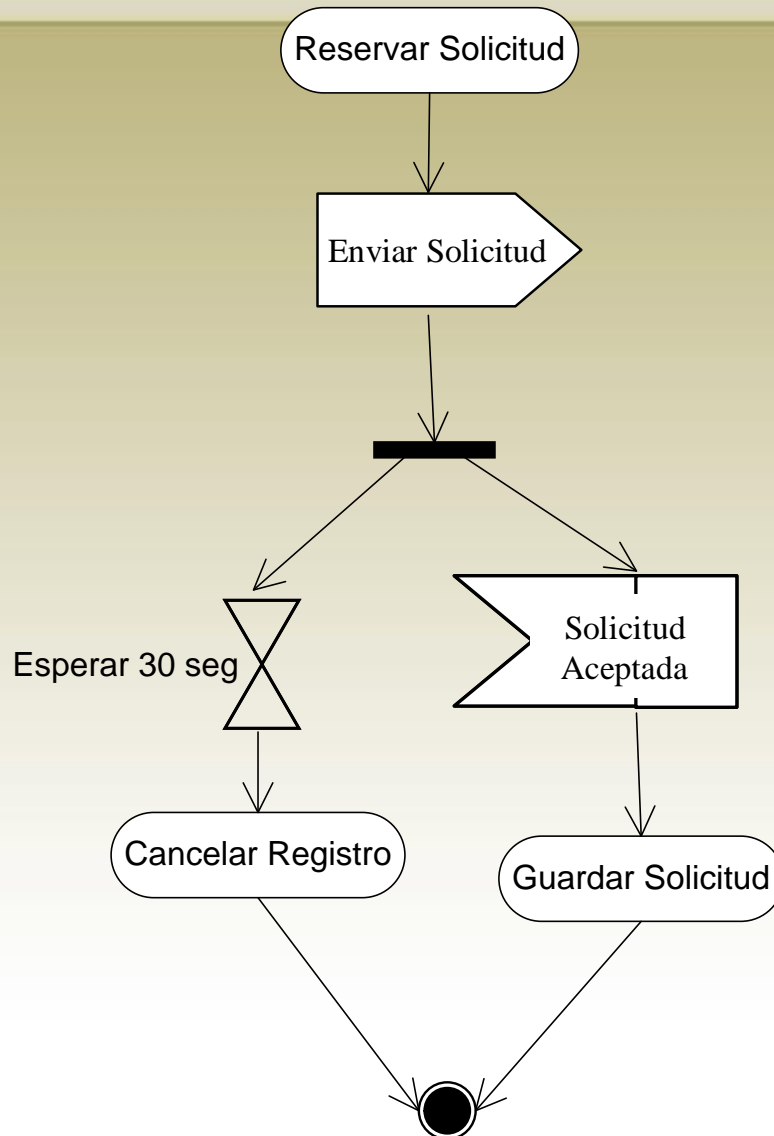
- Recepción de señal





7. Diagramas de Actividades

- Señales
 - Ejemplo:





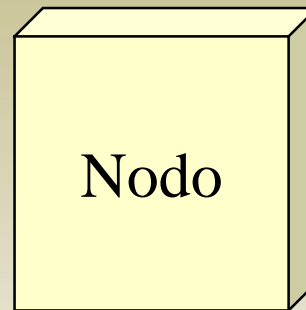
8. Diagramas de Despliegue

- Un diagrama de despliegue muestra las *relaciones físicas* entre los componentes hardware y software en el sistema final, es decir, la configuración de los elementos de procesamiento en tiempo de ejecución y los componentes software (procesos y objetos que se ejecutan en ellos).
- En el diagrama de despliegue se indica la situación física de los componentes lógicos desarrollados. Es decir se sitúa el software en el hardware que lo contiene.

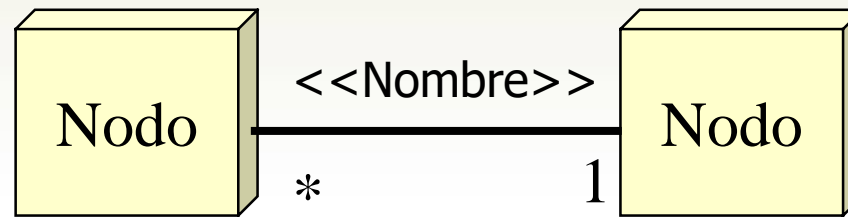


8. Diagramas de Despliegue

- **Nodo:** Elemento donde se ejecutan los componentes, representan el despliegue físico de estos componentes.

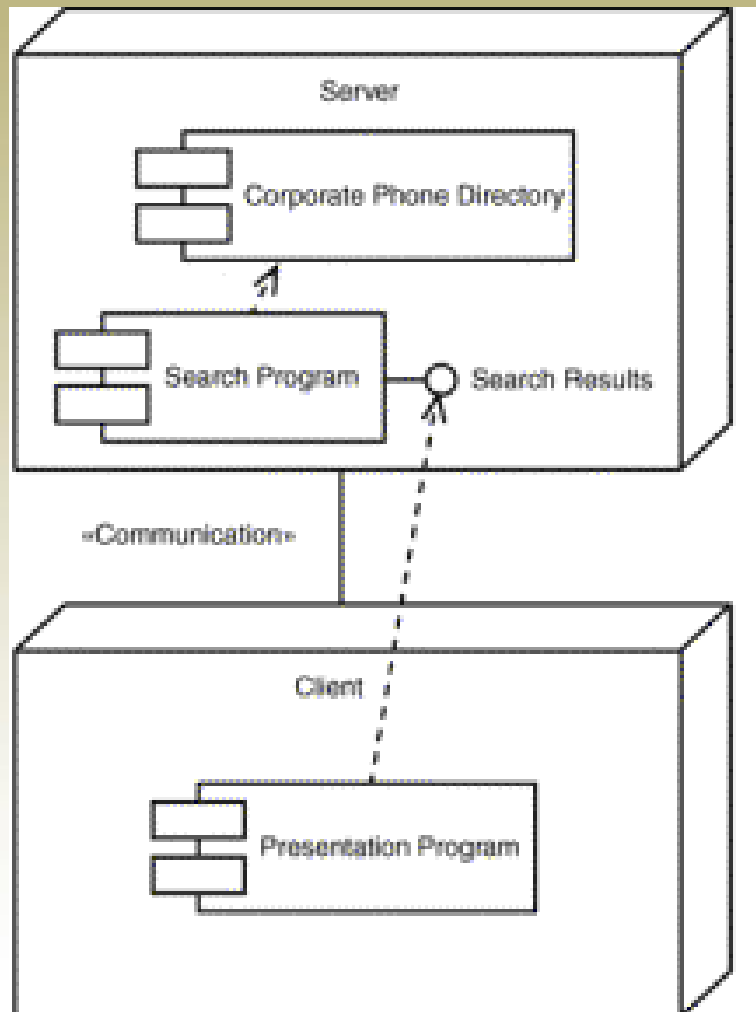


- **Asociación:** Representa el tipo relación que soporta la comunicación entre nodos





8. Diagramas de Despliegue





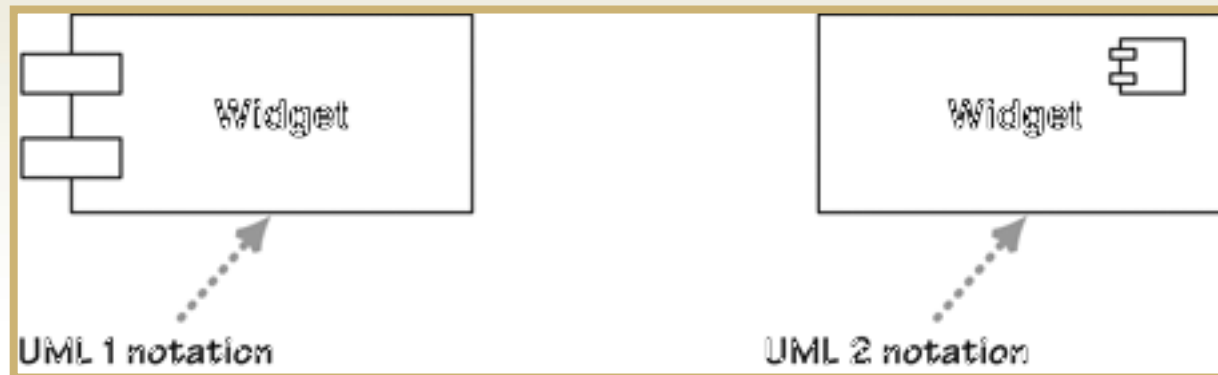
9. Diagramas de Componentes

- El diagrama de componentes muestra la relación entre componentes de *software*, sus dependencias, su comunicación, su ubicación y otras condiciones.
- Un diagrama de componentes muestra las organizaciones y dependencias lógicas entre componentes *software*, sean éstos componentes de código fuente, binarios o ejecutables.
- Define los módulos físicos del software y sus relaciones.



9. Diagramas de Componentes

- **Componente**
 - Es un tipo de contenedor.
 - Provee una vista encapsulada de las funcionalidades definidas en las clases.
 - Por ejemplo, un **paquete** en un diagrama de componentes representa una división física del sistema.

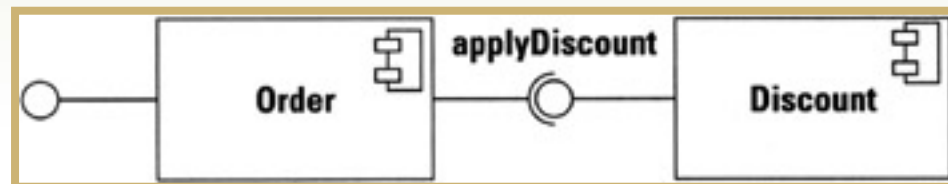
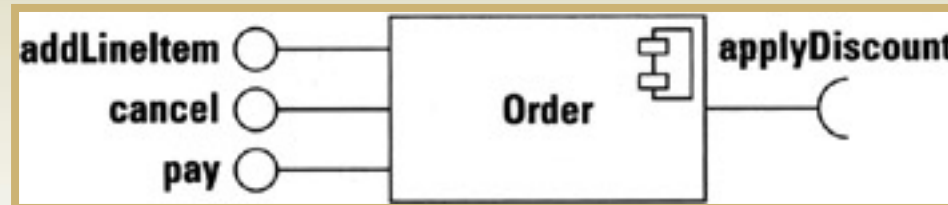




9. Diagramas de Componentes

- **Interfaces**

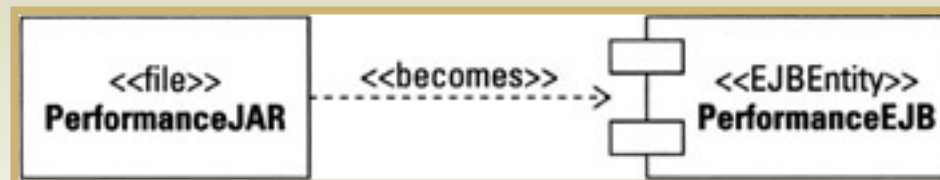
- Las interfaces son los **puntos visibles** de entrada o los servicios que un componente está ofreciendo y dejando disponibles a otros componentes de software y clases.
- Exponen funcionalidades para otros componentes y las requeridas de otros.





9. Diagramas de Componentes

- Dependencia de módulos
 - Abstrae la implementación de la interfaz e indica la dependencia entre módulo





10. Ingeniería Inversa

- En este capítulo aprenderemos a generar los diagramas de clases y de secuencia utilizando el entorno Eclipse.
- Pluging ObjectAid UML y eUML2.