

# Bases de datos

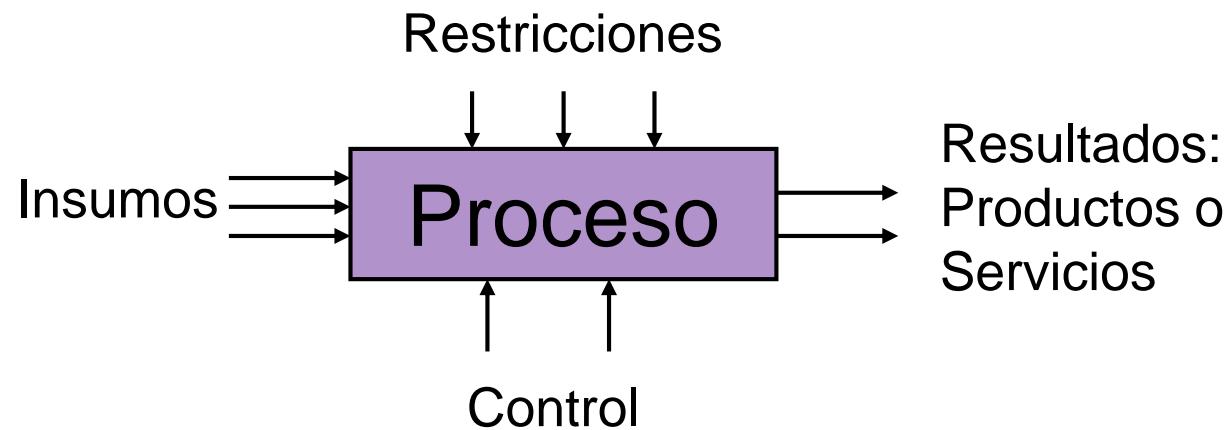
## Unidad 2

Universidad de Los Andes  
Escuela de Ingeniería de Sistemas  
Departamento de Computación

Tema 2. Orientación por objetos y modelado de datos en  
UML

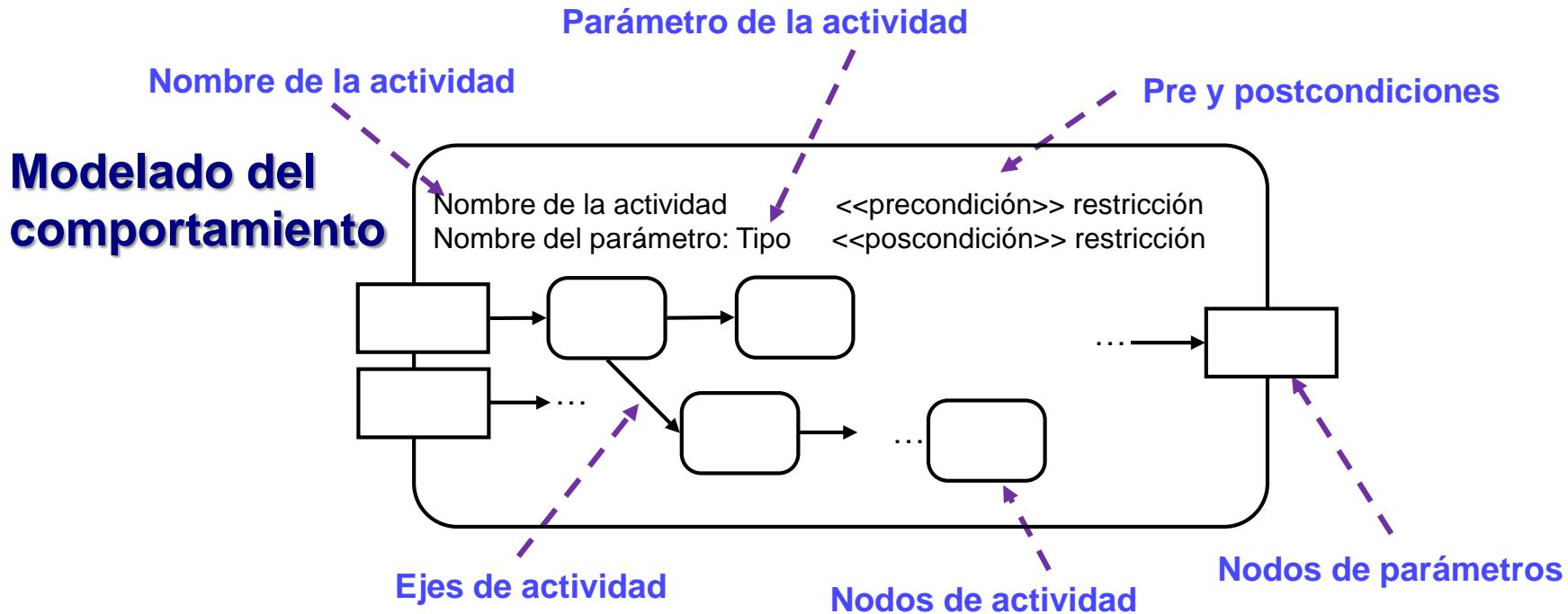
# Vista de procesos

- ▶ **Describe los procesos del sistema y cómo ellos se comunican entre sí**
- ▶ **Un proceso es un conjunto estructurado de actividades diseñado para alcanzar un objetivo establecido**
  - ▶ El modelo Caja Negra de un proceso:



# Diagramas de actividades

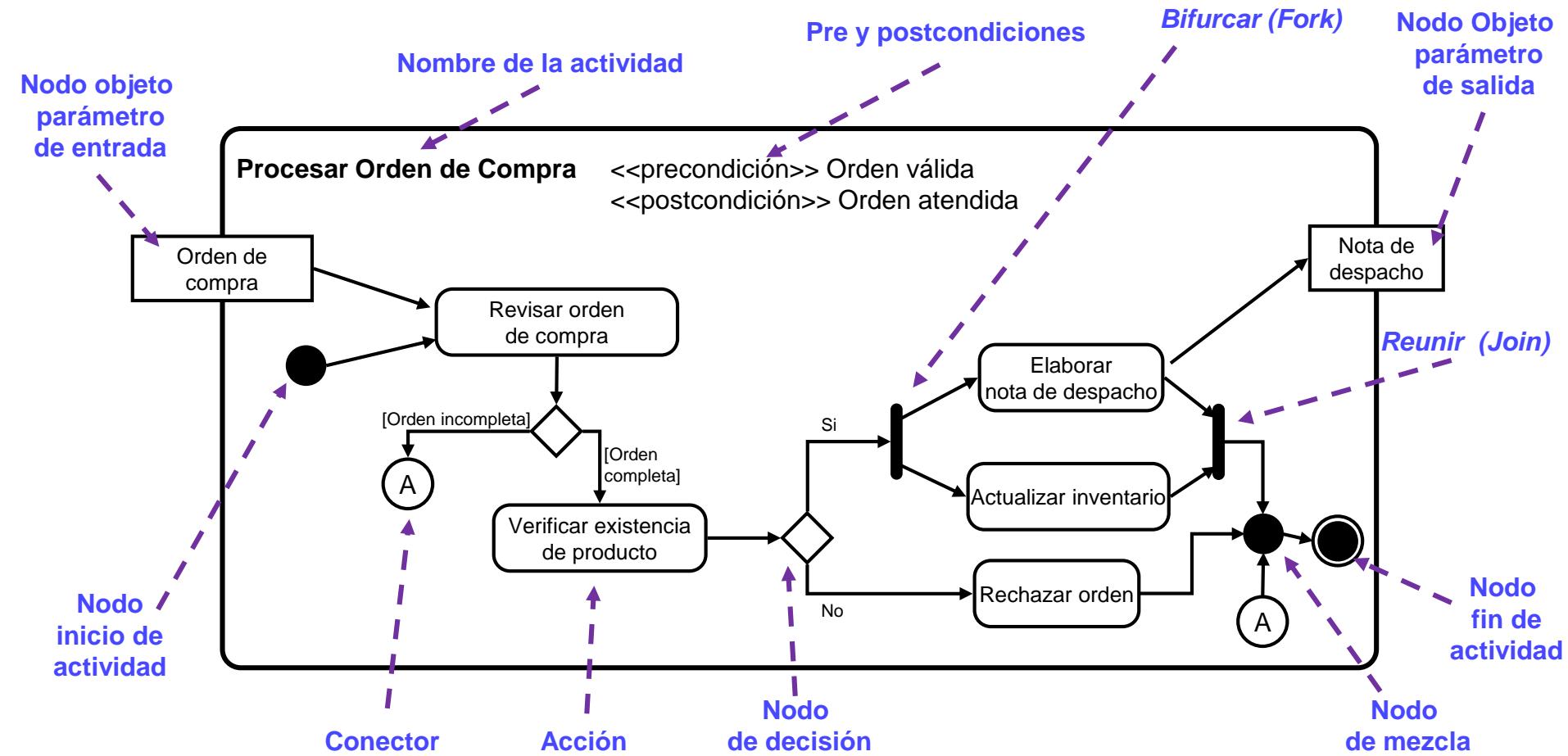
- ▶ Los diagramas de actividades son usados para elaborar modelos de flujos de trabajo (workflow) de un sistema:
  - ▶ Expresan qué acciones se requieren
  - ▶ Qué hacen estas acciones
  - ▶ Cuándo tienen lugar
  - ▶ Dónde estas acciones tienen lugar (partición de actividades)



# Actividad vs. Acción

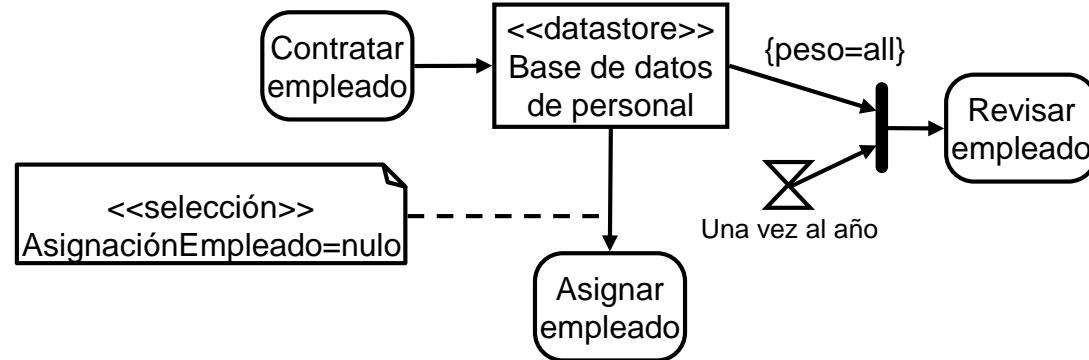
- ▶ **Actividad: especificación de un comportamiento que puede ser parametrizado y que define la secuenciación coordinada de unidades subordinadas denominadas acciones**
- ▶ **Una actividad consta de un conjunto estructurado e interrelacionado (conectado) de acciones**
  - ▶ Las acciones son los bloques de construcción de las actividades
- ▶ **La actividad se ejecuta cuando es invocada:**
  - ▶ directamente por otra actividad, o
  - ▶ mediante pase de mensajes
- ▶ **Una acción es la unidad fundamental de especificación de comportamiento**
- ▶ **Una acción es atómica**
  - ▶ No puede ser descompuesta en otras acciones
- ▶ **Una acción toma uno o más objetos de entradas (*tokens*) y las transforma en uno o más objetos de salidas (*tokens*)**

# Diagrama de actividades

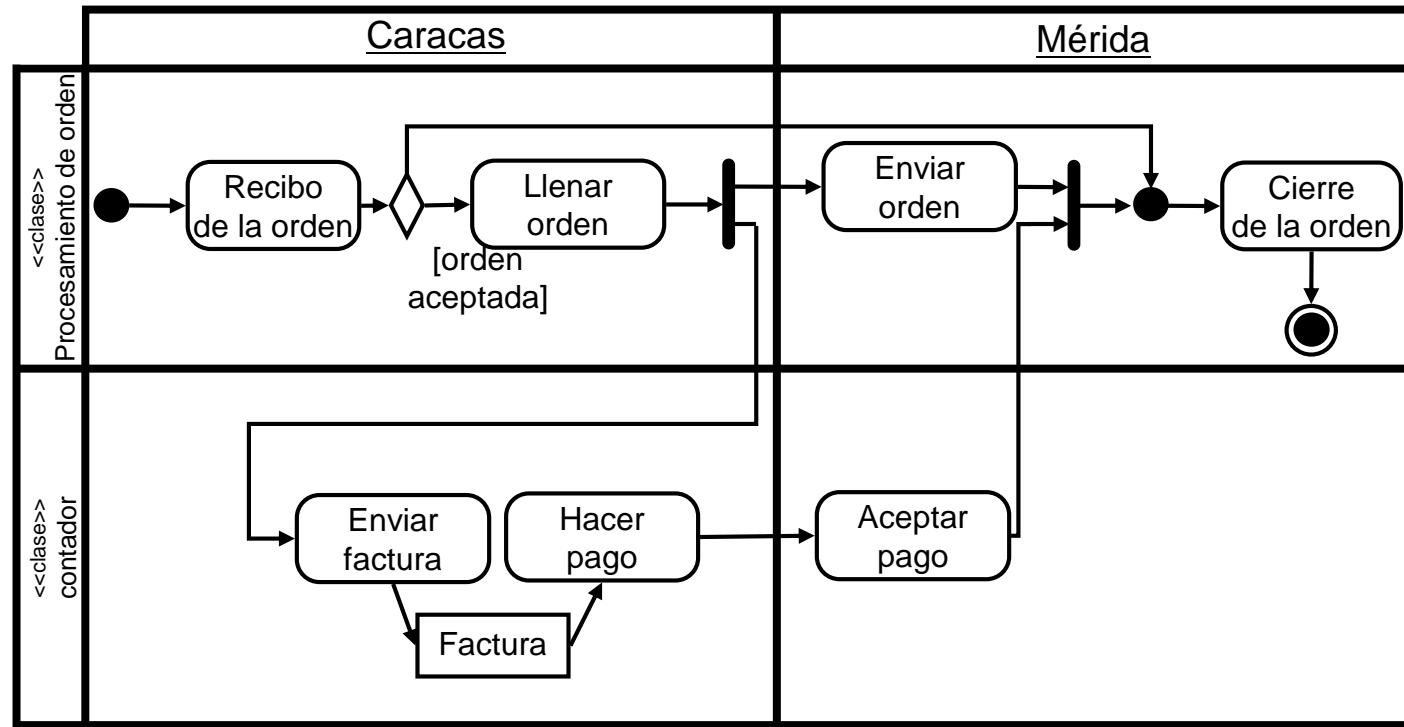


# Diagrama de actividades

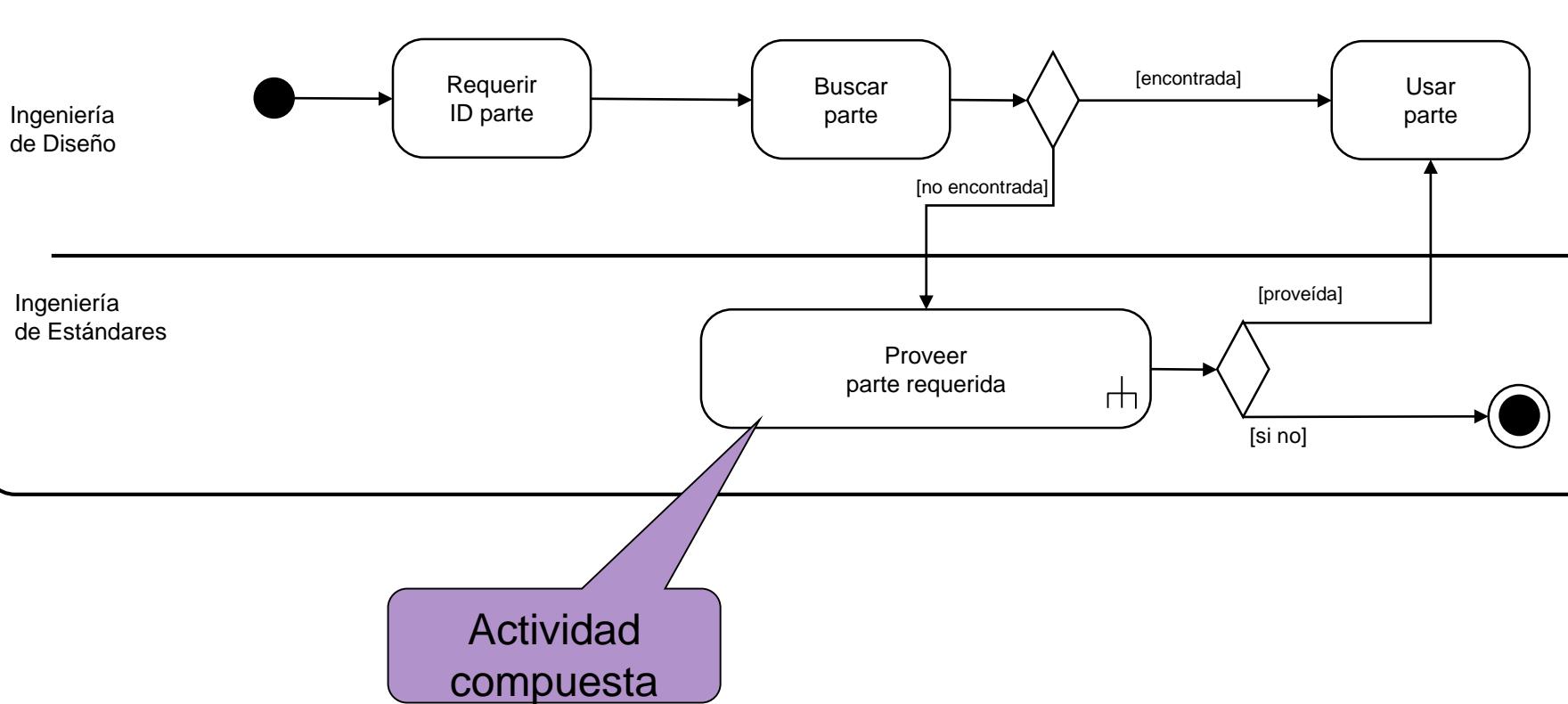
- ▶ **Capturan las acciones de una actividad y sus resultados**
  - ▶ Enfatizan la secuencia de acciones de una actividad
  - ▶ Establecen las condiciones para coordinar comportamiento de bajo nivel
  - ▶ Modelan el flujo de control y/o el flujo de objetos de una actividad



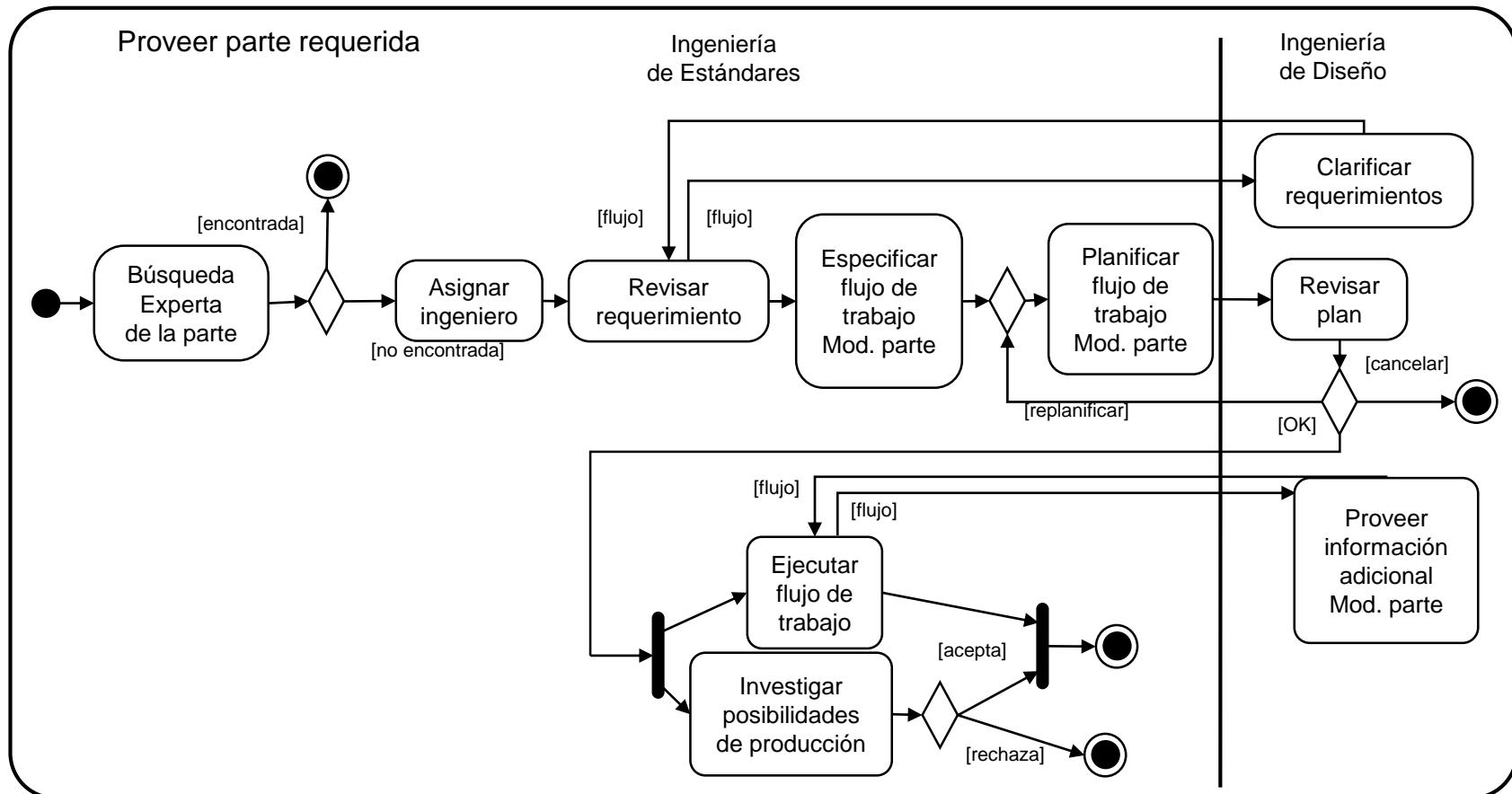
# Diagramas de actividades



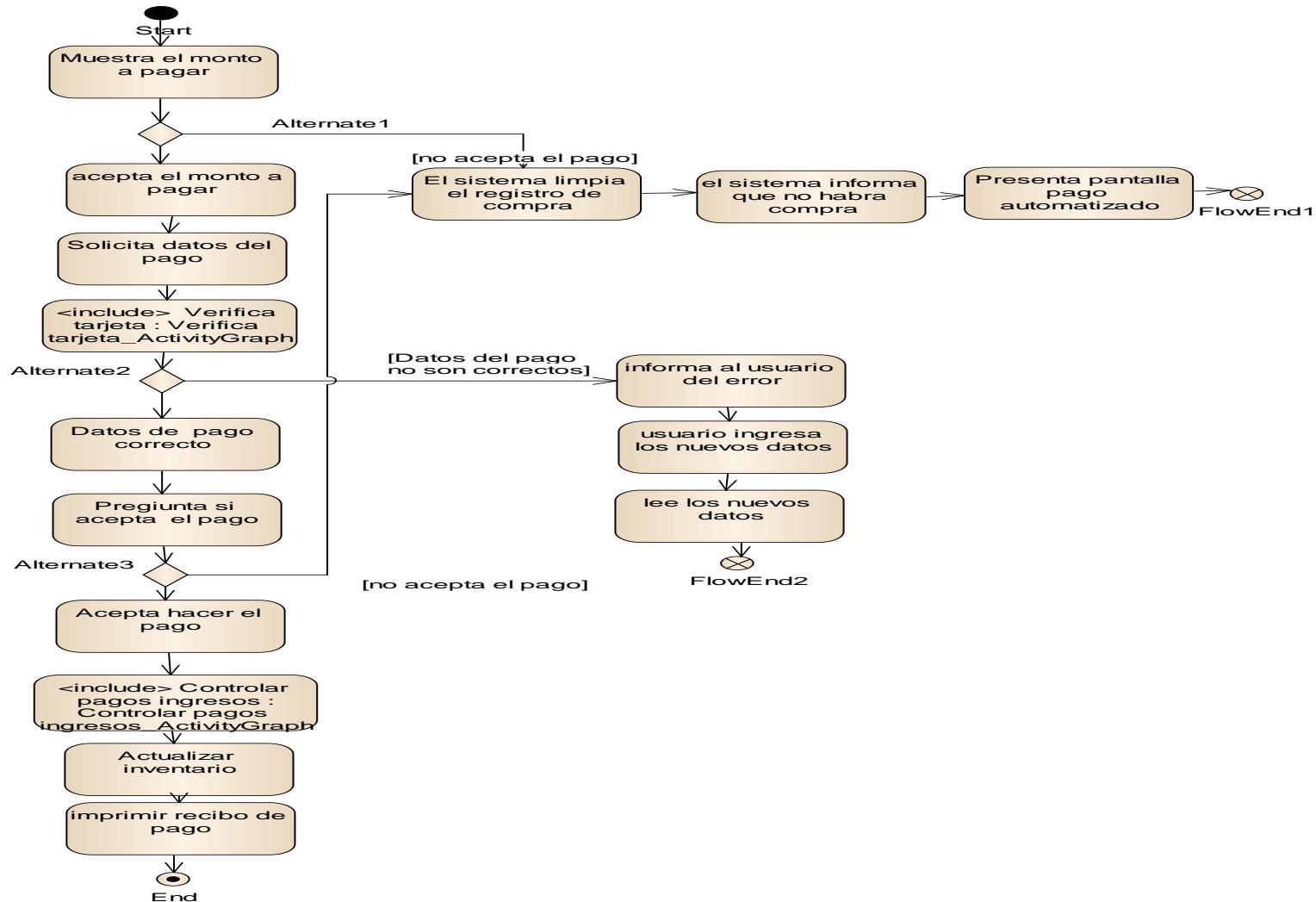
# Diagramas de actividades



# Diagramas de actividades



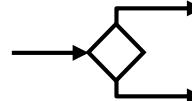
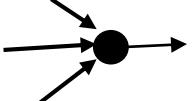
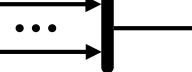
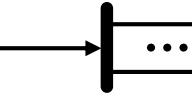
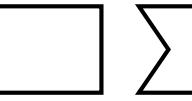
# Diagramas de actividades



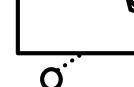
# Diagramas de actividades

Nombre	Símbolo	Descripción
Acción	nombre	Nodo de actividad Primitiva ejecutable de asignación o computación
Nodo de inicio	●	Nodo de control que indica el inicio de un flujo de control cuando una actividad es invocada
Nodo fin de actividad	○	Nodo de control que Indica el fin de todos los flujos dentro de una actividad. Muestra el fin de la actividad
Final de flujo	⊗	Nodo de control que muestra el fin de un flujo particular
Flujo de control	→	Eje de actividad para flujo de control. Conecta dos acciones. Usado para indicar secuencia
Flujo de objetos	→	Eje de actividad para flujo de objetos. Conecta acciones con nodos de objetos o pines

# Diagramas de actividades

Nombre	Símbolo	Descripción
Almacén de datos	<<datastore>> nombre	Nodo que representa un buffer o un almacén de información persistente (archivo o base de datos)
Nodo de decisión		Nodo de control que selecciona entre dos o más flujos de salida
Nodo de mezcla ( <i>merge</i> )		Nodo de control que une dos o más flujos alternativos
(Join) Nodo de concurrencia		Nodo de control que sincroniza múltiples flujos
(Fork) Nodo de sincronización		Nodo de control que divide un flujo en dos o más flujos concurrentes (paralelos)
Nodos de objeto		Nodo de objetos usados en los flujos de objetos. Representan objetos de datos o señales

# Diagramas de actividades

<i>Nombre</i>	<i>Símbolo</i>	<i>Descripción</i>
Conector	○	Usados para separar un flujo y reestablecer su conexión en un diagrama
Pin	□	Usado para indicar valores de entrada o salida en una acción
Manejador de excepciones		Nodo que indica la ejecución de una acción excepcional invocada desde otra acción
Notas		Nodo que permite agregar notas al diagrama (Ej. Pre y postcondiciones locales)
Acción señal de envío		Acción que indica la creación de una señal que es transmitida a otra acción receptora
Acción de señal de aceptación		Acción que espera la ocurrencia de un evento para disparar una señal a otra acción receptora
Acción evento de tiempo		Acción que espera la ocurrencia de un evento temporal (fecha/hora) para disparar una señal a otra acción receptora

# Diagramas de vistas de interacción

- ▶ **Modelan la interacción entre los objetos y relaciones de una aplicación, sistema, subsistema, programa o clase**
  - ▶ Muestran el flujo de mensajes entre los objetos de una aplicación ocasionados por una interacción
  - ▶ Una interacción es una relación entre los objetos y su entorno (actores)
    - ▶ Ejemplo, la activación de una función descrita por un caso de uso
- ▶ **La interacción se puede representar con los diagramas**
  - ▶ Secuencia
  - ▶ Comunicación
  - ▶ Vistas de interacción

**Modelado del comportamiento**

# Diagramas de secuencia

- ▶ **Muestran las interacciones entre los objetos desde el punto de vista temporal insistiendo en la cronología del envío de mensajes**
- ▶ Permiten mostrar las relaciones entre los objetos de la aplicación o programa y sus usuarios (actores)
- ▶ Los objetos y los enlaces creados o destruidos en el curso de una interacción pueden llevar las restricciones de: {nuevo}, {destruido}, {transitorio}, iteración sobre todos los objetos de un tipo: \*{todos}

**Modelado del comportamiento**

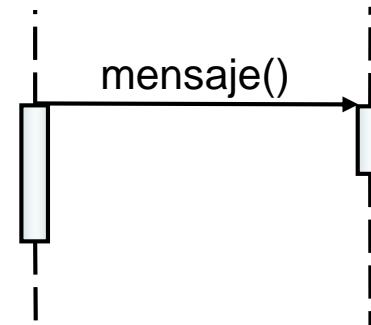
# Diagramas de secuencia

Objeto o actor

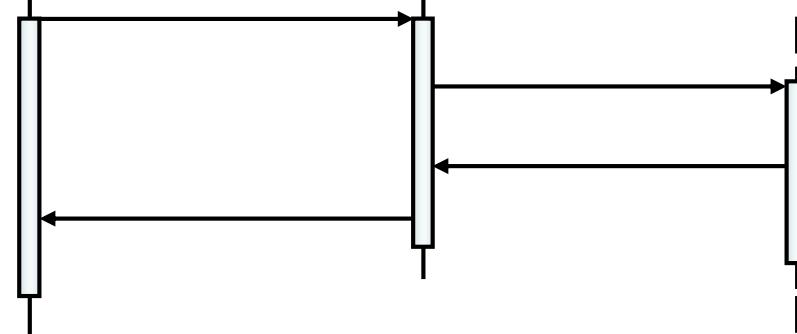
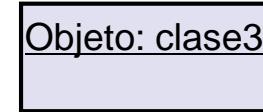
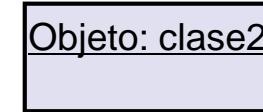
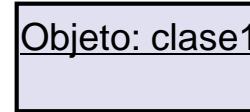
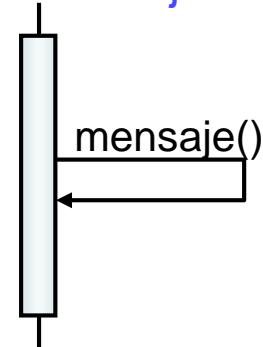


Línea de vida del objeto

Mensaje de un objeto a otro



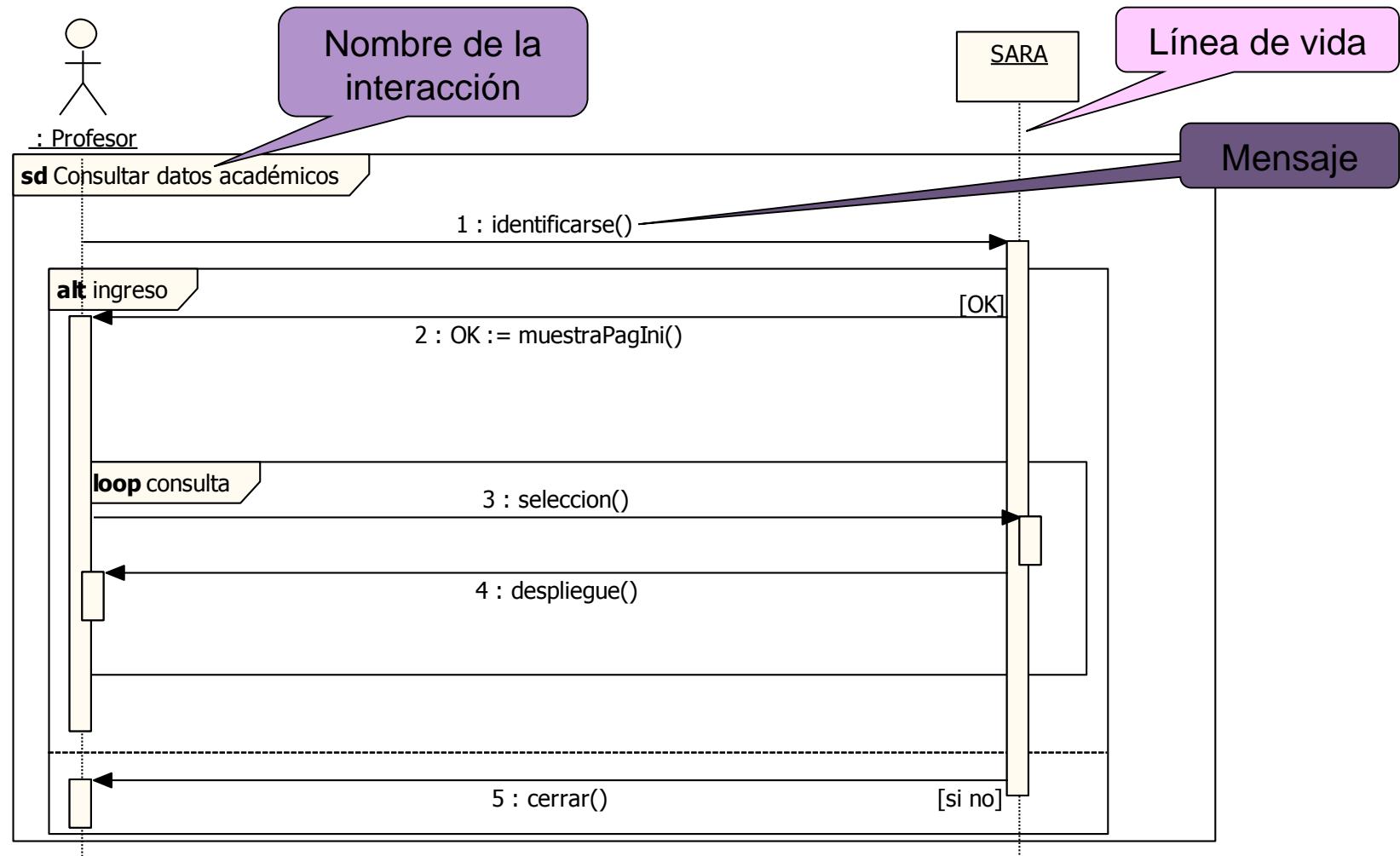
Mensaje a un mismo objeto



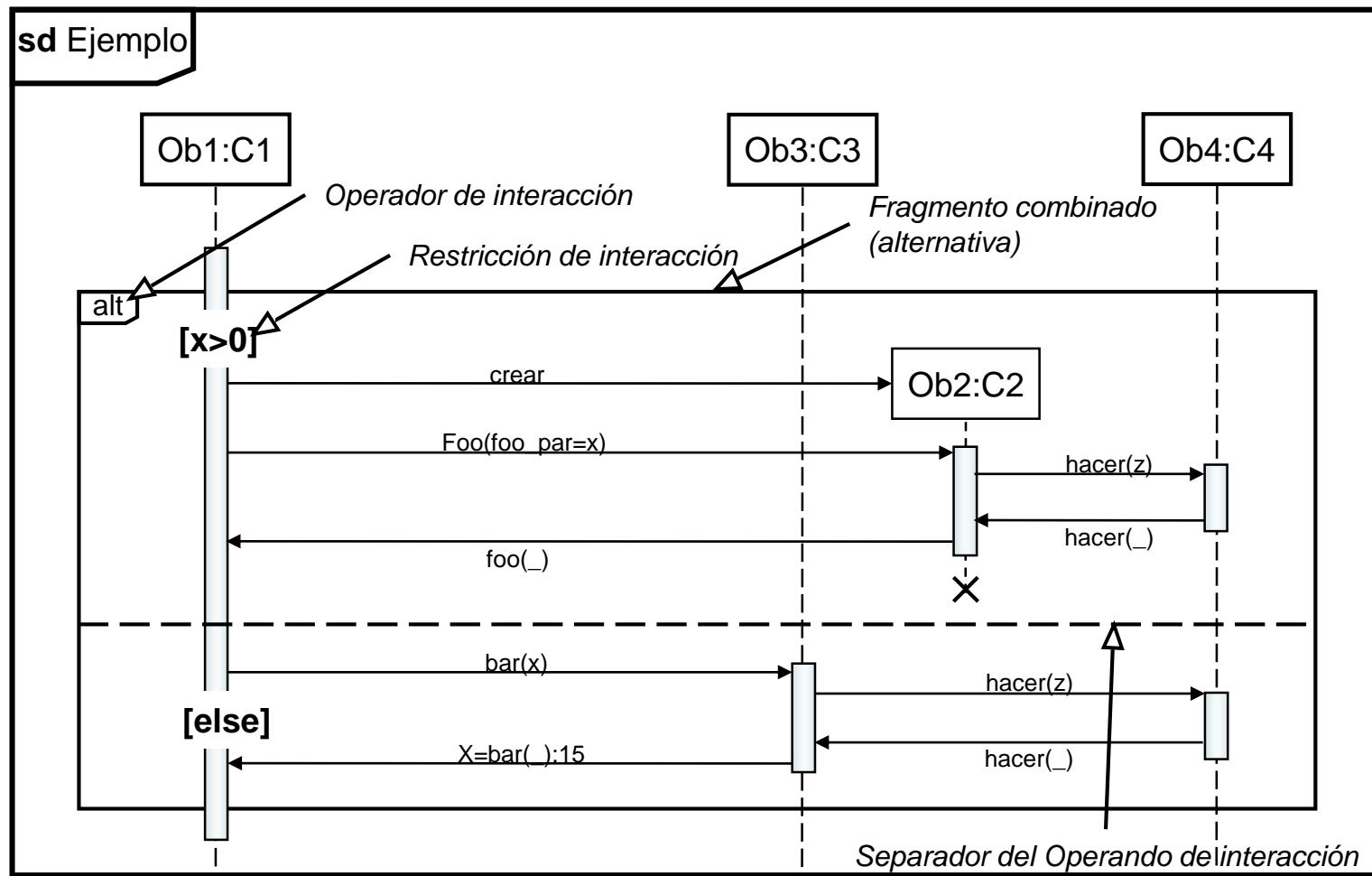
# Diagramas de secuencia

- ▶ **Describen la dinámica de una aplicación (o de una parte de ella)**
  - ▶ Muestran como los actores (humanos u otros sistemas) se relacionan con los objetos de la aplicación y cómo los objetos del programa se comunican entre sí mediante el pase de mensajes
  - ▶ La interacción entre objetos se muestra en forma secuencial (temporal) usando las líneas de vida (de arriba hacia abajo)
  - ▶ En sistemas interactivos, estos diagramas permiten mostrar el flujo de mensajes entre objetos que origina la activación de una función
- ▶ **Forma del mensaje:**  
***secuencia : resultado := mensaje(argumentos)***

# Diagrama de secuencia



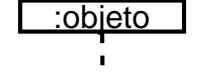
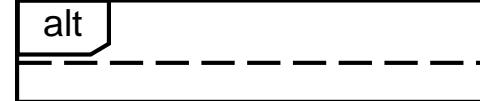
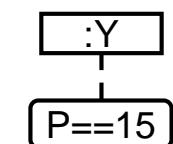
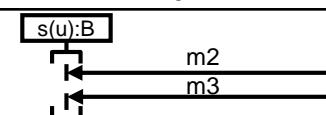
# Diagrama de secuencia



[OMG, 2003a]

## Fragmento combinado

# Diagramas de secuencia

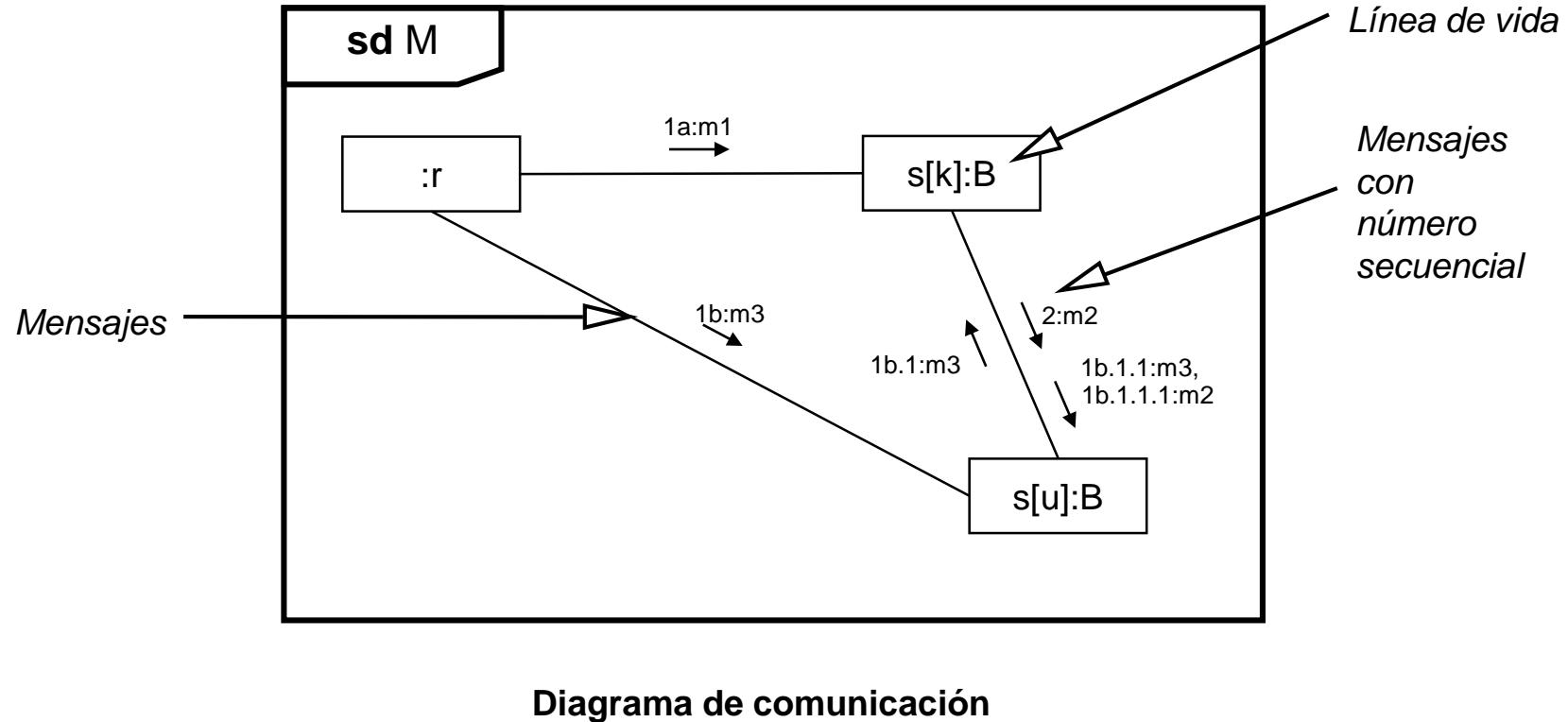
Nombre	Notación	Descripción
Marco (frame)		Recuadro para incluir el diagrama
Línea de vida		Línea de vida asociada al objeto
Ejecución		Indicación de la duración
Uso de la interacción		Uso de la interacción
Fragments combinados		Alternativas de interacción
Invariante de estado/combinación		Invariantes de estado o continuaciones
coregión		Región común
parada	X	Finalización

# Diagramas de comunicación

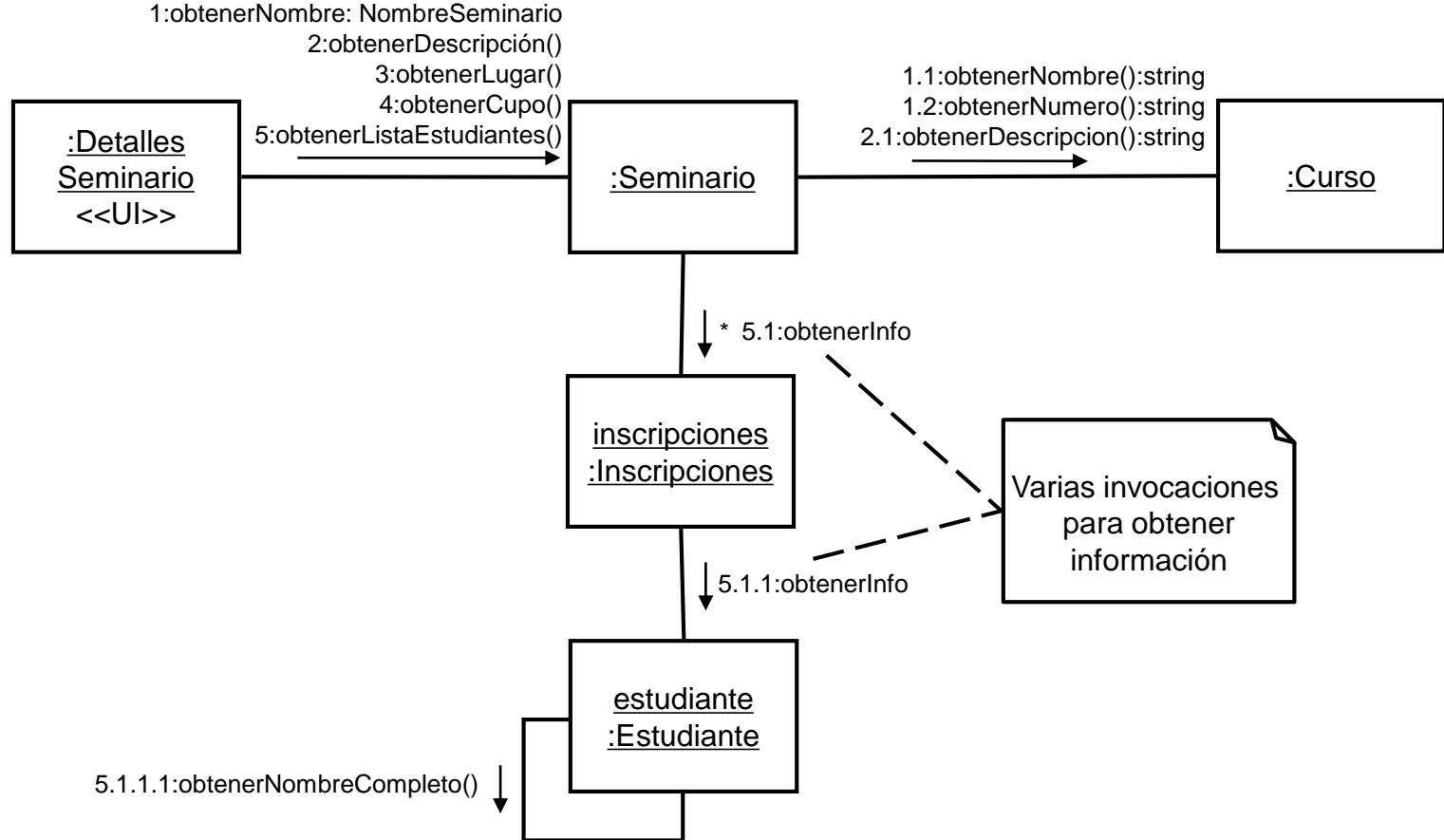
- ▶ **Expresan el contexto de un grupo de objetos (los objetos y sus enlaces) y la interacción entre ellos (envío de mensajes entre ellos)**
  - ▶ Muestran el flujo de mensajes entre los objetos de una misma o diferentes clases que participan en una interacción
  - ▶ Un diagrama de comunicación es un grafo dirigido cuyos nodos representan objetos y sus arcos los mensajes que se pasan entre los objetos
- ▶ **Hacen énfasis en la organización estructural de los objetos y el pase de mensajes entre ellos**

**Modelado del comportamiento**

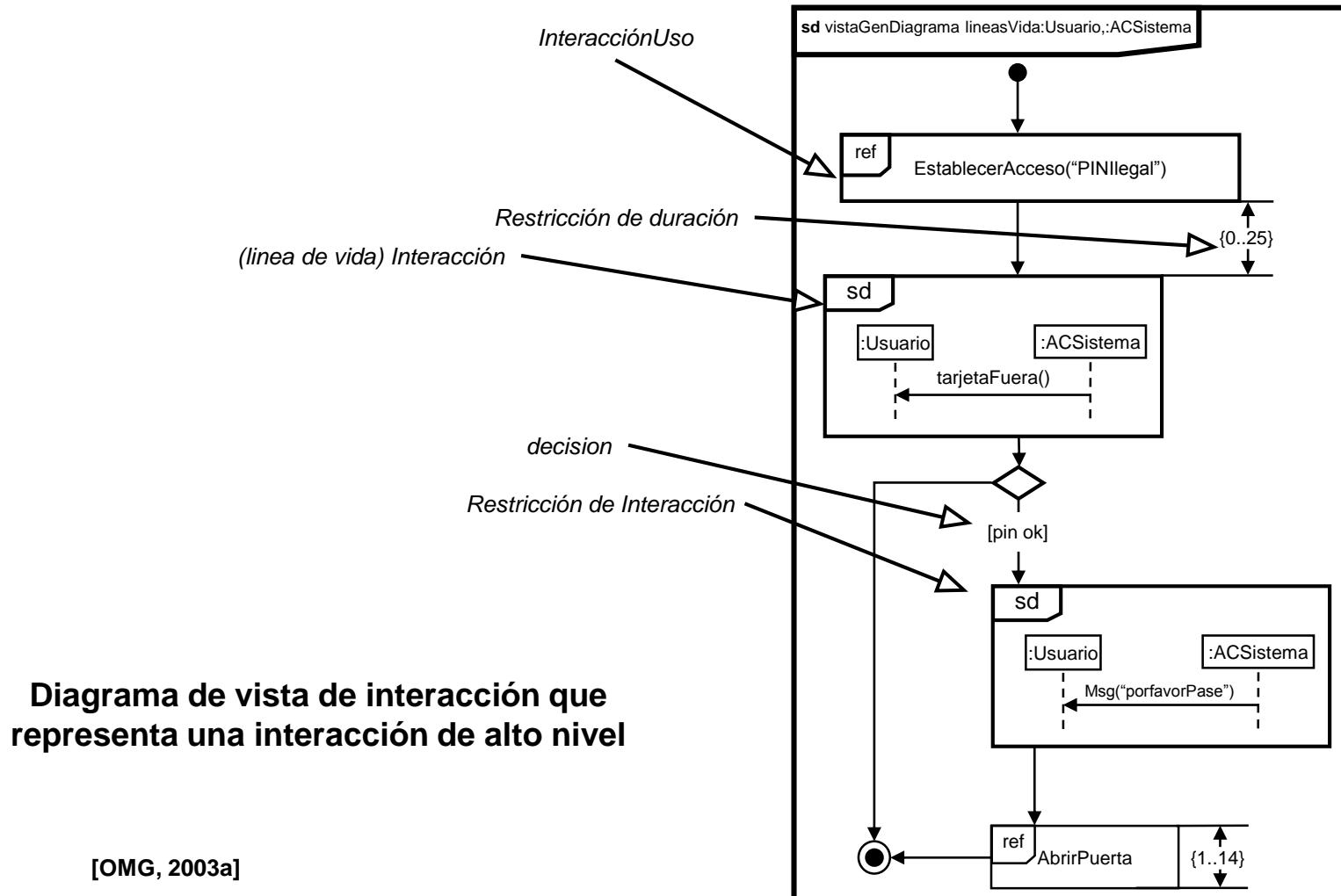
# Diagramas de comunicación



# Diagramas de comunicación



# Diagrama de vistas de interacción

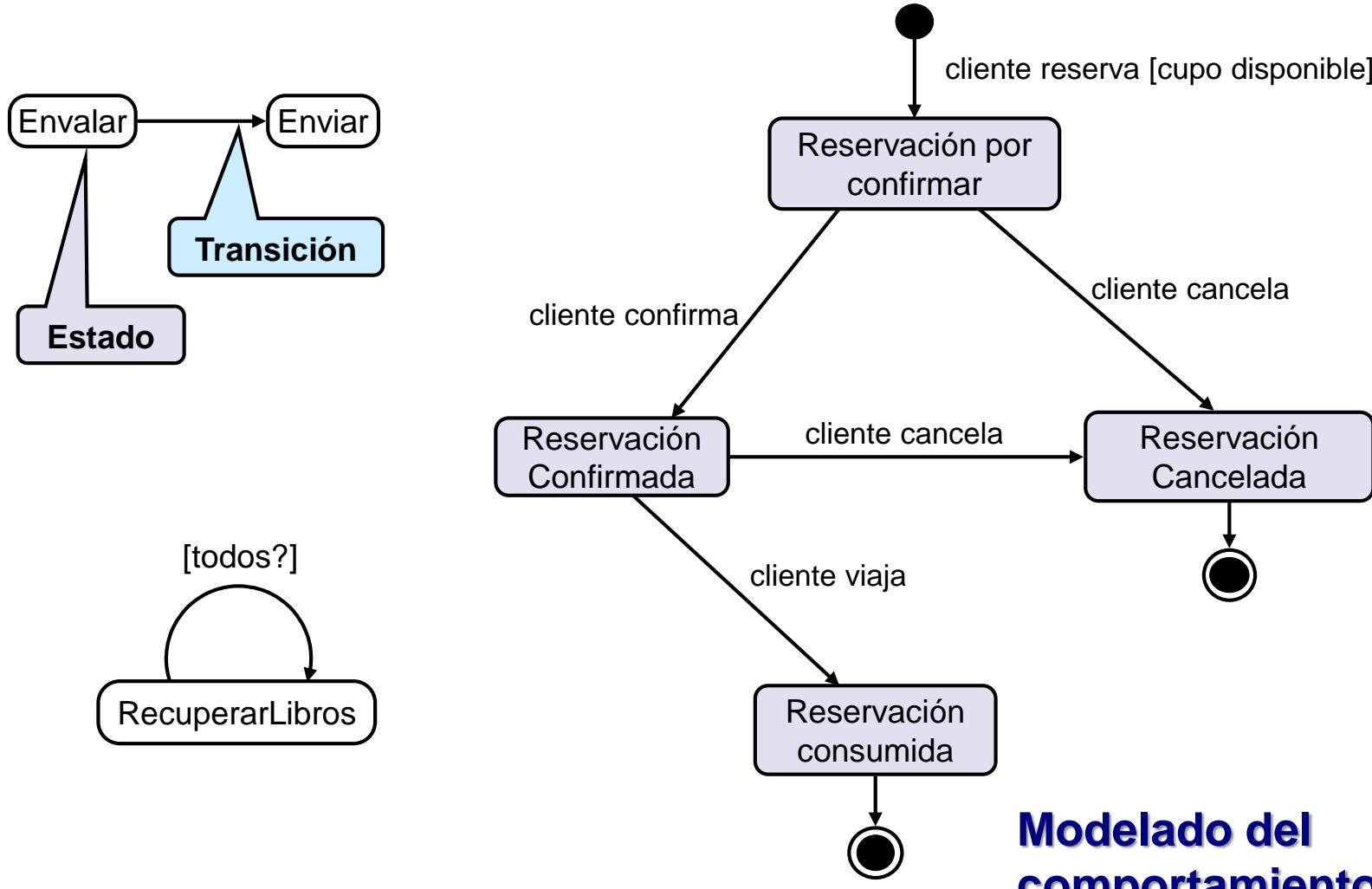


# Diagramas de máquinas de estado

- ▶ **Modelan la dinámica de los objetos de una sociedad**
- ▶ **A diferencia de los diagramas de interacción, los diagramas de estado modelan la dinámica de un objeto individual**
- ▶ **El objeto es modelado como una máquina de estado**

Una **máquina de estado** es un modelo que especifica las secuencias de estados que un objeto puede recorrer durante su existencia
- ▶ **Estado de un objeto:**
  - ▶ Conjunto de valores que tienen sus atributos y relaciones en instante de tiempo dado
- ▶ **Permiten modelar los diferentes estados de un objeto o de un sistema**
- ▶ **Muestran los eventos que ocasionan las transiciones de estado**
- ▶ **Los cambios de estado del objeto obedecen a los eventos que ocasionan la transición entre estados**

# Diagrama de máquinas de estado



**Modelado del  
comportamiento**

# Diagramas de máquina de estado

## ▶ **Objetos:**

- ▶ Siguen globalmente el comportamiento descrito en el autómata asociado a su clase y se encuentra en un momento dado en un estado que caracteriza sus condiciones dinámicas
  - ▶ se comportan como elementos pasivos controlados por los eventos que provienen del sistema
  - ▶ que no presentan un comportamiento reactivo muy marcado se pueden considerar como objetos que siempre están en un mismo estado
- ▶ **Un estado es siempre la imagen de la conjunción instantánea de los valores contenidos en sus atributos y la presencia o no de sus enlaces a otro objetos**

# Diagramas de máquina de estado

## ▶ Evento:

- ▶ se corresponde con la ocurrencia de una situación dada en el dominio del problema, es por naturaleza una información instantánea que debe ser tratada en forma inmediata, sin espera
- ▶ sirve de disparador para pasar de un estado a otro, donde las transiciones indican el camino en el digrafo de estados y los eventos determinan que camino seguir

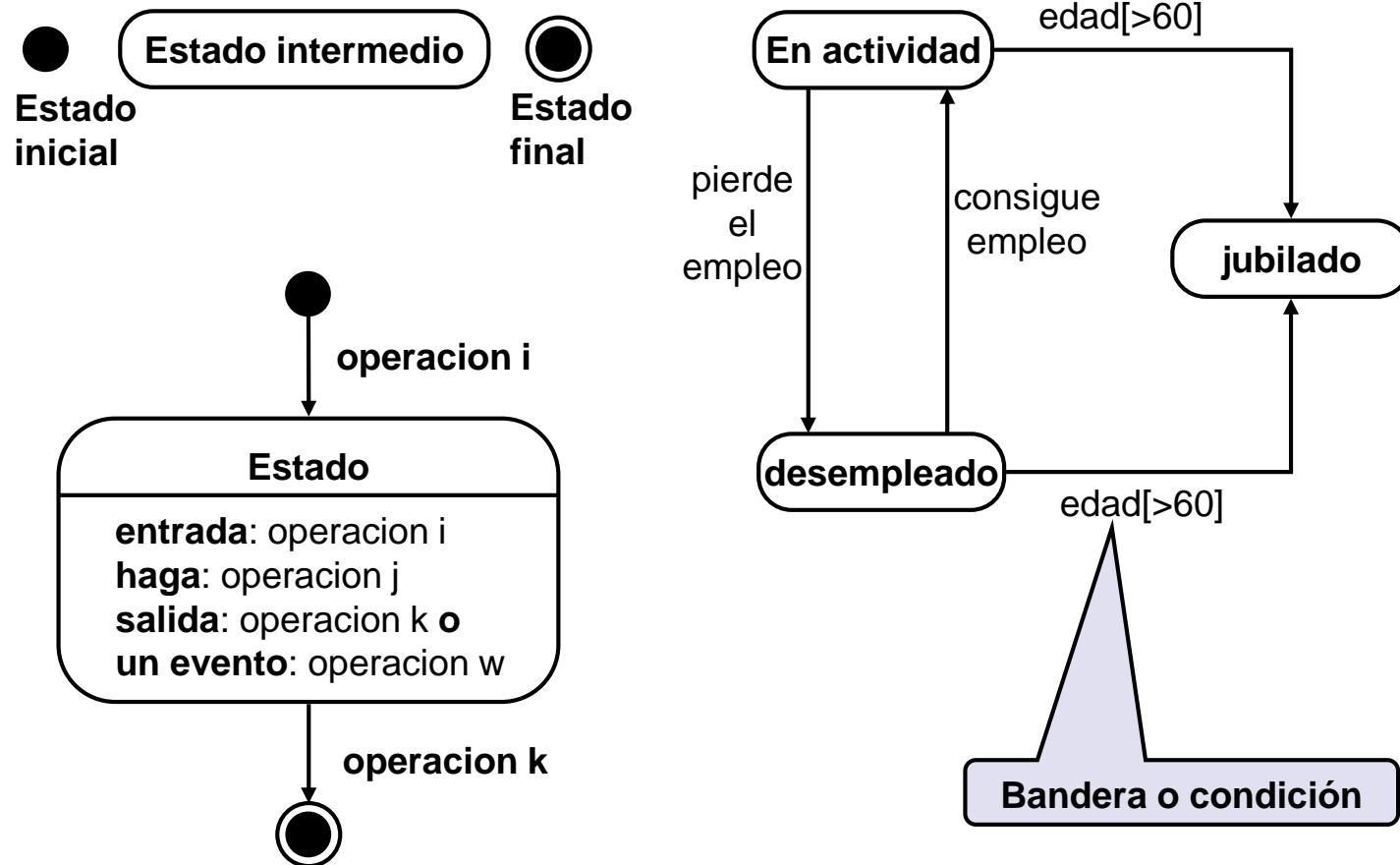
## ▶ Sintaxis de un evento:

**NombreDelEvento(nombreDelParámetro : tipo, ...)**

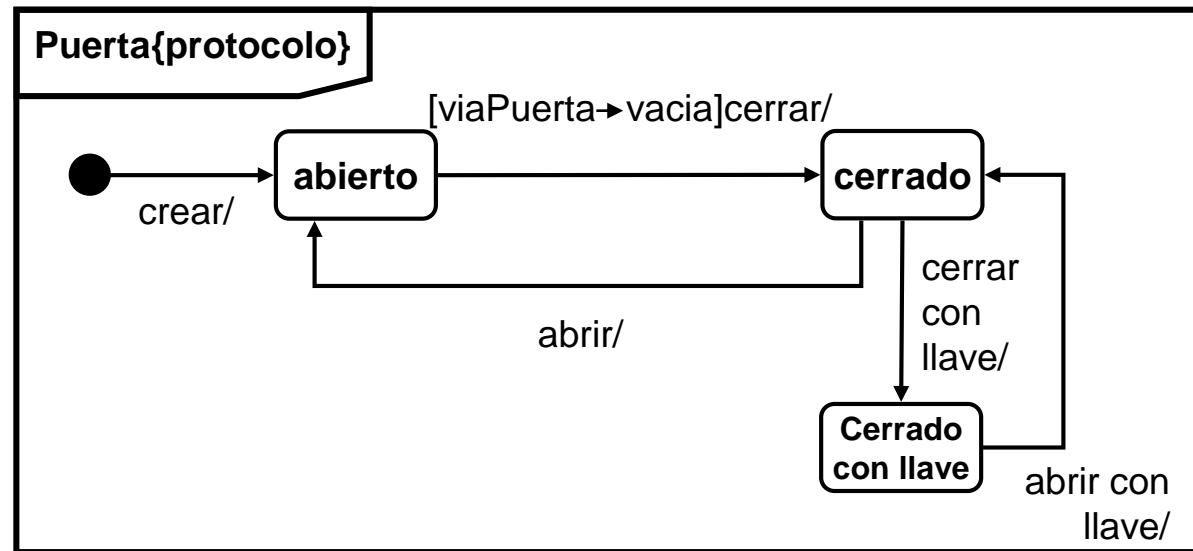
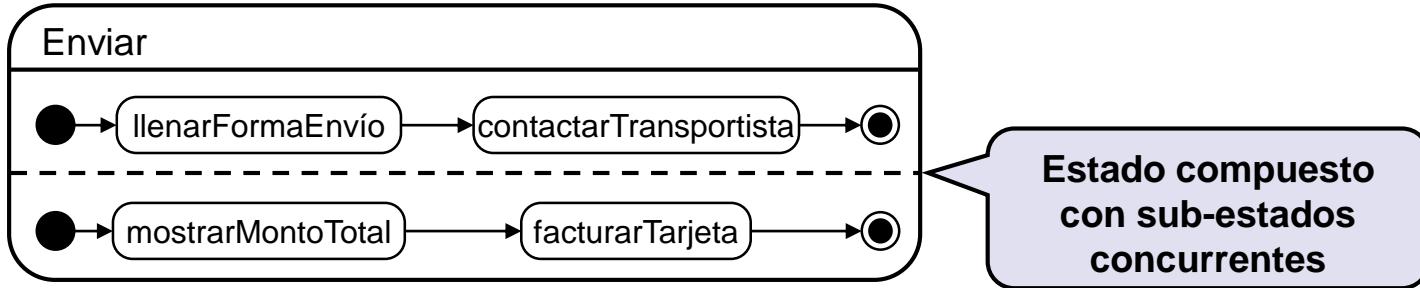
## ▶ La especificación completa de un evento comprende:

- ▶ *El nombre del evento*
- ▶ *La lista de parámetros*
- ▶ *El objeto fuente*
- ▶ *El objeto destino*
- ▶ *La descripción del significado del evento*

# Diagrama de máquina de estado

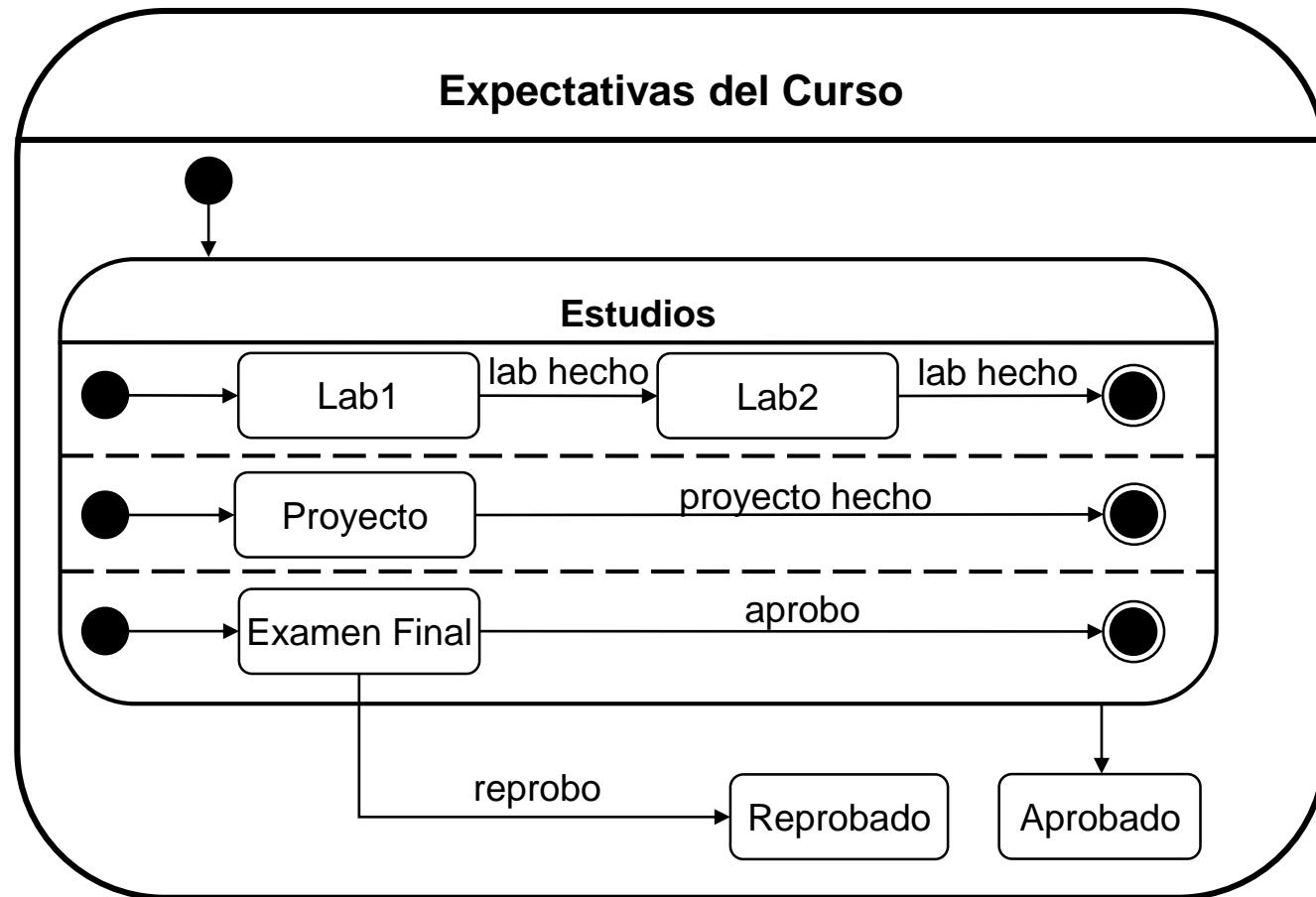


# Diagrama de máquina de estado



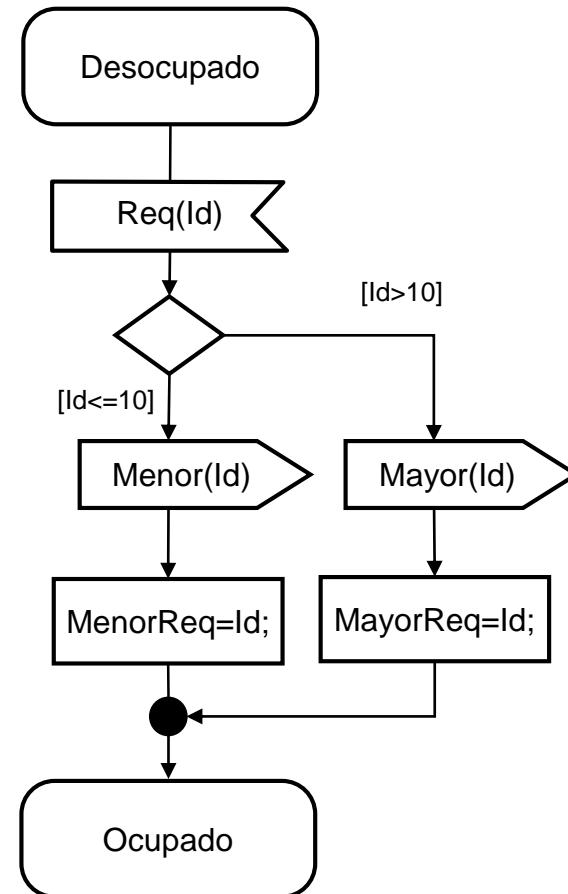
Protocolo de máquina de Estado

# Diagrama de máquina de estado



**Estado ortogonal con regiones**

# Diagrama de máquina de estado

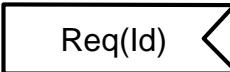
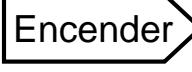
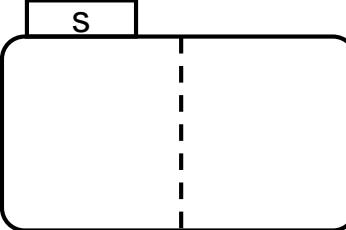


**Símbolos para señalar recepción, envío y acciones de transición**

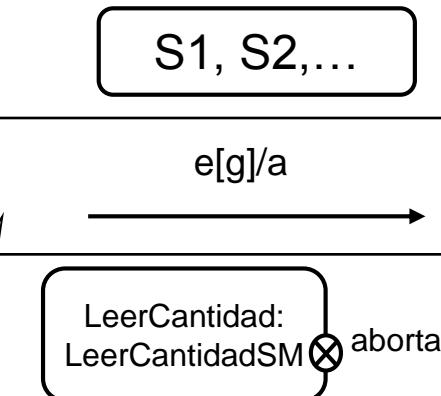
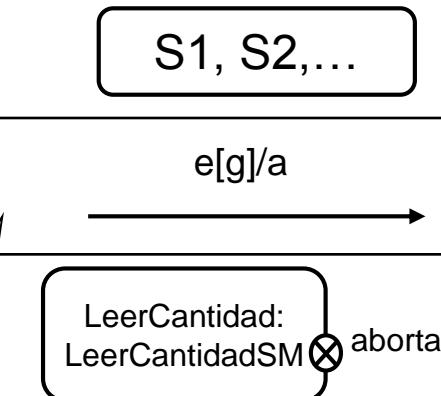
# Diagrama de máquina de estado

Nombre	Notación	Descripción
acción	req:d	Acciones
escogencia		Decisiones dobles
Estado compuesto		Estado compuesto de subestados
entrada	○ otra vez	Punto de entrada
salida	⊗ abortar	Punto de salida
Estado inicial	●	Estado inicial del diagrama
Estado final	○	Estado final del diagrama

# Diagrama de máquina de estado

Nombre	Notación	Descripción
estado		Estados del diagrama
historia		Recordar el último estado
Historia completa		Recordar todos los estados
Unión		Punto de unión
recepción		Recepción de señal
envío		Envío de señal
región		Regiones del diagrama

# Diagrama de máquina de estado

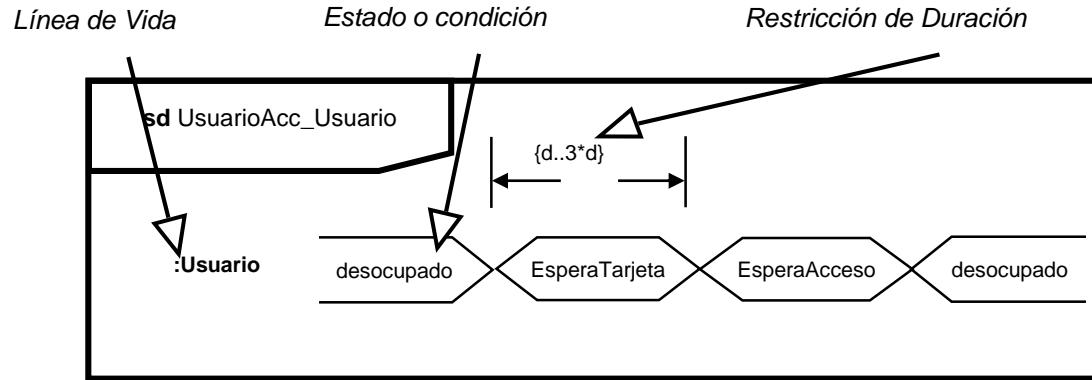
Nombre	Notación	Descripción
Lista de estados	S1, S2,...	Listado de estados
transición	e[g]/a	Transiciones de estado
Submáquina de estado	 <pre> graph LR     A[LeerCantidad: LeerCantidadSM] --&gt; B[abortar]   </pre>	Indica un diagrama de estado de otro diagrama de estado
 <pre> graph LR     A[LeerCantidad: LeerCantidadSM] --&gt; B[abortar]   </pre>		<p>evento[condición]/acción guardia = condición</p>

# Diagramas temporales

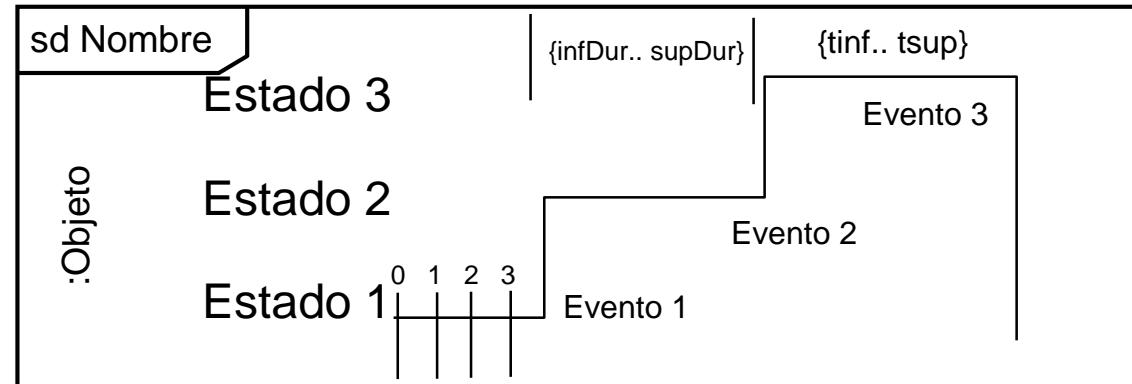
- ▶ **Describe el comportamiento de objetos e interacciones entre ellos, enfocándose en el tiempo de ocurrencia de los eventos que causan los cambios de estado**
- ▶ **Elementos**
  - ▶ Marcas
  - ▶ Restricciones de duración: intervalos de duración
  - ▶ Restricciones de tiempo: intervalos de tiempo {t1..t2}

**Modelado del  
comportamiento**

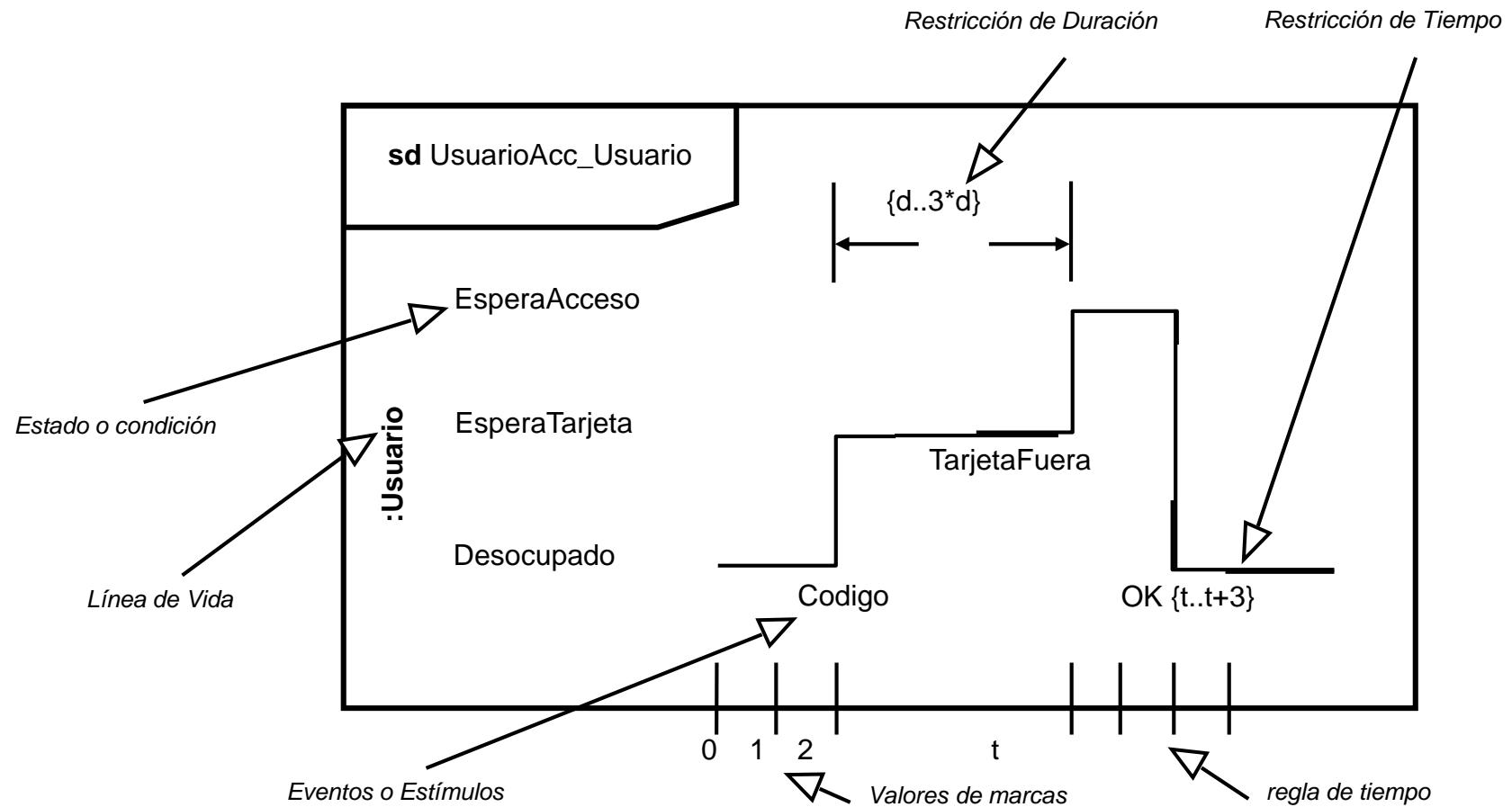
# Diagramas temporales



**Línea de Vida compacta con Estados**

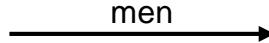
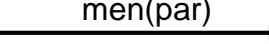
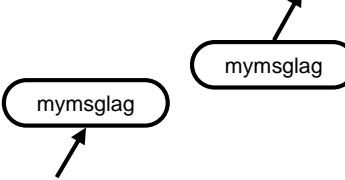
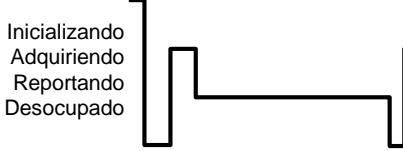


# Diagramas temporales



Una Línea de Vida para objetos discretos

# Diagramas temporales

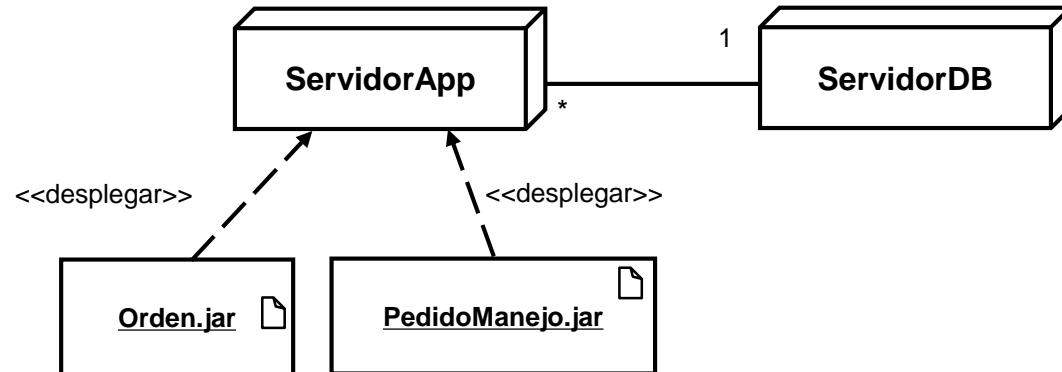
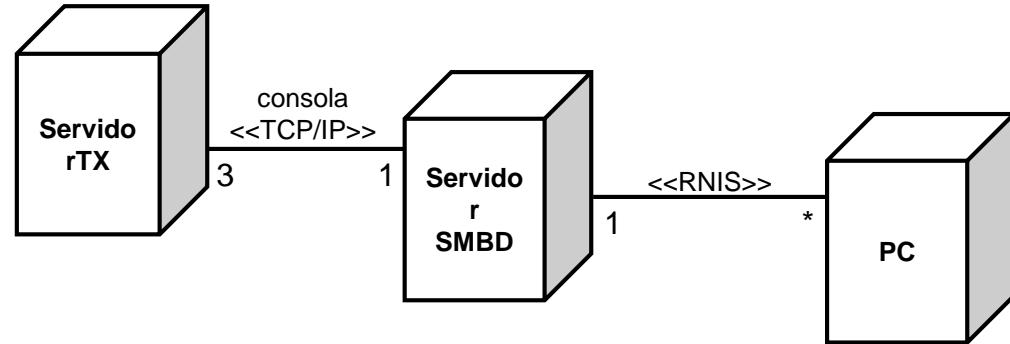
Nombre	Notación	Descripción
Mensaje asíncrono		Mensaje asíncrono entre estados
mensaje		mensaje
etiquetas		Etiquetas de mensaje
Estados o condiciones		Estados o condiciones de la línea de tiempo
Valor		Valor de la línea de tiempo
Línea de tiempo		Indica la línea temporal

# Vista de despliegue

- ▶ **Describe como el sistema se instala en su ambiente de operación y como ejecuta en una red**
- ▶ **UML:**
  - ▶ se expresa a través de los diagrama de despliegue
- ▶ **Diagramas de despliegue**
  - ▶ Muestran la disposición física del hardware (nodos) que conforman la composición de un sistema y la distribución de los programas ejecutables en dicho hardware

**Modelado estructural**

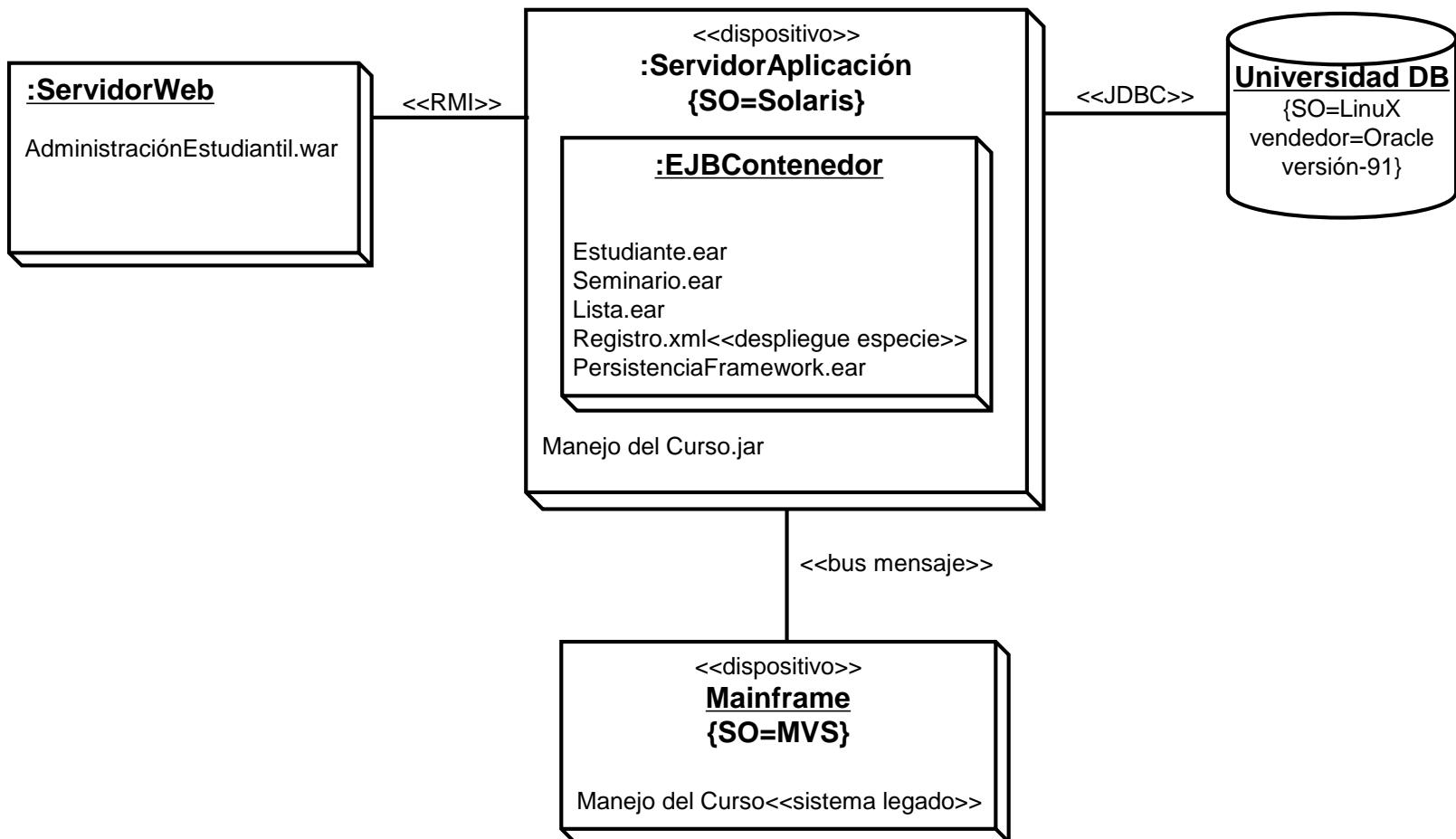
# Diagramas de despliegue



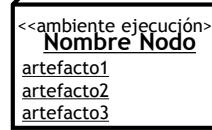
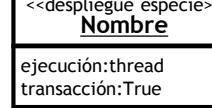
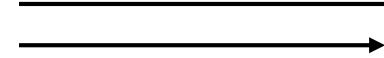
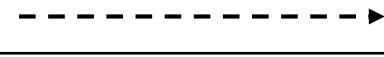
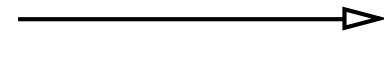
# Diagramas de despliegue

- ▶ **Todo sistema se describe por un pequeño número de diagramas de despliegue y normalmente, uno solo es suficiente**
- ▶ **Los cubos pueden relacionarse por asociaciones, a priori bidireccionales, donde la naturaleza del soporte se puede indicar con un estereotipo, i.e. <<Conexión>>, <<TCP/IP>>, etc.**
- ▶ **Un determinado cubo puede representar clases de hardware o uno en específico, colocando su nombre subrayado**

# Diagrama de despliegue



# Diagrama de despliegue

Nombre	Notación	Descripción
artefacto		Indica un artefacto
nodo		Indica un nodo
Nodo con artefactos		Nodo con artefactos desplegados
Especificación de despliegue		Especificación de despliegue con propiedades y valores
asociación		Relación
dependencia		Relación de dependencia
generalización		Relación de generalización/especialización
despliegue		Relación de despliegue
manifestación		Relación de manifestación

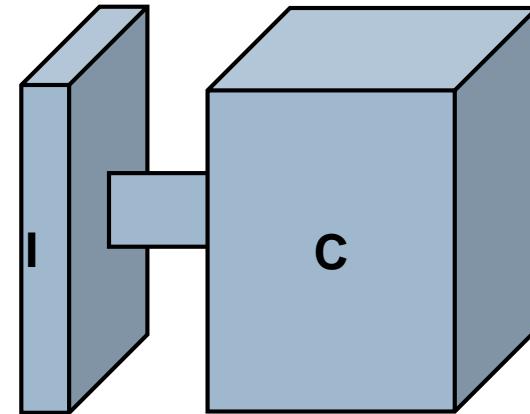
# Vista de implementación

- ▶ **Describe la estructura de módulos, archivos y paquetes que integran el sistema**
- ▶ **UML: se expresa con los diagramas de paquetes y de componentes**
- ▶ **Programas:**
  - ▶ Los programas principales que son los puntos de entrada en las aplicaciones se identifican con un nombre que se utiliza para asociar el programa ejecutable con la aplicación
  - ▶ Esto permite asociar el modelo de componentes con el modelo de procesos
  - ▶ Los subprogramas reagrupan los procedimientos y las funciones que no pertenecen a ninguna clase

## Modelado estructural

# Componentes de software

- ▶ **Un componente de software es un artefacto de software auto-contenido y claramente identifiable que:**
  - ▶ ejecuta funciones específicas,
  - ▶ tiene una interfaz clara a través de la cual se integra a otros sistemas,
  - ▶ tiene una documentación apropiada
  - ▶ puede ser reutilizado en múltiples aplicaciones



# Componentes de software

## ▶ Definición del CBDi Forum [1999]:

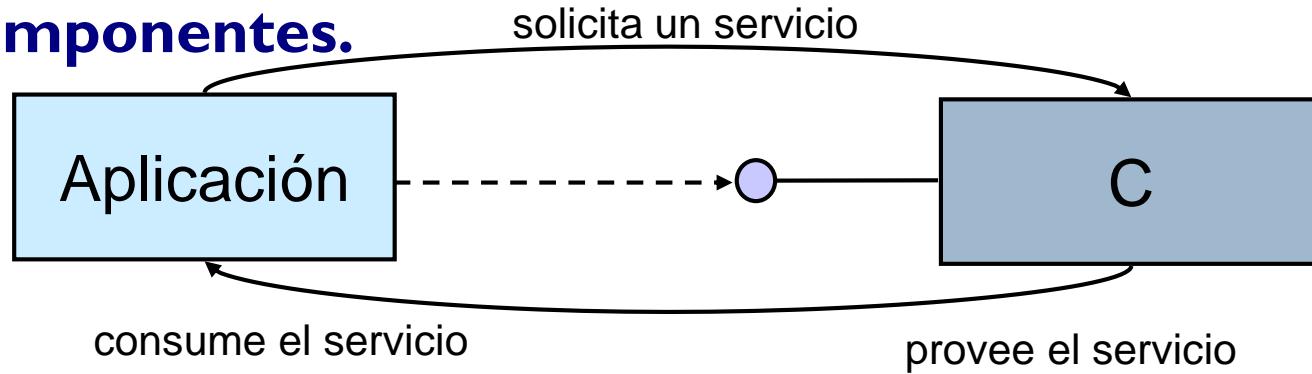
- ▶ “Un componente es una pieza de software que describe y/o libera un conjunto de servicios que son usados sólo a través de interfaces bien definidas”

## ▶ Características esenciales de un componente:

- ▶ Identifiable
- ▶ Autocontenido
- ▶ Rastreable a través de su ciclo de desarrollo
- ▶ Reemplazable por otro componente
- ▶ Accesible solamente a través de su interfaz
- ▶ Inmutabilidad de sus servicios
- ▶ Documentación de sus servicios
- ▶ Mantenido sistemáticamente

# Interfaz de un componente

- ▶ Una interfaz de un componente determina la manera en que él se reutiliza y se conecta a otros componentes.

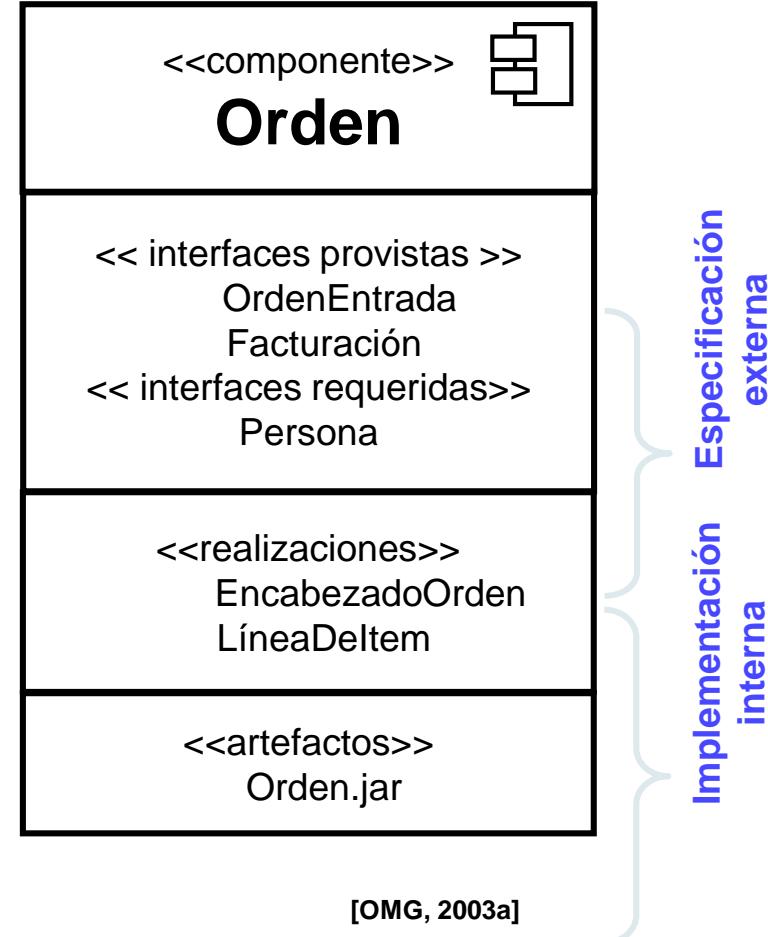


- ▶ Una interfaz define un conjunto de operaciones
  - ▶ Estas operaciones son llamadas, también, servicios o responsabilidades
  - ▶ Un componente produce un resultado que es consumido por otro componente o una aplicación que solicita sus servicios

# Diagramas de componentes

## ▶ Componentes de software

- ▶ Un componente de software es una clase especializada (clasificador) que tiene:
  - ▶ Una **especificación externa**:
    - interfaces requeridas
    - interfaces provistas
  - ▶ Una **implementación interna**
    - Consta de uno o más clasificadores (Ej. Clases) que implementan el comportamiento del componente

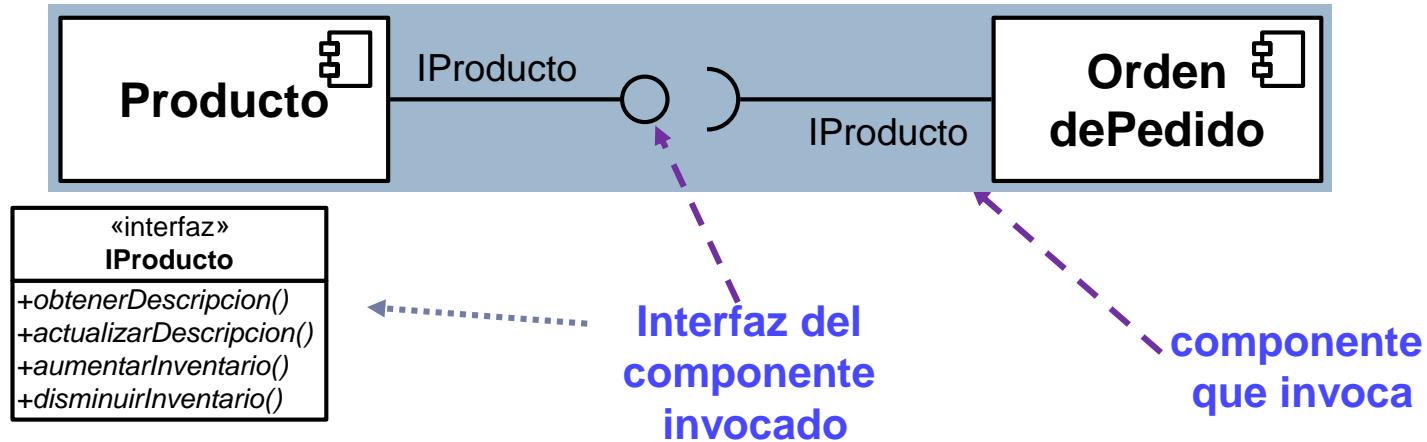


[OMG, 2003a]

# Diagramas de componentes

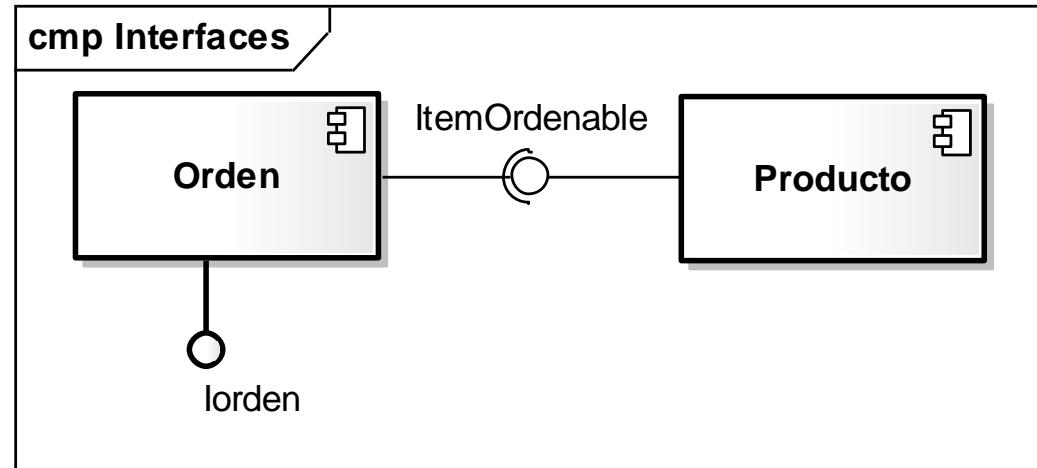
## ▶ Componentes de software

- ▶ Los detalles internos de un componente se ocultan a los otros componentes que lo utilizan (invocan)
- ▶ Los componentes usuarios sólo requieren conocer la interfaz del componente invocado
  - ▶ Las operaciones del componente invocado están disponibles a los componentes usuarios sólo a través de la interfaz



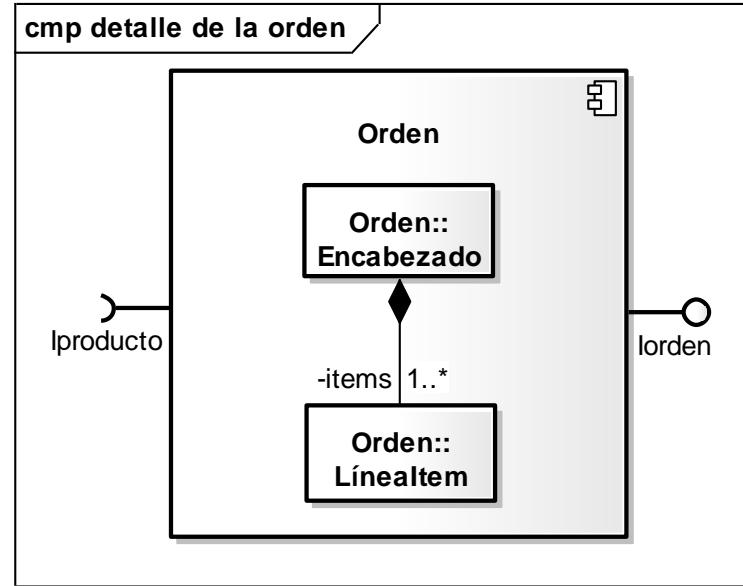
# Diagramas de componentes

- ▶ **Ensamblaje o composición de componentes**
  - ▶ Las interfaces provistas y requeridas son los puntos de ensamblaje de una aplicación basada en componentes

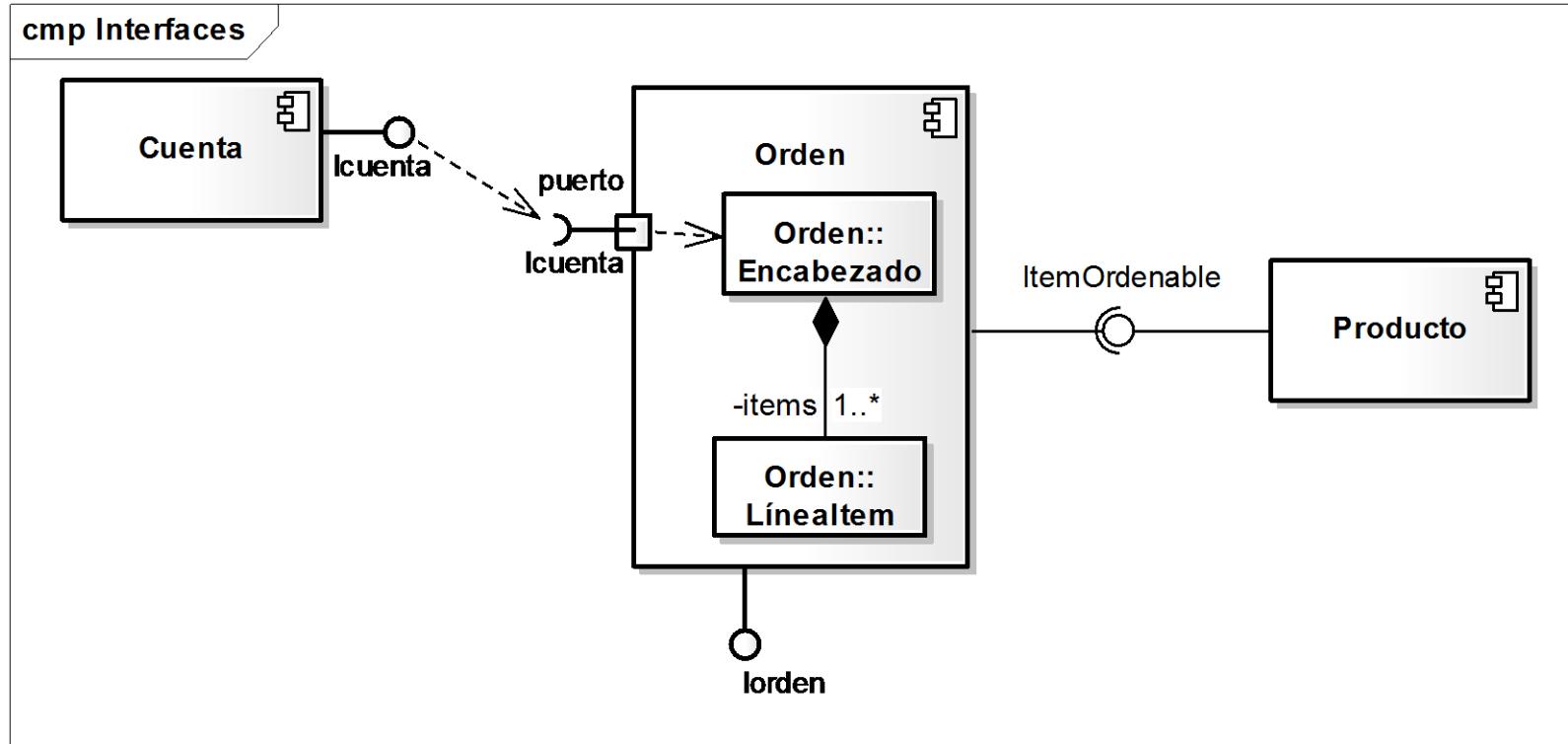


# Diagramas de componentes

- ▶ **Representación anidada de la realización (implementación) de un componente**
  - ▶ Muestra los elementos internos de un componente:
    - ▶ Los clasificadores que realizan el comportamiento del componente



# Dependencias entre componentes



# Autoevaluación

1. **¿Qué es un diagrama de actividades y cuáles son sus elementos?**
2. **¿Qué es un diagrama de secuencias y cuáles son sus elementos?**
3. **¿Qué es un diagrama de comunicación y cuáles son sus elementos?**
4. **¿Qué es un diagrama de vistas de interacción y cuáles son sus elementos?**
5. **¿Qué es un diagrama de máquinas de estado y cuáles son sus elementos?**
6. **¿Qué es un diagrama temporal y cuáles son sus elementos?**
7. **¿Qué es un diagrama de despliegue y cuáles son sus elementos?**
8. **¿Qué es un diagrama de componentes y cuáles son sus elementos?**

# Ejercicios

**Realice para la descripción dada:**

1. **Al menos uno de cada uno de los diagramas de comportamiento (actividad, secuencia, comunicación, vista de interacción, máquina de estado y temporal)**
2. **Al menos uno de cada uno de los diagramas estructurales (despliegue y componentes)**
3. **La base de datos debe contener información sobre los pacientes, los médicos y los exámenes realizados a los pacientes. Los pacientes tienen número de historia, nombre, dirección, teléfono, y fecha de nacimiento. Los médicos tienen nombre, dirección, teléfono, especialidad y número del colegio de médicos. Los exámenes tienen número de examen, tipo, fecha de realización, resultado y fecha de entrega. Los exámenes son realizados a los pacientes por solicitud del médico que tratan al paciente. Los pacientes acuden al médico previa cita. Durante la cita, el médico anota en la historia del paciente el motivo de la consulta, la fecha, el diagnóstico, el tratamiento y los exámenes que le solicitó. Un paciente sólo consulta a un médico de la clínica, pero puede hacerlo en varias consultas. Un médico puede solicitar el mismo examen a un paciente, pero en fechas diferentes**

# Ejercicios

2. **La base de datos debe contener información sobre los libros, revistas, manuales, carpetas, fotocopias, mapas y DVDs de una librería. Cada libro tiene ISBN, título, autores, editorial, año de publicación, costo y precio de venta. Las revistas tienen ISBN, título, editorial, número, año, costo y precio de venta. Los manuales tienen ISBN, título, autor, editorial, año, costo y precio de venta. Las carpetas tienen tipo, tamaño, color, costo y precio de venta. Las fotocopias tienen tipo de papel y precio por copia. Los mapas tienen código, título, escala, tamaño, costo y precio de venta. Los DVD tienen código, título, autores, productora, distribuidora, año, costo y precio de venta. Cada uno de los productos se venden a los clientes generando una factura que tiene un número, fecha, nombre, dirección, teléfono y cédula de identidad o RIF del cliente, descripción y cantidad de cada producto vendido, el monto total, el IVA y el monto a pagar.**

# Ejercicios

3. Los organizadores del mundial de Futbol desean tener un sistema de BD para registrar los datos del evento. Se deben almacenar los partidos jugados, los resultados de cada partido, los jugadores de cada partido y las estadísticas individuales de cada jugador en cada partido. Cada jugador tiene número, nombre, apellido, edad y nacionalidad. Cada partido tiene un identificador, día y hora del encuentro, lugar donde se celebra, equipos que se enfrentan, árbitros que lo regulan, alineación de cada equipo, resultado del partido con los detalles del resultado (jugador que anotó y en que tiempo lo hizo). Los árbitros tienen nombre, apellido, edad, nacionalidad y nivel de experticia. Entre las estadísticas se tienen: por cada equipo los goles a favor y en contra, el número de tarjetas amarillas y rojas, el jugador con mayor número de goles por cada equipo y en el mundial, etc. Realice cualquier consideración que crea conveniente en forma razonada.