

## **TEMA 3.5:**

# **DIAGRAMAS DE ESTADO**

# Índice

- Diagramas de estado
  - Representación gráfica
  - Elementos
  - Tipos de eventos
  - Tipos de estados

# Diagramas de estado

- Los diagramas de estado se utilizan para **modelar el comportamiento** de un **único objeto**, están **orientados a eventos**
- Muestran cómo las partes de un modelo UML **cambian con el tiempo**
- Al pasar el tiempo y según **suceden los acontecimientos** hay cambios que afectan los objetos que nos rodean
- Las interacciones de un sistema con los usuarios y con otros sistemas provocan una serie de cambios en los objetos que lo conforman

# Diagramas de estado

- Una forma de caracterizar un cambio en un sistema es decir que los objetos que lo componen modificaron su estado como respuesta a una serie de sucesos y al tiempo:
  - Cuando se acciona el interruptor la luz cambia del estado apagada a encendida
  - Cuando se presiona un botón del mando a distancia, la televisión cambia su estado para mostrar otro canal

# Diagramas de estado

- El **diagrama de estados** en UML **captura los cambios**
- Presenta los **estados** en los que puede encontrarse un **objeto**
- También incluye las **transiciones entre estados**
- Muestra los **puntos inicial y final** de una **secuencia de cambios de estado**
- Los diagramas vistos hasta ahora modelan el comportamiento del sistema o de un grupo de clases u objetos.
- El diagrama de estados muestra las condiciones de **un único objeto**

# Diagramas de estado

- Representan autómatas de estados finitos, desde el punto de vista de los estados y las transiciones
- Son útiles sólo para los objetos con un comportamiento significativo
- El resto de objetos se puede considerar que tienen un único estado

# Diagramas de estado

- Cada objeto está en un estado en cierto instante
- El estado está caracterizado parcialmente por los valores de los atributos del objeto
- El estado en el que se encuentra un objeto determina su comportamiento
- Cada objeto sigue el comportamiento descrito en el Diagrama de Estados asociado a su clase

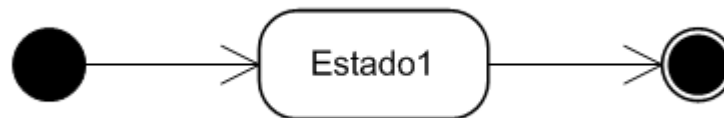
# Relación con otros diagramas

- **Diagramas de interacción**
  - Modelan el comportamiento de una **sociedad de objetos**, mientras que la máquina de estados modela el comportamiento de un **objeto individual**
- **Diagramas de actividades**
  - Se centran en el **flujo de control entre actividades**, no en el **flujo de control entre estados**.
    - El evento para pasar de una actividad a otra es la finalización de la anterior actividad



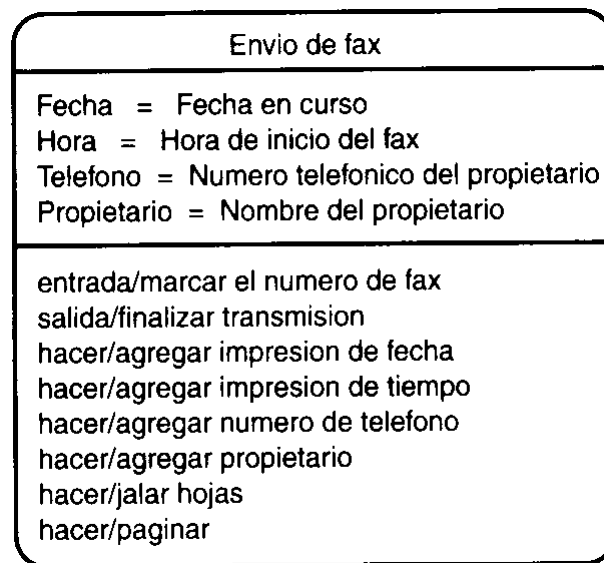
# Representación gráfica

- Un estado se representa como un rectángulo con bordes redondeados
- Las flechas indican una transición de estado
- La punta de flecha apunta hacia el estado donde se hará la transición
- El estado inicial se representa como un círculo relleno
- El estado final se representa como un círculo relleno con borde



# Representación gráfica

- En el diagrama de clases una clase tenía tres áreas: nombre de la clase, atributos y métodos.
- En el diagrama de estados **un estado** también puede tener **tres áreas**: nombre del estado, variables de estado y actividades



# Representación gráfica:

## Actividades internas

- Se puede especificar el hacer una acción como consecuencia de entrar, salir o estar en un estado:

estado A

---

entry: acción por entrar

exit: acción por salir

do: acción mientras en estado

# Representación gráfica:

## Actividades internas

- Se puede especificar el hacer una acción cuando ocurre en dicho estado un evento que no conlleva salir del estado:

estado A

---

on evento\_activador( arg1 )[ condición ]: acción por evento

# Representación gráfica

- Las variables de estado como cronómetros o contadores son, en ocasiones, de ayuda
- Las actividades constan de sucesos y acciones
- Tres de las más utilizadas son:
  - Entrada: qué sucede cuando el sistema entra al estado
  - Salida: qué sucede cuando el sistema sale del estado
  - Hacer: qué sucede cuando el sistema está en el estado

# Elementos: Estados

- Un **estado** es una **situación en la vida de un objeto** caracterizada por satisfacer una condición: esperar un evento (estática) o realizar una actividad (dinámica)
- Cada estado tiene un nombre
- El estado de un objeto está relacionado con los valores de sus atributos, los enlaces con otros objetos y las actividades que esté realizando

# Elementos: Eventos

- Un **evento** representa la ocurrencia de un suceso, dentro o fuera del objeto, que **provoca un cambio de estado** en el objeto (dispara una transición)
  - La sucesión de transiciones marca el “camino” seguido por el objeto entre los estados

# Elementos: Transiciones

- Una transición es una relación entre dos estados: indica que cuando ocurre un evento, el objeto pasa de un estado a otro
- Un **estado** tiene **duración**, una **transición** es **inmediata**
- Una transición puede tener varios eventos vinculados

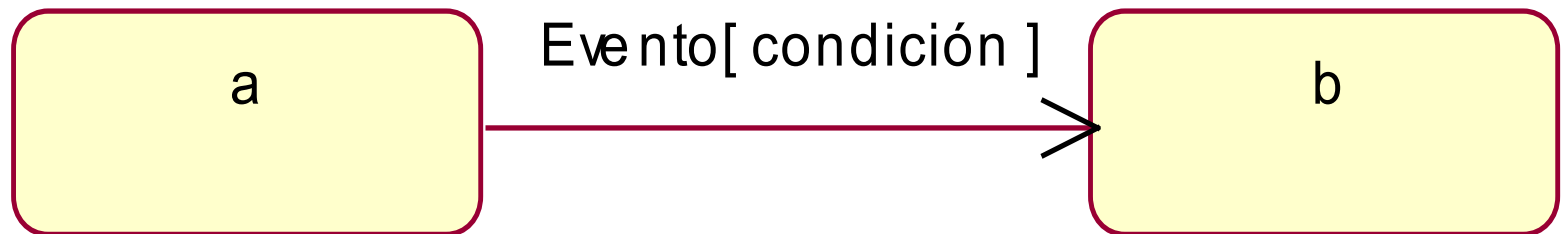


# Elementos: Transiciones

- Una transición tiene tres partes opcionales:
  - **Evento**: suceso en el tiempo
  - **Condición o guarda**: autoriza la transición si se cumple la condición
  - **Acción**: operación atómica que se ejecuta antes de que la transición alcance el nuevo estado
- Se representan mediante la siguiente notación:
  - Evento[condición]/acción

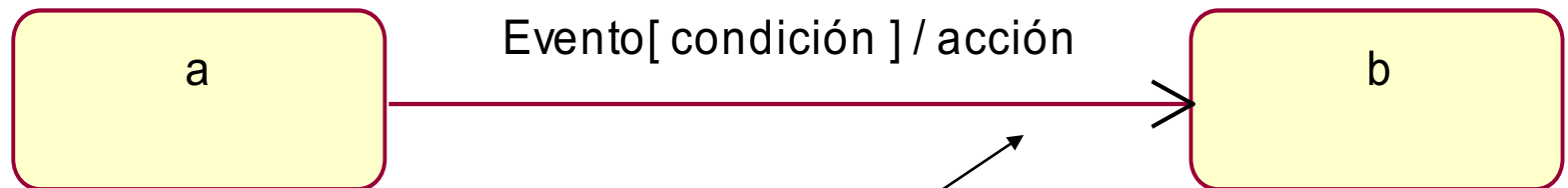
# Elementos: Transiciones

- La transición puede depender de que se cumplan ciertas condiciones:



# Elementos: Transiciones

- Podemos especificar la ejecución de una acción como consecuencia de la transición:



Dicha acción también  
se considera  
instantánea

# Elementos: Transiciones

- Podemos especificar el envío de un evento a otro objeto como consecuencia de la transición:

a

**Evento( arg1, arg2 )[ condición ] / ^otro\_objeto.evento(arg2)**

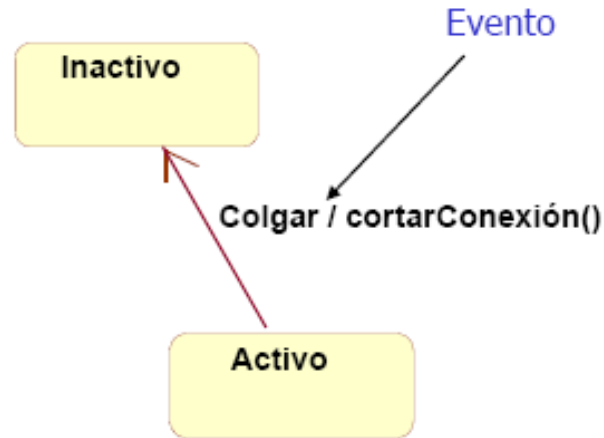
∨

b

# Tipos de eventos

- En cuanto a dónde acontecen, pueden ser:
  - **Externos**, si fluyen entre el sistema y sus actores (pulsación del ratón)
  - **Internos**, si fluyen entre objetos del sistema (una excepción)
- En UML se pueden modelar cuatro clases de eventos:
  - **De Señal**: Recepción de una comunicación asíncrona, explícita y con nombre, entre objetos.
  - **De Llamada**: Recepción, por un objeto, de una petición explícita síncrona.
  - **De Tiempo**: Llegada de un tiempo absoluto o transcurso de una cantidad relativa de tiempo.
  - **De Cambio**: Un cambio en el valor de una expresión booleana.

# Tipos de eventos



- Declaración de un evento Colgar de tipo Señal
- El evento Colgar produce un cambio desde el estado Activo al estado Inactivo
- Además se lleva a cabo la acción cortarConexión

# Ejemplo:

## estados, transiciones y eventos

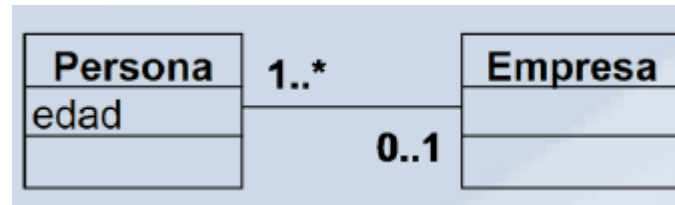


Diagrama de clases

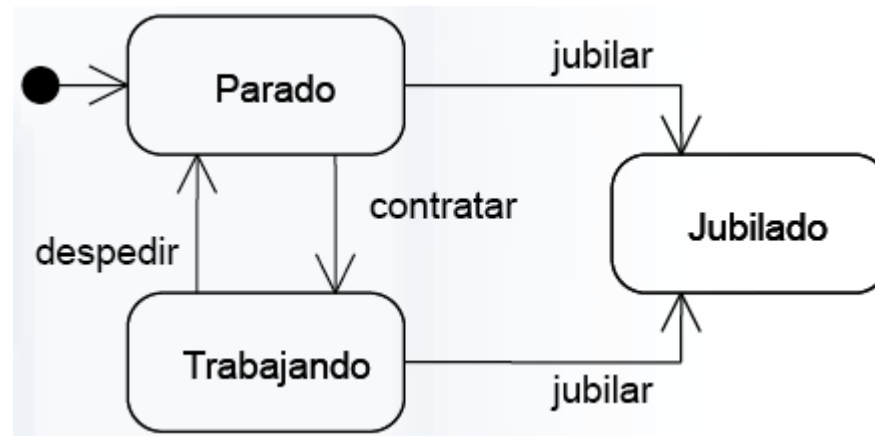


Diagrama de estados

# Elementos

- **Evento**

- Acontecimiento significativo que ocupa un lugar en el tiempo y en el espacio (un estímulo que puede disparar una transición entre estados):
  - Levantar el auricular telefónico

- **Estado**

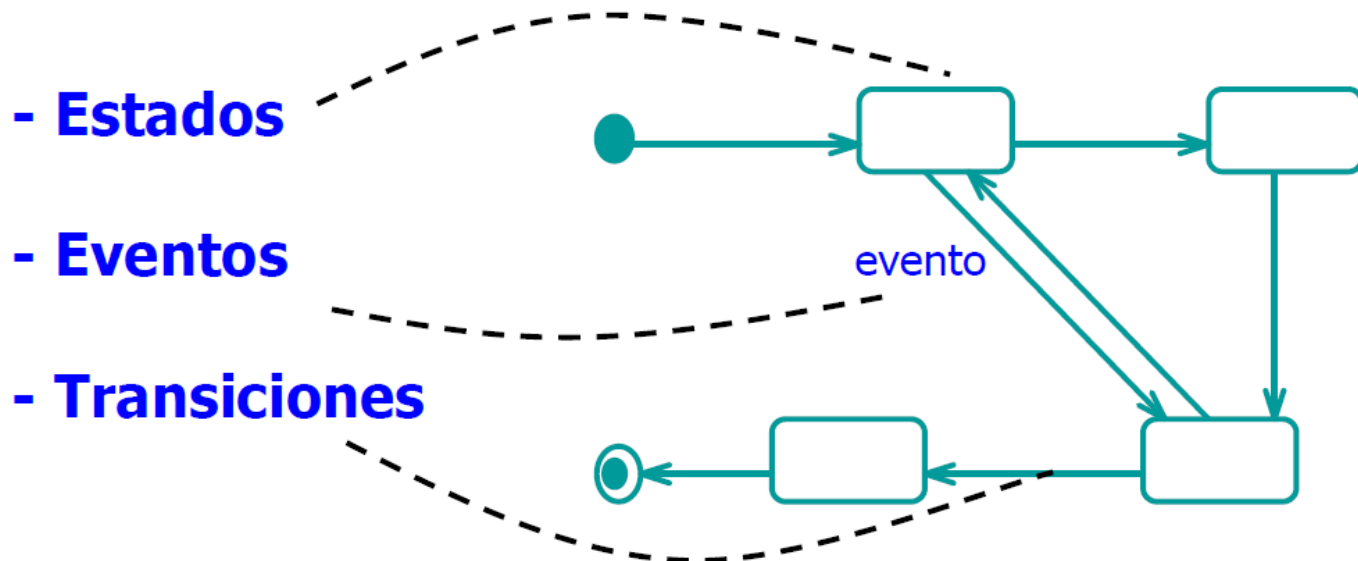
- Condición o situación de un objeto en un momento determinado donde:
  - Se satisface alguna condición, se realiza alguna actividad o se espera algún evento
  - Un teléfono está en estado “ocioso” una vez que se pone en su sitio el auricular y mientras no lo levantemos



# Elementos

- **Transición**

- Relación entre dos estados: indica que cuando ocurre un evento, el objeto pasa de un estado a otro (Evento/[Condiciones]/Acción)
  - Cuando ocurre el evento “levantar auricular” el teléfono realiza la transición del estado “ocioso” al estado “activo”



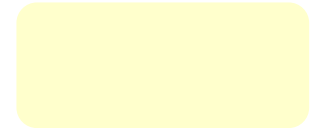
# Tipos de eventos

- Un evento es la especificación de un acontecimiento significativo, ubicado en el tiempo y en el espacio
- Se utilizan en máquinas de estado para modelar la aparición de un estímulo que puede disparar la transición de un estado a otro
- Pueden ser:
  - **Síncronos**
    - Llamadas (invocación de operaciones)
  - **Asíncronos**
    - Señales, Paso de tiempo, Cambio de estado

# Tipos de estados

- **Simple**

- Sin estructura interna

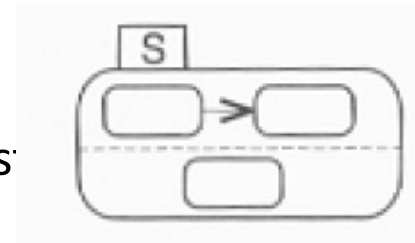


- **Compuesto**

- Tiene estructura interna con varios estados interiores (subestados)

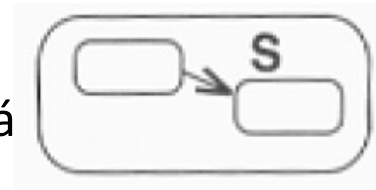
- Estado ortogonal (subestados concurrentes)

- Se divide en dos o más regiones
    - Cuando el estado está activo significa que lo es uno de los subestados de cada región



- Estado no ortogonal (subestados secuenciales)

- Contiene uno o más subestados directos
    - Cuando el estado está activo significa que lo está uno y solo uno de los subestados



# Tipos de estados

- **Estado inicial**

- Indica el punto de comienzo por defecto para la máquina de estados o para el subestado

- **Estado final**

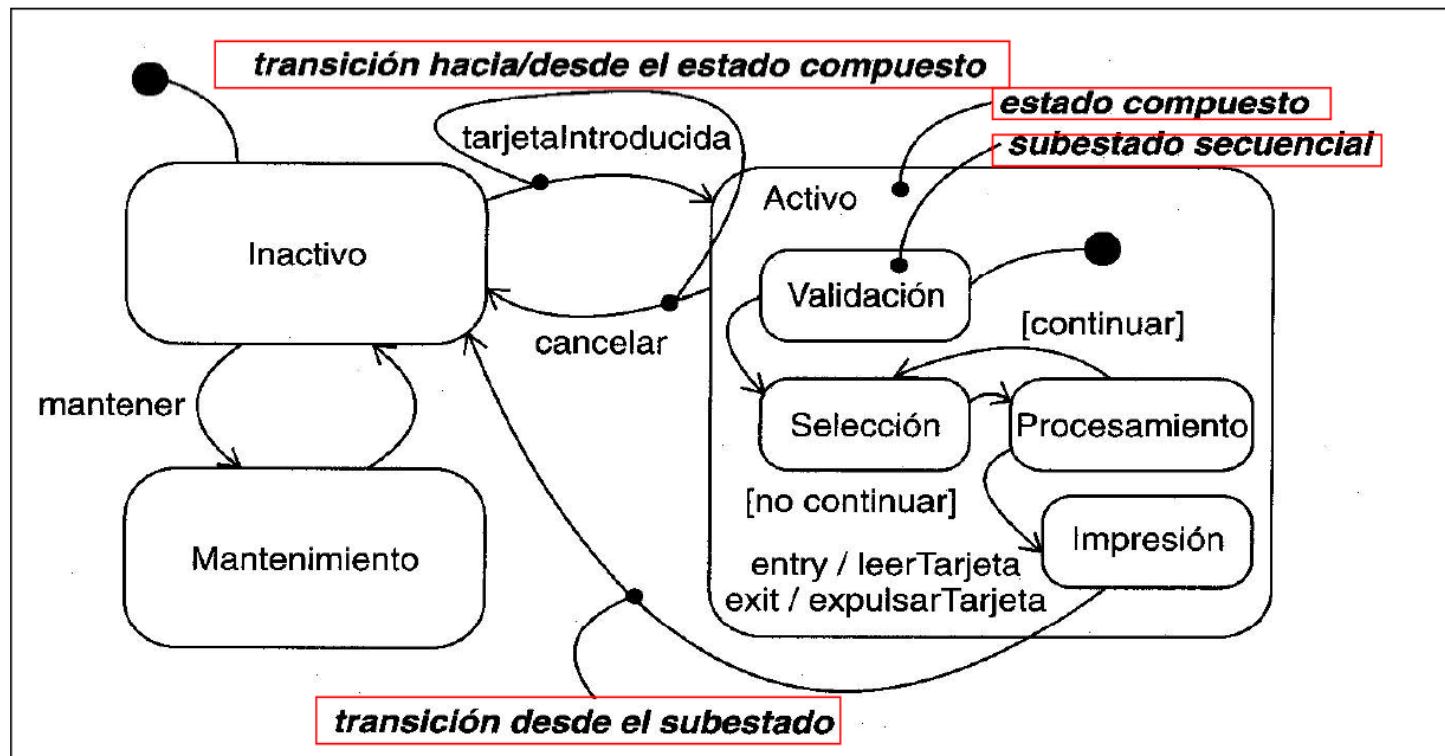
- Indica que la ejecución de la máquina de estados o estado que lo contiene, ha finalizado
- Si la máquina tiene uso infinito, puede no tener estado final (pero siempre tendrá estado inicial)

# Tipos de estados

- Un **subestado** es un estado **anidado** dentro de un **estado compuesto**
  - Los subestados dentro de un estado compuesto pueden ser **concurrentes** (estado compuesto ortogonal) o **secuenciales** (estado compuesto no ortogonal)
- Una transición desde fuera de un estado compuesto puede apuntar a:
  - El estado compuesto (la máquina de estados anidada debe incluir un estado inicial, al cual pasa el control al entrar al estado compuesto)
  - Un subestado anidado (el control pasa directamente a él)

# Tipos de estados

- Ejemplo subestados secuenciales:
  - Cajero automático



# Tipos de estados

- Subestados concurrentes
  - Las regiones ortogonales permiten especificar dos o más máquinas de estados anidadas que se **ejecutan en paralelo** en el contexto del objeto que las contiene
  - El estado compuesto acaba mediante una sincronización de las regiones ortogonales: las regiones que alcanzan sus estados finales quedan a la espera hasta que **todas las regiones acaban**, y entonces concluye el estado compuesto
  - Cada región ortogonal puede tener un estado inicial, un estado final y un estado de historia (estado histórico).

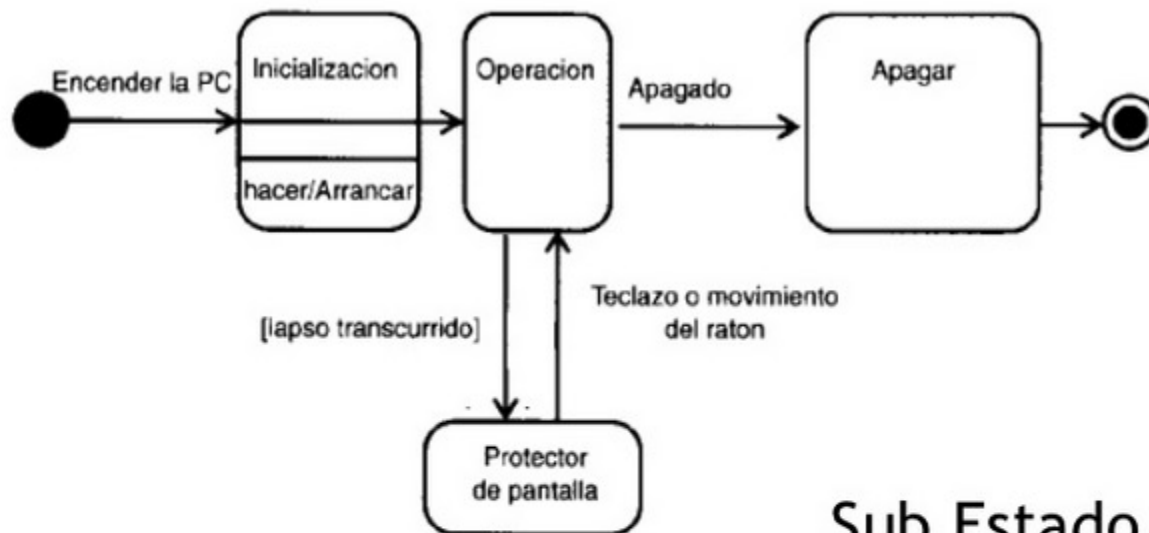
# Tipos de estados

- Un estado histórico indica que un estado compuesto recordará el último subestado activo cuando el objeto salga y vuelva al estado compuesto
- Ejemplo: Cuando se desactiva el protector de pantalla por el movimiento del ratón, la pantalla no vuelve a su estado inicial como si se reiniciara el PC, sino que se mostrará tal y como se dejó antes de que se activara el protector de pantalla



# Tipos de estados

- Ejemplo: Protector de pantalla

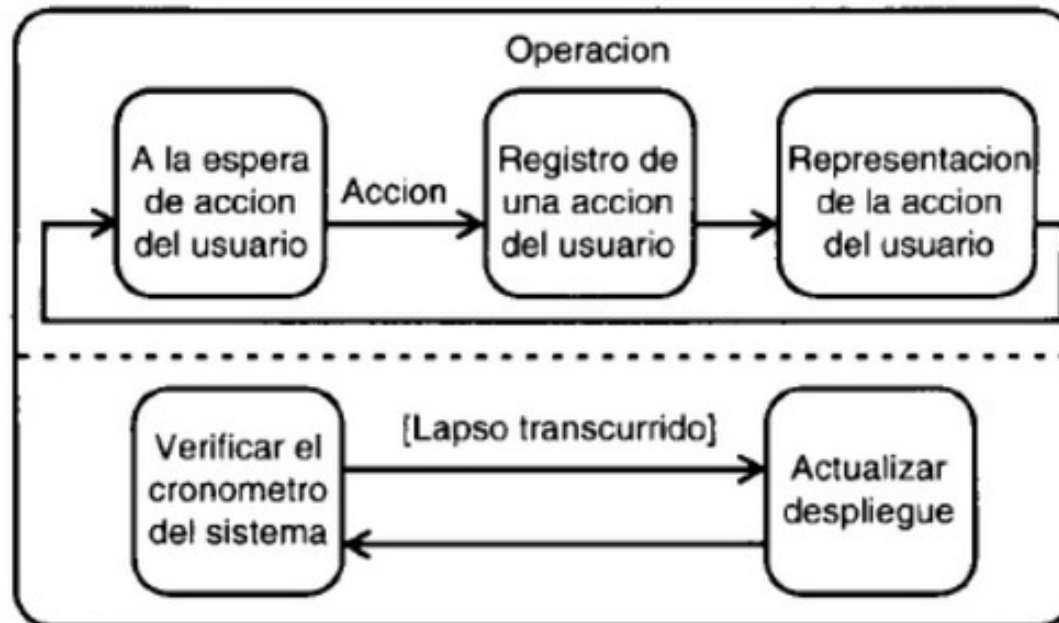


## Sub Estado del proceso Operación



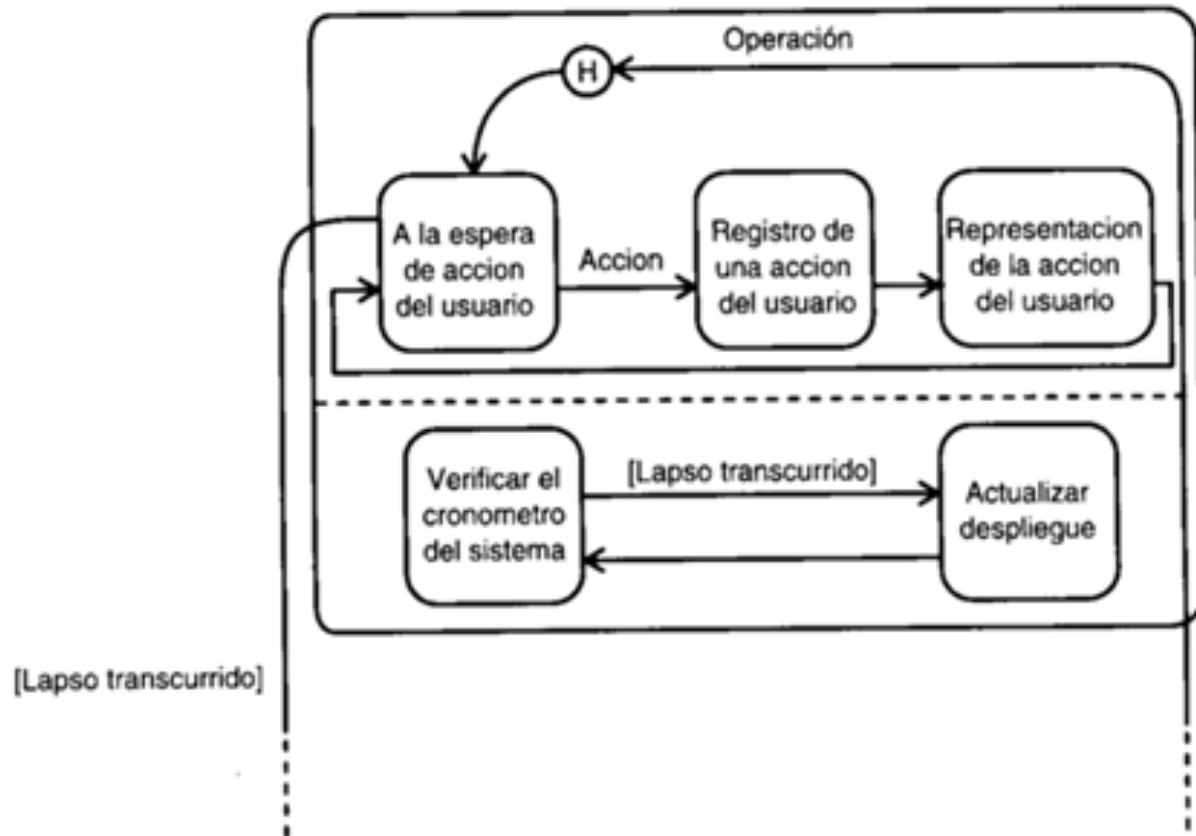
# Tipos de estados

- Subestado concurrente: Operación



# Tipos de estados

- Estado histórico dentro del estado concurrente Operación

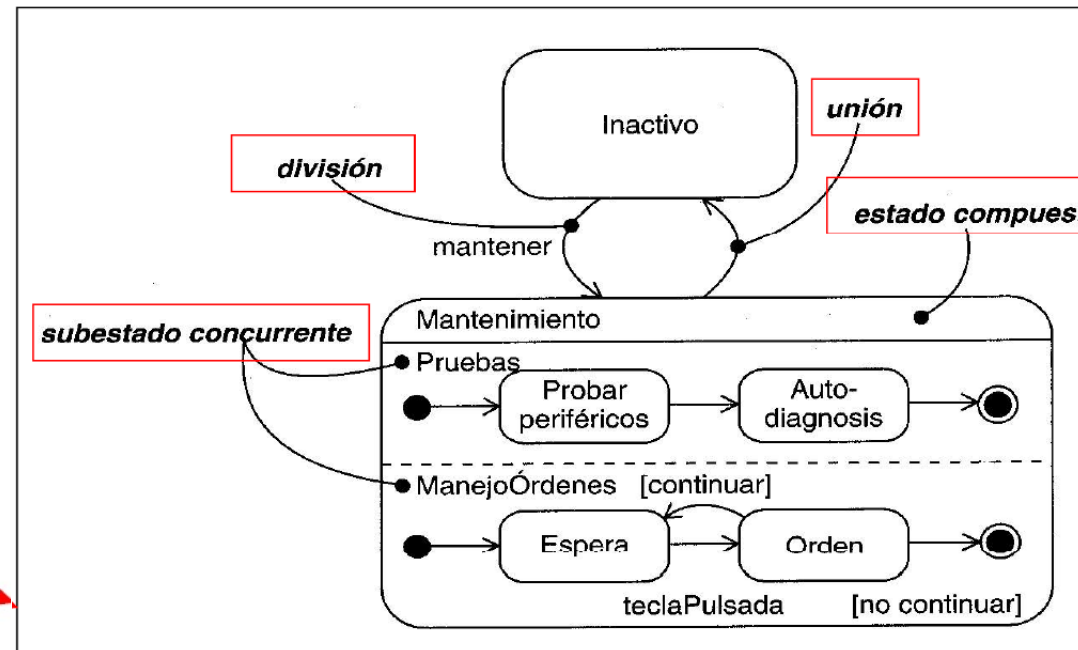
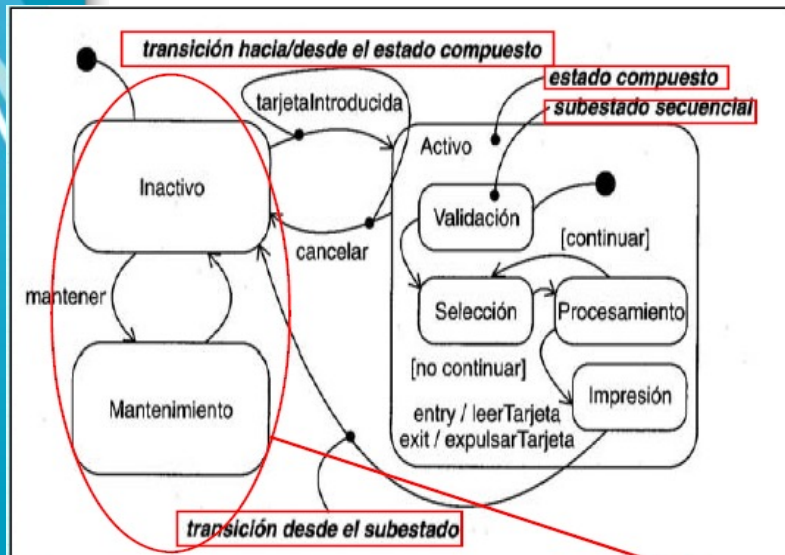


# Tipos de estados

- Transición de **División (Fork)**
  - El control pasa de un estado simple a varios estados ortogonales, cada uno de una región ortogonal diferente
  - Las regiones para las que no se especifica subestado destino toman como tal, por defecto, el estado inicial de la región
- Transición de **Unión (Join)**
  - Varias entradas, cada una de un subestado de una región ortogonal diferente, pasan el control a un único estado simple
  - Puede tener un evento disparador
  - La transición ocurre si todos los subestados origen están activos
  - El control sale de todas las regiones ortogonales, no solo de las que tienen subestado de entrada a la unión

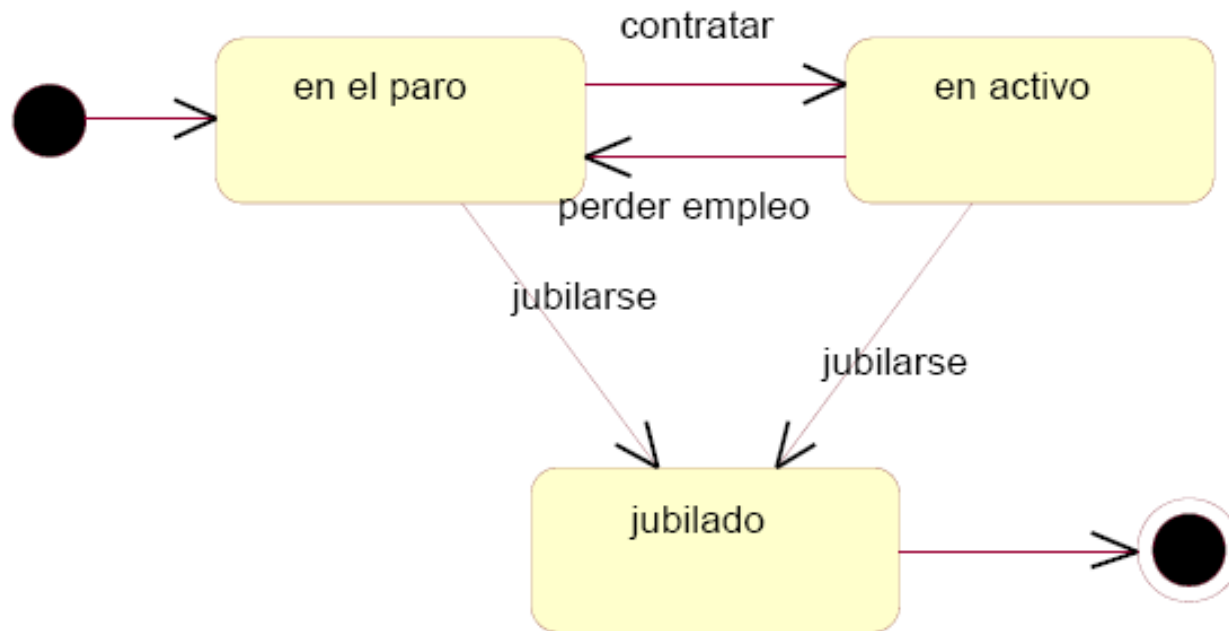
# Tipos de estados

- Ejemplo subestados concurrentes:
  - Cajero automático



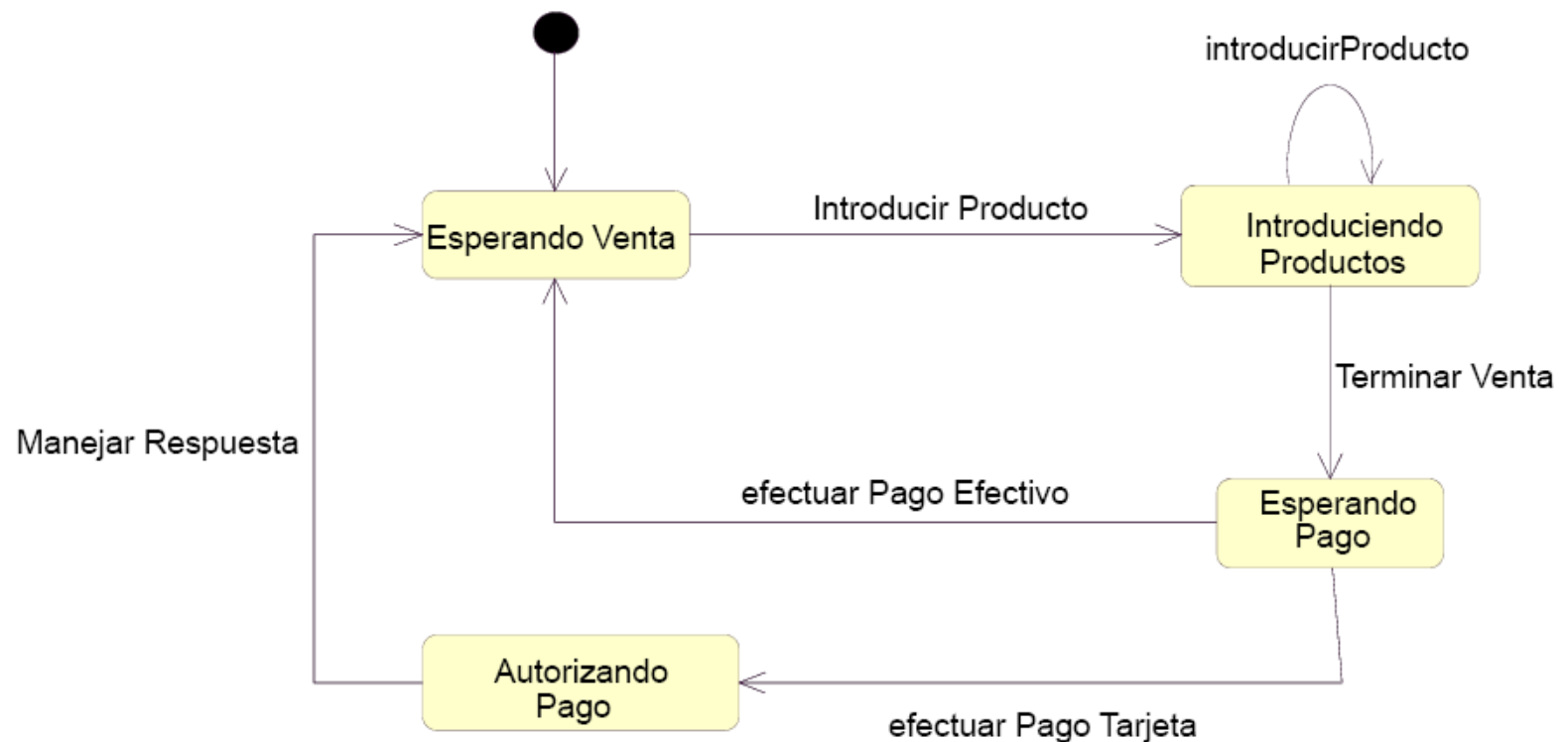
# Ejemplo: Diagrama de estado

- Ejemplo: para la clase persona



# Ejemplo: Diagrama de estados

- Ejemplo de Diagrama de Estados de un caso de uso



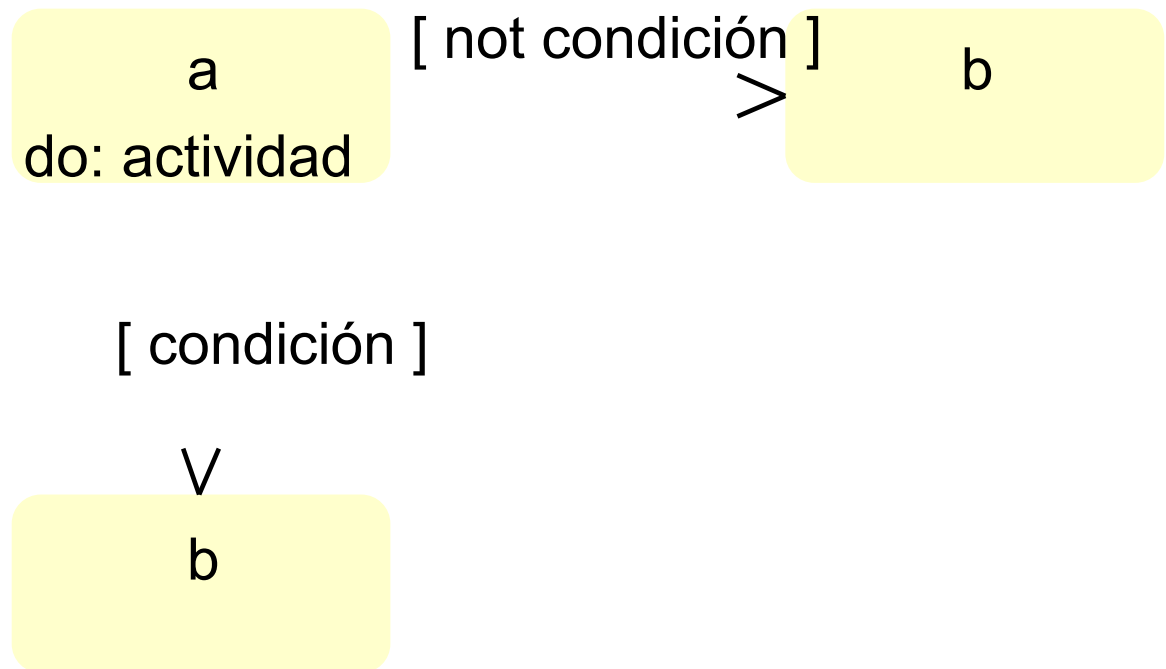
# Actividades

- Las actividades son similares a las acciones pero tienen duración y se ejecutan dentro de un estado del objeto.
- Las actividades pueden interrumpirse en todo momento, cuando se desencadena la operación de salida del estado



# Actividades

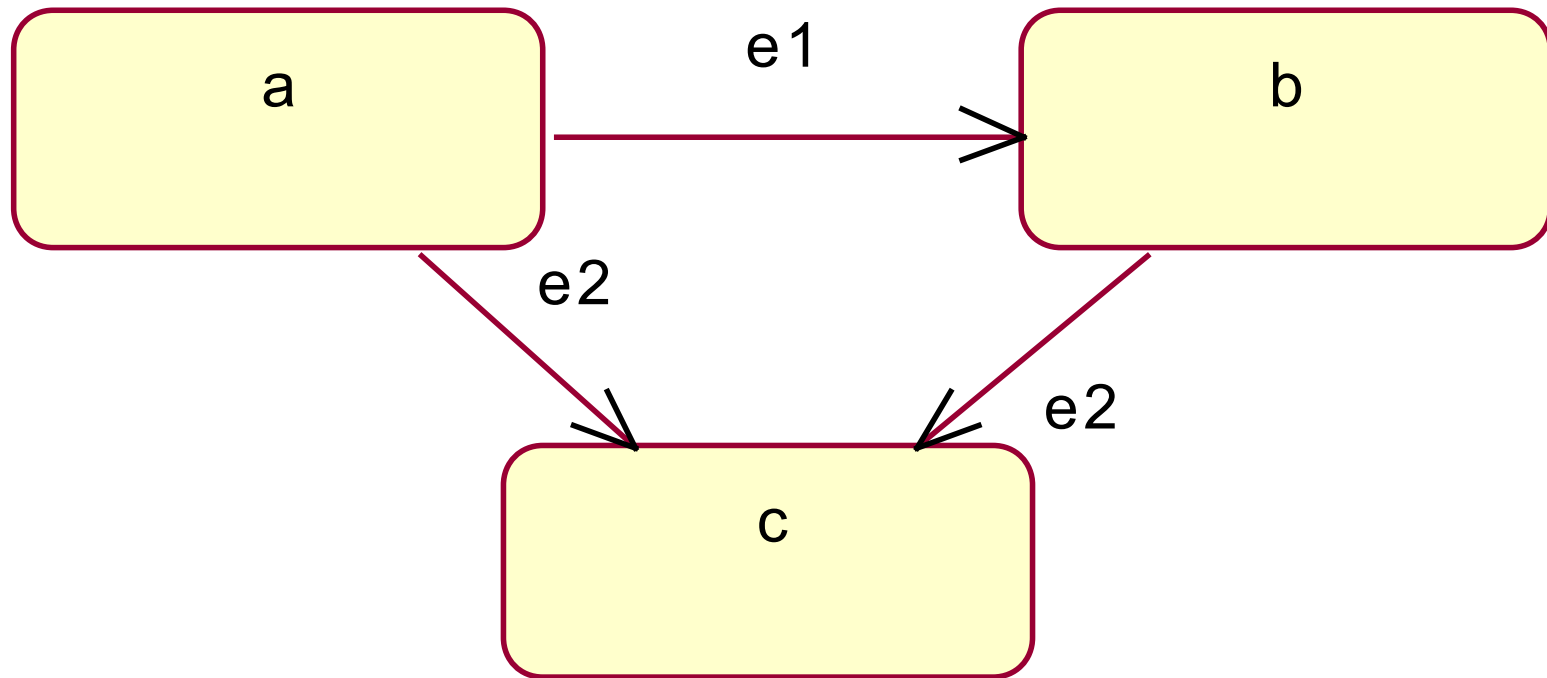
- Cuando una actividad finaliza se produce una transición automática de salida del estado



# Generalización de Estados

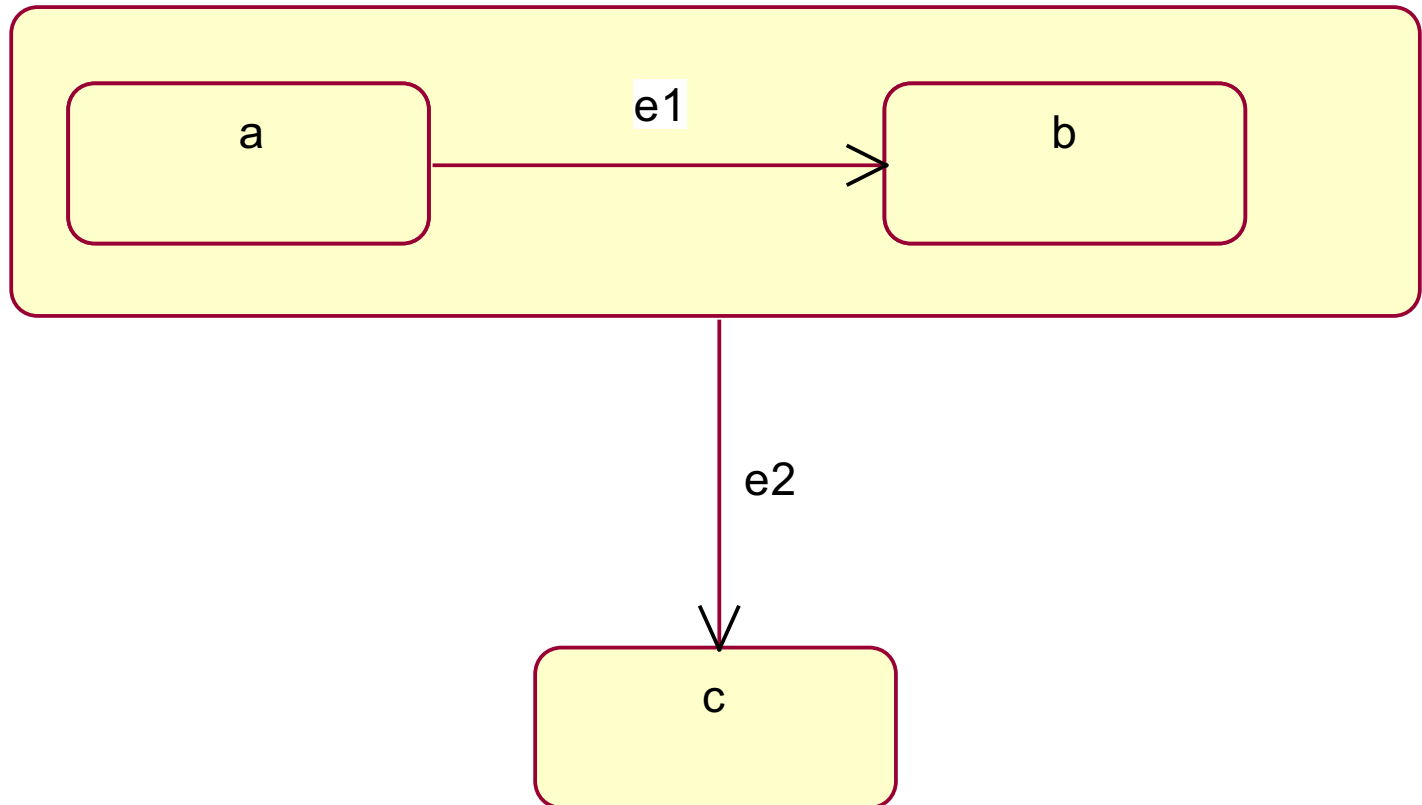
- Podemos reducir la complejidad de estos diagramas usando la generalización de estados.
- Distinguimos así entre **superestado** y **subestados**
- Un estado puede contener varios subestados disjuntos.
- Los subestados heredan las variables de estado y las transiciones externas.

# Generalización de Estados



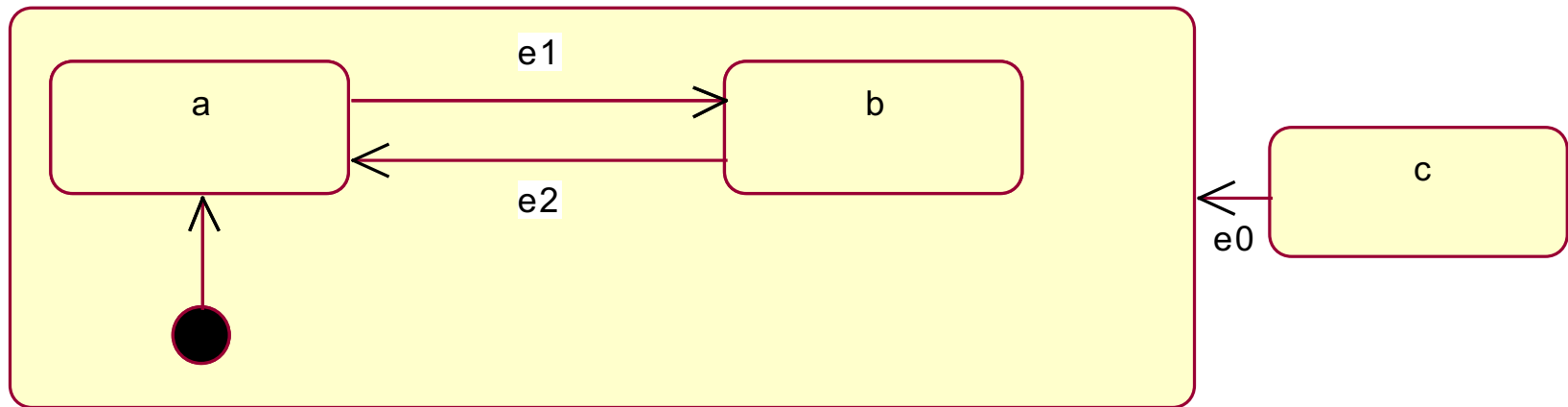
# Generalización de Estados

Quedaría como:



# Generalización de Estados

- Es preferible tener estados iniciales de entrada a un nivel de manera que desde los niveles superiores no se sepa a qué subestado se entra:

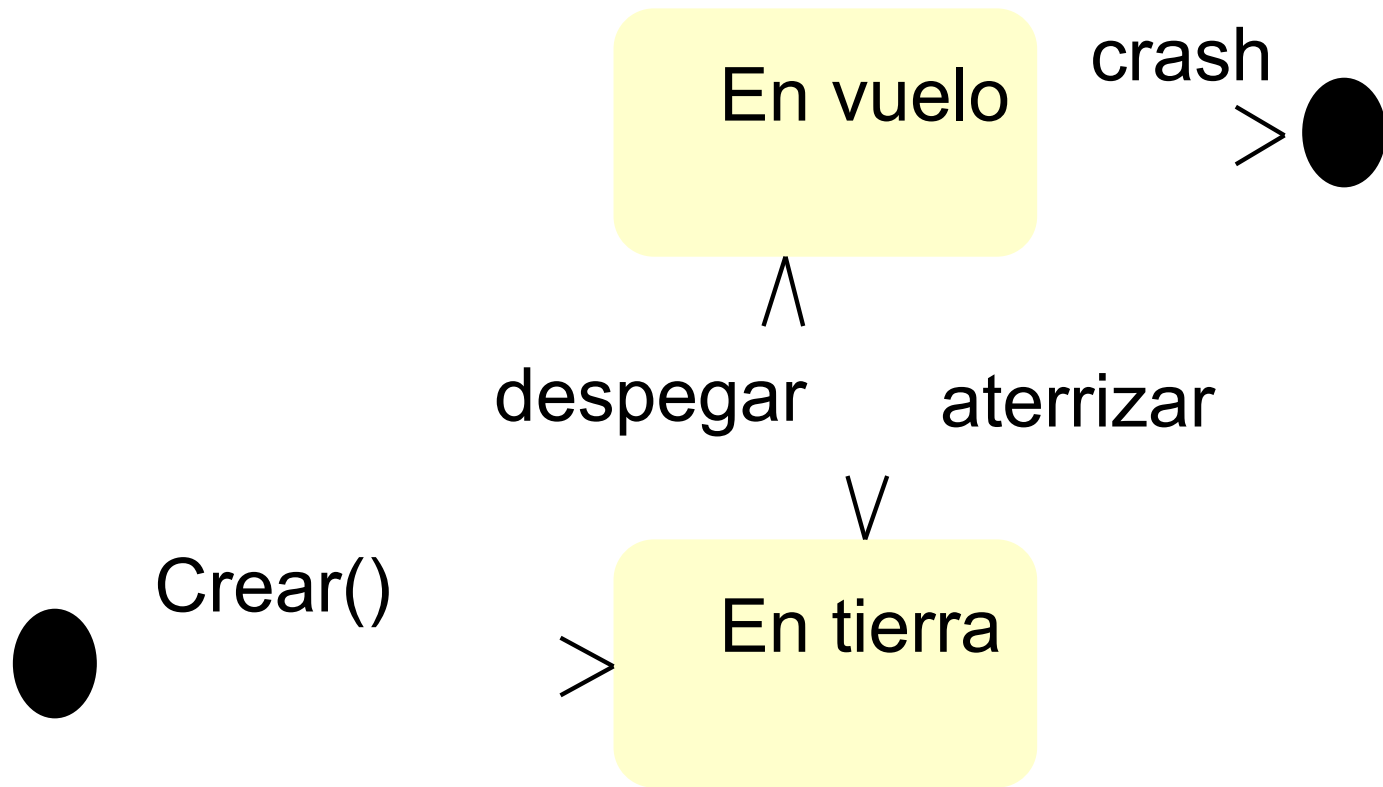


# Destrucción del Objeto

## Destrucción de objetos

- La destrucción de un objeto es efectiva cuando el flujo de control del autómata alcanza un estado final no anidado.
- La llegada a un estado final anidado implica la “subida” al superestado asociado, no el fin del objeto.

# Destrucción del Objeto

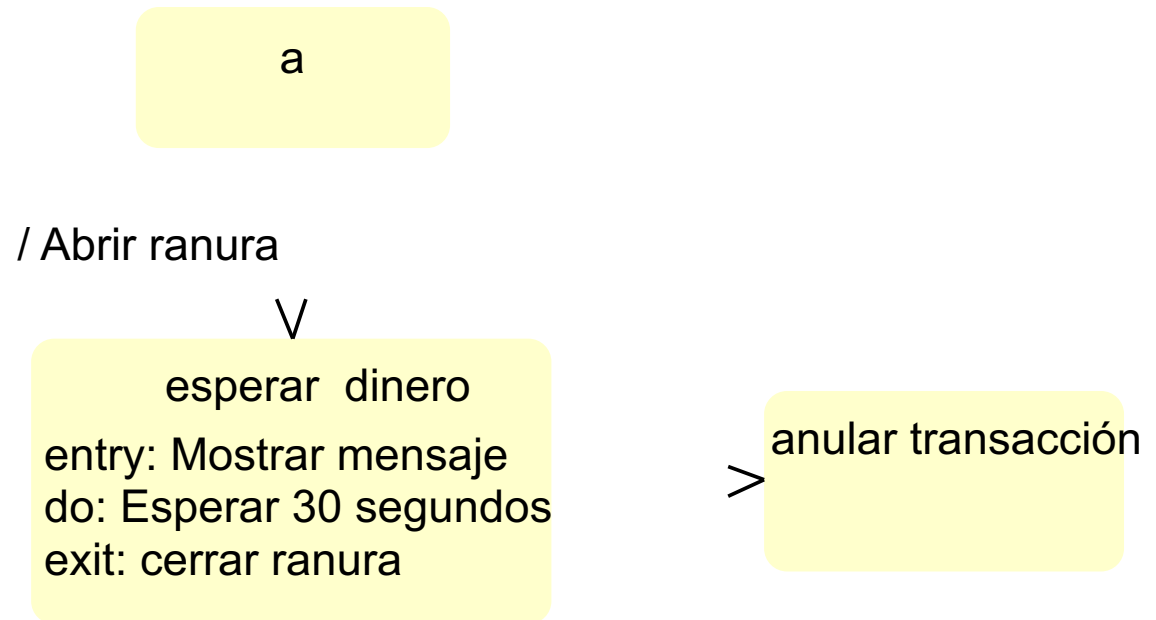


# Transiciones temporizadas

- Las esperas son actividades que tienen asociada cierta duración
- La actividad de espera se interrumpe cuando el evento esperado tiene lugar
- Este evento desencadena una transición que permite salir del estado que alberga la actividad de espera. El flujo de control se transmite entonces a otro estado



# Transiciones temporizadas



Si en 30 segundos no se introduce el dinero se termina la actividad pasando a anular la transacción. En cualquier caso se cierra la ranura.

# Transiciones temporizadas

a

/ Abrir ranura

∨

esperar dinero  
entry: Mostrar mensaje  
exit: cerrar ranura

Temporizador  
(30 segundos)

> anular transacción

Depósito efectuado

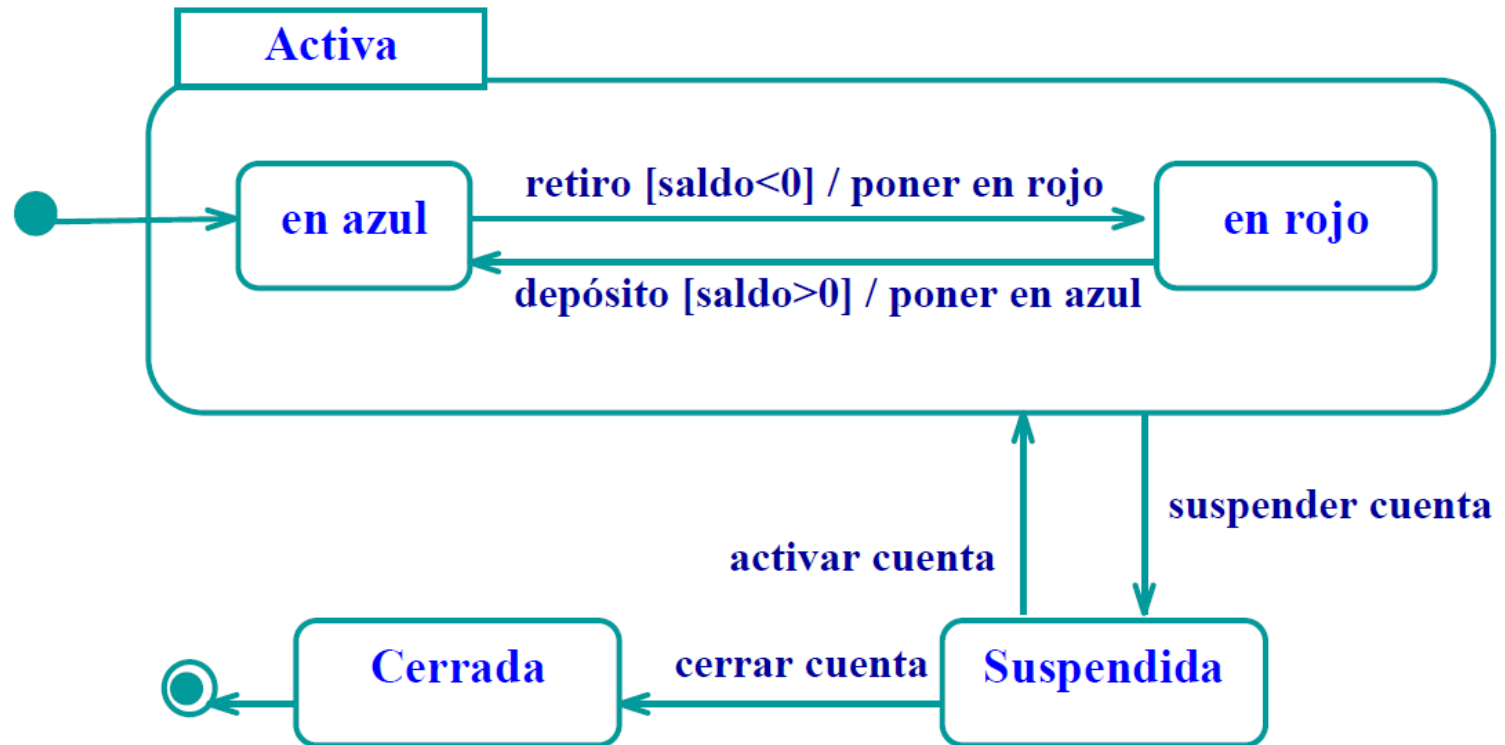
∨

b

# Ejercicio

- Una cuenta bancaria puede estar activa, suspendida o cerrada.
- Cuando está activa puede estar en azul (si el saldo es positivo) o en rojo (si el saldo es negativo)
- Realizar el diagrama de estados

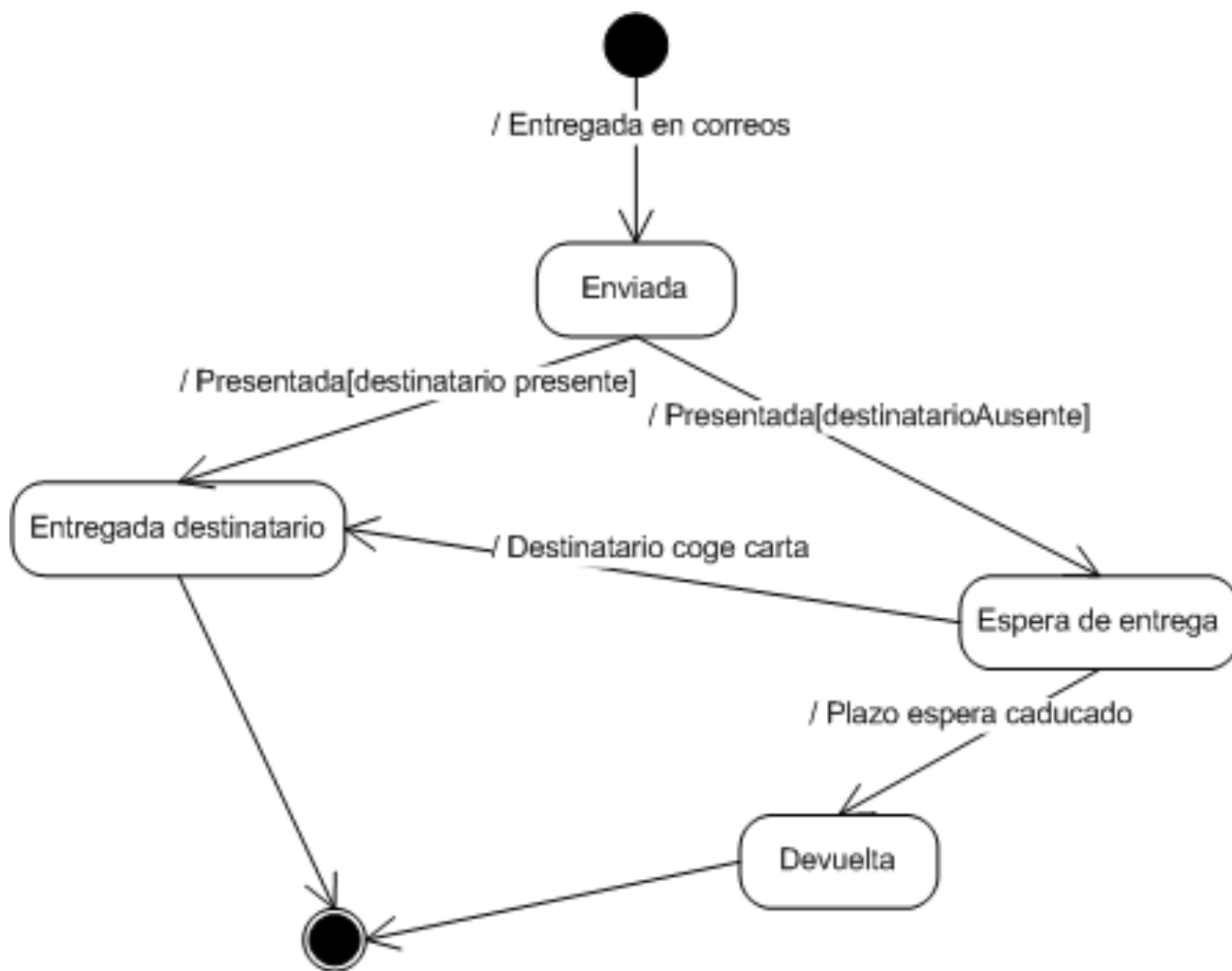
# Solución



# Ejercicio

- Una carta certificada pasa por varios estados: enviada, entregada al destinatario, en espera de entregar en la oficina de correos, reenviada al remitente
- Realiza el diagrama de estados desde que la carta es entregada en la oficina de correos hasta el momento en que se le entrega al destinatario
- Existen dos casos:
  - Si el destinatario está presente se le entrega directamente la carta
  - Si el destinatario está ausente cuando llega la carta, se conserva la carta en la oficina de correos a la espera de una posible entrega. Si el destinatario no la recoge antes del final del plazo máximo se le devuelve al remitente

# Solución



# Ejercicio

- Modelar el comportamiento de una cadena de música. Esta puede estar encendida (ON) o apagada (Standby). La cadena tiene reproductor de CD, Radio y Cinta. Se cambia de uno a otro con el botón “mode”. Cuando se enciende la cadena se recuerda el último estado en el que estuvo.

# Solución

