



# UML 2

- Diagrama de Maquina de estado
  - Diagrama de Secuencia
  - Diagrama de Tiempo



# Objetivos

- Describir dos diagramas basado en UML 2
  - Diagrama de Maquina de estado
  - Diagrama de Secuencia
- Utilizar herramienta CASE para la elaboración de dichos diagramas.



# ...UML

## DIAGRAMA DE MAQUINA DE ESTADO

Modelo avanzado del comportamiento



# Evento

- En el mundo suceden muchas cosas, esperadas y no esperadas, incluso suceden al mismo tiempo
- Las “cosas que suceden” se llaman eventos
- En sistemas software son los “estimulos” que dispara una transicion de estado.
- Eventos síncronos: llamadas
- Eventos asíncronos: una señal, el paso del tiempo



# Terminología

- Evento
  - La especificación de un acontecimiento significativo que tiene ubicación en tiempo y espacio
- Estado
  - Una condición o situación durante la vida de un objeto, durante el cual se cumple una condición, realiza acciones o espera algún evento
- Transición
  - El movimiento de un estado a otro en respuesta a un evento
- Acción
  - Ejecución atómica interna de un objeto



# Inicio de un Evento

- Pulsar un botón en un cajero automático
- Un robot autónomo tropieza con un objeto
- Detección de sobrecarga en los buffer de mensajes en un dispositivo enrutador de red
- Tiempo fuera de lo establecido para una acción en una planta química




# Maquina de estado

- Modela el comportamiento de un objeto individual
- Especifica la secuencia de estados por los que atraviesa un objeto durante su vida en respuesta a eventos



# Para que sirven?

- Mostrar el comportamiento dinámicos de las instancias de:
  - Clases, Casos de uso, subsistemas y sistemas
- Modelar la historia del ciclo de vida de un solo objeto reactivo como un maquina de estado finita
- La maquina realiza transiciones entre los estados de un objeto debido a las acciones que realiza el objeto en respuesta a eventos

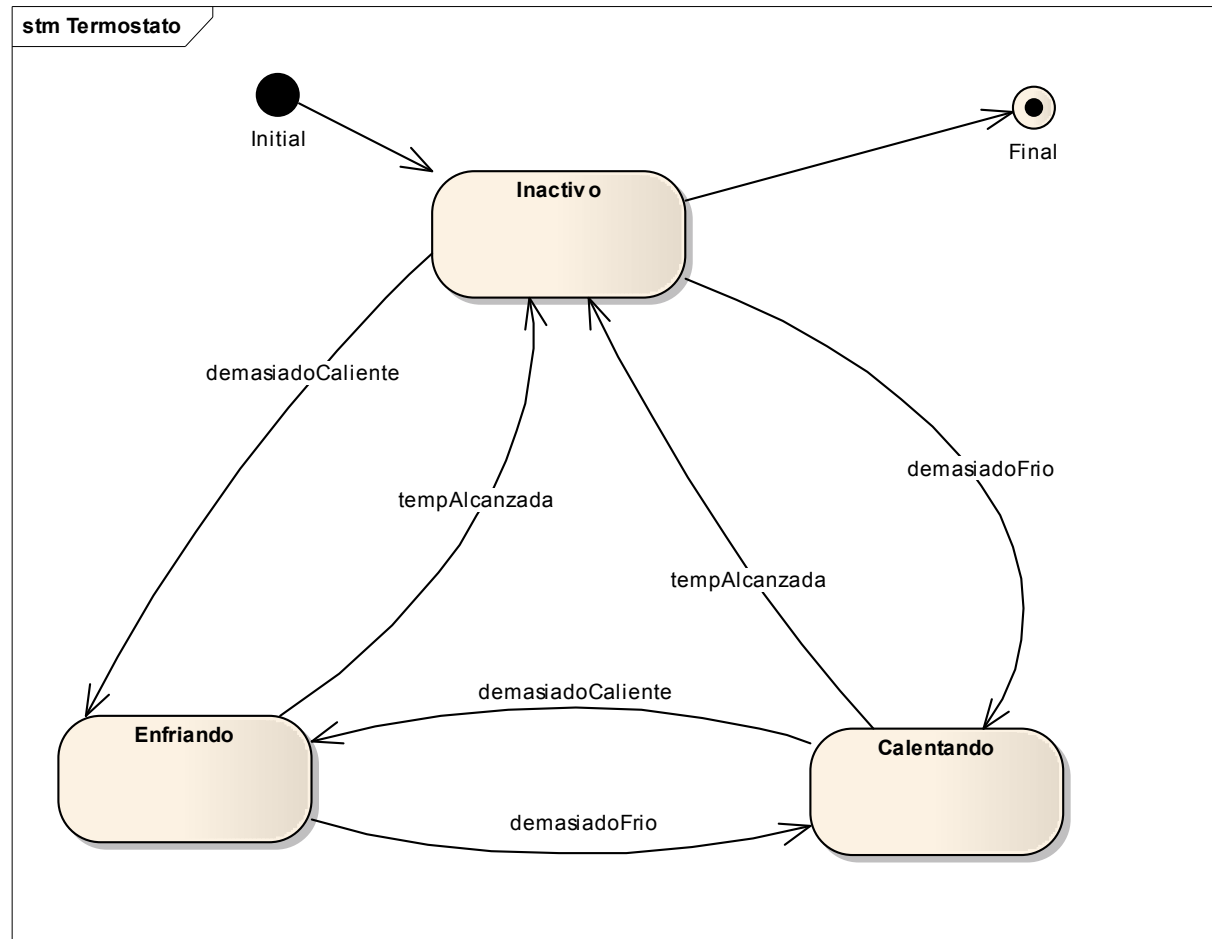


Así trabaja  
la maquina





# Ejemplo





# **DIAGRAMAS DE ESTADO**

---

**“Se usa para mostrar la historia de la vida de un objeto de una clase, los eventos que causan una transición de un estado a otro y las acciones que resultan de un cambio de estado”**

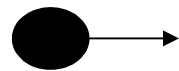


# DIAGRAMAS DE ESTADO

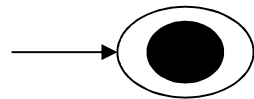
---

## Estado

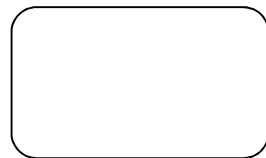
**“Es una de las posibles situaciones en la cual un objeto puede existir y representa una combinación de todas las propiedades de un objeto”**



**Estado inicial**



**Estado final**



**Estado**

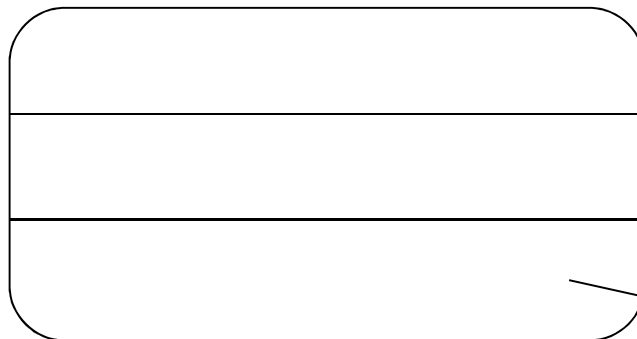


# DIAGRAMAS DE ESTADO

---

## Estado

**Cuando se está en un estado se pueden ejecutar actividades que requieren de un tiempo de ejecución mayor que el instantáneo que transcurre cuando se produce la transición hacia él. También pueden validarse restricciones, que de incumplirse provocan que el objeto salga de ese estado.**



## Estado

**Provoca una reacción interna, pero no cambia el estado del objeto**



# **DIAGRAMAS DE ESTADO**

---

## **Evento**

**“Un evento es un hecho que ocurre en algún momento y que es de importancia para la aplicación”**

**Internos**

**Externos**

**Temporales**

## **Condición guardiana**

**“Una condición guardián es una expresión booleana de los valores de los atributos. Permite transiciones sólo si la condición es true”**

**Se encierra entre [ ]**



# **DIAGRAMAS DE ESTADO**

---

## **Acción de una transición**

**“Una acción es una operación que se ejecuta instantáneamente y es no interrumpible”**

**Los nombres se muestran precedidas por un /**

## **Transición**

**“Es una relación entre dos estados que indica que cuando el evento ocurra pasa del estado anterior al siguiente.”**

**Es una flecha con orientación hacia el estado siguiente, con una etiqueta para el evento.**



# **DIAGRAMAS DE ESTADO**

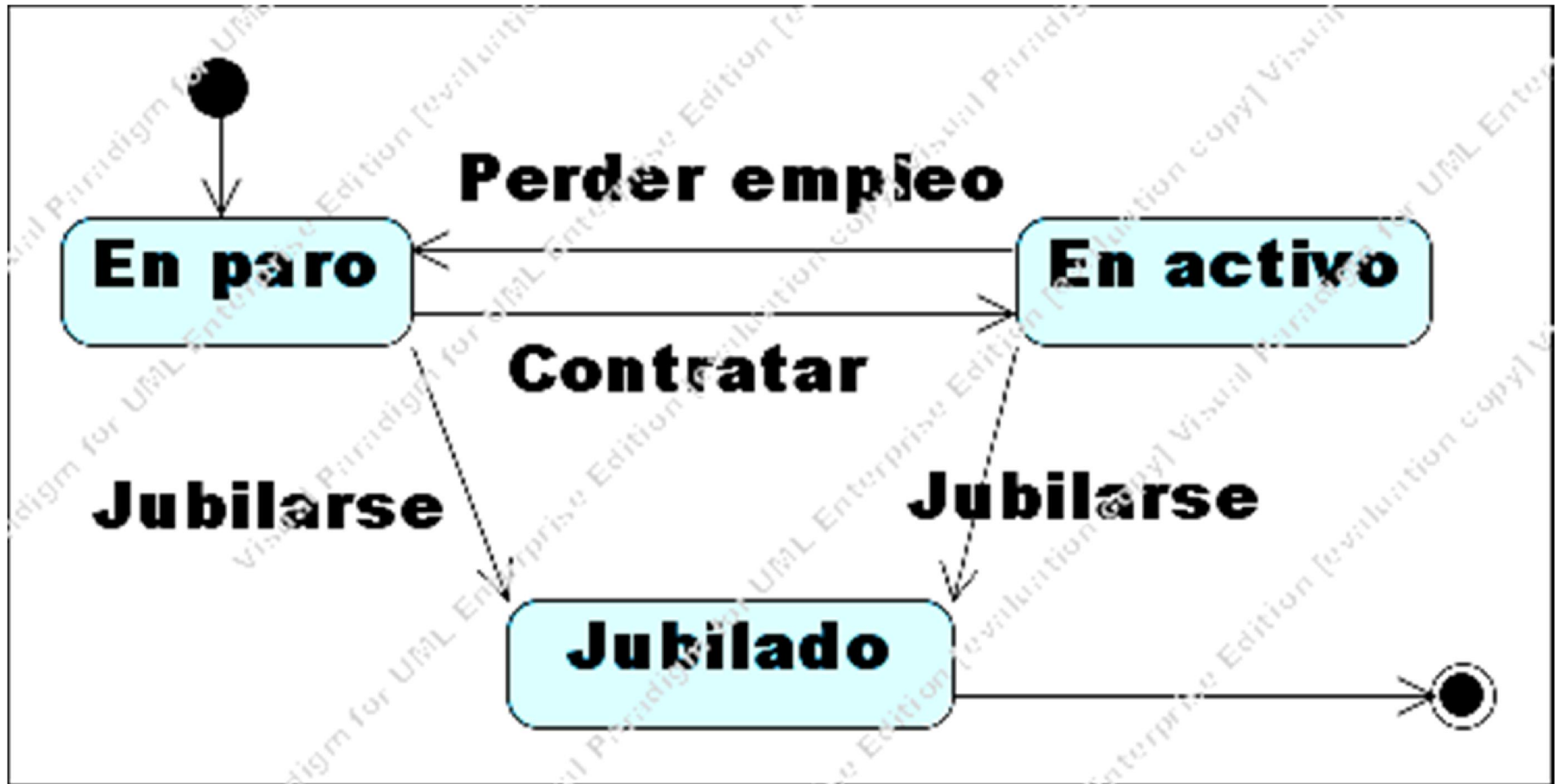
---

**Para especificar una transición se sigue el siguiente formato:**

**<Evento> [[Condición guardiana]][/Acción]**



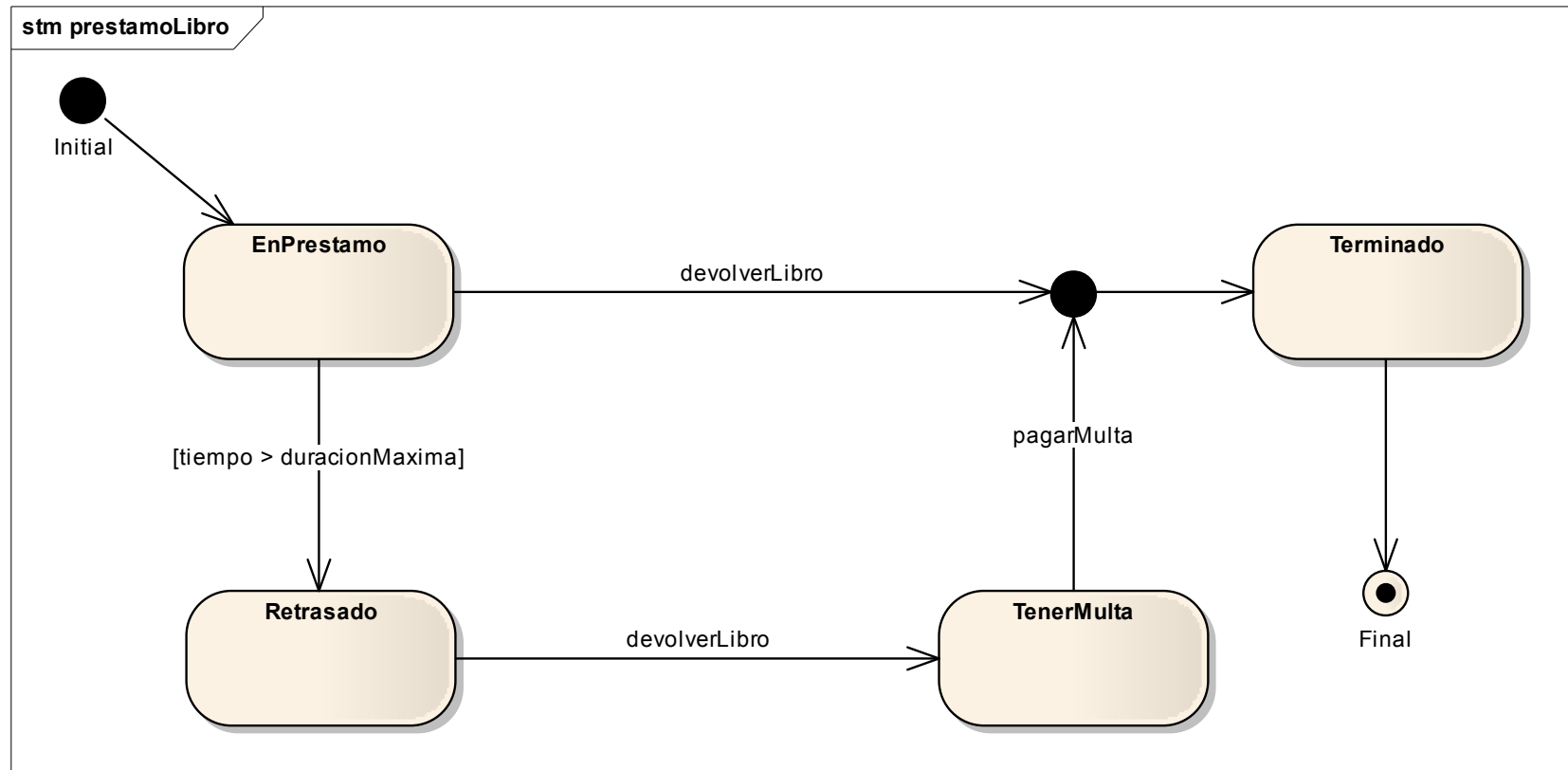
# Clase Trabajador





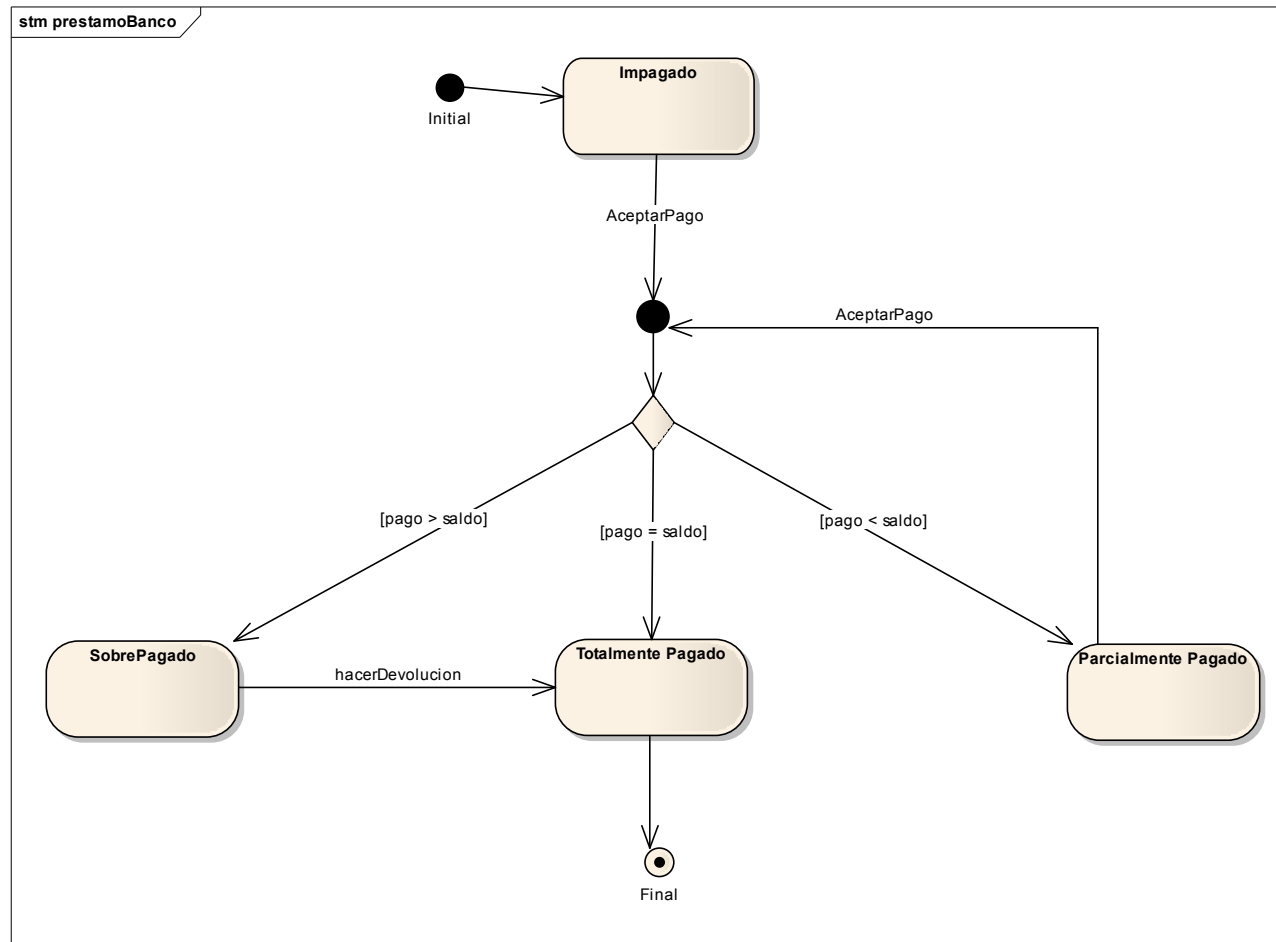


## ... otro ejemplo





# + ejemplo





# ...UML

## DIAGRAMA DE SECUENCIA



# Diagrama de secuencia

- Es una forma de diagrama de interacción
- Muestra los objetos como líneas de vida
- Muestra las interacciones ordenadas en el tiempo como mensajes desde la línea de vida origen hasta la línea de vida destino
- No están pensados para mostrar lógicas de procedimientos complejos



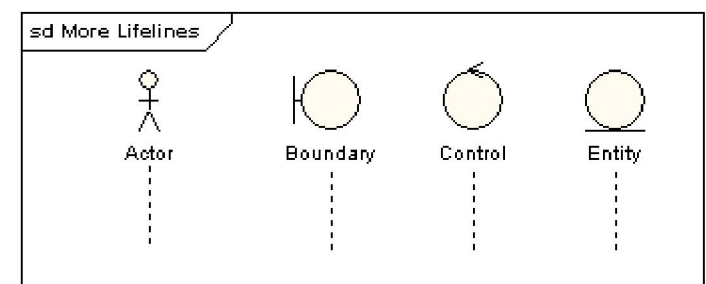
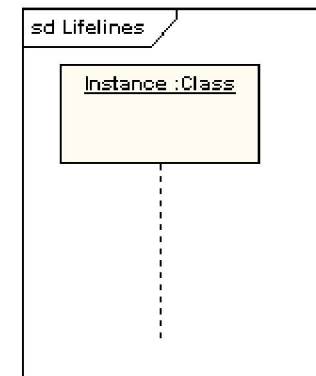
# Para que sirven?

- Mostrar qué objetos se comunican con qué otros objetos y qué mensajes disparan esas comunicaciones
- Para especificar casos de uso
- Para diseñar casos de uso
- Para diseñar operaciones de clases



# Línea de vida

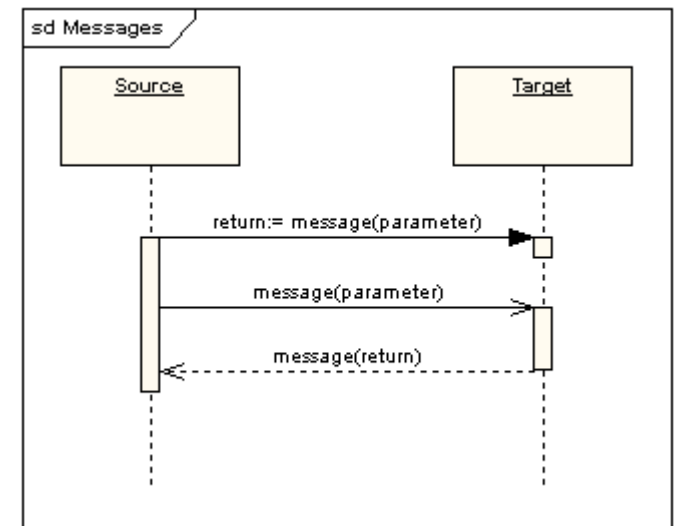
- Representa un participante individual y usualmente tiene un rectángulo que contiene el nombre del objeto
- Algunas veces tendrá una línea de vida con un símbolo del elemento actor o clases del análisis, si es contenido por un caso de uso





# Mensajes

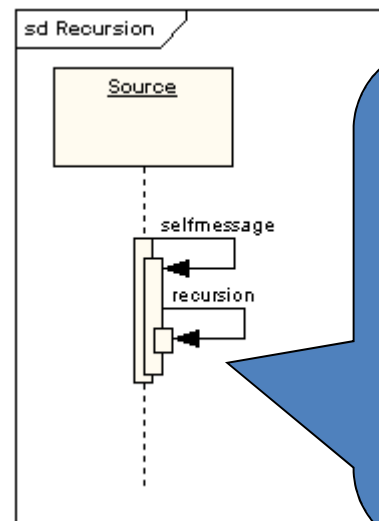
- Es el mecanismo de interacción entre objetos
- Se muestran como flechas
- Los mensajes pueden ser síncronos o asíncronos
- En el ejemplo:
  - El primer mensaje es un mensaje síncrono (denotado por una punta de flecha oscura), completo con un mensaje de retorno implícito
  - El segundo mensaje es asíncrono (denotado por una punta de flecha en línea)
  - El tercero es un mensaje de retorno asíncrono (denotado por una línea punteada)





# Foco de control

- Denota la ocurrencia de ejecución o activación de un foco de control de un objeto
- Es representado como un rectángulo fino a lo largo de la línea de vida



**Mensaje Self :**

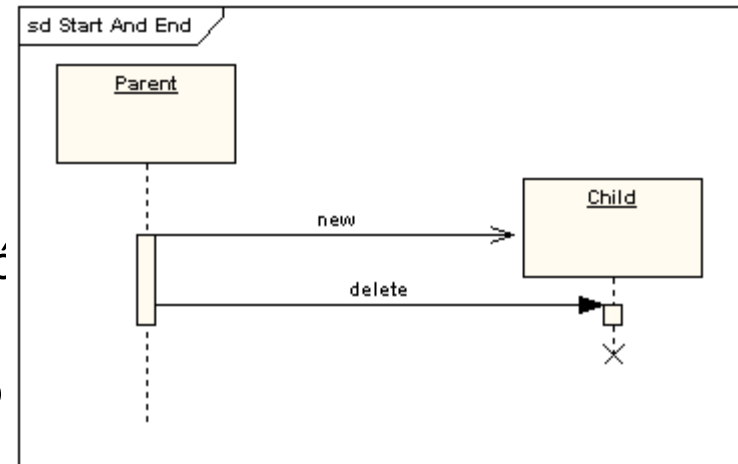
- Llamada recursiva de una operación
- Un método llamando a otro método perteneciente al mismo objeto





# Inicio y final de línea de vida

- Se puede crear o destruir objetos durante la escala de tiempo
- Representación:
  - El símbolo al inicio de la línea de vida se muestra en un nivel más bajo del objeto que causó la creación
  - La línea de vida se termina por un símbolo de detención, representado como una cruz



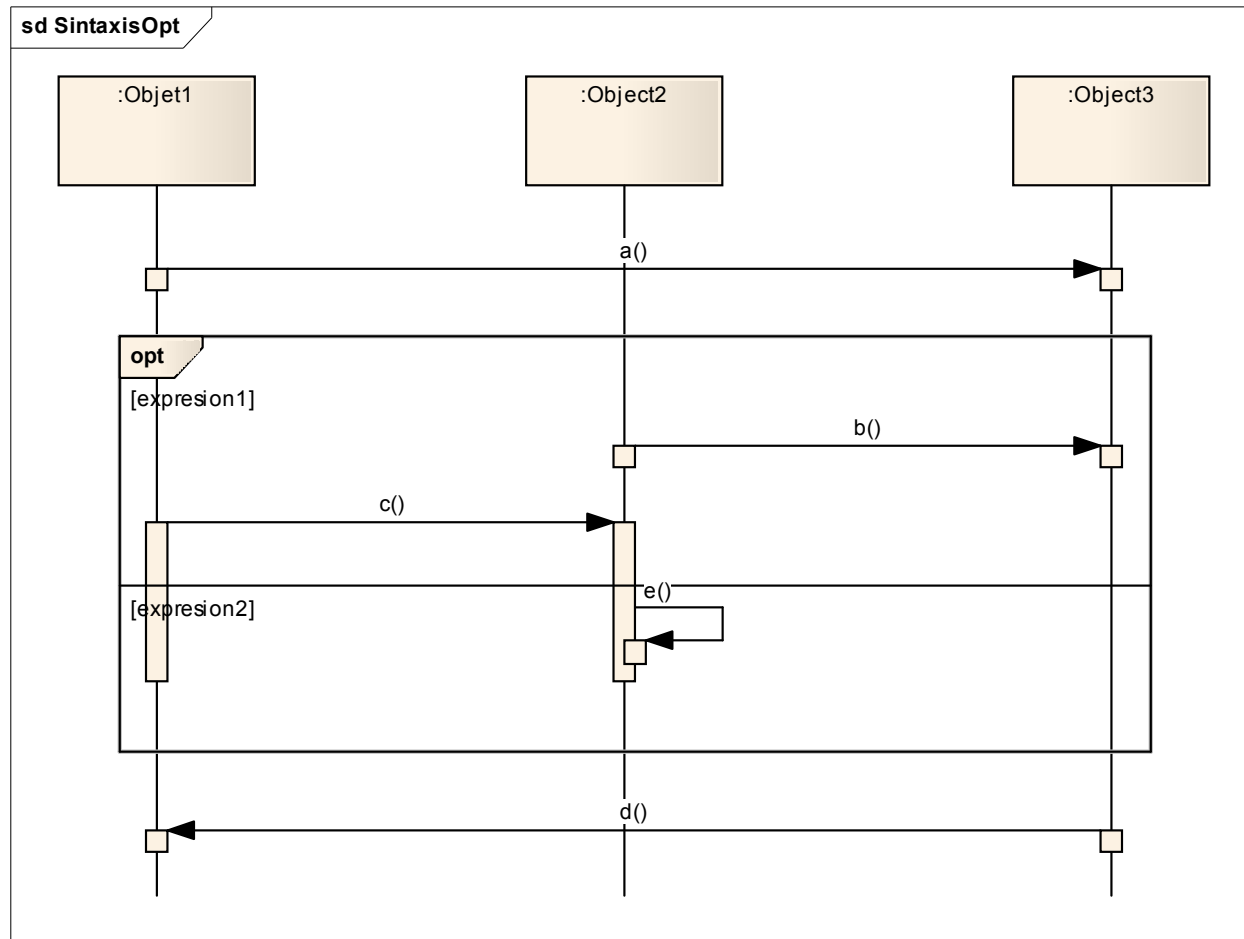


# Fragmentos combinados

- Mecanismos que permiten agregar lógica de procedimiento y se ejecutan bajo circunstancias definidas por los operadores
- Operadores:
  - Alternativas “alt” modela estructuras switch....
  - Opción “opt” modela estructuras if...then...
  - Paralelo “par” modela procesos concurrentes
  - Critico “critical” se ejecutan sin interrupcion
  - Bucle “loop” serie de mensajes iterativos

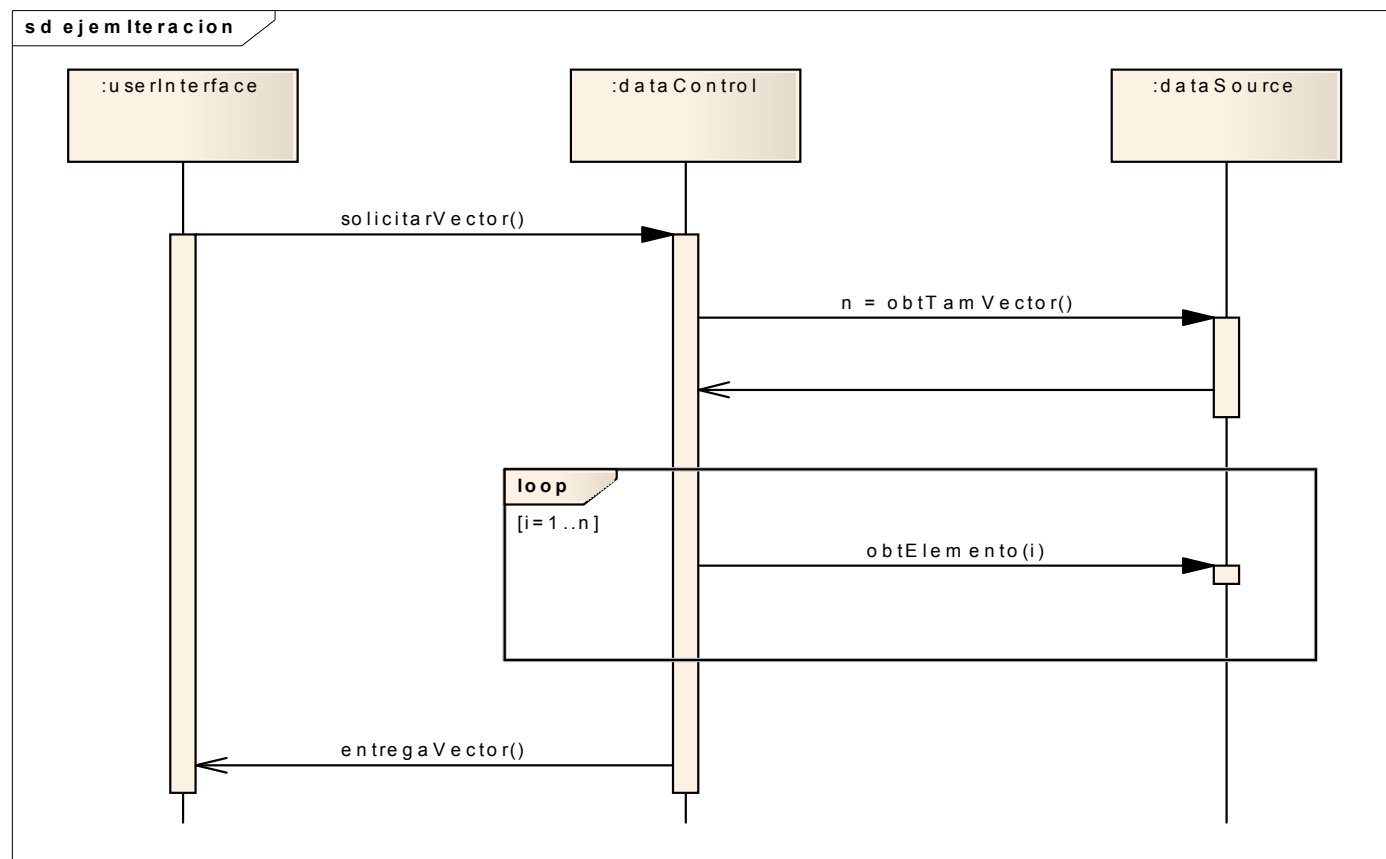


# Operador “Opt”





# Operador “loop”





# Objetivos

- Describir el diagrama de tiempo y diagrama de clases basado en UML 2
- Generar código a partir del diagrama de clases
- Utilizar herramienta CASE para la elaboración de dichos diagramas



# ...UML

## DIAGRAMA DE TIEMPO



# Diagrama de Tiempo

- Define el comportamiento de objetos con una escala de tiempo
- Provee una representación visual de los objetos cambiando de estado a lo largo del tiempo
- Muestra la interacción entre los eventos, las restricciones de tiempos y la duración
- Línea de vida del estado:
  - El eje X muestra el tiempo transcurrido en cualquier unidad que se elija
  - El eje Y se nombra con una lista de estados proporcionados



# Para que sirven?

- Para modelar sistemas en tiempo real
- Para definir componentes de software dirigidos por hardware o embebidos
  - Controladores de trafico
  - Un controlador de microondas
- También puede usar diagramas de tiempo para especificar procesos de negocio dirigidos por tiempo





# Ejemplo: Diag. de Tiempo

