

# Paradigmas de Programación

## UML y la POO

Moisés García Villanueva

23 de Octubre de 2007

# UML - Modelado

- El Lenguaje Unificado de Modelado, UML es una notación estándar para el modelado de sistemas software.
- UML no es un proceso de desarrollo, es decir, no describe los pasos sistemáticos a seguir para desarrollar software.
- UML sólo permite documentar y especificar los elementos creados mediante un lenguaje común describiendo modelos.

# ¿Que es modelado?

- En todos los ámbitos de la ingeniería se construyen modelos, que corresponden a simplificaciones de la realidad, para comprender mejor el sistema que vamos a desarrollar: los arquitectos utilizan y construyen planos (modelos) de los edificios, los grandes diseñadores de coches preparan modelos en sistemas CAD con todos los detalles.
- Los ingenieros de software deberían igualmente construir modelos de los sistemas de software.

# ¿CUANDO UTILIZAR UML?

- Cuando involucramos a varios desarrolladores trabajando y compartiendo información.
- Es vital una vez finalizado el proyecto, cuando se requiere algún cambio en el sistema.
- En desarrollos simples se hace algo de modelado, porque no utilizar el estándar?

# Representar un modelo

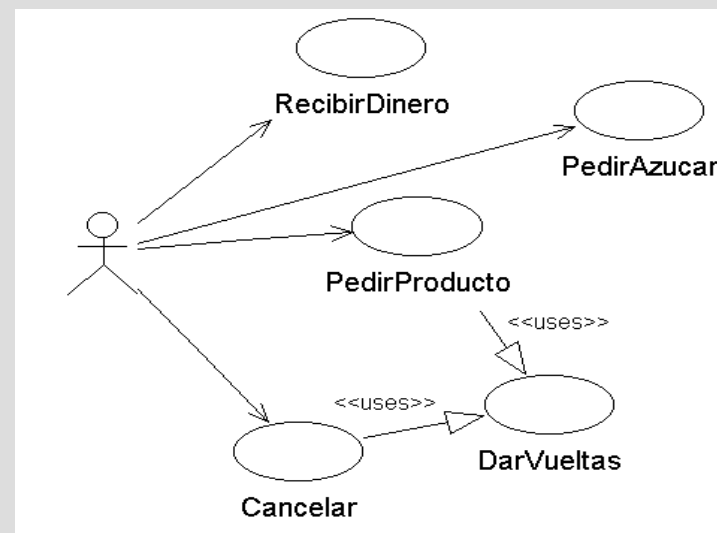
- Para la construcción de modelos, hay que centrarse en los detalles relevantes mientras se ignoran los demás, por lo cual con un único modelo no tenemos bastante.
- Varios modelos aportan diferentes vistas de un sistema los cuales nos ayudan a comprenderlo desde varios frentes.
- UML recomienda la utilización de nueve diagramas para representar las distintas vistas de un sistema.

# Máquina de café

- Supongamos que se requiere desarrollar el control de una máquina de entrega de café automática.
- La máquina debe permitir a una persona entregar una cantidad de dinero en monedas de 1, 2 o 5 pesos, escoger uno de los productos de acuerdo a su precio (café negro, café claro, caldo), escoger (si es pertinente) un nivel de azúcar y entregar el producto y las vueltas. El dinero que los usuarios introducen se guarda en un recipiente aparte al disponible para vueltas, el cual se encuentra ordenado por denominación.
- El usuario puede en cualquier momento antes de escoger el azúcar cancelar la operación, mediante un botón existente para este objetivo.

# Diagramas UML

- Diagrama de Casos de Uso: modela la funcionalidad del sistema agrupándola en descripciones de acciones ejecutadas por un sistema para obtener un resultado.

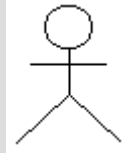


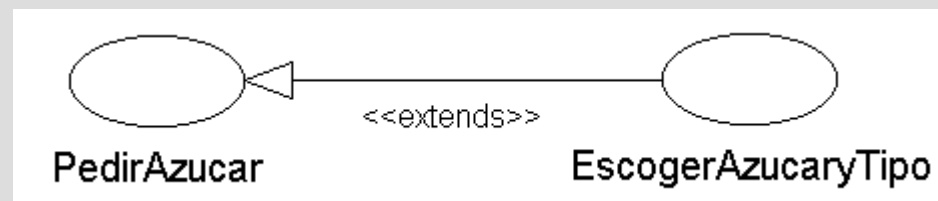
# Casos de Uso

- Se representa en el diagrama por una elipse, denota un requerimiento solucionado por el sistema.
- Cada caso de uso es una operación completa desarrollada por los actores y por el sistema en un diálogo.
- El conjunto de casos de uso representa la totalidad de operaciones desarrolladas por el sistema.



# Casos de uso

- Actor: es un usuario del sistema, que necesita o usa algunos de los casos de uso.
- Se representa mediante un:  acompañado de un nombre significativo
- Relaciones de casos de uso:
  - Comunica: relación entre un actor y un caso de uso
  - Usa: Relación entre dos casos de uso, denota la inclusión del comportamiento de un escenario en otro.
  - Extiende: Relación entre dos casos de uso, denota cuando un caso de uso es una especialización de otro.



# Diagramas de clases

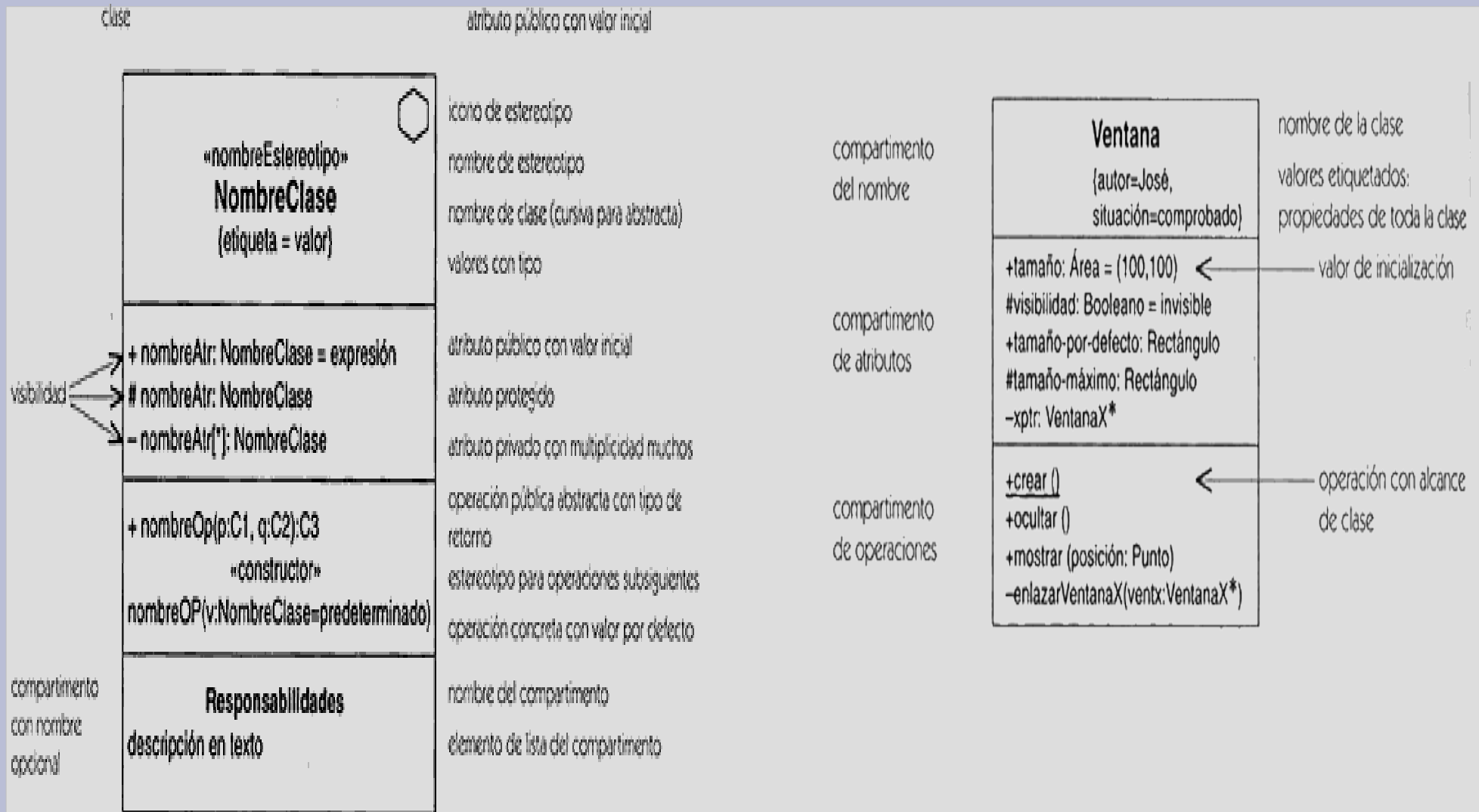
- Muestra las clases (descripciones de objetos que comparten características comunes) que componen el sistema y cómo se relacionan entre sí.
- El Diagrama de Clases es el diagrama principal para el análisis y diseño. Un diagrama de clases presenta las clases del sistema con sus relaciones estructurales y de herencia.
- La definición de clase incluye definiciones para atributos y operaciones.
- El modelo de casos de uso aporta información para establecer las clases, objetos, atributos y operaciones.

# Mecanismos de abstracción

- La clasificación es uno de los mecanismos de abstracción más utilizados.
- La clase define el ámbito de definición de un conjunto de objetos, y cada objeto pertenece a una clase, Los objetos se crean por instanciación de las clases.
- 
- Cada clase se representa en un rectángulo con tres compartimientos:
  - \* nombre de la clase
  - \* atributos de la clase
  - \* operaciones de la clase

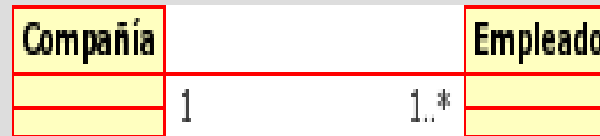
- Los atributos de una clase no deberían ser manipulables directamente por el resto de objetos. Por esta razón se crearon niveles de visibilidad para los elementos que son:
  - (-) Privado : es el más fuerte. Esta parte es totalmente invisible
  - (#) Los atributos/operaciones protegidos están visibles para las clases friends y para las clases derivadas de la original.
  - (+) Los atributos/operaciones públicos son visibles a otras clases.

# Diagrama de clases

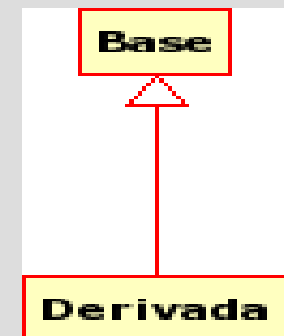


# Relaciones entre clases

- Los enlaces entre objetos pueden representarse entre las respectivas clases y sus formas de relación son:
  - Asociación y Agregación (vista como un caso particular de asociación)



- Generalización/Especialización.



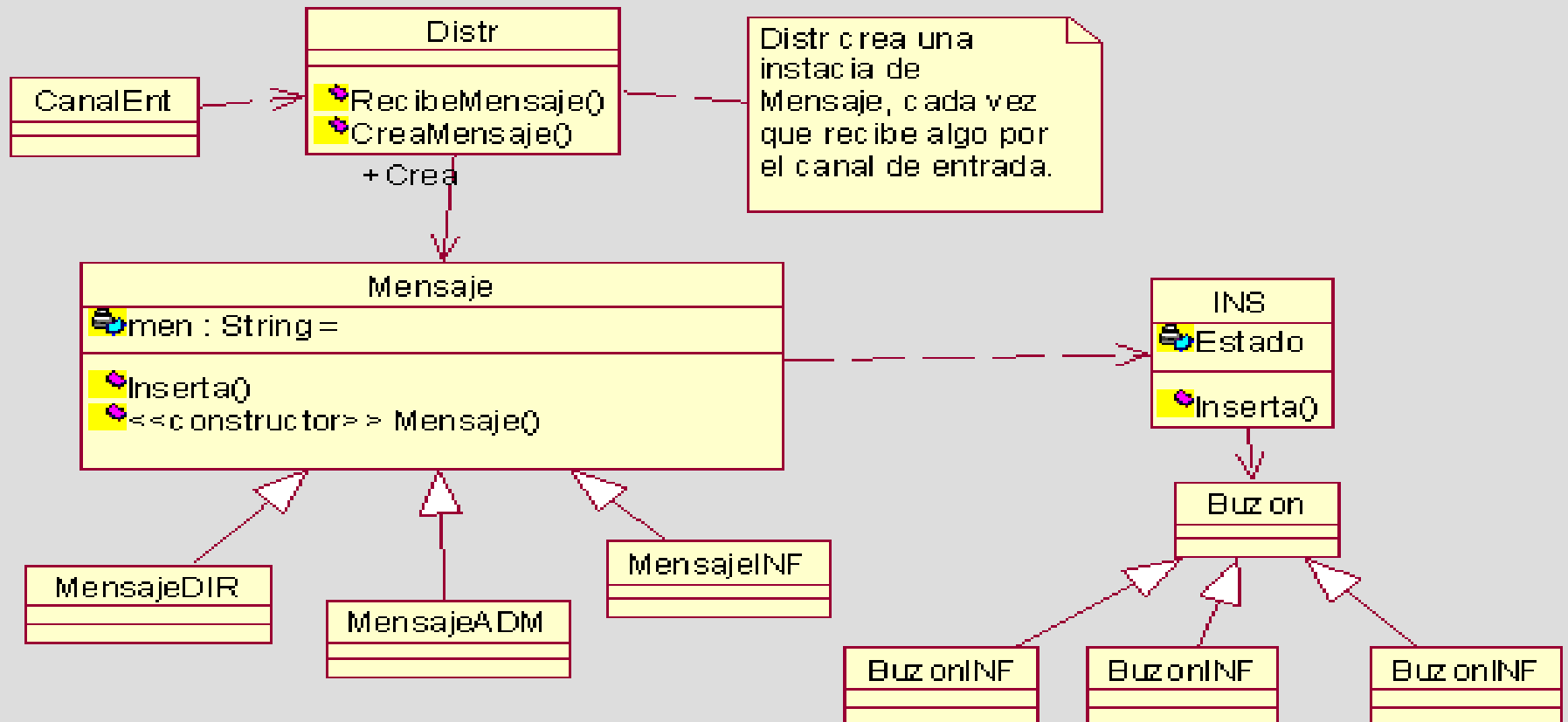
- Composición: Son asociaciones que representan acumulaciones muy fuertes. Esto significa que las composiciones también forman relaciones completas, pero dichas relaciones son tan fuertes que las partes no pueden existir por sí mismas. Únicamente existen como parte del conjunto, y si este es destruido las partes también lo son.

Acumulación



# Diagramas de objetos

- Muestra una serie de objetos (instancias de las clases) y sus relaciones.



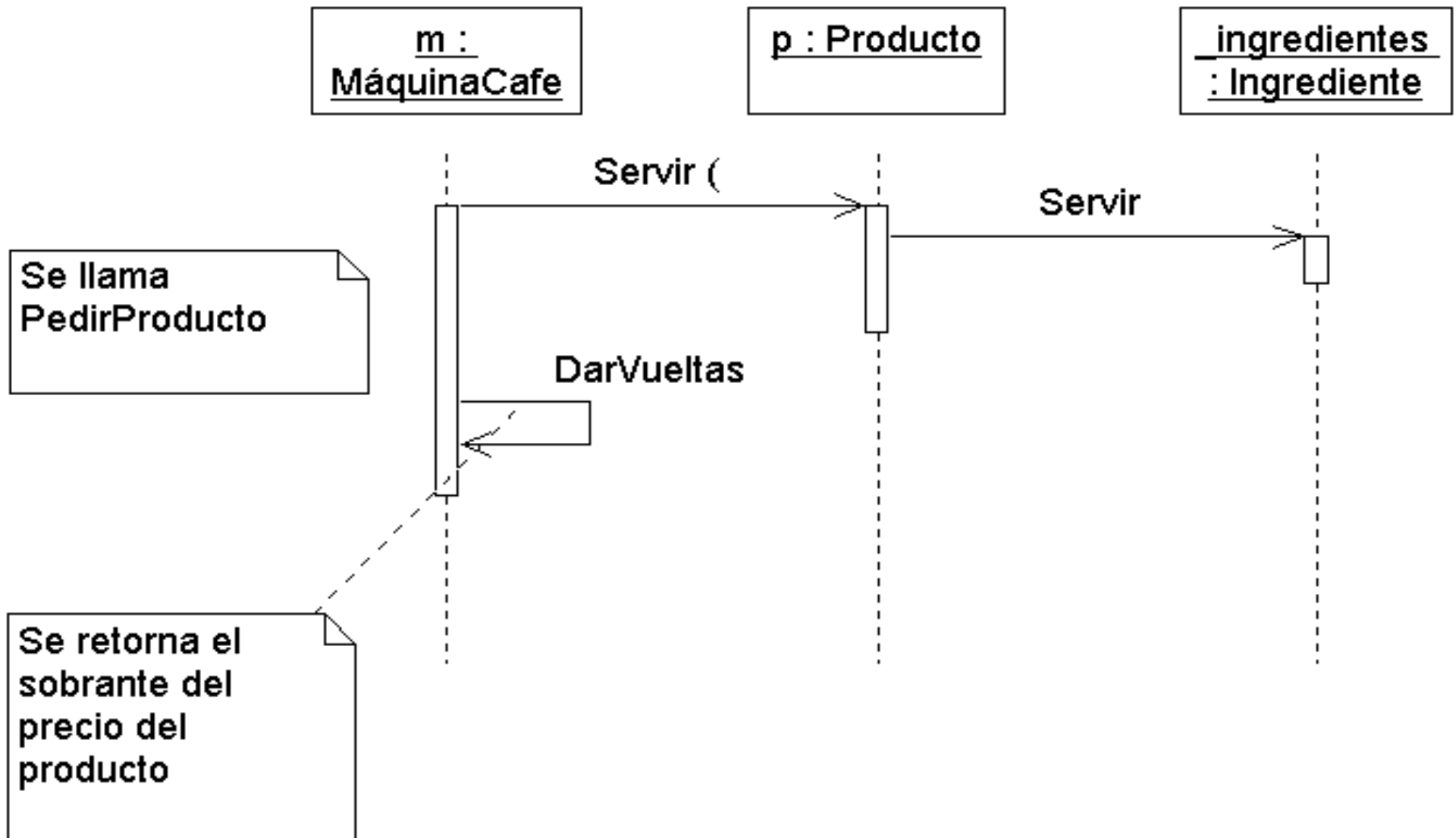
Forman parte de la parte estática del sistema



# Diagrama de secuencia

- Enfatiza la interacción entre los objetos y los mensajes que intercambian entre sí junto con el orden temporal de los mismos.
- El diagrama de secuencia forma parte del modelado dinámico del sistema.
- Se modelan las llamadas entre clases desde un punto concreto del sistema.
- Es útil para observar la vida de los objetos en sistema, identificar llamadas a realizar o posibles errores del modelado estático, que imposibiliten el flujo de información o de llamadas entre los componentes del sistema.

# Diagramas de secuencia

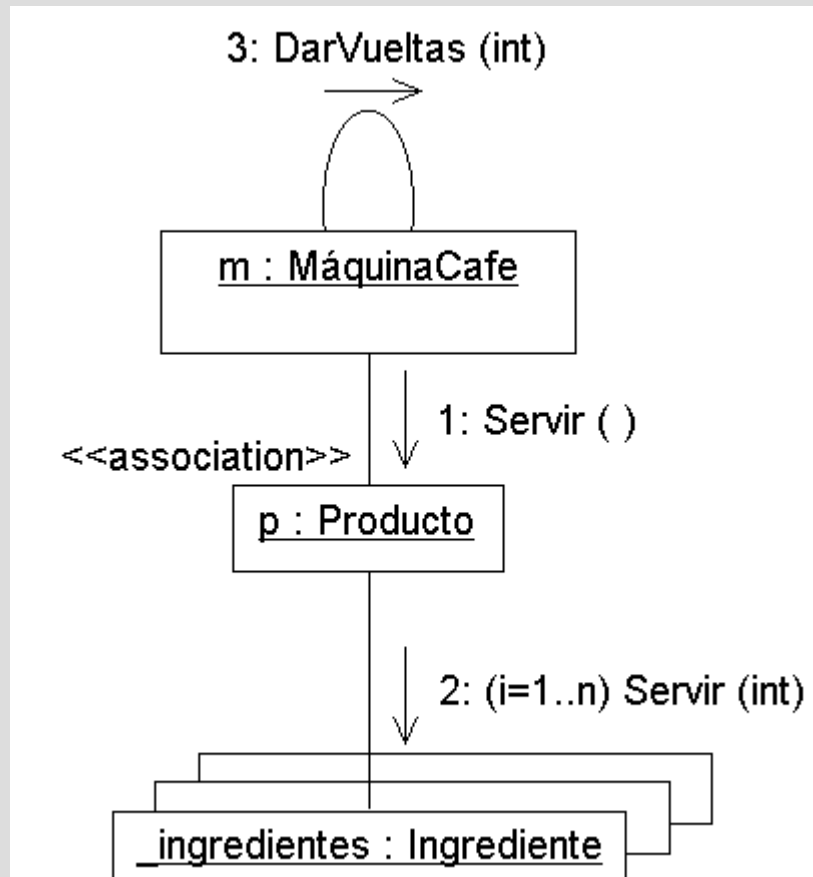


# Diagramas de secuencia

- Línea de vida de un objeto. Un objeto se representa como una línea vertical punteada con un rectángulo de encabezado y con rectángulos a través de la línea principal que denotan la ejecución de métodos.
- Activación. Muestra el periodo de tiempo en el cual el objeto se encuentra desarrollando alguna operación, bien sea por sí mismo o por medio de delegación a alguno de sus atributos. Se denota como un rectángulo delgado sobre la línea de vida del objeto.
- Mensaje. El envío de mensajes entre objetos se denota mediante una línea sólida dirigida, desde el objeto que emite el mensaje hacia el objeto que lo ejecuta.

# Diagrama de Colaboración

- Igualmente, muestra la interacción entre los objetos resaltando la organización estructural de los objetos en lugar del orden de los mensajes intercambiados.

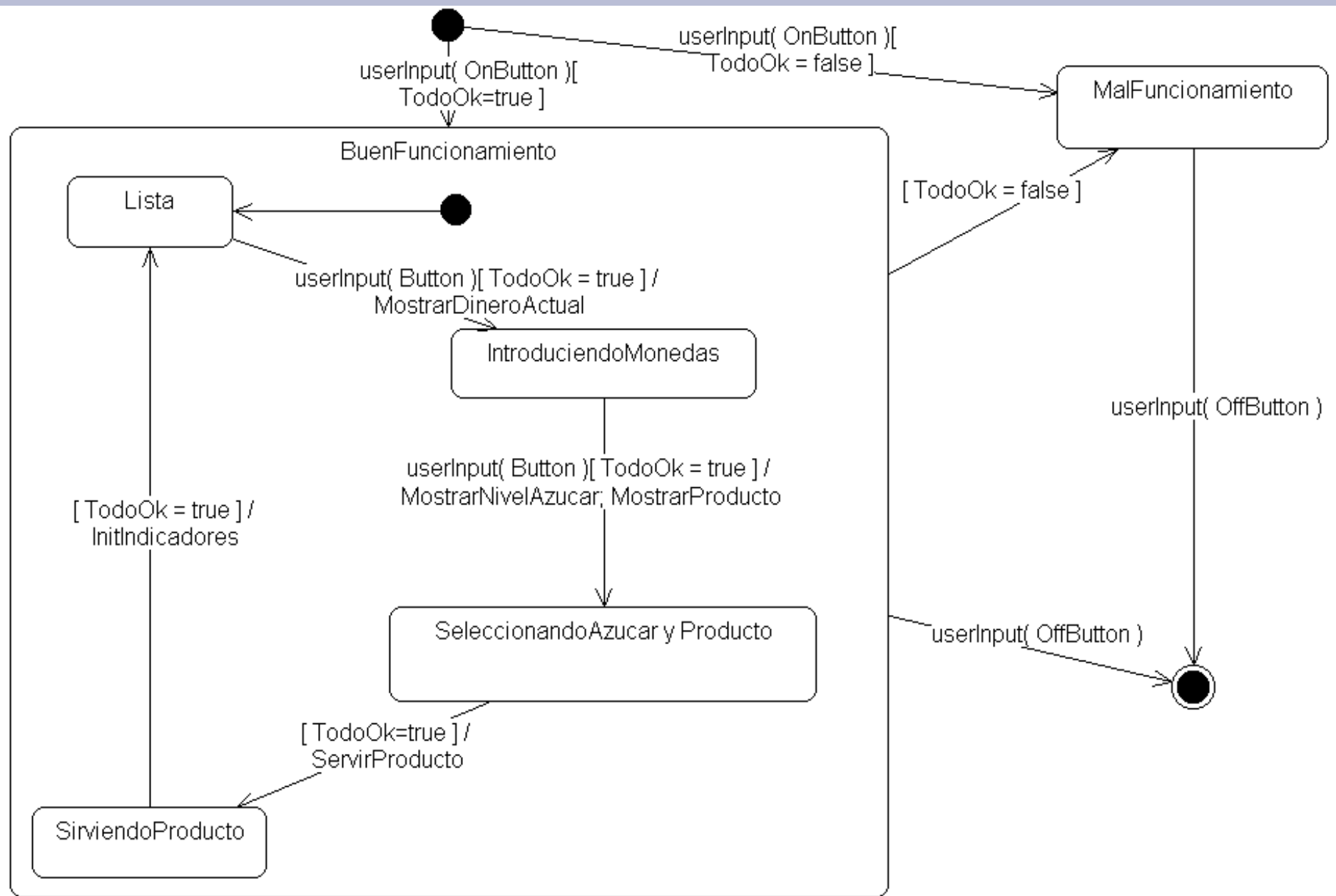


# Diagramas de Colaboración



- Un diagrama de colaboración es una forma alternativa al diagrama de secuencia de mostrar un escenario. Este tipo de diagrama muestra las interacciones entre objetos organizadas entorno a los objetos y los enlaces entre ellos.
- Conceptos:
  - Objeto: Se representa con un rectángulo que contiene el nombre y la clase del objeto en un formato *nombreObjeto : nombreClase*.
  - Enlaces: Un enlace es una instancia de una asociación en un diagrama de clases. Se representa como una línea continua que une a dos objetos, acompañada por un número que indica el orden dentro de la interacción.
  - Flujo de mensajes: Expresa el envío de un mensaje. Se representa mediante una flecha dirigida cerca de un enlace.

# Diagramas de Estado

- Modela el comportamiento de acuerdo con eventos.
- Un **estado** es una condición durante la vida de un objeto, de forma que cuando dicha condición se satisface se lleva a cabo alguna acción o se espera por un evento.
- El estado de un objeto se puede caracterizar por el valor de uno o varios de los atributos de su clase, además, el estado de un objeto también se puede caracterizar por la existencia de un enlace con otro objeto.
- Los diagramas de estados muestran el comportamiento de los objetos, es decir, el conjunto de estados por los cuales pasa un objeto durante su vida, junto con los cambios que permiten pasar de un estado a otro.



# Diagrama de estados

- El estado inicial se marca como: 
- El estado final se marca como: 
- Eventos
- Es una ocurrencia que puede causar la transición de un estado a otro de un objeto. Esta ocurrencia puede ser una de varias cosas:
  - Condición que toma el valor de verdadero o falso
  - Recepción de una señal de otro objeto en el modelo
  - Recepción de un mensaje
  - Paso de cierto período de tiempo, después de entrar al estado o de cierta hora y fecha particular



**Gracias**