

# Análisis Orientado a Objetos en UML

## TEMA 2

Departamento de Lenguajes y Sistemas  
Informáticos

## Tema 2. Análisis Orientado a Objetos en UML

### 2.1. Introducción al Análisis Orientado a Objetos en UML

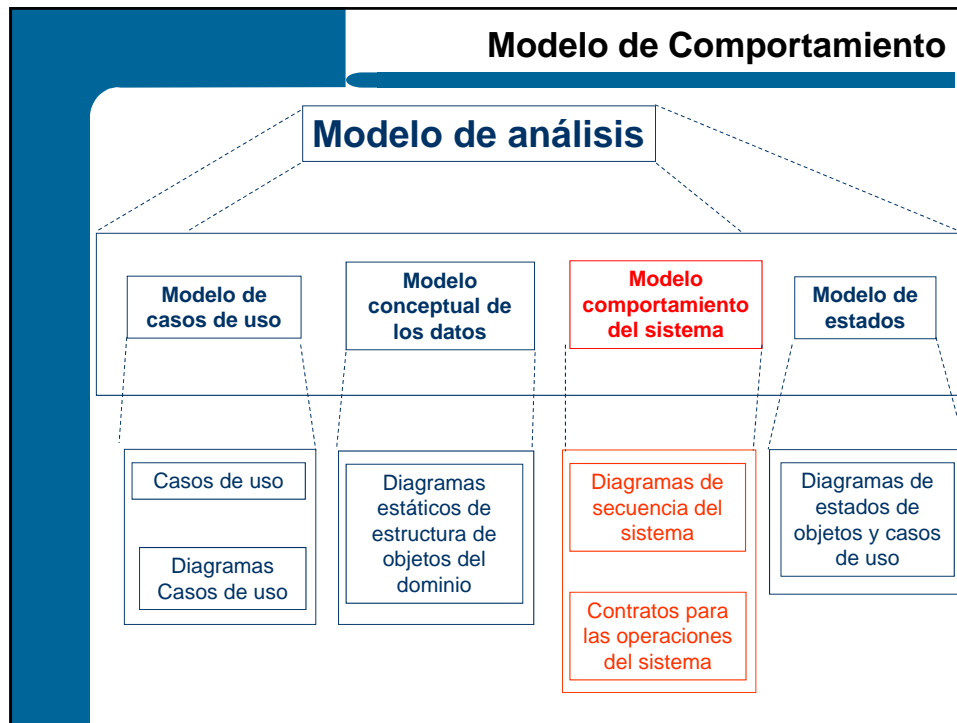
### 2.2 Modelo de Casos de Uso en UML

### 2.3. Modelo Conceptual de Datos en UML

### 2.4. Modelo de Comportamiento del Sistema en UML

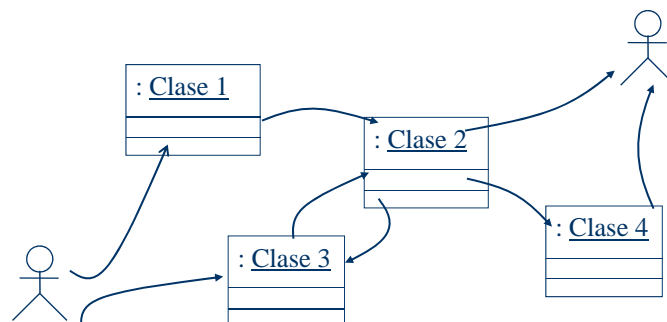
### Bibliografía

## Modelo de Comportamiento

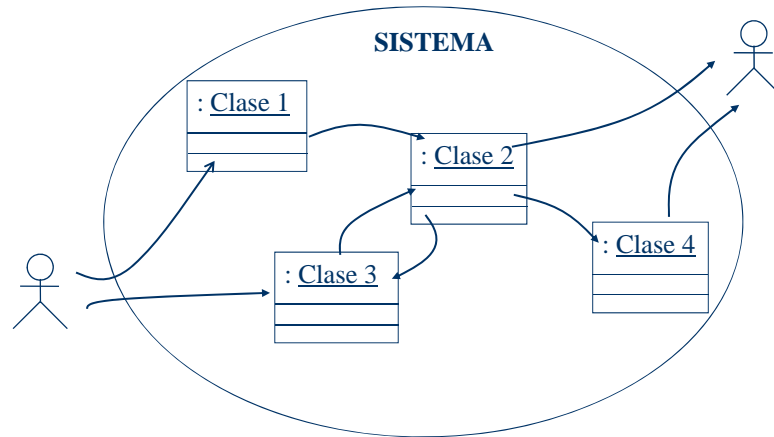


## Descripción del comportamiento en OO

- Los **objetos** se comunican mediante la **invocación de operaciones** de otros objetos



## “Especificación” del comportamiento en OO



- Consideramos un **“sistema”** que engloba a todos los objetos
- La **especificación del comportamiento** se hace con el **modelo de comportamiento del “sistema”**

## Modelo de comportamiento del “sistema”

- **Diagramas de secuencia del sistema:**
  - Muestran la **secuencia de eventos entre los actores y el sistema**
  - Permiten identificar las **operaciones** del sistema
- **Contratos para las operaciones del sistema:**
  - Describen el **efecto** de las **operaciones del sistema**

## Diagramas de secuencia del sistema

- **Objetivos:**
  - Identificar los **eventos** y las **operaciones** del sistema
- **Punto de partida:**
  - **Casos de uso**
  - La descripción de los diagramas de secuencia del sistema es **posterior a la descripción de los casos de uso**
- **Casos de uso:**
  - Describen cómo los **actores interactúan con el sistema software**
  - El **actor genera eventos** hacia el sistema que exigen la **ejecución de alguna operación como respuesta** (durante la interacción)
  - **A partir de los casos de uso** podemos **identificar cuáles son los eventos que van de los actores hacia el sistema**

## Diagramas de secuencia del sistema

- Muestra para un **escenario particular** de un caso de uso:
  - los **eventos generados** por los **actores externos**
  - su **orden**
  - las **operaciones del sistema que resultan de la invocación**
- Definiremos un diagrama de secuencia para cada **escenario relevante** de un caso de uso

## Ejemplo: Venta de productos

**Caso de uso:** Venta de productos

**Descripción:** Capturar una venta y su pago en efectivo

**Actores:** Cliente, Cajero (principal)

**Resumen:** un cliente llega a la caja con productos para comprar. El cajero registra los productos y gestiona el pago en efectivo. Al acabar, el cliente se va con los productos.

**Escenario principal:**

1. El caso de uso comienza cuando un *Cliente* llega a la caja con los productos para comprar.
2. El *Cajero* indica que comienza una nueva venta e introduce el punto de venta (*pv*).
3. El *Sistema* comprueba que existe el punto de venta de la compra y registra el inicio de una nueva venta.
4. El *Cajero* introduce el identificador (*prod*) y la cantidad (*cant*) de cada producto.
5. El *Sistema* comprueba que existe el producto y registra la línea de venta.
6. Al acabar la entrada de productos el *Cajero* lo indica.
7. El *Sistema* calcula el total de la compra y lo muestra.

## Ejemplo: Venta de productos

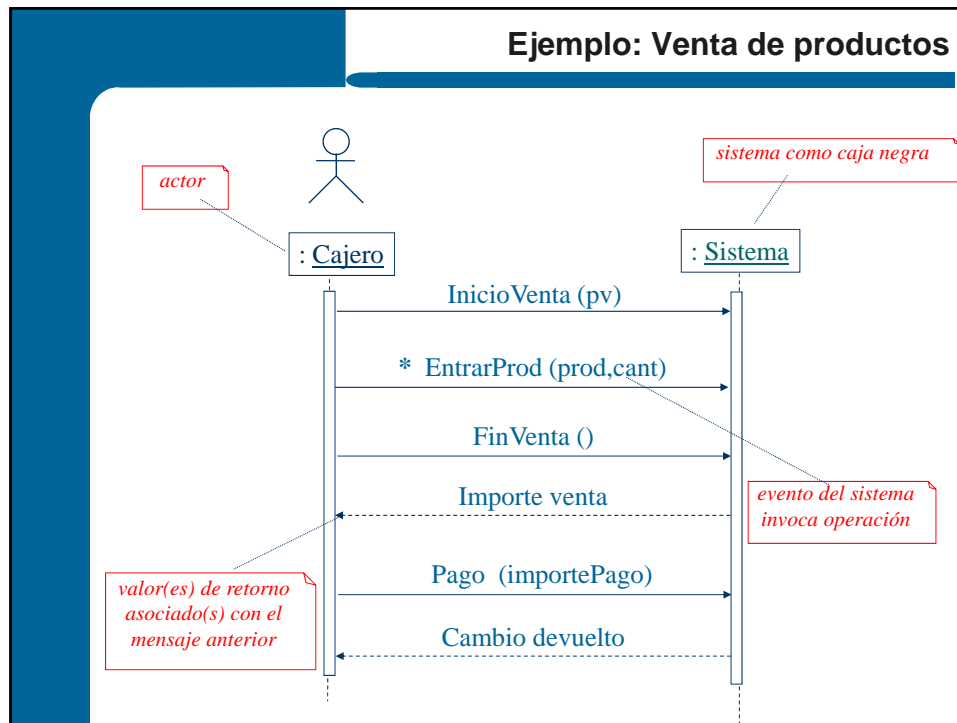
**Escenario principal (cont.):**

8. El *Cajero* dice el total de la compra al *Cliente*.
9. El *Cliente* entrega una cantidad de dinero posiblemente superior al total de la compra.
10. El *Cajero* introduce el dinero que ha recibido (*importePago*)
11. El *Sistema* calcula y muestra el cambio al *Cliente* y registra el pago.
12. El *Cajero* deposita el dinero recibido en la caja y coge el cambio. El *Cajero* da el cambio al *Cliente*.
13. El *Cliente* se va con los productos comprados.

**Escenarios alternativos**

- 3a. El punto de venta de la compra no existe.
  1. El *Sistema* indica el error.
  2. El *Cajero* vuelve a introducir el punto de venta.
- 5a. El producto no existe.
  1. El *Sistema* indica el error.
  2. El *Cajero* vuelve a introducir el identificador del producto.

## Ejemplo: Venta de productos



## Diagramas de secuencia del sistema

### Construcción de un diagrama de secuencia

1. Dibujar una **línea vertical** que representa el sistema.
2. Dibujar una **línea para cada actor** que interactúa **directamente** con el sistema.
3. Del escenario del caso de uso, **identificar los eventos externos** generados por los actores. **Mostrarlos en el diagrama**

### Eventos y operaciones

- **Evento del sistema:** evento externo generado por un actor.
- **Operación del sistema:** Operación interna que se ejecuta como respuesta a la comunicación del evento.

La comunicación de un evento del sistema provoca la ejecución de una operación del sistema con el **mismo nombre** y los **mismos parámetros**

### Operaciones del sistema

- Las **operaciones del sistema** se agrupan como operaciones del tipo especial "sistema"
- En cambio, las operaciones no se asignan a objetos concretos durante la etapa de especificación

#### Ejemplo:

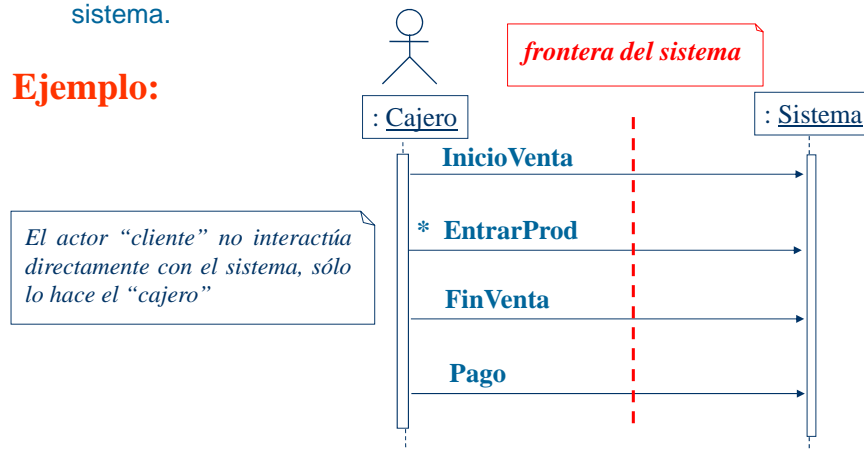
SISTEMA
InicioVenta (pv)
EntrarProd (prod,cant)
FinVenta ( )
Pago (importePago)

## Diagramas de secuencia del sistema

### Eventos y el límite del sistema

- Para **identificar los eventos del sistema** es necesario haber delimitado claramente la **frontera del sistema**.
- Los **eventos del sistema** son los que estimulan **directamente** el sistema.

#### Ejemplo:



## Contratos de las operaciones

### Concepto de Contrato

- Describen el **comportamiento** del sistema en función de los **cambios de estado de los objetos del Modelo Conceptual de Datos** (Modelo de Dominio) **después de la ejecución de una operación** del sistema.
- Definición basada en:
  - **Precondiciones** relevantes sobre los objetos del Modelo Conceptual de Datos antes de la ejecución de la operación.
  - **Postcondiciones** que describen los cambios de estado de los objetos del Modelo Conceptual de Datos.



## Contratos de las operaciones

### Plantilla de un Contrato

Nombre operación	Signatura de la operación
Responsabilidades	Descripción informal del propósito de la operación
Referencias cruzadas (opcional)	Casos de uso en los que puede tener lugar esta operación
Precondiciones	Suposiciones relevantes sobre el estado del sistema o de los objetos del Modelo Conceptual, antes de ejecutar la operación. Suposiciones no triviales
Postcondiciones	Estado de los objetos del Modelo Conceptual después de que se ejecute la operación

## Contratos de las operaciones

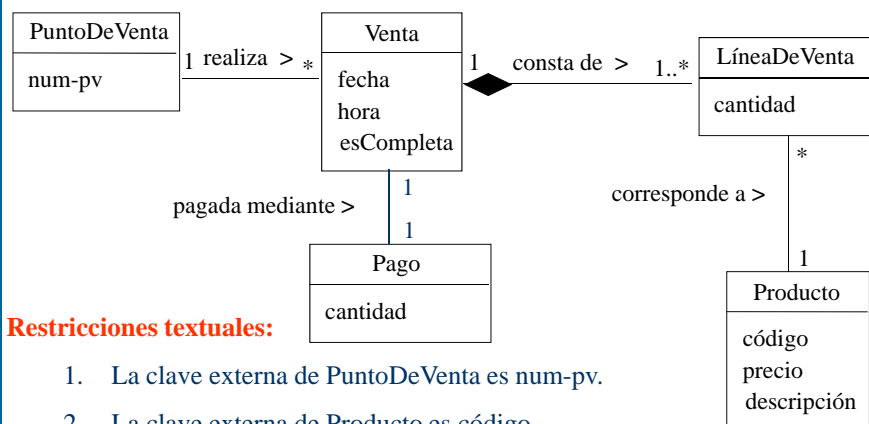
### Postcondiciones

- **No son acciones** que se ejecutarán durante la operación
- Son los **cambios de estado en los objetos del Modelo Conceptual** de datos que se producen tras la ejecución de la operación. Indican:
  - **Creación y eliminación de objetos**
  - **Creación y eliminación de asociaciones**
  - **Modificación de atributos**
- Escribimos las postcondiciones a partir del **modelo conceptual**.
- Las **postcondiciones serán incompletas** y se descubrirán detalles en el diseño.

### Guía para crear los Contratos

- Identificar operaciones del sistema en DSS
- Construir un contrato para operaciones complejas y quizá sutiles en sus resultados, o que no están claras en el caso de uso.
- Describir postcondiciones
  - Creación/Eliminación de objetos
  - Creación/Eliminación de asociaciones
  - Modificación de atributos
  - ¡No olvidar crear asociaciones!
  - Se puede usar OCL (*Object Constraint Language, Lenguaje de Restricciones de Objetos*)

### Ejemplo: Esquema conceptual de partida



#### Restricciones textuales:

1. La clave externa de PuntoDeVenta es num-pv.
2. La clave externa de Producto es código.
3. Un punto de venta no puede tener más de una venta el mismo día y a la misma hora

## Ejemplo: Operación InicioVenta

Operación: InicioVenta (pv)

Responsabilidades: Iniciar el registro de una nueva venta

Referencias Cruzadas: **Caso de Uso:** Venta de Productos

Precondiciones: Existe un objeto punto de venta **PV** con *num-pv = pv*

Postcondiciones:

- Se creó una instancia **V** de *Venta* (creación de objeto)
- Se inicializaron los atributos de **V** (**V.fecha** = fecha del sistema, **V.hora** = hora del sistema y **V.esCompleta** = "false")
- Se asoció **V** al objeto **PV** (creación de enlace)

## Ejemplo: Operación EntrarProd

Operación: EntrarProd (prod, cant)

Responsabilidades: Registrar una línea de la venta en curso

Referencias Caso de Uso: Venta de Productos

Precondiciones Cruzadas: Existe una venta en curso **V** y un producto **P** con *código = prod*

Postcondiciones:

- Se creó una instancia **LDV** de *LíneaDeVenta* en el objeto **V** (creación de un objeto componente de **V**)
- Se asignó *wcant* a **LDV.cantidad** (**LDV.cantidad** = *cant*) (modificación de atributos)
- Se asoció **LDV** al producto **P** (creación de enlace)

## Ejemplo: Operación FinVenta

Operación: FinVenta()

Responsabilidades: Finalizar el registro de una venta y mostrar el importe a pagar (importe venta)

Referencias Cruzadas: **Caso de Uso:** Venta de Productos

Precondiciones: Existe una venta en curso **V**.

Postcondiciones:

- Se asignó “**true**” a **V.esCompleta** (modificación de atributo)

## Ejemplo: Operación Pago

Operación: Pago (importePago)

Responsabilidades: Registrar el pago de la venta y mostrar el cambio a devolver

Referencias Cruzadas: **Caso de Uso:** Venta de Productos

Precondiciones: Existe una venta en curso **V** y **V.esCompleta** = “**true**”

Postcondiciones:

- Se creó una instancia **P** de **Pago** (creación de objeto)
- Se asignó **importePago** a **P.cantidad** (modificación de atributo)
- Se asoció **P** con la venta en curso **V** (creación de enlace)

## Referencias

- G. Booch, J. Rumbaugh, I. Jacobson, “*El Lenguaje Unificado de Modelado. Guía de Usuario*”, Addison Wesley, 1999.
- G. Booch, J. Rumbaugh, I. Jacobson, “*El Lenguaje Unificado de Modelado. Manual de Referencia*”, Addison Wesley, 1999.
- C. Larman, “*UML y Patrones: Una introducción al análisis y diseño orientado a objetos y al proceso unificado*”, Cap. 13, 15