

# Unidad 1: UML 2.0

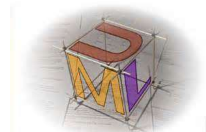
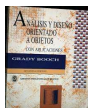
— Diseño de Sistemas – Curso: 3K3 —

Docentes: Ing. Marcela F. Cattaneo  
Ing. Germán Vélez

## UML

### Surgimiento de UML

- Los **lenguajes de modelado O.O** aparecieron entre la mitad de los años '70 y finales de los '80.
- El número de métodos OO se incrementó increíblemente entre 1989 y 1994 **y cada metodología tenía sus propias notaciones y simbología.**
- Comenzaron a destacarse: la metodología de **Booch**, OOSE de **Jacobson** y OMT de **Rumbaugh**.



- En Enero de 1.997 se presentó UML 1.0 al Object Management Group (OMG)

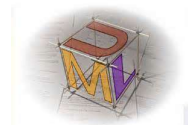
## UML 2.0

- **Por qué modelamos?**

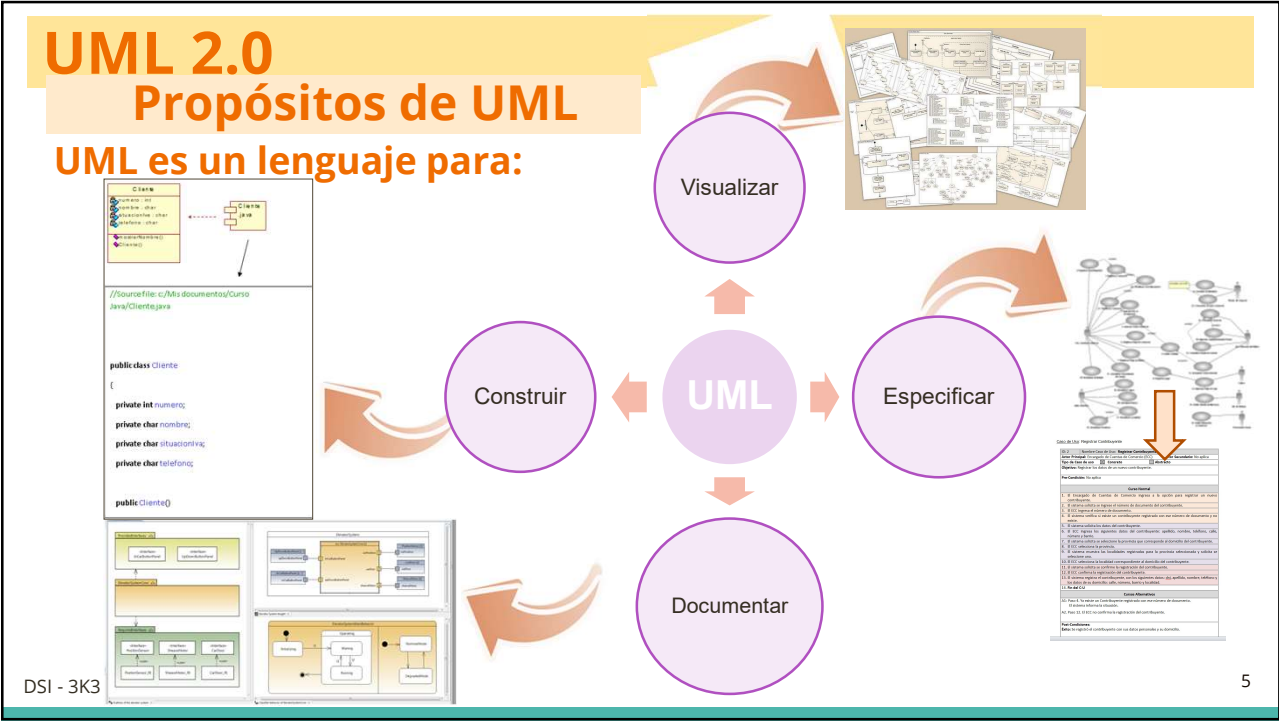
1. Los modelos ayudan a visualizar cómo es o queremos que sea un sistema.
2. Los modelos permiten especificar la estructura o el comportamiento de un sistema.
3. Los modelos proporcionan plantillas que sirven de guía en la construcción de un programa.
4. Los modelos documentan las decisiones que se han adoptado.

## UML 2.0

- **Qué es UML?**



- El **Lenguaje Unificado de Modelado** es un lenguaje estándar para escribir “planos” de software.
- Un **lenguaje** proporciona un vocabulario y las reglas para combinar palabras de ese vocabulario con el objetivo de posibilitar la comunicación.
- Un **lenguaje de modelado** es un lenguaje cuyo vocabulario y reglas se centran en la representación conceptual y física de un sistema.



DSI - 3K3

5

# UML 2.0

## Por qué “unificado”?

- La unificación de UML no es sólo histórica, sino que trata de estar unificado en distintos dominios:
  - **Ciclo de vida de desarrollo:** UML proporciona sintaxis visual para modelar software desde los requisitos a la implementación.
  - **Dominios de aplicación:** UML se ha utilizado para modelar de todo, desde sistemas incorporados en tiempo real a sistemas de soporte a la toma de decisión.

DSI - 3K3 - UML 2.0 (Unidad 1)

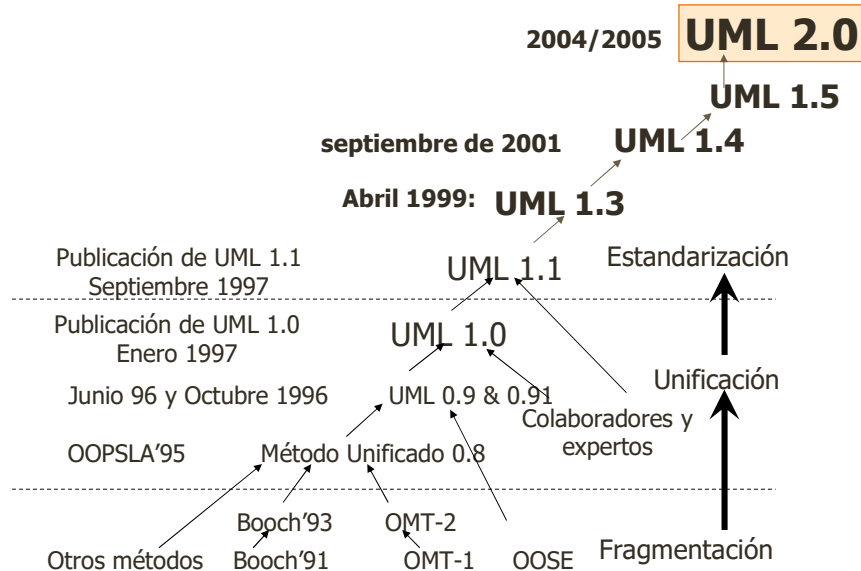
6

## UML 2.0

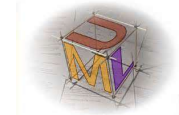
### Por qué “unificado”?

- **Lenguajes y plataforma de implementación:** UML es neutro tanto en lenguaje como en plataforma.
- **Procesos de desarrollo:** UML puede soportar muchos otros procesos de ingeniería de software además de UP y sus variantes.
- **Sus propios conceptos internos:** UML trata de ser coherente y uniforme en su aplicación de un pequeño grupo de conceptos internos.

## Evolución de UML



## Autores de UML



DSI - 3K3 - UML 2.0 (Unidad 1)

9

## UML 2.0

- **Objetivos** de esta nueva versión:
  - Hacer el lenguaje de modelado más extensible.
  - Permitir la validación y ejecución de modelos.
- Cambios en UML 2.0
  - UML 2.0 presenta numerosa sintaxis visual nueva.
  - Realiza muchos cambios sintácticos respecto de las versiones 1.x.



capturar mucho más comportamiento de modo que se crearan herramientas que soporten la automatización y generación de código ejecutable, a partir de modelos UML.

DSI - 3K3 - UML 2.0 (Unidad 1)

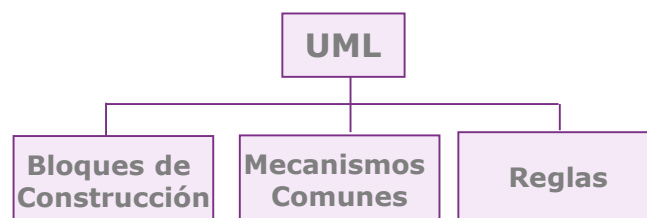
10

## UML 2.0

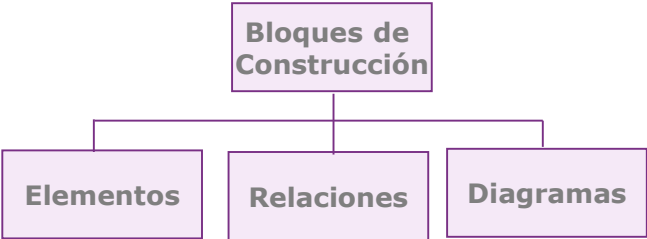
### • MDA y UML

- MDA = *Model Driven Architecture*, es una iniciativa de la OMG que define el futuro de UML.
- MDA define una visión de cómo se puede desarrollar software basándose en modelos.
- La esencia de esta visión es que los modelos dirigen la producción de la arquitectura de software ejecutable.

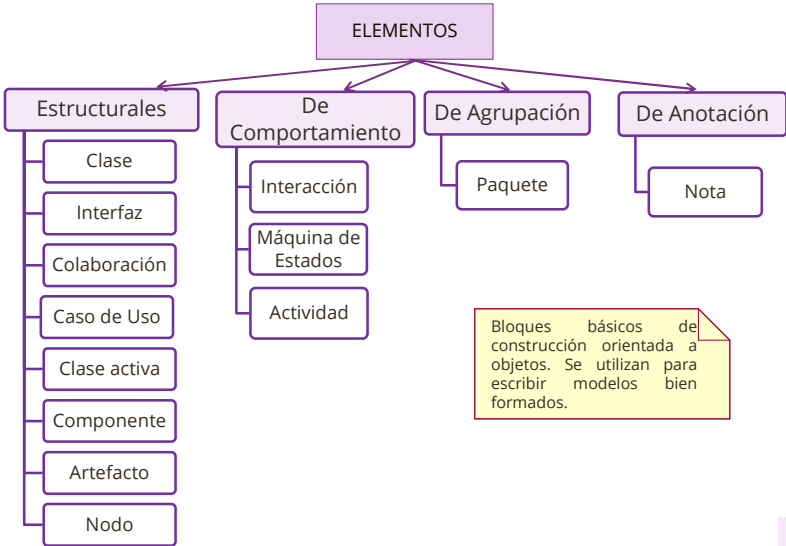
## Estructura UML 2.0



# Bloques de Construcción UML 2.0



# Bloques de Construcción UML 2.0



# UML 2.0

## Elementos estructurales

Clase	Descripción de un conjunto de objetos que comparten los mismos atributos, operaciones relaciones y semántica. Una clase implemente una o más interfaces.	
Interfaz	Colección de operaciones que especifican un servicio de una clase o componente. Una interfaz puede representar el comportamiento completo de una clase sólo una parte. Debe tener al menos una clase que la implemente.	
Colaboración	Representa una interacción y es una sociedad de roles y otros elementos que colaboran para lograr un comportamiento cooperativo mayor que la suma de los comportamientos de sus elementos.	
Caso de Uso	Conjunto de secuencias de acciones que un sistema ejecuta y que produce un resultado observable de interés para un actor particular.	

DSI - 3K3 - UML 2.0 (Unidad 1)

15

# UML 2.0

## Elementos estructurales

Clase activa	Una clase cuyos objetos tienen uno o más procesos o hilos de ejecución y por lo tanto pueden dar origen a actividades de control. Sus objetos representan elementos cuyo comportamiento es concurrente con otros elementos.	
Componente	Es una parte modular del diseño del sistema que oculta su implementación tras un conjunto de interfaces externas. La implementación de un componente puede expresarse conectando partes y conectores; las partes pueden incluir componentes más pequeños.	
Artefacto	Es una parte física y reemplazable de un sistema que contiene información física. En un sistema hay diferentes tipos de artefactos de despliegue, como archivos de código fuente, ejecutables, scripts, librerías, etc.	
Nodo	Es un recurso físico que existe en tiempo de ejecución y representa un recurso computacional, que dispone de memoria y con frecuencia capacidad de almacenamiento. Un conjunto de artefactos puede residir en un nodo.	

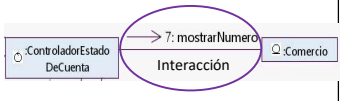


DSI - 3K3 - UML 2.0 (Unidad 1)

16



# UML 2.0

## Elementos de comportamiento


Interacción	Es un comportamiento que comprende un conjunto de mensajes intercambiados entre un conjunto de objetos, para alcanzar un propósito específico. Una interacción involucra a muchos elementos; incluye mensajes, acciones y enlaces (conexiones entre objetos)	
Máquina de Estados	Es un comportamiento que especifica las secuencias de estados por las que pasa un objeto o una interacción durante su vida en respuesta a eventos.	
Actividad	Es un comportamiento que especifica la secuencia de pasos que ejecuta un proceso computacional.	

- En una interacción, el énfasis se pone en el conjunto de objetos que interactúan.
- En una máquina de estados, el énfasis se pone en el ciclo de vida de un objeto cada vez.
- En una actividad, el énfasis se pone en los flujos entre los pasos sin mirar qué objeto ejecuta cada paso.

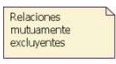
# UML 2.0

## Elementos de agrupación y anotación

### • Elemento de Agrupación

Paquete	Estos elementos, son las partes organizativas de los modelos UML, representan las cajas en las que puede descomponerse un modelo. Hay un elemento de agrupación principal: los paquetes, que representan un mecanismo de propósito general para organizar elementos en grupos. Un paquete es puramente conceptual, sólo existe en tiempo de desarrollo (al contrario que los componentes que existen en tiempo de ejecución).	
---------	---	---

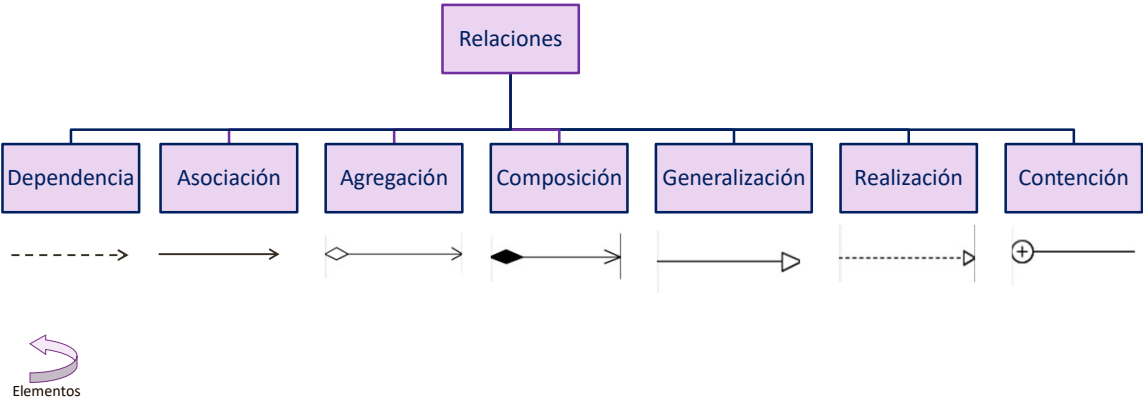
### • Elemento de Anotación

Nota	Estos elementos, son las partes explicativas de los modelos UML, son comentarios que se pueden aplicar para hacer observaciones sobre cualquier elemento del modelo. Hay un tipo principal de elemento de anotación llamado "Nota" que se utiliza para mostrar restricciones y comentarios.	
------	---	---



# UML 2.0

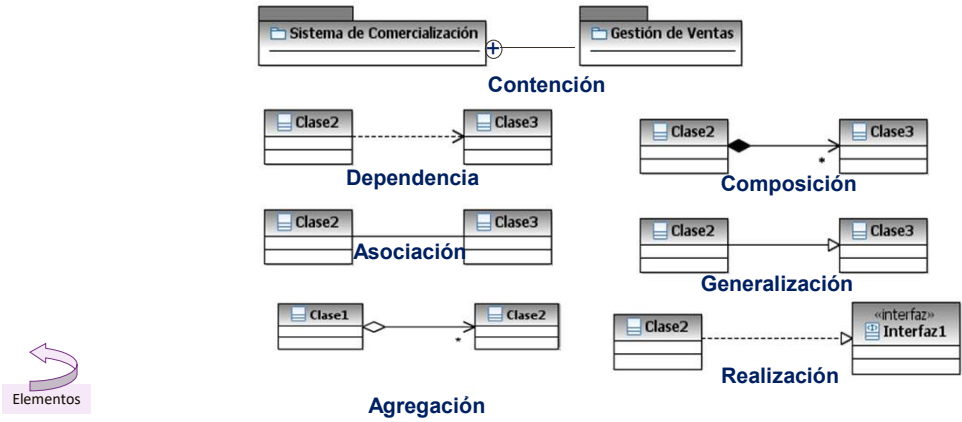
## Bloques de construcción de UML



# UML 2.0

## Bloques de construcción de UML

- Relaciones en UML 2.0

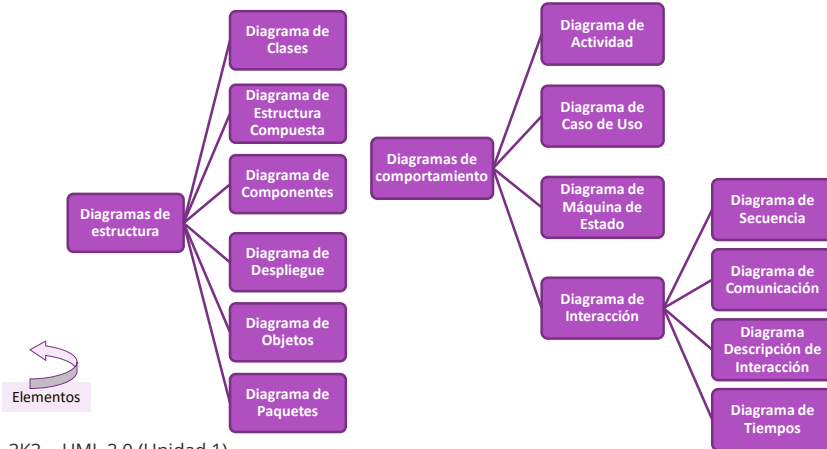


# UML 2.0

## Bloques de construcción de UML

- Diagramas →

Representación gráfica de un conjunto de elementos que permiten visualizar un sistema desde diferentes perspectivas, visualizado generalmente como un grafo conexo de nodos (elementos) y arcos (relaciones).



DSI - 3K3 - UML 2.0 (Unidad 1)

21

# UML 2.0

## Diagramas de Estructura

- Los diagramas estructurales representan elementos y así componen un sistema o una función.
- Estos diagramas pueden reflejar:
  - las relaciones estáticas de una estructura, como lo hacen los diagramas de clases o de paquetes
  - o arquitecturas en tiempo de ejecución, tales como diagramas de Objetos o de Estructura de Composición.



DSI - 3K3 - UML 2.0 (Unidad 1)

22

## UML 2.0

### Diagramas de Comportamiento

- Los diagramas de comportamiento representan las características de comportamiento de un sistema o proceso de negocios
- Incluyen a los diagramas de: actividades, casos de uso, máquinas de estados, tiempos, secuencias, visión de interacciones y comunicaciones.



## UML 2.0 – Diagramas de Estructura

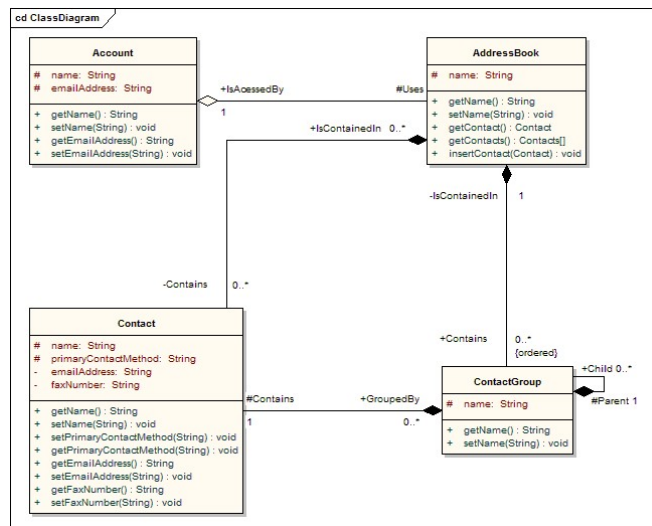
### Diagramas de Clase

- El diagrama de Clase muestra los bloques de construcción de cualquier sistema orientado a objetos.
- Los diagramas de clases describen la vista estática del modelo o parte del modelo, describiendo que atributos y comportamientos tienen en lugar de detallar los métodos para realizar operaciones.
- Los diagramas de Clase son más útiles para ilustrar relaciones entre clases e interfaces. Las generalizaciones, agregaciones, y asociaciones son todas valiosas al reflejar herencias, composición o uso, y conexiones respectivamente.

## UML 2.0 – Diagramas de Estructura

### Diagramas de Clase

- Ejemplo:



DSI - 3K3 - UML 2.0 (Unidad 1)

25

## UML 2.0 – Diagramas de Estructura

### Diagramas de Objetos

- Un diagrama de objetos puede ser considerado un caso especial de un diagrama de clase.
- Los diagramas de objetos utilizan un subconjunto de los elementos de un diagrama de clases con el fin de enfatizar la relación entre las instancias de clases en algún momento en el tiempo. Son útiles en la comprensión de diagramas de clases.
- No muestran nada diferentes en arquitectura a los diagramas de clases, sino que reflejan la multiplicidad y roles.

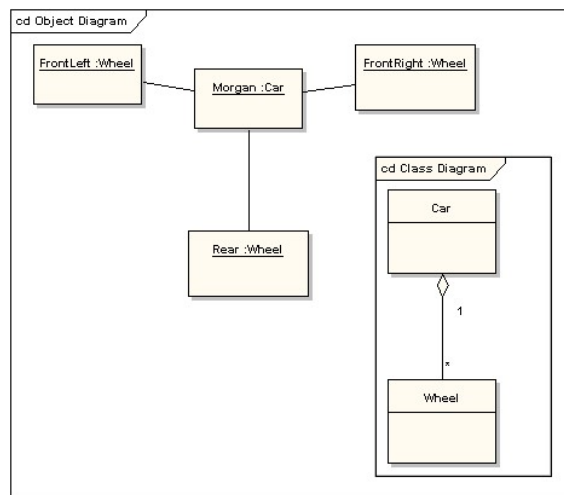
DSI - 3K3 - UML 2.0 (Unidad 1)

26

## UML 2.0 – Diagramas de Estructura

### Diagramas de Objetos

- Ejemplo:



DSI - 3K3 - UML 2.0 (Unidad 1)

27

## UML 2.0 – Diagramas de Estructura

### Diagramas de Estructuras Compuestas

- Un diagrama de estructura compuesta es un diagrama que muestra la estructura interna de un clasificador, incluyendo sus puntos de interacción a otras partes del sistema. Esto muestra la configuración y relación de las partes que juntas realizan el comportamiento de clasificador contenido.
- La *estructura interna* es la implementación de un componente a través de un conjunto de partes conectadas de una manera específica.

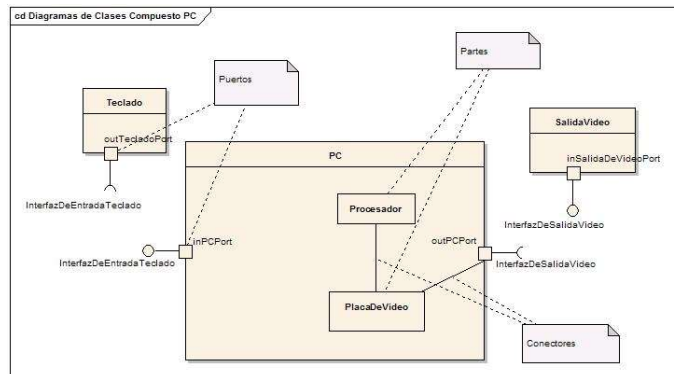
DSI - 3K3 - UML 2.0 (Unidad 1)

28

## UML 2.0 – Diagramas de Estructura

### Diagramas de Estructuras Compuestas

- Los conceptos principales para describir la estructura interna son: Partes, Puertos y Conectores.
- Ejemplo “mi PC”



DSI - 3K3 - UML 2.0 (Unidad 1)

29

## UML 2.0 – Diagramas de Estructura

### Diagramas de Componentes

- Los diagramas de componentes muestran las piezas de software, controladores embebidos, etc, que componen un sistema.
- Un diagrama de componentes tiene un mayor nivel de abstracción que un diagrama de clase - por lo general un componente es ejecutado por uno o más clases (u objetos) en tiempo de ejecución.
- Construyen bloques como un componente que eventualmente puede abarcar una gran parte de un sistema.

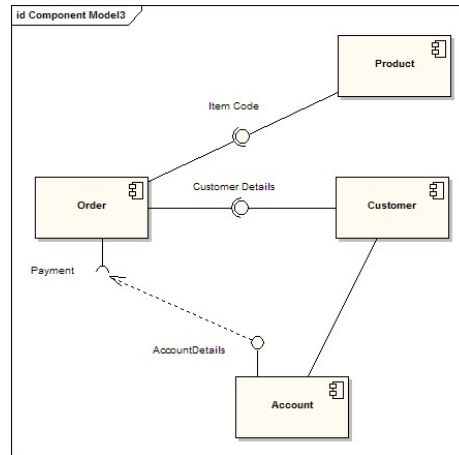
DSI - 3K3 - UML 2.0 (Unidad 1)

30

## UML 2.0 – Diagramas de Estructura

### Diagramas de Componentes

- Ejemplo:



DSI - 3K3 - UML 2.0 (Unidad 1)

31

## UML 2.0 – Diagramas de Estructura

### Diagrama de Despliegue

- Un diagrama de despliegue modela la arquitectura en tiempo de ejecución de un sistema.
- Se muestra la configuración de los elementos de hardware (nodos) y muestra cómo los elementos de software y los artefactos se asignan en los nodos.

DSI - 3K3 - UML 2.0 (Unidad 1)

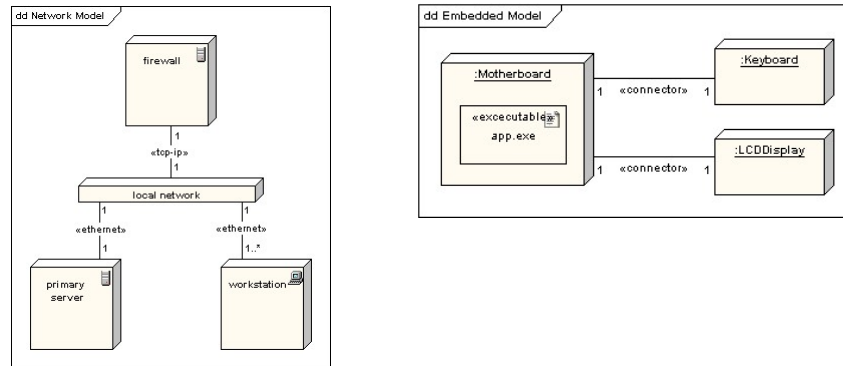
32



## UML 2.0 – Diagramas de Estructura

### Diagrama de Despliegue

- Ejemplo:



DSI - 3K3 - UML 2.0 (Unidad 1)

33

## UML 2.0 – Diagramas de Estructura

### Diagrama de Paquetes

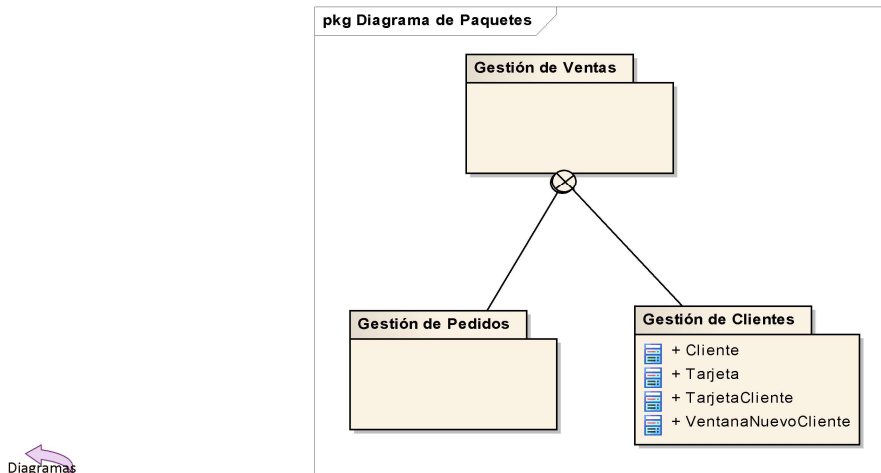
- Los diagramas de paquetes se usan para reflejar la organización de paquetes y sus elementos.
- Los usos más comunes para los diagramas de paquete son para organizar diagramas de casos de uso y diagramas de clase, a pesar de que el uso de los diagramas de paquete no es limitado a estos elementos UML.

DSI - 3K3 - UML 2.0 (Unidad 1)

34

## UML 2.0 – Diagramas de Estructura

### Diagrama de Paquetes



DSI - 3K3 - UML 2.0 (Unidad 1)

35

## UML 2.0 – Diagramas de Comportamiento

### Diagrama de Actividades

- En UML un diagrama de actividades se usa para mostrar la secuencia de actividades.
- Los diagramas de actividades muestran el flujo de trabajo desde el punto de inicio hasta el punto final detallando muchas de las rutas de decisiones que existen en el progreso de eventos contenidos en la actividad.
- También pueden usarse para detallar situaciones donde el proceso paralelo puede ocurrir en la ejecución de algunas actividades.
- Los Diagramas de Actividades son útiles para el Modelado de Negocios donde se usan para detallar el proceso involucrado en las actividades de negocio.

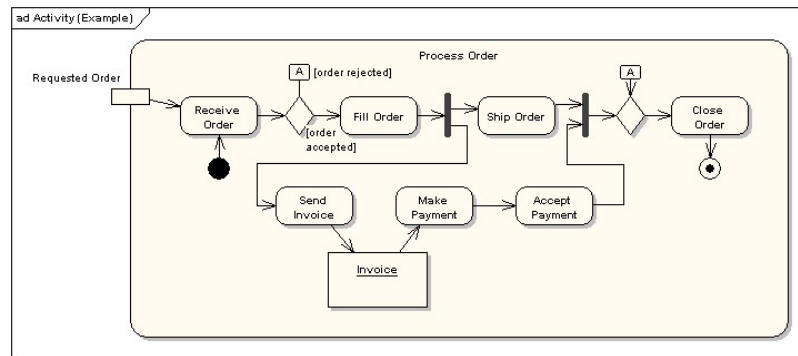
DSI - 3K3 - UML 2.0 (Unidad 1)

36

## UML 2.0 – Diagramas de Comportamiento

### Diagrama de Actividades

- Ejemplo:



DSI - 3K3 - UML 2.0 (Unidad 1)

37

## UML 2.0 – Diagramas de Comportamiento

### Diagrama de Casos de Uso

- El modelo de casos de uso captura los requisitos de un sistema.
- Un diagrama de casos de uso muestra la interacción entre el sistema y las entidades externas al sistema. Estas entidades externas se denominan agentes. Los actores representan los papeles que pueden incluir los usuarios humanos, hardware externo o de otros sistemas.
- Un caso de uso es una sola unidad de trabajo significativo. Proporciona una vista de alto nivel de la conducta observable a alguien o algo fuera del sistema

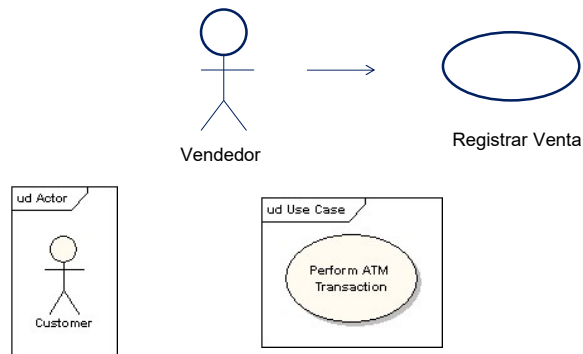
DSI - 3K3 - UML 2.0 (Unidad 1)

38

## UML 2.0 – Diagramas de Comportamiento

### Diagrama de Casos de Uso

- Ejemplo:



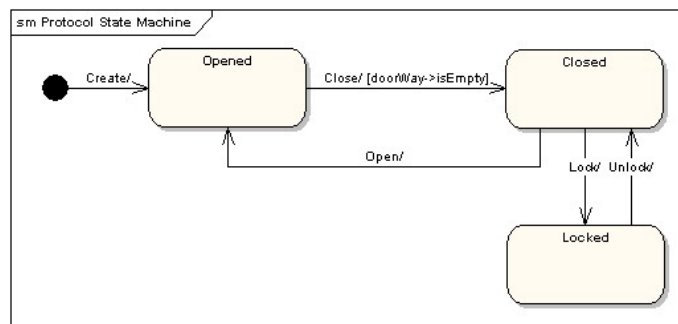
DSI - 3K3 - UML 2.0 (Unidad 1)

39

## UML 2.0 – Diagramas de Comportamiento

### Diagrama de Máquina de Estados

- Un Diagrama de Máquina de Estados modela el comportamiento de un objeto especificando la secuencia de situaciones que un objeto atraviesa durante su ciclo de vida en respuesta a eventos.



DSI - 3K3 - UML 2.0 (Unidad 1)

40

## UML 2.0 – Diagramas de Comportamiento

### Diagrama de Visión de Interacciones

- Un diagrama de descripción de la interacción es una forma de diagrama de actividad en el cual los nodos representan diagramas de interacción.
- Es un diagrama que muestra cómo interactúan varios diagramas de interacciones.
- Los diagramas de interacción pueden incluir diagramas de secuencia, comunicación, de descripción de la interacción y de tiempos.
- Es muy útil para mostrar de qué manera distintos escenarios se combinan.

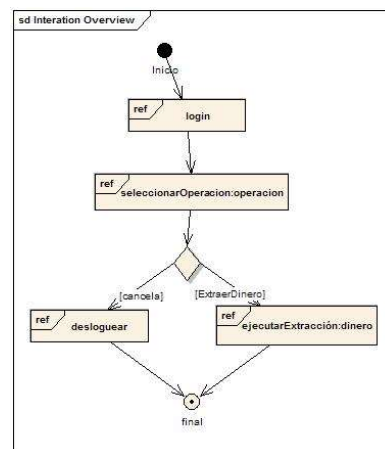
DSI - 3K3 - UML 2.0 (Unidad 1)

41

## UML 2.0 – Diagramas de Comportamiento

### Diagrama de Visión de Interacciones

- Ejemplo:
- La mayoría de la notación es la misma que para los diagramas de actividad: los nodos inicial, final, decisión, combinación, bifurcación y unión.
- Sin embargo, introducen dos elementos nuevos: ocurrencias de interacción y elementos de interacción.



DSI - 3K3 - UML 2.0 (Unidad 1)

42

## UML 2.0 – Diagramas de Comportamiento

### Diagrama de Comunicación

- Un Diagrama de Comunicación (llamado Diagrama de Colaboración en las versiones anteriores de UML) es un diagrama de interacción que muestra información similar al diagrama de secuencia pero se enfoca principalmente en las relaciones entre los objetos.
- En los diagramas de comunicación los objetos se muestran con conectores de asociación entre ellos (enlaces). Los mensajes se agregan a los enlaces y se muestran como una flecha corta en la dirección en que fluye el mensaje. La secuencia de los mensajes se muestra con un esquema numérico.

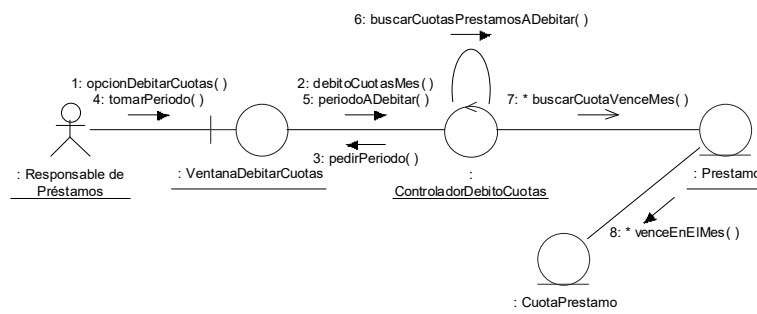
DSI - 3K3 - UML 2.0 (Unidad 1)

43

## UML 2.0 – Diagramas de Comportamiento

### Diagrama de Comunicación

- Ejemplo:



DSI - 3K3 - UML 2.0 (Unidad 1)

44

## UML 2.0 – Diagramas de Comportamiento

### Diagrama de Secuencia

- Un Diagrama de Secuencia es una forma de diagrama de interacción que muestra objetos como líneas de vida hacia abajo en la página con sus interacciones en el tiempo representadas como flechas desde la línea de vida origen a la línea de vida destino.
- Los diagramas de secuencia son buenos para mostrar cuáles objetos se comunican con otros objetos y qué mensajes disparan esas comunicaciones.
- Los diagramas de secuencia no intentan mostrar lógica de procedimiento compleja.

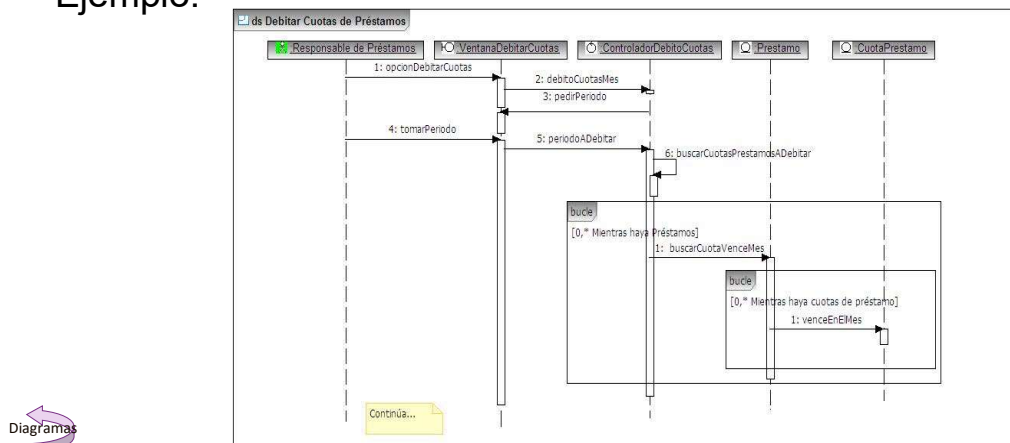
DSI - 3K3 - UML 2.0 (Unidad 1)

45

## UML 2.0 – Diagramas de Comportamiento

### Diagrama de Secuencia

- Ejemplo:



DSI - 3K3 - UML 2.0 (Unidad 1)

46

## UML 2.0 – Diagramas de Comportamiento

### Diagrama de Tiempo

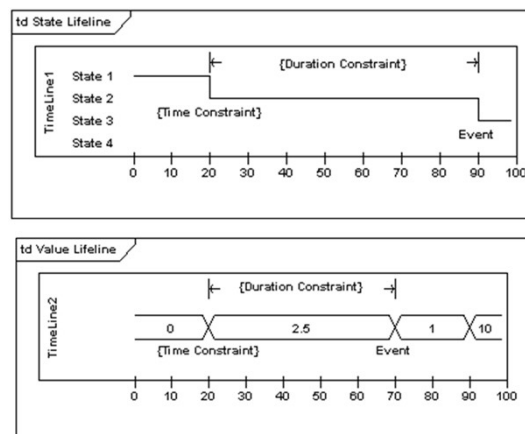
- El propósito primario de los diagramas de tiempos (o temporizados) es mostrar los cambios en el estado, o la condición, de una línea de vida de una instancia (de un Clasificador o un Rol de un clasificador), a lo largo del tiempo y de manera lineal.
- El uso más común es mostrar el cambio de estado de un objeto a lo largo del tiempo, en respuesta a los eventos o estímulos aceptados.

DSI - 3K3 - UML 2.0 (Unidad 1)

47

## UML 2.0 – Diagramas de Comportamiento

### Diagrama de Tiempos



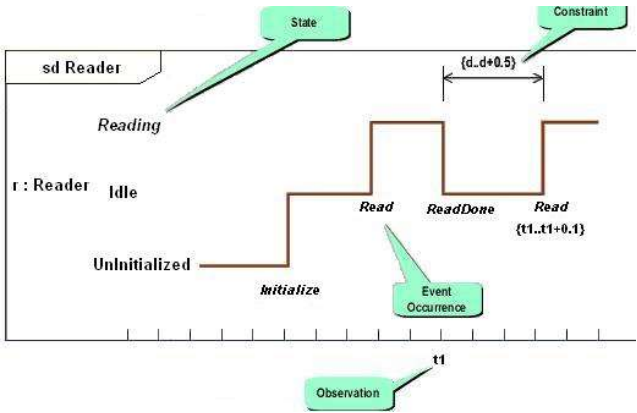
DSI - 3K3 - UML 2.0 (Unidad 1)

48



# UML 2.0 – Diagramas de Comportamiento

## Diagrama de Tiempos



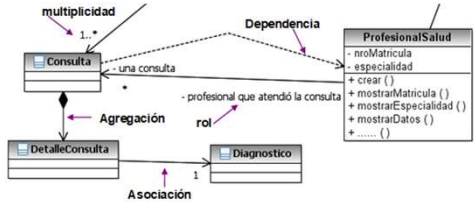
Diagramas

# UML 2.0

## Mecanismos Comunes



- **Especificaciones:** Detrás de cada elemento de notación gráfica hay una especificación que proporciona una explicación textual de la sintaxis y semántica de ese bloque de construcción.
- **Adornos:** Todos los elementos en la notación UML comienzan con un símbolo básico al cual pueden añadirse una variedad de adornos específicos de ese símbolo.
- **Divisiones comunes:**
  - Clasificador/Instancia (Ej.: Clase/Objeto)
  - Interfaz/Implementación (Ej.: Caso de Uso / Colaboración, Operación/Método).



Estructura

## UML 2.0

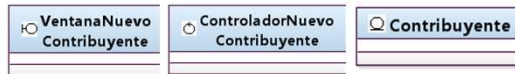
### Mecanismos Comunes



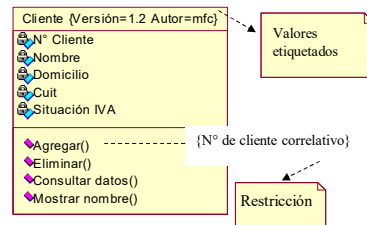
DSI - 3K3 - UML 2.0 (Unidad 1)

- **Mecanismos de extensibilidad:** UML proporciona un lenguaje estándar pero puede no ser suficiente para expresar todos los matices posibles de todos los modelos en todos los dominios. Por eso es posible extender el lenguaje de manera controlada mediante los siguientes mecanismos de extensión:

- Estereotipos. Por ej. estereotipos para clases de interfaz, control, entidad:



- Valores etiquetados
- Restricciones



51

## Reglas de UML



DSI - 3K3 - UML 2.0 (Unidad 1)

### UML tiene reglas semánticas para:

- **Nombres:** Cómo llamar a los elementos, relaciones y diagramas.
- **Alcance:** El contexto que da un significado específico a un nombre.
- **Visibilidad:** Cómo se pueden ver y utilizar esos nombres por otros.
- **Integridad:** Cómo se relacionan apropiada y consistentemente unos elementos con otros.
- **Ejecución:** Qué significa ejecutar o simular un modelo dinámico.

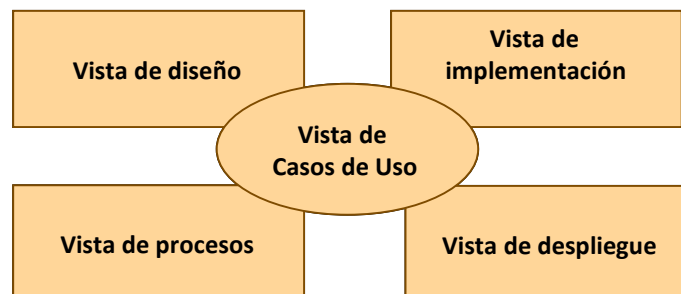
52

## UML 2.0

- Arquitectura del sistema: Entendemos por **arquitectura**:
  - La organización de un sistema de software.
  - La selección de elementos estructurales y sus interfaces.
  - Su comportamiento (colaboraciones entre esos elementos).
  - La composición de los elementos estructurales y de comportamiento en subsistemas cada vez más grandes.
  - El estilo arquitectónico que guía esta organización (los elementos estáticos y dinámicos y sus interfaces, colaboraciones y su composición).

## UML 2.0

- Vistas de un sistema con UML



# UML 2.0

## Vistas de un sistema con UML

Vista de casos de uso	Describe el comportamiento del sistema a través de los casos de uso.	•Aspecto estático: Diagrama de casos de uso. •Aspecto dinámico: Diagramas de interacción, de estados y de actividades.
Vista de diseño	Comprende las clases, interfaces y colaboraciones que conforman el vocabulario del problema y la solución.	•Aspecto estático: Diagramas de clases y objetos. •Aspecto dinámico: Diagramas de interacción, de estados y de actividades.
Vista de procesos	Comprende los hilos y procesos que forman los mecanismos de sincronización y concurrencia del sistema	Los aspectos <b>estáticos y dinámicos</b> de esta vista se capturan con los mismos diagramas que la vista de diseño pero poniendo énfasis en las clases activas.

DSI - 3K3 - UML 2.0 (Unidad 1)

55

# UML 2.0

## Vistas de un sistema con UML

Vista de Implementación	Comprende los componentes y archivos que se utilizan para ensamblar y hacer disponible el sistema físico.	•Aspecto estático: Diagramas de componentes. •Aspecto dinámico: Diagramas de interacción, de estados y de actividades.
Vista de despliegue	Contiene los nodos que conforman la topología de hardware sobre la que se ejecuta el sistema.	•Aspecto estático: Diagramas de despliegue •Aspecto dinámico: Diagramas de interacción, de estados y de actividades.

DSI - 3K3 - UML 2.0 (Unidad 1)

56

## Bibliografía

- Booch Grady, Rumbaugh James, Jacobson Ivar, (2006), “El lenguaje de Modelado Unificado – 2da. Edición”, España, Editorial Addison Wesley – Pearson Educación SA.
- Arlow Jim, Neustadt Ila, “Programación UML 2”, (2006), Ediciones Anaya Multimedia
- Tutorial UML 2.0 Sparx Systems