

UML

UML

- Conceptos básicos

- **Objetos:** es una unidad atómica que posee un estado y un comportamiento. El estado está determinado por el valor que toman los atributos del objeto en un instante dado. Un objeto representa una entidad física, o bien una entidad abstracta,
- **Abstracción:** Se ignoran detalles para centrarse en características esenciales. Permite simplificar la información acerca del objeto. Consiste en la extracción de las propiedades fundamentales de un concepto. No preocuparse por los detalles no esenciales, es decir, no es necesario conocer los detalles exactos de la implementación
- **Encapsulamiento:** Los datos están “escondidos” dentro del objeto. Permite determinar qué información es publicada y qué información es ocultada al resto de los objetos. El encapsulamiento permite reducir la complejidad, ya que las Clases se tratarán como cajas negras donde sólo se conoce el comportamiento pero no los detalles internos.

UML

- **Asociación:** Forma de interacción entre objetos. Dos objetos están asociados cuando uno “usa” servicios u operaciones brindados por el otro.
.
- **Agregación:** Es un tipo particular de asociación. Describe una relación “tiene un...”. Se refiere al concepto de definir un objeto en términos de sus componentes.
- **Composición:** Es un tipo particular de asociación. Describe una relación “contiene...” Indica que un objeto no puede existir sin contener a otro.

UML

- **Cohesión:** Es el grado de relación entre clases en relación a un propósito dentro del sistema
- **Acoplamiento:** Indica el nivel de dependencia entre objetos.
 - Es deseable tener bajo acoplamiento y alta cohesión
- **Herencia** Permite definir una nueva clase en términos de otra clase ya existente. Describe una relación “es un ...”
- **Polimorfismo:** Mecanismo basado en herencia. Permite redefinir una misma operación para dos o más clases relacionadas mediante herencia. El comportamiento de una operación polimórfica depende del objeto en concreto sobre la cual es aplicada.

UML

- Definición
- El Unified Modeling Language es un lenguaje visual para especificar, construir y documentar los artefactos de los sistemas. Es un lenguaje de modelado de propósito general que se puede utilizar con todos los métodos de objetos y componentes, y que se puede utilizar en todos los dominios de aplicación (por ejemplo, salud, finanzas, telecomunicaciones, aeroespacial) y las plataformas de ejecución (por ejemplo, J2EE,. NET).
- UML infrastructure Specification v2,4

UML

- Versión actual es la 2.5 (Junio 2015)
- la presente definición se encuentra en www.omg.org
- gráficos y textos basados en tutorial
 - http://www.sparxsystems.com/resources/uml2_tutorial/

UML

- Algunos cambios menores se resolvieron por la RTF (Revision Task Force)
- Debido a la necesidad de cambios más importantes la OMG definió dividir en 4 partes la definición de UML
 - UML Infraestructure
 - UML Superstructure
 - UML Object Constraint Language
 - UML Diagram Interchange

UML

- UML define varios tipos de diagramas que se utilizan para definir diferentes aspectos
- Algunos diagramas se centran en el aspecto estático del modelo, otros en el aspecto dinámico

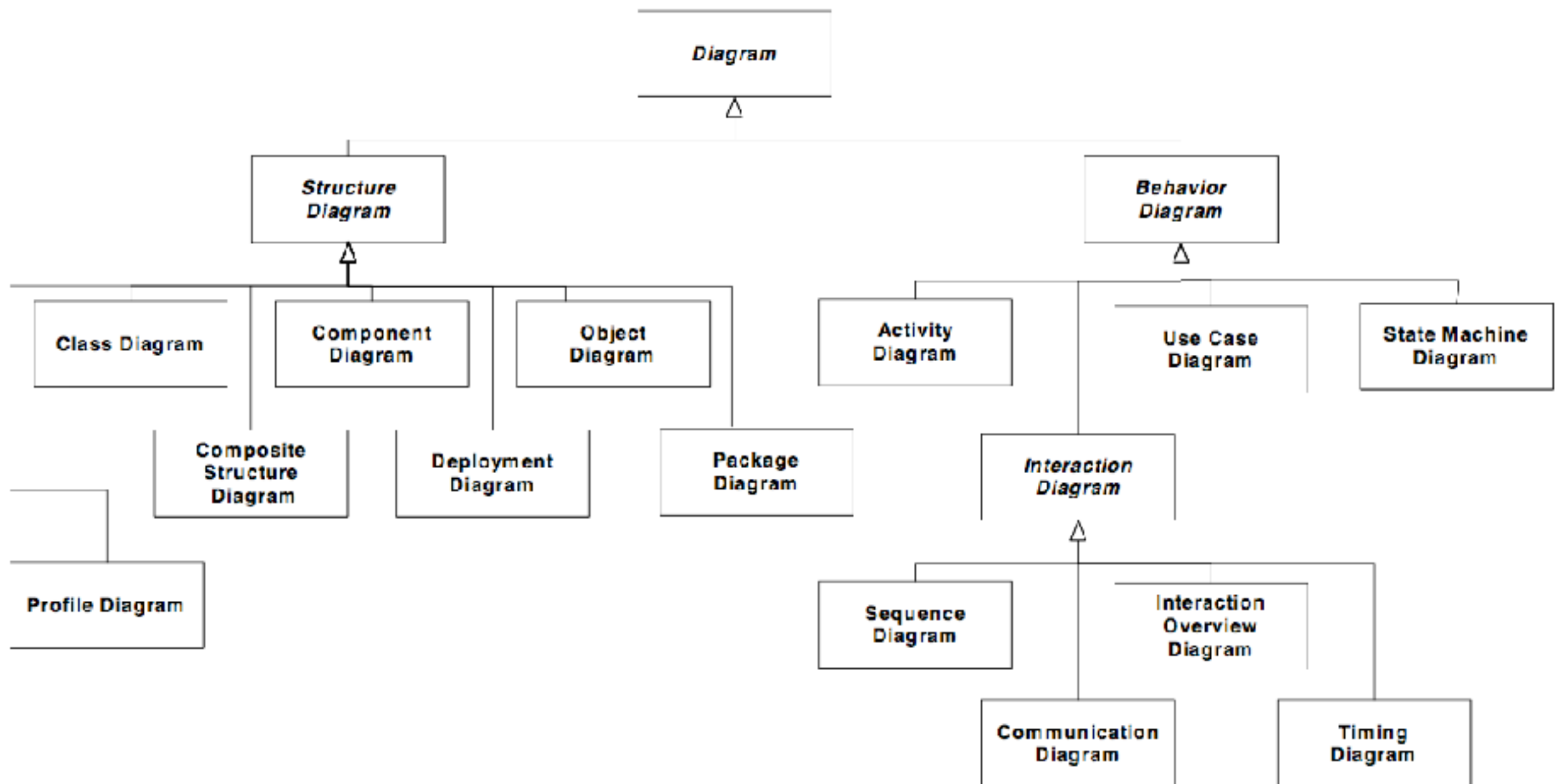
UML

- Diagramas Estáticos (estructurales)
 - Diagrama de clases
 - Diagrama de objetos
 - Diagrama de componentes
 - Diagrama de distribución/despliegue
 - Diagrama de paquetes
 - Diagrama de estructuras

UML

- Diagramas de comportamiento
 - Diagrama de caso de uso
 - Diagrama de secuencia
 - Diagrama de colaboración
 - Diagrama de estado
 - Diagrama de actividad
 - Diagrama de tiempos
 - Diagrama de interacción

UML

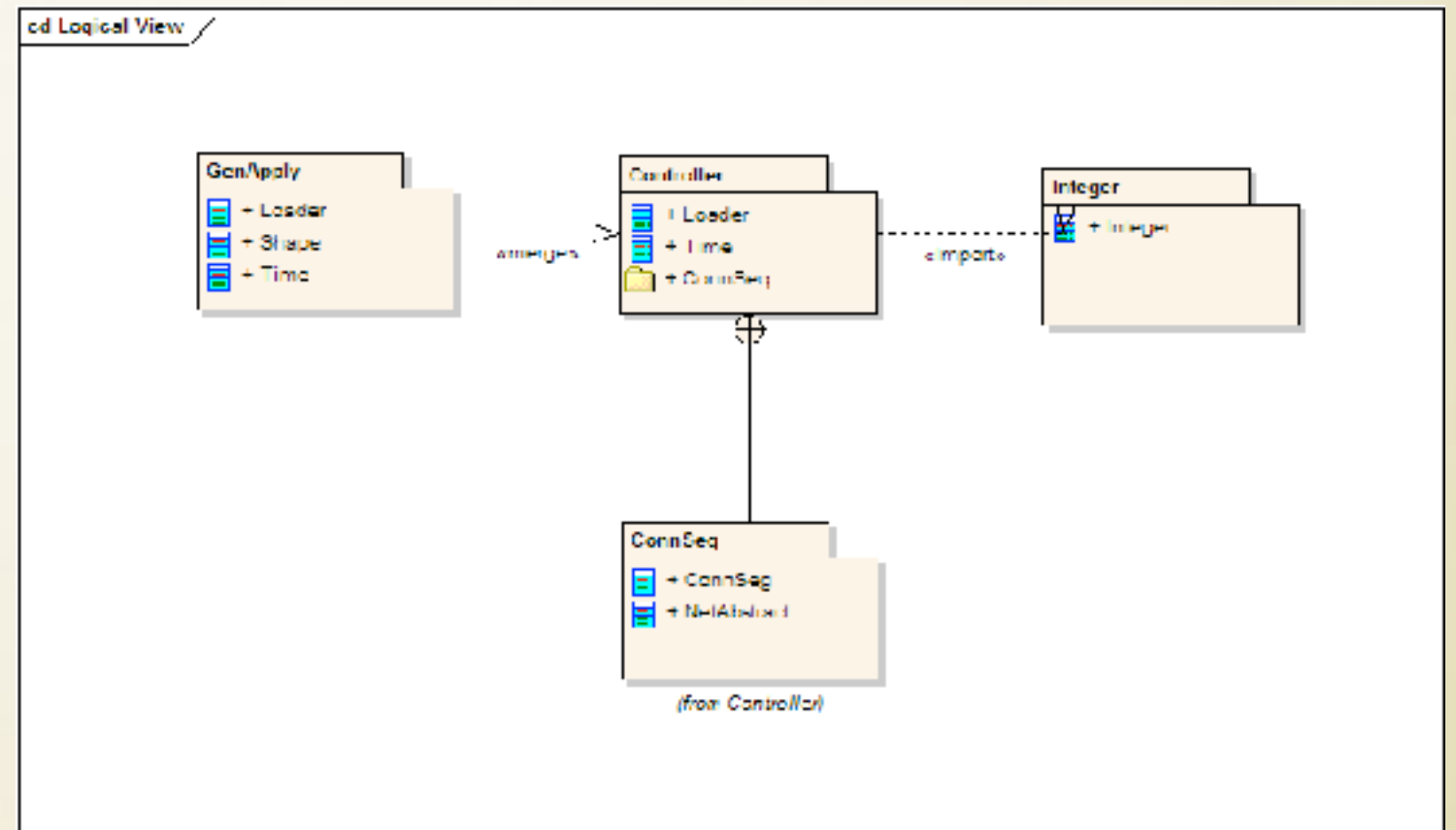


UML

DIAGRAMAS DE ESTRUCTURAS

UML

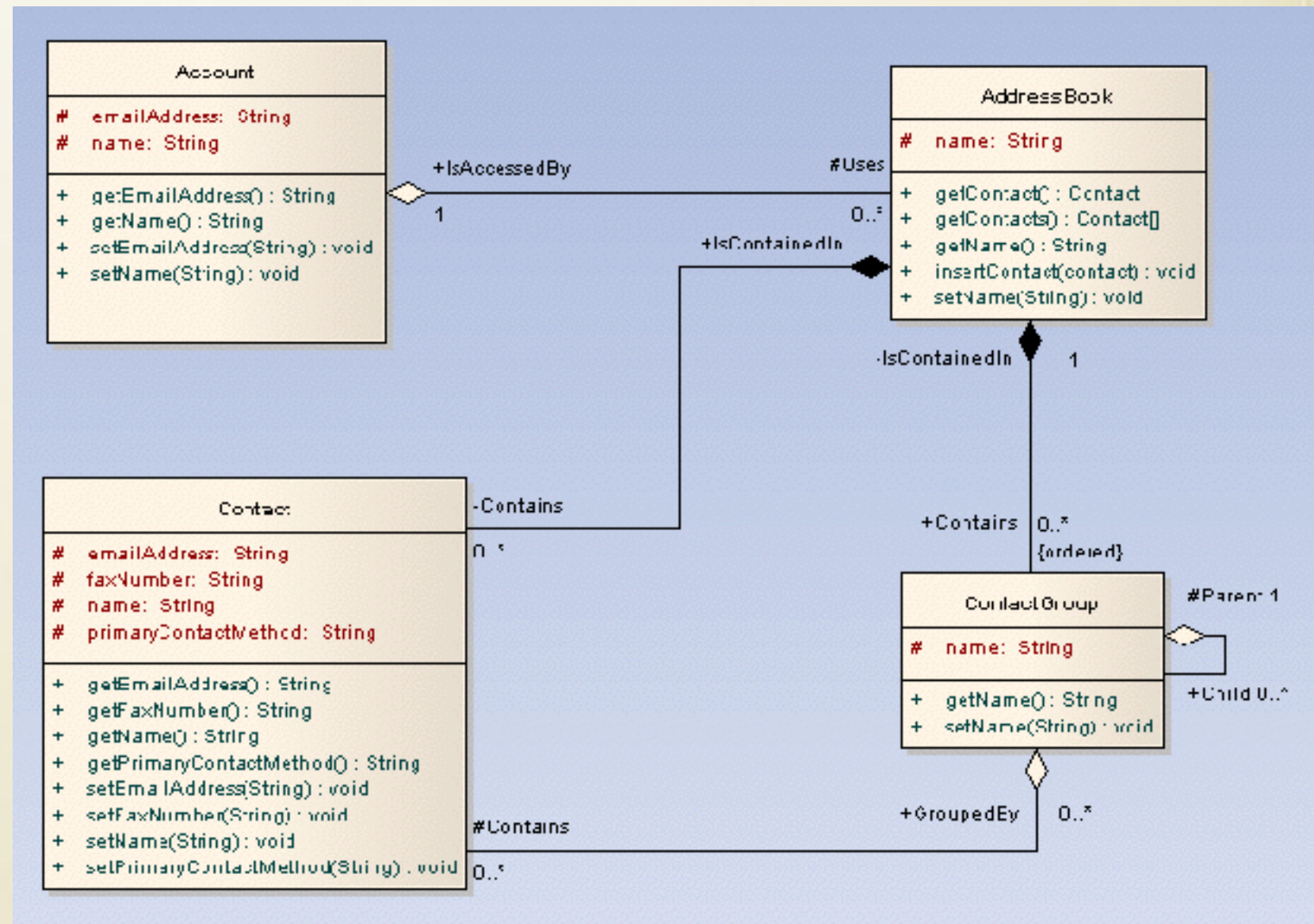
- Diagrama de paquetes
 - Divide el modelo en contenedores lógicos



El uso más común es para agrupar diagrama de clases o de casos de uso

UML

- Diagrama de clases
 - Define en bloques básicos el modelo



UML

- **Clase**

- Una clase describe un conjunto de objetos que tienen en común los mismos atributos, operaciones, relaciones y semántica.
Una clase normalmente incluye su nombre, atributos y operaciones (métodos).
- Los atributos identifican datos de los objetos de la clase y los métodos especifican sus operaciones o comportamiento.
- Las Instancias, son los objetos que componen cada clase. Una vez que una instancia es creada su comportamiento es igual al comportamiento de las otras instancias. A éstas, se le aplican un conjunto de operaciones.
Muchos de los atributos y métodos se conocen recién en la fase de diseño. No obstante, hay otros que se evidencian en la fase de análisis y deben incluirse en el diagrama en cuanto se detectan.

UML

- Tipo de clases
 - Clase abstracta: son clases que no crean o producen instancias de si mismas
 - Clase concreta: de aquella que pueden generarse instancias
 - Superclase: clase donde se heredan comportamiento o estructuras
 - Subclase: clase que hereda comportamiento

UML

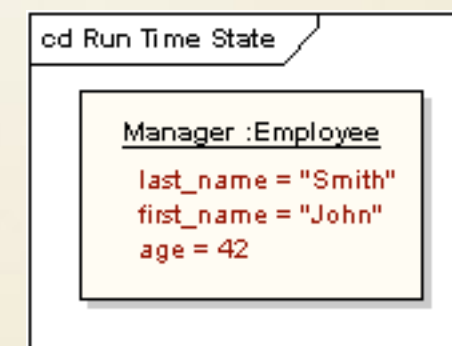
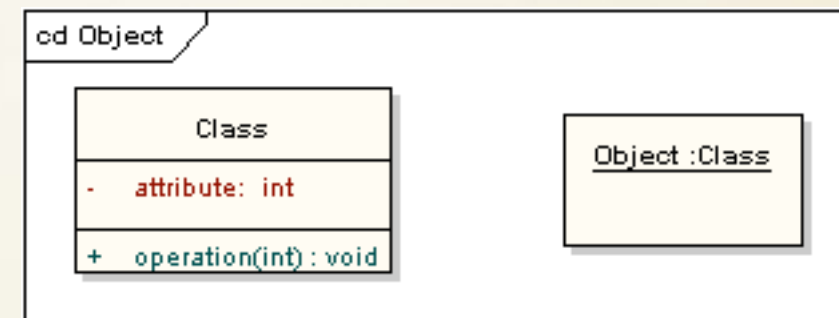
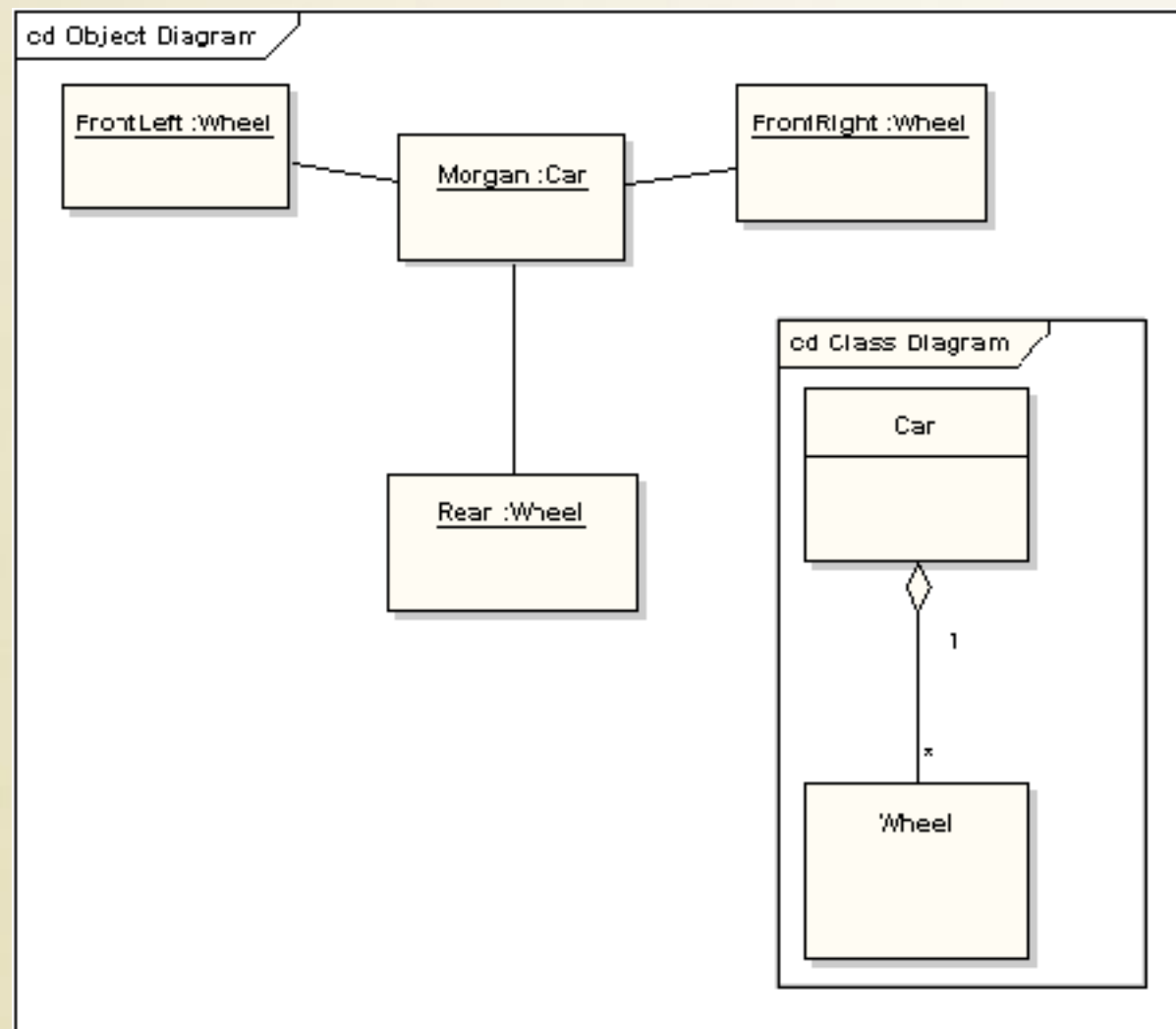
- Interfaz: es el conjunto de operaciones que especifican un servicio de una clase pudiendo representar el comportamiento total o parcial de la clase
- Responsabilidades: las responsabilidades de un objeto son todos los servicios que provee
- incluye
 - el conocimiento que el objeto tiene
 - las acciones que un objeto puede realizar

UML

- Tipo de diagrama de clases
 - Diagrama de clases conceptual: se construyen en la etapa de análisis, en esto no se incluyen los métodos y los atributos.
 - Diagrama de implementación: se construyen en la etapa de diseño, se agrega información mas específica, incluyen los atributos y métodos

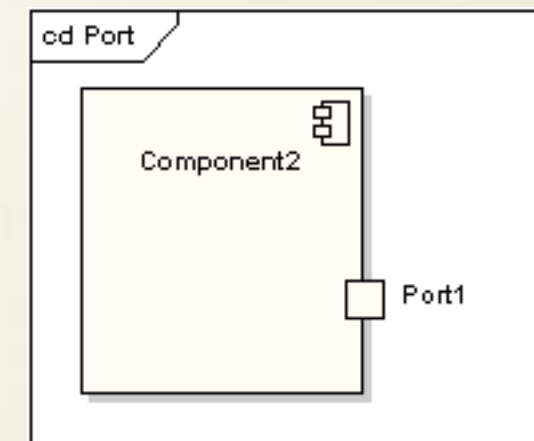
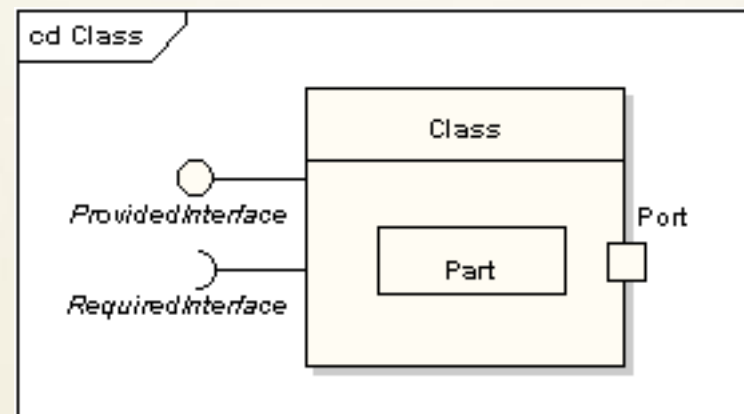
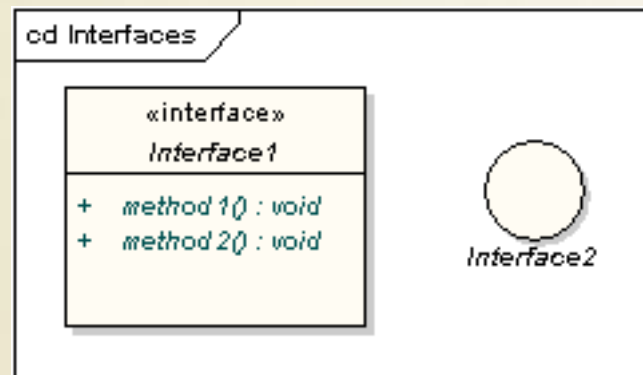
UML

- Diagrama de objetos

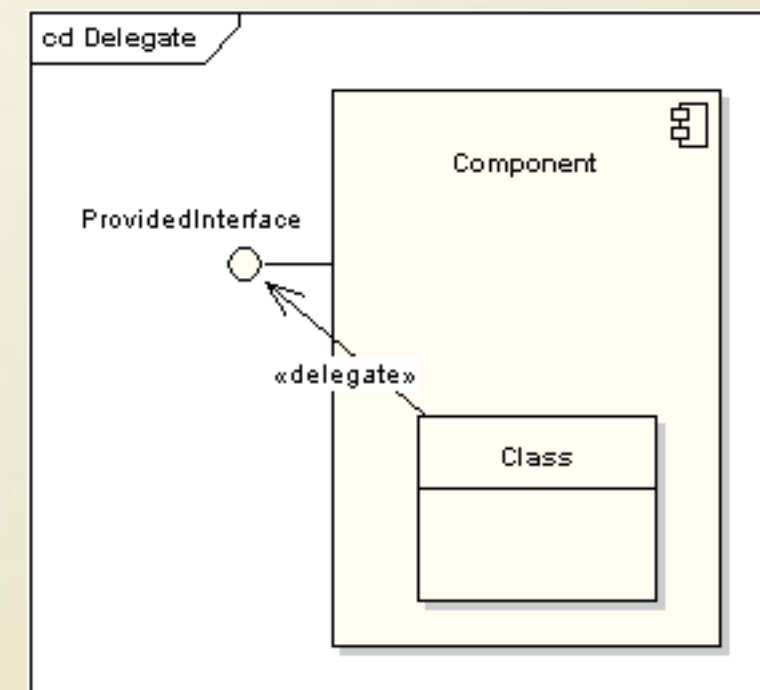
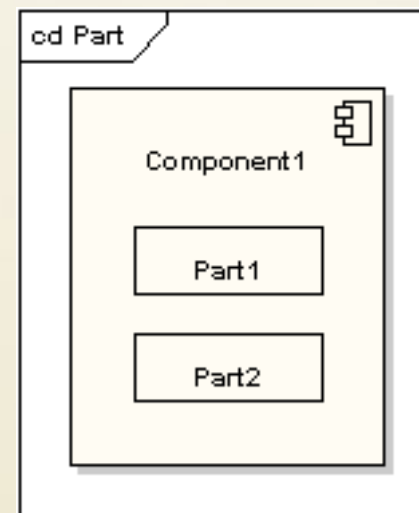


- Muestra como las instancias de los elementos estructurales se relacionan y usan en tiempo de ejecución

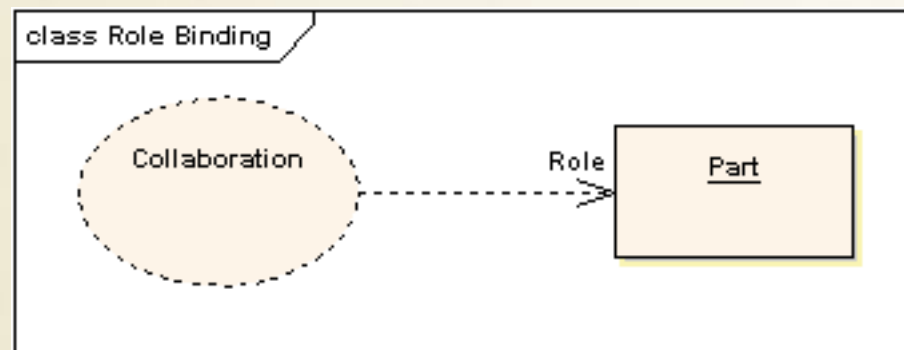
UML



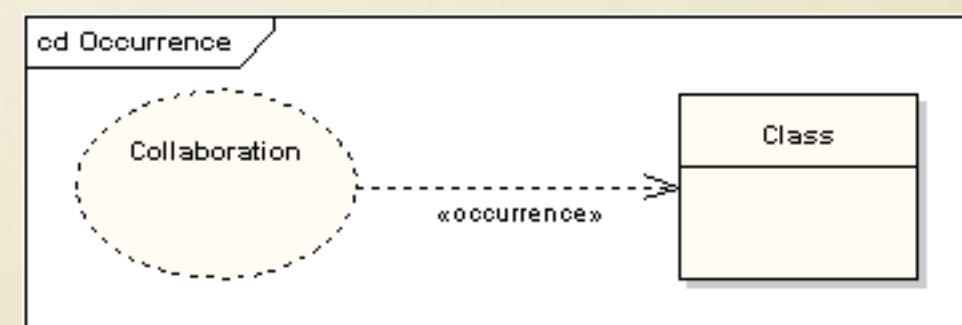
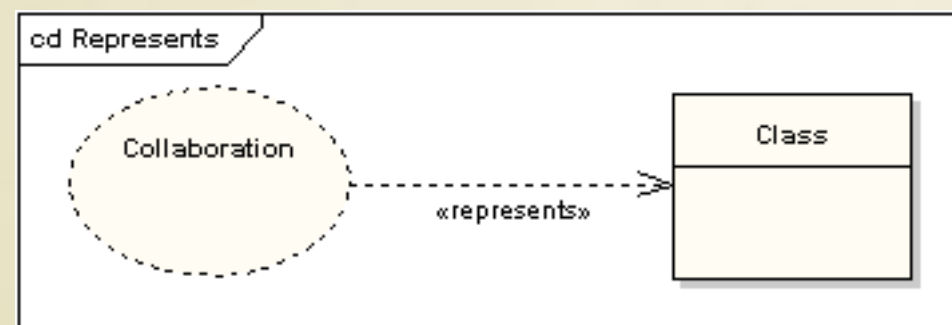
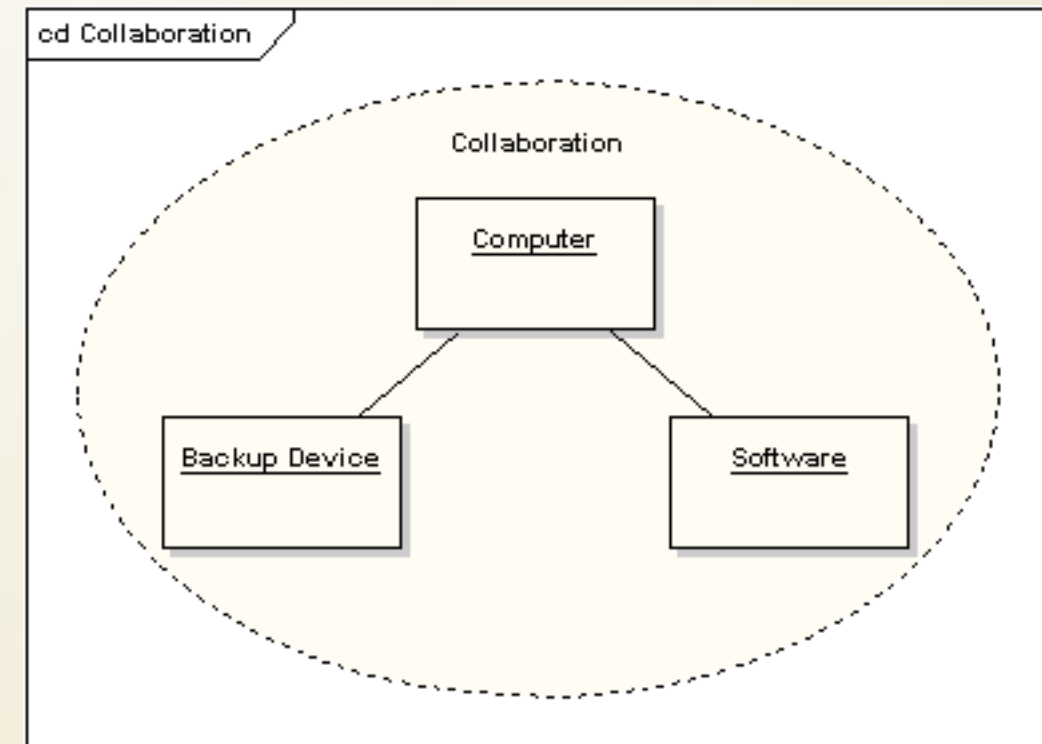
- Diagrama de composición de estructuras
 - provee una manera de separar en capas los elementos y hacer foco en los detalles de construcción y de relaciones



UML

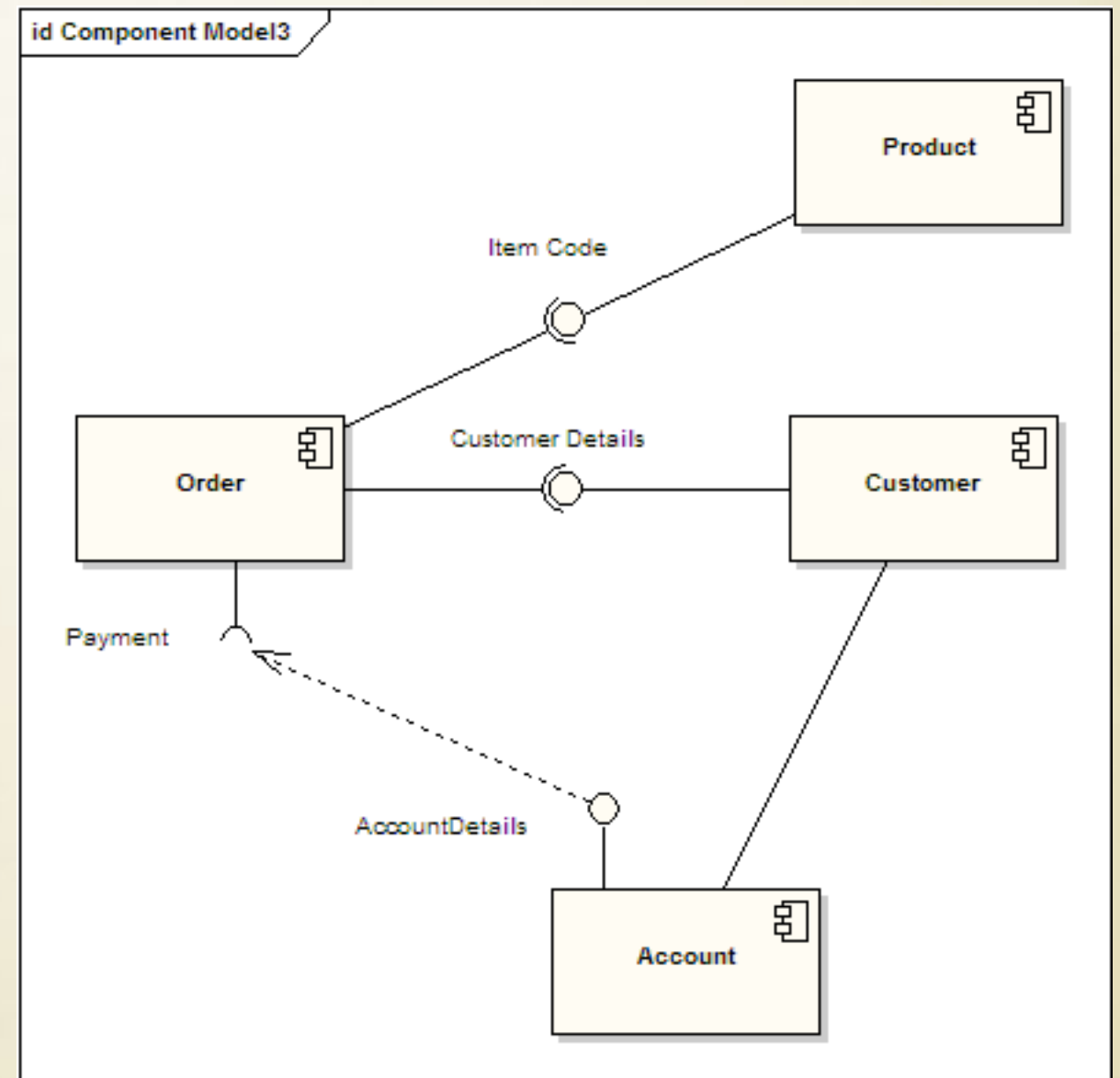


- Diagrama de composición de estructuras



UML

Diagrama de componentes
Se utiliza para modelar
estructuras complejas o de
alto nivel



UML

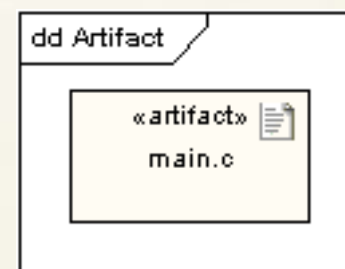
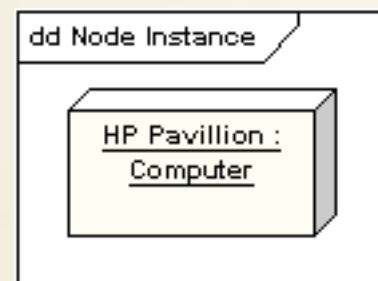
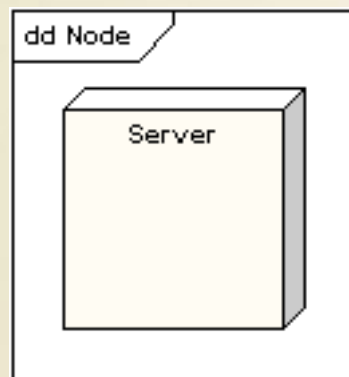
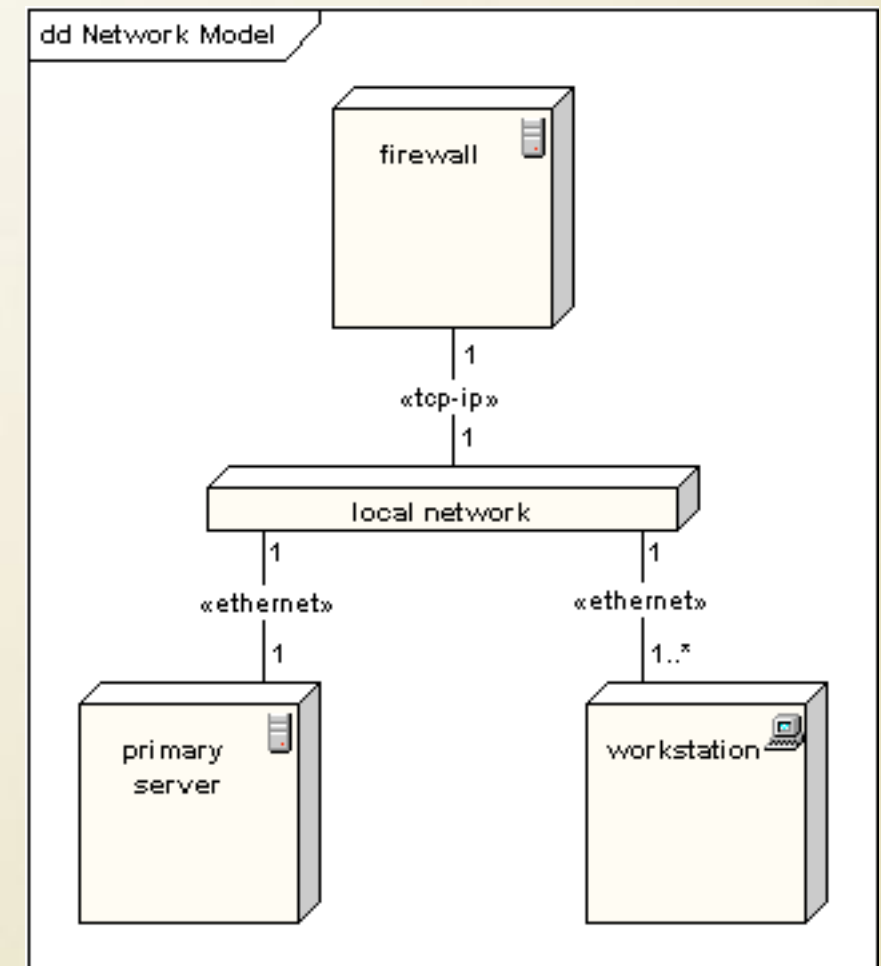
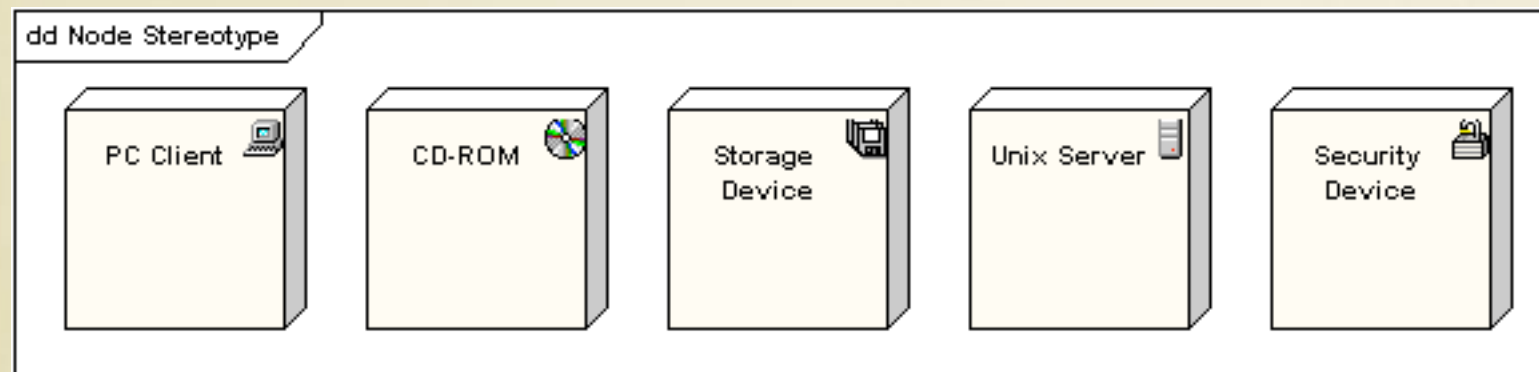


Diagrama de Despliegue
Muestra la disposición física

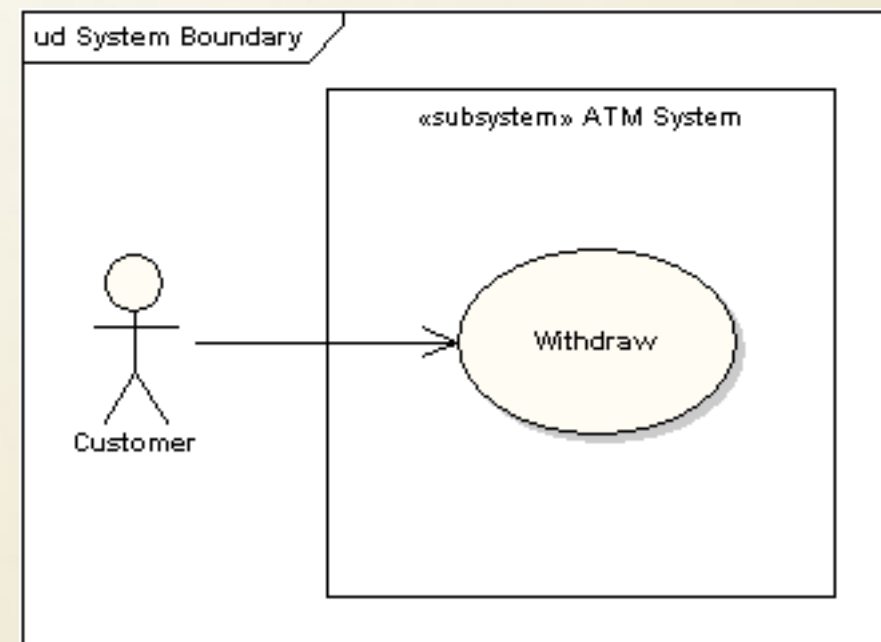


UML

DIAGRAMAS DE COMPORTAMIENTO

UML

Caso de Uso
modela la interacción
define el comportamiento,
requerimiento y restricciones
en forma de escenarios



UML

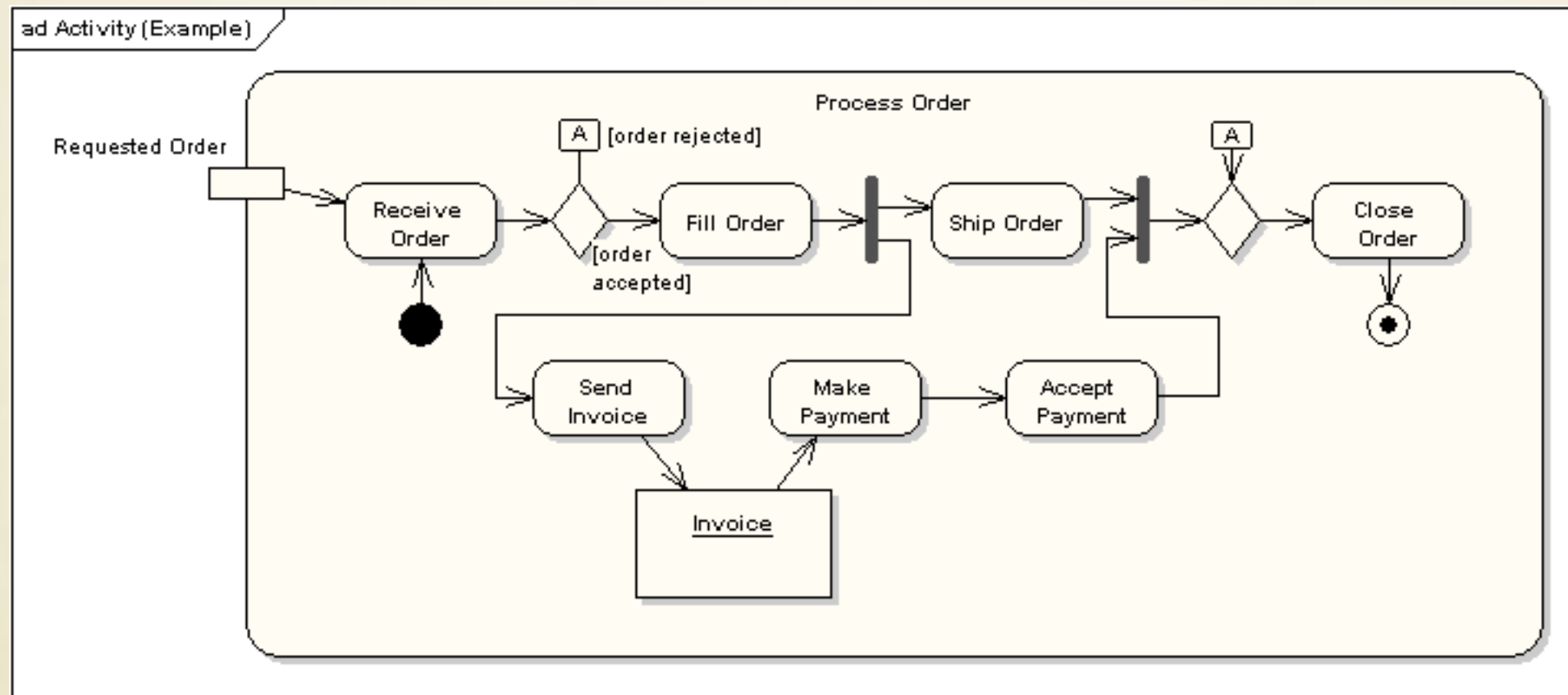


Diagrama de actividad
define el flujo del programa hasta la
captura de los puntos de decisión y acción

UML

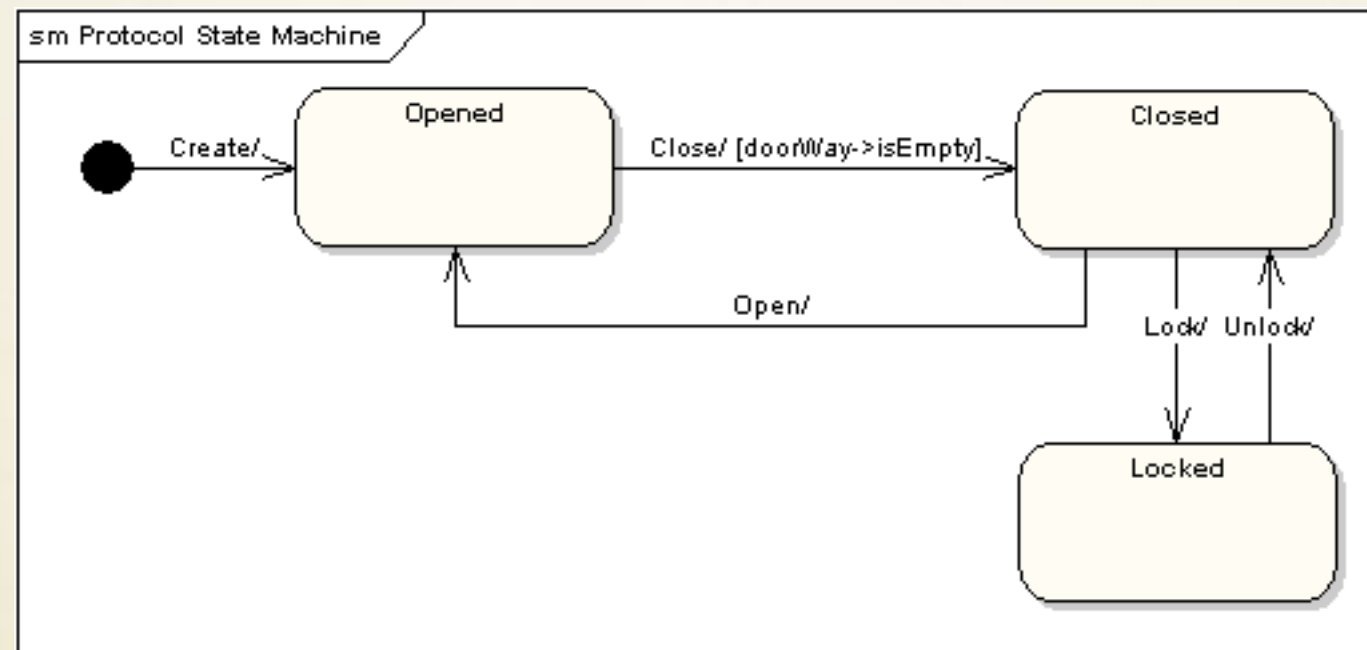


Diagrama de estados

Para definir instante a instante el estado del modelo
cuando se ejecuta

UML

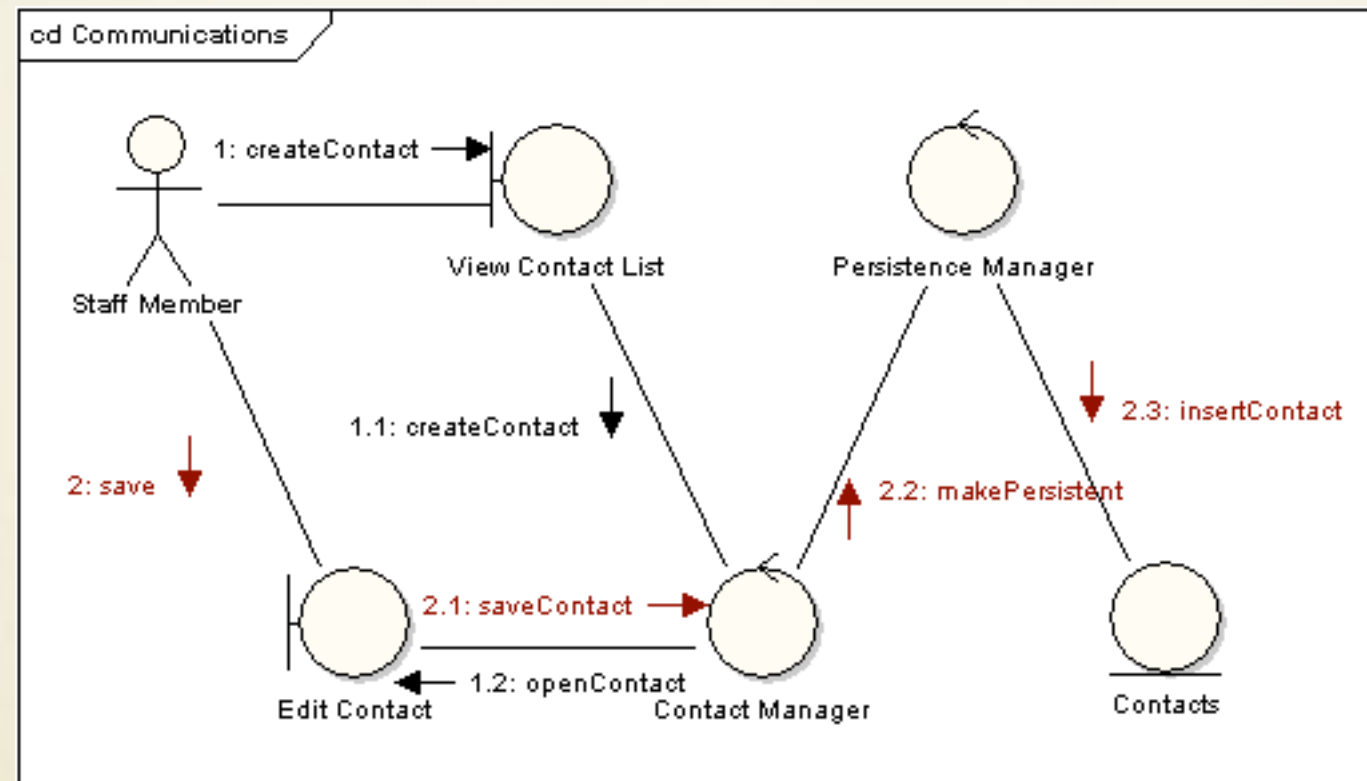
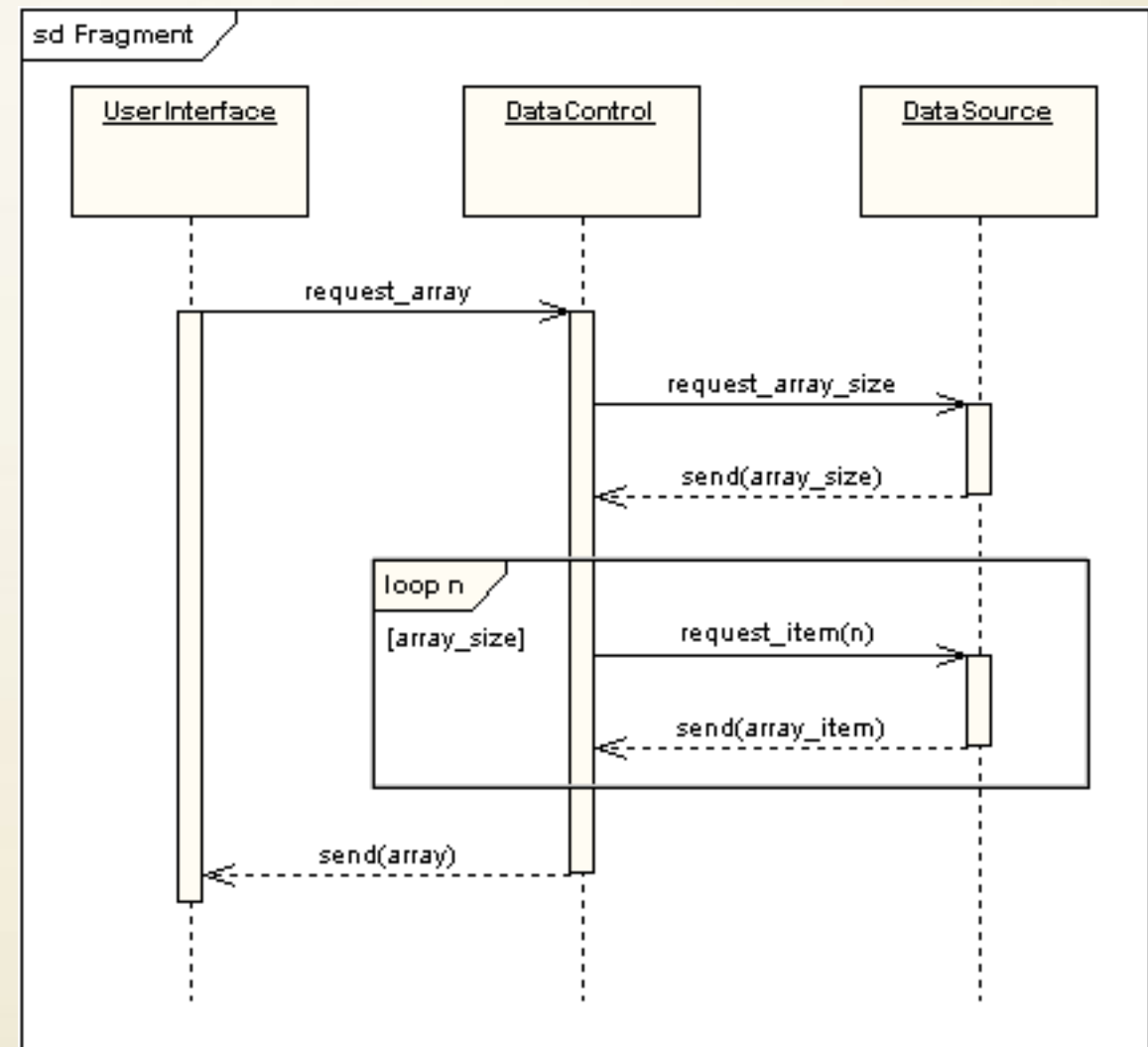


Diagrama de colaboración o comunicación

Muestra la red y la secuencia de los mensajes o comunicación entre los objetos durante la instancia de colaboración

UML

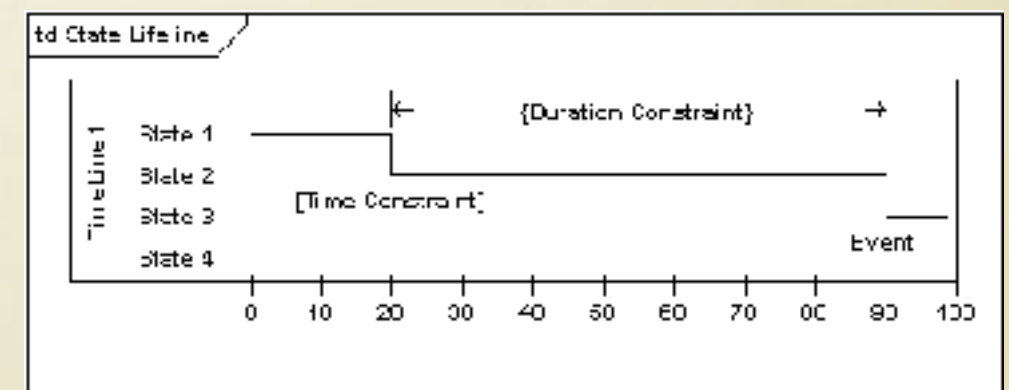
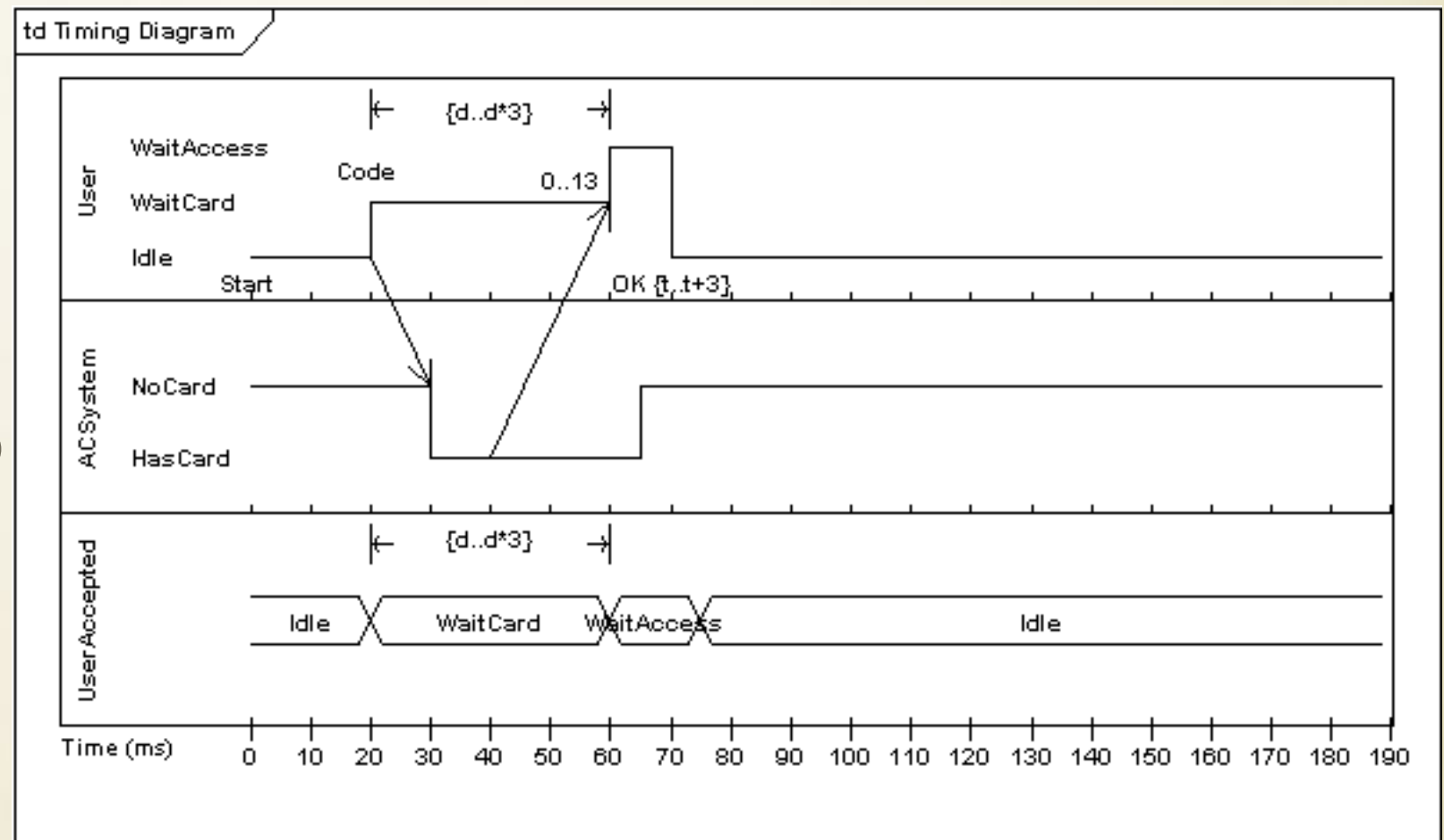
Diagrama de secuencia
Muestra la secuencia de
mensajes que se
intercambian los objetos
usando una linea de
tiempo vertical



UML

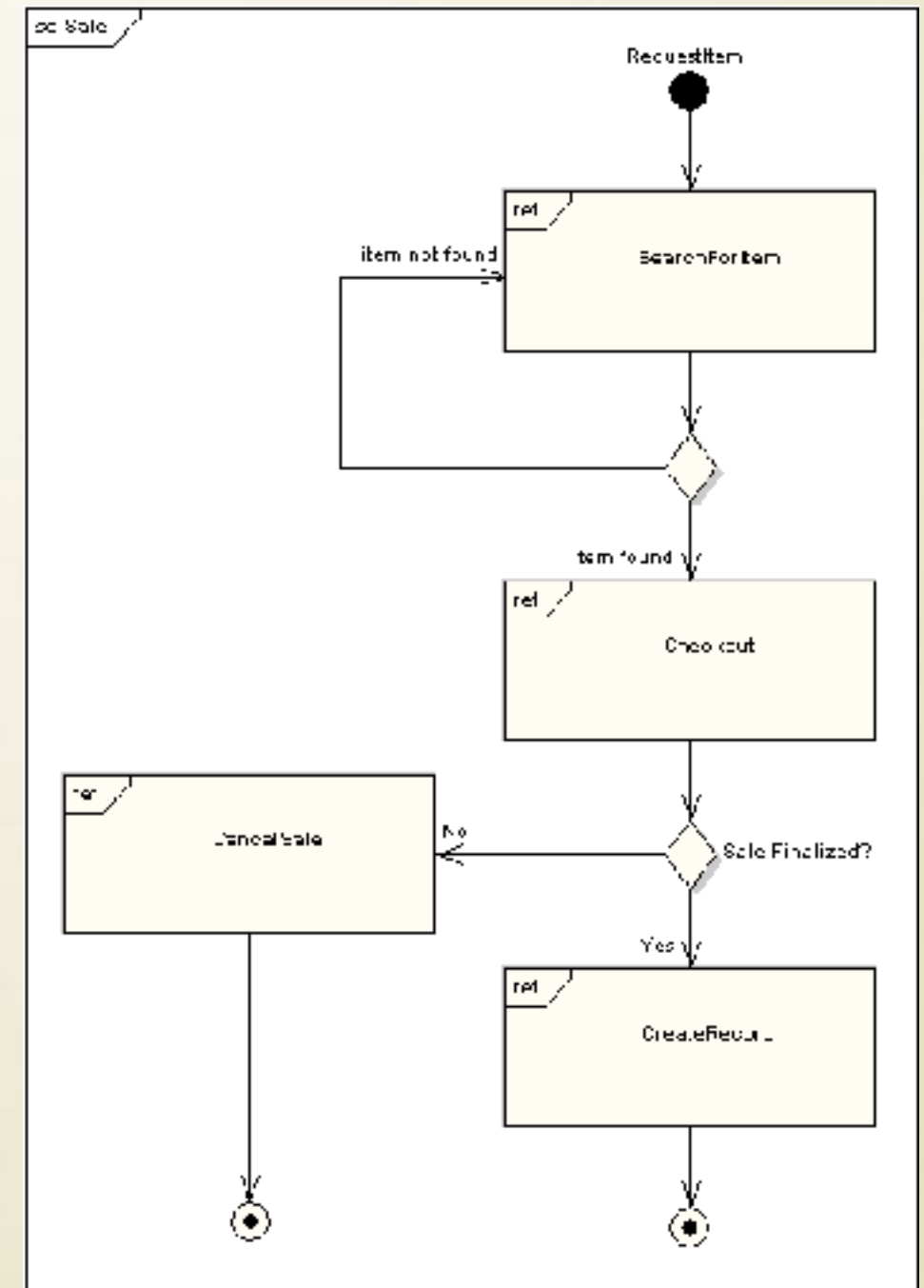
Diagrama de tiempo
es una combinación
de diagrama de
secuencia y estado.

proporciona una vista de cómo
cambian los objetos en el tiempo
o por los mensajes



UML

Diagrama de interacción
combina los diagramas de
actividad y secuencia



Preguntas?