

Diagramas de Clases

- ✓ Un diagrama de clases o estructura estática muestra el conjunto de clases y objeto importantes que forman parte de un sistema, junto con las relaciones existentes entre clases y objetos.
- ✓ Muestra de una manera estática la estructura de información del sistema y la visibilidad que tiene cada una de las clases, dada por sus relaciones con los demás en el modelo

Elementos Diagramas de Clases

Clase:

- Representa un conjunto de entidades que tienen propiedades comunes.
- Una clase es un constructor que define la estructura y comportamiento de una colección de objeto denominados instancia de la clase.
- En UML la clase está representada por un rectángulo con tres divisiones internas, son los elementos fundamentales del diagrama.

Elementos Diagramas de Clases

Ejemplo:

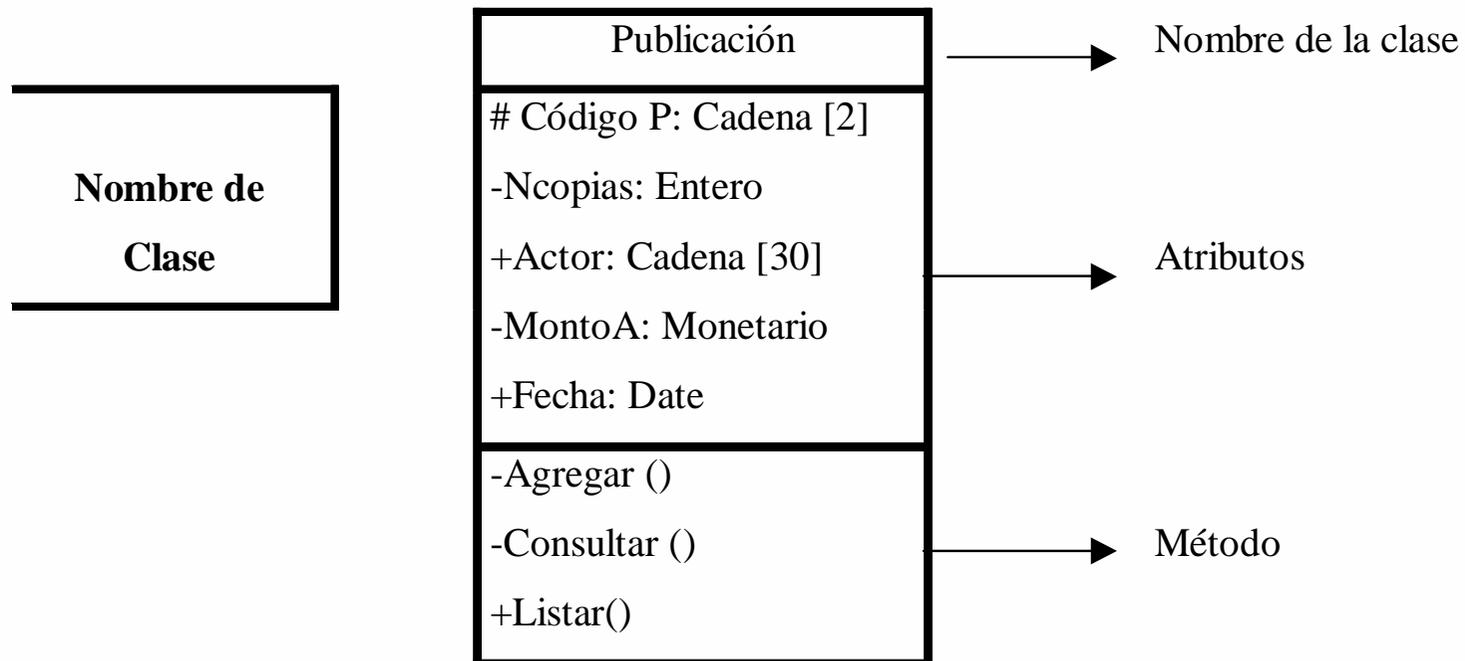


Fig.2 Representación de una clase

Elementos Diagramas de Clases

Atributo:

- Representa una propiedad de una entidad. Cada atributo de un objeto tiene un valor que pertenece a un dominio de valores determinado.
- Las sintaxis de una atributo es:
- *Visibilidad <nombre>: tipo = valor inicial { propiedades}*
- Donde visibilidad es uno de los siguientes:
 - + público.
 - # protegido.
 - - privado.

Elementos Diagramas de Clases

Operación:

- El conjunto de operaciones que describen el comportamiento de los objetos de una clase. La sintaxis de una operación en UML es:
- *Visibilidad nombre (lista de parámetros): tipo que retorna { propiedades}*

Elementos Diagramas de Clases

Objeto:

- ✓ Es una instancia de una clase. Se caracteriza por tener una identidad única, un estado definido por un conjunto de valores de atributos y un comportamiento representado por sus operaciones y métodos.

Asociación:

- ✓ (Rol, multiplicidad, calificador): representan las relaciones entre instancias de clase. Una asociación es una línea que une dos o más clases.

Elementos Diagramas de Clases

Rol:

- ✓ Identificado como un nombre a los finales de la línea, describe la semántica de la relación en el sentido indicado. Cada asociación tiene dos roles; cada rol es una dirección en la asociación. El rol puede estar representado en el nombre de la clase.

Multiplicidad:

- ✓ Describe la cardinalidad de la relación, es decir, cuanto objetos de esa clase pueden participar en la relación dada.

Lenguaje Unificado de Modelado (UML)

Diagramas de Secuencia

- ✓ El Diagrama de Secuencia es uno de los diagramas más efectivos para modelar interacción entre objetos en un sistema.
- ✓ Un diagrama de secuencia se modela para cada caso de uso. Mientras que el diagrama de caso de uso permite el modelado de una vista 'business' del escenario, el diagrama de secuencia contiene detalles de implementación del escenario, incluyendo los objetos y clases que se usan para implementar el escenario, y mensajes pasados entre los objetos.
- ✓ Típicamente uno examina la descripción de un caso de uso para determinar qué objetos son necesarios para la implementación del escenario.

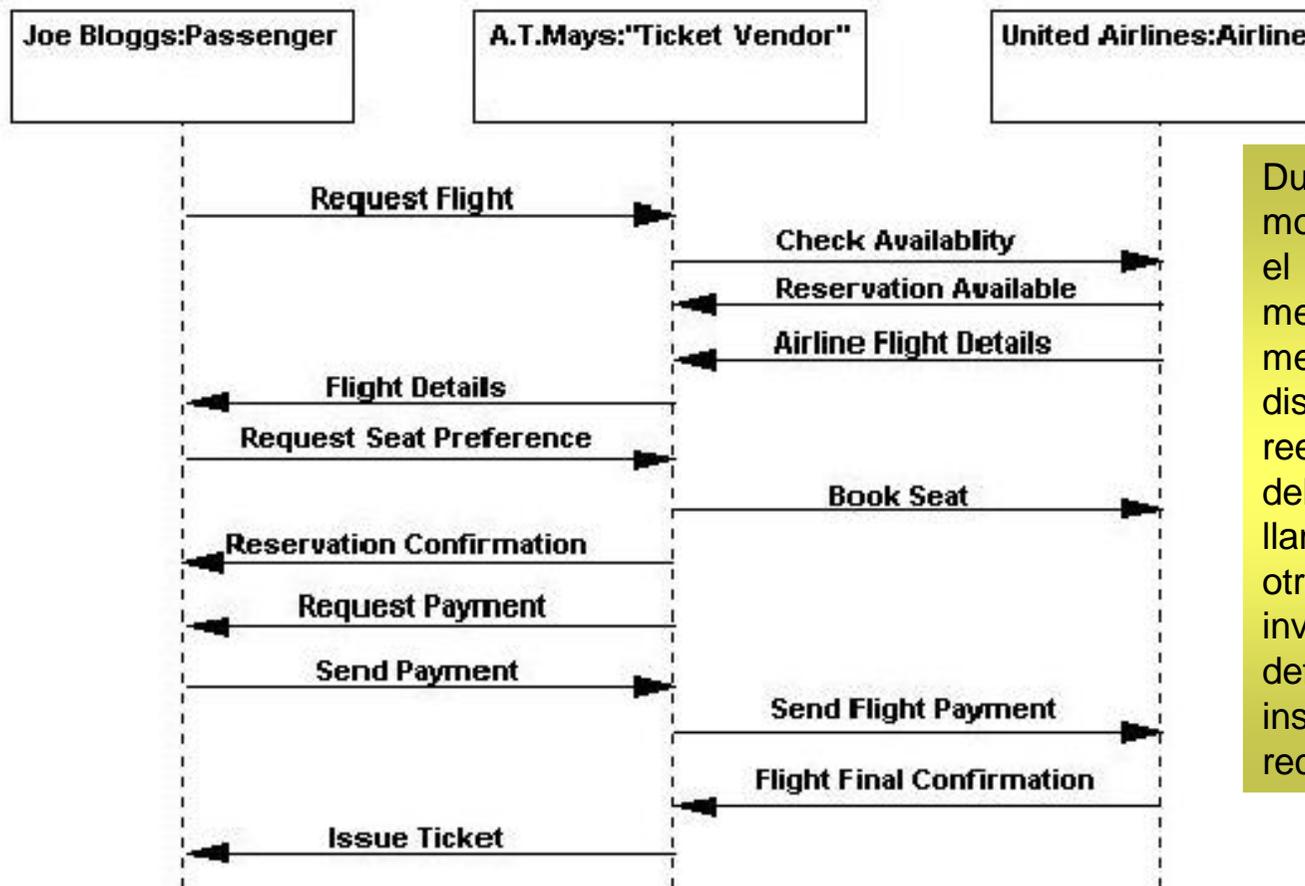
Lenguaje Unificado de Modelado (UML)

Diagramas de Secuencia

- ✓ Si tienes modelada la descripción de cada caso de uso como una secuencia de varios pasos, entonces puedes "caminar sobre" esos pasos para descubrir qué objetos son necesarios para que se puedan seguir los pasos.
- ✓ Un diagrama de secuencia muestra los objetos que intervienen en el escenario con líneas discontinuas verticales, y los mensajes pasados entre los objetos como vectores horizontales. Los mensajes se dibujan cronológicamente desde la parte superior del diagrama a la parte inferior; la distribución horizontal de los objetos es arbitraria.

Lenguaje Unificado de Modelado (UML)

Diagramas de Secuencia



Durante el análisis inicial, el modelador típicamente coloca el nombre 'business' de un mensaje en la línea del mensaje. Más tarde, durante el diseño, el nombre 'business' es reemplazado con el nombre del método que está siendo llamado por un objeto en el otro. El método llamado, o invocado, pertenece a la definición de la clase instanciada por el objeto en la recepción final del mensaje.

Lenguaje Unificado de Modelado (UML)

Diagramas de Colaboración

- ✓ El Diagrama de Colaboración presenta una alternativa al diagrama de secuencia para modelar interacciones entre objetos en el sistema.
- ✓ Mientras que el diagrama de secuencia se centra en la secuencia cronológica del escenario que estamos modelando, el diagrama de colaboración se centra en estudiar todos los efectos de un objeto dado durante un escenario.
- ✓ Los objetos se conectan por medio de enlaces, cada enlace representa una instancia de una asociación entre las clases implicadas.
- ✓ El enlace muestra los mensajes enviados entre los objetos, el tipo de mensaje (sincrónico, asincrónico, simple, blanking, y 'time-out'), y la visibilidad de un objeto con respecto a los otros.

Lenguaje Unificado de Modelado (UML)

Diagramas de Colaboración

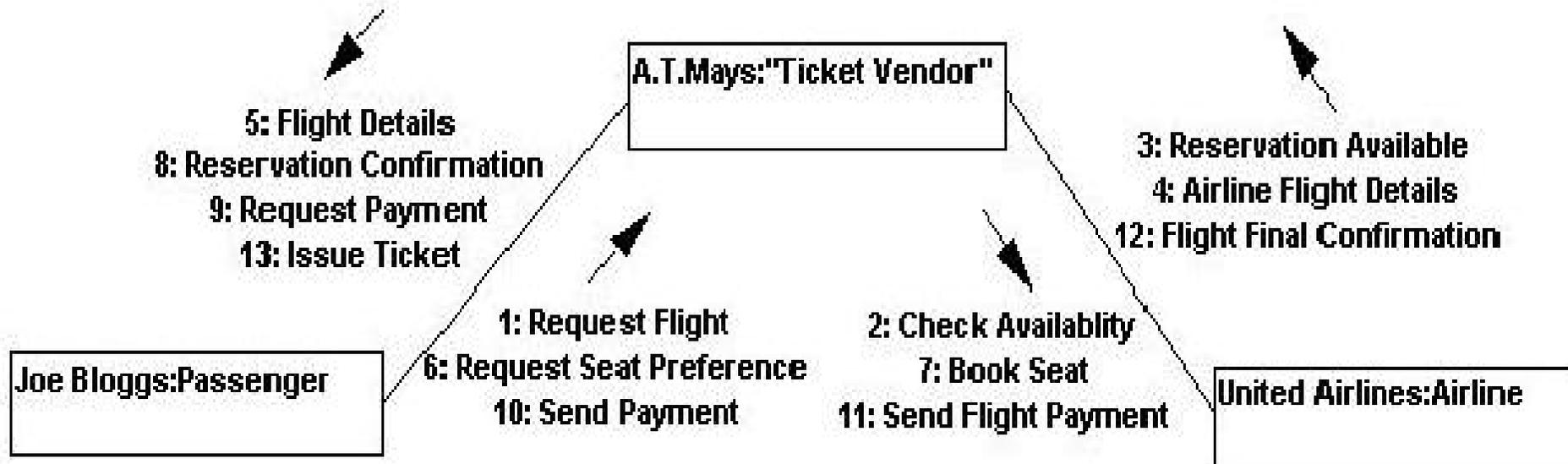


Figura :Diagrama de Colaboración para un grupo de Objetos

Lenguaje Unificado de Modelado (UML)

Diagramas de Estado

- ✓ Mientras los diagramas de interacción y colaboración modelan secuencias dinámicas de acción entre grupos de objetos en un sistema, el diagrama de estado se usa para modelar el comportamiento dinámico de un objeto en particular, o de una clase de objetos.
- ✓ Un diagrama de estado se modela para todas las clases que se consideran con un comportamiento dinámico. En él, modelas la secuencia de estado que un objeto de la clase atraviesa durante su vida en respuesta a los estímulos recibidos, junto con sus propias respuestas y acciones.
- ✓ Por ejemplo, un comportamiento de un objeto se modela en términos de en qué estado está inicialmente, y a qué estado cambia cuando recibe un evento en particular. También modelas qué acciones realiza un objeto en un estado en concreto.

Lenguaje Unificado de Modelado (UML)

Diagramas de Estado

- ✓ Los estados representan las condiciones de objetos en ciertos puntos en el tiempo. Los eventos representan incidentes que hacen que los objetos pasen de un estado a otro.
- ✓ Las líneas de transición describen el movimiento desde un estado hasta otro. Cada línea de transición se nombra con el evento que causa esta transición. Las acciones ocurren cuando un objeto llega a un estado.

Lenguaje Unificado de Modelado (UML)

Diagramas de Estado

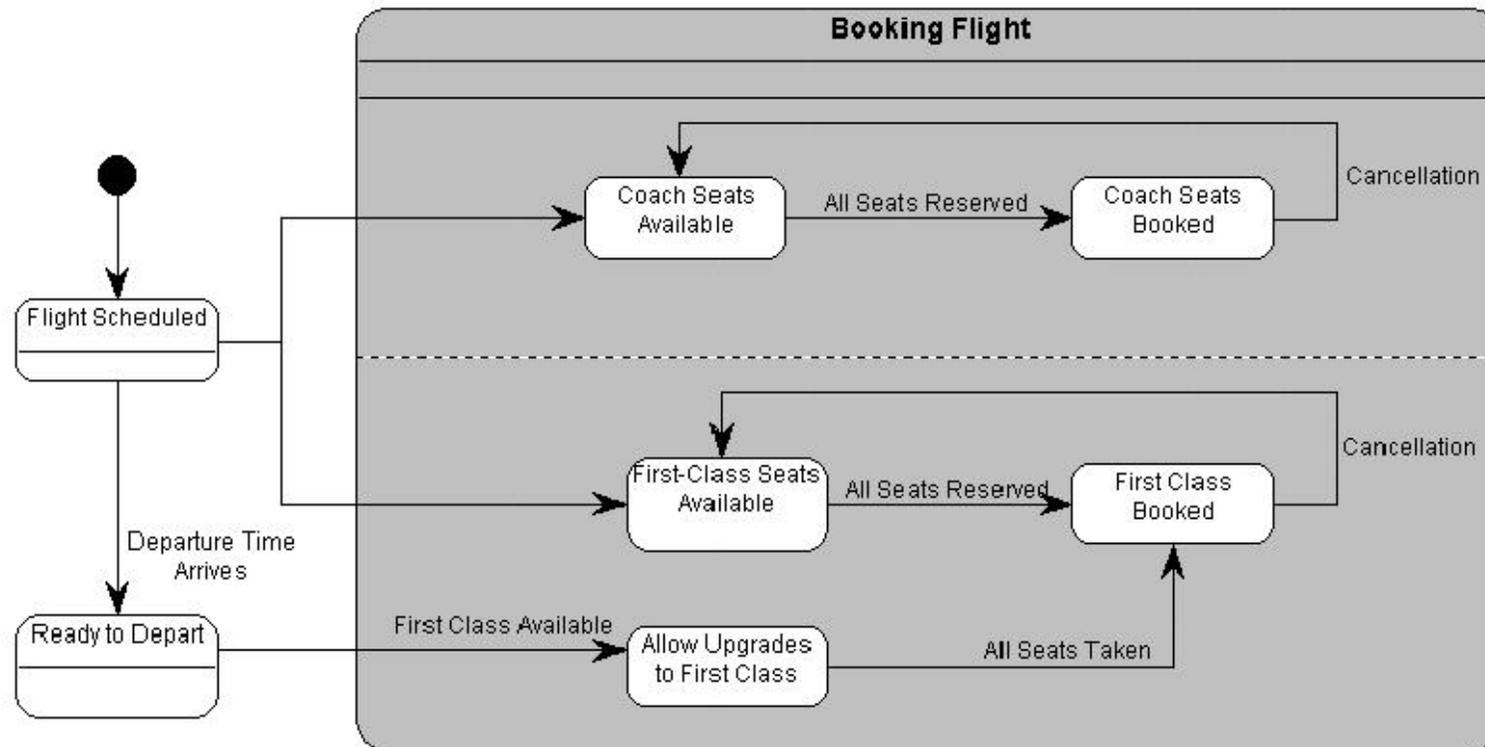


Figura: Modelando Comportamiento Dinámico de un objeto 'Vuelo' con un diagrama de estado

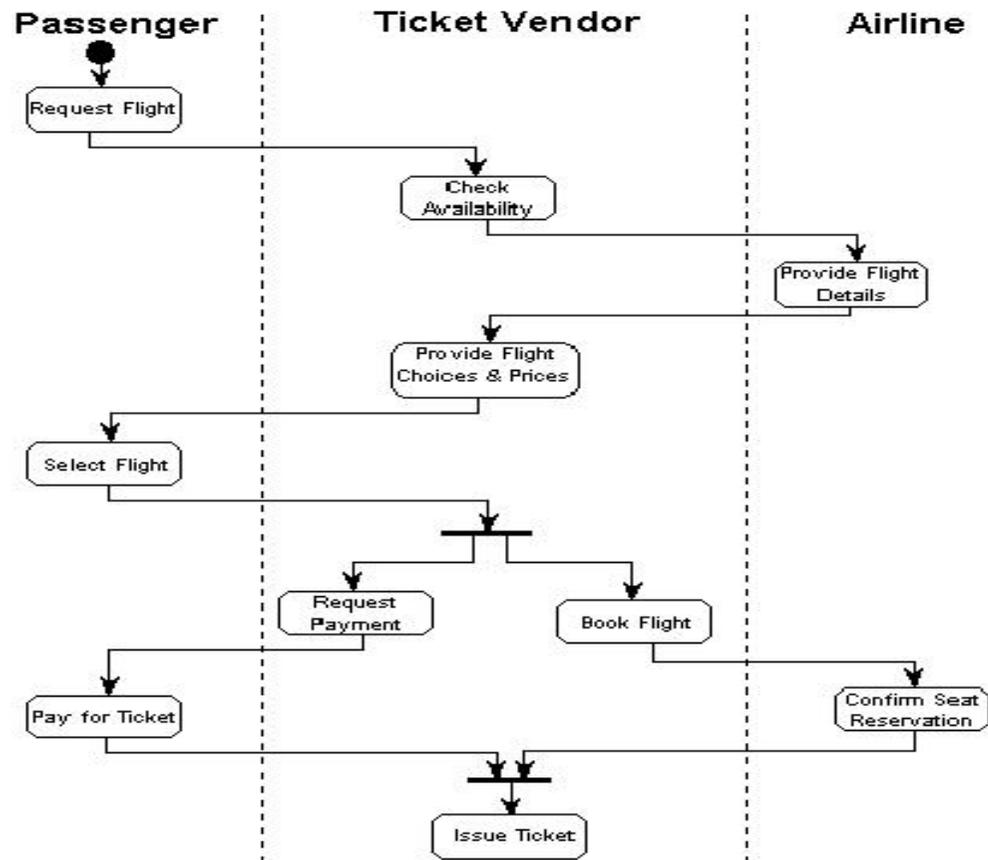
Lenguaje Unificado de Modelado (UML)

Diagramas de Actividad

- ✓ El Diagrama de Actividad es un diagrama de flujo del proceso multipropósito que se usa para modelar el comportamiento del sistema.
- ✓ Los diagramas de actividad se pueden usar para modelar un Caso de Uso, o una clase, o un método complicado.
- ✓ Un diagrama de actividad es parecido a un diagrama de flujo; la diferencia clave es que los diagramas de actividad pueden mostrar procesamiento paralelo (parallel processing).
- ✓ Esto es importante cuando se usan diagramas de actividad para modelar procesos 'business' algunos de los cuales pueden actuar en paralelo, y para modelar varios hilos en los programas concurrentes.

Lenguaje Unificado de Modelado (UML)

Diagramas de Actividad



Lenguaje Unificado de Modelado (UML)

Modelando Componentes de Software

- ✓ El Diagrama de Componentes se usa para modelar la estructura del software, incluyendo las dependencias entre los componentes de software, los componentes de código binario, y los componentes ejecutables.
- ✓ En el Diagrama de Componentes modela componentes del sistema, a veces agrupados por paquetes, y las dependencias que existen entre componentes (y paquetes de componentes).

Lenguaje Unificado de Modelado (UML)

Modelando Componentes de Software

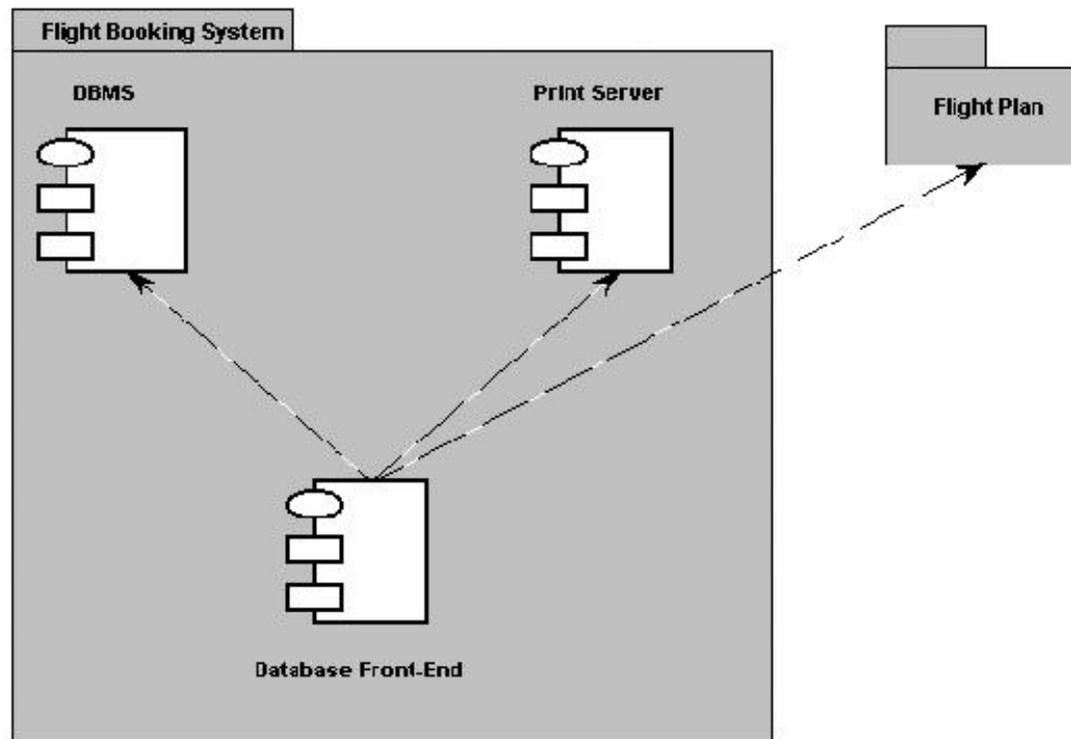


Figura : Modelando componentes con el Diagrama de Componentes

Lenguaje Unificado de Modelado (UML)

Modelando la Distribución y la Implementación

- ✓ Los Diagramas de Implementación se usan para modelar la configuración de los elementos de procesamiento en tiempo de ejecución (run-time processing elements) y de los componentes, procesos y objetos de software que viven en ellos.
- ✓ En el diagrama 'deployment', empiezas modelando nodos físicos y las asociaciones de comunicación que existen entre ellos. Para cada nodo, puedes indicar qué instancias de componentes viven o corren (se ejecutan) en el nodo.
- ✓ También puedes modelar los objetos que contiene el componente.
- ✓ Los Diagramas de Implementación se usan para modelar sólo componentes que existen como entidades en tiempo de ejecución; no se usan para modelar componentes solo de tiempo de compilación o de tiempo de enlazado.

Lenguaje Unificado de Modelado (UML)

Modelando la Distribución y la Implementación

- ✓ Puedes también modelar componentes que migran de nodo a nodo u objetos que migran de componente a componente usando una relación de dependencia con el estereotipo 'becomes' (se transforma)

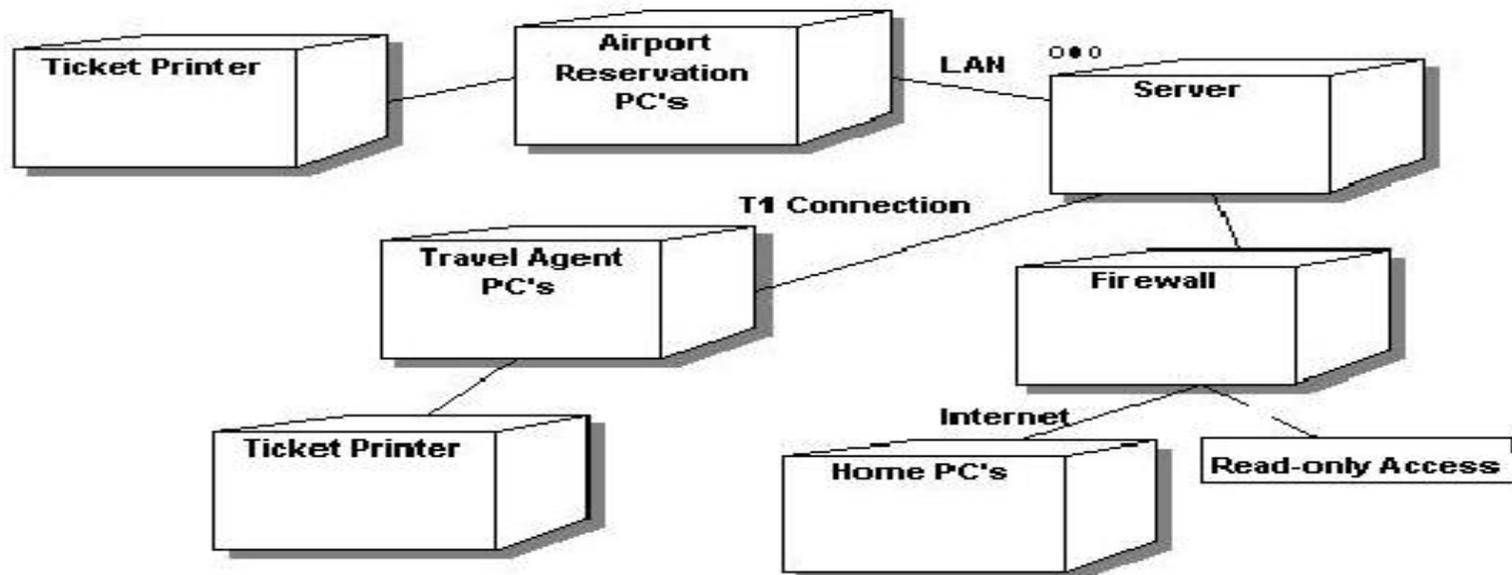


Figura : Modelando la Distribución del Sistema con el Diagrama de Implementación