

HORA 5

Agregación, composición, interfaces y realización

Continuaremos con las relaciones entre clases y comprenderá nuevos conceptos respecto a las clases y sus diagramas.

En esta hora se tratarán los siguientes temas:

- Agregaciones
- Composiciones
- Contextos
- Interfaces y realizaciones
- Visibilidad



Ya ha visto lo concerniente a asociación, multiplicidad y herencia y está casi listo para crear diagramas de clases significativos. Conforme explore otros tipos de relaciones y detalles relacionados con las clases comprenderá las piezas finales del rompecabezas. La meta final es crear una idea estática de un sistema, con todas las conexiones entre las clases que lo conforman.

Agregaciones

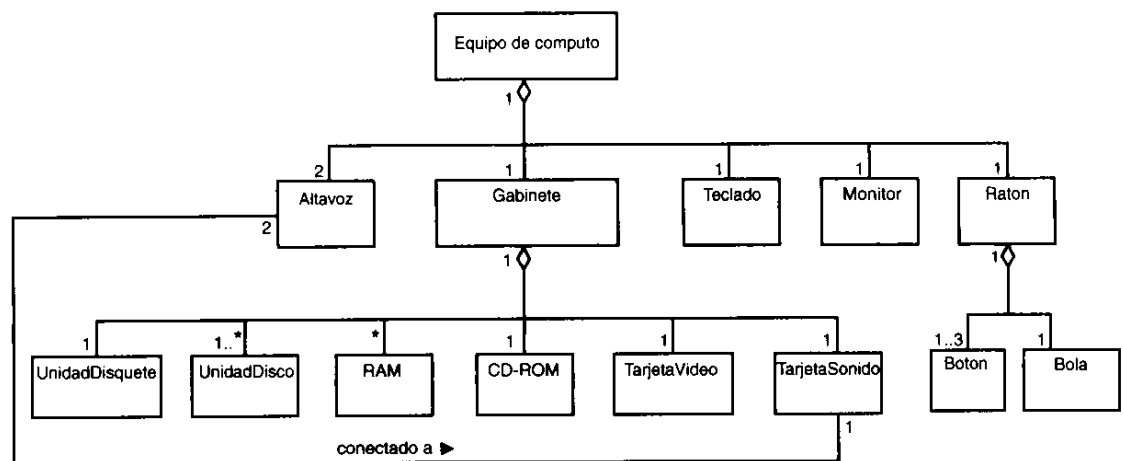
TÉRMINO NUEVO

En ocasiones una clase consta de otras clases. Éste es un tipo especial de relación conocida como *agregación* o *acumulación*. Los componentes y la clase que constituyen son una asociación que conforma un todo. En la hora 2, "Orientación a objetos", mencioné que su computadora es un conjunto de elementos que consta de gabinete, teclado, ratón, monitor, unidad de CD-ROM, una o varias unidades de disco duro, módem, unidad de disquete, impresora y, posiblemente, altavoces. Además de las unidades de disco, el gabinete contiene la memoria RAM, una tarjeta de vídeo y una tarjeta de sonido (tal vez algunos otros elementos).

Puede representar una agregación como una jerarquía dentro de la clase completa (por ejemplo el sistema computacional) en la parte superior, y los componentes por debajo de ella. Una línea conectará el todo con un componente mediante un rombo sin relleno que se colocará en la línea más cercana al todo. La figura 5.1 le muestra el sistema de cómputo como una agregación.

FIGURA 5.1

Una asociación por agregación se representa por una línea entre el componente y el todo con un rombo sin relleno que conforma al todo.



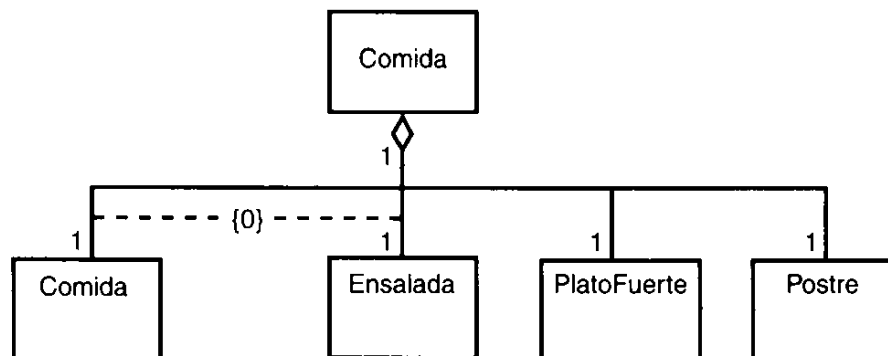
Aunque este ejemplo le muestra cada componente correspondiente a un todo, en una agregación éste no será necesariamente el caso. Por ejemplo: en un sistema casero de entretenimiento, un control remoto podría ser un componente de una televisión, aunque también podría ser un componente de una reproductora de casetes de vídeo.

Restricciones en las agregaciones

En ocasiones el conjunto de componentes posibles en una agregación se establece dentro de una relación O. En ciertos restaurantes, una comida consta de sopa o ensalada, el plato fuerte y el postre. Para modelar esto, utilizaría una restricción: la palabra O dentro de llaves con una línea discontinua que conecte las dos líneas que conforman al todo, como lo muestra la figura 5.2.

FIGURA 5.2

Puede establecer una restricción a una agregación para mostrar que un componente u otro es parte del todo.

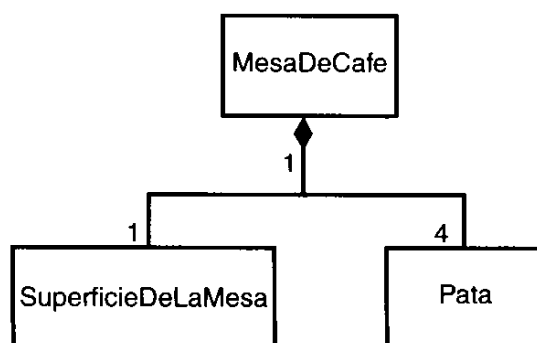


Composiciones

Una composición es un tipo muy representativo de una agregación. Cada componente dentro de una composición puede pertenecer tan sólo a un todo. Los componentes de una mesa de café (la superficie de la mesa y las patas) establecen una composición. El símbolo de una composición es el mismo que el de una agregación, excepto que el rombo está relleno (vea la figura 5.3).

FIGURA 5.3

En una composición, cada componente pertenece solamente a un todo. Un rombo relleno representa esta relación.



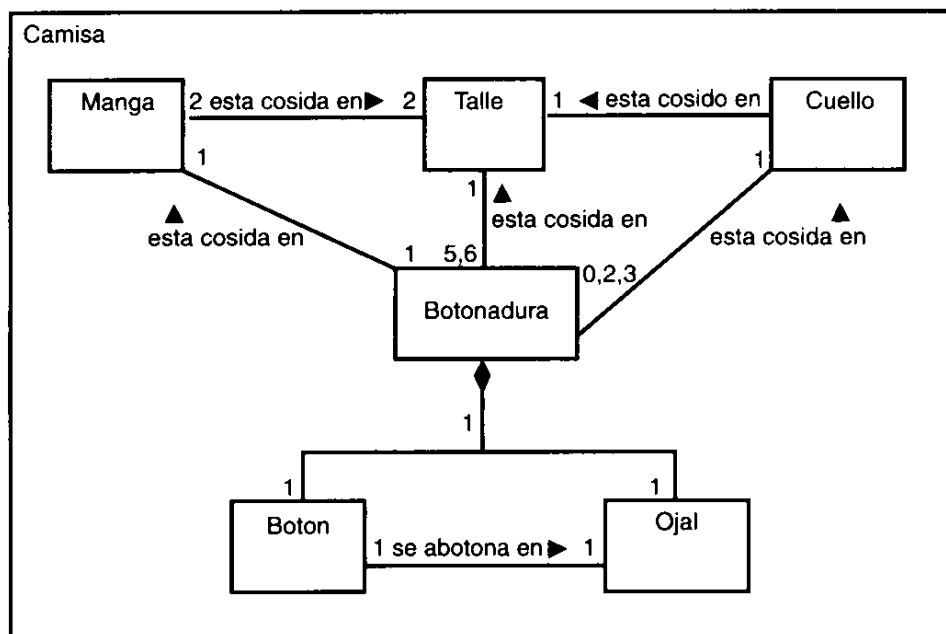
Contextos

Cuando modele un sistema podrían producirse, con frecuencia, agrupamientos de clases, como agregaciones o composiciones. En tal caso, deberá enfocar su atención en un agrupamiento o en otro, y el diagrama de contexto le proporciona la característica de modelaje que requiere para tal fin. Las composiciones figuran en gran medida dentro de los diagramas de contexto. Un diagrama de contexto es como un mapa detallado de alguna sección de un mapa de mayores dimensiones. Pueden ser necesarias varias secciones para capturar toda la información detallada.

He aquí un ejemplo: suponga que está creando un modelo de una camisa y la forma en que se podría combinar con algún atuendo y un guardarropa. Un tipo de diagrama de contexto (vea la figura 5.4) le mostrará la camisa como un gran rectángulo de clase, con un diagrama anidado en el interior, el cual le muestra cómo los componentes de la camisa están relacionados entre sí. Éste es un diagrama de contexto de composición (dado que la sola camisa reúne a cada componente se le denomina *de composición*).

FIGURA 5.4

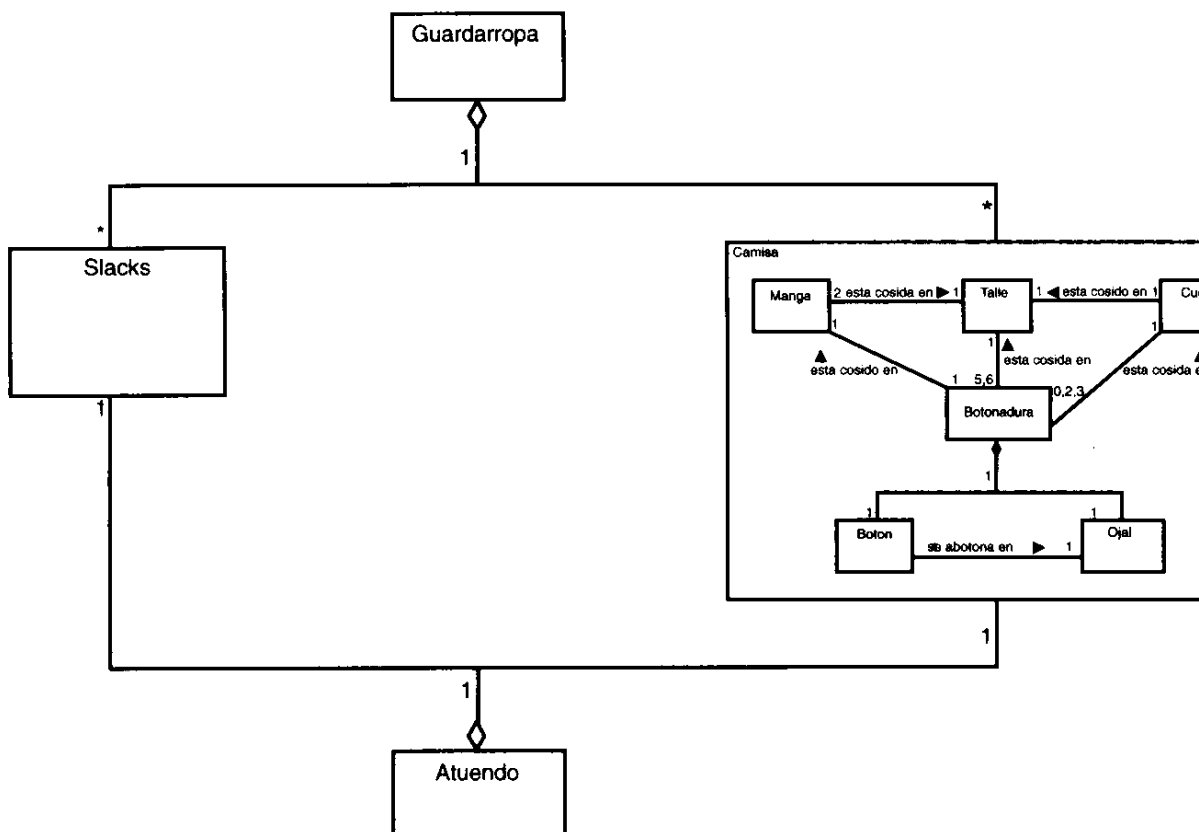
Un diagrama de contexto de composición le muestra los componentes de una clase como un diagrama anidado dentro de un enorme rectángulo de clase.



El diagrama de contexto de composición enfoca la atención en la camisa y sus componentes. Para mostrar la camisa en el contexto del guardarropa y de algún atuendo, tendrá que ampliar su ámbito. Un diagrama de contexto del sistema lo hará por usted. Podrá mostrar la forma en que la clase Camisa se conecta con las clases Guardarropa y Atuendo, como se ve en la figura 5.5.

FIGURA 5.5

Un diagrama de contexto del sistema le muestra los componentes de una clase y la forma en que la clase se relaciona con las otras que hay en el sistema.



Podrá ver de cerca alguna otra clase y presentar sus detalles en algún otro diagrama de contexto.

Interfaces y realizaciones

Una vez que haya creado varias clases, tal vez se dé cuenta que no pertenecen a una clase principal, pero en su comportamiento debe incluir algunas de las mismas operaciones con las mismas firmas de la primera clase. Podría codificar las operaciones en una de las clases y reutilizarlas en otras. Una segunda posibilidad es que desarrolle una serie de operaciones para las clases en un sistema, y reutilizarlas para las clases de otro sistema.

TERMINO NUEVO

De cualquier manera, deseará contar con algún medio para capturar el conjunto reutilizable de operaciones. La interfaz es la estructura del UML que le permite hacerlo. Una *interfaz* es un conjunto de operaciones que especifica cierto aspecto de la funcionalidad de una clase, y es un conjunto de operaciones que una clase presenta a otras.

Con un ejemplo podríamos aclarar lo anterior. El teclado que usted utiliza para comunicarse con su equipo es una interfaz reutilizable. Su operación basada en la opresión de teclas ha provenido de la máquina de escribir. La disposición de las teclas es casi la misma que en una máquina de escribir, pero el punto principal es que la operación por opresión de teclas ha sido cedida de un sistema a otro. Otras operaciones (Mayús, Bloq Mayús y Tab) también se integraron a partir de la máquina de escribir.

Por supuesto, el teclado de una computadora incluye diversas operaciones que no encontrará en una máquina de escribir: Control, Alt, RePág, AvPág y otras. Así pues, la interfaz puede establecer un subconjunto de las operaciones de una clase y no necesariamente todas ellas.

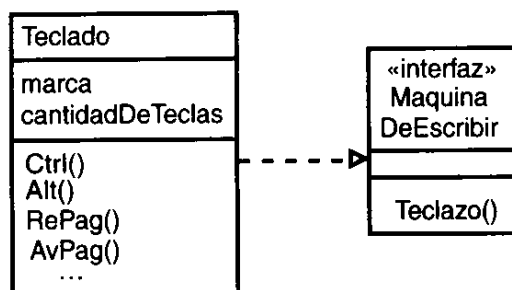
Puede modelar una interfaz del mismo modo en que modelaría una clase, con un símbolo rectangular. La diferencia será que, como un conjunto de operaciones, una interfaz no tiene atributos. Recordará que puede omitir los atributos de la representación de una clase. ¿Entonces, cómo distinguiría entre una interfaz y una clase que no muestra sus atributos? Una forma es utilizar la estructura “estereotipo” y especificar la palabra «interfaz» sobre el nombre de la interfaz en el rectángulo. Otra es colocar la letra “I” al principio del nombre de una interfaz.

TERMINO NUEVO

En cierto sentido, es como si el teclado de la computadora garantizara que esta parte de su funcionalidad “haría las veces” del teclado de una máquina de escribir. Bajo este esquema, la relación entre una clase y una interfaz se conoce como *realización*. Esta relación está modelada como una línea discontinua con una punta de flecha en forma de triángulo sin rellenar que adjunte y apunte a la interfaz. La figura 5.6 le muestra cómo se lleva a cabo esto.

FIGURA 5.6

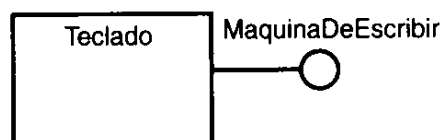
Una interfaz es un conjunto de operaciones que realiza una clase. Esta última se relaciona con una interfaz mediante la realización, misma que se indica por una línea discontinua con una punta de flecha en forma de triángulo sin rellenar que apunte a la interfaz.



Otra forma (omitida) de representar una clase y su interfaz es con un pequeño círculo que se conecta mediante una línea a la clase, como se ve en la figura 5.7.

FIGURA 5.7

La forma omitida de representar una clase que realice una interfaz.



Una clase puede realizar más de una interfaz, y una interfaz puede ser realizada por más de una clase.

Visibilidad

TERMINO NUEVO

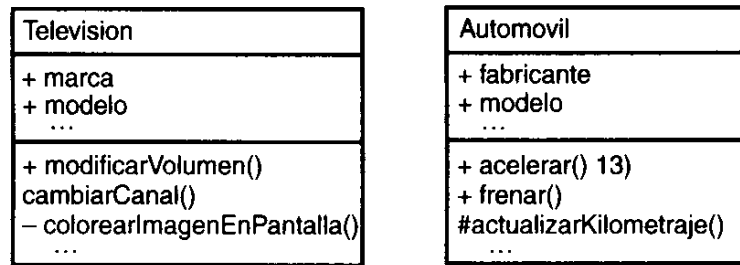
El concepto de visibilidad está muy relacionado con las interfaces y la realización. La *visibilidad* se aplica a atributos u operaciones, y establece la proporción en que otras clases podrán utilizar los atributos y operaciones de una clase dada (o en operaciones de una interfaz). Existen tres niveles de visibilidad: Nivel *público*, en el cual la funcionalidad se extiende a otras clases. En el nivel *protegido* la funcionalidad se otorga sólo a las clases que se heredan de la clase original. En el nivel *privado* sólo la clase original puede utilizar el atributo u operación. En una televisión, `modificarVolumen()` y `cambiarCanal()` son operaciones públicas, en tanto que `dibujarImagenEnPantalla()` es privada. En un automóvil, `acelerar()` y `frenar()` son operaciones públicas, pero `actualizarKilometraje()` o `actualizarMillaje()` es protegida.

La realización, como podría imaginar, implica que el nivel público se aplique a cualquier operación en una interfaz. La protección de operaciones mediante cualquiera de los otros niveles tal vez no tendría sentido, dado que una interfaz se orienta a ser realizada por diversas clases.

Para indicar el nivel público, anteceda el atributo u operación con un signo de suma (+), para revelar un nivel protegido, antecédalo con un símbolo de número (#), y para indicar el nivel privado, antecédalo con un guión (-). La figura 5.8 muestra los atributos y operaciones públicos, protegidos y privados tanto en una televisión como en un automóvil.

FIGURA 5.8

Los atributos y operaciones públicos y privados, tanto de una televisión como de un automóvil.



Ámbito

TERMINO NUEVO

El ámbito es otro concepto referente a los atributos y operaciones, y la forma en que se relacionan dentro de un sistema. Hay dos tipos de ámbitos, el de instancia y el de archivador. En el primero cada instancia cuenta con su propio valor en un atributo u operación. En un ámbito de *archivado*, sólo habrá un valor del atributo u operación en todas las instancias de la clase. Un atributo u operación con el ámbito de archivador, aparece con su nombre subrayado. Este tipo de ámbito se utiliza con frecuencia cuando un grupo específico de instancias (ningunas otras) tienen que compartir los valores exactos de un atributo privado. El ámbito de instancia es, por mucho, el tipo más común de ámbito.

Resumen

Para completar sus nociones de clases y la forma en que se conectan, es necesario comprender algunas relaciones adicionales. Una agregación establece una asociación para conformar un todo: una clase “todo” se genera de clases que la componen. Un componente en una agregación puede ser parte de más de un todo. Una composición es una conformación muy íntimamente ligada con la agregación en el sentido de que un componente de una composición puede ser parte solamente de un todo. La representación del UML de las agregaciones es similar a la representación de las composiciones. La línea de asociación que une la parte con un todo tiene un rombo. En una agregación, el rombo no está relleno, en tanto que en una composición sí lo está.

Un diagrama de contexto enfoca la atención en una clase específica dentro de un sistema. Un diagrama de contexto de composición es como un mapa detallado de un mapa mayor. Muestra un diagrama de clases anidado dentro de un gran símbolo rectangular de clase. Un diagrama de contexto de sistema muestra la forma en que el diagrama de clases compuestas se relaciona con otros objetos del sistema.

Una realización es una asociación entre una clase y una interfaz, una colección de operaciones que cierta cantidad de clases podrá utilizar. Una interfaz se representa como una clase sin atributos. Para distinguirla de una clase cuyos atributos hayan sido omitidos del diagrama, el estereotipo «interfaz» aparecerá por encima del nombre de la interfaz. Otra posibilidad es la de anteceder el nombre de la interfaz con una “I” mayúscula. La realización se representa en el UML mediante una línea discontinua con una punta de flecha en forma de triángulo sin rellenar que conecta a la clase con la interfaz. Otra forma para representar una realización es con una línea continua que conecte a una clase con un pequeño círculo, para que el círculo se interprete como la interfaz.

En términos de visibilidad, todas las operaciones en una interfaz son *públicas*, de modo que cualquier clase podrá utilizarlas. Los otros dos niveles de visibilidad son *protegido* (la funcionalidad se extiende a las clases secundarias de aquella que contiene los atributos y operaciones) y *privado* (atributos y operaciones que se pueden utilizar sólo dentro de la clase que los contiene). Un signo de suma (+) denota a la visibilidad pública, el símbolo de número (#) la protegida y el guión (-) la privada.

El ámbito es otro aspecto de los atributos y operaciones. En un ámbito de instancia, cada objeto de una clase cuenta con su propio valor en un atributo u operación. En un ámbito de archivador, sólo hay un valor para un atributo u operación en particular a través de un conjunto de objetos de una clase. Los objetos que no estén en este conjunto no podrán acceder al valor contenido en el ámbito de archivador.

Preguntas y respuestas

- P** ¿Se considera transitiva a la agregación? Es decir, si la clase 3 es un componente de la clase 2, y la clase 2 es un componente de la clase 1, ¿la clase 3 será un componente de la clase 1?
- R** Así es, la agregación es transitiva. En nuestro ejemplo, los botones y la bola del ratón son parte del ratón, a la vez que son parte de la computadora.
- P** ¿La palabra “interfaz” implica “interfaz de usuario” o GUI?
- R** No. Es algo más genérico. Una interfaz es tan sólo un conjunto de operaciones que una clase presenta a las demás clases. De hecho, una de estas operaciones podría ser (aunque no necesariamente) la del usuario.

Taller

El cuestionario y los ejercicios verificarán y fortalecerán su conocimiento respecto al tema de las agregaciones, composiciones, contextos e interfaces. Las respuestas las podrá ver en el Apéndice A, “Respuestas a los cuestionarios”.

Cuestionario

1. ¿Cuál es la diferencia entre una agregación y una composición?
2. ¿Qué es la realización?
3. Mencione los tres niveles de visibilidad y describa lo que significa cada uno de ellos.

Ejercicios

1. Cree un diagrama de contexto de composición de una revista. Tome en cuenta la tabla de contenido, la editorial, los artículos y las columnas. Luego, cree un diagrama de contexto del sistema que muestre a la revista junto con el suscriptor y el comprador en el puesto de revistas.

2. En la actualidad, el tipo más popular de GUI es la interfaz WIMP (ventanas, iconos, menús y puntero, por sus siglas en inglés). Dibuje un diagrama de clases de la interfaz WIMP, y haga uso de todo el conocimiento adecuado del UML que ha adquirido hasta ahora. Además de las clases indicadas en las siglas, incluya los elementos relacionados como las barras de desplazamiento y el cursor, así como cualquiera de las otras clases necesarias.

