



HORA 4

Uso de relaciones

En la hora anterior creamos un conjunto de clases que representaban el vocabulario del baloncesto. Aunque ello le da las bases para una mayor exploración de lo que es el baloncesto, tal vez haya sentido que algo le falta.

Ese “algo” es un sentido en el que las clases se relacionan entre sí. Si observa el modelo (vea la figura 3.15), verá que no se indica la manera en que un jugador se relaciona con un balón, ni cómo los jugadores conforman un equipo, ni la forma en que procede el juego. Es como si hubiera construido una lista de elementos, en lugar de una representación de un área del conocimiento. Es importante saber cómo se conectan las clases entre sí.

Ahora trazaremos las conexiones entre las clases y completaremos la representación.

En esta hora se tratarán los siguientes temas:

- Asociaciones
- Multiplicidad
- Asociaciones calificadas

- Asociaciones reflexivas
- Herencia y generalización
- Dependencias

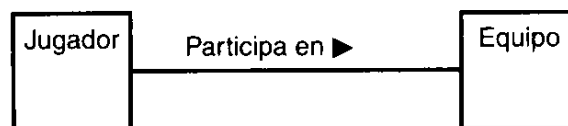
Asociaciones

TÉRMINO NUEVO

Cuando las clases se conectan entre sí de forma conceptual, esta conexión se conoce como *asociación*. El modelo inicial de baloncesto le dará algunos ejemplos. Examinemos uno de ellos: la asociación entre un jugador y un equipo. Podrá caracterizar tal asociación con la frase: “un jugador participa en un equipo”. Visualizará la asociación como una línea que conectará a ambas clases, con el nombre de la asociación (“participa en”) justo sobre la línea. Es útil indicar la dirección de la relación, y lo hará con un triángulo relleno que apunte en la dirección apropiada. La figura 4.1 le muestra cómo visualizar la asociación “Participa en” entre el jugador y el equipo.

FIGURA 4.1

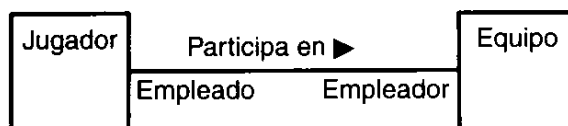
Una asociación entre un jugador y un equipo.



Cuando una clase se asocia con otra, cada una de ellas juega un papel dentro de tal asociación. Puede representar estos papeles en el diagrama escribiéndolos cerca de la línea que se encuentra junto a la clase que juega el papel correspondiente. En la asociación entre un jugador y un equipo, si el equipo es profesional, éste es un empleador y el jugador es un empleado. La figura 4.2 le muestra cómo representar dichos papeles.

FIGURA 4.2

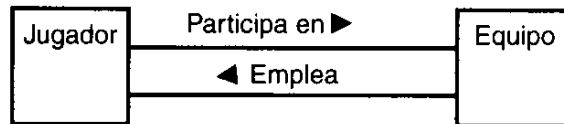
Por lo general, en una asociación cada clase juega un papel. Puede representar tales papeles en el diagrama.



La asociación puede funcionar en dirección inversa: un equipo emplea a jugadores. Podrá mostrar ambas asociaciones en el mismo diagrama con un triángulo relleno que indique la dirección de cada asociación, como en la figura 4.3.

FIGURA 4.3

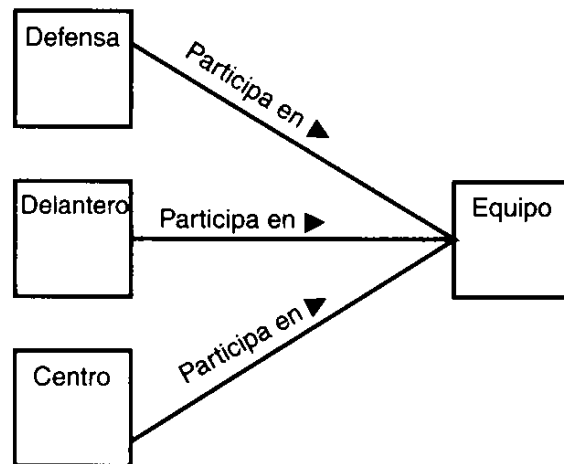
Pueden aparecer dos asociaciones entre clases en el mismo diagrama.



Las asociaciones podrían ser más complejas que tan sólo una clase conectada a otra. Varias clases se pueden conectar a una. Si toma en cuenta los defensas, delanteros y central, así como sus asociaciones con la clase Equipo, tendrá el diagrama de la figura 4.4.

FIGURA 4.4

Pueden asociarse diversas clases con una en particular.

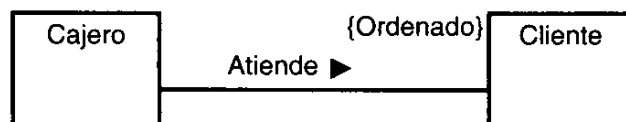


Restricciones en las asociaciones

En ocasiones una asociación entre dos clases debe seguir cierta regla. Ésta se indica al establecer una restricción junto a la línea de asociación. Por ejemplo: un Cajero atiende a un Cliente, pero cada Cliente es atendido en el orden en que se encuentre en la formación. Puede capturar este modelo colocando la palabra *ordenado* entre llaves (para indicar la restricción) junto a la clase Cliente, como se ve en la figura 4.5.

FIGURA 4.5

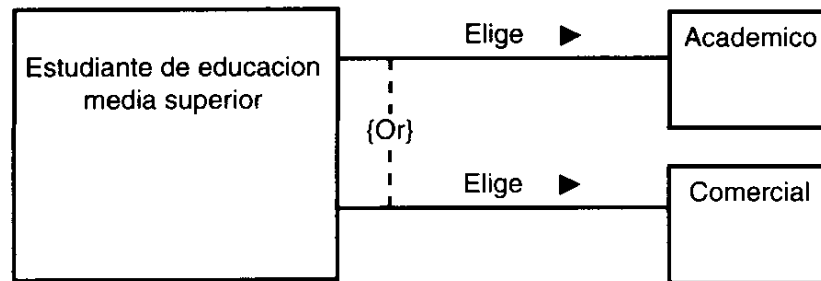
Puede establecer una restricción en una asociación. En este caso, la asociación Atiende está restringida para que el Cajero atienda al Cliente en turno.



Otro tipo de restricción es la relación O (distinguida como {Or}) en una línea discontinua que conecte a dos líneas de asociación. La figura 4.6 modela a un estudiante de educación media superior que elegirá entre un curso académico o uno comercial.

FIGURA 4.6

La relación *O* entre dos asociaciones en una restricción.



Clases de asociación

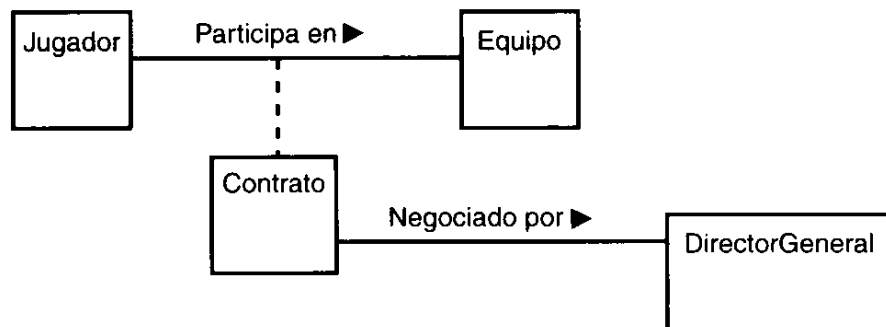
TÉRMINO NUEVO

Una asociación, al igual que una clase, puede contener atributos y operaciones. De hecho, cuando éste sea el caso, usted tendrá una *clase de asociación*.

Puede concebir a una clase de asociación de la misma forma en que lo haría con una clase estándar, y utilizará una línea discontinua para conectarla a la línea de asociación. Una clase de asociación puede tener asociaciones con otras clases. La figura 4.7 le muestra una clase de asociación para la asociación “Participa en” entre un jugador y un equipo. La clase de asociación, Contrato, se asocia con la clase DirectorGeneral.

FIGURA 4.7

Una clase de asociación modela los atributos y operaciones de una asociación. Se conecta a una asociación mediante una línea discontinua, y puede asociarse a otra clase.

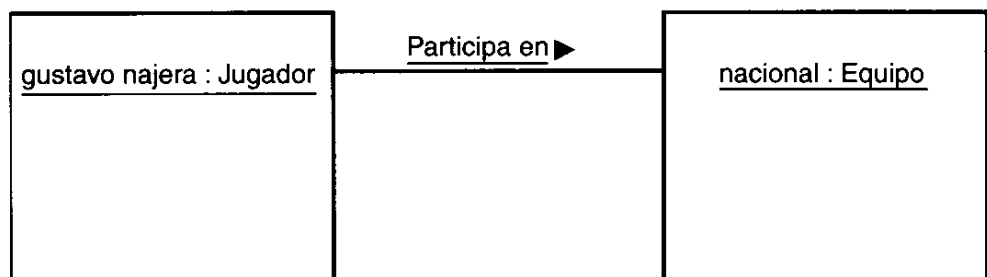


Vínculos

Así como un objeto es una instancia de una clase, una asociación también cuenta con instancias. Si podemos imaginar a un jugador específico que juega para un equipo específico, la relación “Participa en” se conocerá como *vínculo*, y usted lo representará como una línea que conecta a dos objetos. Tal como tuvo que subrayar el nombre de un objeto, deberá subrayar el nombre de un vínculo, como en la figura 4.8.

FIGURA 4.8

Un vínculo es la instancia de una asociación. Conecta a los objetos en lugar de las clases. Deberá subrayar el nombre del vínculo, como se hace en el nombre de un objeto.



Multiplicidad

La asociación trazada entre Jugador y Equipo sugiere que las dos clases tienen una relación de uno a uno. No obstante, el sentido común nos indica que éste no es el caso. Un equipo de baloncesto cuenta con cinco jugadores (sin contar a los sustitutos). La asociación Tiene (Has) debe participar en este recuento. En la otra dirección, un jugador puede participar sólo en un equipo, y la asociación “Participa en” debe responder de esto.

TÉRMINO NUEVO

Tales especificaciones son ejemplos de la *multiplicidad*: la cantidad de objetos de una clase que se relacionan con un objeto de la clase asociada. Para representar los números en el diagrama, los colocará sobre la línea de asociación junto a la clase correspondiente, como se denota en la figura 4.9.

FIGURA 4.9

La multiplicidad señala la cantidad de objetos de una clase que pueden relacionarse con un objeto de una clase asociada.



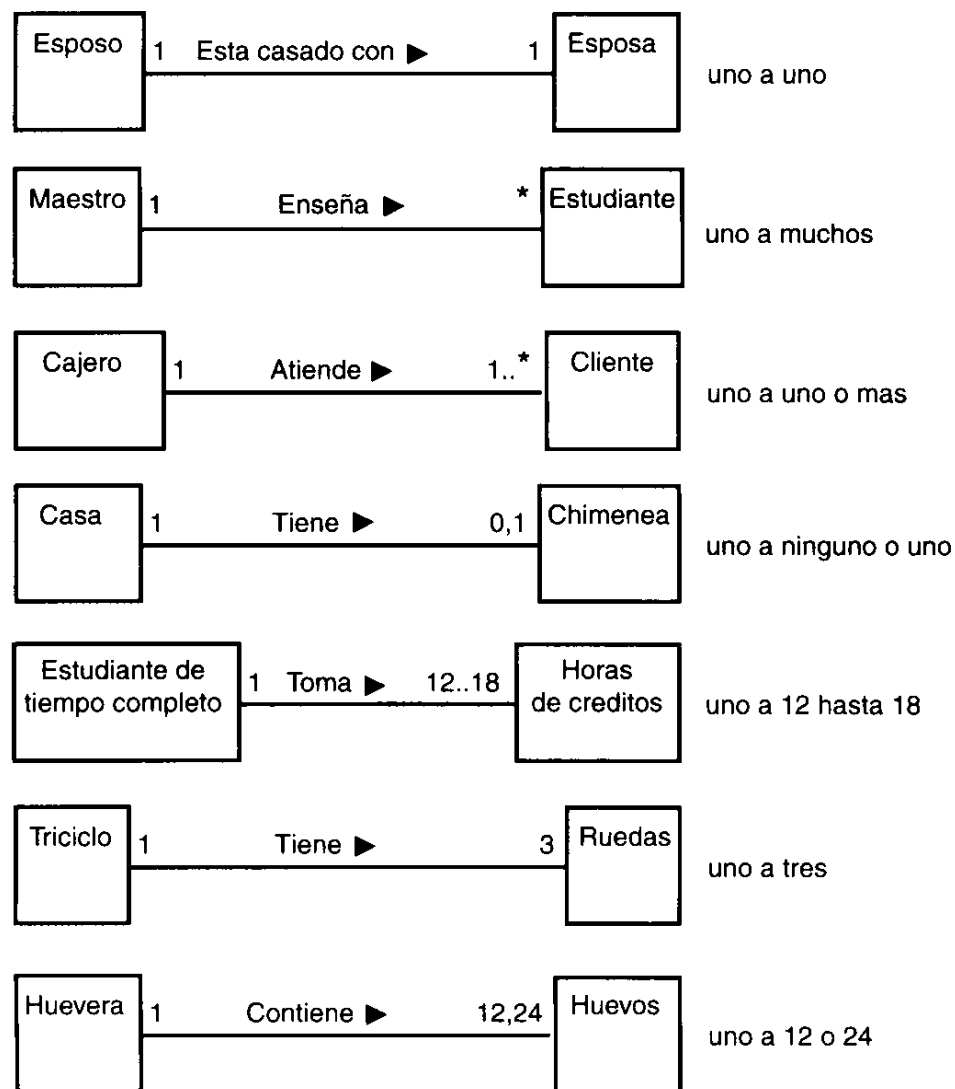
La multiplicidad de este ejemplo no es la única que existe. Hay varios tipos de multiplicidades (una multiplicidad de multiplicidades, por decirlo así). Una clase puede relacionarse con otra en un esquema de uno a uno, uno a muchos, uno a uno o más, uno a ninguno o uno, uno a un intervalo definido (por ejemplo: uno a cinco hasta diez), uno a exactamente n (como en este ejemplo), o uno a un conjunto de opciones (por ejemplo, uno a nueve o diez). El UML utiliza un asterisco (*) para representar *más* y para representar *muchos*. En un contexto O se representa por dos puntos, como en “1..*” (“uno o más”). En otro contexto, O se representa por una coma, como en “5, 10” (“5 o 10”). La figura 4.10 le muestra cómo concebir las posibles multiplicidades.



Cuando la clase A tiene una multiplicidad de uno a ninguno o uno con la clase B, la clase B se dice que es opcional para la clase A.

FIGURA 4.10

Posibles multiplicidades y cómo representarlas en el UML.



Asociaciones calificadas

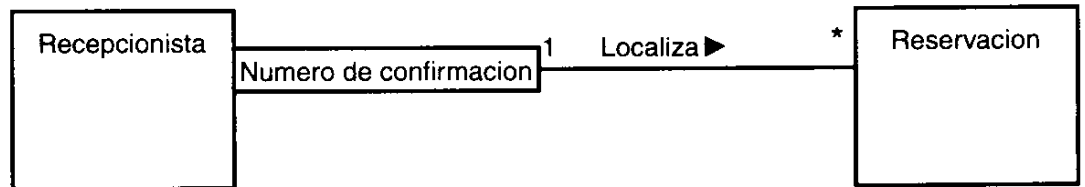
Cuando la multiplicidad de una asociación es de uno a muchos, con frecuencia se presenta un reto muy particular: la búsqueda. Cuando un objeto de una clase tiene que seleccionar un objeto particular de otro tipo para cumplir con un papel en la asociación, la primera clase deberá atenerse a un atributo en particular para localizar al objeto adecuado. Normalmente, dicho atributo es un identificador que puede ser un número de identidad. Por ejemplo, cuando usted realiza una reservación en un hotel, el hotel le asigna un número de confirmación. Si usted quiere hacer preguntas respecto a la reservación, deberá proporcionar el número de confirmación.

TÉRMINO NUEVO

En el UML la información de identidad se conoce como *calificador*. Su símbolo es un pequeño rectángulo adjunto a la clase que hará la búsqueda. La figura 4.11 muestra la representación. La idea es reducir, con eficiencia, la multiplicidad de uno a muchos a una multiplicidad de uno a uno.

FIGURA 4.11

Un calificador en una asociación resuelve el problema de la búsqueda.

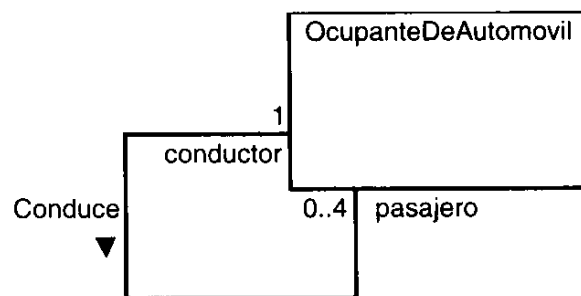


Asociaciones reflexivas

En ocasiones, una clase es una asociación consigo misma. Esto puede ocurrir cuando una clase tiene objetos que pueden jugar diversos papeles. Un OcupanteDeAutomovil puede ser un Conductor o un Pasajero. En el papel del conductor, el OcupanteDeAutomovil puede llevar ninguno o más OcupanteDeAutomovil, quienes jugarán el papel de pasajeros. Esto lo representará mediante el trazado de una línea de asociación a partir del rectángulo de la clase hacia el mismo rectángulo de la clase, y en la línea de asociación indicará los papeles, nombre de la asociación, dirección de la asociación y multiplicidad como ya lo hizo antes. La figura 4.12 le presenta este ejemplo.

FIGURA 4.12

En una asociación reflexiva, trazará la línea de la clase hacia sí misma y podrá incluir los papeles, nombre de la asociación y su dirección, así como su multiplicidad.



Herencia y generalización

Uno de los sellos distintivos de la orientación a objetos es que captura uno de los mayores aspectos del sentido común en cuanto a la vida diaria: si usted conoce algo de una categoría de cosas, automáticamente sabrá algunas cosas que podrá transferir a otras categorías. Si usted sabe que algo es un electrodoméstico, ya sabrá que contará con un interruptor, una marca y un número de serie. Si sabe que algo es un animal dará por hecho que come, duerme, tiene una forma de nacer, de trasladarse de un lugar a otro y algunos otros atributos (y operaciones) que podría listar si pensara en ello por algunos instantes.

TÉRMINO NUEVO

La orientación a objetos se refiere a esto como *herencia*. El UML también lo denomina *generalización*. Una clase (la clase secundaria o subclase) puede heredar los atributos y operaciones de otra (la clase principal o superclase). La clase principal (o madre) es más genérica que la secundaria (o hija).



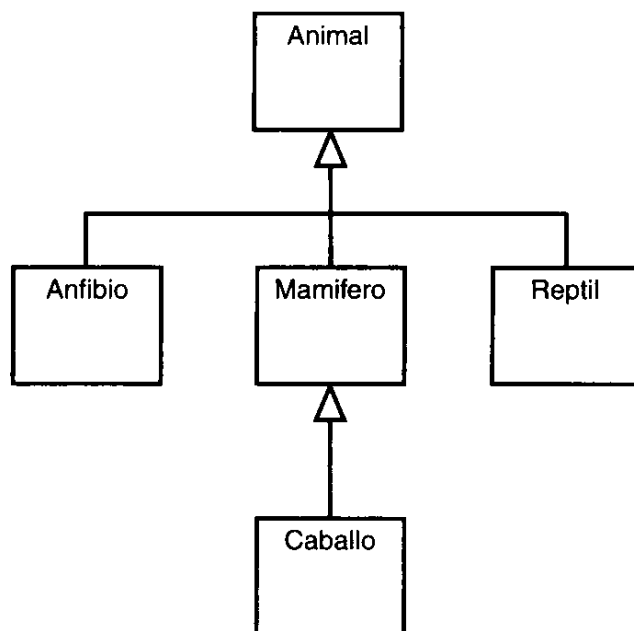
En la generalización, una clase secundaria (hija) es sustituible por una clase principal (madre). Es decir, donde quiera que se haga referencia a la clase madre, también se hace referencia a la clase hija. Sin embargo, en el caso contrario no es aplicable.

La jerarquía de la herencia no tiene que finalizar en dos niveles: una clase secundaria puede ser principal para otra clase secundaria. Un Mamífero es una clase secundaria de Animal, y Caballo es una clase secundaria de Mamífero.

En el UML representará la herencia con una línea que conecte a la clase principal con la secundaria. En la parte de la línea que se conecta con la clase principal, colocará un triángulo sin rellenar que apunte a la clase principal. Este tipo de conexión se interpreta con la frase *es un tipo de*. Un Mamífero *es un tipo de* Animal, y un Caballo *es un tipo de* Mamífero. La figura 4.13 le muestra esta particular jerarquía de la herencia, junto con otras clases. Observe la apariencia del triángulo y las líneas cuando varias clases secundarias son herencia de una clase principal. Al disponer el diagrama de este modo, trae por resultado un diagrama más ordenado en lugar de mostrar todas las líneas y triángulos, aunque el UML no le prohíbe colocarlos todos en la imagen. También vea que no colocó los atributos y operaciones heredadas en los rectángulos de las subclases, dado que ya los había representado en la superclase.

FIGURA 4.13

Una jerarquía de herencia en el reino animal.



Cuando modele la herencia, tenga la seguridad de que la clase secundaria satisfaga la relación *es un tipo de* con la clase principal. Si no se cumple tal relación, tal vez una asociación de otro tipo podría ser más adecuada.

Con frecuencia las clases secundarias agregan otras operaciones y atributos a los que han heredado. Por ejemplo: un Mamífero tiene pelo y da leche, dos atributos que no se encuentran en la clase Animal.

TERMINO NUEVO

Una clase puede no provenir de una clase principal, en cuyo caso será una *clase base* o *clase raíz*. Una clase podría no tener clases secundarias, en cuyo caso será una clase final o clase hoja. Si una clase tiene exactamente una clase principal, tendrá una *herencia simple*. Si proviene de varias clases principales, tendrá una *herencia múltiple*.

Descubrimiento de la herencia

En el proceso de plática con un cliente, un analista descubrirá la herencia de varias formas. Es posible que las clases candidatas que aparezcan incluyan tanto clases principales como clases secundarias. El analista deberá darse cuenta que los atributos y operaciones de una clase son generales y que se aplicarán a, quizá, varias clases (mismas que agregarán sus propios atributos y operaciones).

El ejemplo del baloncesto de la hora 3, “Uso de la orientación a objetos”, tiene las clases Jugador, Defensa, Delantero y Central. El Jugador tiene atributos como nombre, estatura, peso, velocidadAlCorrer y saltoVertical. Tiene operaciones como driblar(), pasar(), rebotar() y tirar(). Las clases Defensa, Delantero y Centro heredarán tales atributos y operaciones, y agregarán los suyos. La clase Defensa podría tener las operaciones correrAlFrente() y quitarBalon(). El Central podría tener retacarBalon(). De acuerdo con los comentarios del entrenador respecto a las estaturas de los jugadores, el analista tal vez quisiera colocar restricciones en las estaturas para cada posición.

Otra posibilidad es que el analista note que dos o más clases tienen ciertos atributos y operaciones en común. El modelo del baloncesto tiene un CronometroDeJuego (que controla el tiempo que resta en un periodo de juego) y un LapsodeTiro (que controla el tiempo restante desde el instante que un equipo tomó posesión del balón, hasta que intente encestar). Si nos damos cuenta de que ambos controlan el tiempo, el analista podría formular una clase Reloj con una operación controlarTiempo() que podrían heredar tanto CronometroDeJuego como LapsodeTiro.



Dado que LapsodeTiro controla 24 segundos (profesional) o 35 segundos (colegial) y el CronometroDeJuego controla 12 minutos (profesional) o 20 minutos (colegial), controlarTiempo() será polimórfico.

Clases abstractas

En el modelo del baloncesto, el par de clases que mencioné —Jugador y Reloj— son útiles puesto que funcionan como clases principales para clases secundarias importantes. Las clases secundarias son importantes en el modelo dado que finalmente usted querrá

tener instancias de tales clases. Para desarrollar el modelo, necesitará instancias de Defensa, Delantero, Centro, CronometroDeJuego y LapsodeTiro.

No obstante, Jugador y Reloj no proporcionan ninguna instancia al modelo. Un objeto de la clase Jugador no serviría a ningún propósito, así como tampoco uno de la clase Reloj.

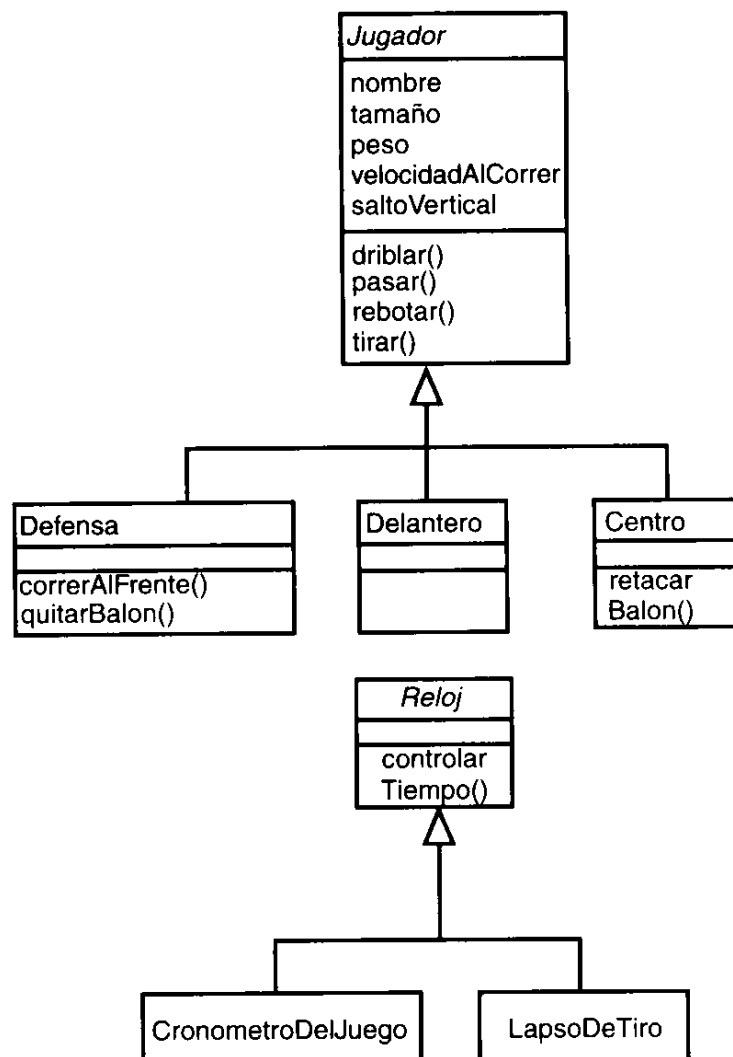
TERMINO NUEVO

Las clases como Jugador y Reloj —que no proveen objetos— se dice que son *abstractas*. Una clase abstracta se distingue por tener su nombre en cursivas.

La figura 4.14 muestra las dos clases abstractas y sus clases secundarias.

FIGURA 4.14

Dos jerarquías de herencia con clases abstractas en el modelo de baloncesto.



Dependencias

TERMINO NUEVO

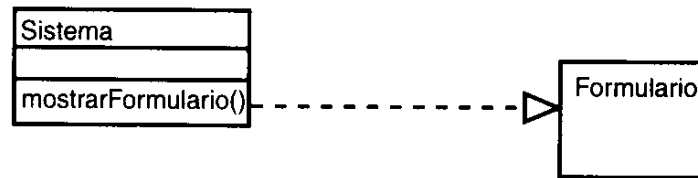
En otro tipo de relación, una clase utiliza a otra. A esto se le llama dependencia. El uso más común de una dependencia es mostrar que la firma de la operación de una clase utiliza a otra clase.

Suponga que diseñará un sistema que muestra formularios corporativos en pantalla para que los empleados los llenen. El empleado utiliza un menú para seleccionar el formulario por llenar. En su diseño, tiene una clase Sistema y una clase Formulario. Entre sus

muchas operaciones, la clase Sistema tiene mostrarFormulario(f:Form). El formulario que el sistema desplegará, dependerá, obviamente, del que elija el usuario. La notación del UML para ello es una línea discontinua con una punta de flecha en forma de triángulo sin relleno que apunta a la clase de la que depende, como muestra la figura 4.15.

FIGURA 4.15

Una flecha representada por una línea discontinua con una punta de flecha en forma de triángulo sin relleno simboliza una dependencia.



Resumen

Sin las relaciones, un modelo de clases sería poco menos que una lista de cosas que representarían un vocabulario. Las relaciones le muestran cómo se conectan los términos del vocabulario entre sí para dar una idea de la sección del mundo que se modela. La asociación es la conexión conceptual fundamental entre clases. Cada clase en una asociación juega un papel, y la multiplicidad especifica cuántos objetos de una clase se relacionan con un objeto de la clase asociada. Hay muchos tipos de multiplicidad. Una asociación se representa como una línea entre los rectángulos de clases con los papeles y multiplicidades en cada extremo. Al igual que una clase, una asociación puede contener atributos y operaciones.

Una clase puede heredar atributos y operaciones de otra clase. La clase heredada es secundaria de la clase principal que es de la que se hereda. Descubrirá la herencia cuando encuentre clases en su modelo inicial que tengan atributos y operaciones en común. Las clases abstractas sólo se proyectan como bases de herencia y no proporcionan objetos por sí mismas. La herencia se representa como una línea entre la clase principal y la secundaria, con un triángulo sin rellenar que se adjunta (y apunta a) la clase principal.

En una dependencia, una clase utiliza a otra. El uso más común de una dependencia es mostrar que una firma en la operación de una clase utiliza a otra clase. Una dependencia se proyecta como una línea discontinua que reúne a las dos clases en la dependencia, con una punta de flecha en forma de triángulo sin relleno que adjunta (y apunta a) la clase de la que se depende.

Preguntas y respuestas

- P** ¿En alguna ocasión se le puede poner nombre a una relación de herencia, como se hace en una asociación?
- R** El UML no le impide que adjudique un nombre a una relación de herencia, pero por lo general esto no es necesario.

Taller

El cuestionario y los ejercicios se han diseñado para reafirmar su conocimiento del UML en el área de las relaciones. Cada pregunta y ejercicio requiere que usted piense en la simbología del modelado que ha aprendido y la aplique a una situación. Las respuestas se encuentran en el Apéndice A, “Respuestas a los cuestionarios”.

Cuestionarios

1. ¿Cómo representaría la multiplicidad?
2. ¿Cómo descubrirá la herencia?
3. ¿Qué es una clase abstracta?
4. ¿Cuál es el efecto de un calificador?

Ejercicios

1. Tome como base el modelo del baloncesto de la Hora 3, y agregue vínculos que expresen las relaciones que ha visto en esta hora. Si conoce el juego del baloncesto, siéntase con libertad de agregar los vínculos que representen su conocimiento.
2. De acuerdo con un viejo adagio: “Un abogado que se defiende a sí mismo, tiene por cliente a un tonto.” Cree un modelo que refleje esta pieza de sabiduría.