

Relación entre actores:

- *Generalización de actor*: Una relación de generalización entre un actor más general y uno más específico. Por ejemplo, entre Empleado (actor más general) y Administrativo y Técnico (actores más específicos). Empleado es un actor abstracto ya que es una abstracción introducida simplemente para capturar el comportamiento de los actores concretos Administrativo y Técnico.

Los actores concretos heredan todos los roles y relaciones de los casos de uso de su padre abstracto. Un uso juicioso de actores puede simplificar los diagramas de casos de uso.

También simplifica la semántica porque trata elementos diferentes de la misma forma.

Merece la pena destacar que el actor padre en la generalización de actor no siempre tiene que ser abstracto, puede ser un rol concreto que una persona o sistema puede desempeñar. Sin embargo, un buen estilo dicta que los actores padre son normalmente abstractos para mantener la semántica de generalización sencilla.

Si dos actores se comunican con el mismo conjunto de casos de uso de la misma forma, este hecho se puede expresar como una generalización a otro actor (posiblemente abstracto). Los actores descendentes (hijos) heredan los roles y relaciones a casos de uso albergados por el actor padre. Se puede sustituir un actor descendente (hijo) en cualquier lugar en el que se espere al ascendiente (padre). Éste es el principio de sustitución que es una prueba importante para un uso correcto de generalización con cualquier clasificador.

Relación entre actores y casos de uso:

- *Asociación*. La relación entre un actor y un caso de uso se representa por una línea continua, que es normalmente el símbolo de asociación de UML. La asociación entre el actor y el caso de uso indica que el actor y el caso de uso se comunican de alguna manera.

Relaciones entre casos de uso:

- *Relación <<include>>*. Esta relación entre casos de uso permite incluir el comportamiento de un caso de uso en otro caso de uso. Al caso de uso que incluye se le denomina caso de uso base, y al caso de uso incluido caso de uso de inclusión. El caso de uso de inclusión proporciona comportamiento adicional a su caso de uso base.

La semántica de <<include>> es sencilla. El caso de uso base se ejecuta hasta que se alcanza el punto de inclusión, luego la ejecución se pasa al caso de uso de inclusión. Cuando termina el caso de uso de inclusión, el control regresa de nuevo al caso de uso base.

Un caso de uso base no está completo sin todos sus casos de uso de inclusión. Los casos de uso de inclusión forman partes integrales del caso de uso base. Sin embargo, los casos de uso de inclusión pueden, o no, estar completos. Si un caso de uso de inclusión no está completo, simplemente contiene un comportamiento que solamente tendrá sentido cuando se incluya en una base apropiada. En este caso, se dice que el caso de uso de inclusión no se puede instanciar, es decir, no se puede activar directamente por los actores, solamente se puede ejecutar cuando esté incluido en un caso de uso base. Sin embargo, si los casos de uso de inclusión están completos, actúan como casos de uso normales y se pueden instanciar. Es bastante razonable activarlos por actores. Por ejemplo, en la transparencia 27 del tema 2.3, el caso de uso de inclusión “Disponibilidad de película” es completo, puesto que está activado por el actor Empleado; sin embargo, el caso de uso “Identificación de socio” no está completo, su existencia solo tiene sentido como parte integral de los casos de uso “Alquilar película” y “Comprar película”

La representación gráfica de esta relación en un diagrama de casos de uso es de una flecha con línea discontinua desde el caso de uso base hacia el caso de uso de inclusión y el estereotipo <<include>>

- *Relación <<extend>>*. Esta relación proporciona un medio para insertar un nuevo comportamiento (caso de uso de extensión) en un caso de uso existente (caso de uso base). El caso de uso base proporciona un conjunto de puntos de extensión que son enganches donde se puede añadir nuevo comportamiento, el caso de uso de extensión proporciona un conjunto de segmentos de inserción que se pueden insertar en el caso de uso base en esos enganches. La relación <<extend>> se puede utilizar para especificar exactamente qué puntos de extensión se están extendiendo en el caso de uso base.

Lo que es interesante de <<extend>> es que el caso de uso base no sabe nada de los casos de uso de extensión; simplemente proporciona enganches para estos. De hecho, el caso de uso base está perfectamente completo sin sus extensiones. Esto es muy diferente de <<include>>, donde los casos de uso base están incompletos sin sus casos de uso de inclusión.

Los casos de uso de extensión son, por lo general, casos de uso no completos y, por lo tanto, no se pueden instanciar.(no son activados por ningún actor). Simplemente constan de uno o más fragmentos de comportamiento, conocidos como segmentos de inserción. La relación <<extend>> especifica el punto de extensión en el caso de uso base donde se insertará el segmento de inserción.

Los casos de uso de extensión pueden tener casos de uso de extensión y/o casos de uso de inclusión. Sin embargo, se deberá evitar esto, ya que puede hacer que el modelo de casos de uso sea demasiado complejo.

La relación <<extend>> se representa gráficamente en un diagrama de casos de uso mediante una flecha de línea discontinua desde el caso de uso de extensión hacia el caso de uso base con el estereotipo <<extend>>, y tiene tres notaciones posibles (ver transparencias 30 y 31 del tema 2.3): básica, extendida y extendida con condiciones. La extendida, incluye en la representación del caso de uso el nombre de los puntos de extensión y la extendida con condiciones en la que, además del punto de extensión, se representa explícitamente la condición que se debe cumplir para que se inserte el caso de uso de extensión.

- *Relación de generalización*. La generalización de casos de uso se utiliza cuando se tiene uno o más casos de uso que son realmente especificaciones o un caso de uso más general. Igual que la generalización de actores, solamente se debería utilizar cuando simplifique el modelo de casos de uso (ver transparencias 34 y 35 del tema 2.3).

-

-