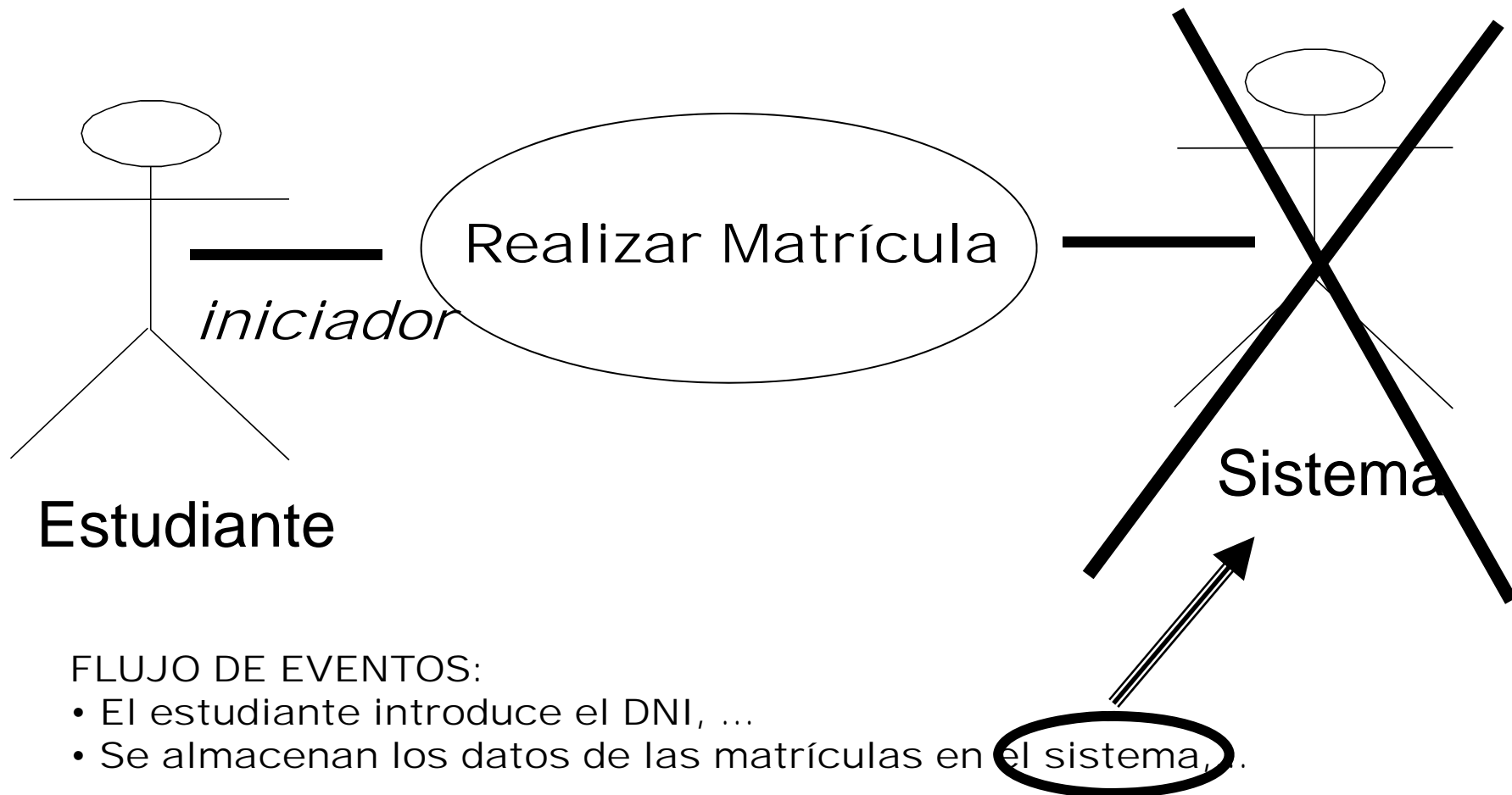


Errores típicos en captura de requisitos

Error típico: poner el sistema como actor del CU

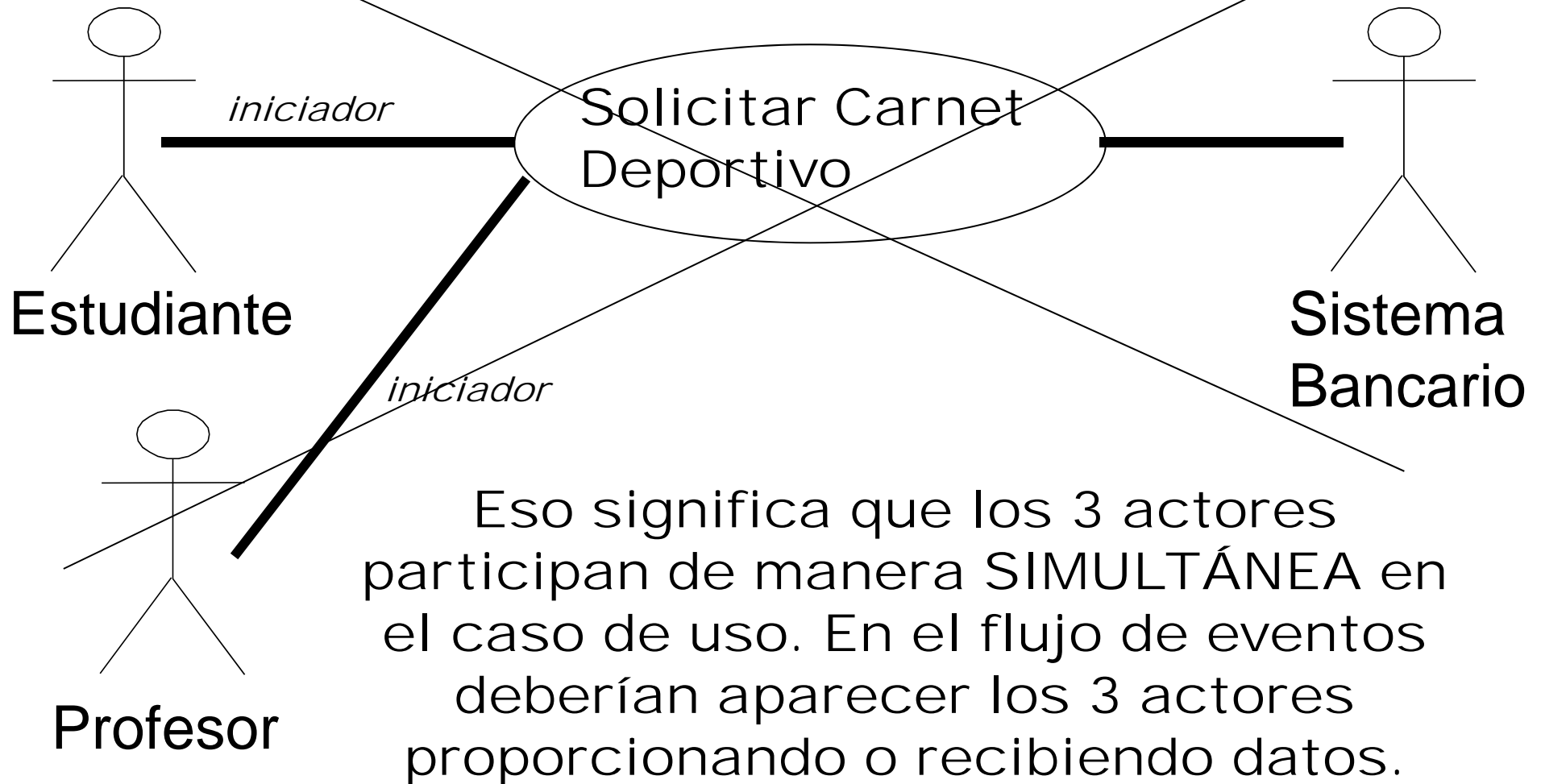


FLUJO DE EVENTOS:

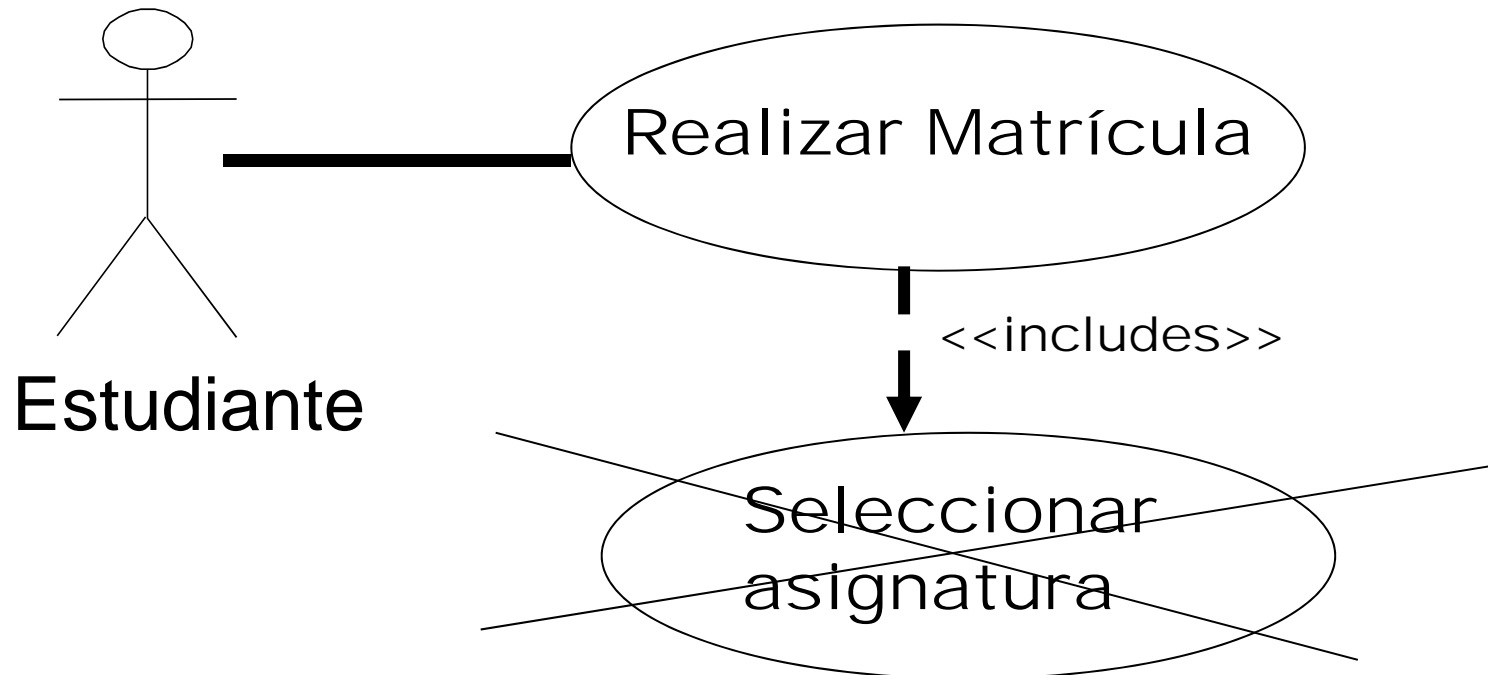
- El estudiante introduce el DNI, ...
- Se almacenan los datos de las matrículas en el sistema, ..

NO SE TRATA DE UN SISTEMA EXTERNO
SINO DEL PROPIO SISTEMA (EL C.U. ES PARTE DE ÉL)

Error típico: poner actores que no participan en el CU

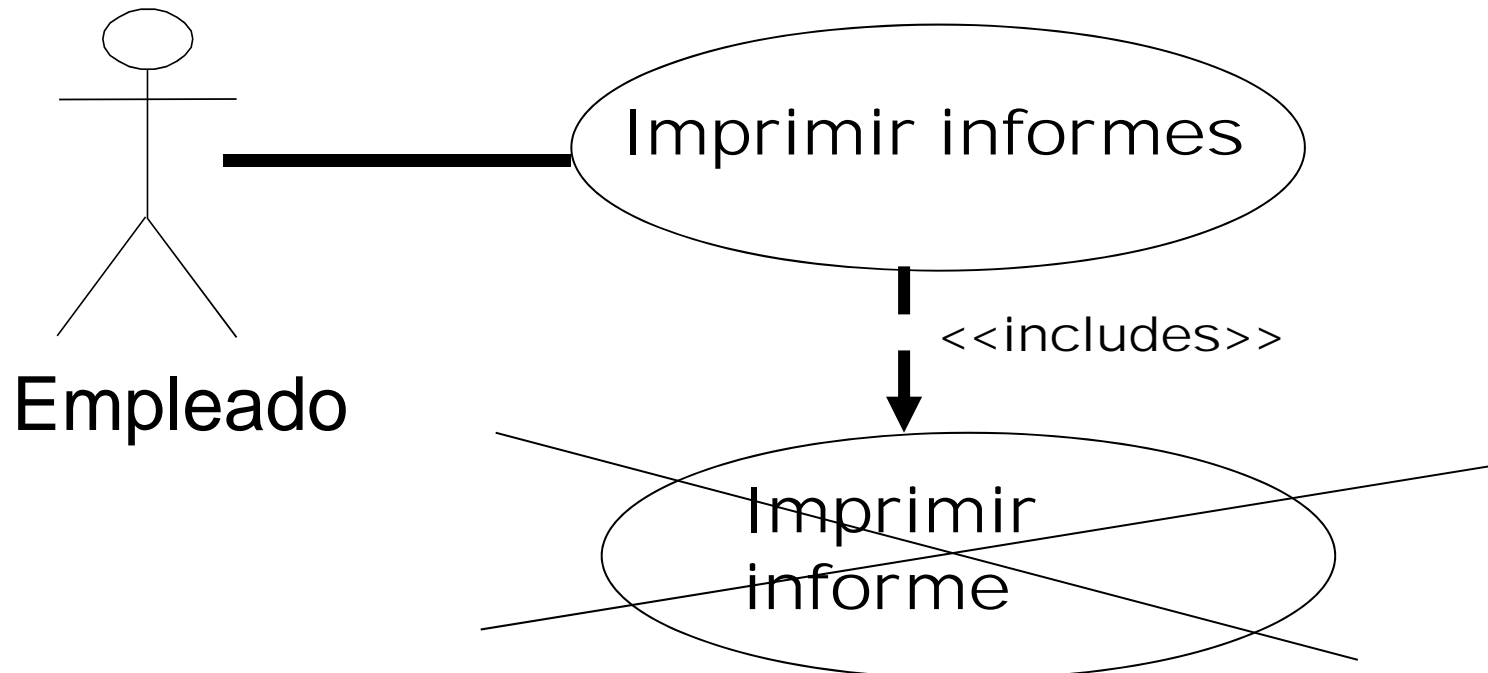


Errores típicos en CU: definir funcionalidad interna como CU



Si al realizar la matrícula, se van seleccionando (en la interfaz) asignaturas en las que matricularse NO ES CASO DE USO

Errores típicos en CU: definir funcionalidad interna como CU



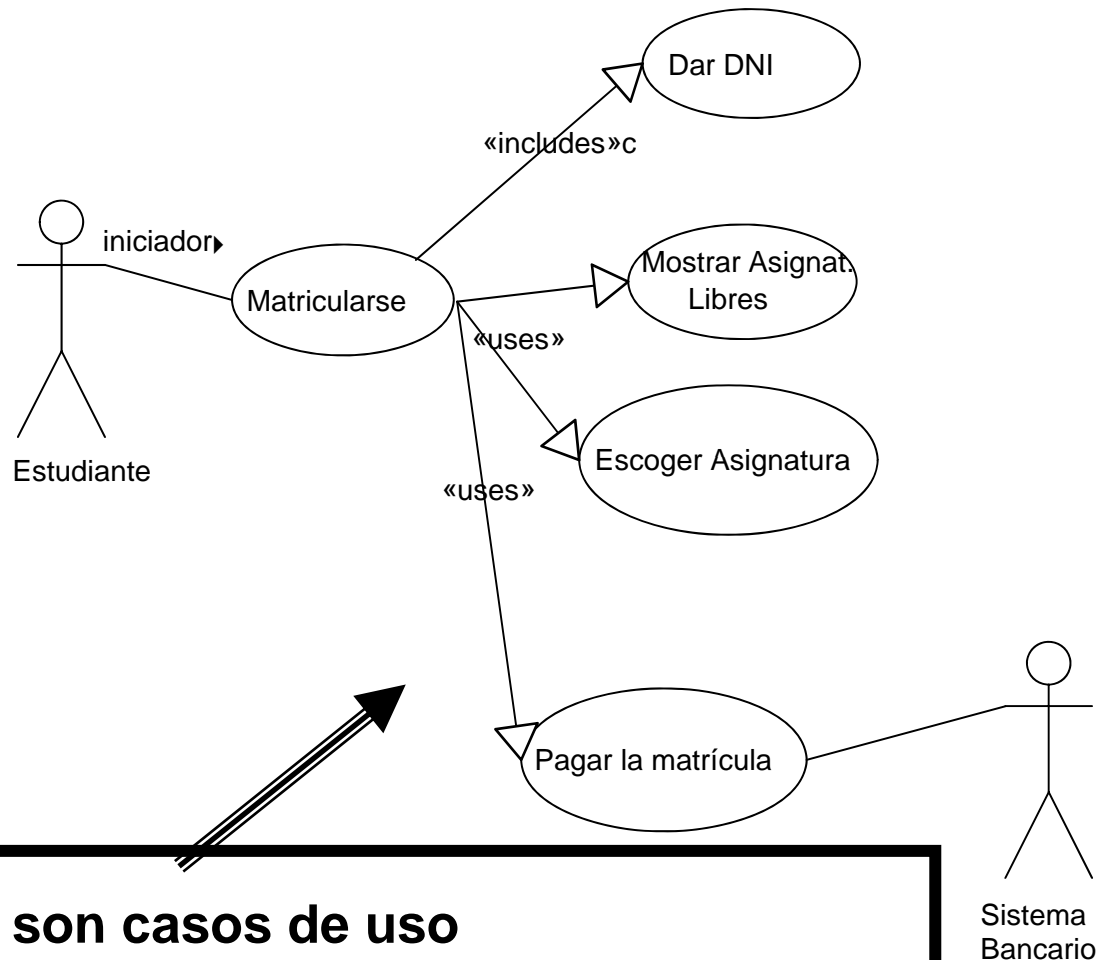
Un posible flujo de eventos sería:

- El empleado proporciona su identificador
- Se seleccionan los informes del empleado aún no impresos
- Se imprime cada uno de ellos

Error típico: realizar una descomposición funcional.

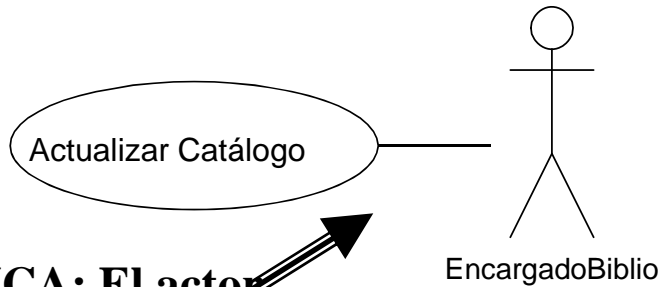
- Flujo de eventos

- El estudiante proporciona su dni.
- El sistema muestra las asignaturas en las que se puede matricular, si están libres todavía.
- El estudiante escoge las asignaturas que desee.
- El sistema calcula el precio de la matrícula y se realiza el pago en la cuenta del alumno por medio del sistema bancario.



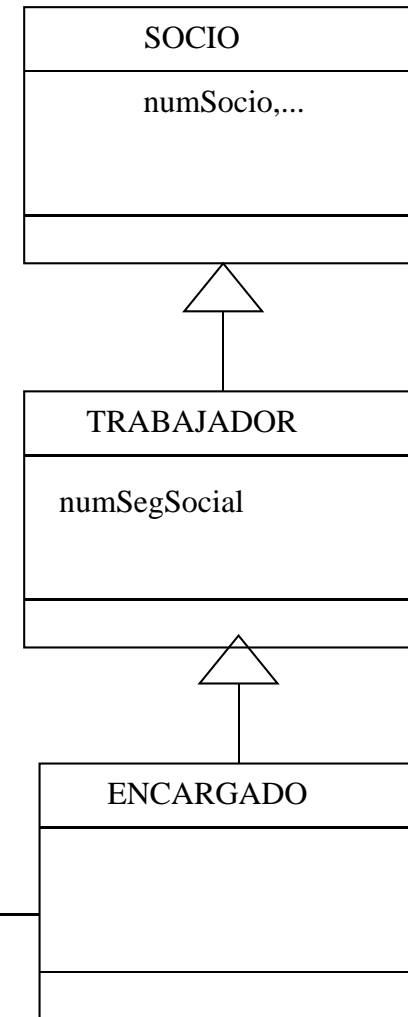
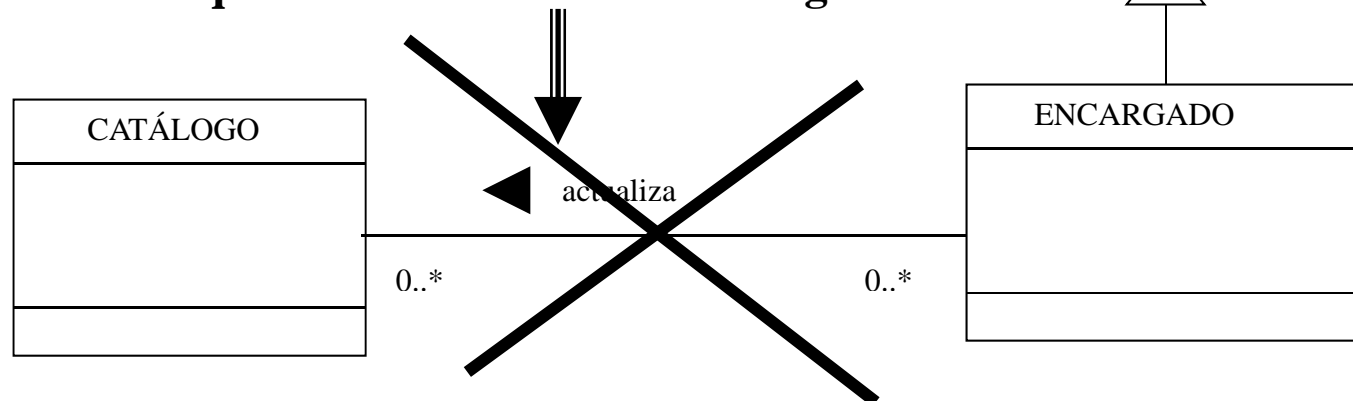
No son casos de uso
- sino funcionalidad interna
-que no puede ser ejecutada de manera independiente por ningún actor

Error típico: Confundir una relación Actor/CU con asociación entre clases

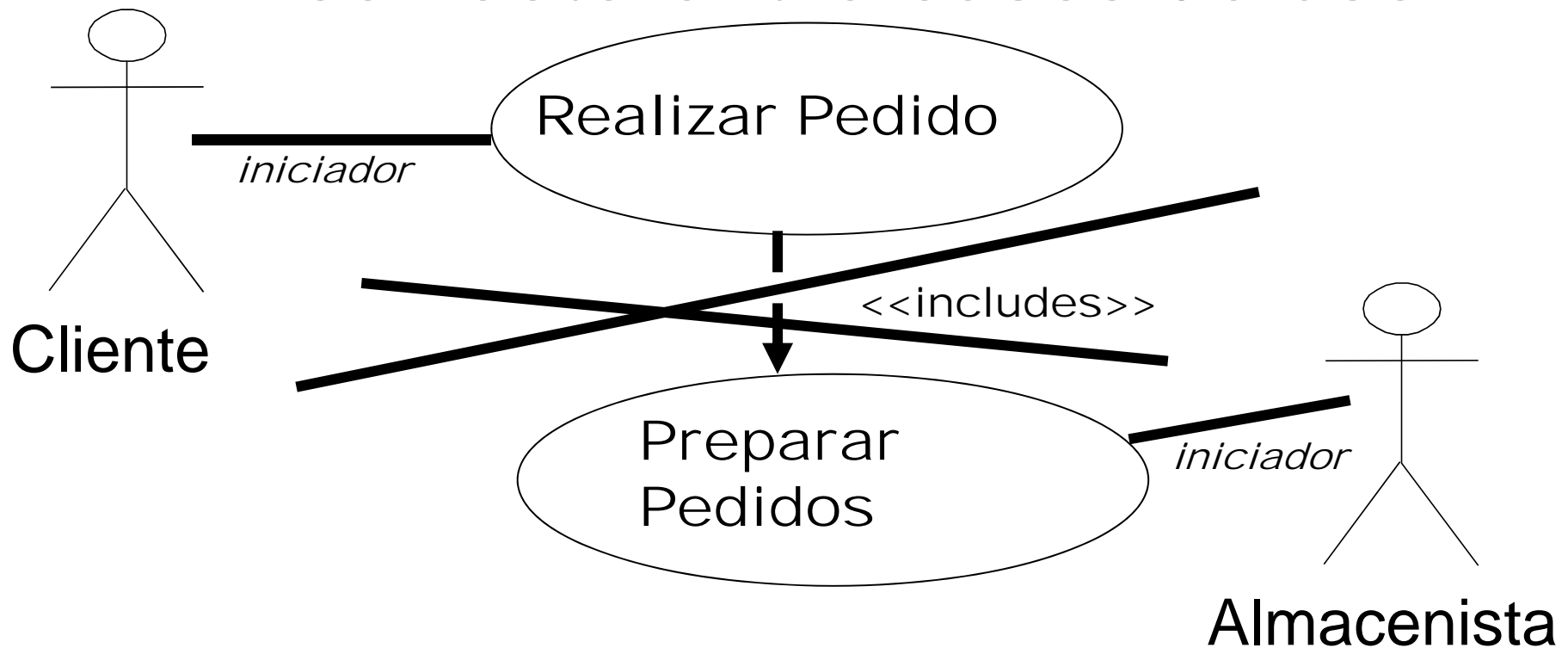


SIGNIFICA: El actor EncargadoBiblio es el que DECIDE ejecutar el caso de uso Actualizar Catálogo

SIGNIFICA: SE DESEA ALMACENAR información sobre quién ha actualizado el catálogo



Errores típicos en CU: relación incorrecta entre casos de uso



AMBOS son C.U. Después de realizar un pedido, el almacenista lo preparará para enviarlo al cliente, PERO NO SE HACE EN EL MOMENTO DE REALIZAR EL PEDIDO (el <<includes>> ESTÁ MAL).

El almacenista decidirá ejecutar el CU "Preparar Pedidos"

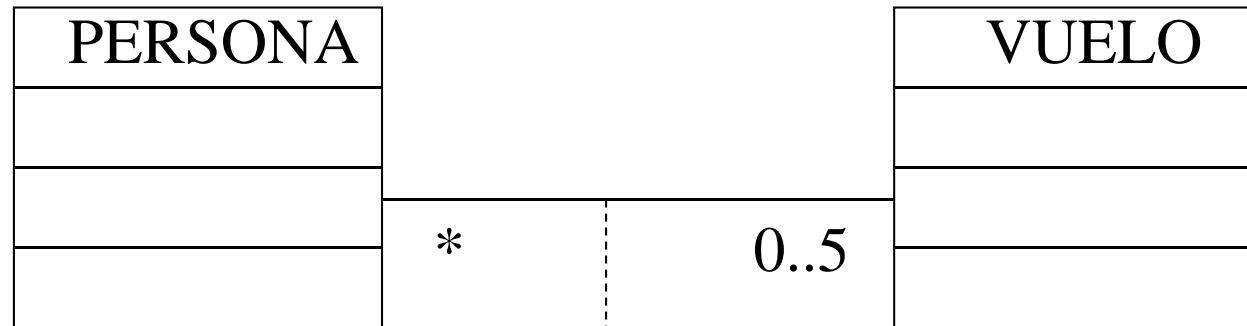
Flujo de Eventos de "Realizar Pedido"

→ registrará el pedido

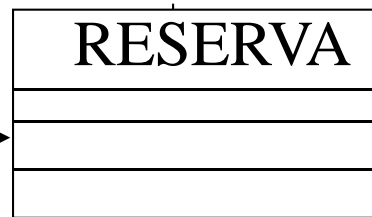
F.Eventos de "Preparar Pedidos"

→ obtendrá pedidos registrados y todavía no enviados

Error típico: usar clase asociación en vez de clase con asociaciones



En este caso cada objeto de RESERVA está asociado con 1 persona y 1 vuelo

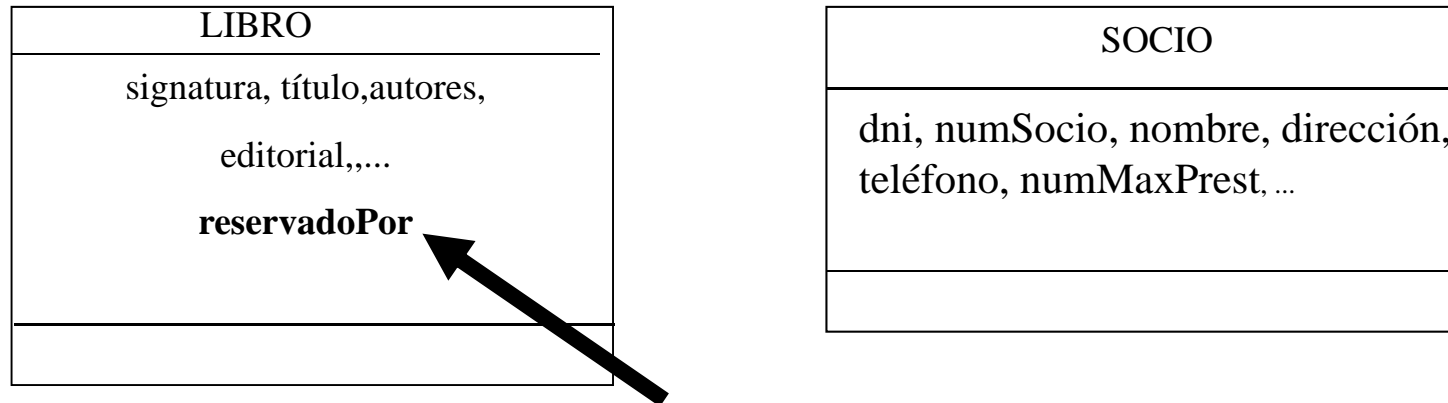


← Clase Asociación

Si se quisiera reflejar que una reserva puede ser para VARIAS personas y que lo es de 0 a 5 vuelos habría que modelarlo así:

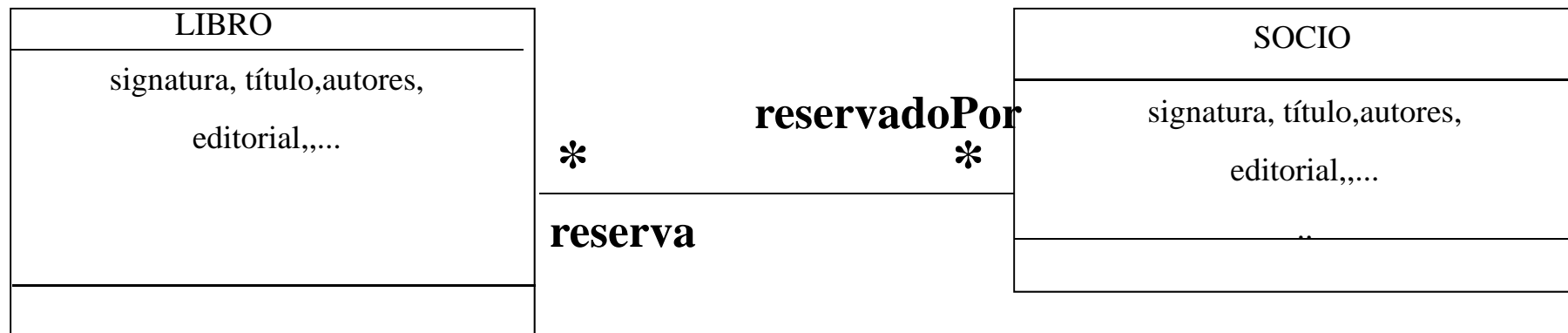


Error típico: usar atributos en vez de asociaciones

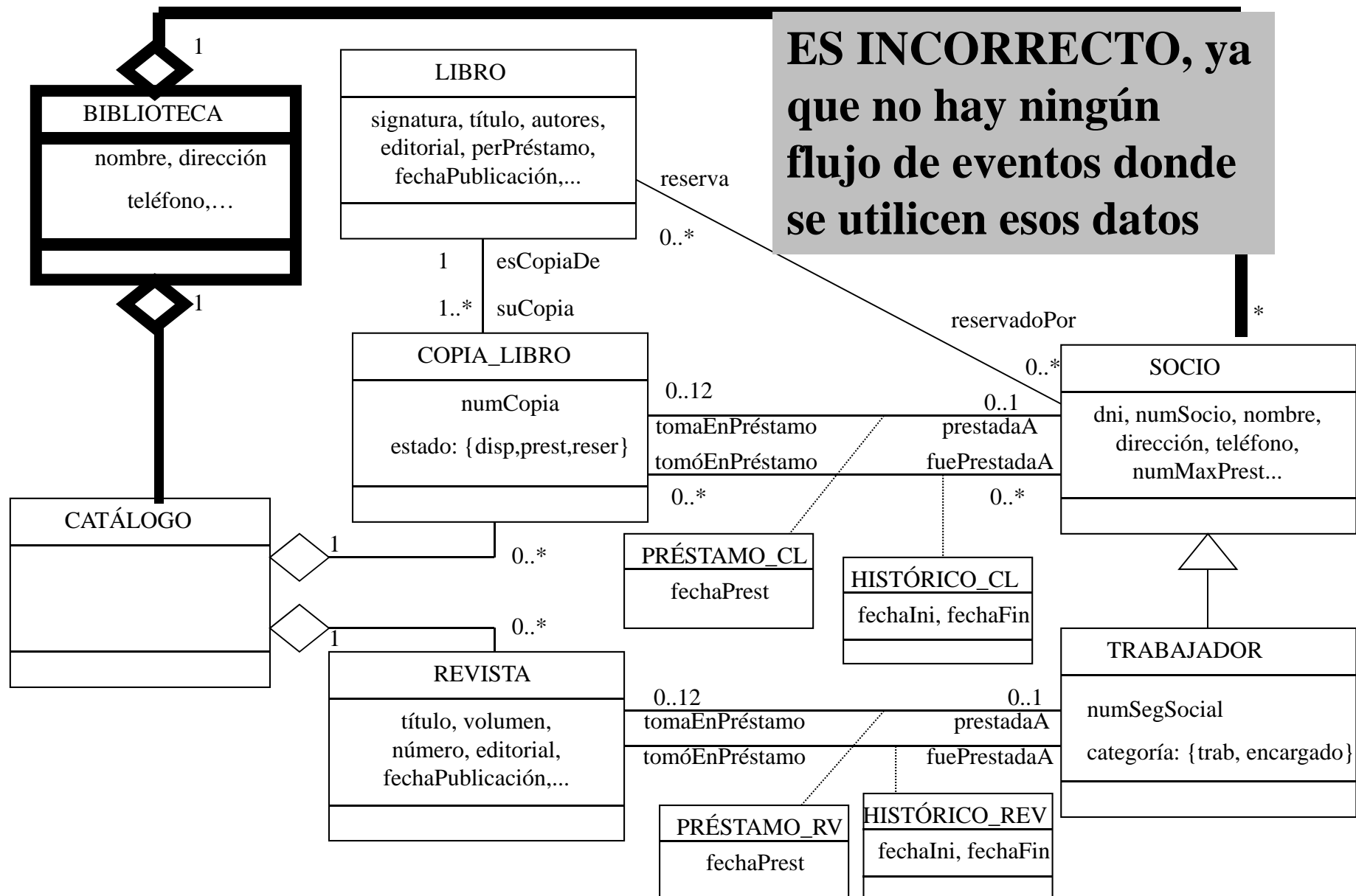


El valor de "reservadoPor" será un socio o lista de socios...
PERO, ya tenemos la clase SOCIO en el MODELO DEL DOMINIO

HAY QUE HACERLO USANDO ASOCIACIÓN

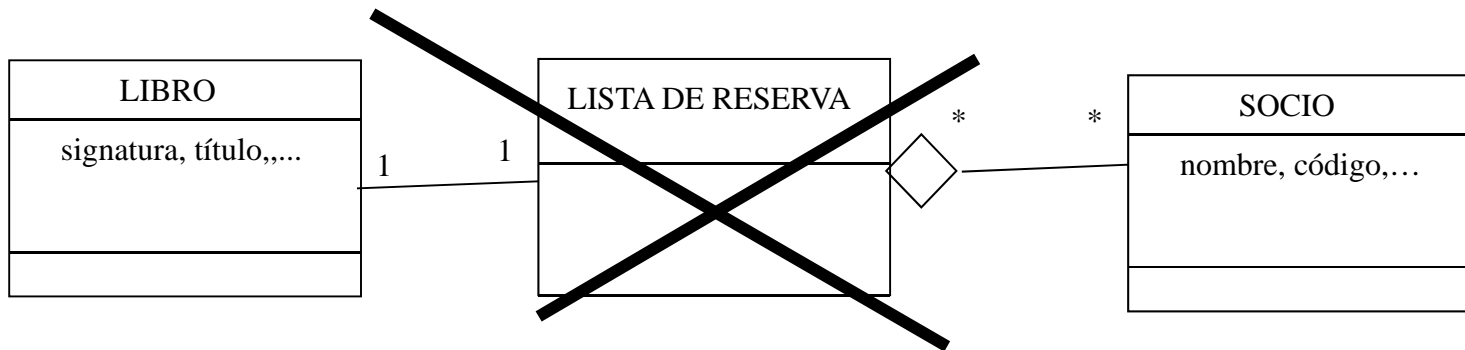


Error típico: añadir elementos no necesarios según Flujo de Eventos

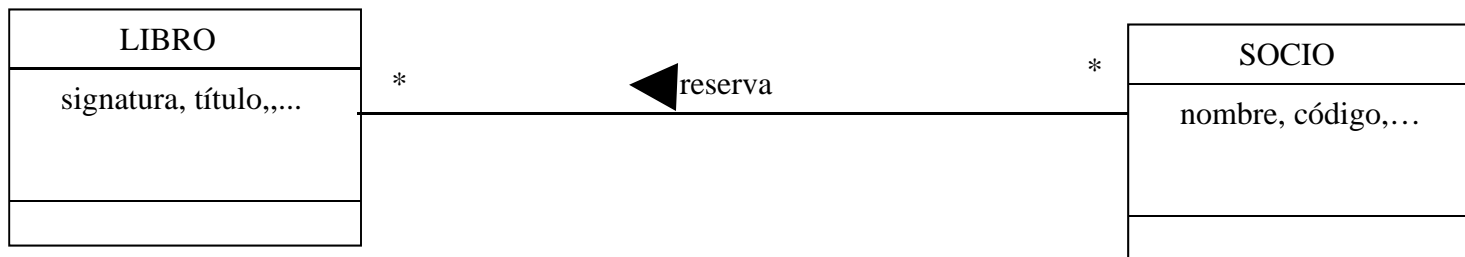


Error típico: añadir clases que no son conceptuales (o que son más de diseño/implementación)

Requisitos: Los socios podrán reservar libros. Por cada libro, habrá una lista de reservas con los socios interesados.



PERO ESO SE MODELA MUY BIEN ASÍ:



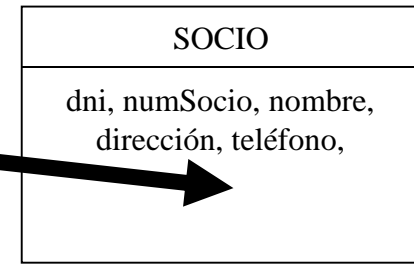
En el modelo del dominio no hay por qué decidir si usaremos una LISTA, o una TABLA HASH, o lo que sea. Ya se decidirá durante el DISEÑO y/o IMPLEMENTACIÓN

Error típico: falta de coherencia entre MCU y MD

Hay datos en los FLUJOS DE EVENTOS que no existen en el Modelo del Dominio.

F.Eventos de "Tomar Préstamo Copia Libro"
-...
- se comprueba que el socio no se pasa del núm. máximo de préstamos

NO EXISTE:
numMaxPrest...



Hay clases, atributos o asociaciones que no se usan en los FLUJOS de EVENTOS.

F.Eventos 1 ...
...
F.Eventos 2 ...
...

En ningún sitio de los
Flujos de Eventos
SE UTILIZA EL EMAIL

