

# **Metodología para el Análisis de Requisitos de Sistemas Software**

**Versión 2.2**

**Amador Durán Toro**

**Beatriz Bernárdez Jiménez**



---

**UNIVERSIDAD  
de SEVILLA**

**Departamento de Lenguajes y Sistemas Informáticos**

**Escuela Técnica Superior de Ingeniería Informática**

**Sevilla, diciembre de 2001**

Este trabajo ha sido o está siendo financiado por los siguientes proyectos:

Proyecto CICYT "MENHIR"(TIC 97-0593-C05-03)

Proyecto CICYT "GEOZOCO"(TIC 2000-1106-C02-01)

Proyecto CYTED "WEST"(VII.18)



## Lista de cambios

Núm.	Fecha	Descripción	Autor/es
0	14/12/1998	Versión 1.0	A. Durán y B. Bernárdez
1	29/11/1999	<p>En la versión 2.0 se han introducido numerosos cambios, los principales son los siguientes:</p> <p>El título cambia de <i>Norma para el Análisis de Requisitos de un Sistema Software</i> a <i>Metodología para el Análisis de Requisitos de Sistemas Software</i></p> <p>La primera sección del documento pasa de denominarse <i>Objetivo y alcance</i> a <i>Objetivo de la metodología</i></p> <p>La tarea 3 pasa a denominarse <i>Desarrollar prototipos</i> en lugar de <i>Desarrollar y validar prototipos</i>. Los aspectos de validación se contemplarán en una metodología aparte</p> <p>Se ha cambiado el contenido de las secciones <i>Introducción</i> y <i>Objetivos del sistema</i> del DAS</p> <p>Se ha contemplado la posibilidad de no incluir los requisitos no funcionales en la matriz de rastreabilidad</p> <p>Se han eliminado las referencias a las operaciones de clases y asociaciones para pasar a una especificación más declarativa</p> <p>Se han añadido los campos correspondientes a la plantilla de escenario para poder especificar las pre y postcondiciones en lenguaje natural además de hacerlo en OCL</p> <p>Se ha reescrito la mayor parte del ejemplo de aplicación de la metodología</p>	A. Durán
3	29/11/1999	Versión 2.0	A. Durán
4	27/11/2000	En la versión 2.1 se han corregido algunos errores ortográficos. Queda pendiente la sincronización con la versión realizada en la tesis doctoral de A. Durán [Durán 2000].	A. Durán
5	27/11/2000	Versión 2.1	A. Durán
6	17/12/2001	Adaptación a la propuesta realizada en la tesis doctoral de A. Durán [Durán 2000]	A. Durán
7	17/12/2001	Versión 2.2	A. Durán



# Índice General

<b>1</b>	<b>Objetivo de la metodología</b>	<b>1</b>
<b>2</b>	<b>Tareas recomendadas</b>	<b>1</b>
2.1	Tarea 1: Desarrollar el modelo estático del sistema . . . . .	2
2.1.1	Objetivos . . . . .	2
2.1.2	Descripción . . . . .	2
2.1.3	Productos internos . . . . .	2
2.1.4	Productos entregables . . . . .	2
2.1.5	Técnicas . . . . .	2
2.2	Tarea 2: Desarrollar el modelo de comportamiento del sistema	3
2.2.1	Objetivos . . . . .	3
2.2.2	Descripción . . . . .	3
2.2.3	Productos internos . . . . .	3
2.2.4	Productos entregables . . . . .	3
2.2.5	Técnicas . . . . .	3
2.3	Tarea 3: Desarrollar prototipos . . . . .	4
2.3.1	Objetivos . . . . .	4
2.3.2	Descripción . . . . .	4
2.3.3	Productos internos . . . . .	4
2.3.4	Productos entregables . . . . .	4
2.3.5	Técnicas . . . . .	4
<b>3</b>	<b>Productos entregables</b>	<b>5</b>
3.1	Documento de análisis del sistema . . . . .	5
3.1.1	Portada . . . . .	5
3.1.2	Lista de cambios . . . . .	5
3.1.3	Índice . . . . .	8
3.1.4	Listas de figuras y tablas . . . . .	8
3.1.5	Introducción . . . . .	8

3.1.6	Modelo estático del sistema . . . . .	8
3.1.7	Diagramas de tipos . . . . .	9
3.1.8	Tipo <i>X</i> . . . . .	9
3.1.9	Asociación <i>Y</i> entre <i>A, B, ..., C</i> . . . . .	9
3.1.10	Modelo de comportamiento del sistema . . . . .	10
3.1.11	Operación del sistema <i>Z</i> . . . . .	10
3.1.12	Matriz de rastreabilidad . . . . .	10
3.1.13	Apéndices . . . . .	11
<b>4</b>	<b>Técnicas</b> . . . . .	<b>11</b>
4.1	Diagrama de tipos . . . . .	11
4.2	Plantillas para especificación de tipos y asociaciones . . . . .	11
4.2.1	Plantilla de tipo de objetos . . . . .	12
4.2.2	Plantillas de atributos . . . . .	13
4.2.3	Plantilla de expresión de invariante . . . . .	14
4.2.4	Plantilla de asociación . . . . .	15
4.2.5	Plantilla de rol . . . . .	16
4.3	Diagrama de estados . . . . .	17
4.4	Plantilla para especificación de operaciones de sistema . . . . .	17
4.5	Diagrama de traza de eventos . . . . .	23
4.6	Construcción de prototipos . . . . .	23
<b>A</b>	<b>Ejemplo: gestión de un vídeo-club</b> . . . . .	<b>25</b>
A.1	Diagramas de tipos . . . . .	25
A.2	Tipos y asociaciones . . . . .	28
A.2.1	Subsistema Películas y Cintas . . . . .	28
A.2.2	Tipo Actor . . . . .	28
A.2.3	Tipo ArtistaCine . . . . .	29
A.2.4	Tipo Cinta . . . . .	30
A.2.5	Tipo Director . . . . .	32
A.2.6	Tipo Película . . . . .	33

A.2.7	Tipo Productora . . . . .	35
A.2.8	Asociación actúaEn entre Actor y Película . . . . .	36
A.2.9	Asociación contiene entre Cinta y Película . . . . .	37
A.2.10	Asociación dirige entre Director y Película . . . . .	38
A.2.11	Asociación produce entre Productora y Película . . . . .	39
A.2.12	Asociación tieneAlquilada entre Película y Cinta . . . . .	40
A.2.13	Asociación tieneDisponible entre Película y Cinta . . . . .	41
A.2.14	Subsistema Socios y Alquileres . . . . .	42
A.2.15	Tipo Alquiler . . . . .	42
A.2.16	Tipo Cargo . . . . .	45
A.2.17	Tipo CargoAlquiler . . . . .	46
A.2.18	Tipo CargoMulta . . . . .	47
A.2.19	Tipo Cuenta . . . . .	48
A.2.20	Tipo Ingreso . . . . .	49
A.2.21	Tipo Movimiento . . . . .	50
A.2.22	Tipo Socio . . . . .	51
A.2.23	Asociación corresponde entre CargoAlquiler y Alquiler . . . . .	54
A.2.24	Asociación esObjetoDe entre Cinta y Alquiler . . . . .	55
A.2.25	Asociación esObjetoActualmenteDe entre Cinta y Alquiler . . . . .	56
A.2.26	Asociación motivadoPor entre CargoMulta y Alquiler . . . . .	57
A.2.27	Asociación realiza entre Socio y Alquiler . . . . .	58
A.2.28	Asociación realizaActualmente entre Socio y Alquiler . . . . .	59
A.2.29	Asociación tiene entre Socio y Cuenta . . . . .	60
A.2.30	Asociación tieneAlquilada entre Socio y Cinta . . . . .	61
A.3	Operaciones del sistema . . . . .	62
A.3.1	Operación AltaDeSocio . . . . .	62
A.3.2	Operación AlquilarCinta . . . . .	65
A.3.3	Operación ConsultarPelícula . . . . .	68
A.4	Conflictos detectados en los requisitos-C . . . . .	70

## Índice de Figuras

1	Estructura del Documento de Análisis del Sistema . . . . .	6
2	Portada del Documento de Análisis del Sistema . . . . .	7
3	Lista de cambios del Documento de Análisis del Sistema . . . . .	7
4	Matriz de rastreabilidad requisitos/elementos de modelado . . . . .	11
5	Plantilla de descripción de tipo de objetos . . . . .	12
6	Plantillas de descripción de atributos . . . . .	14
7	Plantilla de descripción de expresión de invariante . . . . .	14
8	Plantillas de descripción de asociación . . . . .	15
9	Plantilla de descripción de rol . . . . .	16
10	Esquema para la especificación de operaciones de actualización del sistema (notación Catalysis) . . . . .	18
11	Esquema para la especificación de operaciones de consulta al sistema (notación Catalysis) . . . . .	18
12	Diagrama de tipos para el ejemplo de consulta en OCL . . . . .	18
13	Plantilla de descripción de operación de sistema . . . . .	21
14	Diagrama de <i>subsistemas</i> . . . . .	26
15	Diagrama de tipos del subsistema <i>Películas y Cintas</i> . . . . .	26
16	Diagrama de tipos del subsistema <i>Alquileres y Socios</i> . . . . .	27
17	Diagrama de secuencia de <i>AltaDeSocio</i> . . . . .	63
18	Interfaz de usuario de <i>AltaDeSocio</i> . . . . .	64
19	Diagrama de secuencia de <i>AlquilarCinta</i> . . . . .	66
20	Interfaz de usuario de <i>AlquilarCinta</i> . . . . .	67
21	Interfaz de usuario del operación <i>ConsultarPelícula</i> . . . . .	69

# 1 Objetivo de la metodología

El objetivo de esta metodología es la definición de las tareas a realizar, los productos a obtener y las técnicas a emplear durante la actividad de *análisis de requisitos* de la fase de *ingeniería de requisitos* del ciclo de vida de la *ingeniería del software*.

En esta metodología se distinguen dos tipos de productos: los productos *entregables* y los productos *no entregables* o *internos*. Los productos entregables son aquellos que se entregan oficialmente al cliente como parte del desarrollo en fechas previamente acordadas, mientras que los no entregables son productos internos al desarrollo que no se entregan al cliente.

El único producto entregable definido en esta metodología es el *Documento de Análisis del Sistema* (DAS), definido en la sección 3.1, pág. 5. En el caso de que se desarrolle un prototipo, éste puede considerarse producto entregable o no, en función de los acuerdos alcanzados previamente con el cliente.

La estructura de este documento es la siguiente: en la sección 2 se describen las tareas recomendadas para obtener el DAS, en la sección 3 se definen los productos entregables, en este caso el DAS, y por último, en la sección 4 se describen las técnicas recomendadas para obtener los productos. También se incluye como apéndice un ejemplo de aplicación de esta metodología.

## 2 Tareas recomendadas

Las tareas recomendadas para obtener los productos descritos en esta metodología son las siguientes:

**Tarea 1:** Desarrollar el modelo estático del sistema

**Tarea 2:** Desarrollar el modelo de comportamiento del sistema

**Tarea 3:** Desarrollar prototipos

El orden recomendado de realización para estas tareas es: 1, 2 y 3, aunque las tareas 1 y 2 pueden realizarse simultáneamente y el tipo de sistema puede determinar un orden u otro. La tarea 3 es opcional, dependiendo del ciclo de vida adoptado en el desarrollo.

## **2.1 Tarea 1: Desarrollar el modelo estático del sistema**

### **2.1.1 Objetivos**

- Detectar posibles conflictos en los requisitos de información, tanto en los de almacenamiento como en los de restricciones
- Profundizar en el conocimiento de los requisitos

### **2.1.2 Descripción**

En esta tarea se debe obtener el modelo estático del sistema a partir de los requisitos de información obtenidos en la actividad de elicitación. Se deben identificar los tipos o clases de objetos que forman el estado del sistema, así como sus asociaciones, composiciones y clasificaciones.

### **2.1.3 Productos internos**

No hay productos internos en esta tarea.

### **2.1.4 Productos entregables**

- Modelo estático del sistema como parte del DAS (ver sección 3.1.6, pág. 8)
- Conflictos detectados

### **2.1.5 Técnicas**

- Diagrama de tipos (ver sección 4.1, pág. 11)
- Plantillas para tipos de objetos (ver sección 4.2.1, pág. 12)
- Plantillas para asociaciones entre tipos (ver sección 4.2.4, pág. 15)

## **2.2 Tarea 2: Desarrollar el modelo de comportamiento del sistema**

### **2.2.1 Objetivos**

- Detectar posibles conflictos en los requisitos funcionales, tanto en los expresados de forma tradicional como en los expresados mediante casos de uso
- Profundizar en el conocimiento de los requisitos

### **2.2.2 Descripción**

En esta tarea se debe obtener el modelo de comportamiento del sistema a partir de los requisitos funcionales, tanto los expresados de forma tradicional como los expresados mediante casos de uso, obtenidos en la actividad de elicitación. Se deben especificar las operaciones del sistema y los estados y transiciones de las tipos de objetos identificados en la tarea anterior que tengan un dinamismo significativo.

### **2.2.3 Productos internos**

No hay productos internos en esta tarea.

### **2.2.4 Productos entregables**

- Modelo de comportamiento como parte del DAS (ver sección 3.1.10, pág. 10)
- Conflictos detectados durante el análisis

### **2.2.5 Técnicas**

- Plantilla para operaciones del sistema (ver sección 4.4, pág. 17)
- Diagrama de traza de eventos (ver sección 4.5, pág. 23)
- Diagrama de estados (ver sección 4.3, pág. 17)

## 2.3 Tarea 3: Desarrollar prototipos

### 2.3.1 Objetivos

- Obtener requisitos relativos a la interfaz de usuario
- Detectar posibles conflictos en los requisitos
- Profundizar en el conocimiento de los requisitos

### 2.3.2 Descripción

En esta tarea se deben desarrollar prototipos que permitan tanto a los clientes y usuarios como a los desarrolladores tener una idea más clara del sistema a desarrollar e identificar nuevos requisitos o conflictos que hayan permanecido *ocultos* hasta el momento. Lo más habitual es que el prototipo sea *desechable*, es decir, que una vez que se haya utilizado no se desarrolle tomando su código como base.

### 2.3.3 Productos internos

No hay productos internos en esta tarea.

### 2.3.4 Productos entregables

- Prototipo del sistema
- Aspectos de interfaz de usuarios del modelo de comportamiento como parte del DAS (ver sección 3.1.10, pág. 10)
- Conflictos detectados

### 2.3.5 Técnicas

- Prototipado de sistemas software (ver sección 4.6, pág. 23).

## 3 Productos entregables

El único producto entregable que se contempla en esta norma es el *Documento de Análisis del Sistema* (DAS).

### 3.1 Documento de análisis del sistema

La estructura del DAS puede verse en la figura 1. En las siguientes secciones se describe con detalle cada sección del DAS.

#### 3.1.1 Portada

La portada del DAS debe tener el formato que puede verse en la figura 2. Los elementos que deben aparecer son los siguientes:

- **Nombre del proyecto:** el nombre del proyecto al que pertenece el DAS.
- **Versión:** la versión del DAS que se entrega al cliente. La versión se compone de dos números  $X$  e  $Y$ . El primero indica la versión, y se debe incrementar cada vez que se hace una nueva entrega formal al cliente. Cuando se incremente el primer número, el segundo debe volver a comenzar en cero. El segundo número indica cambios dentro de la misma versión aún no entregada, y se debe incrementar cada vez que se imprima una versión con cambios respecto a la última que se imprimió y que no se vaya a entregar formalmente todavía. Este tipo de versiones pueden ser internas al equipo de desarrollo o ser entregadas al cliente a título orientativo.
- **Fecha:** fecha de la publicación de la versión.
- **Equipo de desarrollo:** nombre de la empresa o equipo de desarrollo.
- **Cliente:** nombre del cliente, normalmente otra empresa.

#### 3.1.2 Lista de cambios

El documento debe incluir una lista de cambios en la que se especifiquen, para cada versión del documento, los cambios producidos en el mismo con un formato similar al que puede verse en la figura 3. Para cada cambio

Portada
Lista de cambios
Índice
Lista de figuras
Lista de tablas
<b>1</b> Introducción
<b>2</b> Modelo estático del sistema
<b>2.1</b> Diagramas de tipos
<b>2.2</b> Tipo $X$
<b>2.2.1</b> Descripción del tipo $X$
<b>2.2.2</b> Atributos del tipo $X$
<b>2.2.3</b> Invariante del tipo $X$
<b>2.2.4</b> Diagrama de estados del tipo $X$ [opcional]
<b>2.3</b> Asociación $Y$ entre $A, B, \dots, C$
<b>2.3.1</b> Descripción de la asociación $Y(A, B, \dots, C)$
<b>2.3.2</b> Roles de la asociación $Y(A, B, \dots, C)$
<b>2.3.3</b> Atributos de la asociación $Y(A, B, \dots, C)$ [opcional]
<b>2.3.4</b> Invariante de la asociación $Y(A, B, \dots, C)$ [opcional]
<b>2.3.5</b> Diagrama de estados de la asociación $Y(A, B, \dots, C)$ [opcional]
<b>3</b> Modelo de comportamiento del sistema
<b>3.1</b> Operación $Z$
<b>3.1.1</b> Descripción operación $Z$
<b>3.1.2</b> Diagrama de traza de eventos operación $Z$ [opcional]
<b>3.1.3</b> Interfaz de usuario operación $Z$ [opcional]
<b>4</b> Matrices de rastreabilidad
Apéndices [opcional]

Figura 1: Estructura del Documento de Análisis del Sistema

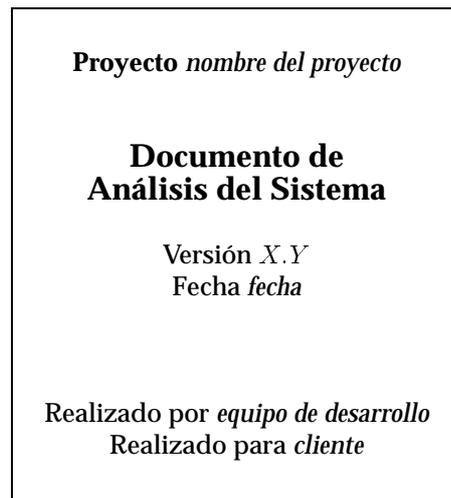


Figura 2: Portada del Documento de Análisis del Sistema

Núm.	Fecha	Descripción	Autor/es
0	<i>fecha<sub>0</sub></i>	Versión x.y	<i>autor<sub>0</sub></i>
1	<i>fecha<sub>1</sub></i>	<i>descripción cambio<sub>1</sub></i>	<i>autor<sub>1</sub></i>
⋮	⋮	⋮	⋮
<i>n</i>	<i>fecha<sub>n</sub></i>	<i>descripción cambio<sub>n</sub></i>	<i>autor<sub>n</sub></i>

Figura 3: Lista de cambios del Documento de Análisis del Sistema

realizado se debe incluir el número de orden, la fecha, una descripción y los autores.

### **3.1.3 Índice**

El índice del DAS debe indicar la página en la que comienza cada sección, subsección o apartado del documento. En la medida de lo posible, se sangrarán las entradas del índice para ayudar a comprender la estructura del documento.

### **3.1.4 Listas de figuras y tablas**

El DAS deberá incluir listas de las figuras y tablas que aparezcan en el mismo. Dichas listas serán dos índices que indicarán el número, la descripción y la página en que aparece cada figura o tabla del DAS. El formato para dichas listas será similar al empleado en este mismo documento.

### **3.1.5 Introducción**

Esta sección debe contener una descripción breve de las principales características del sistema software que se va a desarrollar, la situación actual que genera la necesidad del nuevo desarrollo, la problemática que se acomete, y cualquier otra consideración que sitúe al posible lector en el contexto oportuno para comprender el resto del documento.

### **3.1.6 Modelo estático del sistema**

Esta sección se divide en los siguientes apartados en los que se describe el modelo estático del sistema, es decir, los tipos de objetos y las asociaciones entre tipos de objetos que componen el estado del sistema y que se han identificado durante el análisis. Si en los diagramas de tipos se han agrupado los tipos y asociaciones en paquetes, el modelo estático deberá reflejar también dicha división. Tanto los tipos como las asociaciones deben estar ordenadas alfabéticamente por su nombre dentro de cada paquete.

### 3.1.7 Diagramas de tipos

Este apartado debe contener los diagramas de tipos (ver sección 4.1, pág. 11) del modelo estático del sistema construido durante el análisis. Para sistemas complejos puede ser conveniente agrupar los diagramas en *paquetes*, tal como se describe en [Booch *et al.* 1999].

### 3.1.8 Tipo *X*

Este apartado, que se repite para cada tipo *X* identificada en el análisis, debe tener el siguiente contenido:

- **Descripción:** descripción del tipo mediante la plantilla de tipo (ver sección 4.2.1, pág. 12).
- **Atributos:** descripción, ordenada alfabéticamente, de los atributos del tipo mediante las plantillas de atributo (ver sección 4.2.2, pág. 13).
- **Invariante:** descripción de las expresiones de invariante del tipo mediante la plantilla de expresión de invariante (ver sección 4.2.3, pág. 14).
- **Diagrama de estados:** diagrama de estados correspondiente al tipo, si se ha identificado alguno (ver sección 4.3, pág. 17).

### 3.1.9 Asociación *Y* entre *A*, *B*, ..., *C*

Este apartado, que se repite para cada asociación *Y* entre los tipos *A*, *B*, ..., *C* identificada en el análisis, debe tener el siguiente contenido, muy similar al de los tipos:

- **Descripción:** descripción de la asociación mediante la plantilla de asociación (ver sección 4.2.4, pág. 15).
- **Roles:** descripción, ordenada alfabéticamente, de los roles que juegan los tipos participantes en la asociación mediante la plantilla de rol (ver sección 4.2.5, pág. 16).
- El resto del contenido de este apartado es opcional y su descripción coincide con el de la descripción de tipos:

- **Atributos** (si se ha identificado alguno)
- **Invariante**
- **Diagrama de estados** (si es significativo)

### 3.1.10 Modelo de comportamiento del sistema

Esta sección se divide en los siguientes apartados en los que se describen las operaciones del sistema identificadas durante el análisis.

### 3.1.11 Operación del sistema Z

Este apartado, que se repite para cada operación del sistema Z identificada en el análisis, debe tener el siguiente contenido:

- **Descripción:** descripción de la operación del sistema mediante la plantilla de operación del sistema (ver sección 4.4, pág. 17).
- **Diagrama de traza de eventos:** diagrama de traza de eventos correspondiente a la operación del sistema, si se considera oportuno (ver sección 4.5, pág. 23).
- **Interfaz de usuario:** interfaz de usuario asociada a la operación del sistema, en el caso de que se haya construido y validado algún prototipo (ver sección 4.6, pág. 23). Se deberán mostrar la/s pantalla/s y/o informe/s asociados a la operación del sistema así como su descripción.

### 3.1.12 Matriz de rastreabilidad

Este apartado debe contener una o más matrices *requisitos-elementos de modelado*, de forma que para cada requisito identificado en el *Documento de Requisitos del Sistema* se pueda conocer con qué tipos, asociaciones y operaciones del sistema está asociado. El formato de las matrices de rastreabilidad puede verse en la figura 4.

Dado que los requisitos no funcionales no suelen recogerse en el modelo del sistema, si se considera oportuno pueden no incluirse en las matrices de rastreabilidad.

	RI-01	RI-02	...	UC-01	UC-02	...	RF-01	RF-02	...
Tipo $X_1$	•	•						•	
Tipo $X_2$		•							
...									
Asoc. $Y_1$	•							•	
Asoc. $Y_2$	•	•							
...									
Op. $Z_1$				•			•		
Op. $Z_2$					•		•		
...									

Figura 4: Matriz de rastreabilidad requisitos/elementos de modelado

### 3.1.13 Apéndices

Los apéndices se usarán para proporcionar información adicional a la documentación obligatoria del documento. Sólo deben aparecer si se consideran oportunos y se identificarán con letras ordenadas alfabéticamente: A, B, C, etc.

## 4 Técnicas

A continuación, se describen algunas de las técnicas que se proponen en esta metodología para obtener los productos de las tareas que se han descrito.

### 4.1 Diagrama de tipos

Los diagramas de tipos permiten expresar gráficamente aspectos estáticos del modelo de un sistema. En esta metodología se propone que los diagramas de tipos usen la notación UML [Booch *et al.* 1999].

### 4.2 Plantillas para especificación de tipos y asociaciones

En la propuesta metodológica se recomienda el siguiente conjunto de plantillas para especificar los tipos de objetos, y sus asociaciones, identificados en el análisis de los requisitos de información:

- **Plantilla de tipo:** una por cada tipo de objetos.

- **Plantilla de atributo:** una por cada atributo de cada tipo de objetos o de cada asociación. Hay tres versiones distintas de esta plantilla, según el atributo sea constante, variable o derivado.
- **Plantilla de expresión de invariante:** una por cada expresión que forme la invariante de cada tipo de objetos o de cada asociación.
- **Plantilla de asociación:** una por cada asociación entre tipos de objetos.
- **Plantilla de rol:** una por cada rol de cada asociación (al menos dos).

En las siguientes secciones se describen cada una de estas plantillas. Todas ellas tienen un apartado de *comentarios* cuya descripción es la siguiente:

- **Comentarios:** en este apartado se deben recoger otras consideraciones que no se hayan especificado en otros apartados de la plantilla.

#### 4.2.1 Plantilla de tipo de objetos

El formato de la plantilla para describir los tipos de objetos puede verse en la figura 5. Los apartados específicos de esta plantilla son los siguientes:

<b>Tipo</b>	<nombre tipo>		
<b>Versión</b>	<nº de la versión actual> (<fecha de la versión actual>)		
<b>Autores</b>	• <autor de la versión actual> (<organización del autor>) ...		
<b>Requisitos asociados</b>	• RI-x <nombre del requisito> ...		
<b>Descripción</b>	Este tipo { abstracto, concreto } representa <concepto que representa el tipo>		
<b>Supertipos</b>	<supertipos directos del tipo>		
<b>Subtipos</b>	<subtipos directos del tipo> ({ disjuntos, solapados })		
<b>Componentes</b>	<b>Nombre</b>	<b>Tipo OCL</b>	<b>Mult.</b>
	<nombre componente>	<tipo OCL del enlace>	mult.
	...	...	...
<b>Comentarios</b>	<comentarios adicionales sobre el tipo>		

Figura 5: Plantilla de descripción de tipo de objetos

- **Nombre tipo:** cada tipo se identifica por un nombre, que deberá ser único en el modelo y que se recomienda que sea un sustantivo o similar en singular.

- **Descripción:** en este apartado se indica si el tipo es abstracto o concreto y se describe el concepto que representa. En este contexto, se entenderá que un tipo es abstracto si los conjuntos de instancias o *extensiones* de sus subtipos forman una clasificación *completa* [Booch *et al.* 1999], es decir, todas las instancias de la extensión del tipo que se está describiendo deben ser también instancias del algún subtipo.
- **Requisitos asociados:** en este apartado se indican los requisitos del *Documento de Requisitos del Sistema* (DRS) a los que está asociado el tipo, es decir, aquellos requisitos que justifican su existencia.
- **Supertipos:** en este apartado se indican los supertipos directos del tipo.
- **Subtipos:** en este apartado se indican los subtipos directos del tipo, especificando si son disjuntos o solapados.
- **Componentes:** en este apartado se indican los componentes del tipo, es decir aquellos tipos *agregados mediante composición* por el tipo. Para cada componente se incluyen los siguientes apartados:
  - **Nombre:** nombre del enlace por el que el objeto agregado accede a sus componentes.
  - **Tipo OCL:** tipo OCL del enlace por el que el objeto agregado accede a sus componentes.
  - **Multiplicidad:** multiplicidad de la agregación.

#### 4.2.2 Plantillas de atributos

Dentro de los atributos de un tipo se distingue entre atributos constantes, variables y derivados. Los formatos de las plantillas para describir estos tres tipos de atributos pueden verse en la figura 6. Los apartados específicos de estas tres plantillas son los siguientes:

- **Nombre atributo:** cada atributo se identifica por un nombre que deberá ser único dentro de un tipo, recomendándose que sea un sustantivo o similar.
- **Descripción:** en este apartado se proporciona una breve descripción del atributo.
- **Tipo OCL:** en este apartado se indica el tipo OCL del atributo.

<b>Atributo constante</b>	< <i>nombre tipo</i> > :: < <i>nombre atributo</i> >
<b>Descripción</b>	< <i>descripción del atributo</i> >
<b>Tipo OCL</b>	< tipo OCL del atributo >
<b>Comentarios</b>	< <i>comentarios adicionales sobre el atributo</i> >

<b>Atributo variable</b>	< <i>nombre tipo</i> > :: < <i>nombre atributo</i> >
<b>Descripción</b>	< <i>descripción del atributo</i> >
<b>Tipo OCL</b>	< tipo OCL del atributo >
<b>Valor inicial</b>	< valor inicial del atributo > [opcional]
<b>Comentarios</b>	< <i>comentarios adicionales sobre el atributo</i> >

<b>Atributo derivado</b>	< <i>nombre tipo</i> > :: < <i>nombre atributo</i> >
<b>Descripción</b>	< <i>descripción del atributo</i> >
<b>Tipo OCL</b>	< tipo OCL del atributo >
<b>Expresión</b>	< expresión OCL asociada al atributo >
<b>Comentarios</b>	< <i>comentarios adicionales sobre el atributo</i> >

Figura 6: Plantillas de descripción de atributos

- **Valor inicial:** para los atributos variables, en este apartado se indica, opcionalmente, el valor inicial del atributo.
- **Expresión:** para los atributos derivados, en este apartado se indica la expresión OCL asociada a la evaluación del atributo.

#### 4.2.3 Plantilla de expresión de invariante

El formato de la plantilla para describir las expresiones de invariante de los tipos de objetos puede verse en la figura 7. Los apartados específicos de esta plantilla son los siguientes:

<b>Invariante</b>	< <i>nombre tipo</i> > :: < <i>nombre expresión</i> >
<b>Descripción</b>	• < <i>descripción de la expresión de invariante en lenguaje natural</i> >
<b>Expresión</b>	< expresión de la invariante en OCL >
<b>Comentarios</b>	< <i>comentarios adicionales sobre la invariante</i> >

Figura 7: Plantilla de descripción de expresión de invariante

- **Nombre expresión:** cada expresión de invariante se identifica por un nombre único dentro de un tipo. Este nombre podrá usarse posteriormente para simplificar la escritura de expresiones complejas.

- **Descripción:** en este apartado se proporciona una descripción en lenguaje natural de la expresión de invariante para facilitar la comprensión de la expresión OCL del apartado siguiente.
- **Expresión:** en este apartado se proporciona una descripción en OCL de la expresión de invariante, asumiendo el contexto del propio tipo [IBM 1997, pág. 3].

#### 4.2.4 Plantilla de asociación

El formato de la plantilla para describir las asociaciones entre tipos de objetos puede verse en la figura 8. Los apartados específicos de esta plantilla son los siguientes:

<b>Asociación [derivada]</b>	<nombre asociación> entre <participantes>
<b>Versión</b>	<nº de la versión actual> (<fecha de la versión actual>)
<b>Autores</b>	<ul style="list-style-type: none"> <li>• &lt;autor de la versión actual&gt; (&lt;organización del autor&gt;)</li> <li>...</li> </ul>
<b>Requisitos asociados</b>	<ul style="list-style-type: none"> <li>• RI-x &lt;nombre del requisito&gt;</li> <li>...</li> </ul>
<b>Descripción</b>	Esta asociación [derivada] representa <concepto que representa la asociación>
<b>Comentarios</b>	<comentarios adicionales sobre la asociación>

Figura 8: Plantillas de descripción de asociación

- **Nombre asociación y participantes:** cada asociación se identifica por una combinación de nombre de asociación y de nombres de participantes única en el modelo. Se recomienda que el nombre sea una forma verbal o similar, de forma que tenga sentido al construir una frase con el nombre de la asociación y los participantes. Por ejemplo la asociación *trabajaEn* entre *Empleado* y *Empresa*.

Para hacer referencia a la asociación se utilizará el formato:

```
asociación( A, B, ..., C )
```

por ejemplo: `trabajaEn( Empleado, Empresa )`

- **Descripción:** en este apartado se describe el concepto que representa la asociación y si se trata de una asociación derivada o no.

En el caso de las asociaciones derivadas, deberá acompañarse una descripción de su invariante en la que se especifique el criterio para saber qué objetos están asociados. La estructura de esta invariante suele ser la siguiente, donde  $P_1 \dots P_n$  son los tipos de los objetos participantes en la asociación, y  $A$  es la asociación derivada:

```
P1.allInstances->forall( p1 |
  ...
  Pn.allInstances->forall( pn |
    A.allInstances->forall( a |
      ( a.rol1 = p1 and ... and a.roln = pn ) =
      < predicado sobre p1 ... p1 > ) ) ... )
```

teniendo en cuenta que la igualdad entre expresiones booleanas es equivalente a la doble implicación lógica.

#### 4.2.5 Plantilla de rol

El formato de la plantilla para describir los roles de las asociaciones entre tipos de objetos puede verse en la figura 9. Los apartados específicos de esta plantilla son los siguientes:

<b>Rol</b>	<nombre tipo> juega rol <nombre rol> en <nombre asociación>
<b>Descripción</b>	<descripción del rol>
<b>Tipo OCL</b>	<tipo OCL del rol>
<b>Multiplicidad</b>	<multiplicidad del rol>
<b>Comentarios</b>	<comentarios adicionales sobre el rol>

Figura 9: Plantilla de descripción de rol

- **Nombre rol:** cada rol se identifica por una combinación de nombre de asociación y de nombre de rol única en el modelo, siempre que no haya posibilidad de ambigüedades [IBM 1997, pág. 8]. Se recomienda que el nombre sea un sustantivo o similar.
- **Descripción:** en este apartado se proporciona una breve descripción del rol.
- **Tipo OCL:** en este apartado se especifica el tipo OCL del rol.
- **Multiplicidad:** en este apartado se indica la multiplicidad del rol.

### 4.3 Diagrama de estados

Los diagramas de estados permiten expresar aspectos de comportamiento a nivel de un tipo de objetos. En esta metodología se propone que los diagramas de estados se realicen utilizando la notación UML [Booch *et al.* 1999].

### 4.4 Plantilla para especificación de operaciones de sistema

En esta metodología se recomienda utilizar las *acciones conjuntas* de *Catalysis* [D'Souza y Wills 1999] aunque denominándolas *operaciones de sistema* como en *Fusion* [Coleman *et al.* 1994], ya que las acciones conjuntas que se especifican son las que forman la interfaz del sistema con los actores.

Para el estilo de especificación se ha optado por una combinación del utilizado en las especificaciones realizadas en Z [Diller 1990, Ratcliff 1994] y del propuesto en *Catalysis*, en el que la semántica de las operaciones se especifica mediante disyunciones en las que aparecen la conjunción de la precondición y de la postcondición, o bien la conjunción de una expresión de excepción y la postcondición en dicho caso. Esto permite especificar de forma clara la conducta del sistema, tanto en situaciones normales como en situaciones de error.

Para poder especificar la respuesta del sistema, se propone la utilización de dos *pseudovariables*: *respuesta*, de tipo OCL Set (String), donde se le comunican al usuario las respuestas del sistema y, para aquellas operaciones que sean de consulta o que devuelvan algún tipo de resultado, *resultado* (denominada *result* en la documentación original de OCL [IBM 1997]), cuyo tipo depende de la consulta que se realice.

Siguiendo estas ideas, el esquema (en la notación de *Catalysis*) para especificar una operación de sistema que no sea de consulta es el que puede verse en la figura 10, donde  $p_1, \dots, p_q$  son los parámetros de la operación, *pre* es la precondición, *post* es la postcondición,  $excep_1, \dots, excep_n$  son las condiciones de excepción y la respuesta del sistema se devuelve en *respuesta*.

Este esquema asume que la disyunción de la precondición y de las condiciones de excepción es siempre cierta y que la precondición no puede ser cierta si alguna condición de excepción también lo es. Es decir:

```
pre xor ( excep1 or ... or excepn )
```

```

acción Operación( p1, ... pq )
post: ( pre and post and respuesta = Set{ "<éxito>" } )
      or
      ( excep1 and respuesta->includes( "<error n° 1>" ) )
      or
      ...
      or
      ( excepn and respuesta->includes( "<error n° n>" ) )

```

Figura 10: Esquema para la especificación de operaciones de actualización del sistema (notación Catalysis)

```

acción Consulta( p1, ... pq ) : <tipo del resultado>
post: ( pre and
      resultado = <resultado de la consulta> and
      respuesta = Set{ "<éxito>" } )
      or
      ( excep1 and respuesta->includes( "<error n° 1>" ) )
      or
      ...
      or
      ( excepn and respuesta->includes( "<error n° n>" ) )

```

Figura 11: Esquema para la especificación de operaciones de consulta al sistema (notación Catalysis)

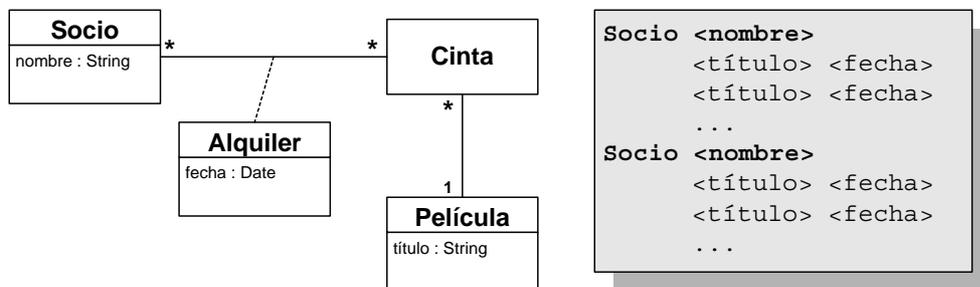


Figura 12: Diagrama de tipos para el ejemplo de consulta en OCL

En el caso de una operación de consulta (ver figura 11), el esquema es similar, aunque en este caso ha sido necesario ampliar OCL para que permita la utilización de *tipos sin nombre* para poder especificar los resultados de las consultas, que se devuelven usando `resultado`.

Hay que tener en cuenta que el objetivo del análisis respecto a las consultas es asegurar que las consultas pueden realizarse, y la expresividad de un lenguaje como OCL no siempre permite especificar una consulta de forma completa. En nuestra opinión, es necesario ampliar OCL para permitir la definición de tipos sin nombre que puedan utilizarse para especificar los resultados de las consultas.

Si después de analizar un caso de uso en el que se especifica qué información debe devolver el sistema como respuesta a una consulta, se puede comprobar que el modelo estático contempla dicha información y que dicha información es accesible, el objetivo del análisis está cubierto. Para ello, en [D'Souza y Wills 1999, pág. 598] se propone realizar diagramas de instancias (denominados *snapshots* o *instantáneas* en Catalysis) representando la información a obtener del sistema.

Por ejemplo, supongamos un modelo sencillo como el de la figura 12 para el que es necesario especificar una operación de consulta que genere un listado de todos los socios, indicando para cada socio los alquileres realizados. El listado debe estar ordenado por nombre de socio y por fecha de alquiler.

Para expresar esta consulta en OCL es necesario indicar que el resultado será de tipo `Sequence( String x Sequence( String x Date ) )`, que siguiendo estrictamente [IBM 1997], sólo podría expresarse incluyendo en el modelo los tipos necesarios para ello. En el caso de las consultas, esto implicaría tener que definir al menos un tipo de objetos por cada consulta que se deseara especificar.

Como ya se ha comentado, en esta metodología se propone ampliar OCL para permitir formar *tipos sin nombre*, de forma que se permitan expresiones de la forma  $( exp_1, \dots, exp_n )$ , cuyo tipo sería  $(T_1 \times \dots \times T_n)$ , siendo cada  $T_i$  el tipo correspondiente a cada  $exp_i$ .

Además de permitir tipos sin nombre, se ha considerado también la posibilidad de poder nombrar los distintos elementos de dichos tipos, de forma similar a los esquemas en Z. De esta forma, el tipo del resultado de la consulta de la figura 12 se expresaría como:

```
Sequence( nombre:String x alquileres:Sequence( título:String x fecha:Date ) )
```

Con lo que la consulta podría especificarse de la siguiente forma:

```
acción ObtenerListadoSociosAlquileres() :
Sequence( nombre:String x alquileres:Sequence( título:String x fecha:Date ) )
post: resultado->isOrderedBy( <=, nombre ) and
resultado->forAll( r | r.alquileres->isOrderedBy( <=, fecha ) ) and
resultado->collect( nombre ) = Socio.allInstances->collect( nombre ) and
resultado->forAll( r | Socio.allInstances->exists( s |
s.nombre = r.nombre and
r.alquileres->collect( título, fecha ) =
s.alquiler->collect( cinta.película.título, fecha ) ) ) and
respuesta = Set{"Consulta realizada con éxito"}
```

donde isOrderedBy es una *macro* definida sobre secuencias de la forma:

$$\frac{\text{Sequence}( T )::\text{isOrderedBy}( op, exp ) : \text{Boolean}}{\text{resultado} = \text{Sequence}\{1..\text{self}\rightarrow\text{size} - 1\}\rightarrow\text{forAll}( i |$$

$$\text{self}\rightarrow\text{at}( i ).exp \text{ op } \text{self}\rightarrow\text{at}( i + 1 ).exp )$$

y se ha permitido que la operación collect sobre colecciones pueda manejar también tipos sin nombre.

Si se hubiera querido restringir de alguna forma el resultado filtrando los socios o los alquileres, podría haberse hecho mediante el uso de la operación *select*, definida sobre colecciones.

La forma de expresar este estilo de especificación de operaciones es mediante la plantilla que puede verse en la figura 13, cuyos apartados específicos son los siguientes:

- **Nombre operación:** cada operación de sistema se identifica por un nombre único en el modelo que se recomienda que sea una forma verbal o similar.
- **Tipo del resultado:** en el caso de que la operación devuelva algún tipo de resultado, es necesario especificarlo en este apartado opcional.
- **Descripción:** en este apartado se proporciona una breve descripción de la operación en lenguaje natural.
- **Requisitos asociados:** en este apartado se indican los requisitos del *Documento de Requisitos del Sistema* (DRS) a los que está asociado la operación del sistema, es decir, aquellos requisitos que justifican su existencia, normalmente requisitos funcionales expresados como casos de uso.

<b>Operación Sistema</b>	< <b>nombre operación</b> >
<b>Tipo resultado</b>	tipo del resultado [opcional]
<b>Versión</b>	< n <sup>o</sup> de la versión actual > (< fecha de la versión actual >)
<b>Autores</b>	• < autor de la versión actual > (< organización del autor >) ...
<b>Requisitos asociados</b>	• RF-x < nombre del requisito > ...
<b>Descripción</b>	< descripción de la operación en lenguaje natural >
<b>Parámetros</b>	p <sub>1</sub> : Tipo <sub>1</sub> -- descripción p <sub>1</sub> ... p <sub>q</sub> : Tipo <sub>n</sub> -- descripción p <sub>q</sub>
<b>Precondiciones</b>	<b>pre</b> <sub>1</sub> : < descripción de pre <sub>1</sub> en lenguaje natural > ... <b>pre</b> <sub>n</sub> : < descripción de pre <sub>n</sub> en lenguaje natural >
<b>Precondiciones (OCL)</b>	<b>pre</b> <sub>1</sub> : < descripción de pre <sub>1</sub> en OCL > ... <b>pre</b> <sub>n</sub> : < descripción de pre <sub>n</sub> en OCL >
<b>Postcondiciones</b>	<b>post</b> <sub>1</sub> : < descripción de post <sub>1</sub> en lenguaje natural > ... <b>post</b> <sub>m</sub> : < descripción de post <sub>m</sub> en lenguaje natural >
<b>Postcondiciones (OCL)</b>	<b>post</b> <sub>1</sub> : < descripción de post <sub>1</sub> en OCL > ... <b>post</b> <sub>m</sub> : < descripción de post <sub>m</sub> en OCL >
<b>Excepciones</b>	¬ <b>excep</b> <sub>1</sub> : < postcond. si se da excep <sub>1</sub> en lenguaje natural > ... ¬ <b>excep</b> <sub>n</sub> : < postcond. si se da excep <sub>n</sub> en lenguaje natural >
<b>Excepciones (OCL)</b>	<b>excep</b> <sub>1</sub> : < postcond. si se da excep <sub>n</sub> en OCL > ... <b>excep</b> <sub>n</sub> : < postcond. si se da excep <sub>n</sub> en OCL >
<b>Comentarios</b>	< comentarios adicionales sobre la operación >

Figura 13: Plantilla de descripción de operación de sistema

- **Parámetros:** en este apartado se deben enumerar los parámetros de la operación siguiendo la sintaxis de OCL con el formato:

nombre : tipo - descripción

siendo:

- **nombre:** nombre del parámetro.
  - **tipo:** tipo OCL del parámetro
  - **descripción:** descripción del parámetro.
- **Precondiciones:** en este apartado se describen en lenguaje natural las expresiones de precondición de la operación. Para facilitar la legibilidad y hacer patente la relación entre las expresiones en lenguaje natural y las expresadas en OCL, se ha optado por etiquetar cada expresión de las que forman la precondición. La precondición de la operación será la conjunción de todas las expresiones etiquetadas.
  - **Precondiciones (OCL):** en este apartado se describen en OCL las expresiones de precondición de la operación, definidas previamente en lenguaje natural. Cada expresión en OCL asociada con una frase de la precondición expresada en lenguaje natural debe tener la misma etiqueta.
  - **Postcondiciones:** en este apartado se describen en lenguaje natural las expresiones de postcondición de la operación en el caso de que todas las expresiones de la precondición sean ciertas. Al igual que en el caso de la precondición, la postcondición de la operación será la conjunción de todas las expresiones y cada frase se etiquetará de forma única.
  - **Postcondiciones (OCL):** en este apartado se describen en OCL las expresiones de postcondición de la operación, previamente descritas en lenguaje natural, en el caso de que todas las expresiones de la precondición sean ciertas. Al igual que en el caso de la precondición, cada expresión OCL se etiquetará coincidiendo con su expresión en lenguaje natural.
  - **Excepciones:** en este apartado se describen en lenguaje natural las expresiones de postcondición de la operación en el caso de que produzca alguna condición de excepción. Para expresar las condiciones de excepción se puede recurrir a las etiquetas de las expresiones de

precondición previamente definidas, por ejemplo:  $\neg pre_i \wedge \neg pre_j$  o  $\neg pre_i \vee \neg pre_j$ .

- **Excepciones (OCL):** en este apartado se describen en OCL las expresiones de postcondición de la operación en el caso de que produzca alguna condición de excepción. Al igual que en el apartado anterior, para expresar las condiciones de excepción se puede recurrir a las etiquetas de las expresiones de precondición previamente definidas, con lo que se facilita la legibilidad de las expresiones.

## 4.5 Diagrama de traza de eventos

Los diagramas de traza de eventos permiten expresar aspectos de comportamiento a nivel de varios tipos de objetos o a nivel de todo el sistema. En esta metodología se propone que los diagramas de traza de eventos utilicen la notación UML [Booch *et al.* 1999], en la que se denominan *diagramas de secuencia*.

## 4.6 Construcción de prototipos

Esta metodología no establece ninguna restricción para la elaboración de prototipos. Se podrán utilizar herramientas de desarrollo rápido, prototipos en papel o cualquier otra técnica que se considere oportuna.

## Referencias

- [Booch *et al.* 1999] G. Booch, J. Rumbaugh, y I. Jacobson. *The Unified Modeling Language User Guide*. Addison-Wesley, 1999.
- [Coleman *et al.* 1994] D. Coleman, P. Arnold, S. Bodoff, C. Dollin, H. Gilchrist, F. Hayes, y P. Jeremaes. *Object-Oriented Development. The Fusion Method*. Prentice-Hall, 1994.
- [Diller 1990] A. Diller. *Z: An Introduction to Formal Methods*. John Wiley, 1990.
- [D'Souza y Wills 1999] D. F. D'Souza y A. C. Wills. *Objects, Components, and Frameworks with UML: The Catalysis Approach*. Addison-Wesley, 1999.

- [Durán 2000] A. Durán. *Un Entorno Metodológico de Ingeniería de Requisitos para Sistemas de Información*. Tesis doctoral, Universidad de Sevilla, 2000.
- [IBM 1997] IBM Corporation. *Object Constraint Language Specification*, 1.1 edición, Septiembre 1997. Disponible en <http://www.rational.com>.
- [Ratcliff 1994] B. Ratcliff. *Introducing Specification Using Z: A Practical Case Study Approach*. International Series in Software Engineering. McGraw-Hill, 1994.

## A Ejemplo: gestión de un vídeo-club

En este apéndice se ofrecen algunos ejemplos de aplicación de las técnicas propuestas en esta metodología. Para ello se ha continuado con el ejemplo de la gestión de un pequeño vídeo-club que se utilizó en la *Metodología para la Elicitación de Requisitos de Sistemas Software*, aunque por motivos de espacio se ha optado por no incluir todas las operaciones del sistema correspondientes a los casos de uso identificados en el ejemplo citado, sino solamente un pequeño subconjunto.

Aunque no están definidos en la actual versión de OCL [IBM 1997], se han asumido los tipos básicos `Date` y `Time` para representar fechas y horas respectivamente. Se ha asumido también que se ha definido la operación menor o igual sobre ambos tipos (`<=`) y que `Date.today` y `Time.now` devuelven la fecha y hora actual respectivamente.

Para simplificar las expresiones de las expresiones de las postcondiciones de las operaciones en las que se crean o destruyen objetos se han asumido dos propiedades derivadas aplicables a cualquier tipo de objetos, `new` y `dead`, que corresponden a las siguientes expresiones para cualquier tipo `T`:

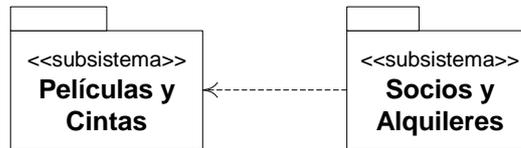
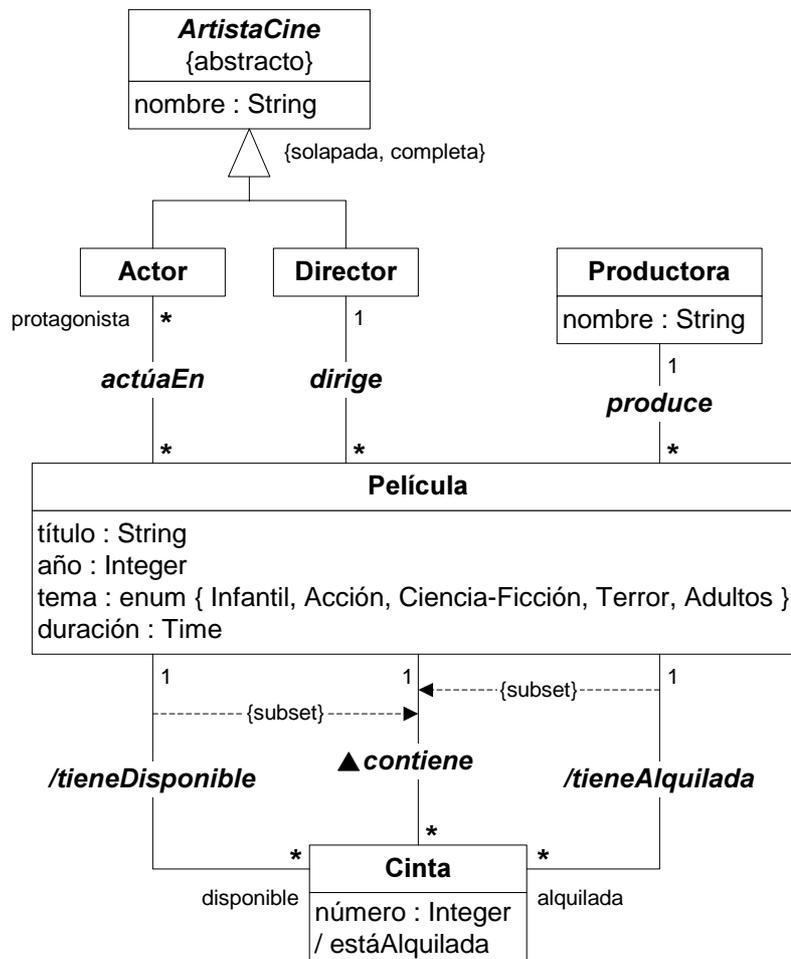
```
T.new = T.allInstances - T.allInstances@pre
T.dead = T.allInstances@pre - T.allInstances
```

y que representan, respectivamente, el conjunto de objetos creados durante la operación y el conjunto de objetos destruidos durante la operación.

En las operaciones del sistema especificadas se ha asumido que se ha desarrollado un prototipo de la interfaz de usuario del sistema.

### A.1 Diagramas de tipos

Los diagramas de tipos correspondientes al modelo, y realizados utilizando la notación UML [Booch *et al.* 1999], pueden verse en las figuras 15 y 16. Para mayor claridad se han dividido en dos *paquetes* o *subsistemas* que pueden verse en la figura 14.

Figura 14: Diagrama de *subsistemas*Figura 15: Diagrama de tipos del subsistema *Películas y Cintas*



## A.2 Tipos y asociaciones

### A.2.1 Subsistema Películas y Cintas

Este subsistema contiene los tipos y asociaciones relacionadas con las películas y cintas del vídeo-club.

### A.2.2 Tipo Actor

#### A.2.2.1 Descripción tipo Actor

<b>Tipo</b>	<b>Actor</b>		
<b>Requisitos asociados</b>	• RI-01 Información sobre películas		
<b>Descripción</b>	Este tipo concreto representa los actores protagonistas de las películas del vídeo-club		
<b>Supertipos</b>	ArtistaCine		
<b>Subtipos</b>	-		
<b>Componentes</b>	<b>Nombre</b>	<b>Tipo OCL</b>	<b>Mult.</b>
	-	-	-
<b>Comentarios</b>	Ninguno		

#### A.2.2.2 Atributos tipo Actor

No se han identificado atributos para este tipo.

#### A.2.2.3 Enlaces tipo Actor

<b>Enlace variable</b>	<b>Actor::película</b>
<b>Descripción</b>	Conjunto de películas en las que el actor es protagonista
<b>Tipo OCL</b>	Set( Película )
<b>Asociación</b>	<b>actúaEn(Actor,Película)</b>
<b>Comentarios</b>	Ninguno

#### A.2.2.4 Invariante tipo Actor

No se han identificado restricciones para este tipo.

### A.2.3 Tipo ArtistaCine

#### A.2.3.1 Descripción tipo ArtistaCine

<b>Tipo</b>	<b>ArtistaCine</b>		
<b>Requisitos asociados</b>	• RI-01 Información sobre películas		
<b>Descripción</b>	Este tipo abstracto representa los actores protagonistas de las películas del vídeo-club		
<b>Supertipos</b>	-		
<b>Subtipos</b>	<b>Actor, Director</b> (solapadas, completas)		
<b>Componentes</b>	<b>Nombre</b>	<b>Tipo OCL</b>	<b>Mult.</b>
	-	-	-
<b>Comentarios</b>	Puede haber artistas de cine que sean actores y directores a la vez, por ejemplo Woody Allen		

#### A.2.3.2 Atributos tipo ArtistaCine

<b>Atributo constante</b>	<b>ArtistaCine::nombre</b>
<b>Descripción</b>	Nombre del artista de cine
<b>Tipo OCL</b>	String
<b>Comentarios</b>	Ninguno

#### A.2.3.3 Enlaces tipo ArtistaCine

Este tipo no participa en ninguna asociación.

#### A.2.3.4 Invariante tipo ArtistaCine

<b>Invariante</b>	<b>ArtistaCine</b>
<b>Descripción</b>	• No puede haber dos artistas de cine con el mismo nombre (nombre es <i>clave</i> )
<b>Expresión</b>	<code>ArtistaCine.allInstances-&gt;forall( a1, a2   (a1 &lt;&gt; a2) = (a1.nombre &lt;&gt; a2.nombre) )</code>
<b>Comentarios</b>	Ninguno

## A.2.4 Tipo Cinta

### A.2.4.1 Descripción tipo Cinta

<b>Tipo</b>	<b>Cinta</b>		
<b>Requisitos asociados</b>	• RI-01 Información sobre películas		
<b>Descripción</b>	Este tipo concreto representa las cintas actualmente en poder del vídeo-club		
<b>Supertipos</b>	-		
<b>Subtipos</b>	-		
<b>Componentes</b>	<b>Nombre</b>	<b>Tipo OCL</b>	<b>Mult.</b>
	-	-	-
<b>Comentarios</b>	Ninguno		

### A.2.4.2 Atributos tipo Cinta

<b>Atributo derivado</b>	<b>Cinta::estáAlquilada</b>
<b>Descripción</b>	Indica si la cinta está alquilada actualmente
<b>Expresión</b>	<code>self.alquilerActual-&gt;notEmpty</code>
<b>Comentarios</b>	Ninguno

<b>Atributo constante</b>	<b>Cinta::número</b>
<b>Descripción</b>	Número identificativo de la cinta
<b>Tipo OCL</b>	Integer
<b>Comentarios</b>	Se debe generar automáticamente; es imprescindible para que el empleado del vídeo-club localice las cintas en los estantes

### A.2.4.3 Enlaces tipo Cinta

<b>Enlace variable</b>	<b>Cinta::alquiler</b>
<b>Descripción</b>	Alquileres de los que ha sido o es objeto la cinta
<b>Tipo OCL</b>	<code>Sequence( Alquiler )</code>
<b>Asociación</b>	<b>esObjetoDe(Cinta, Alquiler)</b>
<b>Comentarios</b>	Los alquileres están ordenados por fecha de alquiler

<b>Enlace derivado</b>	<b>Cinta::alquilerActual</b>
<b>Descripción</b>	Alquiler en curso de la cinta
<b>Expresión</b>	<code>self.alquiler-&gt;select( a   not a.estáDevuelto )</code>
<b>Asociación</b>	<b>esObjetoActualmenteDe(Cinta, Alquiler)</b>
<b>Comentarios</b>	Ninguno

<b>Enlace variable</b>	<b>Cinta::película</b>
<b>Descripción</b>	Película contenida en la cinta
<b>Tipo OCL</b>	Película
<b>Asociación</b>	<b>contiene(Cinta, Película)</b>
<b>Comentarios</b>	Ninguno

<b>Enlace derivado</b>	<b>Cinta::socio</b>
<b>Descripción</b>	Socio que tiene alquilada la cinta actualmente
<b>Expresión</b>	<code>self.alquilerActual.socio</code>
<b>Asociación</b>	<b>tieneAlquilada(Socio, Cinta)</b>
<b>Comentarios</b>	Ninguno

#### A.2.4.4 Invariante tipo Cinta

<b>Invariante</b>	<b>Cinta</b>
<b>Descripción</b>	<ul style="list-style-type: none"> <li>• No puede haber dos cintas con el mismo número (número es <i>clave</i>)</li> <li>• Los alquileres deben estar ordenados por fecha de alquiler</li> <li>• No hay alquileres duplicados</li> <li>• La cinta sólo puede tener pendiente un alquiler a la vez</li> </ul>
<b>Expresión</b>	<pre>Cinta.allInstances-&gt;forall( c1, c2       ( c1 &lt;&gt; c2 ) = ( c1.número &lt;&gt; c2.número ) ) and self.alquiler-&gt;isOrderedBy( &lt;=, fechaAlquiler ) and self.alquiler-&gt;size = self.alquiler-&gt;asSet-&gt;size and self.alquilerActual-&gt;size &lt;= 1</pre>
<b>Comentarios</b>	Ninguno

## A.2.5 Tipo Director

### A.2.5.1 Descripción tipo Director

<b>Tipo</b>	<b>Director</b>		
<b>Requisitos asociados</b>	• RI-01 Información sobre películas		
<b>Descripción</b>	Este tipo concreto representa los directores de las películas del vídeo-club		
<b>Supertipos</b>	ArtistaCine		
<b>Subtipos</b>	-		
<b>Componentes</b>	<b>Nombre</b>	<b>Tipo OCL</b>	<b>Mult.</b>
	-	-	-
<b>Comentarios</b>	Ninguno		

### A.2.5.2 Atributos tipo Director

No se han identificado atributos para este tipo.

### A.2.5.3 Enlaces tipo Director

<b>Enlace variable</b>	<b>Director::película</b>
<b>Descripción</b>	Conjunto de películas dirigidas por el director
<b>Tipo OCL</b>	Set( Película )
<b>Asociación</b>	<b>dirige(Director,Película)</b>
<b>Comentarios</b>	Ninguno

### A.2.5.4 Invariante tipo Director

No se han identificado restricciones para este tipo.

## A.2.6 Tipo Película

### A.2.6.1 Descripción tipo Película

<b>Tipo</b>	<b>Película</b>		
<b>Requisitos asociados</b>	• RI-01 Información sobre películas		
<b>Descripción</b>	Este tipo concreto representa las películas que han estado o están disponibles en el vídeo-club		
<b>Supertipos</b>	-		
<b>Subtipos</b>	-		
<b>Componentes</b>	<b>Nombre</b>	<b>Tipo OCL</b>	<b>Mult.</b>
	-	-	-
<b>Comentarios</b>	Ninguno		

### A.2.6.2 Atributos tipo Película

<b>Atributo constante</b>	<b>Película::año</b>
<b>Descripción</b>	Año en el que se estrenó la película
<b>Tipo OCL</b>	Integer
<b>Comentarios</b>	Ninguno

<b>Atributo constante</b>	<b>Película::duración</b>
<b>Descripción</b>	Duración de la película
<b>Tipo OCL</b>	Time
<b>Comentarios</b>	Ninguno

<b>Atributo constante</b>	<b>Película::tema</b>
<b>Descripción</b>	Tema de la película
<b>Tipo OCL</b>	enum {Infantil, Acción, Ciencia-Ficción, Terror, Adultos}
<b>Comentarios</b>	Ninguno

<b>Atributo constante</b>	<b>Película::título</b>
<b>Descripción</b>	Título de la película
<b>Tipo OCL</b>	String
<b>Comentarios</b>	Ninguno

### A.2.6.3 Enlaces tipo Película

<b>Enlace derivado</b>	<b>Película::alquilada</b>
<b>Descripción</b>	Cintas de la película actualmente alquiladas
<b>Expresión</b>	<code>self.cinta-&gt;select( c   c.estáAlquilada )</code>
<b>Asociación</b>	<b>tieneAlquilada(Película, Cinta)</b>
<b>Comentarios</b>	Ninguno

<b>Enlace variable</b>	<b>Película::cinta</b>
<b>Descripción</b>	Cintas de la película de las que dispone actualmente el vídeo-club
<b>Tipo OCL</b>	Set( Cinta )
<b>Asociación</b>	<b>contiene(Cinta, Película)</b>
<b>Comentarios</b>	Ninguno

<b>Enlace constante</b>	<b>Película::director</b>
<b>Descripción</b>	Director de la película
<b>Tipo OCL</b>	Director
<b>Asociación</b>	<b>dirige(Director, Película)</b>
<b>Comentarios</b>	Ninguno

<b>Enlace derivado</b>	<b>Película::disponible</b>
<b>Descripción</b>	Cintas de la película actualmente disponibles (no alquiladas)
<b>Expresión</b>	<code>self.cinta-&gt;select( c   not c.estáAlquilada )</code>
<b>Asociación</b>	<b>tieneDisponible(Película, Cinta)</b>
<b>Comentarios</b>	Ninguno

<b>Enlace constante</b>	<b>Película::productora</b>
<b>Descripción</b>	Productora de la película
<b>Tipo OCL</b>	Productora
<b>Asociación</b>	<b>produce(Productor, Película)</b>
<b>Comentarios</b>	Ninguno

<b>Enlace constante</b>	<b>Película::protagonista</b>
<b>Descripción</b>	Conjunto de actores protagonistas de la película
<b>Tipo OCL</b>	Set( Actor )
<b>Asociación</b>	<b>actúaEn(Actor, Película)</b>
<b>Comentarios</b>	Ninguno

#### A.2.6.4 Invariante tipo Película

<b>Invariante</b>	<b>Película</b>
<b>Descripción</b>	• Las cintas alquiladas y no alquiladas forman una partición de todas las cintas de la película
<b>Expresión</b>	<code>self.cinta = self.disponible-&gt;union( self.alquilada ) and self.disponible-&gt;intersection( self.alquilada )-&gt;isEmpty</code>
<b>Comentarios</b>	Ninguno

## A.2.7 Tipo Productora

### A.2.7.1 Descripción tipo Productora

<b>Tipo</b>	<b>Productora</b>		
<b>Requisitos asociados</b>	• RI-01 Información sobre películas		
<b>Descripción</b>	Este tipo concreto representa las productoras de las películas del vídeo-club		
<b>Supertipos</b>	-		
<b>Subtipos</b>	-		
<b>Componentes</b>	<b>Nombre</b>	<b>Tipo OCL</b>	<b>Mult.</b>
	-	-	-
<b>Comentarios</b>	Ninguno		

### A.2.7.2 Atributos tipo Productora

<b>Atributo constante</b>	<b>Productora::nombre</b>
<b>Descripción</b>	Nombre de la productora
<b>Tipo OCL</b>	String
<b>Comentarios</b>	Ninguno

### A.2.7.3 Enlaces tipo Productora

<b>Enlace variable</b>	<b>Productora::película</b>
<b>Descripción</b>	Conjunto de películas producidas por la productora
<b>Tipo OCL</b>	Set ( Película )
<b>Asociación</b>	<b>produce(Productora,Película)</b>
<b>Comentarios</b>	Ninguno

### A.2.7.4 Invariante tipo Productora

<b>Invariante</b>	<b>Productora</b>
<b>Descripción</b>	• No puede haber dos productoras con el mismo nombre (nombre es clave)
<b>Expresión</b>	Productora.allInstances->forall( a1, a2   (a1 <> a2) = (a1.nombre <> a2.nombre) )
<b>Comentarios</b>	Ninguno

## A.2.8 Asociación actúaEn entre Actor y Película

### A.2.8.1 Descripción asociación actúaEn(Actor, Película)

<b>Asociación</b>	<b>actúaEn entre Actor y Película</b>
<b>Requisitos asociados</b>	• RI-01 Información sobre películas
<b>Descripción</b>	Esta asociación representa el hecho de que un actor es protagonista en una película
<b>Comentarios</b>	Ninguno

### A.2.8.2 Roles asociación actúaEn(Actor, Película)

<b>Rol</b>	<b>Actor juega rol protagonista en actúaEn(Actor, Película)</b>
<b>Descripción</b>	Actor protagonista de una película
<b>Tipo OCL</b>	Set ( Actor )
<b>Multiplicidad</b>	0..n
<b>Comentarios</b>	Ninguno

<b>Rol</b>	<b>Película juega rol película en actúaEn(Actor, Película)</b>
<b>Descripción</b>	Películas en la que un actor es protagonista
<b>Tipo OCL</b>	Set ( Película )
<b>Multiplicidad</b>	0..n
<b>Comentarios</b>	Ninguno

## A.2.9 Asociación contiene entre Cinta y Película

### A.2.9.1 Descripción asociación contiene(Cinta, Película)

<b>Asociación</b>	<b>contiene</b> entre <b>Cinta</b> y <b>Película</b>
<b>Requisitos asociados</b>	• RI-01 Información sobre películas
<b>Descripción</b>	Esta asociación representa el hecho de que una cinta de vídeo contiene una película
<b>Comentarios</b>	Ninguno

### A.2.9.2 Roles asociación contiene(Cinta, Película)

<b>Rol</b>	<b>Cinta</b> juega rol <b>cinta</b> en <b>contiene(Cinta, Película)</b>
<b>Descripción</b>	Cinta que contiene una película
<b>Tipo OCL</b>	Set( Cinta )
<b>Multiplicidad</b>	0..n
<b>Comentarios</b>	Ninguno

<b>Rol</b>	<b>Película</b> juega rol <b>película</b> en <b>contiene(Cinta, Película)</b>
<b>Descripción</b>	Cinta que contiene una película
<b>Tipo OCL</b>	Película
<b>Multiplicidad</b>	0..n
<b>Comentarios</b>	Ninguno

## A.2.10 Asociación dirige entre Director y Película

### A.2.10.1 Descripción asociación dirige(Director, Película)

<b>Asociación</b>	<b>dirige</b> entre <b>Director</b> y <b>Película</b>
<b>Requisitos asociados</b>	• RI-01 Información sobre películas
<b>Descripción</b>	Esta asociación representa el hecho de que un director dirige una película
<b>Comentarios</b>	Ninguno

### A.2.10.2 Roles asociación dirige(Director, Película)

<b>Rol</b>	<b>Director</b> juega rol <b>director</b> en <b>dirige(Director, Película)</b>
<b>Descripción</b>	Director de una película
<b>Tipo OCL</b>	Director
<b>Multiplicidad</b>	1
<b>Comentarios</b>	Ninguno

<b>Rol</b>	<b>Película</b> juega rol <b>película</b> en <b>dirige(Director, Película)</b>
<b>Descripción</b>	Películas dirigidas por un director
<b>Tipo OCL</b>	Set ( Película )
<b>Multiplicidad</b>	0..n
<b>Comentarios</b>	Ninguno

## A.2.11 Asociación produce entre Productora y Película

### A.2.11.1 Descripción asociación produce(Productora, Película)

<b>Asociación</b>	<b>produce</b> entre <b>Productora</b> y <b>Película</b>
<b>Requisitos asociados</b>	• RI-01 Información sobre películas
<b>Descripción</b>	Esta asociación representa el hecho de que una productora produce una película
<b>Comentarios</b>	Ninguno

### A.2.11.2 Roles asociación produce(Productora, Película)

<b>Rol</b>	<b>Película</b> juega rol <b>película</b> en <b>produce(Productora, Película)</b>
<b>Descripción</b>	Películas producidas por una productora
<b>Tipo OCL</b>	Set( Película )
<b>Multiplicidad</b>	0..n
<b>Comentarios</b>	Ninguno

<b>Rol</b>	<b>Director</b> juega rol <b>Productora</b> en <b>produce(Productora, Película)</b>
<b>Descripción</b>	Productora de una película
<b>Tipo OCL</b>	Productora
<b>Multiplicidad</b>	1
<b>Comentarios</b>	Ninguno

## A.2.12 Asociación tieneAlquilada entre Película y Cinta

### A.2.12.1 Descripción asociación tieneAlquilada(Película, Cinta)

<b>Asociación derivada</b>	<b>tieneAlquilada entre Película y Cinta</b>
<b>Requisitos asociados</b>	• RI-01 Información sobre películas
<b>Descripción</b>	Esta asociación representa el hecho de que una película tiene alquiladas cintas
<b>Comentarios</b>	Ver invariante, es una asociación derivada

### A.2.12.2 Roles asociación tieneAlquilada(Película, Cinta)

<b>Rol</b>	<b>Cinta juega rol alquilada en tieneAlquilada(Película, Cinta)</b>
<b>Descripción</b>	Cintas alquiladas
<b>Tipo OCL</b>	Set( Cinta )
<b>Multiplicidad</b>	0..n
<b>Comentarios</b>	Ninguno

<b>Rol</b>	<b>Película juega rol película en tieneAlquilada(Película, Cinta)</b>
<b>Descripción</b>	Películas contenida en una cinta
<b>Tipo OCL</b>	Película
<b>Multiplicidad</b>	1
<b>Comentarios</b>	Ninguno

### A.2.12.3 Invariante asociación tieneAlquilada(Película, Cinta)

<b>Invariante</b>	<b>tieneAlquilada(Película, Cinta)</b>
<b>Descripción</b>	• Un par (película,cinta) pertenece a esta asociación si y sólo si la cinta contiene a la película y la cinta está alquilada
<b>Expresión</b>	<pre>Película.allInstances-&gt;forall( p     Cinta.allInstances-&gt;forall( c       tieneAlquilada(Película,Cinta).allInstances-&gt;forall( ta         ( ta.película = p and ta.alquilada = c ) =       ( p.cinta-&gt;includes( c ) and c.película = p and         c.estáAlquilada ) ) ) )</pre>
<b>Comentarios</b>	Ninguno

## A.2.13 Asociación tieneDisponible entre Película y Cinta

### A.2.13.1 Descripción asociación tieneDisponible(Película, Cinta)

<b>Asociación derivada</b>	<b>tieneDisponible</b> entre Película y Cinta
<b>Requisitos asociados</b>	• RI-01 Información sobre películas
<b>Descripción</b>	Esta asociación representa el hecho de que una película tiene disponible cintas para alquilar
<b>Comentarios</b>	Ver invariante, es una asociación derivada

### A.2.13.2 Roles asociación tieneDisponible(Película, Cinta)

<b>Rol</b>	<b>Cinta</b> juega rol <b>disponible</b> en <b>tieneDisponible(Película, Cinta)</b>
<b>Descripción</b>	Cintas disponibles para alquilar
<b>Tipo OCL</b>	Set( Cinta )
<b>Multiplicidad</b>	0..n
<b>Comentarios</b>	Ninguno

<b>Rol</b>	<b>Película</b> juega rol <b>película</b> en <b>tieneDisponible(Película, Cinta)</b>
<b>Descripción</b>	Películas contenida en una cinta
<b>Tipo OCL</b>	Película
<b>Multiplicidad</b>	1
<b>Comentarios</b>	Ninguno

### A.2.13.3 Invariante asociación tieneDisponible(Película, Cinta)

<b>Invariante</b>	<b>tieneDisponible(Película, Cinta)</b>
<b>Descripción</b>	• Un par ( <i>película,cinta</i> ) pertenece a esta asociación si y sólo si la cinta contiene a la película y la cinta no está alquilada
<b>Expresión</b>	<pre> Película.allInstances-&gt;forall( p     Cinta.allInstances-&gt;forall( c       tieneDisponible(Película,Cinta).allInstances-&gt;forall( td         ( td.película = p and td.alquilada = c ) =       ( p.cinta-&gt;includes( c ) and c.película = p and         not c.estáAlquilada ) ) ) ) </pre>
<b>Comentarios</b>	Ninguno

### A.2.14 Subsistema Socios y Alquileres

Este subsistema contiene los tipos y asociaciones relacionadas con los socios y alquileres del vídeo-club.

### A.2.15 Tipo Alquiler

#### A.2.15.1 Descripción tipo Alquiler

<b>Tipo</b>	<b>Alquiler</b>		
<b>Requisitos asociados</b>	<ul style="list-style-type: none"> <li>• RI-02 Información sobre socios</li> <li>• RI-03 Información sobre cuentas de socios</li> </ul>		
<b>Descripción</b>	Este tipo concreto representa los alquileres de cintas actuales y pasados realizados por parte de los socios del vídeo-club		
<b>Supertipos</b>	-		
<b>Subtipos</b>	-		
<b>Componentes</b>	<b>Nombre</b>	<b>Tipo OCL</b>	<b>Mult.</b>
	-	-	-
<b>Comentarios</b>	Este tipo no se ha modelado como una asociación como tipo porque un socio puede alquilar una misma cinta más de una vez		

#### A.2.15.2 Atributos tipo Alquiler

<b>Atributo variable</b>	<b>Alquiler::estáDevuelto</b>
<b>Descripción</b>	Indica si el socio ha devuelto la cinta alquilada
<b>Tipo OCL</b>	Boolean
<b>Valor inicial</b>	false
<b>Comentarios</b>	Ninguno

<b>Atributo constante</b>	<b>Alquiler::fechaAlquiler</b>
<b>Descripción</b>	Fecha en la que el socio retira la cinta
<b>Tipo OCL</b>	Date
<b>Comentarios</b>	Ninguno

<b>Atributo constante</b>	<b>Alquiler::fechaDevolución</b>
<b>Descripción</b>	Fecha tope en la que el socio <i>debe</i> devolver la cinta
<b>Tipo OCL</b>	Date
<b>Comentarios</b>	Ninguno

<b>Atributo constante</b>	<b>Alquiler::horaAlquiler</b>
<b>Descripción</b>	Hora en la que el socio retira la cinta
<b>Tipo OCL</b>	Time
<b>Comentarios</b>	Ninguno

<b>Atributo constante</b>	<b>Alquiler::horaDevolución</b>
<b>Descripción</b>	Hora tope en la que el socio <i>debe</i> devolver la cinta
<b>Tipo OCL</b>	Time
<b>Comentarios</b>	Ninguno

### A.2.15.3 Enlaces tipo Alquiler

<b>Enlace variable</b>	<b>Alquiler::cargoAlquiler</b>
<b>Descripción</b>	Cargo correspondiente al alquiler
<b>Tipo OCL</b>	Alquiler
<b>Asociación</b>	<b>corresponde(CargoAlquiler, Alquiler)</b>
<b>Comentarios</b>	Ninguno

<b>Enlace variable</b>	<b>Alquiler::cargoMulta</b>
<b>Descripción</b>	Cargo correspondiente a la multa por devolución tardía
<b>Tipo OCL</b>	Set( Alquiler ) o Alquiler
<b>Asociación</b>	<b>motivadoPor(CargoMulta, Alquiler)</b>
<b>Comentarios</b>	Los enlaces asociados a los roles con cardinalidad 0..1 puede actuar como conjuntos o como objetos en OCL

<b>Enlace constante</b>	<b>Alquiler::cinta</b>
<b>Descripción</b>	Cinta que es objeto del alquiler
<b>Tipo OCL</b>	Cinta
<b>Asociación</b>	<b>esObjetoDe(Cinta, Alquiler)</b>
<b>Comentarios</b>	Ninguno

<b>Enlace constante</b>	<b>Alquiler::socio</b>
<b>Descripción</b>	Cinta que es objeto del alquiler
<b>Tipo OCL</b>	Socio que realiza el alquiler
<b>Asociación</b>	<b>realiza(Socio, Alquiler)</b>
<b>Comentarios</b>	Ninguno

## A.2.15.4 Invariante tipo Alquiler

Invariante	Alquiler
Descripción	<ul style="list-style-type: none"> <li>• No puede haber dos alquileres de la misma cinta en la misma fecha y hora</li> <li>• No puede haber dos alquileres pendientes de la misma cinta</li> <li>• La fecha y hora de devolución tienen que ser posteriores o iguales a las de alquiler</li> </ul>
Expresión	<pre> Alquiler.allInstances-&gt;forall( a1, a2     ( a1 &lt;&gt; a2 ) = (     a1.cinta &lt;&gt; a2.cinta or     a1.fechaAlquiler &lt;&gt; a2.fechaAlquiler or     a1.horaAlquiler &lt;&gt; a2.horaAlquiler ) ) and Alquiler.allInstances-&gt;forall( a1, a2     ( a1 &lt;&gt; a2 and a1.cinta = a2.cinta ) implies   ( a1.estáDevuelto or a2.estáDevuelto ) ) and self.fechaAlquiler &lt;= self.fechaDevolución and ( self.fechaAlquiler = self.fechaDevolución ) implies ( self.horaAlquiler &lt; self.horaDevolución ) </pre>
Comentarios	Ninguno

## A.2.16 Tipo Cargo

### A.2.16.1 Descripción tipo Cargo

<b>Tipo</b>	<b>Cargo</b>		
<b>Requisitos asociados</b>	• RI-03 Información sobre cuentas de socios		
<b>Descripción</b>	Este tipo abstracto representa los cargos de las cuentas de los socios del vídeo-club		
<b>Supertipos</b>	Movimiento		
<b>Subtipos</b>	CargoMulta, CargoAlquiler (disjuntas, completas)		
<b>Componentes</b>	<b>Nombre</b>	<b>Tipo OCL</b>	<b>Mult.</b>
	-	-	-
<b>Comentarios</b>	Ninguno		

### A.2.16.2 Atributos tipo Cargo

<b>Atributo variable</b>	<b>Cargo::estáPendiente</b>
<b>Descripción</b>	Indica si el cargo está pendiente de pago
<b>Tipo OCL</b>	Boolean
<b>Valor inicial</b>	true
<b>Comentarios</b>	Cuando se realiza un cargo, si no hay saldo disponible para pagarlo se considera pendiente de pago hasta que haya saldo suficiente

### A.2.16.3 Enlaces tipo Cargo

No se han identificado enlaces para este tipo.

### A.2.16.4 Invariante tipo Cargo

<b>Invariante</b>	<b>Cargo</b>
<b>Descripción</b>	• El importe del movimiento es negativo
<b>Expresión</b>	<code>self.importe &lt;= 0</code>
<b>Comentarios</b>	Ninguno

## A.2.17 Tipo CargoAlquiler

### A.2.17.1 Descripción tipo CargoAlquiler

<b>Tipo</b>	<b>CargoAlquiler</b>		
<b>Requisitos asociados</b>	• RI-03 Información sobre cuentas de socios		
<b>Descripción</b>	Este tipo concreto representa los cargos de las cuentas de los socios del vídeo-club correspondientes a los alquileres de cintas		
<b>Supertipos</b>	Cargo		
<b>Subtipos</b>	-		
<b>Componentes</b>	<b>Nombre</b>	<b>Tipo OCL</b>	<b>Mult.</b>
	-	-	-
<b>Comentarios</b>	Ninguno		

### A.2.17.2 Atributos tipo CargoAlquiler

No se han identificado atributos para este tipo.

### A.2.17.3 Enlaces tipo CargoAlquiler

<b>Enlace constante</b>	<b>CargoAlquiler::alquiler</b>
<b>Descripción</b>	El alquiler al que corresponde el cargo
<b>Tipo OCL</b>	Alquiler
<b>Asociación</b>	<b>corresponde(CargoAlquiler, Alquiler)</b>
<b>Comentarios</b>	Ninguno

### A.2.17.4 Invariante tipo CargoAlquiler

No se han identificado restricciones para este tipo.

## A.2.18 Tipo CargoMulta

### A.2.18.1 Descripción tipo CargoMulta

<b>Tipo</b>	<b>CargoMulta</b>		
<b>Requisitos asociados</b>	• RI-03 Información sobre cuentas de socios		
<b>Descripción</b>	Este tipo concreto representa los cargos de las cuentas de los socios del vídeo-club correspondientes a las multas por devoluciones tardías		
<b>Supertipos</b>	Cargo		
<b>Subtipos</b>	-		
<b>Componentes</b>	<b>Nombre</b>	<b>Tipo OCL</b>	<b>Mult.</b>
	-	-	-
<b>Comentarios</b>	Ninguno		

### A.2.18.2 Atributos tipo CargoMulta

No se han identificado atributos para este tipo.

### A.2.18.3 Enlaces tipo CargoMulta

<b>Enlace constante</b>	<b>CargoMulta::alquiler</b>
<b>Descripción</b>	El alquiler con devolución tardía que motiva el cargo por multa
<b>Tipo OCL</b>	Alquiler
<b>Asociación</b>	<b>motivadoPor(CargoMulta,Alquiler)</b>
<b>Comentarios</b>	Ninguno

### A.2.18.4 Invariante tipo CargoMulta

No se han identificado restricciones para este tipo.

## A.2.19 Tipo Cuenta

### A.2.19.1 Descripción tipo Cuenta

<b>Tipo</b>	<b>Cuenta</b>		
<b>Requisitos asociados</b>	• RI-03 Información sobre cuentas de socios		
<b>Descripción</b>	Este tipo concreto representa las cuentas de los socios del vídeo-club		
<b>Supertipos</b>	-		
<b>Subtipos</b>	-		
<b>Componentes</b>	<b>Nombre</b>	<b>Tipo OCL</b>	<b>Mult.</b>
	movimiento	Sequence( Movimiento )	0..n
<b>Comentarios</b>	Los movimientos están ordenados por fecha		

### A.2.19.2 Atributos tipo Cuenta

<b>Atributo derivado</b>	<b>Cuenta::saldo</b>
<b>Descripción</b>	Saldo actual de la cuenta
<b>Expresión</b>	<code>self.movimiento-&gt;collect( importe )-&gt;sum</code>
<b>Comentarios</b>	Ninguno

### A.2.19.3 Enlaces tipo Cuenta

<b>Enlace constante</b>	<b>Cuenta::socio</b>
<b>Descripción</b>	El socio de la cuenta
<b>Tipo OCL</b>	Socio
<b>Asociación</b>	<b>tiene(Socio,Cuenta)</b>
<b>Comentarios</b>	Ninguno

### A.2.19.4 Invariante tipo Cuenta

<b>Invariante</b>	<b>Cuenta</b>
<b>Descripción</b>	• Los movimientos están ordenados por fecha
<b>Expresión</b>	<code>self.movimiento-&gt;isOrderedBy( &lt;=, fecha )</code>
<b>Comentarios</b>	Ninguno

## A.2.20 Tipo Ingreso

### A.2.20.1 Descripción tipo Ingreso

<b>Tipo</b>	<b>Ingreso</b>		
<b>Requisitos asociados</b>	• RI-03 Información sobre cuentas de socios		
<b>Descripción</b>	Este tipo concreto representa los ingresos de las cuentas de los socios del vídeo-club		
<b>Supertipos</b>	Movimiento		
<b>Subtipos</b>	-		
<b>Componentes</b>	<b>Nombre</b>	<b>Tipo OCL</b>	<b>Mult.</b>
	-	-	-
<b>Comentarios</b>	Ninguno		

### A.2.20.2 Atributos tipo Ingreso

No se han identificado atributos para este tipo.

### A.2.20.3 Enlaces tipo Ingreso

No se han identificado enlaces para este tipo.

### A.2.20.4 Invariante tipo Ingreso

<b>Invariante</b>	<b>Ingreso</b>
<b>Descripción</b>	• El importe del cargo es positivo
<b>Expresión</b>	<code>self.importe &gt; 0</code>
<b>Comentarios</b>	Ninguno

## A.2.21 Tipo Movimiento

### A.2.21.1 Descripción tipo Movimiento

<b>Tipo</b>	<b>Movimiento</b>		
<b>Requisitos asociados</b>	• RI-03 Información sobre cuentas de socios		
<b>Descripción</b>	Este tipo abstracto representa los movimientos de las cuentas de los socios del vídeo-club		
<b>Supertipos</b>	-		
<b>Subtipos</b>	Ingreso, Cargo (disjuntas, completas)		
<b>Componentes</b>	<b>Nombre</b>	<b>Tipo OCL</b>	<b>Mult.</b>
	-	-	-
<b>Comentarios</b>	Ninguno		

### A.2.21.2 Atributos tipo Movimiento

<b>Atributo constante</b>	<b>Movimiento::fecha</b>
<b>Descripción</b>	Fecha en la que se realiza el movimiento
<b>Tipo OCL</b>	Date
<b>Comentarios</b>	Ninguno

<b>Atributo constante</b>	<b>Movimiento::importe</b>
<b>Descripción</b>	Cantidad correspondiente al importe del movimiento
<b>Tipo OCL</b>	Integer
<b>Comentarios</b>	Ninguno

### A.2.21.3 Enlaces tipo Movimiento

No se han identificado enlaces para este tipo.

### A.2.21.4 Invariante tipo Movimiento

No se han identificado restricciones para este tipo.

## A.2.22 Tipo Socio

### A.2.22.1 Descripción tipo Socio

<b>Tipo</b>	<b>Socio</b>		
<b>Requisitos asociados</b>	• RI-02 Información sobre socios		
<b>Descripción</b>	Este tipo concreto representa los socios actuales del vídeo-club		
<b>Supertipos</b>	-		
<b>Subtipos</b>	-		
<b>Componentes</b>	<b>Nombre</b>	<b>Tipo OCL</b>	<b>Mult.</b>
	-	-	-
<b>Comentarios</b>	Ninguno		

### A.2.22.2 Atributos tipo Socio

<b>Atributo variable</b>	<b>Socio::dirección</b>
<b>Descripción</b>	Dirección actual del socio
<b>Tipo OCL</b>	String
<b>Comentarios</b>	Ninguno

<b>Atributo constante</b>	<b>Socio::dni</b>
<b>Descripción</b>	Número del Documento Nacional de Identidad del socio
<b>Tipo OCL</b>	String
<b>Comentarios</b>	Ninguno

<b>Atributo constante</b>	<b>Socio::fechaAlta</b>
<b>Descripción</b>	Fecha de alta del socio
<b>Tipo OCL</b>	Date
<b>Comentarios</b>	Ninguno

<b>Atributo constante</b>	<b>Socio::fechaNacimiento</b>
<b>Descripción</b>	Fecha de nacimiento del socio
<b>Tipo OCL</b>	Date
<b>Comentarios</b>	Ninguno

<b>Atributo constante</b>	<b>Socio::nombre</b>
<b>Descripción</b>	Nombre y apellidos del socio
<b>Tipo OCL</b>	String
<b>Comentarios</b>	Ninguno

<b>Atributo constante</b>	<b>Socio::número</b>
<b>Descripción</b>	Número identificativo del socio
<b>Tipo OCL</b>	Integer
<b>Comentarios</b>	Se debe generar automáticamente

<b>Atributo constante</b>	<b>Socio::sexo</b>
<b>Descripción</b>	Sexo del socio
<b>Tipo OCL</b>	enum { Hombre, Mujer }
<b>Comentarios</b>	Se asume que la probabilidad de que un socio cambie de sexo es muy baja

<b>Atributo constante</b>	<b>Socio::teléfonos</b>
<b>Descripción</b>	Teléfonos del socio
<b>Tipo OCL</b>	Set( Integer )
<b>Comentarios</b>	Siempre tiene que haber al menos un teléfono de contacto

### A.2.22.3 Enlaces tipo Socio

<b>Enlace variable</b>	<b>Socio::alquiler</b>
<b>Descripción</b>	Conjunto de alquileres pasados y presentes del socio
<b>Tipo OCL</b>	Sequence( Alquiler )
<b>Asociación</b>	<b>realiza(Socio, Alquiler)</b>
<b>Comentarios</b>	No se permiten duplicados en la secuencia de alquileres y deben estar ordenados por fecha de alquiler (ver invariante)

<b>Enlace derivado</b>	<b>Socio::alquilerActual</b>
<b>Descripción</b>	Conjunto de alquileres actuales del socio
<b>Expresión</b>	<code>self.alquiler-&gt;select( a   not a.estáDevuelto )</code>
<b>Asociación</b>	<b>realizaActualmente(Socio, Alquiler)</b>
<b>Comentarios</b>	Ninguno

<b>Enlace derivado</b>	<b>Socio::cinta</b>
<b>Descripción</b>	Conjunto de cintas actualmente alquiladas por el socio
<b>Expresión</b>	<code>self.alquilerActual.cinta-&gt;asSet</code>
<b>Asociación</b>	<b>tieneAlquilada(Socio, Cinta)</b>
<b>Comentarios</b>	Ninguno

<b>Enlace constante</b>	<b>Socio::cuenta</b>
<b>Descripción</b>	Cuenta del socio
<b>Tipo OCL</b>	Cuenta
<b>Asociación</b>	<b>tiene(Socio, Cuenta)</b>
<b>Comentarios</b>	Ninguno

## A.2.22.4 Invariante tipo Socio

Invariante	Socio
<b>Descripción</b>	<ul style="list-style-type: none"> <li>• No puede haber dos socios con el mismo DNI ni con el mismo número (dni y número son <i>clave</i>)</li> <li>• Al menos debe tener un teléfono de contacto</li> <li>• Los alquileres deben estar ordenados por fecha de alquiler</li> <li>• No hay alquileres duplicados</li> </ul>
<b>Expresión</b>	<pre>Socio.allInstances-&gt;forAll( s1, s2     ( s1 &lt;&gt; s2 ) =   ( s1.dni &lt;&gt; s2.dni and s1.número &lt;&gt; s2.número ) ) and not (self.teléfonos-&gt;isEmpty) and self.alquiler-&gt;isOrderedBy( &lt;=, fechaAlquiler ) and self.alquiler-&gt;size = self.alquiler-&gt;asSet-&gt;size</pre>
<b>Comentarios</b>	Ninguno

### A.2.23 Asociación corresponde entre CargoAlquiler y Alquiler

#### A.2.23.1 Descripción asociación corresponde(CargoAlquiler, Alquiler)

<b>Asociación</b>	<b>corresponde</b> entre CargoAlquiler y Alquiler
<b>Requisitos asociados</b>	• RI-03 Información sobre cuentas de socios
<b>Descripción</b>	Esta asociación representa el hecho de que un cargo por alquiler corresponde a un determinado alquiler
<b>Comentarios</b>	Ninguno

#### A.2.23.2 Roles asociación corresponde(CargoAlquiler, Alquiler)

<b>Rol</b>	<b>Alquiler</b> juega rol alquiler en corresponde(CargoAlquiler, Alquiler)
<b>Descripción</b>	Alquiler correspondiente al cargo
<b>Tipo OCL</b>	Alquiler
<b>Multiplicidad</b>	1
<b>Comentarios</b>	Ninguno

<b>Rol</b>	<b>CargoAlquiler</b> juega rol cargoAlquiler en corresponde(CargoAlquiler, Alquiler)
<b>Descripción</b>	Cargo correspondiente al alquiler
<b>Tipo OCL</b>	CargoAlquiler
<b>Multiplicidad</b>	1
<b>Comentarios</b>	Ninguno

## A.2.24 Asociación esObjetoDe entre Cinta y Alquiler

### A.2.24.1 Descripción asociación esObjetoDe(Cinta, Alquiler)

<b>Asociación</b>	<b>esObjetoDe entre Cinta y Alquiler</b>
<b>Requisitos asociados</b>	<ul style="list-style-type: none"> <li>• RI-02 Información sobre socios</li> <li>• RI-03 Información sobre cuentas de socios</li> </ul>
<b>Descripción</b>	Esta asociación representa el hecho de que una cinta es objeto de alquileres
<b>Comentarios</b>	Ninguno

### A.2.24.2 Roles asociación esObjetoDe(Cinta, Alquiler)

<b>Rol</b>	<b>Alquiler juega rol alquiler en esObjetoDe(Cinta, Alquiler)</b>
<b>Descripción</b>	Alquileres de los que ha sido objeto una cinta
<b>Tipo OCL</b>	Sequence( Alquiler )
<b>Multiplicidad</b>	0..n
<b>Comentarios</b>	Los alquileres están ordenados por fecha de alquiler

<b>Rol</b>	<b>Cinta juega rol cinta en esObjetoDe(Cinta, Alquiler)</b>
<b>Descripción</b>	Cinta que es objeto del alquiler
<b>Tipo OCL</b>	Cinta
<b>Multiplicidad</b>	1
<b>Comentarios</b>	Ninguno

## A.2.25 Asociación esObjetoActualmenteDe entre Cinta y Alquiler

### A.2.25.1 Descripción asociación esObjetoActualmenteDe(Cinta, Alquiler)

<b>Asociación derivada</b>	<b>esObjetoActualmenteDe entre Cinta y Alquiler</b>
<b>Requisitos asociados</b>	<ul style="list-style-type: none"> <li>• RI-02 Información sobre socios</li> <li>• RI-03 Información sobre cuentas de socios</li> </ul>
<b>Descripción</b>	Esta asociación derivada representa el hecho de que una cinta es objeto actualmente de un alquiler
<b>Comentarios</b>	Ninguno

### A.2.25.2 Roles asociación esObjetoActualmenteDe(Cinta, Alquiler)

<b>Rol</b>	<b>Alquiler</b> juega rol <b>alquilerActual</b> en <b>esObjetoActualmenteDe(Cinta, Alquiler)</b>
<b>Descripción</b>	Alquiler actual (aún no devuelto) del que es objeto la cinta
<b>Tipo OCL</b>	Alquiler o Set( Alquiler )
<b>Multiplicidad</b>	0..1
<b>Comentarios</b>	Los roles con cardinalidad 0..1 puede actuar como conjuntos o como objetos en OCL

<b>Rol</b>	<b>Cinta</b> juega rol <b>cinta</b> en <b>esObjetoActualmenteDe(Cinta, Alquiler)</b>
<b>Descripción</b>	Cinta que es objeto del alquiler
<b>Tipo OCL</b>	Cinta
<b>Multiplicidad</b>	1
<b>Comentarios</b>	Ninguno

### A.2.25.3 Invariante asociación esObjetoActualmenteDe(Cinta, Alquiler)

<b>Invariante</b>	<b>esObjetoActualmenteDe(Cinta, Alquiler)</b>
<b>Descripción</b>	• Un par ( <i>cinta, alquiler</i> ) pertenece a esta asociación si y sólo existe un alquiler en el que la cinta es alquilada y dicho alquiler está sin devolver
<b>Expresión</b>	<pre>Cinta.allInstances-&gt;forall( c     Alquiler.allInstances-&gt;forall( a       esObjetoActualmenteDe(Cinta, Alquiler).allInstances-&gt;     forall( eoa         ( eoa.cinta = c and eoa.alquiler = c ) =       ( a.cinta = c and c.alquiler-&gt;includes( a ) and         not a.estáDevuelto ) ) ) )</pre>
<b>Comentarios</b>	Ninguno

## A.2.26 Asociación motivadoPor entre CargoMulta y Alquiler

### A.2.26.1 Descripción asociación motivadoPor(CargoMulta, Alquiler)

<b>Asociación</b>	<b>motivadoPor</b> entre <b>CargoMulta</b> y <b>Alquiler</b>
<b>Requisitos asociados</b>	• RI-03 Información sobre cuentas de socios
<b>Descripción</b>	Esta asociación representa el hecho de que un cargo por multa por alquiler con devolución tardía corresponde a un determinado alquiler
<b>Comentarios</b>	Ninguno

### A.2.26.2 Roles asociación motivadoPor(CargoMulta, Alquiler)

<b>Rol</b>	<b>Alquiler</b> juega rol <b>alquiler</b> en <b>motivadoPor(CargoMulta, Alquiler)</b>
<b>Descripción</b>	Alquiler correspondiente al cargo por multa
<b>Tipo OCL</b>	Alquiler
<b>Multiplicidad</b>	1
<b>Comentarios</b>	Ninguno

<b>Rol</b>	<b>CargoMulta</b> juega rol <b>cargoMulta</b> en <b>motivadoPor(CargoMulta, Alquiler)</b>
<b>Descripción</b>	Cargo correspondiente a la multa por alquiler con devolución tardía
<b>Tipo OCL</b>	Set( CargoAlquiler ) o CargoAlquiler
<b>Multiplicidad</b>	0..1
<b>Comentarios</b>	Los roles con cardinalidad 0..1 puede actuar como conjuntos o como objetos en OCL

## A.2.27 Asociación realiza entre Socio y Alquiler

### A.2.27.1 Descripción asociación realiza(Socio, Alquiler)

<b>Asociación</b>	<b>realiza entre Socio y Alquiler</b>
<b>Requisitos asociados</b>	• RI-02 Información sobre socios
<b>Descripción</b>	Esta asociación representa el hecho de que un socio realiza alquileres de cintas
<b>Comentarios</b>	Ninguno

### A.2.27.2 Roles asociación realiza(Socio, Alquiler)

<b>Rol</b>	<b>Alquiler juega rol alquiler en realiza(Socio, Alquiler)</b>
<b>Descripción</b>	Alquileres realizados por el socio
<b>Tipo OCL</b>	Sequence( Alquiler )
<b>Multiplicidad</b>	0..n
<b>Comentarios</b>	Los alquileres están ordenados por fecha de alquiler

<b>Rol</b>	<b>Socio juega rol socio en realiza(Socio, Alquiler)</b>
<b>Descripción</b>	Socio que realiza el alquiler
<b>Tipo OCL</b>	Socio
<b>Multiplicidad</b>	1
<b>Comentarios</b>	Ninguno

## A.2.28 Asociación realizaActualmente entre Socio y Alquiler

### A.2.28.1 Descripción asociación realizaActualmente(Socio, Alquiler)

<b>Asociación derivada</b>	realizaActualmente entre Socio y Alquiler
<b>Requisitos asociados</b>	• RI-02 Información sobre socios
<b>Descripción</b>	Esta asociación derivada representa el hecho de que un socio realiza actualmente alquileres de cintas
<b>Comentarios</b>	Ninguno

### A.2.28.2 Roles asociación realizaActualmente(Socio, Alquiler)

<b>Rol</b>	Alquiler juega rol alquilerActual en realizaActualmente(Socio, Alquiler)
<b>Descripción</b>	Alquileres realizados actualmente (aún no devueltos) por el socio
<b>Tipo OCL</b>	Set( Alquiler )
<b>Multiplicidad</b>	0..n
<b>Comentarios</b>	Ninguno

<b>Rol</b>	Socio juega rol socio en realizaActualmente(Socio, Alquiler)
<b>Descripción</b>	Socio que realiza el alquiler
<b>Tipo OCL</b>	Socio
<b>Multiplicidad</b>	1
<b>Comentarios</b>	Ninguno

### A.2.28.3 Invariante asociación realizaActualmente(Socio, Alquiler)

<b>Invariante</b>	realizaActualmente(Socio, Alquiler)
<b>Descripción</b>	• Un par ( <i>socio, alquiler</i> ) pertenece a esta asociación si y sólo si el socio tiene dicho alquiler sin devolver
<b>Expresión</b>	<pre>Socio.allInstances-&gt;forall( s     Alquiler.allInstances-&gt;forall( a       realizaActualmente(Socio, Alquiler).allInstances-&gt;       forall( ra           ( ra.socio = s and ra.alquiler = a ) =         ( s.alquiler-&gt;includes( a ) and a.socio = s and           not a.estáDevuelto ) ) ) )</pre>
<b>Comentarios</b>	Ninguno

## A.2.29 Asociación tiene entre Socio y Cuenta

### A.2.29.1 Descripción asociación tiene(Socio, Cuenta)

<b>Asociación</b>	<b>tiene entre Socio y Cuenta</b>
<b>Requisitos asociados</b>	• RI-03 Información sobre cuentas de socios
<b>Descripción</b>	Esta asociación representa el hecho de que un socio tiene una cuenta en el vídeo-club
<b>Comentarios</b>	Ninguno

### A.2.29.2 Roles asociación tiene(Socio, Cuenta)

<b>Rol</b>	<b>Cuenta juega rol cuenta en tiene(Socio, Cuenta)</b>
<b>Descripción</b>	Cuenta del socio
<b>Tipo OCL</b>	Cuenta
<b>Multiplicidad</b>	1
<b>Comentarios</b>	Ninguno

<b>Rol</b>	<b>Socio juega rol socio en tiene(Socio, Cuenta)</b>
<b>Descripción</b>	Socio titular de la cuenta
<b>Tipo OCL</b>	Cuenta
<b>Multiplicidad</b>	1
<b>Comentarios</b>	Ninguno

## A.2.30 Asociación tieneAlquilada entre Socio y Cinta

### A.2.30.1 Descripción asociación tieneAlquilada(Socio, Cinta)

<b>Asociación derivada</b>	<b>tieneAlquilada</b> entre <b>Socio</b> y <b>Cinta</b>
<b>Requisitos asociados</b>	• RI-02 Información sobre socios
<b>Descripción</b>	Esta asociación derivada representa el hecho de que un socio tiene actualmente cintas alquiladas (sin devolver)
<b>Comentarios</b>	Ninguno

### A.2.30.2 Roles asociación tieneAlquilada(Socio, Cinta)

<b>Rol</b>	<b>Cinta</b> juega rol <b>cinta</b> en <b>tieneAlquilada(Socio, Cinta)</b>
<b>Descripción</b>	Cintas alquiladas actualmente (aún no devueltas) por un socio
<b>Tipo OCL</b>	Set( Cinta )
<b>Multiplicidad</b>	0..n
<b>Comentarios</b>	Ninguno

<b>Rol</b>	<b>Socio</b> juega rol <b>socio</b> en <b>tieneAlquilada(Socio, Cinta)</b>
<b>Descripción</b>	Socio que tiene alquilada la cinta
<b>Tipo OCL</b>	Set( Cinta ) o Cinta
<b>Multiplicidad</b>	0..1
<b>Comentarios</b>	Los roles con cardinalidad 0..1 puede actuar como conjuntos o como objetos en OCL

### A.2.30.3 Invariante asociación tieneAlquilada(Socio, Cinta)

<b>Invariante</b>	<b>tieneAlquilada(Socio, Cinta)</b>
<b>Descripción</b>	• Un par ( <i>socio,cinta</i> ) pertenece a esta asociación si y sólo existe un alquiler no devuelto realizado por el socio en el que la cinta es alquilada
<b>Expresión</b>	<pre>Socio.allInstances-&gt;forall( s     Cinta.allInstances-&gt;forall( c       tieneAlquilada(Socio, Cinta).allInstances-&gt;forall( ta         ( ta.socio = s and ta.cinta = c ) =       ( s.alquilerActual.cinta-&gt;includes( c ) and         c.alquilerActual.socio = s ) ) ) )</pre>
<b>Comentarios</b>	Ninguno

## A.3 Operaciones del sistema

### A.3.1 Operación AltaDeSocio

#### A.3.1.1 Descripción operación AltaDeSocio

<b>Operación Sistema</b>	<b>AltaDeSocio</b>
<b>Requisitos asociados</b>	• UC-01 Alta de socio
<b>Descripción</b>	El empleado del vídeo-club da de alta un nuevo socio
<b>Tipo resultado</b>	Socio -- para imprimir el carnet
<b>Parámetros</b>	dni : String -- DNI del nuevo socio n : String -- nombre y apellidos del nuevo socio fn : Date -- fecha nacimiento del nuevo socio sx : enum{ Hombre, Mujer } -- sexo del nuevo socio d : String -- dirección del nuevo socio t : Set( Integer ) -- teléfonos del nuevo socio
<b>Precondiciones</b>	<b>pre<sub>1</sub></b> : No existe en el sistema un socio con el mismo número de DNI
<b>Precondiciones (OCL)</b>	<b>pre<sub>1</sub></b> : <code>not Socio.allInstaces-&gt;exist( s   s.dni = dni )</code>
<b>Postcondiciones</b>	<b>post<sub>1</sub></b> : Existe un nuevo socio cuyos atributos coinciden con los parámetros, que no tiene ningún alquiler y que tiene una nueva cuenta sin movimientos <b>post<sub>2</sub></b> : Sólo se ha creado un nuevo socio y una nueva cuenta <b>post<sub>3</sub></b> : El sistema informa de que el proceso ha terminado con éxito <b>post<sub>4</sub></b> : El resultado de la operación es el nuevo socio (para imprimir el carnet)
<b>Postcondiciones (OCL)</b>	<b>post<sub>1</sub></b> : <code>Socio.new-&gt;exists( s   s.dni = dni and s.nombre = n and s.fechaNacimiento = fn and s.sexo = sx and s.fechaAlta = Date.today and s.dirección = d and s.teléfonos = t and s.alquiler-&gt;isEmpty and Cuenta.new-&gt;exists( c   (s.cuenta = c) and (c.socio = s) and c.movimiento-&gt;isEmpty ) )</code> <b>post<sub>2</sub></b> : <code>Socio.new-&gt;size = 1 and Cuenta.new-&gt;size = 1</code> <b>post<sub>3</sub></b> : <code>respuesta = Set{"Proceso de alta de socio terminado con éxito"}</code> <b>post<sub>4</sub></b> : <code>Socio.new-&gt;includes( resultado )</code>
<b>Excepciones</b>	$\neg$ <b>pre<sub>1</sub></b> : El sistema informa de que el socio ya está registrado
<b>Excepciones (OCL)</b>	<code>not pre<sub>1</sub>: respuesta-&gt;includes( "Error: socio ya registrado" )</code>
<b>Comentarios</b>	Ninguno

### A.3.1.2 Diagrama de secuencia de AltaDeSocio



Figura 17: Diagrama de secuencia de *AltaDeSocio*

### A.3.1.3 Interfaz de usuario de AltaDeSocio

En el diálogo, el empleado del vídeo-club introduce los datos del nuevo socio. El número del socio y la fecha de alta son proporcionados por el sistema. Una vez que se finaliza pulsando el botón OK el sistema imprime el carnet de socio.

```

VÍdeo--Club Ejemplo                               Núm: 23456
-----
dirección y teléfono del vídeo--club

                                CARNET DE SOCIO

Nombre: Amador Durán Toro
Dirección: Departamento de Lenguajes y Sistemas Infor-
máticos, Avda. Reina Mercedes s/n, 41012 Sevilla

DNI: 12345678-X                                   Firma
Fecha alta: 21/12/1998

```

Figura 18: Interfaz de usuario de *AltaDeSocio*

## A.3.2 Operación AlquilarCinta

### A.3.2.1 Descripción operación AlquilarCinta

<b>Operación Sistema</b>	<b>AlquilarCinta</b>
<b>Requisitos asociados</b>	• UC-06 Alquiler de cintas de vídeo
<b>Descripción</b>	Un socio alquila una cinta de vídeo
<b>Tipo resultado</b>	Alquiler -- para imprimir el ticket
<b>Parámetros</b>	s : Socio -- socio que realiza el alquiler c : Cinta -- cinta que se alquila imp : Integer -- importe del alquiler fd : Date -- fecha de devolución hd : Time -- hora de devolución fp : enum{ Contado, Cuenta } -- forma de pago
<b>Precondiciones</b>	<b>pre<sub>1</sub></b> : La cinta no está registrada como alquilada
<b>Precondiciones (OCL)</b>	<b>pre<sub>1</sub></b> : not c.estáAlquilada@pre
<b>Postcondiciones</b>	<b>post<sub>1</sub></b> : Existe un nuevo alquiler no devuelto correspondiente al socio y a la cinta y existe también el cargo correspondiente en la cuenta del socio <b>post<sub>2</sub></b> : Sólo se ha creado un nuevo alquiler y un nuevo cargo <b>post<sub>3</sub></b> : Si se paga al contado se crea el ingreso oportuno en la cuenta del socio <b>post<sub>3</sub></b> : El sistema informa de que el proceso ha terminado con éxito <b>post<sub>4</sub></b> : El resultado de la operación es el nuevo alquiler (para imprimir el ticket)
<b>Postcondiciones (OCL)</b>	<b>post<sub>1</sub></b> : Alquiler.new->exists( a   (not a.estáDevuelto) and a.socio = s and a.cinta = c and a.fechaAlquiler = Date.today and a.horaAlquiler = Time.now and a.fechaDevolución = fd and a.horaDevolución = hd and CargoAlquiler.new->exists( ca   ca.cuenta = s.cuenta and ca.alquiler = a and ca.importe = -imp and ca.fecha = Date.today ) ) <b>post<sub>2</sub></b> : Alquiler.new->size = 1 and CargoAlquiler.new->size = 1 <b>post<sub>3</sub></b> : ( fp = #Contado ) = Ingreso.new->exists( i   i.cuenta = s.cuenta and i.fecha = Date.today and i.importe = imp ) <b>post<sub>4</sub></b> : <b>respuesta</b> = Set{ "Proceso de alquiler terminado con éxito" } <b>post<sub>5</sub></b> : Alquiler.new->includes( <b>resultado</b> )
<b>Excepciones</b>	$\neg$ <b>pre<sub>1</sub></b> : El sistema informa de que la cinta está registrada como alquilada
<b>Excepciones (OCL)</b>	not <b>pre<sub>1</sub></b> : <b>respuesta</b> ->includes( "Error: la cinta está registrada como alquilada" )
<b>Comentarios</b>	Ninguno

### A.3.2.2 Diagrama de secuencia de AlquilerCinta

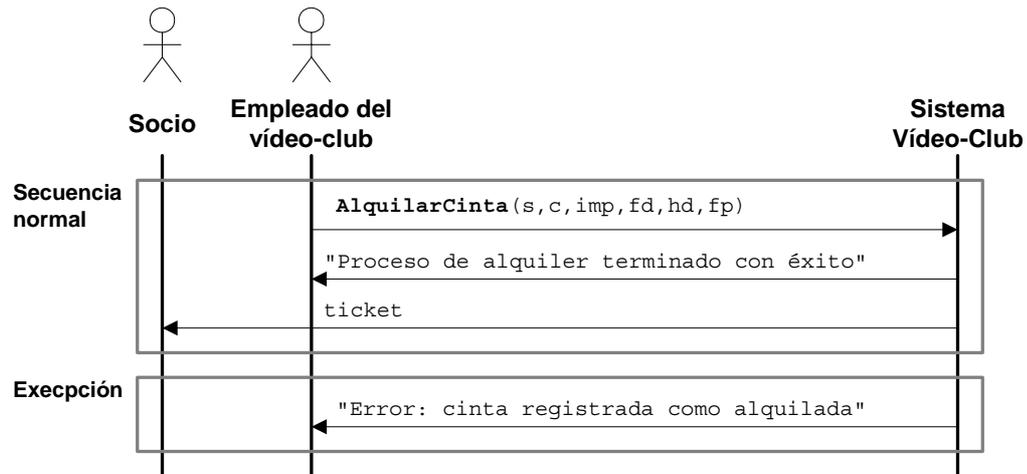


Figura 19: Diagrama de secuencia de *AlquilerCinta*

### A.3.2.3 Interfaz de usuario de AlquilerCinta

En el diálogo, el empleado del vídeo-club introduce el número del socio que realiza el alquiler y selecciona el nombre de la película. El sistema le muestra el nombre y el saldo del socio y le propone una cinta disponible (si hay) de la película. El sistema propone la fecha y hora de devolución así como el importe del alquiler. El empleado puede aceptar los valores propuestos o cambiarlos. Por último el empleado indica la forma de pago. Una vez que se finaliza pulsando el botón OK el sistema imprime el ticket.

**Alquiler de cinta**

Nº del socio: 2345

Nombre: Durán Toro, Amador Saldo: 1000

Película: La guerra de las galaxias Cinta: 1234

Fecha alquiler: 21/12/1998 Importe: 300

Hora alquiler: 18:05:32

Fecha devolución: 22/12/1998

Hora devolución: 21:30:00

Forma de pago:

- A cuenta
- Al contado

OK Cancel

Vídeo--Club Ejemplo

-----

dirección y teléfono del vídeo--club

**TICKET DE ALQUILER**

Socio: 23456 (Amador Durán Toro)

Cinta: 1234 (La guerra de las galaxias)

Fecha alquiler: 21/12/1998 18:05:32

Devolver antes de: 22/12/1998 21:30:00

Son 300 Ptas

Figura 20: Interfaz de usuario de *AlquilarCinta*

### A.3.3 Operación ConsultarPelícula

#### A.3.3.1 Descripción operación ConsultarPelícula

<b>Operación Sistema</b>	<b>ConsultarPelícula</b>
<b>Requisitos asociados</b>	• UC-10 Consulta de una película
<b>Descripción</b>	El empleado del vídeo-club consulta la información de una película
<b>Tipo resultado</b>	titulo:String × tema:enum{Infantil,Acción,Ciencia-Ficción,Terror,Adultos} × productora:String × protagonistas:Set(String) × disponible:Integer
<b>Parámetros</b>	p : Película -- la película a consultar
<b>Precondiciones</b>	pre <sub>1</sub> : Ninguna
<b>Precondiciones (OCL)</b>	pre <sub>1</sub> : true
<b>Postcondiciones</b>	post <sub>1</sub> : El resultado coincide con la información deseada post <sub>2</sub> : El sistema informa de que el proceso ha terminado con éxito
<b>Postcondiciones (OCL)</b>	<pre> post<sub>1</sub>: resultado.título = p.título           and         resultado.tema = p.tema                 and         resultado.productora = p.productora.nombre and         resultado.protagonistas =           p.protagonistas.nombre-&gt;asSet         and         resultado.disponibles = p.disponible-&gt;size post<sub>2</sub>: respuesta = Set{ "Proceso de consulta terminado con éxito" } </pre>
<b>Comentarios</b>	Ninguno

### A.3.3.2 Interfaz de usuario operación ConsultarPelícula

Consultar Película

Título: La guerra de las galaxias

Tema: Ciencia-Ficción

Productora: 20th Century Fox Año: 1977

Protagonistas: Alec Guinness, Harrison Ford, Mark Hamill

Cintas disponibles: 2

Buttons: [Printer], [>>], [<<], [OK]

```
Consulta de película                                21/12/1998
-----
Título:      La Guerra de las Galaxias
Tema:        Ciencia-Ficción
Productora:  20th Century Fox (1977)
Protagonistas:
             Mark Hamill
             Harrison Ford
             Alec Guinness
Cintas disponibles: 2
```

Figura 21: Interfaz de usuario del operación *ConsultarPelícula*

En el diálogo, el empleado del vídeo-club introduce el nombre de la película que quiere consultar y el sistema le muestra la información asociada. El empleado puede avanzar o retroceder en orden alfabético por la películas utilizando los botones » y «. Si lo desea, puede imprimir los datos de la película pulsando el botón con una imagen de impresora. La consulta finaliza una vez que se pulsa el botón OK.

## A.4 Conflictos detectados en los requisitos-C

Como resultado del análisis realizado, se deberían realizar los siguientes cambios en los requisitos-C originales:

<b>CFL-1</b>	<b>Necesidad de un identificador de cintas</b>
<b>Objs./Reqs. en conflicto</b>	<ul style="list-style-type: none"> <li>• RI-01 <i>Información sobre películas</i></li> <li>• UC-05 <i>Alta de cinta de vídeo</i></li> </ul>
<b>Descripción</b>	Es necesario que cada cinta tenga un identificador (un número, una combinación de letras y números, ...), para que el empleado del vídeo-club las pueda localizar en las estanterías
<b>Alternativas</b>	<ul style="list-style-type: none"> <li>• Crear un nuevo requisito para registrar específicamente la necesidad de que el sistema genere y conozca el identificador de cada cinta (-)</li> <li>• Añadir al requisito RI-01 un nuevo dato específico que recoja la necesidad del identificador de cinta (-)</li> </ul>
<b>Solución</b>	Crear un nuevo requisito RI-04 <i>Información sobre cintas</i> que recoja la necesidad del identificador
<b>Comentarios</b>	Ninguno

<b>CFL-2</b>	<b>Necesidad de especificar la información sobre alquileres</b>
<b>Objs./Reqs. en conflicto</b>	<ul style="list-style-type: none"> <li>• RI-02 <i>Información sobre socios</i></li> <li>• RI-03 <i>Información sobre cuentas de socios</i></li> <li>• UC-06 <i>Alquiler de cintas de vídeo</i></li> <li>• UC-08 <i>Devolución de cintas de vídeo</i></li> </ul>
<b>Descripción</b>	No se describe en ningún requisito la información que se debe almacenar sobre los alquileres de cintas
<b>Alternativas</b>	<ul style="list-style-type: none"> <li>• Crear un nuevo requisito de almacenamiento de información para registrar específicamente la información que se debe almacenar sobre los alquileres (-)</li> </ul>
<b>Solución</b>	Crear un nuevo requisito RI-05 <i>Información sobre alquileres</i> que describa la información a almacenar sobre los alquileres
<b>Comentarios</b>	Ninguno

<b>CFL-3</b>	<b>Especificar que un pago en metálico es un ingreso</b>
<b>Objs./Reqs. en conflicto</b>	<ul style="list-style-type: none"> <li>• RI-03 <i>Información sobre cuentas de socios</i></li> <li>• UC-06 <i>Alquiler de cintas de vídeo</i></li> </ul>
<b>Descripción</b>	No queda claro que cuando el socio paga en metálico se considera un ingreso en su cuenta
<b>Alternativas</b>	<ul style="list-style-type: none"> <li>• Modificar el requisito RI-03 para aclarar el concepto de ingreso y para cambiar el intervalo temporal a pasado y presente (-)</li> </ul>
<b>Solución</b>	Modificar el requisito RI-03 y dejar también claro el concepto de <i>pago pendiente</i> , definiéndolo como aquellos cargos para los que no hay saldo suficiente
<b>Comentarios</b>	Ninguno