

Modelado de Anomalías mediante Escenarios

Marcela Ridao ⁽¹⁾, Jorge Doorn ⁽¹⁾⁽²⁾

⁽¹⁾ INTIA, Fac. de Ciencias Exactas -
Univ. Nacional del Centro de la Provincia de Buenos Aires

⁽²⁾ Universidad Tecnológica Nacional, FRBA
email: {jdoorn, mridao}@exa.unicen.edu.ar

Abstract. La importancia del manejo de anomalías por parte de los artefactos de software y el impacto que aquéllas provocan en el desarrollo de los mismos es ampliamente reconocida. El ingeniero de software no sólo debe esforzarse por lograr la máxima confiabilidad del sistema en condiciones de uso normal, sino también tratar de proteger la aplicación de los inconvenientes que surgen del uso de la misma bajo condiciones adversas. Sin embargo, las circunstancias anómalas en las que se va a desempeñar el software son muy poco percibidas en la elicitación de requisitos del Universo de Discurso (UdeD). La Ingeniería de Requisitos se interesa en la elicitación, modelado y análisis de las características que el sistema a desarrollar debe presentar. La falta de anticipación de contextos excepcionales da por resultado requisitos no realistas, inalcanzables y/o incompletos. Como consecuencia, el software desarrollado a partir de esos requisitos no será lo suficientemente robusto, con consecuencias a menudo críticas. En el presente artículo, se analizan las diferentes formas de representación de anomalías utilizando Escenarios y los beneficios que se pueden obtener utilizando patrones.

1 Introducción

Los programas computacionales son descripciones formales del mundo real. Por ello, a la hora de crear un sistema de software, se hace necesario dar descripciones formales de aquellas partes del mundo que interesa describir. Esto es difícil porque el mundo es infinito: por mucho que se describa, siempre hay más. “Cada regla tiene una excepción”, se dice. Por cuidado que se ponga en construir una regla, en tener en cuenta cada aspecto, siempre hay algo que queda afuera y que puede llevar a un absurdo. Por esto, las reglas y formalizaciones acerca del mundo real informal son siempre imperfectas. La dificultad de formalizar lo informal es real. Hacerlo bien es una gran parte - a menudo la parte más grande y más importante -, de lo que se necesita para construir un buen sistema de software [1].

Los programas fallan principalmente por dos razones: errores lógicos en el código y excepciones. Estas últimas pueden representar hasta las dos terceras partes de las fallas de un sistema [2], por lo que son merecedoras de especial atención.

La importancia del manejo de excepciones es bien reconocida por los diseñadores de sistemas e ingenieros de software. El manejo de excepciones es muy a menudo la parte más importante del sistema. El objetivo de los mecanismos de manejo de excepciones es hacer programas robustos y confiables. Sin embargo, muchas fallas

son causadas por manejo incompleto o incorrecto de estas anomalías. Esto ocurre, como ha sido reportado por diversos autores [3], por una variedad de razones, entre las cuales no es menor el hecho que más de la mitad del código es a menudo dedicado a la detección y manejo de excepciones. El análisis de accidentes en sistemas computacionales ha mostrado que muy a menudo sus causas tienden a ser el manejo inadecuado de circunstancias excepcionales. Los requisitos para el comportamiento correcto del sistema durante el manejo de excepciones están en algún sentido por encima de los del sistema operando en modo normal. Esto implica que el manejo de excepciones necesita ser más poderoso que lo que suele creerse.

Un proceso de desarrollo de software sistemático seguido por una implementación rigurosa y disciplinada, es la mejor forma de dominar la complejidad del software y lograr confiabilidad en los productos. Desafortunadamente, no importa el cuidado que se ponga en el desarrollo, nunca se puede confiar incondicionalmente en el software. Por ello, el ingeniero de software no sólo debe esforzarse por lograr la máxima confiabilidad al desarrollar un sistema, sino también tratar de proteger a la aplicación de los errores que –desafortunadamente– pueden aparecer durante el desarrollo o pueden surgir debido a circunstancias adversas durante la ejecución del programa. Es decir, se deben construir sistemas robustos. Los programas deberían continuar comportándose razonablemente aun en el caso de circunstancias inesperadas e imprevistas [4].

En virtud que en muchos casos, las anomalías se descubren en forma muy tardía, es bastante común que se las incorpore en el final de la fase de programación o, aún peor, en las fases de implantación o mantenimiento. Esto agrava el problema, ya que se incorporan como parches al sistema con notable deterioro en la calidad del producto de software, introduciendo redundancia en el código por la falta de generalización en su tratamiento. A esto se suman el bajísimo grado de documentación y la falta de integración apropiada.

La Ingeniería de Requisitos se interesa en la definición de objetivos de alto nivel a ser logrados por el sistema a desarrollar, en el refinamiento de tales objetivos y su traducción en especificaciones de servicios y restricciones y la asignación de responsabilidades a agentes humanos, dispositivos y software, según los requisitos resultantes. Los procesos de la Ingeniería de Requisitos dan por resultado, a menudo, objetivos, requisitos, y consideraciones acerca del comportamiento de dichos agentes demasiado ideales; es probable que algunos de ellos no sean satisfechos durante la ejecución del sistema debido a algún comportamiento inesperado. La falta de anticipación de comportamientos excepcionales da por resultado requisitos no realistas, inalcanzables y/o incompletos. Como consecuencia, el software desarrollado a partir de esos requisitos no será lo suficientemente robusto e inevitablemente provocará performance pobre o fallas, con consecuencias a menudo críticas [5].

Las anomalías son muy poco percibidas durante la comprensión del UdeD y la captura (elicitación) de requisitos del sistema de software. Las técnicas de modelado de requisitos no prestan especial atención a este tipo de circunstancias. Por ello, aunque su tratamiento es muy importante para el éxito de un sistema de información, son poco modeladas en los documentos de requisitos. En particular, constituyen uno de los aspectos peor atendidos en el proceso de construcción de escenarios.

En este trabajo, se pretende mejorar la detección de excepciones y definir patrones que faciliten dicha tarea.

2 Escenarios y Patrones de Escenarios

El uso de escenarios como una técnica para entender el problema a resolver usando un sistema de software ha sido recomendado por numerosos autores [6], [7], [8], [9]. Los escenarios describen situaciones teniendo en cuenta aspectos de uso, permitiendo: conocer el problema, unificar criterios, ganar compromiso con clientes/usuarios, organizar los detalles involucrados y entrenar a nuevos participantes.

Aunque se han propuesto diferentes representaciones y técnicas para la construcción de escenarios, existe una falta de precisión acerca de cuándo y cómo los escenarios deberían ser usados, y que ha sido extendida a los ingenieros que están usando estas técnicas en la práctica. La mayoría de los desarrolladores ven la creación de escenarios más como una tarea artesanal que como un trabajo de ingeniería. Estudios recientes respecto al uso real de escenarios en la Ingeniería de Requisitos [10] [11] han probado claramente este hecho, apuntando a la necesidad de definiciones más detalladas acerca de la construcción de escenarios como una contribución inevitable para incrementar su uso en casos reales.

Una de las posibles formas de construir escenarios es basarse en el vocabulario del UdeD. Este vocabulario refleja las palabras peculiares y más usadas en el mismo. Un modelo que permite registrar ese vocabulario es el LEL (Léxico extendido del Lenguaje) [12] [13]. Se trata de una estructura especial, compuesta por un conjunto de símbolos que representan el lenguaje de la aplicación y está basada en una idea muy simple: entender el lenguaje del problema, sin preocuparse por entender el problema en sí.

El proceso de construcción de escenarios en el marco del cual se desarrolla este trabajo, comienza partiendo desde el léxico del dominio de la aplicación, produciendo una primera versión de los escenarios derivados exclusivamente desde el LEL [14]. Estos escenarios son entonces completados desde diferentes fuentes de información y organizados para obtener un conjunto de escenarios consistente que representa el dominio de la aplicación. La estructura utilizada para representar un escenario contiene los siguientes elementos: Título, Objetivo, Contexto, Recursos, Actores, Episodios y Excepciones [15].

El proceso de construcción de escenarios se lleva a cabo mediante las siguientes actividades: *Derivar*, *Describir*, *Organizar*, *Verificar* y *Validar*.

En la actividad *Describir*, presentada en la figura 1, se completan los escenarios candidatos agregando información del UdeD utilizando las heurísticas de descripción y tomando como base el modelo de escenario y los símbolos del LEL. El resultado es un conjunto de escenarios candidatos completamente descriptos. Esta etapa consta de cuatro actividades, como puede verse en la siguiente figura: *Completar componentes*, *Crear subescenarios*, *Completar restricciones* y *Completar Excepciones*. Es en las dos últimas actividades, donde se tratan las anomalías.

Las restricciones se utilizan para caracterizar requisitos no funcionales aplicados a Contexto, Recursos y Episodios. Algunas pueden ser elicidadas desde el UdeD y otras pueden surgir examinando los episodios.

En la actividad *Completar Excepciones*, finalmente, se deben detectar los casos alternativos y excepciones. Algunas causas de excepción son elicidadas desde las fuentes de información mientras que otras pueden deducirse analizando los episodios y la no disponibilidad o malfuncionamiento de los recursos. Cuando se descubren las

causas de una excepción, los ingenieros de requisitos deberían investigar cómo es tratada la excepción en el UdeD, surgiendo así una nueva situación que podría necesitar ser descrita a través de un escenario separado.

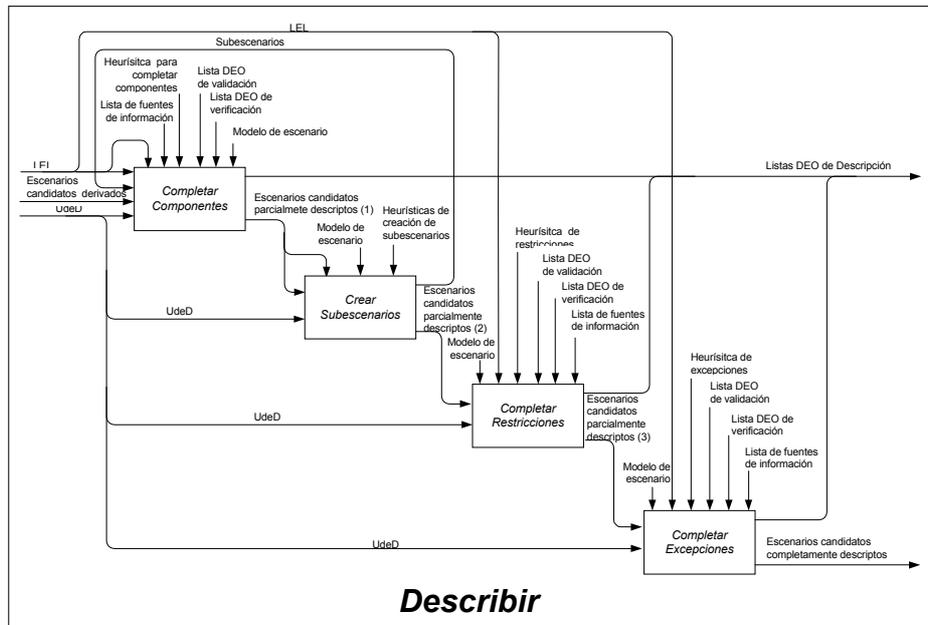


Fig. 1. SADT de la etapa Describir Escenarios

A la heurística correspondiente a este proceso de construcción se ha incorporado un catálogo de patrones de escenarios [16] [17]. De este modo, se logró aportar la experiencia acerca de situaciones similares acumulada en los patrones a las técnicas existentes. El proceso así modificado, analiza las situaciones desde un punto de vista estructural, asociándolas, completa o parcialmente, con uno o más patrones de un catálogo. Una vez determinado el patrón que más se aproxime a la situación real, se reusa su estructura con el fin de derivar el escenario más fácil y directamente.

Sin embargo, ni la heurística original ni los patrones definidos hasta el momento prestan especial atención al tratamiento de las anomalías. Por ello, en este trabajo se analiza el problema de definir patrones que faciliten estas actividades, y permitan detectar con mayor facilidad las anomalías en una etapa temprana del desarrollo de Software.

3 Modelado de Anomalías con Escenarios

Con el fin de mejorar la detección de anomalías del UdeD en el proceso de construcción de escenarios, se analizaron los escenarios correspondientes a diferentes

casos de estudio, poniendo énfasis en las anomalías presentes en los componentes Episodios y Excepciones. Los casos analizados fueron:

Sistema Nacional para la Emisión de Pasaportes [18] [19]: desarrollado en la Universidad de Belgrano.

Sistema de Agenda de Reuniones [20]: desarrollado en la Universidad de Belgrano, basándose en el caso de estudio propuesto por van Lamsweerde et al. [21].

Sistema de Plan de Ahorro para la Adquisición de Automóviles [22] [23]: desarrollado en INTIA, Universidad Nacional del Centro de la Pcia. de Buenos Aires.

Sistema para el Control de Pos Graduación [24]: desarrollado en la Pontificia Universidade Católica do Rio de Janeiro para el Dto. de Informática de PUC-Río.

Sistema de Biblioteca: desarrollado en la Pontificia Universidade Católica do Rio de Janeiro.

Sistema Editor de Texto: desarrollado en la Pontificia Universidade Católica do Rio de Janeiro.

Uso recursivo de las heurísticas para la construcción de LEL y Escenarios del proceso de construcción de LEL y Escenarios [25]: llevado a cabo en INTIA, Universidad Nacional del Centro de la Provincia de Buenos Aires.

Almacén de una fábrica [26]: desarrollado en INTIA, Universidad Nacional del Centro de la Provincia de Buenos Aires.

A partir de este análisis se pudo determinar la existencia de diferentes tipos de anomalías con características bien definidas.

3.1 Modelado de Anomalías a Nivel Episodio

Pudo observarse que, si el punto del inconveniente puede precisarse en términos del episodio en que se produce, entonces la anomalía se representa como episodio condicional o como restricción al episodio.

Episodios condicionales. Cuando existe un tratamiento específico para la anomalía, se lo modela con episodios condicionales, utilizando la siguiente estructura:

<p>SI no se produce la anomalía ENTONCES PROCESO NORMAL</p> <p>SI se produce la anomalía ENTONCES PROCESO ESPECIAL</p>
--

En la figura 2, se presenta un ejemplo de este tipo de anomalías correspondiente al caso de estudio “Biblioteca”.

<p>Título: RETIRAR OBRA.</p> <p>Objetivo: Retirar obra</p> <p>Contexto: <i>Ubicación Geográfica:</i> Biblioteca. <i>Ubicación Temporal:</i> Horario de funcionamiento de la Biblioteca.</p> <p>Actores: Bibliotecario Usuario.</p> <p>Recursos: Obra. Carnet del usuario</p> <p>Episodios: El usuario elige la obra. El usuario solicita la obra al bibliotecario. El bibliotecario verifica la disponibilidad de la obra. SI al menos un ejemplar de la obra está disponible ENTONCES el bibliotecario obtiene identificación del usuario. SI al menos un ejemplar de la obra está disponible ENTONCES el bibliotecario registra el préstamo en la ficha de la obra. SI al menos un ejemplar de la obra está disponible ENTONCES ... SI ningún ejemplar de la obra está disponible ENTONCES el bibliotecario ofrece al usuario la opción de reserva.</p>

Fig. 2. Ejemplo de anomalía a nivel episodio con tratamiento especial

Restricciones. Cuando no existe un tratamiento específico para la anomalía, se lo modela por medio de restricciones al episodio en cuestión, utilizando la siguiente estructura:

<p>Episodios: Episodio 1 ... Episodio n Restricción: no debe producirse la anomalía Episodio n + 1 ...</p>
--

En este caso, si las restricciones se cumplen, continúa el procesamiento normal del escenario. En caso contrario, es decir, cuando se presenta la anomalía, el escenario se cancela, impidiéndose el cumplimiento de su objetivo, ya que no existe un tratamiento especial para la misma.

A continuación, se presenta otro escenario correspondiente al caso de estudio “Almacén de una Fábrica”, que ejemplifica este tipo de anomalías. En este caso, pueden producirse dos inconvenientes diferentes en el mismo episodio.

<p>Título: ATENDER SOLICITUD DE EGRESO DE MERCADERIA.</p> <p>Objetivo: Entregar mercadería al <u>Usuario solicitante</u>.</p> <p>Contexto: <i>Ubicación Geográfica:</i> Oficina de recepción del Almacén. <u>Depósito</u>. <i>Ubicación Temporal:</i> Jornada laboral del <u>Personal de mostrador</u>. <i>Precondiciones:</i> Debe haberse PROCESADO EL <u>ALTA DE REMITO DEL PROVEEDOR</u>.</p> <p>Actores: <u>Personal de mostrador</u>. <u>Usuario solicitante</u>.</p> <p>Recursos: <u>Formulario de Vale de Retiro de Almacén</u>. <u>Listado del personal autorizado a firmar</u>. <u>Material de depósito</u>. <u>Material que no es de depósito</u>.</p> <p>Episodios: El <u>Usuario solicitante</u> le entrega al <u>Personal de mostrador</u> un <u>Formulario de Vale de Retiro de Almacén</u>. Restricción I: El <u>Formulario de Vale de Retiro de Almacén</u> debe estar firmado. Restricción II: La firma del <u>Usuario solicitante</u> debe pertenecer al <u>Listado del personal autorizado a firmar</u>. El <u>Personal de mostrador</u> recibe el <u>Formulario de Vale de Retiro de Almacén</u>. ...</p>
--

Fig. 3. Ejemplo de anomalía a nivel episodio sin tratamiento especial

3.2 Modelado de Anomalías a Nivel Escenario

A partir del estudio de los escenarios de los casos de estudio pudo observarse también que, cuando el punto del inconveniente es impreciso, en el sentido que puede ocurrir en más de un episodio del escenario, entonces se modela como una Excepción. En este caso, también puede darse que exista o no un tratamiento para la anomalía.

En la figura 4 se presenta un ejemplo de escenario correspondiente al caso de estudio “Sistema Nacional para la Emisión de Pasaportes”, con una excepción sin tratamiento. Este ejemplo contiene además, anomalías a nivel episodio que fueron presentadas en la sección anterior.

En este caso, como no existe un tratamiento especial para la anomalía, si ésta se produce, se cancelará el escenario. El efecto neto es idéntico al no cumplimiento de una restricción en un episodio sin tratamiento especial.

Cuando existe un tratamiento especial para una anomalía a nivel escenario, dicho tratamiento puede presentar tres formas diferentes. Es posible que se preserve el objetivo del escenario, es decir que el tratamiento implique la ejecución de una acción alternativa que logre el mismo efecto final que el comportamiento normal del escenario. Puede suceder, también, que el tratamiento de la anomalía contenga una acción cuyo objetivo sea diferente al del escenario donde se originó. Y, finalmente, el tratamiento puede ser una mera secuencia de actividades destinadas a restaurar las condiciones iniciales, es decir que se ejecutarán acciones que irán deshaciendo las actividades que se habían llevado a cabo hasta el momento en que se produjo el inconveniente.

<p>Título: SACAR FOTOGRAFIA</p> <p>Objetivo: Obtener la fotografía del <u>solicitante</u>.</p> <p>Contexto: <i>Ubicación Geográfica:</i> Se efectúa en la <u>Cabina de Fotografía</u>. <i>Precondiciones:</i> El <u>solicitante</u> debe tener el <u>formulario de solicitud</u> con sus datos.</p> <p>Recursos: Formulario de solicitud Sello Cámara fotográfica</p> <p>Actores: <u>Solicitante</u> Empleado de <u>caja</u> Fotógrafo</p> <p>Episodios: El <u>solicitante</u> pasa a un sector donde hay sillas con un display sobre ellas que indican el número de fotografía policial que tiene el <u>formulario</u>. Se sienta en la que le corresponde. El fotógrafo saca 4 copias y entrega al <u>solicitante</u> 3. La cuarta la guarda en una <u>caja</u>. <i>Restricción:</i> tener asignado un número de fotografía policial en el <u>formulario de solicitud</u></p> <p>Excepciones: La cámara fotográfica no funciona.</p>

Fig. 4. Ejemplo de anomalía a nivel escenario sin tratamiento especial

En la figura 5, se presenta una taxonomía para los diferentes tipos de inconvenientes presentados y las diferentes maneras de representarlos en un escenario, según exista o no tratamiento para los mismos:

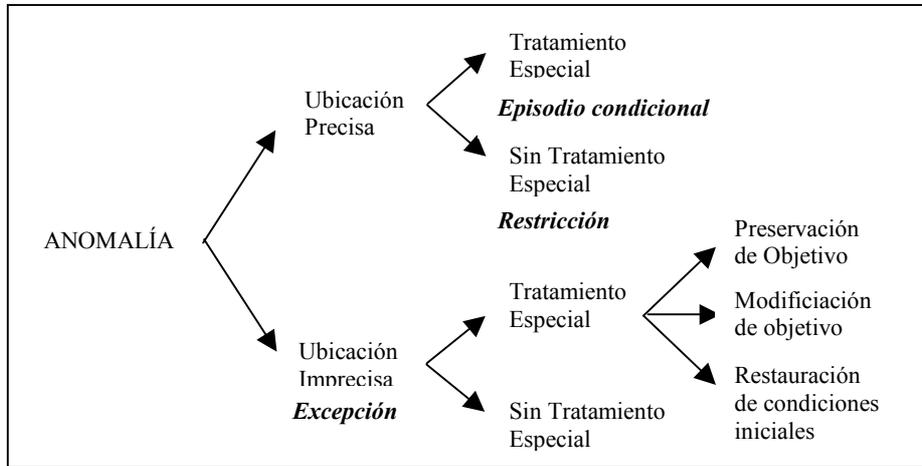


Fig. 5. Clasificación de anomalías según su ubicación y tratamiento

4 Casos de Estudio

Un análisis de los escenarios correspondientes a siete casos de estudio permitió observar los siguientes resultados:

Tabla 1. Anomalías detectadas en los casos de estudio analizados

Caso de Estudio	Número de Escenarios	Ubicación precisa		Ubicación imprecisa Excepción			
		Con tratamiento especial Episodio condicional	Sin tratamiento especial Restricción	Tratamiento especial			Sin tratamiento especial
				Preserva objetivo	Modifica objetivo	Restaura condición inicial	
Pasaportes	19	1	16	0	0	0	16
Agenda de Reuniones	14	1	8	2	0	1	7
Plan de Ahorro	24	4	10	0	1	1	1
Pos Graduación	8	0	5	3	0	0	0
Biblioteca	4	3	2	0	0	0	3
Autoaplic. Metodo	64	2	2	0	0	0	0
Almacén	37	4	2	0	0	0	1

Estos datos permiten corroborar que, indudablemente, este es un aspecto muy pobremente tratado en el proceso habitual de elicitación de requisitos. Esto se evidencia en particular para las anomalías a nivel escenario. Justamente, el objetivo de este trabajo es aportar soluciones a esta tendencia a minimizar el estudio de anomalías.

5 Patrones de Anomalías

En la presente sección se analiza el grado de patronización posible para las anomalías a nivel escenario y se describen algunas regularidades de anomalías a nivel episodio.

La incorporación de patrones al proceso de construcción de escenarios permite reconocer un patrón para cada situación y así poder refinar y corregir la definición inicial con el fin de describir más precisamente la realidad, y poder derivar un producto de software de mejor calidad.

En [27] [28] se estudiaron diferentes conjuntos de escenarios con el fin de detectar características recurrentes que permitieran identificar la naturaleza intrínseca de las situaciones. De esta manera, fue posible determinar el tipo de escenario correspondiente a cada situación y así, definir un patrón para representar, total o parcialmente a cada una de ellas. En la definición de los patrones se buscó tener una estructura básica, que pudiera ser reusada, de modo de facilitar la construcción.

Así, se definieron los siguientes patrones: *Producción*, con un solo actor llevando a cabo una actividad productiva que provocará un efecto sobre el macrosistema, *Colaboración*, con dos o más actores realizando una actividad cooperativa con un objetivo común, *Servicio*, que describe la prestación de un servicio en beneficio de uno o más actores, y *Negociación* con dos o más actores ejecutando una secuencia coordinada de acciones. Las Negociaciones, dependiendo de si la situación comienza, continúa, finaliza o se desarrolla completamente en el escenario, se clasifican en *Negociación Inconclusa*, *Etapa de Negociación*, *Fin de Negociación* o *Negociación Terminada*, respectivamente.

Cuando la actividad queda inconclusa en un escenario, es posible que su finalización dependa de varias situaciones futuras. En ese caso, los patrones serán *Negociación Inconclusa con Disparador de Escenarios* y *Etapa de Negociación con Disparador de Escenarios*.

Se observa que existen situaciones donde se presentan combinaciones de diferentes tipos de episodios, dando lugar a patrones compuestos, como por ejemplo, *Negociación inconclusa con producción o servicio o colaboración*.

Los patrones definidos consisten en un texto guía, que para cada sección del escenario incluye pautas acerca del contenido que deberá tener dicha sección. Por ejemplo, para la sección de episodios, se da una descripción general del tipo de episodios, dando pautas acerca de la cantidad de episodios de cada tipo que deberán aparecer en el escenario y el orden en que deberán escribirse [29].

5.1 Excepciones

Toda anomalía tiene necesariamente una causa. Esta causa, directa o indirectamente, afecta un Recurso, un Actor, el Contexto o una combinación más compleja de estos elementos. Estas combinaciones se presentan de muchas maneras, existiendo habitualmente un fenómeno de cascada. Por ejemplo, una causa cualquiera provoca la indisponibilidad de un recurso. Esto impide la realización del escenario “normal”. Sin embargo, se encuentra previsto un escenario alternativo que preserva el objetivo del escenario original utilizando otro recurso. Este nuevo recurso puede exigir un cambio en la ubicación geográfica o temporal, o eventualmente, la participación de un actor que no participaba en el escenario original.

Al analizar escenarios que contienen excepciones (anomalías a nivel escenario) y los escenarios que tratan dichas excepciones, se observa inmediatamente que los patrones de ambos escenarios están fuertemente vinculados, ya que se dispone de escasos grados de libertad. Por ejemplo, si se produce una excepción en un escenario de tipo Producción por causa de un inconveniente relacionado con un actor, el escenario corrector que preserva el objetivo podrá ser otra Producción o una Colaboración. Los casos más relevantes de las combinaciones Escenario Original – Escenario de Tratamiento, en función de la causa de la excepción, se muestran en la Tabla 2.

Table 2. Relación entre los patrones para Escenario Original – Escenario de Tratamiento

		Contexto	Actores	Recursos
Con tratamiento especial	Preserva objetivo	Mismo patrón	Puede haber cambio de patrón (Tabla 3)	Mismo patrón
	Restaura condiciones iniciales	Mismo patrón	Puede haber cambio de patrón (Tabla 3)	Mismo patrón
	Nuevo objetivo	-----	-----	-----
Sin tratamiento		-----	-----	-----

En la tabla 3 se presentan los patrones de los escenarios de tratamiento para excepciones en escenarios de tipo Producción, Colaboración, Servicio y Negociación frente a modificaciones en el Contexto, Recursos o Actores. Este análisis corresponde a las dos primeras filas de la tabla anterior, es decir, a los casos en que se preserva el objetivo o se restauran las condiciones iniciales.

Debe notarse que, si una modificación en el contexto o recursos provoca un cambio en los actores, pueden producirse también los cambios de patrón indicados en la segunda columna.

Table 3. Relación entre Escenario Original – Escenario de Tratamiento para los casos Preservación de Objetivo y Restauración de Condiciones Iniciales

	Contexto	Actores	Recursos
Producción	Producción	Colaboración (actor reemplazado por varios actores) Producción (actor reemplazado por otro)	Producción
Colaboración	Colaboración	Producción (actores originales reemplazados por un solo actor) Colaboración (uno o más actores reemplazados por otro/s)	Colaboración
Servicio	Servicio	Servicio (actor/es activo/s reemplazado/s por otro/s)	Servicio
Negociación	Negociación	Negociación	Negociación

En el caso de las negociaciones, puede afirmarse que en cualquiera de los casos analizados en la tabla 3, el escenario de tratamiento será una nueva negociación. Es decir, si el escenario original tiene como objetivo la realización de una actividad que requiere una secuencia coordinada de acciones por parte de los actores, el escenario de tratamiento, ya sea que preserve el objetivo o restaure las condiciones iniciales, también requerirá una secuencia coordinada de acciones.

Sin embargo, dada la naturaleza de las negociaciones, es posible que una modificación en el contexto, actores o recursos, origine modificaciones en el tipo de negociación. Una actividad que, en su desarrollo normal, comienza, continúa o finaliza dentro de un escenario y se ve interrumpida por una excepción, puede ser tratada por un nuevo escenario, que, en las nuevas condiciones, continúe con la actividad iniciada pero deje una situación inconclusa, finalice la actividad iniciada, comience y finalice una nueva negociación que restaure las condiciones iniciales, etc., todo esto independientemente del tipo de negociación que resuelva. Si, por ejemplo, se produce una excepción en un escenario de tipo Negociación Inconclusa, el escenario de tratamiento podría ser de tipo Fin de Negociación.

Table 4. Ejemplos de relación entre Escenario Original – Escenario de Tratamiento

Caso de Estudio	Escenario Original	Escenario de Tratamiento
Agenda de Reuniones	<p><i>Título:</i> DESEÑAR LA AGENDA DE REUNIONES</p> <p><i>Objetivo:</i> Asegurar el desarrollo eficiente de la reunión</p> <p>Patrón: Producción y Servicio</p> <p><i>Excepción:</i> Conflictos en la disponibilidad de material físico</p>	<p><i>Título:</i> ANULAR LA REUNIÓN</p> <p><i>Objetivo:</i> Liberar la agenda de la reunión que se cancela</p> <p>Patrón: Producción y Servicio (Restauración de condiciones iniciales)</p>

Agenda de Reuniones	<p><i>Título:</i> ORGANIZAR LA REUNIÓN</p> <p><i>Objetivo:</i> Determinar los requerimientos, oportunidad y lugar de la reunión</p> <p>Patrón: Producción</p> <p><i>Excepción:</i> Conflictos en los horarios disponibles de los convocados</p>	<p><i>Título:</i> TRASLADAR LA FECHA DE LA REUNIÓN</p> <p><i>Objetivo:</i> Actualizar la agenda por modificación de la fecha de la reunión</p> <p>Patrón: Producción (Preservación de Objetivo)</p>
Plan de Ahorro	<p><i>Título:</i> LIQUIDAR INTERESES PUNITORIOS</p> <p><i>Objetivo:</i> Regularizar la situación de un adherente que ha incurrido en mora.</p> <p>Patrón: Negociación Terminada con Producción</p> <p><i>Excepción:</i> Falta de cumplimiento de pago durante tres meses consecutivos o alternados</p>	<p><i>Título:</i> RESOLVER LA SOLICITUD DE ADHESIÓN</p> <p><i>Objetivo:</i> Dar por resuelta la solicitud de un adherente</p> <p>Patrón: Negociación Inconclusa con Producción (Restauración de condiciones iniciales)</p>

En la tabla 4 se muestra la relación entre los escenarios original y de tratamiento para excepciones detectadas en algunos de los casos de estudio analizados.

5.2 Episodios Condicionales

Cuando aparece en un escenario una anomalía registrada mediante episodios condicionales se está en presencia de un caso similar a las excepciones en el sentido que el objetivo del escenario puede o no preservarse. Esta similitud es natural ya que se trata de dos formas de representar anomalías que comparten el hecho de tener un tratamiento especial, pero difieren en que se puede o no identificar con precisión su ubicación.

Hay, sin embargo, un aspecto que oscurece el modelado de los episodios condicionales y que no ocurre con las excepciones. En estas últimas existe un registro específico que indica si se preserva o no el objetivo del escenario original, mientras que en los primeros esto no es tan visible.

En el caso de los episodios condicionales, si se preserva de alguna manera el objetivo del escenario, esto puede ocurrir o no en el mismo escenario. Si no ocurre en el mismo escenario, existirá entonces un escenario complementario similar en su función a los escenarios de tratamiento para las excepciones, aunque su percepción es mucho menos evidente en el modelo. Algo similar ocurre si se restablecen las condiciones iniciales o si se persigue un nuevo objetivo.

Los escenarios que contienen anomalías no resueltas en el mismo escenario muestran dos características perceptibles:

- Existe al menos un episodio condicional no consistente con el objetivo del escenario.
- El Escenario Integrador que contiene al escenario en cuestión contendrá, en forma explícita, la condición que refleja la anomalía como condicional del escenario de tratamiento.

6 Conclusiones y Trabajo Futuro

La patronización de las excepciones y sus tratamientos queda naturalmente incluida en el conjunto de patrones existente, con la ventaja adicional que para cada combinación de patrón del escenario original y causa se reduce significativamente el número de patrones candidatos para el escenario de tratamiento de la excepción.

En el caso que el nuevo escenario persiga un objetivo diferente al del escenario original, poco puede precisarse acerca de las características del escenario de tratamiento de la excepción.

En el caso de los episodios condicionales originados en anomalías, se ha encontrado que existen claras regularidades cuya inclusión en los patrones existentes será objeto de futuros trabajos.

Además, con el fin de verificar las hipótesis de trabajo, se planifica una exhaustiva comparación con otros trabajos relacionados [30].

Referencias

1. Jackson, M.: *Software Requirements & Specifications: A lexicon of practice, principles and prejudices*, Addison Wesley, ACM Press (1995)
2. Cristian, F.: *Exception Handling and Tolerance of Software Faults*, *Software Fault Tolerance*, M.R.Lyu (ed.), Wiley, Chichester (1995) 81-107
3. Perry, D.E.; Romanovsky, A., Tripathi, A., *Current Trends in Exception Handling*, *IEEE Transactions on Software Engineering*, Vol.26, N° 9 (2000) 817-819
4. Ghezzi, C. ; Jazayeri, M., Mandrioli, D., *Fundamentals of Software Engineering*, Prentice Hall, Englewood Cliffs, New Jersey (1991)
5. van Lamsweerde, A., Letier, E.: *Handling Obstacles in Goal-Oriented Requirements Engineering*, *IEEE Transactions on Software Engineering*, Vol.26, N° 10 (2000) 978-1005
6. Potts, C. : *Using Schematic Scenarios to Understand User Needs*, *Proceedings of DIS'95 - Symposium on Designing Interactive Systems: Processes, Practices and Techniques*, ACM Press, University of Michigan (1995)
7. Booch, G.: *Object Oriented Design with Applications*, The Benjamin Cumming Publishing Company, Inc., Redwood City, (1991)
8. Jacobson, I., Christerson, M., Jonsson, P., Overgaard, G.: *Object-Oriented Software Engineering - A Use Case Driven Approach*, Reading, MA: Addison Wesley, New York: ACM Press (1992)
9. Zorman, L.: *Requirements Envisaging by Utilizing Scenarios (Rebus)*, Ph.D. Dissertation, University of Southern California (1995)
10. Rolland, C., Ben Achour, C.: *Guiding the construction of textual use case specifications*, *Data & Knowledge Engineering* 25 (1998) 125-160
11. Weidenhaupt, K., Pohl, K., Jarke, M., Haumer, P.: *Scenarios in System Development: Current Practice*, *IEEE Software* (1998)
12. Leite, J.C.S.P., Franco, A.P.M.: *A Strategy for Conceptual Model Acquisition*, *IEEE International Symposium on Requirements Engineering*, IEEE Computer Society Press (1993)
13. Leite, J.C.S.P., Rossi, G., Balaguer, F., Maiorana, V., Kaplan, G., Hadad, G., Oliveros, A.: *Enhancing a Requirements Baseline with Scenarios*, *Requirements Engineering Journal*, Vol.2, N° 4 (1997) 184-198

14. Leite, J.C.S.P., Hadad, G.D.S., Doorn, J.H., Kaplan, G.N.: A Scenario Construction Process, *Requirements Engineering Journal*, Vol.5, N° 1 (2000) 38-61
15. Hadad, G.; Kaplan, G.; Oliveros, A.; Leite, J.C.S.P.: Construcción del Léxico Extendido del Lenguaje y derivación de Escenarios para la elicitación de requerimientos (1997)
16. Ridao, M., Doorn, J., Leite, J.C.S.P.: Domain Independent Regularities in Scenarios, *Proceedings of Fifth IEEE International Symposium on Requirements Engineering*, Toronto, Canada (2001) 120-127
17. Ridao, M., Doorn, J., Leite, J.C.S.P.: Incorporación de patrones al proceso de construcción de escenarios, *Proceedings de WER 2001*, Buenos Aires, Argentina (2001) 107-123
18. Leite, J.C.S.P., Oliveros, A., Rossi, G., Balaguer, F., Hadad, G., Kaplan, G., Maiorana, V.: Léxico extendido del lenguaje y escenarios del sistema nacional para la obtención de pasaportes, *Technical Report #7*, Departamento de Investigación, Universidad de Belgrano, Buenos Aires (1996)
19. Leite, J.C.S.P.: Ingeniería de Requisitos. Notas de Cátedra (1997)
20. Hadad, G., Kaplan, G., Leite, J.C.S.P.: Léxico extendido del lenguaje y escenarios del meeting scheduler. *Technical Report # 13*, Dto. Investigación, U. Belgrano, Bs. As. (1998)
21. van Lamsweerde, K., Darimont, R., Massonet, Ph.: The Meeting Scheduler System – Preliminary Definition. *Internal Report*, Université Catholique de Louvain (1993)
22. Mauco, V., Ridao M., del Fresno, M., Rivero, L., Doorn, J.: Ingeniería de Requisitos, Proyecto: Sistema de Planes de Ahorro, *Technical Report*, ISISTAN, UNCPBA, Tandil, Argentine (1997)
23. Rivero, L., Doorn, J., del Fresno, M., Mauco, V., Ridao, M., Leonardi, M.C.: Una Estrategia de Análisis Orientada a Objetos basada en Escenarios: Aplicación en un Caso Real, *Proceedings of WER'98, Workshop en Engenharia do Requisitos*, Maringá, Brazil, (1998) 79-90
24. Breitman, K.K., Leite, J.C.S.P.: A Framework for Scenario Evolution, *Proceedings of the IEEE Int. Conf. on Requirements Engineering*, IEEE Computer Society Press (1998) 214-221
25. Garcia, O., Gentile, C.: Diseño de una herramienta para construcción de LEL y Escenarios, *Graduation dissertation*, Universidad Nacional del Centro de la Provincia de Buenos Aires, Argentina (1999)
26. Sánchez, M.: Léxico Extendido del Lenguaje y Escenarios para el Almacén de una Fábrica, *Graduation dissertation*, Universidad Nacional del Centro de la Provincia de Buenos Aires, Argentina (2002)
27. Ridao, M., Doorn, J., Leite, J.C.S.P.: Aspectos Recurrentes en la Construcción de Escenarios, *Memorias de IDEAS'00 – Jornadas Iberoamericanas de Ingeniería de Requisitos y Ambientes de Software*, Cancún, México (2000) 349-360
28. Ridao, M., Doorn, J., Leite, J.C.S.P.: Uso de Patrones en la Construcción de Escenarios, *Anais de WER 2000*, Río de Janeiro, Brazil, (2000) 140-157
29. Ridao, M: Uso de Patrones en el Proceso de Construcción de Escenarios, *Tesis de Maestría en Ingeniería de Software*, Univ. Nacional de La Plata, (2001)
30. Sindre G., Opdahl A.: Templates for Misuse Case Description. *Seventh International Workshop on Requirements Engineering: Foundation for Software Quality*, Interlaken, Switzerland (2001)