

Integración de Escenarios con el Léxico Extendido del Lenguaje en la elicitación de requerimientos* : Aplicación a un caso real.

Graciela Hadad, Gladys Kaplan, Alejandro Oliveros
Departamento de Investigación
Universidad de Belgrano
Zabala y Villanueva Piso 12, (1426) Buenos Aires
Argentina
e-mail:(ghadad,gkaplan)@ub.edu.ar

Julio Cesar Sampaio do Prado Leite
Departamento de Informática
Pontificia Universidade Católica do Rio de Janeiro
R. Marquês de S. Vicente 225 Rio de Janeiro 22453
Brasil
e-mail:julio@inf.puc-rio.br

Mayo 1996

1 INTRODUCCION

La ingeniería de requerimientos es un área de estudio que apunta a mejorar la forma en que comprendemos y definimos sistemas de software complejos. Aunque el problema de cómo establecer sistemas no es nuevo, sólo recientemente la comunidad científica de computación, principalmente la comunidad de ingeniería de software, comenzó a prestar especial atención a la tarea de definición de sistemas. La primera conferencia internacional sobre este tema fue en 1993 (RE 93) y en 1995 se realizó la primera reunión del grupo de trabajo IFIP 2.9 (Ingeniería de Requerimientos de Software).

La ingeniería de software ha estado prestando demasiada atención a la modelización de productos, pero la ingeniería de requerimientos comenzó examinando los aspectos de la generación de requerimientos y su intrínseca dependencia de los aspectos sociales que rodean la construcción y que rodearán a la operación del software. Nuestro trabajo se centra en el uso de escenarios a lo largo del proceso de desarrollo de software, investigando cómo evolucionan los escenarios hasta las estrategias de testeo, lo cual es el tema principal del proyecto conjunto entre la PUC-Rio y la Universidad de Belgrano. Este artículo es nuestro primer resultado en la tarea de construcción de escenarios.

Los escenarios [Carrol 95] [Rubin 92] [Potts 94] [Jacobson 92] son bien reconocidos como una importante estrategia para la comprensión de la interfase entre el medio ambiente y el sistema, así como también, un medio para elicitación y especificación del comportamiento del software. Nosotros tenemos una comprensión más amplia de escenarios. Para nosotros, un escenario es una descripción, que evoluciona, de situaciones en el medio ambiente. Un escenario comienza en el macrosistema, para el cual el software es un componente, y continúa evolucionando a medida que el "artefacto" de software se va construyendo. Es importante destacar que el proceso de construcción de software también modificará el macrosistema, como bien observó Lehman [Lehman 80] en su trabajo sobre E-sistemas. Como tal, es de mayor importancia investigar el proceso de construcción de escenarios iniciales. Este artículo describe la estrategia utilizada en nuestro proyecto, así como justifica las decisiones que fueron tomadas para

* El término Requerimientos se ha utilizado como sinónimo de Requisitos.

producir los escenarios. Es importante notar que el foco principal de este artículo, y su mayor contribución, es la descripción del proceso de construcción de escenarios del macrosistema para un caso real, el “Sistema Nacional para la Obtención de Pasaportes”¹.

En la fase de establecimiento de los requerimientos, el ingeniero de software se enfrenta con diferentes dificultades, dos de ellas son de particular interés para nosotros:

- *Establecer una buena comunicación con el usuario.*
Con el propósito de:
 - ⇒ Comprender sus necesidades [Leite 89].
 - ⇒ Detectar errores y omisiones. Esto evitará una “catarata de errores” [Mizuno 83] y su propagación durante el resto del proceso de desarrollo, agravándose en cada etapa siguiente.
 - ⇒ Tener un vocabulario propio de forma tal de “echar anclas” en dichos términos..

- *Garantizar el seguimiento (“traceability”) de los requerimientos durante todo el ciclo de vida del software* [Leite 95].
Con el propósito de:
 - ⇒ Mantener los orígenes de los requerimientos [Gotel 94].
 - ⇒ Generar casos de prueba a partir de los requerimientos.
 - ⇒ Mantener actualizado los propios requerimientos [Lehman 80].

Estas dificultades han conducido a la búsqueda de soluciones en el proceso de producción de los requerimientos. Este proceso involucra la elicitación, la modelización y el análisis² de los requerimientos [Leite 91].

En nuestro trabajo nos hemos centrado en la elicitación de los requerimientos, para ello estudiamos dos herramientas denominadas **LEL** [Leite 89] y **escenarios** [Carrol 95] [Rubin 92] [Potts 94] [Jacobson 92], y las aplicamos a un caso de estudio real: Sistema Nacional para la Obtención de Pasaportes. Ambas herramientas utilizan el lenguaje natural lo que permite una fácil validación con el usuario. El LEL (Léxico Extendido del Lenguaje) [Leite 89] tiene como propósito exclusivo conocer la semántica del vocabulario de la aplicación, dejando para una etapa posterior la comprensión del problema [Franco 92]. Utilizamos los escenarios para comprender el problema. En cada escenario se describe un momento específico de la aplicación, lo cual permite refinar el problema y analizarlo parcialmente.

Este informe está organizado de la siguiente manera: primero describimos las características básicas del Léxico Extendido del Lenguaje, luego comentamos sobre escenarios y la representación utilizada en este trabajo, seguimos con un detalle del caso de estudio y las observaciones derivadas de él, y para finalizar dejamos explícitas nuestra contribución y los futuros temas de estudio.

2 LEXICO EXTENDIDO DEL LENGUAJE

La comunicación es un elemento complejo pero fundamental en cualquier relación humana. Permite que dos o más personas, manteniendo sus propios intereses, puedan entenderse. Durante todo el proceso de desarrollo del software, la comunicación es un elemento

¹ El caso utilizado se basa en el sistema de expedición de pasaportes de la Policía Federal Argentina, vigente hasta el 1º de julio de 1996.

² El análisis comprende la validación y la verificación de los hechos modelizados.

indispensable que todos los actores³ tienen que manejar a diario [Gause 89]. Cuando el ingeniero de software se comunica con el usuario utiliza su propio vocabulario. Paralelamente, el usuario utiliza el lenguaje habitual de su medio ambiente. Una comunicación basada en diferentes léxicos tiene una mayor probabilidad de fracasar.

Con el fin de establecer, a partir de la fase de elicitación, una comunicación clara y eficaz con el usuario, el ingeniero de software debe ocuparse de homogeneizar el vocabulario. Utilizar un mismo vocabulario se potencia si todos los actores utilizan el *lenguaje propio del usuario* [Leite 89].

El LEL (Léxico Extendido del Lenguaje) es una herramienta que permite al ingeniero de software conocer el lenguaje del usuario y enfocar el traceability de los requerimientos. Como se refleja en el Cuadro 1, de cada uno de estos objetivos se derivan consecuencias importantes.

Objetivo	Consecuencia
CONOCER EL VOCABULARIO DEL USUARIO	<ul style="list-style-type: none"> • Asegurar la comunicación • Facilitar la validación de los requerimientos con el usuario • Mantener el mismo vocabulario durante todo el proceso de desarrollo
CONTAR CON UN INSTRUMENTO SIMPLE DE TRACEABILITY	<ul style="list-style-type: none"> • Documentar consistentemente • Capacitar a nuevos miembros del equipo en la terminología empleada • Generar versiones del LEL a medida que evoluciona el proceso de desarrollo

Cuadro 1: Objetivo-Consecuencia del LEL

Leite [Leite 89] propone construir el LEL como primer paso en la fase de elicitación de los requerimientos: despreocuparse durante este procedimiento de entender el problema y abocarse sólo a conocer el vocabulario que utiliza el usuario en su mundo real. Una vez familiarizados con ese léxico, el camino para comunicarse con el usuario y comprender el problema tendrá un obstáculo menos: el vocabulario.

³ Los actores son todas las personas involucradas en el proceso de desarrollo: los usuarios y los ingenieros de software.

2.1 CARACTERISTICAS DEL LEL

El LEL está compuesto por un conjunto de símbolos que identifican el lenguaje de la aplicación. Los *símbolos* son, en general, las palabras o frases utilizadas por el usuario y que repite con más frecuencia. También se incluyen aquellas palabras o frases que son relevantes para el dominio del problema más allá de su frecuencia de repetición. Los símbolos se adquieren, por ejemplo, de entrevistas, observaciones, lectura de documentos [Leite 90]. Se genera una lista con todos los símbolos reconocidos. Durante el proceso de recolección, el ingeniero de software procura entender el significado de cada símbolo. La semántica de cada símbolo se representa con una o más nociones y uno o más impactos. La *noción* indica qué es el símbolo y el *impacto* cómo repercute en el sistema. Por lo tanto, cada símbolo tiene un “nombre” que lo identifica, una “noción” y un “impacto” que lo describen [Leite 93]. El conjunto de símbolos forman una red, que permite representar al LEL en un hipertexto [Leite 90] y navegar en él para conocer todo el vocabulario del dominio.

En la descripción de las nociones e impactos existen dos reglas básicas [Leite 93] que se deben cumplir simultáneamente: una tiende a maximizar el uso de símbolos en el significado de otros símbolos. Esto se denomina “principio de circularidad”. La otra requiere que se minimice el uso de símbolos externos al lenguaje de la aplicación. Este principio se denomina “principio del vocabulario mínimo”.

3 ESCENARIOS

Los escenarios son descripciones parciales del funcionamiento del sistema, focalizan un momento específico de la aplicación. Varios autores han estudiado esta herramienta como por ejemplo [Potts 94] [Booch 91] [Jacobson 92] [Carrol 95] [Zorman 95]. Los escenarios no son formales, se pueden representar de diferentes maneras: narrativa textual, storyboards, video mock-ups, prototipos escritos o situaciones físicas. El nivel de detalle con el que se describen depende de dos factores:

- el grado de importancia que el usuario le otorgue a los hechos específicos del problema, y
- la fase en la que se encuentra el proceso de desarrollo.

Si bien cada escenario es una descripción parcial del comportamiento de la aplicación, ningún escenario es totalmente independiente del resto de los escenarios. Cada escenario tiene una relación semántica con otros escenarios [Booch 94].

Los escenarios deben evolucionar durante el ciclo de vida del software y esta evolución debe ser registrada. Esto permite realizar el seguimiento de los requerimientos hasta sus orígenes.

Los escenarios cumplen distintos objetivos dependiendo en qué fase se describen durante el ciclo de vida del sistema.

Mencionaremos en el Cuadro 2 los objetivos que cumplen los escenarios en la fase de producción de requerimientos y las consecuencias que se derivan del uso de escenarios:

Objetivo	Consecuencia
----------	--------------

<p>CAPTURAR REQUERIMIENTOS</p>	<ul style="list-style-type: none"> • Evitar abstracciones orientadas a una solución • Brindar una visión más amplia del comportamiento del macro sistema.
<p>PROVEER UN MEDIO DE COMUNICACION</p>	<p>(Derivan de la utilización del lenguaje de la aplicación)</p> <ul style="list-style-type: none"> • Asegurar la comunicación entre el usuario y el ingeniero de software • Garantizar la comunicación entre los miembros del equipo de desarrollo • Facilitar la validación de los requerimientos con el usuario • <i>Comprometer</i> al usuario en el desarrollo • Validar el LEL
<p>CONTAR CON UN INSTRUMENTO DE TRACEABILITY</p>	<ul style="list-style-type: none"> • Documentar los requerimientos • Capacitar a nuevos miembros del equipo para comprender la aplicación • Promover la evolución de los escenarios a medida que avanza el proceso de desarrollo • Generar casos de prueba

Cuadro 2: Objetivo-Consecuencia de Escenarios

4 CASO DE ESTUDIO

Utilizamos como caso de estudio el Sistema Nacional para la Obtención de Pasaportes [Hada 96]. Su enfoque es estrictamente teórico y sólo lo estudiamos para la fase de producción de requerimientos.

El pasaporte es un documento que permite a los habitantes del país salir del mismo, siendo requisito indispensable para viajar a países no limítrofes.

El sistema es único para todo el país y se centraliza en el Departamento de la Policía Federal, con sede en la Capital. Desde el interior del país, se puede solicitar un pasaporte pero la gestión y emisión del mismo queda centralizada en la Capital. Esto implica que es un sistema que maneja un gran volumen de información, requiere un gran movimiento de papeles, verificación de documentos y controles internos, y presenta una gran variedad de alternativas posibles. Es casi enteramente manual y soporta poca información computarizada.

Para realizar el estudio dispusimos de :

- entrevistas con personal del Departamento de Policía;
- entrevistas con personas que realizaron el trámite con anterioridad;
- los formularios utilizados para realizar el trámite;
- un estudio de campo (observación y rol de actor).

Nuestra estrategia de trabajo (Fig.1) consistió en construir el LEL partiendo del Universo de Discurso⁴, y luego en base éste y al LEL ya refinado, generamos los escenarios. En ambos casos, validamos el producto obtenido contra el Universo de Discurso, y los escenarios nos sirvieron para verificar el léxico. Una vez finalizada esta tarea, obtuvimos conocimiento sobre todos los hechos relevantes de la aplicación, cumpliendo con la fase de elicitación de los requerimientos.

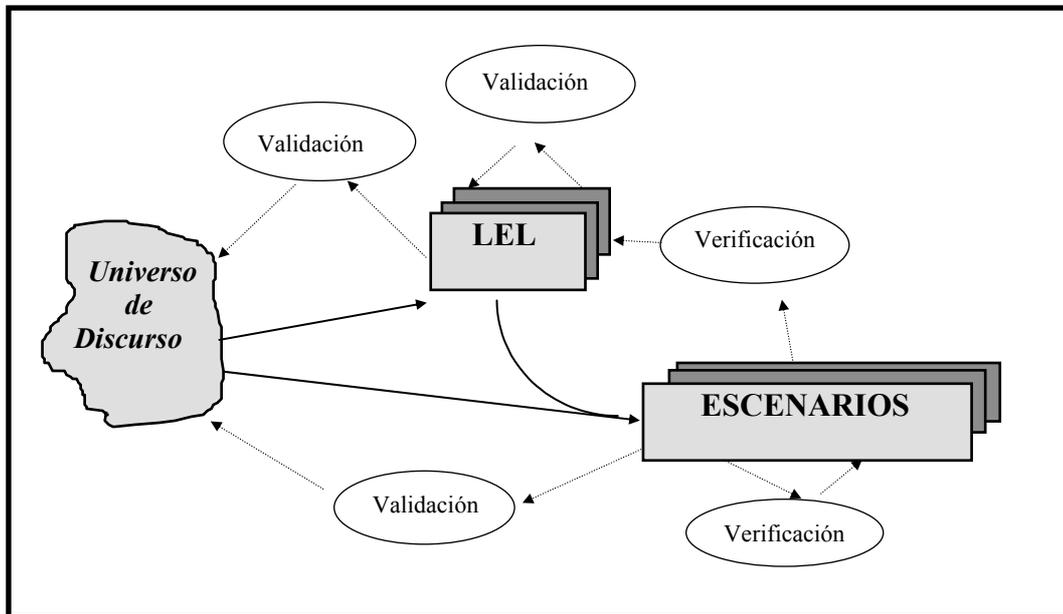


Fig.1 - Estrategia utilizada en la elicitación de los requerimientos.

Debemos distinguir entre la “Validación del LEL” y la “Verificación de Escenarios” (Fig.1). En el proceso de construcción del LEL, distintos actores modelaron los mismos símbolos. Es decir, dispusimos de distintos *puntos de vista del LEL*. Por lo tanto, para obtener un LEL refinado, realizamos un proceso de validación de todos los símbolos en base a los distintos puntos de vista.

En el caso de la construcción de escenarios, distribuimos entre los actores la descripción de los escenarios. Luego, realizamos una verificación de los escenarios a cargo de los actores no involucrados en su descripción. Es decir, en este caso, no hubo distintos puntos de vistas de los escenarios.

A continuación detallaremos cómo generamos el LEL para el sistema de pasaportes y luego describiremos el proceso de construcción de los escenarios.

4.1 APLICACION DEL LEL

Estudiamos el problema desde dos puntos de vista:

- a) quien gestiona el trámite para obtener un pasaporte, y
- b) quien administra el sistema (nuestro **usuario**).

⁴ Universo de Discurso es todo el contexto en el que el software se desarrolla. El universo de discurso incluye todas las fuentes de información y toda las personas relacionadas con el software. Otros términos utilizados son: Mundo Real o Dominio del Problema.

Inicialmente, llevamos a cabo un estudio de campo, que consistió en realizar un trámite de obtención de pasaporte. Luego, realizamos entrevistas con el usuario orientadas a conocer el problema. Simultáneamente, entrevistamos a personas que habían gestionado algún trámite.

En base a la información recolectada y a los formularios disponibles, asentamos por escrito el Universo de Discurso [Leite 89]. De este texto, obtuvimos los símbolos tentativos, correspondientes a palabras o frases que aparecían repetidas con frecuencia, aquellas relevantes al contexto, aquellas cuyo significado se desconocía o cuyo significado era dudoso.

Partiendo de esta lista inicial, realizamos nuevas entrevistas con el usuario para conocer y/o clarificar el significado de algunos símbolos; surgieron nuevos símbolos y eliminamos otros. Este proceso se repitió hasta concluir en una lista casi definitiva. Con ella, tres integrantes del grupo comenzaron a definir las mismas entradas⁵ del LEL, en forma independiente, con el propósito de verificar posteriormente lo realizado. Cada integrante obtuvo un LEL distinto, es decir contamos con tres puntos de vista del léxico. Esto nos llevó a confirmar y/o eliminar algunas entradas, quedando la lista final con 39 símbolos.

En reiteradas oportunidades, analizamos la forma de volcar los conocimientos adquiridos del problema en la estructura del LEL. Parte de esta dificultad, fue consecuencia de haber realizado inicialmente entrevistas orientadas a conocer el problema y no a conocer exclusivamente el vocabulario del problema.

Debimos adaptar las reglas propias del LEL [Leite 89] a la aplicación en estudio. No obstante esto y los criterios propios establecidos, concluimos que existe un factor muy importante a tener en cuenta, que es la subjetividad con la que el léxico puede ser descripto.

Por último, estas tres versiones del léxico fueron unificadas por un cuarto integrante del grupo. Esta última versión la verificamos entre todos los participantes. Con este LEL, realizamos el proceso de validación con el usuario (ver detalle del proceso en la Fig.8).

Durante el desarrollo de cada versión individual, nos vimos en la necesidad de establecer nuestros propios criterios para obtener un LEL consistente. Algunos de estos criterios los establecimos luego como regla para concluir el LEL en esta fase.

Mencionaremos algunos **problemas** presentados en la construcción del LEL y los **criterios adoptados** en cada caso:

Problema:

- Determinar las entradas al léxico: cuáles debían ir, cuáles no eran necesarias, cuáles no correspondían al dominio del problema.

Criterio Adoptado:

- Incorporar solo frases o palabras correspondientes al dominio de la aplicación. Excluimos de éstas:
 - ⇒ entradas obvias, ya que son de conocimiento general de todas las personas (usuarios, ingenieros de sistemas y personas externas al Universo de Discurso).
 - ⇒ entradas excesivamente específicas, por considerarlas que no forman parte central del problema en cuestión.

⁵ El término Entrada se utiliza como sinónimo de Símbolo.

⇒ entradas con un significado amplio (ambiguas), porque su definición puede quedar muy genérica o muy detallada.

Problema:

- Describir símbolos afines con la misma estructura. Este problema fue crítico debido a la participación de varias personas.

Criterio Adoptado:

- Clasificar las entradas y establecer para cada tipo de entrada la forma de definirla, tanto la noción como el impacto.
-

Problema:

- Repetir o no impactos comunes en distintos símbolos o hacer referencia a un nuevo símbolo que contenga dichos impactos aunque éste no sea parte del dominio del problema.

Criterio Adoptado:

- Repetir los impactos comunes en todos los símbolos.
-

Problema:

- Determinar el nivel de detalle al definir las entradas.

Criterio Adoptado:

- Volcar en cada definición de un símbolo todo el conocimiento que sobre él se posee.
-

Problema:

- Dificultad en la detección de sinónimos.

Criterio Adoptado:

- Realizar varias entrevistas con diversos usuarios ayuda a obtener sinónimos o clarificar sinónimos confusos.
 - Identificar sinónimos comparando símbolos con iguales nociones e iguales impactos.
-

Problema:

- Al buscar símbolos, determinar la utilización del “verbo” o del “sustantivo” como nombre del símbolo. Ejemplo: *Controlar las Huellas Digitales* o *Control de Huellas Digitales*.

Criterio Adoptado:

- Utilizar el “sustantivo” como nombre del símbolo.
-

Problema:

- Determinar el nombre de la entrada en caso de frases largas.

Criterio Adoptado:

- Incluir la frase completa como nombre de la entrada, ya que se debe respetar el vocabulario propio del usuario.
-

Problema:

- En algunos casos, dificultad para establecer si una frase corresponde a una noción o a un impacto en un determinado símbolo.

Criterio Adoptado:

- Identificar el símbolo dentro de la clasificación de entradas. De acuerdo al tipo de símbolo, reconocer la frase como una noción o como un impacto.
-

Cada criterio recién mencionado se debe aplicar conjuntamente con los demás.

Cabe destacar, que algunos de estos problemas se presentaron como consecuencia de la participación de varias personas en la construcción del LEL. El uso de distintos puntos de vista enriquece al LEL, pero dificulta la obtención de un LEL consistente.

Detallamos seguidamente el criterio correspondiente a la **clasificación de entradas**, dado que este criterio tuvo primordial importancia para obtener la versión final del LEL en esta fase. El uso de distintos puntos de vista requirió necesariamente de esta clasificación.

Primero agrupamos los símbolos de la siguiente manera:

- sectores
- acciones
- personas
- formularios
- índices
- depósitos

Luego, determinamos el contenido y la narrativa utilizada para cada caso. Describimos a continuación, para cada tipo de entrada sus características y un ejemplo.

Para los **sectores**, en la noción identificamos *dónde* se desarrolla y las *acciones* principales que cumple, y en el impacto los *efectos sobre* las personas, los formularios, los depósitos y la generación o uso de índices.

DIVISION DACTILOSCOPIA

Noción

- sector de la Policía Federal
- se encarga de controlar las huellas digitales
- contiene el archivo de huellas digitales

Impacto

- firman y sellan el reverso del formulario de solicitud en el casillero ‘Dactiloscopia’.

Fig.2 - Descripción de un símbolo correspondiente a un Sector.

Para las **acciones**, en la noción identificamos el sector *donde* se lleva a cabo y el *objetivo* de la acción; en el impacto describimos los *efectos sobre* las personas, los formularios, los depósitos y la generación o uso de índices (contenido similar a Sectores).

CONTROL DE HUELLAS DIGITALES

Noción

- acción que ejecuta la División Dactiloscopia para comprobar que el solicitante es quien dice ser.

Impacto

- se verifican las huellas digitales del formulario contra las del Archivo de Huellas Digitales en el caso de pasaporte original.
- se almacena la ficha dactiloscópica en el Archivo de Dactiloscopia, en el caso de pasaporte original.
- se verifica la impresión del pulgar derecho del formulario contra la ficha dactiloscópica, en caso de reválida de pasaporte.

Fig.3 - Descripción de un símbolo correspondiente a una Acción.

En el caso de **índices**, en la noción identificamos cuál es su *objetivo*, *quién* lo genera y sus *características*; en el impacto describimos *dónde* se lo utiliza, *en qué* formularios y *para qué* acciones.

NUMERO DE IDENTIFICACION / NUMERO DE PASAPORTE

Noción:

- número único que identifica a una persona en la Policía Federal.
- coincide con el número de cédula de identidad, el del pasaporte y el del legajo.

Impacto:

- este número se perfora en el pasaporte.

Fig.4 - Descripción de un símbolo correspondiente a un Índice.

En el caso de las **personas**, en la noción identificamos las *características y condiciones* que debe cumplir; en el impacto describimos todas las *acciones* que pueden realizar.

SOLICITANTE / INTERESADO

Noción

- persona que inicia un trámite relativo al pasaporte.
- debe ser ciudadano argentino, hijo de funcionario argentino nacido en el exterior, extranjero casado con ciudadano argentino, habitante argentino carente de nacionalidad o habitante argentino extranjero...

Impacto

- completa un formulario solicitud.
- presenta la documentación.
- paga el trámite.
- le sacan fotografías.
- le toman la impresión de huellas digitales.
- ...

Fig.5 - Descripción de un símbolo correspondiente a una Persona.

Para los **depósitos**, en la noción identificamos su *contenido y ubicación*; en el impacto describimos *en qué acciones* se lo utiliza.

ARCHIVO DE HUELLAS DIGITALES / ARCHIVO DE DACTILOSCOPIA

Noción:

- conjunto de fichas dactiloscópicas que se encuentran en la División Dactiloscopia.
- se archivan las fichas dactiloscópicas.

Impacto:

- es utilizado por el control de las huellas digitales.

Fig.6 - Descripción de un símbolo correspondiente a un Depósito.

Para los **formularios**, en la noción identificamos la información que *contiene* y su *finalidad*; en el impacto describimos todas las *acciones que sobre él se efectúan*.

FORMULARIO DE SOLICITUD / FORMULARIO

Noción:

- es un papel preimpreso obligatorio para realizar el trámite de pasaporte
- se registran todos los datos personales del solicitante
- se escriben las verificaciones que realiza la Policía Federal

Impacto:

- lo llena el solicitante.
- lo sellan en la Cabina de Fotografía al abonar la fotografía.
- lo sellan en la Caja al abonar la fotografía.
- lo llenan, firman y sellan en el reverso en el casillero “Índice General” en la División Índice General.
- ...

Fig.7 - Descripción de un símbolo correspondiente a un Formulario.

4.2 VALIDACION DEL LEL

Utilizamos el concepto de *escenario* (Fig.9) para describir el proceso de validación del documento obtenido.

Nombre: Validación del LEL

Objetivo: Validar el LEL con el usuario. Obtener nuevos símbolos.

Contexto: El ingeniero de software se reúne con usuarios de diferentes ámbitos y niveles. Con cada uno, controla los símbolos del LEL correspondientes a su ámbito.

Recursos: LEL, formularios, manuales, documentos

Actores: Usuario, Ingeniero de software

Set de Episodios:

- El ingeniero de software separa del LEL aquellos símbolos correspondientes al ámbito en que cada usuario se desenvuelve.
- El ingeniero de software realiza entrevistas no estructuradas con cada usuario, entregando una copia del LEL que contiene sólo los símbolos a validar con él.
- El ingeniero de software examina con el usuario cada noción e impacto.
- El usuario confirma, corrige, explica ítems confusos o amplía descripciones incompletas.
- El ingeniero de software detecta posibles sinónimos y los ratifica con el usuario.
- El usuario corrige, confirma y/o agrega otros sinónimos.
- El ingeniero de software toma nota de posibles ‘nuevos símbolos’.
- El ingeniero de software pregunta qué significa cada ‘nuevo símbolo’ y cómo impacta en el sistema (noción e impacto).
- El usuario describe los ‘nuevos símbolos’.
- El ingeniero de software anota todo lo referido a los ‘nuevos símbolos’ para un posterior análisis de los mismos.

Fig.8 - Validación realizada en el caso de estudio.

Otras validaciones realizadas consistieron en entrevistas estructuradas, donde el usuario no contaba con una copia del LEL. Estas reuniones estuvieron basadas en preguntas programadas orientadas a chequear las descripciones existentes. Con esta información rectificamos o ratificamos los símbolos involucrados. Este proceso permitió además, detectar nuevos símbolos y sinónimos.

El proceso de validación del LEL depende del tipo de sistema y del usuario.

4.3 APLICACION DE ESCENARIOS

Con el LEL terminado, comenzamos a elaborar los escenarios. Para ello, generamos una lista de situaciones, posibles candidatas a escenarios.

Representamos los escenarios en la forma de un documento escrito en lenguaje natural, utilizando descripciones textuales como las empleadas por Jacobson en los Use-Cases [Jacobson 92]. A continuación, describimos el esquema de escenarios aplicado, con una breve explicación de cada ítem:

Nombre: *identifica al escenario.*

Objetivo: *establece la finalidad del escenario.*

Contexto: *describe las acciones previas necesarias para iniciar el escenario, las precondiciones, la ubicación física y temporal.*

Recursos: *identifica los objetos pasivos con los cuales los actores trabajan.*

Actores: *detalla las entidades que se involucran activamente en el escenario.*

Set de Episodios: *cada episodio representa una acción realizada por un actor, donde participan otros actores y se utilizan recursos. Los episodios se ejecutan secuencialmente. Un episodio también puede referenciar a un escenario.*

*Se incluyen las **restricciones** del escenario o de cada episodio según corresponda.*

Casos alternativos: *menciona los casos de excepción, que pueden corresponder a otros escenarios.*

Dudas: *describe los puntos pendientes a clarificar con el usuario.*

Fig. 9 - Esquema utilizado para la descripción de escenarios

Dado que nuestras primeras entrevistas fueron orientadas a conocer el problema, y no sólo a conocer el vocabulario, contamos desde el inicio con la información suficiente para comenzar a describir los escenarios. Nuestro primer interrogante fue: *¿Qué escenarios comenzar a describir?*

Esto nos llevó a clasificar los escenarios de la siguiente manera:

a) escenarios particulares, son aquellos que describen un momento específico de la aplicación (Fig.10). Usualmente, los episodios de estos escenarios corresponden a simples acciones.

b) escenarios generales, son aquellos que representan las funciones fundamentales de la aplicación (Fig.11). Usualmente, los episodios de estos escenarios referencian a otros escenarios. Esto permite establecer las relaciones entre los escenarios particulares.

Se podrían describir más niveles de jerarquía. Es decir, en un escenario particular también podría existir un episodio que correspondiese a un escenario en un nivel de detalle más bajo.

En nuestro caso de estudio, sólo se nos presentó dos niveles de jerarquía de escenarios.

Decidimos definir inicialmente los escenarios particulares, debido a tres factores:

- la facilidad para comprender situaciones acotadas,
- el conocimiento detallado adquirido durante el proceso de generación del LEL, y
- la facilidad para validar estos escenarios con el usuario.

Para cada situación, definimos el contexto y los episodios. Con esta información, de la lista de situaciones se confirmaron algunas en uno o más escenarios, otras situaciones se eliminaron. Este proceso permitió generar la lista de escenarios con sus respectivos nombres. Definimos el objetivo de cada escenario y de los episodios, obtuvimos los actores y los recursos. Luego, identificamos las restricciones y posteriormente, mencionamos los casos alternativos. Algunos de estos casos surgieron al describir los episodios. Las dudas las anotamos a medida que surgían en la descripción de los episodios.

Las mayoría de las restricciones se obtuvieron del contexto, debido a que en el contexto identificamos las precondiciones. Las restantes restricciones surgieron como necesidades para ejecutar algún episodio.

Algunos casos alternativos se determinaron como negación de un episodio frente al uso de un recurso. Ejemplo de un caso alternativo: *Máquina timbradora falla*. Otros casos alternativos se detectaron dentro de la situación, pero no correspondían a la secuencia normal de los episodios. Ejemplo: *Envío a incineración de pasaportes no retirados*.

Repetiendo el procedimiento empleado en la construcción del LEL, tres integrantes redactaron los mismos escenarios (aproximadamente 5) con el propósito de validarlos. Las diferencias entre las versiones se debieron principalmente al conocimiento que cada participante tenía del tema y al punto de vista aplicado. Unificamos las versiones, fijamos criterios y repartimos los restantes escenarios para su descripción. Luego, todos los escenarios fueron verificados por cada participante. Con esta versión, realizamos el proceso de validación con el usuario.

La lista inicial de escenarios sufrió escasas alteraciones. Esto se debió, principalmente, a que el LEL nos permitió contar con información bien estructurada, la cual simplificó la posterior tarea de obtener más conocimiento sobre el problema. La lista final contó con 24 escenarios, correspondiendo a 20 escenarios particulares y 4 escenarios generales.

Durante el proceso de construcción de los escenarios, surgieron dificultades para representar las diferentes situaciones. Presentamos a continuación, los problemas y los respectivos criterios adoptados:

Problema:

- ¿ Qué tratamiento dar a los escenarios que representan distintas situaciones, pero su descripción difiere solo en uno o dos episodios ?

Criterio Adoptado:

- Unificamos ambos escenarios si esto no dificulta la comprensión del escenario. En caso contrario, presentamos los escenarios separadamente.

Problema:

- ¿ Cómo representar episodios sujetos a ciertas condiciones ?

Criterio Adoptado:

- Incluimos la condición para indicar si se realiza un episodio u otro, siempre que la condición corresponda a una sola acción. Si la condición involucra varias acciones, redactamos escenarios separados, para favorecer la comprensión del escenario. Estos escenarios no presentan la condición, ésta se incluye en el contexto de cada escenario.
 - Al describir los recursos y los actores del escenario, indicamos a qué caso corresponden.
-

Problema:

- ¿Cómo describir los actores y los recursos en los escenarios generales ?

Criterio Adoptado:

- Indicamos sólo los recursos y los actores más importantes del escenario general, para evitar un grado de detalle que dificulte la comprensión.
-

Problema:

- ¿Desde que punto de vista describir los escenarios ?

Criterio Adoptado:

- Los escenarios los redactamos desde el punto de vista de la aplicación. Por ejemplo, el trámite para obtener el pasaporte tiene un costo. Podemos expresarlo en las siguientes formas:
 - a) *Abonar el trámite*, desde el punto de vista externo a la aplicación;
 - b) *Cobrar el trámite*, desde el punto de vista interno a la aplicación.Adoptamos la segunda alternativa, porque la validación se hace con la persona que opera la aplicación y no con quien hace uso de ella.
-

Problema:

- ¿Cómo mantener el vocabulario del dominio de la aplicación en los escenarios ?

Criterio Adoptado:

- Utilizamos el vocabulario del LEL en la descripción de los escenarios, subrayando cada símbolo perteneciente al LEL. De esta forma, establecemos una fuerte relación entre el LEL y los escenarios. Esto permite:
 - ⇒ mantener el lenguaje del usuario,
 - ⇒ concentrarse en las acciones del escenario, sin desviar la atención en el significado de los términos, y
 - ⇒ profundizar los conocimientos propios del escenario.
-

Problema:

- ¿Cómo identificar un episodio que es un escenario ?

Criterio Adoptado:

- Escribimos en minúscula los episodios correspondientes a acciones simples, y en mayúsculas los episodios que referencian a otros escenarios. Esto nos permitió detectar fácilmente las relaciones entre los escenarios.
-

Problema:

- ¿Cómo identificar los escenarios que están “cerrados” para esta fase ?

Criterio Adoptado:

- Consideramos un escenario “cerrado” cuando no presenta el ítem **Dudas**. Esto nos permitió detectar con rapidez aquella información faltante o dudosa.
-

Problema:

- ¿En qué momento explicitar los casos alternativos ?

Criterio Adoptado:

- Describimos los escenarios correspondientes a casos alternativos, una vez definidos todos los escenarios correspondientes a casos normales. Esto permitió dar mayor claridad a cada escenario, desvinculando la especificación de las excepciones a un plano posterior.
-

¿ Por qué describimos estos escenarios y no otros ?

Dado que, el sistema en estudio se desarrolla en distintos lugares físicos, analizamos las funciones que se cumplen en cada sector. Por lo tanto, cada escenario particular involucra las acciones de un sólo sector.

Cada escenario general representa una función fundamental de la aplicación, abarcando desde la solicitud de un trámite de pasaporte hasta la entrega del correspondiente pasaporte.

Es decir, nuestra lista inicial de situaciones (candidatas a escenarios) correspondían a acciones en cada sector o a funciones generales de la aplicación.

A continuación describimos un *escenario particular* y un *escenario general* correspondiente a nuestro caso de estudio.

Nombre: COBRAR TRAMITE

Objetivo: Cobrar el trámite al solicitante.

Contexto: El solicitante debió completar el formulario y pasar por el control de documentación.

Recursos: formulario
máquina timbradora

Actores: Solicitante
Cajero

Set de Episodios:

- El solicitante se presenta con el formulario en la Caja.
- El cajero informa el importe del trámite según el tipo de trámite que figura en el formulario.
- El solicitante paga el trámite.
- El cajero timbra el formulario con el importe.
- El cajero entrega el formulario al solicitante.

Restricciones: el formulario debe tener los datos del solicitante y la marca del tipo de trámite.

Casos Alternativos:
Máquina timbradora falla.

Fig.10 - Ejemplo de escenario particular.

Nombre: TRAMITE DE PASAPORTE ORIGINAL

Objetivo: Dar un pasaporte al interesado por primera vez.

Contexto: El solicitante nunca obtuvo un pasaporte.

Recursos: formulario de solicitud
formulario de donación de órganos
documentación
pasaporte vacío
fotografías
ficha dactiloscópica

Actores: Solicitante
División Documentos y Certificados
División Índice General
División Dactiloscopia
División Legajos y Antecedentes

Set de Episodios:

- ENTREGA Y LLENADO DE FORMULARIOS VACIOS
- CONTROL DE DOCUMENTACION
- SACAR FOTOGRAFIA
- COBRAR TRAMITE
- Si el solicitante tiene número de identificación, se hace OBTENCION DE HUELLAS DIGITALES PARA CONTROL. Caso contrario, se hace OBTENCION DE HUELLAS DIGITALES PARA LEGAJO.
- DERIVACION A CABINA DE RECEPCION
- RECEPCION Y ARMADO DE PASAPORTE ORIGINAL
- Traslado de la carpeta a la División Índice General.
- CONTROL DE PRONTUARIO
- CONTROL DACTILOSCOPICO
- Se asigna un número de identificación al solicitante, si éste no lo posee.
- Perforación del pasaporte con el número de identificación en la parte superior del mismo y en todas las páginas.
- ENTREGA DE PASAPORTE
- Derivación de papeles al Archivo de Legajos.
- Carga de información al Sistema.

Restricciones: solicitante no posee pasaporte.

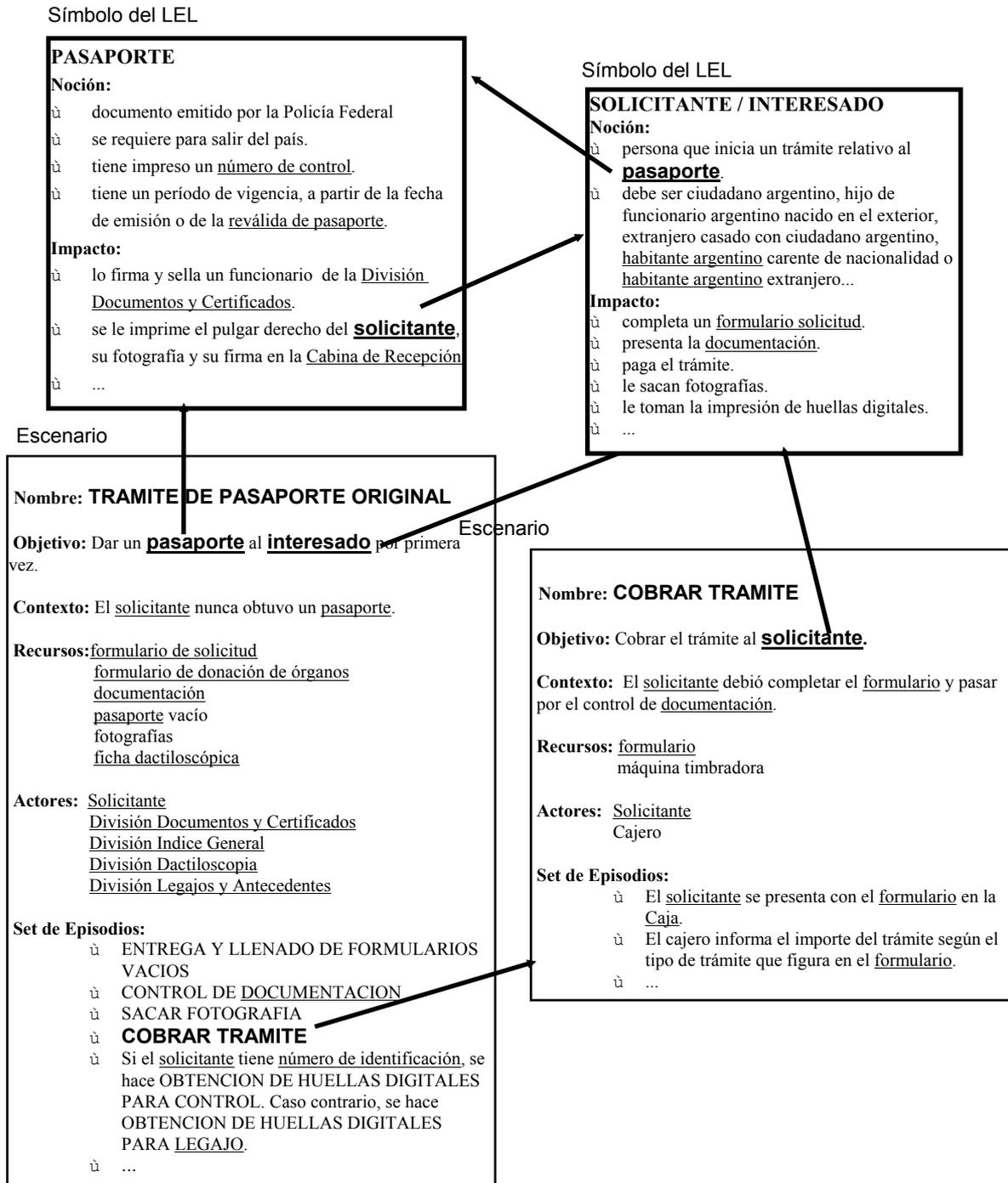
Casos Alternativos:

ENVIO A INCINERACION
DERIVACION DE PASAPORTES OBSERVADOS

Fig.11 - Ejemplo de escenario general.

4.4 RELACION ENTRE EL LEL Y LOS ESCENARIOS

En la descripción de los escenarios, mantuvimos el lenguaje de la aplicación. Para vincular los escenarios y el LEL, subrayamos en los escenarios los símbolos del LEL utilizados. Presentamos a continuación un esquema que muestra: **1)** las relaciones entre los símbolos del LEL, **2)** las relaciones entre los escenarios, y **3)** las relaciones entre el LEL y los escenarios. Estas relaciones se soportan fácilmente con tecnologías de hipertexto [Leite 96].



5 OBSERVACIONES GENERALES

5.1 OBSERVACIONES SOBRE EL LEL

Gran esfuerzo de recolección de vocabulario al no disponer de material escrito.

Para la construcción del LEL, pueden presentarse dos caminos. Uno de ellos parte de información escrita referida al Universo de Discurso, lo que lleva a identificar sobre dicho escrito los símbolos y sus respectivas nociones e impactos. El otro camino consiste en realizar entrevistas orientadas exclusivamente a conocer el vocabulario, en ellas se detectan los símbolos y se reconocen sus significados. Estos caminos pueden combinarse dependiendo de la información que se posee y se requiere.

Utilizar una de estos caminos o ambos, garantiza cumplir con los objetivos del LEL (Cuadro 1).

Dificultad en determinar el nivel de detalle del LEL, cuando se realizan entrevistas orientadas a conocer el problema.

En el caso de realizarse entrevistas orientadas a conocer el problema, se cuenta con un volumen de información mayor, lo que implica mayor nivel de detalle en la información. Esto puede dificultar el proceso de construcción del LEL. Queda pendiente de estudio, si este nivel de detalle volcado en el LEL, también dificulta la comprensión y la utilización del LEL en las fases siguientes del desarrollo.

Dificultad en obtener un LEL homogéneo al trabajar en grupo.

Para obtener un LEL consistente es indispensable fijar criterios sintácticos y semánticos en la definición de los símbolos, principalmente cuando la construcción del LEL está a cargo de un equipo de trabajo. Los criterios a fijar dependen del Universo de Discurso. Cabe destacar, que aún fijando criterios, la utilización de éstos es subjetiva.

Discusión sobre la obtención de un LEL completo al finalizar la fase de elicitación de requerimientos.

Es imposible asegurar la obtención de un LEL que contemple todos los términos involucrados en el dominio de la aplicación. La realización de varias entrevistas a diversos usuarios ayuda a evitar la omisión de símbolos. A pesar de ello, cualquier término del dominio omitido puede incorporarse al LEL en diferentes fases del desarrollo. Esto forma parte de la evolución del léxico a lo largo del desarrollo del sistema.

Conflicto en la detección de sinónimos, al trabajar con varios usuarios.

Para detectar los sinónimos, se requieren varias entrevistas con distintos usuarios. También se pueden detectar sinónimos al identificar nociones e impactos iguales en distintos símbolos.

Detección del mal uso de sinónimos.

La utilización de ciertos términos como sinónimos se debe a usuarios que emplean términos con el mismo significado. Estos términos tienen distinto significado en el dominio de la aplicación. La detección del mal uso de sinónimos puede realizarse a través de entrevistas con diferentes usuarios o en la validación del LEL. Otra forma de detectarlos es en la verificación del LEL, ya que aparece el mismo símbolo como sinónimo y, a la vez, como entrada independiente con otra noción e impacto.

5.2 OBSERVACIONES SOBRE ESCENARIOS

El tiempo de recolección de información se reduce debido al proceso previo de construcción del LEL.

La tarea de generación de escenarios se ve ampliamente beneficiada al disponer del LEL, ya que facilita la comunicación con el usuario, permite contar con información validada y evita distraer la atención en aclarar términos dentro de los escenarios.

Los escenarios facilitan la detección de:

⇒ *omisión en la información*

⇒ *información incorrecta*

⇒ *información confusa.*

La descripción de escenarios permite detectar fácilmente la carencia de información o la existencia de información conflictiva. Esto se debe a la minuciosa descripción que requieren los episodios, en cuanto a detallar el flujo de los eventos y para cada acción, los actores y los recursos involucrados y las restricciones u obstáculos presentes.

Dificultad en obtener escenarios homogéneos al trabajar en grupo.

Al igual que en la construcción del LEL, la redacción de escenarios requiere la determinación de criterios, con el fin de obtener un estilo consistente.

El tiempo de descripción de escenarios es extenso.

La definición de los escenarios involucra un gran esfuerzo y tiempo de realización, debido a la representación utilizada: la descripción narrativa y el esquema empleado (Fig.9).

La representación narrativa :

⇒ *facilita la comprensión de situaciones concretas*

⇒ *dificulta la visión general de la aplicación*

Nuestros escenarios se describen en lenguaje natural, esto facilita la validación con el usuario pero dificulta la comprensión general de la aplicación, o sea las relaciones entre los escenarios. Para visualizar estas relaciones, consideramos conveniente acompañar la descripción narrativa de los escenarios con diagramas que muestren dichas relaciones. Para tal fin, se está investigando una representación gráfica que permita establecer todo tipo de relación entre escenarios y que simultáneamente sea comprensible para el usuario.

6 CONCLUSIONES

Hemos descrito dos representaciones: el LEL y los escenarios, y hemos mostrado cómo usarlas para la elicitación de requerimientos. Es importante notar que ni el LEL ni los escenarios son los requerimientos, ellos son representaciones auxiliares que darán luz en la descripción de los requerimientos de un sistema de software dado. Su uso ayudará a establecer no sólo los requerimientos funcionales sino también los requerimientos no funcionales. Los requerimientos son una colección de sentencias sobre lo que el software debe hacer y los requerimientos no funcionales son condiciones especiales acerca de estos requerimientos o acerca de su cumplimiento. El LEL y los escenarios serán representaciones adjuntas no sólo para la producción de mejores sentencias de requerimientos sino también como una extensión, siendo perspectivas diferentes, acerca de estos requerimientos.

El principal aspecto de nuestro artículo y su mayor contribución es la descripción del fundamento (“rationale”) y de los pormenores en la producción del léxico y de los escenarios. En particular, el uso del conocimiento del dominio para guiar la descripción de nociones e impactos fue una importante contribución para la producción del LEL. Originalmente, [Leite 89] [Franco 92] no utilizaron el conocimiento del dominio elicitado durante la construcción del LEL, para luego pulir su descripción, pero las heurísticas generales presentadas aquí pueden mejorar la calidad total del LEL. Con respecto a los escenarios, es importante destacar cómo se

ha aplicado la idea de la jerarquía en los escenarios. La idea de “escenarios generales” y “escenarios particulares” como una heurística general parece funcionar bien para ayudar a la construcción de esta jerarquía.

Mientras las restricciones reflejan requerimientos no funcionales, los episodios y los objetivos de los escenarios son un origen para derivar los requerimientos funcionales. El LEL sirve como amarra para los significados compartidos, y su relación con los escenarios está fácilmente soportada por tecnología de hipertexto [Leite 96].

Con respecto a escenarios, nuestro trabajo estuvo influenciado por diversos autores, en especial: [Carrol 95], [Zorman 95], [Rubin 92], [Jacobson 92] y [Potts 94]. De Rubin y Jacobson, nos hemos convencido de la importancia de los escenarios para describir mejor el comportamiento de objetos, y la importancia de seguir este comportamiento en los aspectos de interfase del sistema de software. Carrol nos proporcionó una mejor comprensión de los aspectos cognitivos del desarrollo basado en escenarios, así como también una confirmación de nuestra idea de que los escenarios deben nacer en el macrosistema (Universo de Discurso), y no sólo en la interfase del sistema de software. Esta idea fue además reafirmada por el trabajo de Zorman, que bien define a los escenarios como situaciones. De Zorman, también utilizamos su estudio de la representación de escenarios. Potts nos mostró la importancia de relacionar escenarios con objetivos, por la misma característica, fue importante nuestro conocimiento previo del meta modelo orientado a metas propuesto por Dardenne [Dardenne 93]. En particular, diferimos de Potts [Potts 94] en que usamos una jerarquía de escenarios en lugar de una jerarquía de metas. También es importante hacer notar que nuestra construcción de jerarquías utiliza una estrategia bottom-up, diferente de la descomposición de metas de Potts. También, hemos producido heurísticas para encontrar restricciones y excepciones, las cuales serán tratadas a su vez como escenarios.

Los trabajos futuros estarán centrados en la evolución de escenarios, éste es el principal tema del proyecto de PUC-Rio y UB. Particularmente, observaremos cómo los escenarios evolucionan en el contexto de métodos de desarrollo orientados a objeto y cómo éstos se relacionan.

REFERENCIAS

- [Booch91] Booch, G., *Object Oriented Design with Applications*, The Benjamin/Cumming Publishing Company, Inc., Redwood City, 1991
- [Booch94] Booch, G., “Scenarios”, in *Report on Object Oriented Analysis and Design*, Vol 1, 3, Sep-Oct 1994
- [Carrol95] Carrol, J. (ed.), *Scenario Based Design: Envisioning Work and Technology in System Development*, Wiley, New York, 1995.
- [Dardenne93] Dardenne, Al, van Lamsweerde, Fickas, S., “Goal Directed Requirements Acquisition”, *Science of computer Programming*, Vol.20. Apr. 1993.
- [Franco92] Franco, A.P.M., Leite, J.C.S.P., “Uma estratégia de Suporte a Engenharia de Requisitos”, *Anais do XIX Seminario Integrado de Software y Hardware*, pp.200-213, Rio de Janeiro, 1992.
- [Gause89] Gause, D.C., Weinberg, G.M., *Exploring Requirements. Quality Before Design*, Dorset House Publishing, New York, 1989.
- [Gotel94] Gotel, O.C.Z. and Finkelstein, A.C.W.. “An Analysis of the Requirements Traceability Problem”, in *Proceedings of the First International Conference on Requirements Engineering*, Colorado Springs, IEEE Computer Society Press, 1994.
- [Hadad96] Hadad, G., Kaplan, G., Maiorana, V., Balaguer, F., Oliveros, A., Leite, J.C.S.P., Rossi, G. Informe Técnico: “Léxico Extendido del Lenguaje y Escenarios del Sistema Nacional para la Obtención de Pasaportes”. *Proyecto de Investigación*, Departamento de Investigación, Universidad de Belgrano, Buenos Aires, 1996.
- [Jacobson92] Jacobson, Y., Christerson, M., Jonsson, P., Overgaard, G., *Object-Oriented Software Engineering - A Use Case Driven Approach*, Reading, MA: Addison-Wesley; New York: Acm Press, 1992.

- [Lehman80] Lehman, M.M., "Programs, Life Cycles, and Laws of Software Evolution", in *Proceedings of the IEEE* 68(9), pp. 1060-1076 (1980).
- [Leite89] Leite, J.C.S.P., *Application Languages: A Product of Requirements Analysis*, Departamento de Informática, PUC-/RJ, January 1989.
- [Leite90] Leite, J.C.S.P., Franco, A.P.M., "O Uso de Hipertexto na Elicitação de Linguagens da Aplicação", in *Anais de IV Simpósio Brasileiro de Engenharia de Software*, SBC, October 1990.
- [Leite91] Leite, J.C.S.P. and Freeman, P.A., "Requirements Validation Through Viewpoint Resolution", *IEEE Transactions on Software Engineering*, Vol. 17, No. 12, December 1991.
- [Leite93] Leite, J.C.S.P., "Eliciting Requirements Using a Natural Language Based Approach: The Case of the Meeting Scheduler Problem", March 1993.
- [Leite95] Leite, J.C.S.P. and Oliveira, A.P.A., "A Client Oriented Requirements Baseline", in *Proceedings of the Second IEEE International Symposium on Requirements Engineering*, IEEE Computer Society Press, 1995.
- [Mizuno83] Mizuno, Y., "Software Quality Improvement", *Computer*, Vol. 16, 3, march 1983, pp 66-72.
- [Potts94] Potts, C., Takahashi, K., Antón, A.I., "Inquiry-Based Requirements Analysis", *IEEE Software*, Vol. 11, n.2, Mar. 1994.
- [Rubin92] Rubin K.S., Goldberg J., "Object Behavior Analysis", *Communications of the ACM*, Vol. 35, No. 9, Sep. 1992.
- [Zorman95] Zorman, L., *Requirements Envisaging by Utilizing Scenarios (Rebus)*, Ph.D. Dissertation, University of Southern California, 1995.