

Modelos de especificación de requerimientos para la obtención de esquemas conceptuales en un dominio restringido

Fernández Taurant, Juan Pablo – Ing. Marciszack, Marcelo M.

Universidad Tecnológica Nacional, Facultad Regional Córdoba

Abstract

El objetivo de este trabajo es presentar un nuevo enfoque para la especificación de requerimientos en la obtención de un esquema conceptual sobre un dominio de aplicación restringido, en este caso el dominio de las gramáticas formales y máquinas abstractas. Concretamente se utilizará como metodología la requirements baseline, particularmente haciendo uso del léxico extendido del lenguaje (LEL) para modelar el lenguaje del dominio y el modelo de escenarios para representar su comportamiento. La cantidad de información producida por estos modelos será manipulada por un conjunto de heurísticas cuyo objetivo es la obtención del modelo de objetos del dominio. Esta modelización se compone del modelo de tarjetas CRC, que definen al dominio en términos de clases, responsabilidades y colaboraciones. Este trabajo se centrará en la aplicación de esta metodología sobre el dominio propuesto a los fines de utilizar los resultados obtenidos como herramienta de comparación con otras metodologías existentes para evaluar el nivel de isomorfismo que hay entre ellas.

Palabras clave: Requirements Baseline, LEL (Léxico extendido del lenguaje), escenarios, tarjetas CRC, Baseline Mentor Workbench (BMW)

Introducción:

En la actualidad, muchas son las propuestas para solucionar el problema de la llamada crisis del software, dando como resultado una gran variedad de metodologías y herramientas con el objetivo de capturar requerimientos para la obtención de esquemas conceptuales y establecer una definición sin ambigüedades de lo que se quiere producir, lo que constituye una tarea fundamental de la ingeniería de

requerimientos. Está demostrado que los errores cometidos en la fase de especificación de requerimientos crecen en forma exponencial con el transcurso de las siguientes etapas de desarrollo, por lo que las debilidades de la mayoría de los métodos para la obtención de esquemas conceptuales, se reflejan en las primeras etapas del desarrollo. El principal problema derivado de estas debilidades metodológicas radica en la dificultad en determinar si el modelo conceptual refleja fiel y completamente la esencia del dominio. [Insfrán 2002].

Es por esto que se plantea trabajar con una metodología orientada al cliente propuesta por Leite [Leite 1997], que trabaja con una documentación integrada en una estructura llamada Requirements Baseline [Leite 1995; Leite 1997] como soporte del proceso de desarrollo de software basándose en el LEL (Léxico extendido del lenguaje) como elemento principal para modelar el vocabulario del sistema, y mediante el uso de escenarios para representar su comportamiento.

Si bien hay documentación existente sobre trabajos relacionados con propuestas de Especificaciones de requerimientos para la obtención de Esquemas conceptuales, aplicando y comparando diversas metodologías, no se evidencia ningún trabajo que realice esta comparación de aplicación de diversas metodologías / herramientas sobre un determinado caso de estudio (Dominio restringido de Gramáticas Formales y Máquinas abstractas). El

objetivo de éste trabajo se enfoca a la aplicación de éste nuevo enfoque sobre el caso de estudio planteado y obtener un modelo conceptual de especificación de requerimientos que podrá ser utilizado a su vez, como elemento de comparación con otras metodologías y observar el nivel de isomorfismo entre ellas.

Requirements Baseline: *Requirements baseline* es un enfoque orientado al cliente propuesto por Leite [Leite 1997] y se basa en un metamodelo que contiene descripciones sobre el universo de discurso y el sistema de software que será construido dentro de ese universo de discurso. Estas descripciones son escritas en lenguaje natural siguiendo patrones determinados y relacionadas entre sí. El uso de lenguaje natural posibilita validar en todo momento con los stakeholders las especificaciones obtenidas en el análisis de requerimientos. Por otra parte la *Requirements Baseline* es una estructura que evoluciona dinámicamente a lo largo del proceso de desarrollo, acompañando las tareas de mantenimiento y en forma independiente del proceso de desarrollo que se utilice. Esto permite el seguimiento de los requisitos en cualquier punto durante el desarrollo y el mantenimiento hacia su punto de origen [García 1999], que es a lo que llamamos trazabilidad de requerimientos. La *Requirements Baseline*, de acuerdo con [Leite 1997], se compone de cinco vistas:

- Vista del modelo léxico LEL
- Vista del modelo básico BMV
- Vista del modelo de escenarios SMV
- Vista de hipertexto HV
- Vista de configuración CV

2.1.1.- Vista del modelo léxico LEL:

El léxico extendido del lenguaje tiene por objetivo entender el vocabulario manejado por los stakeholders, para de esta forma lograr una mejor comunicación con éstos entendiendo los términos que utiliza para expresarse sin preocuparse por entender el problema que se debe solucionar. Las terminologías utilizadas serán entonces representadas en el LEL como símbolos que describen el lenguaje del dominio del problema o universo de discurso, que con frecuencia son aquellas palabras o frases que más se repiten en las entrevistas y otras que se consideren relevantes para el dominio, puede ocurrir en un gran número de posibilidades, que las personas hagan referencia al mismo símbolo, utilizando diferentes terminologías, que llamaremos sinónimos, para representarlos en las entradas del LEL se utiliza el como separador el elemento ”/”.

Los símbolos deben ser descriptos en dos términos: noción e impacto, la noción representa el significado del símbolo y el impacto las connotaciones o efectos que tendrá el símbolo en el sistema.

Entrada de LEL - Sinónimos	Identificador del símbolo
Noción	Denota el significado del símbolo
Impacto	Connotación o repercusión en el sistema

Tabla 1. Descripción de una entrada de LEL.

La descripción de los símbolos debe hacerse cumpliendo dos reglas [Hadad

1996] [Leite 1993], que se deben cumplir en forma simultánea:

- Principio de circularidad: acotando el lenguaje en función del dominio mediante la maximización de símbolos del lenguaje del LEL, que se logra utilizando en las definiciones de noción e impacto símbolos ya descriptos dentro del LEL.
- Principio del vocabulario mínimo: en donde la tarea es minimizar el uso de símbolos externos al dominio de la aplicación.

Por medio de estos dos principios se logra un vocabulario en forma de red altamente vinculado y auto contenido que permite representar al LEL con formato de hipertexto [Leite 1997]. Los símbolos del LEL pueden definir objetos o entidades pasivas, sujetos o entidades activas, verbos y estados, según sea el tipo de símbolo, sus nociones e impactos tienen una semántica diferente [Leonardi 2001].

Sujeto	<i>Nociones:</i> describen quien es el sujeto.
	<i>Impactos:</i> registran acciones ejecutadas por el sujeto
Objeto	<i>Nociones:</i> definen al objeto e identifica a otros términos con los cuales el objeto tiene algún tipo de relación.
	<i>Impactos:</i> describen las acciones que pueden ser aplicadas al objeto.
Verbo	<i>Nociones:</i> describen quien ejecuta la acción, cuando ocurre, y cuales son los procedimientos involucrados.
	<i>Impactos:</i> describen las restricciones sobre la acción, cuáles son las acciones desencadenadas en el ambiente y las nuevas situaciones que aparecen como resultado de la acción.
Estado	<i>Nociones:</i> describen que significa y que acciones pueden desencadenarse como consecuencia de ese estado.
	<i>Impactos :</i> describen otras situaciones y acciones relacionadas

Tabla 2. Heurísticas para la definición de los símbolos del modelo léxico. [Leite1997][Leonardi 2001]

2.1.2.- Vista del Modelo de Escenarios SMV:

Los escenarios describen la funcionalidad del sistema en un momento determinado teniendo como objetivo principal comprender el sistema en su totalidad. Los escenarios se vinculan fuertemente al LEL ya que lo adoptan como referencia.

Los escenarios adoptan la siguiente estructura [Leonardi 2001]:

- Título: identifica al escenario, puede ser uno o varios.
- Objetivo: meta a lograr en el Macrosistema.
- Contexto: describe la ubicación geográfica y temporal del escenario, así como un estado inicial o precondition.

- Recursos: son los medios de soporte, dispositivos que se necesita estén disponibles en el escenario.
- Actores: son las personas o estructuras de organización que tienen un rol en el escenario.
- Episodios: son una serie ordenada de sentencias escritas de manera simple, que posibilitan la descripción de comportamiento. Pueden ser opcionales, condicionales o simples, y estar agrupados según su forma de ocurrencia en grupos secuenciales o no secuenciales. Para cada escenario se prevé la descripción de excepciones: causas y soluciones a situaciones que discontinúan su evolución natural, e impiden el cumplimiento del objetivo. Estas

generalmente reflejan la falta o mal funcionamiento de un recurso. El tratamiento de la excepción puede estar dado por un escenario.

Se puede hacer una analogía de los escenarios con la metodología de [Jacobson 1992] de casos de uso, que es utilizada ampliamente en la actualidad, con algunas diferencias. Los escenarios pueden ser vistos como casos de uso donde también encontramos un objetivo como meta a cumplir, un contexto, recursos asociados, actores, etc. Siendo los episodios una suerte de descripción de los cursos normales y alternativos de los casos de uso. Pero con las siguientes diferencias [Leonardi 2001]:

- Un escenario describe situaciones del Macrosistema.
- Un escenario evoluciona durante el proceso de desarrollo del software siguiendo la filosofía de la *Requirements Baseline* a la cual pertenece.
- Los escenarios están naturalmente conectados al LEL.
- Un escenario describe situaciones, poniendo énfasis en el comportamiento del macrosistema. Usa lenguaje natural para su representación.

Cada episodio de un escenario puede ser representado como un escenario en sí mismo. Por otro lado también pueden representarse los requerimientos no funcionales del dominio como restricciones.

2.1.3.- Vista del modelo básico BMV

La vista del modelo básico contiene todos los requerimientos de los stakeholders con todas las acciones y subacciones que realizan, las acciones son sus actividades concretas, que son disparadas por los eventos externos y que representan necesidades de información que el sistema debe realizar

con restricciones impuestas por el dominio (normas, etc.) y los diagnósticos, que son observaciones que señalan problemas.

2.1.4.- Vista de hipertexto VH

La vista de hipertexto integra las vistas de LEL, BMV y SMV. El principio de circularidad del LEL permite que la forma ideal de representarlo sea mediante hipertexto, ya que es un hipertexto en sí mismo. El SMV posee las mismas características puesto que su descripción está basada en el lenguaje contenido en el LEL, lo mismo sucede con la BMV, como consecuencia la vista de hipertexto es ortogonal a las otras tres vistas y al estar todas ellas vinculadas constituye una herramienta vital que integra a las demás vistas.

2.1.5.- Vista de configuración CV

Esta vista es la que contiene el sistema de versiones de la *Requirements Baseline*, mantiene todos los cambios efectuados en el resto de las vistas. El registro de los cambios que se producen en las vistas permite mantener un registro de la evolución de las vistas, esto es imprescindible ya que contribuye formar un modelo consistencia y con una fuerte trazabilidad.

3.- Construcción del LEL y Escenarios

Para el proceso de construcción del LEL y escenarios [Hadad 1997; Doorn 1998; Hadad 1999; Leite 2000] proponen hacerlo en dos etapas:

1. Construcción del LEL.
2. Construcción de los Escenarios a partir del LEL.

Esto se debe a la fuerte vinculación que tienen estos dos elementos.

3.1.- Proceso de Construcción del LEL

El modelo léxico se construye en etapas interdependientes, que a su vez pueden hacerse de forma simultánea:

- Entrevistas
- Generación de la lista de símbolos
- Clasificación de los símbolos
- Descripción de los símbolos
- Validación con los clientes
- Control del LEL

Entrevistas:

Las entrevistas tienen como objetivo conocer el lenguaje de los stakeholders, en las primeras entrevistas el entrevistador debe dejarlos expresarse libremente para que usen su propio vocabulario, a medida que la entrevista avanza, el entrevistado irá repitiendo algunos términos y a relacionarlos con otros utilizando de forma natural el principio de circularidad, el entrevistador por su parte, comienza a adquirir el lenguaje del dominio de la aplicación.

Generación de la lista de símbolos:

Luego de la primera entrevista se arma una lista de símbolos candidatos, que contiene las palabras y frases que los stakeholders han utilizado con mayor frecuencia. Con el transcurso de las entrevistas la lista de símbolos candidatos se modifica validando símbolos, sinónimos, se agrega nuevo conocimiento y se eliminan las entradas erróneas armando la lista definitiva de símbolos. Otro método que puede utilizarse en forma simultánea a las entrevistas es la lectura de los documentos de la organización, ya que puede brindar al analista gran cantidad de información del lenguaje utilizado en el dominio.

Clasificación de los Símbolos:

La clasificación de los símbolos se utiliza para asegurar la integridad y homogeneidad de las descripciones [Hada 1996], esta clasificación puede hacerse en forma general agrupando los símbolos según sujeto, verbo, objeto, estado, o mediante algún método particular.

Descripción de los símbolos:

Con el desarrollo de las entrevistas el analista va tomando conocimiento del significado de los símbolos que se incluyen en la lista, la descripción de los símbolos es agregar ese significado a los símbolos en términos de la definición de su noción e impacto.

Este proceso puede soportarse a través de una serie de reglas propuestas por [Antonelli 2003]:

- Un símbolo puede tener una o más nociones y cero o más impactos.
- Cada noción e impacto debe ser descrito con oraciones breves y simples.
- Nociones e impacto para un símbolo pueden representar diferentes puntos de vista o pueden ser complementarios.
- Las oraciones breves y simples de las nociones e impactos deben responder a los principios de circularidad y de vocabulario mínimo.
- Para los símbolos que son sujeto de una oración, los impactos deben indicar las acciones que realiza.
- Para los símbolos que cumplen el rol de verbo, las nociones deben decir quién ejecuta la acción, cuándo sucede y el proceso involucrado en la acción. Para los impactos se deben identificar las restricciones sobre la realización de la acción, qué es lo que

origina esta acción y qué es lo que causa esta acción.

- Para los símbolos que son objetos de una oración, la noción debe identificar otros objetos con los cuales se relaciona y los impactos serán las acciones que se pueden realizar con este signo.

Validación con los clientes:

La validación con los stakeholders permite corroborar el lenguaje del vocabulario del dominio que se obtuvo en etapa de entrevistas, validando el significado de los símbolos en las primeras entrevistas, mientras que en las siguientes validaciones se chequea por lo general que el vocabulario obtenido sea el correcto, si falta información se introduce el nuevo conocimiento en el LEL.

Control del LEL

El control del LEL tiene como finalidad verificar que el LEL terminado sea consistente y homogéneo, se controlan los símbolos verificando que pertenezcan a la clasificación correcta y que no existan sinónimos como símbolos diferentes, es fundamental que este proceso comience con las primeras descripciones de los símbolos.

3.2.- Proceso de construcción de escenarios

La construcción de los escenarios se realiza a partir de la derivación del LEL, los pasos para realizar esta derivación son los siguientes:

- Identificación de los actores de la aplicación.
- Generación de la lista de escenarios candidatos, a partir de los actores Principales.
- Descripción de los escenarios candidatos, provenientes de los actores Principales.

- Ampliación de la lista de escenarios candidatos, a partir de los actores Secundarios.
- Descripción de los escenarios candidatos, provenientes de actores Secundarios.
- Revisión de los escenarios.
- Validación de escenarios.

Se define como actores a los usuarios de la aplicación y deben ser englobados bajo la categoría sujeto en el LEL, los actores serán primarios o secundarios según el rol que desempeñe cada usuario, los primarios son los que se encuentran en contacto directo con la aplicación, el secundario solo recibe información. La acción descrita como impacto en el LEL de los actores serán escenarios candidatos y se describen con la misma acción en infinitivo, eliminando aquellos que puedan repetirse. Los escenarios son el desarrollo de la acción que realiza el actor, en donde se especifican las precondiciones que debe cumplir esa acción, actores intervinientes (SUJETOS), recursos involucrados (OBJETOS), descripción de cada episodio y un objetivo que los actores deben cumplir. Luego se describe cada escenario utilizando la información contenida en el LEL. En este proceso se comienzan definiendo escenarios generales que representan toda la funcionalidad del sistema, luego se van refinando en detalle dando lugar a subescenarios que también son generados cuando uno o mas episodios necesitan llevarse en forma independiente. Una vez terminada la descripción de los escenarios se hace una revisión de los escenarios para verificar su consistencia y se arma una lista con los escenarios propuestos que serán validados con los stakeholders en posteriores entrevistas, en donde también se evacuarán las dudas surgidas

durante el proceso para realizar las correcciones necesarias. Luego de inspeccionar y validar los escenarios se arman los escenarios integradores, que no son otra cosa que una jerarquía los escenarios y subescenarios ya descritos, conformando grupos en función del contexto y del objetivo a cumplir. En cada grupo se debe establecer un orden de ejecución de escenarios que lo componen, en donde cada uno de ellos pasarán a ser episodios del escenario integrador que también debe ser descrito como un escenario pero teniendo en cuenta el contexto conformado por la combinación de los escenarios que lo componen, uniendo también recursos y actores que pasarán a ser los del escenario integrador.

En el caso de que un escenario no se incluya en ningún escenario integrador se debe analizar su pertenencia al dominio o si es una excepción no detectada.

4.- Tarjetas CRC

Las tarjetas CRC (colaboraciones y responsabilidades de las clases) son obtenidas a partir del LEL y los escenarios, son utilizadas para identificar las posibles clases que corresponden a la solución orientada a objetos.

Al igual que en un modelo de objetos cada tarjeta CRC representa un objeto del mundo real, estos objetos no son solamente físicos, sino que también se modelan como objetos aquellas entidades abstractas que cumplen un rol definido en el dominio del problema y de la solución. Las tarjetas CRC se obtienen a partir del LEL y los escenarios a través de heurísticas de derivación que nos permite encontrar las responsabilidades y colaboraciones de las clases que pasarán a ser de análisis, pudiendo transformarse en clases de diseño y posteriormente las del modelo preliminar de objetos.

La derivación del LEL y escenarios a tarjetas CRC puede hacerse con la estrategia propuesta por [Leonardi 2001] dividida en tres partes:

- *Encontrar CRC primarias.*
- *Encontrar CRC secundarias.*
- *Encontrar colaboraciones.*

Las tarjetas CRC primarias representan objetos con comportamiento relevante dentro del dominio. Los actores de los escenarios son candidatos a convertirse en CRC primarias ya que realizan las tareas del escenario para lograr su objetivo. Los actores son entradas de LEL de la categoría sujeto. Estos sujetos realizan acciones en el dominio, las que son descriptas en los impactos. Entonces, los impactos no son otra cosa más que los servicios que provee, estos servicios son las responsabilidades de la CRC. De esta forma la tarjeta CRC es obtenida de un actor de un escenario y sus responsabilidades son los impactos de la entrada del LEL.

Las CRC secundarias son aquellos colaboradores de las CRC primarias que las ayudan a cumplir con sus responsabilidades. Las CRC secundarias se encuentran en las responsabilidades de las CRC primarias. Aquellos términos de las responsabilidades que también están definidos en el LEL se convierten en CRC secundarias. Las responsabilidades se obtienen de la misma forma que se obtienen las responsabilidades de las CRC primarias. Debido a que las CRC secundarias se obtienen a partir de las responsabilidades de las CRC primarias, es trivial que colaboran en alguna medida. Sin embargo, para completar las colaboraciones, es necesario analizar los escenarios. En los episodios de los escenarios se deben buscar las entradas de LEL que originan a las CRC tanto primarias como secundarias. Las

tarjetas CRC que participan de un mismo escenario colaboran entre ellas. Como las tarjetas CRC se obtienen a partir de la derivación del LEL y escenarios, cualquier cambio producido en el LEL y los escenarios llevará a producir una nueva derivación de tarjetas CRC, hecho de gran importancia, ya que permite mantener una trazabilidad directa entre los requerimientos y el modelo de objetos que les darán soporte.

5.- Herramientas automatizadas de soporte

Forward traceability es de suma importancia en los proyectos de software y en particular en la gestión de requerimientos, puesto que el volumen de información que se debe manejar es inmenso, sin una herramienta esta tarea es imposible [Alspaugh 1999] [Ramesh 1998].

Si sucede algún cambio en cualquiera de los requerimientos se deben reexaminar los links de trazabilidad para evaluar los productos obtenidos a partir de los requerimientos que cambiaron. Luego se debe determinar si todavía son válidos como para que sigan existiendo, o si necesitan crear nuevos [Cockton 2001]. Con grandes volúmenes de información es imposible realizar esta tarea manualmente, porque el tiempo que consumiría sería muy grande y la posibilidad de cometer errores alta [Ramesh 1998].

En virtud de la necesidad de una herramienta automatizada que nos

permita esta gestión de requerimientos se propone en este trabajo el uso de Baseline Mentor Workbench (BMW), una herramienta que nos permite trabajar con LEL, escenarios y tarjetas CRC.

5.1.- Baseline Mentor Workbench

Baseline Mentor Workbench (“BMW”) es una herramienta que tiene como función asistir al experto del dominio durante la fase de ingeniería de requerimientos utilizando la metodología del Client Oriented Requirements Baseline [Antonelli 1999], gestionando las entradas del LEL, escenarios y tarjetas CRC.

Cada dominio dentro de BMW se denomina proyecto, y su evolución a lo largo del tiempo se registra en un sistema de versionado. Las entradas del LEL deben ser ingresadas en forma manual al igual que los escenarios, luego las tarjetas CRC son obtenidas en forma automática por la herramienta. Lógicamente, cada una de las versiones obtenidas conforme a la evolución del proyecto y sus modificaciones contienen sus propias entradas de LEL, escenarios y tarjetas CRC.

La arquitectura de la aplicación es una típica Arquitectura de Repositorio [Sommerville 1995], se compone de cuatro subsistemas: editor de entradas de LEL, editor de escenarios, generador de tarjetas CRC y browser de navegación.

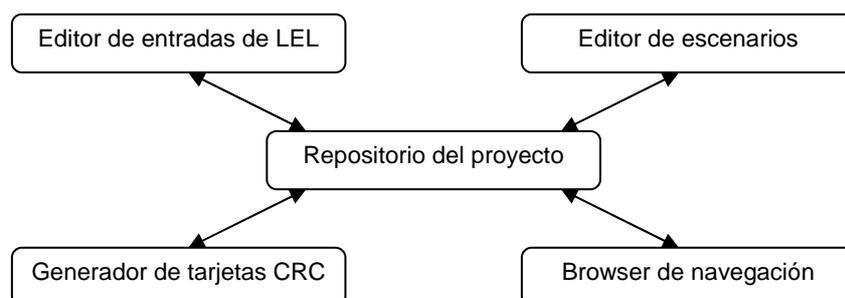


Figura 1. Arquitectura de Baseline Mentor Workbench [Antonelli 2003].

La descripción de las entradas del LEL y escenarios se realizan a través de templates que son seleccionados desde el menú contextual, de esta forma el usuario solo debe completar los templates con la información requerida, durante este procedimiento la aplicación verifica si las palabras que se ingresan existen dentro de las entradas del LEL, si es así se señala la palabra para indicar el link correspondiente en forma automática, si la palabra es nueva el usuario selecciona la palabra e indica que debe ser un link, de esta forma se agrega una nueva entrada al conjunto de entradas del LEL.

BMW deriva tarjetas CRC a partir de las entradas del LEL y los escenarios de acuerdo al algoritmo de derivación expuesto en [Leonardi 2001]. Sin embargo, la implementación en el BMW es una simplificación de este algoritmo que de todas formas cumple con las etapas principales [Antonelli 2003]:

- Hallar clases primarias. Son los actores de los escenarios que también están en el conjunto de entradas del LEL.
- Hallar responsabilidades de las clases primarias. Se toman de los impactos de las entradas del LEL.
- Hallar clases secundarias. Son las referencias a entradas del LEL que se encuentran en las responsabilidades de las clases primarias.
- Hallar responsabilidades de las clases secundarias. Se toman de los impactos de las entradas del LEL.
- Hallar colaboraciones. Consiste en determinar con que otras tarjetas CRC participa en la resolución de cada escenario.
- Depurar las tarjetas CRC. Consiste en revisar y eliminar tarjetas CRC y atributos repetidos.

Cada entrada de LEL o escenario debe tratar de describirse siempre en términos de otras entradas, que es lo que se propone en el principio de circularidad, esto nos brinda la posibilidad de navegar por toda la información contenida dentro del LEL en un hipertexto.

Conclusión:

En la actualidad existe una gran cantidad de documentación sobre trabajos relacionados con propuestas de especificaciones de requerimientos para la obtención de Esquemas conceptuales, pero no se evidencia ninguno que pueda utilizarse para realizar una comparación sobre la aplicación de diversas metodologías / herramientas sobre un determinado caso de estudio, en este caso, el dominio restringido de gramáticas formales y máquinas abstractas.

Se puede visualizar a priori, frente a la elección de este dominio, que al pretender modelar una Metaabstracción (crear una abstracción sobre otra abstracción) de la realidad, como lo son las gramáticas formales y máquinas abstractas, no es intuitivo y la metodología seleccionada presenta algunas ambigüedades.

Para dar continuidad de este trabajo a futuro, se pretende aprovechando que son isomorfos los dominios de Gramáticas y Máquinas, comprobar si los modelos resultantes de la aplicación de la metodología también lo son. Otra línea para la continuidad de este trabajo será el de utilizar el esquema conceptual resultante, para confrontarlo con otras metodologías existentes de manera de poder realizar una comparación de las mismas en este dominio.

Nota

El presente trabajo ha sido realizado dentro del marco del proyecto de investigación *“Construcción de Herramientas Didácticas para la enseñanza y ejercitación práctica en laboratorio de Informática Teórica en las Carreras con Informática”*, con el apoyo de la Agencia Córdoba Ciencia, y bajo la dirección del Ing. Marcelo Marciszack.

Bibliografía

- [1] [Alspaugh 1999] Alspaugh, T.A., Antón, A.I., Barnes, T., Mott, B., “An Integrated Scenario Management Strategy”, Fourth International Symposium on Requirements Engineering, RE'99, Limerick, Ireland, Junio, 1999, pp 142-149.
- [2] [Antonelli 1999] Antonelli, L., Oliveros, A., Rossi, G., “Baseline Mentor, An Application that Derives CRC Cards from Lexicon and Scenario”, XXVIII JAIIO, II Workshop Iberoamericano en Ingeniería de Requerimientos, WER'99, Buenos Aires, Argentina, Septiembre 9 y 10, 1999.
- [3][Cockton 2001] Cockton, G., "Let's Get It all Together: Literate Development and the Integration of HCI Research," ACM CHI'98 Basic Research Symposium, Los Angeles, USA, Abril, 19-20 1998, disponible en http://www.cet.sunderland.ac.uk/~cs0gco/brs_fu11.htm, Version 1.2 de 11/4/01 accedido el 15/1/2003, 2001.
- [4][Doom'98] Doorn J, Kaplan G, Hadad G, Leite JCP “Inspección de escenarios” Anais WER98: Workshop de Engenharia de Requisitos, Departamento de Informática. Puc-Rio, pp. 57-69.
- [5][García'00] García O, Gentile C “Escenarios del proceso de construcción de escenarios Autoaplicación de la metodología” Tesis de grado. Depto De Computación y Sistemas, UNCPBA, Argentina.
- [6][Hadad'96] 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.
- [7][Hadad'97] Hadad, G., Kaplan, G., Oliveros, A., Leite, J.C.S.P., “Construcción de

Escenarios a partir del Léxico Extendido del Lenguaje” JAIIO'97, SADIO Buenos Aires, 1997, pp. 65-77.

[8][Hadad99] Hadad, G., Kaplan, G., Oliveros, A., Leite, J.C.S.P.,

“Integración de Escenarios con el Léxico Extendido del

Lenguaje en la elicitación de requerimientos: Aplicación a un

caso real”, Revista de Informática Teórica y Aplicada, Vol 6,

Nro. 1, Julho 1999.

[9][Insfrán 2002] Emilio Insfrán, Isabel Díaz y Burbano Margarita. Modelado de Requisitos para la Obtención de esquemas conceptuales.

Disponible en : <http://www.dsic.upv.es/~einsfran/papers/39-ideas2002.pdf>

[10][Jacobson92] Jacobson I. et al. Object Oriented Software Engineering: A

Use-Case Driven Approach. Addison Wesley, 1992

[11][Kaplan] Kaplan,V.,Hadad, G., Oliveros, A., Uso de Lexico Extendido

del Lenguaje (LEL) y de Escenarios para la Elicitacion de

Requerimientos. Aplicación a un Caso Real, Informe de

Investigación Dpto. de Investigación de la Universidad de

Belgrano, Argentina

[12][Leite'93] Leite, J.C.S.P., “Eliciting Requirements Using a Natural

Language Based Approach: The Case of the Meeting

Scheduler Problem”, March 1993.

[13][Leite95] Leite J.C.S.P, Albuquerque Oliveira, A P. A Client Oriented

Requirements Baseline. Proceedings of RE 95': Second IEEE

International Symposium on Requirements Engineering.

Inglaterra, Marzo 1995.

[14][Leite'97] Leite J.C.S.P., Rossi G.,et al. Enhancing a Requirements

Baseline with Scenarios. Proceedings of RE 97': International

Symposium on Requirements Engineering, IEEE. Enero 1997

[15][Leonardi 2001] C. Leonardi, J.C.S. Leite, Gustavo Rossi. “ Una estrategia de Modelado

Conceptual de Objetos , basada en Modelos de requisitos en lenguaje natural ”.Tesis de

Maestría Universidad Nacional de la Plata.

<http://postgrado.info.unlp.edu.ar/Carrera/Magister/Ingenieria%20de%20Software/Tesis/Leonardi.pdf>

[16][Loucopoulos 1995] P. Loucopoulos, V. Karakostas. System Requirements Engineering, McGraw-Hill 1995.

[17][Ramesh 1998] Ramesh, B., “Factors Influencing Requirements Traceability Practice”, Communications of the ACM Vol 41 Nro 12, Diciembre, 1998, pp 37-44.