



Universidad Nacional de La Matanza

Unidad Académica donde se encuentra acreditado:

Código: C144

Título del Proyecto: **ESTABILIDAD DE LOS DOCUMENTOS EN EL PROCESO DE REQUISITOS**

Programa de Investigación: PROINCE

Director del Proyecto: Doorn, Jorge Horacio

Co Directora: Kaplan, Gladys Noemí

Integrantes del Proyecto: Guatelli Renata, Gigante Nora Cristina

Fecha de inicio: 2013/01/01

Fecha de finalización: 2014/12/31

La estrategia en Ingeniería de Requisitos que se ha desarrollado en proyectos de investigación precedentes [Leite 04] no ha considerado con suficiente profundidad dos aspectos del proceso de requisitos. Por un lado, la introducción de ambigüedades en los documentos finales del proceso por obsolescencia del LEL. Por otro lado, las técnicas de validación de los escenarios futuros utilizados son susceptibles de ser mejoradas.

Palabras claves: procesos de requisitos, léxico extendido del lenguaje, evolución semántica.

Área de conocimiento: Ingeniería Electrónica y Comunicaciones

Código de Área de Conocimiento: 1800

Disciplina: Computación

Código de Disciplina: 1802

Campo de Aplicación: Computación

Código de Campo de Aplicación: 1802

Otras dependencias de la UNLaM que intervinieron en el Proyecto:

Otras instituciones intervinientes en el Proyecto:

Otros proyectos con los que se relaciona:

**Título del Proyecto: ESTABILIDAD DE LOS DOCUMENTOS EN EL PROCESO DE REQUISITOS****1. Resumen**

Este proyecto es una continuación de los proyectos previos “Validación de Requisitos de Software” (55/C060) 2006-2008, “Consolidación de Requisitos” (55/C097) 2009-2010 y “Contextualización del Proceso de Requisitos” (55/C124) 2011-2012.

Se ha analizado la evolución en el contexto durante el proceso de requisitos en general y en el proceso de requisitos basado en escenarios en particular. Los cambios en el léxico tratados inadecuadamente pueden llevar a modelos con un alto grado de imprecisión, llegando en los peores casos a generar requisitos erróneos. Para evitar estos inconvenientes y asegurar la calidad de los requisitos del software, es necesario que el léxico evolucione a lo largo del proceso de requisitos para reflejar en todo momento el proceso del negocio y los servicios del futuro sistema software. Para cumplir con este objetivo se han analizado varios mecanismos para introducir en el glosario (LEL) nuevos símbolos y modificar algunos existentes. Finalmente se ha definido una heurística para crear un nuevo glosario denominado LEL de Requisitos (LELR) que contiene una lista inicial de cambios semánticos detectados durante la construcción de los escenarios futuros (EF) y terminado una vez que los EF están completos. Esta heurística pone un gran énfasis en la mejora de la legibilidad de los EF y de la ERS alcanzando exitosamente dicho objetivo en los diferentes casos de estudios donde la misma ha sido implementada.

Palabras clave: procesos de requisitos, glosarios, evolución semántica, LEL de Requisitos.



2. INTRODUCCIÓN

Es bien conocido que el lenguaje es un reservorio de conocimiento. Es así que los lenguajes específicos de ciertos dominios de aplicación contienen información acerca de estos dominios, reflejan ambiente y tecnología, como es el caso del árabe que tiene 80 palabras para mencionar a los camellos, los japoneses que tienen más de 20 palabras para el arroz o los inuit con más de 20 palabras para la nieve y el hielo [Nettle 00] [Fishman 99]. En otras palabras, de alguna manera el conocimiento está parcialmente consolidado en la terminología del dominio, por lo que acceder a esa terminología es un camino más directo para la adquisición de conocimiento ya que se aprovecha esa síntesis pre existente. Este es el núcleo central en el que se basa el proceso de requisitos que se inicia con la construcción del LEL y que se ha denominado como un proceso middle out porque se accede primero al núcleo del conocimiento (el léxico utilizado) y luego se lo completa con información adicional. También es conocida la necesidad de construcción de glosarios en forma explícita o implícita que tienen los docentes al abordar un tema nuevo. Por ejemplo, es imposible abordar la dinámica sin definir con precisión los términos fuerza, trabajo, energía y potencia.

Para la representación de estos núcleos de conocimiento adquiridos o generados durante toda la etapa de Ingeniería de Requisitos es utilizado en de modelos el lenguaje natural (LN). Entre los más frecuentes se encuentran diferentes tipos de glosarios, casos de uso y escenarios. Un estudio sobre la práctica de la Ingeniería de Requisitos [Neill 03] ha comprobado que más del 50% de las organizaciones utiliza escenarios y casos de uso.

La construcción de glosarios [Rolland 98] [Oberg 98] [Wiedenhaupt 98] [Alsplaugh 99] [Robertson 06] como parte del proceso de desarrollo de software en general y de la Ingeniería de Requisitos en particular ha sido reconocida como una actividad necesaria. Estos glosarios son utilizados durante el proceso de requisitos haciendo de ancla en toda la documentación generada. Sin embargo se ha prestado poca atención a la obsolescencia que suelen padecer estos glosarios durante el desarrollo del nuevo sistema de software y, en muchos casos, aún antes de definir los requisitos.

En el proyecto objeto del presente informe final se trabajó en el marco de un proceso de requisitos basado en escenarios [Leite 04] donde todos sus modelos son representados en LN. Uno de los pre conceptos erróneos relacionados con el uso de LN en especificaciones de todo tipo, está relacionado con la ambigüedad y flexibilidad del mismo. Sin embargo esto no ocurre en los documentos utilizados en este proceso ya que los mismos son de fácil lectura pero no de fácil construcción ya que la misma es guiada por heurísticas y sintaxis precisas que hace que los distintos componentes guarden entre sí, una coherencia estructural y semántica.

La utilización del LN para modelar acarrea tanto sus ventajas como sus desventajas. Entre las ventajas podemos citar una mejora significativa en la comunicación, una mejor comprensión de los modelos por parte del usuario y de esta manera la posibilidad de realizar una actividad de validación más segura. Pero por otro lado, el LN tiende a incluir ambigüedad y aumentar la falta de precisión en las descripciones. De esta manera se hace necesario incluir mecanismos para controlar estos inconvenientes siendo un glosario un instrumento eficaz ya que permite ser consultado en cualquier momento del proceso de requisitos. Es importante entonces destacar la importancia de contar con estos glosarios muy tempranamente en el proceso de requisitos resaltando la necesidad de actualizarlos para acompañar la evolución del proceso, tema central del presente proyecto.

La estrategia de requisitos propuesto por Leite et al. [Leite 04], en la cual, como ya se mencionó, se ancla el presente proyecto, consiste:



- 1) crear un glosario denominando léxico extendido del lenguaje (LEL) [Hadad 08] y un conjunto de escenarios actuales, que representan situaciones observables para la comprensión del universo de discurso
- 2) construir un conjunto de escenarios futuros (EF) [Doorn 02] que representan situaciones con el nuevo sistema del software a construir, y finalmente, la especificación de requisitos de software (ERS) basándose en el conocimiento adquirido y registrado en los EF.

Otro aspecto relevante es que el conocimiento de un dominio de aplicación no es estático, cambia con el transcurso del tiempo. Es así que la mera consideración de la introducción de un nuevo sistema de software en el proceso del negocio obliga a repensar algunos aspectos de este, lo que naturalmente se ve reflejado en el cambio de significado de algunos términos y en la aparición de otros. Sin embargo, esta es una visión parcial del problema porque esto es lo que ocurre entre los clientes y usuarios pero del lado de los ingenieros de requisitos el mecanismo de introducción o reuso de términos es más intenso ya que se necesita una nueva terminología para describir los servicios del nuevo sistema de software. Es así que el alcance del presente proyecto se extendió desde el LEL original hasta la evolución del mismo en el dominio por un lado, y hasta los términos introducidos por el ingeniero de requisitos por el otro.

El LEL fue creado para bajar la ambigüedad en los modelos construidos en LN y su objetivo principal es representar el léxico utilizado en el dominio. Esta definición restringe el alcance del LEL e impide la inclusión en el mismo glosario, de aquella parte del léxico que se modifica al pensar en la solución para el nuevo sistema de software y en aquellos otros términos inexistentes pero necesarios para describir los nuevos procesos, tareas, recursos, etc. De esta manera, cuando el LEL no evoluciona junto al resto del proceso, paradójicamente en vez de bajar la ambigüedad de los modelos construidos, incluida la ERS, la agrega. Esto motivo la necesidad de estudiar la evolución semántica del léxico durante todo el proceso de requisitos y analizar su incorporación en el mismo LEL o en otro glosario.

Una vez que los cambios comienzan a ser planificados en el proceso de negocio la falta de evolución del léxico incorpora cierto nivel de ambigüedad en los documentos generados. Esto desvía el objetivo principal del LEL y baja la calidad de los EF construidos y de la ERS. Por lo tanto, afecta particularmente a las etapas de definir la solución del proceso de requisitos. De esta manera la continuidad en los modelos del proceso de requisitos se ve seriamente en riesgo. Esto motivo el título del proyecto “proveer la continuidad de los documentos del proceso de requisitos”, centrando la atención en que el LEL debe evolucionar y para ello se definió la necesidad de “profundizar el estudio de mecanismos para determinar la evolución del léxico”, “desarrollar una heurística para la construcción del LEL evolucionado” y “mejorar la visualización de los escenarios futuros”. Estos objetivos se basaron en la hipótesis de que “existe una fuerte permeabilidad conceptual entre los clientes y los desarrolladores dado que los documentos que crean los desarrolladores motivan, por un lado, a los clientes a comprender la interacción entre el nuevo sistema de software y el proceso del negocio y, por el otro, a los desarrolladores en las problemáticas específicas de los clientes.

3. DESARROLLO

El método de trabajo se fundamenta en la selección de casos creados en su mayoría en trabajos previos al presente proyecto de investigación. Por otro lado, el proceso de requisitos basado en escenarios utilizado para realizar todo el análisis que se describe a continuación, también surge de un conjunto de trabajos de investigación previos que datan del año 1995 a la fecha.

Estos casos de estudio existentes se comienzan a generar a partir del año 2001 en la Cátedra de Ingeniería de Requisitos de UTN-FRBA. Estos casos de estudio fueron desarrollados por alumnos de 4to. año de la carrera de grado Ingeniería en Sistemas de Información al realizar



una transferencia de conocimiento desde los proyectos de investigación hacia la cátedra, donde los alumnos realizaron sus Trabajos Prácticos durante todo el cuatrimestre utilizando el proceso de requisitos antes mencionado. Se continuó investigando en la misma línea en UNLaM y nuevamente se transfirió el conocimiento de los proyectos a la Cátedra de Ingeniería de Requerimientos de la Carrera de grado Ingeniería en Informática. En paralelo se trabajó en otras Universidades como ser en la UNLP, UB, UCA, UNICEN donde han aplicado el proceso de requisitos basado en escenarios parcial o totalmente en diferentes proyectos de investigación y lo han transferido a clase con diferente intensidad. Tanto en UTN como en UNLaM, los alumnos han trabajado en grupos llevando adelante una ingeniería de requisitos en una organización y sector a su elección, obteniendo como resultado un documento de requisitos generado de la aplicación misma del proceso. Estas organizaciones, si bien eran reales, no respondieron en todos los casos a la necesidad de construir un nuevo sistema de software. Por lo tanto, se pueden dividir en dos grandes grupos: los que tenían una problemática particular a resolver con un sistema de software y los que no tenían una dificultad emergente en relación a los sistemas de información. El primer grupo generó la ERS conjuntamente con los clientes y los usuarios. En el segundo grupo, los alumnos propusieron los cambios en los procesos del negocio futuro de manera unilateral. Ambos casos fueron muy enriquecedores y un desafío importante tanto para los alumnos como para los docentes.

En simultáneo a los proyectos y a la transferencia en aula, se han desarrollado casos reales en organizaciones privadas con la particularidad de estar a cargo de los docentes de la cátedra. Estos casos, tenían el objetivo de construir un sistema de software. A continuación se mencionan dichos proyectos:

- Sistema de Control de Calidad de Productos Farmacéutico (2002).
- Sistema de Mesa de Ayuda para clientes propietarios de Cajeros Automáticos (2003).
- Sistema de Producción y Comercio exterior (2003-2006)
- Sistema Integral de Alumnos para una Universidad Privada (2009).

Para confirmar la hipótesis del presente proyecto sobre la permeabilidad de los documentos del proceso de requisitos, se tomaron dos caminos. Uno de ellos fue analizar los casos existentes recién mencionados y el otro fue construir otros casos de estudio aplicando el nuevo conocimiento generado en este proyecto.

En relación a los casos existentes, se puede contabilizar aproximadamente 150 casos que tienen la característica de estar libres de toda contaminación y prejuicio proveniente de los objetivos e hipótesis del presente proyecto. Se detectó una evolución del léxico, mencionada en la introducción y eje central del presente trabajo, producida naturalmente en el desarrollo de la ingeniería de requisitos, por lo que han sido seleccionados con el objetivo de analizar qué símbolos del LEL se han utilizado en los escenarios futuros con un significado diferente del registrado originalmente en el glosario.

El otro camino, construir nuevos casos de estudio, se realizó en paralelo con el presente proyecto de investigación, donde se generaron aproximadamente 40 casos nuevos. En estos casos de estudio construidos por los alumnos, se utilizó la heurística para probar su efectividad y realizar los ajustes necesarios. Durante la utilización del proceso de requisitos basado en escenarios se generaron los siguientes modelos:

- LEL,
- Inspección del LEL,
- derivación del LEL a EA,
- EA,
- mapeo de escenarios derivados a EA,
- FiE (ficha de Información Extemporánea),

- EF,
- mapeo de EA a EF,
- LEL_R y
- ERS.

Antes de analizar la representación del salto semántico existente para describir los procesos de negocio actual y aquellos que contendrán al nuevo sistema de software, es necesario detenerse en el concepto de evolución. Se puede decir que el pasaje del LEL al LEL_R se denomina “evolución” del léxico, pero existe una discrepancia en la bibliografía acerca de la utilización de dicho término. Para algunos autores el concepto está ceñido a las transformaciones que sufren los modelos durante todo el proceso de construcción del software, desde el análisis, pasando al diseño y así sucesivamente. Este es el caso de [Britman 04] donde se denomina evolución a las transformaciones de los escenarios y otros documentos durante todo el ciclo de vida del desarrollo de software. Otros, se refieren a la evolución cuando describen los cambios que sufre el dominio al planificar la inserción de un nuevo sistema de software. En este proyecto se adhiere a la última definición pero con una salvedad, los cambios del contexto sólo existen en los modelos del proceso de requisitos, ya que aún no han sido implementados en los procesos del negocio. A esta evolución se la denomina “evolución aparente” debido a que no se sabe aún si esos cambios permanecerán en el dominio. La evolución semántica o real sucederá recién cuando el sistema de software se encuentre en ejecución.

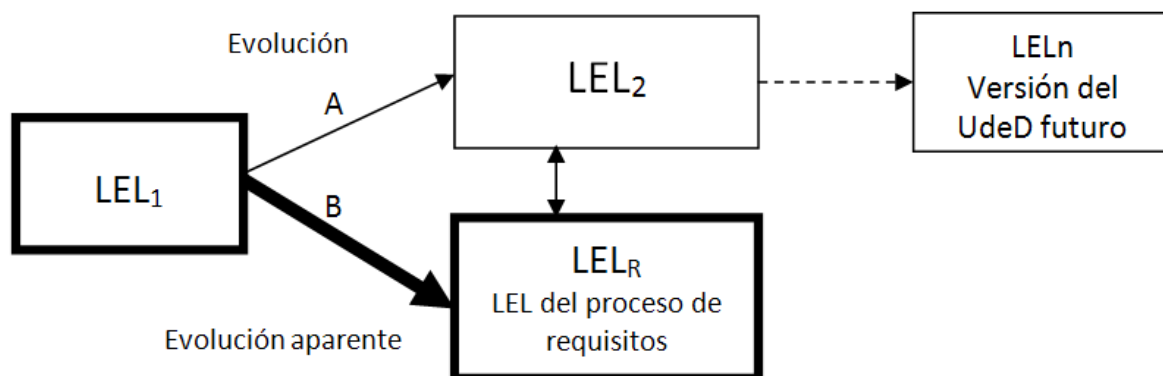


Figura 1 – Evolución vs. Evolución aparente

Como se observa en la Figura 1, el camino A presenta la evolución del léxico hasta que el mismo es incorporado al dominio de la aplicación, o sea que se incorpora al vocabulario utilizado por los usuarios durante la realización de sus tareas. Por otro lado, el camino B, es la evolución aparente que se utiliza en los procesos de requisitos, donde se incluyen términos “artificiales” con el objetivo de tener un soporte semántico que permita garantizar la comprensión de la planificación de los procesos del negocio y de los requisitos del software, induciendo una notable reducción de la ambigüedad en los modelos construidos. El análisis presentado en este informe se realizó desde el punto de vista de la evolución aparente. Esta se efectiviza en la creación de un nuevo glosario denominado LEL_R el cual se construye en un documento independiente al LEL original. Este nuevo glosario conserva todas las características estructurales del LEL pero modifica su objetivo central al permitir la inclusión de términos que no pertenecen al dominio. En este nuevo glosario se migran los términos que se han mantenido inalterables durante el proceso de requisitos, se modifican los términos que sufren cambios y se agregan los términos nuevos necesarios para describir los EF y la ERS. Se puso mucho énfasis en identificar el mejor momento para construir el nuevo glosario, llegando a



la conclusión de hacerlo una vez completos los EF, buscando un mecanismo que altere con el menor costo posible, el proceso existente.

La construcción del LEL_R debe cumplir con el principio de Vocabulario Mínimo y en parte con el principio de Circularidad propuestos en la construcción del LEL. Es aquí donde comienzan a aparecer algunas modificaciones entre ambos procesos. El principio de circularidad sufre algunos cambios ya que en este glosario se reduce el uso de los símbolos tipo Verbo. Los impactos que deben hacer referencia a un símbolo verbo lo hacen ahora a un escenario futuro (ver Figura 2). Por lo tanto, si existe un escenario futuro que responda a la acción que se desea describir en el impacto, se debe vincular a dicho escenario. De lo contrario, si no existe, se puede crear un símbolo Verbo pero se sugiere revisar previamente los escenarios futuros. Los impactos de un Verbo deben ser identificados escribiendo el título del escenario futuro en mayúscula. Con los Sujetos es diferente, ya que el uso de un escenario futuro como impacto puede estar representando acciones de diferentes actores, excediendo la información necesaria para ese sujeto. Se sugiere referenciar al escenario como en los otros casos, pero su lectura se debe reducir sólo a aquellos episodios del escenario donde el símbolo Sujeto en cuestión tiene un rol determinado, o sea que es un actor del episodio.

Símbolo del LEL	Símbolo del LEL _R
Recursos Humanos	Recursos Humanos
Noción	Noción
1. Es un grupo de personas que se encarga de seleccionar al personal	9. Es un grupo de personas que se encarga de seleccionar al personal
Impacto	Impacto
2. Solicita <u>referencias laborales</u>	10. <u>INGRESA DATOS DEL CANDIDATO</u>
3. Gestiona el <u>estudio ambiental</u>	11. <u>GESTIONA INGRESO</u>
4. Gestiona el <u>estudio preocupacional</u>	12. <u>REALIZA BUSQUEDA DE PREFERIDOS</u>
5. Gestiona el estudio psicotécnico	13. ...
6. <u>Determina el CV preferido</u>	
7. <u>Contacta a los candidatos</u>	
8. ...	

Figura 2 - Ejemplo de un símbolo Sujeto y de su Evolución *aparente*

La Figura 2 muestra un ejemplo de un símbolo Sujeto del LEL que se modificó al definir la Planificación el contexto futuro. En el símbolo del LEL se puede observar que los últimos dos impactos son símbolos verbos. En cambio los impactos del LEL_R son todos escenarios futuros.

El símbolo Sistema, propuesto por el proceso de construcción de escenarios futuros para identificar las acciones del nuevo sistema de software, no debe ser incorporado al LEL_R ya que casi todos sus impactos serán los propios requisitos del software. En los escenarios futuros se debe utilizar la palabra "Sistema" con el rol de un Sujeto pero sin identificarlo como un símbolo.

El uso de homónimos estaba restringido a la aparición de símbolos sintácticamente iguales con diferente semántica. Este uso es original del LEL y se mantiene en el LEL_R. Pero se debe tener cuidado porque el uso de homónimos se incrementa en la construcción del LEL_R cuando existe una modificación parcial o total de un símbolo que se mantiene durante todo el proceso de requisitos. Debido a que este glosario cuenta con símbolos artificiales y que se ha utilizado el concepto de homónimo para evidenciar la evolución aparente, se sugiere reducir tanto como se



pueda el uso homónimos entre nuevos símbolos. En caso de ocurrir, una posibilidad es buscar otro nombre para alguno de los símbolos en cuestión.

Es poco relevante tener una estadística profusa de los promedios de las cantidades de términos alterados o nuevos porque estos datos dependerán fuertemente del grado de innovación que se produzca en el proceso de negocio como consecuencia de la introducción de un nuevo sistema de software. Es mucho más importante tener la certeza que esos cambios siempre se van a producir y el haber desarrollado mecanismos para registrarlos y aprovecharlos en beneficio de un mejor desarrollo del sistema de software.

Como resultado final del análisis precedente, se ha generado una Heurística de Construcción del LEL_R la cual se describe a continuación:

Esta heurística se inicia generando sólo una lista de símbolos candidatos en paralelo con la primera versión de los escenarios futuros. Dicha lista se completa durante las entrevistas donde se negocian y se validan estos escenarios. La mayor actividad de la construcción del LEL_R se concentra una vez que los escenarios futuros están totalmente construidos. Es probable que durante la generación de la ERS aparezca nuevo vocabulario o se modifique alguna noción o impacto de un símbolo ya existente en el LEL_R. Se estima que estos cambios son de muy baja incidencia. A continuación se describe la heurística de construcción del LEL_R:

1. IDENTIFICAR SIMBOLOS

1.1. Los escenarios futuros utilizan, en primera instancia, al LEL como referencia léxica, pero se debe prestar una especial atención a la aparición de nuevos términos necesarios para describir los cambios en los procesos del negocio. Estos símbolos nuevos deben ser identificados (marcados en el escenario) de alguna manera que los diferencie de los símbolos del LEL.

1.2. Al describir un escenario futuro también puede ser percibido un mal uso o el cambio de significado de un símbolo del LEL, en este caso se puede identificar, por ejemplo, con una "(h)" al final del símbolo para indicar que se está en presencia de un homónimo.

1.3. CREAR LA LISTA DE SIMBOLOS CANDIDATOS FUTUROS

1.4. Durante las entrevistas para negociar y validar los escenarios futuros se debe generar una lista denominada Lista de Símbolos Candidatos Futuros, con las palabras o frases que el usuario utiliza para describir sus necesidades, los nuevos procesos o sus deseos y que no son parte del vocabulario utilizado hasta el momento, por lo tanto no se encuentran en el LEL.

1.5. Nuevamente, si durante las entrevistas se percibe un mal uso o el cambio de significado de un símbolo del LEL identificarlo como homónimo.

1.6. Clasificar los símbolos en Sujeto, Objeto, Verbo y Estado.

2. CREAR LEL_R

2.1. Al finalizar con la descripción de los escenarios futuros se debe crear el LEL_R e incorporarle los símbolos de la Lista de Símbolos Candidatos Futuros.

3. DESCRIBIR SIMBOLOS

3.1. Describir los símbolos del LEL_R retornando al contexto para completar su definición. Utilizar los patrones del LEL para los tipos Sujetos, Objetos, Verbos y Estados.

3.2. Recorrer cada escenario futuro:

3.2.1. Verificar el cubrimiento del LEL:

3.2.1.1. Por cada símbolo del LEL utilizado en el escenario futuro, ir a su definición original. Si el cubrimiento del LEL es del 100% copiarlo al LEL_R identificado como migrado, o sea que mantiene su significado durante la planificación del contexto futuro.

- 3.2.1.2. Si parte de la noción o del impacto no corresponde, crear un símbolo nuevo con la descripción correcta e identificarlo como homónimo.
- 3.2.2. Por cada símbolo identificado como nuevo:
- 3.2.2.1. Verificar si ya existe en el LEL_R. Si existe agregar el sinónimo. Si no existe, agregar el símbolo y no utilizar ningún identificador.
- 3.2.3. Por cada símbolo identificado como homónimo.
- 3.2.3.1. Verificar si ya existe en el LEL_R. Si existe agregar el sinónimo. Si no existe, agregar el símbolo e identificarlo como homónimo.
- 3.2.3.2. Se debe controlar si este nuevo símbolo reemplaza la definición original del LEL o es necesario tener ambos. En este último caso copiar el símbolo original en el LEL_R.

4. VERIFICAR LEL_R

- 4.1. Con el LEL_R completo utilizar el proceso de inspección del LEL [Kaplan 00] para verificar su consistencia interna. Este proceso deberá ser mínimamente adaptado. Se debe verificar también parte de su consistencia externa, controlando que todos los escenarios futuros utilizados existan en el conjunto de escenarios futuros.

La Figura 3 muestra un resumen de los pasos que se deben seguir para construir el LEL_R.

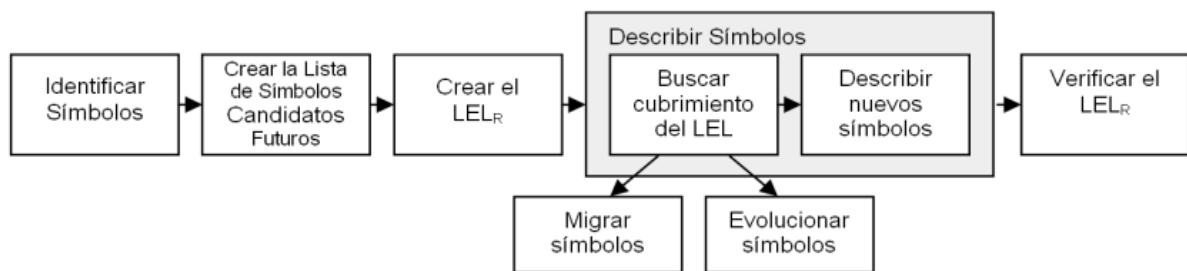


Figura 3 – Resumen de la heurística de construcción del LEL_R

Finalizando este análisis, se puede confirmar la hipótesis enunciada en el protocolo de presentación del proyecto que decía: “Se supone que existe una fuerte permeabilidad conceptual entre los clientes y los desarrolladores dado que los documentos que crean los desarrolladores motivan, por un lado, a los clientes a comprender la interacción entre el nuevo sistema de software y el proceso del negocio y, por el otro, a los desarrolladores en las problemáticas específicas de los clientes. Esto hace que se produzca una evolución en el vocabulario propio del contexto, la que debe ser seguida por los documentos que se construyan.”

4. Bibliografía:

REFERENCIAS

- [Alspaugh 99] Alspaugh, T. A., Antón, A. I., Barnes, T., Mott, B. W. (1999) “An Integrated Scenario Management Strategy”, International Symposium On Requirements Engineering (RE’99), Limerick-Ireland (IEEE Computer Society Press).
- [Britman 04] Brietman K, Liete J., Berry D. (2004) Supporting Software Evolution, Requirements Engineering, May 2005, Volume 10, Issue 2, pp 112-131
- [Doorn 02] Doorn J., Hadad G., Kaplan G. (2002) Comprendiendo el Universo de Discurso Futuro, WER’02 - Workshop on Requirements Engineering, Valencia, Spain.



- [Fishman 99] Fishman, J. (1999). Handbook of Language and Ethnic Identity. Oxford: Oxford University press.
- [Hadad 08] Hadad G.D.S., Doorn J.H., Kaplan G.N. (2008) Creating Software System Context Glossaries, In: Mehdi Khosrow-Pour (ed) Encyclopedia of Information Science and Technology. IGI Global, Information Science Reference, Hershey, PA, USA, ISBN: 978-1-60566-026-4, 2nd edn, Vol. II.
- [Kaplan 00] Kaplan, G.N., Hadad, G.D.S., Doorn, J.H., Leite, J.C.S.P., "Inspección del Léxico Extendido del Lenguaje", proceedings of WER'00 – III Workshop de Engenharia de Requisitos, Río de Janeiro, Brazil, 2000.
- [Leite 04] Leite J.C.S.P., Doorn J.H., Kaplan G.N., Hadad G.D.S., Ridaio M.N. (2004) Defining System Context using Scenarios, In: Leite J.C.S.P. and Doorn J.H (eds) Perspectives on Software Requirements, Kluwer Academic Publishers, ch. 8.
- [Neill 03] Colin J. Neill and Phillip A. Laplante (2003) Requirements Engineering: The State of the Practice, IEEE Software 20(6) pp. 40-45.
- [Nettle 00] Nettle, D., Romaine, S. (2000). Vanishing Voices. Oxford: Oxford University Press.
- [Ober 98] Ober R., Probasco L, Ericsson M. (1998) Applying Requirements Management with Use Cases. Rational Software Corporation.
- [Rolland 98] Rolland C., Ben Achour C. (1998) Guiding the construction of textual use case specifications, Data & Knowledge Engineering.
- [Robertson 06] Robertson S. and Robertson J. (2006) Mastering the Requirements Process, 2nd Ed, Addison-Wesley.
- [Weidenhaupt 98] Weidenhaupt, K., Pohl, K., Jarke, M., Haumer, P. (1998) "Scenarios in System Development: Current Practice", IEEE Software

BIBLIOGRAFIA COMPLEMENTARIA

1. Bell, T.E., Thayer, T. A., "Software Requirements: are they really a problem?", Second International Conference on Software Engineering, 1976.
2. Ben Achour, C., Rolland, C., Maiden, N.A.M., Souveyet, C., "Guiding Use Case Authoring: Results of an Empirical Study", International Symposium On Requirements Engineering (RE'99), Limerick-Ireland (IEEE Computer Society Press), 1999.
3. Booch, G., "Scenarios", Report on Object Analysis and Design, Vol.1, N° 3, 1994.
4. Breitman, K.K, Leite, J.C.S.P., "A Framework for Scenario Evolution", Proceedings of the IEEE International Conference on Requirements Engineering, IEEE Computer Society Press, 1998.
5. Breitman, K.K, Leite, J.C.S.P., "Proceso de Software basado en Cenários", Proceedings of WER'99, Workshop en Requerimientos, 28 JALIO, Buenos Aires, Argentine, 1999.
6. Bubenko, J. A., Wrangler, B., "Objectives driven capture of business rules and information systems requirements", IEEE Conference on Systems, Man and Cybernetics.
7. Carroll, J., "Introduction: The Scenario Perspective on System Development", Scenario-Based Design: Envisioning Work and Technology in System Development, J. Carroll, ed., John Wiley & Sons, New York, 1995.
8. Cockburn, A., "Structuring Use Cases with Goals", Technical Report, Humans and Technology, <http://members.aol.com/acockburn/papers/usecases.htm>, 1995.
9. Constantine, L., "Joint Essential Modeling, User Requirements Modeling for Usability", International Conference on Requirements Engineering, Colorado Springs, Constantine & Lockwood, Ltd., 1998.
10. Cysneiros L.M., y E. Yu, "Non-Functional Requirements Elicitation", Perspectives on Software Requirements, Kluwer Academic Publishers, Estados Unidos, capítulo 6, 2004.
11. Dardenne, A., van Lamsweerde, A., Fickas, S., "Goal Directed Requirements Acquisition", Science of computer Programming, Vol. 20, 1993.
12. Davis, A., "Software Requirements: Objects, Functions and States", Prentice Hall, Englewood Cliffs, NJ, 1993.



13. Desharnais, J., Frappier, M., Khedri, R., Mili, A. "Integration of Sequential Scenarios", IEEE Transactions on Software Engineering, Vo.24, N° 9, 1998.
14. Doorn, J., Kaplan, G., Hadad, G., Leite, J.C.S.P., "Inspección de Escenarios", Proceedings of WER'98, Workshop en Engenharia do Requisitos, Maringá, Brazil, 1998.
15. Egyed, y P.Grünbacher, "Identifying requirements conflicts and cooperation: How quality attributes and automated traceability can help", IEEE Software, 2004.
16. Elkoutbi M, Khriess I, Keller R. , "Generation User Interface Prototypes from Scenarios", IEEE Software, 1999.
17. Feblowitz, M.D., Greenspan, S.J.: Scenario-Based Analysis of COTS Acquisition Impacts, Requirements Engineering Journal, Vol.3, N° 3 & 4, 1998.
18. 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, Rio de Janeiro, 1992.
19. Gause, D.C., Weinberg, G.M., "Exploring Requirements. Quality Before Design", Dorset House Publishing, New York, 1989.
20. Goguen J., Linde C., "Techniques for Requirements Elicitation", IEEE International Symposium on Requirements Engineering , 1992
21. 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.
22. 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 (RITA), Brazil, Vol.6, N° 1, 2000.
23. Hadad, G., Kaplan, G., Oliveros, A., Leite, J.C.S.P., "Léxico Extendido del Lenguaje y Escenarios del Meeting Scheduler", Technical Report # 13, Departamento de Investigación, Universidad de Belgrano, Buenos Aires, 1998.
24. Hadad, G.D.S., Doorn, J.H., Kaplan, G.N., Leite, J.C.S.P. "Enfoque middle-out en la Construcción e Integración de Escenarios", Anales del Wer'99, Argentina, 1999,.
25. Hadad, G.D.S., Kaplan, G.N., Oliveros, A., Leite, J.C.S.P.: Construcción de Escenarios a partir del Léxico extendido del Lenguaje, 26 JAIIO, Buenos Aires, 1997.
26. Hsia, P., Samuel, J., Gao, J., Kung, D., Toyosima, I. and Chen, C., "Formal Approach to Scenario Analysis", IEEE Software, 1994, V.11, N° 2.
27. Jackson, M., "Software Requirements & Specifications. A lexicon of practice, principles and prejudices", Addison Wesley, ACM Press, 1995.
28. 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.
29. Jarke, M., Tung Bui , X., Carroll, J., "Scenario Management: An Interdisciplinary Approach", Requirements Engineering Journal, Vol.3, N° 3 y 4, 1998.
30. Kaindl, H., "A Design Process Based on a Model Combining Scenarios with Goals and Functions", IEEE Transactions on Systems, Man and Cybernetic, Vol. 30, N° 5, September 2000.
31. Kaplan G.N., Doorn J.H., Guatelli R., Gigante N., Hadad G.D.S., Storyboard Basados en Escenarios Futuros, anales de WICC 2009 – XI Workshop de Investigadores en Ciencias de la Computación – UNSJ - Pcia. San Juan, Argentina, ISBN 978-950-605-570-7, 2009.
32. Kaplan G.N., Doorn J.H., Hadad G.D.S., Handling Extemporaneous Information in Requirements Engineering, Encyclopedia of Information Science and Technology, editor: Mehdi Khosrow- Pour, D.B.A., Information Science Reference, EEUU, ISBN: 978-1-60566-026-4, 2º edición,2008 .
33. Kotonya G, y I. Sommerville, "A Framework for Integrating Functional and Non-Functional Requirements", IEEE International Workshop on Systems Engineering for Real Time Applications, Cirencester, UK, 1993.
34. Lehman, M.M., "Programs, Life Cycles, and Laws of Software Evolution", in Proceedings of the IEEE 68(9), pp. 1060-1076, 1980.
35. Leite, J.C.S.P., "Eliciting Requirements Using a Natural Language Based Approach: The Case of the Meeting Scheduler", Monografias em Ciência da Computacao, Departamento de Informática, PUC-Rio, 1993.



36. Leite, J.C.S.P., "Application Languages: A Product of Requirements Analysis", Computer Science Department of PUC-Rio, Brazil, 1989.
37. 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.
38. Leite, J.C.S.P., Franco, A.P.M., "O Uso de Hipertexto na Elicitação de Linguagens da Aplicação", Anais de IV Simpósio Brasileiro de Engenharia de Software, SBC, 1990.
39. 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.
40. 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.
41. 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.
42. Logrippo D, Buhr R, Gray T, "Use Case Maps for the Capture and Validation of Distributed Systems Requirements" , RE'99: Fourth IEEE International Symposium on Requirements Engineering.
43. Loucopoulos, P., Karakostas, V., "System Requirements Engineering", McGraw-Hill, London, 1995.
44. Michael Jackson , "Software Requirements & Specification, Addison Wesley, 1995
45. Pinheiro F.A.C., "Requirements Traceability", Perspectives on Software Requirements, Kluwer Academic Publishers, Estados Unidos, ISBN: 1-4020-7625-8, capítulo 5, 2004.
46. 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.
47. Potts, C., Takahashi, K., Antón, A.I., "Inquiry-Based Requirements Analysis", IEEE Software, Vol. 11, n.2, Mar. 1994.
48. Pressman, R, "Ingeniería de Software", Sexta Edición, McGraw Hill, 2006
49. Reubenstein, H. B., Waters, R. C., "The requirements apprentice: Automated assistance for requirements acquisition", IEEE Transaction on Software Engineering, Vol.17, No.3 (1991).
50. 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.
51. Ridao, M., Doorn, J., Leite, J.C.S.P., "Incorporación de patrones al proceso de construcción de Escenarios", aceptado para publicación en WER 2001, Buenos Aires, Argentina, 2001.
52. Rolland, C., Ben Achour, C., "Guiding the construction of textual use case specifications", Data & Knowledge Engineering 25, 1998.
53. Rolland, C., Souveyet, C, Ben Achour, C., "Guiding Goal Modeling Using Scenarios", IEEE Transactions on Software Engineering, Vol. 24, N° 12, 1998.
54. Sawyer P, Kotonya G., "Software Requirements", Chapter 2, IEEE May.
55. Sommerville, I., "Software Engineering", Reading, MA: Addison Wesley, 6th Edition, 2001.
56. Sutcliffe, A., Fickas, S., & Sohlberg, M. (2006). PC-RE: A method for personal and contextual requirements engineering with some experience. Requirements Engineering Journal.
57. Sutcliffe, A.G., Maiden, N.A.M., Minocha, S., Manuel, D., "Supporting Scenario-Based Requirements Engineering", IEEE Transactions on Software Engineering, Vol. 24, N° 12, 1998.
58. Uchitel, S., Chatley, R., Kramer, J., & Magee, J., Goal and scenario validation: A fluent combination. Requirement Engineering Journal, 2006.
59. Zanlorenci, E.P., Burnett, R.C., "Modelo para Qualificação da Fonte de Informação do Cliente e de Requisito Funcional", Proceedings of WER'98, Workshop em Engenharia do Requisitos, Maringá, Brazil, 1998.
60. Zorman, L., "Requirements Envisaging by Utilizing Scenarios (Rebus)", Ph.D. Dissertation, University of Southern California, 1995.



Producción científico-tecnológica

5.1 Publicaciones

a) Tesis de Posgrado

Autora: Lic. Gladys Noemí Kaplan

Director: Ing. Jorge Horacio Doorn

Defendida y aprobada: 25 de septiembre de 2014

Maestría en Informática - UNLaM

Título de la tesis: "Información Extemporánea en los Procesos de Requisitos"

b) Capítulos de libro

- Gladys Kaplan, Jorge Doorn. "Advanced & Delayed Information in Requirements Engineering". "Encyclopedia of Information Science & Technology". Third Ed. - Volumnes 10. Páginas Totales: 10384. Editores: IGI Global Books (Mehdi Khosrow-Pour, Information Resources Management Association, USA). Aceptado para su publicación prevista para 2015.

c) Artículos

- Gladys Kaplan, Jorge Doorn, Nora Gigante, "Evolución Semántica de los Glosarios en los Procesos de Requisitos". XIX Congreso Argentino de Ciencia de la Computación - CACIC2013 – X Workshop Ingeniería de Software (WIS) - Mar del Plata. 2013.
- Gladys Kaplan, Jorge Doorn, Nora Gigante. "Elicitación de Requisitos guiada por los modelos",. XX Congreso Argentino de Ciencia de la Computación - CACIC2014 – X Workshop Ingeniería de Software (WIS) – UNLaM – San Justo 2014.
- Gladys Kaplan, Jorge Doorn, Nora Gigante, "Construcción del LEL de Requisitos", anales de WICC 2013 – XV Workshop de Investigadores en Ciencias de la Computación, Paraná.
- Renata Silvia Guatelli y Verónica Inés Aubin."Aprender entre pares a través de e-group en la universidad. Análisis de un caso", 1er Congresos Nacional de Ingeniería Informática / sistemas de Información - CoNalISI 2013. Universidad Tecnológica Nacional – Facultad Regional Córdoba. 21 y 22 de noviembre del 2013.

Se presentó un poster:

- Gladys Kaplan, Jorge Doorn, Nora Gigante. Poster "*Construcción del LEL de Requisitos*", WICC 2013 – XV Workshop de Investigadores en Ciencias de la Computación, Paraná. 2013.

d) Congresos Internacionales, Nacionales, Simposios, Jornadas, otros

- Gladys Noemí Kaplan, Jorge Horacio Doorn y Nora Cristina Gigante. "Elicitacion de Conocimiento Guiada por Modelos"- XX Congreso Argentino de Ciencia de la Computación - CACIC2014 – UNLaM – San Justo – Octubre de 2014. Articulo completo.
- Kaplan, Gladys, Doorn, Jorge, Gigante, Nora. "Evolución Semántica de los Glosarios en los Procesos de Requisitos". XIX Congreso Argentino de Ciencia de la Computación -



CACIC2013 – X Workshop Ingeniería de Software (WIS) - CAECE Mar del Plata, del 21 al 25 de Octubre de 2013. Artículo completo

- Gladys Kaplan, Jorge Doorn, Nora Gigante. "Construcción del LEL de Requisitos". anales de WICC 2013 – XV Workshop de Investigadores en Ciencias de la Computación, Paraná - 2013. Artículo completo
- Doorn Jorge, Kaplan Gladys, Hadad Graciela, Gigante Nora, Guatelli Renata, Andrea Vera, Claudia Litvak. "Estabilidad de los Documentos en el Proceso de Requisitos", 1ra. Jornada de Investigación Departamental – UNLaM. 2014
- Doorn Jorge, Kaplan Gladys, Gigante Nora, Guatelli Renata. "Contextualización del Proceso de Requisitos", 1ra. Jornada de Investigación Departamental – UNLaM. 2014

5.2) Otras Actividades académicas:

- Charla "Elicitación en el contexto de los Procesos de Requisitos" de Kaplan Gladys en la Escuela Superior de Informática de la Prefectura Naval Argentina en la Cuarta Jornada de Actualización Profesional en Informática realizada el 30 de Octubre de 2014.
- Participación de la Mg. Nora Gigante en la Organización del CACIC 2014 realizado en UNLaM en Octubre de 2014.
- Participación en concursos docentes

Gladys Noemí Kaplan
UNICEN - Universidad Nacional del Centro de la Provincia de Buenos Aires
Profesor Adjunto con dedicación Semiexclusiva
Facultad de Ciencias Humanas
Cátedra: Informática Aplicada
Tandil – Pcia. Bs. As. – 19 de junio de 2013

- Participación jurado de tesis de maestría

Nora Gigante
Jurado de Tesis de la Maestría en Informática "Información extemporánea en los procesos de requisitos"

Presidente de junto a Dr. Leandro Antonelli, Mag. Mario Oloriz, Mag. Nora Gigante. Sept de 2014

- Evaluación de artículos:

Gladys Noemí Kaplan
XVII Workshop de Ingeniería de Requisitos.
Realizado del 23 al 25 de Abril de 2014
Pucón - Chile

Gladys Noemí Kaplan
XVI Workshop de Ingeniería de Requisitos.
Realizado del 8 al 10 de Abril de 2013
Universidad ORT Uruguay – Montevideo – Uruguay.



Nora Gigante
ExpoProyectos UNLaM.

- Participación en la 1ra. Jornadas de Investigación Interdepartamental en UNLaM

Gladys Noemí Kaplan
Coordinadora de Mesa 18 - “Desarrollos en Ingeniería (Electrónica, Industrial y Civil)”
San Justo, 15 de Septiembre de 2014