

Aplicación Reflexiva de un Proceso de Requisitos

Gladys N. Kaplan¹, Renata S. Guatelli¹, Jorge H. Doorn^{1,2}, Andrea F. Vera¹,
María L. Pepe¹

¹Departamento de Ingeniería e Investigaciones Tecnológicas, Universidad Nacional de la Matanza

²Escuela de Informática, Universidad Nacional del Oeste

gkaplan@unlam.edu.ar, jdoorn@exa.unicen.edu.ar,
{renata.guatelli, andreafabianavera, laurapepe}@gmail.com

RESUMEN

En informática es una práctica habitual aplicar en forma reflexiva los métodos y estrategias que están siendo desarrollados al propio proceso de desarrollo. Ya en las etapas fundacionales de esta disciplina se hizo presente la reflexividad, por ejemplo en las primeras versiones del lenguaje LISP. Sin embargo en la Ingeniería de Requisitos hay pocos ejemplos del uso de esta estrategia. El presente proyecto se enmarca en un proceso de requisitos al cual se le han realizado mejoras en forma continua, pese a lo cual el mismo es aún susceptible de ser perfeccionado. La propuesta consiste en la aplicación en forma reflexiva de los modelos del proceso al proceso mismo de construcción de esos modelos. Se ha concentrado la atención en buscar puntos y mecanismos de creación de rastros en forma automática, en definir los requisitos que debe atender una herramienta que automatice parcialmente el Proceso de Requisitos y en buscar debilidades en el mismo. Se planifica comparar estos requisitos con los servicios provistos por herramientas construidas que no han seguido este proceso.

Palabras clave: Ingeniería de Software, Ingeniería de Requisitos, Proceso de Requisitos, Lenguaje Natural.

CONTEXTO

La línea de investigación que se presenta es parte de los proyectos de investigación “Reflexividad como herramienta en la Ingeniería de Requisitos” de la Universidad

Nacional de La Matanza (UNLaM) y “Tratamiento de los Factores Situacionales y la Completitud en la Ingeniería de Requisitos” de la Universidad Nacional del Oeste (UNO).

1. INTRODUCCIÓN

Tanto en la gestión del proceso de requisitos como en la gestión de los requisitos en sí mismos, la construcción y uso de técnicas de rastreabilidad es un aspecto no atendido o pobremente abordado tanto en los estudios académicos como en la práctica cotidiana. En gran parte esto es debido a la poca disponibilidad de herramientas automáticas o semiautomáticas para identificar y mantener trazas [4] [5] [6] [7].

La recolección y mantenimiento de la información de rastreo es de muy alto costo. Por lo tanto, se deben tener políticas que indiquen qué tipo de rastreos se realizarán y cómo se mantendrá dicha información.

Por otra parte, las herramientas desarrolladas para la estrategia utilizada como base en el presente proyecto, han sido construidas utilizando un proceso de desarrollo ad-hoc, que ha desconocido parcialmente los principios del proceso al que aspiran a servir. Esta es una realidad que brinda un espacio de comparación muy importante ya que permite detectar las potenciales desviaciones entre los requisitos detectados mediante el proceso y los servicios que prestan los sistemas construidos.

Finalmente, las mejoras introducidas al proceso de requisitos, a lo largo del tiempo, parecen estar acercándose asintóticamente a

un valor límite que no ofrece garantías. Posiblemente esta evolución se deba a una falta de crítica acerca de los mecanismos básicos involucrados. Una de las formas que permite visualizar esta falta de crítica consiste en releer las heurísticas de construcción de los diferentes modelos del proceso. En casi todas ellas se puede notar que las mismas son más cercanas a una descripción de los componentes del modelo que a una guía meditada que ayude a afrontar el trabajo en forma eficaz.

2. LÍNEAS DE INVESTIGACIÓN Y DESARROLLO

A lo largo de dos décadas en varios proyectos de investigación, se ha desarrollado una estrategia de Ingeniería de Requisitos basada en modelos en lenguaje natural [8], que ha sido difundida en cursos de grado y posgrado, y puesta en práctica en diversos proyectos de software de mediana a gran envergadura en la industria. Aun cuando esta estrategia puede considerarse suficientemente madura, todavía es concebible que la misma pueda ser mejorada en varios aspectos.

Por un lado los costos en un proyecto de software originados por una inapropiada gestión de requisitos, especialmente en lo que se refiere a la inadecuada atención de los cambios en los mismos, son de una importancia grande o muy grande dependiendo del grado de volatilidad de los mismos.

Casi sin excepción, un tratamiento inadecuado de los cambios en los requisitos deviene de la carencia de registros que indiquen en forma precisa qué aspecto del proceso del negocio originó un determinado requisito y como consecuencia qué requisitos relacionados serán también afectados.

Debe notarse que los cambios en los requisitos no surgen como referencias explícitas y claras a determinadas funcionalidades del sistema de software, sino como problemáticas del dominio del problema usualmente relacionadas en forma poco clara con los requisitos del sistema en desarrollo.

Otro aspecto relevante es que en muchos

casos se consideran cambios en los requisitos a meros errores de interpretación u omisiones en las primeras fases del proyecto.

Por otro lado, la imposibilidad efectiva de crear modelos completos y el fantasma de la existencia de omisiones graves está siempre presente en todo proceso de requisitos. En estudios previos se ha comprobado la importancia de las omisiones, al extremo de carecerse de estimaciones de la completitud lograda en el proceso [9] [10] [11] [12].

Es muy tentador y numerosos autores reportan que incrementa la probabilidad de éxito de un proyecto de software, el estudio del dominio del problema y luego que los requisitos de ese sistema se plasmen en descripciones o modelos creados en Lenguaje Natural [13]. Las mejoras registradas en la calidad de los requisitos tienen su fundamento en que las representaciones en lenguaje natural, tales como glosarios, casos de uso y escenarios, promueven la comunicación entre todos los involucrados en un proyecto de desarrollo de software y facilitan la validación de los requisitos elaborados. Una revisión hecha por Rolland et al. [14] muestra que, de doce enfoques propuestos en la literatura en el ámbito de la Ingeniería de Requisitos, todos ellos usan una notación de texto para describir escenarios, que en algunos casos se combinan con otros medios, tales como gráficos o imágenes. En un estudio relativamente reciente sobre la práctica en Ingeniería de Requisitos [15], se concluyó que el 51% de las organizaciones (sobre un total de 194) usa representaciones informales (por ejemplo, el lenguaje natural) y el 27% semi-formales, quedando los modelos formales relegados a un uso de apenas el 7% de las organizaciones.

Este proyecto está enmarcado en una estrategia en Ingeniería de Requisitos orientada al cliente, pues efectivamente se basa en la construcción de modelos en lenguaje natural [12].

Si bien la estrategia avanza construyendo modelos, su fin no es producir modelos sino alcanzar una profunda comprensión del dominio de la aplicación para definir una solución óptima a través de un sistema de software. Es decir, la estrategia presenta dos

grandes etapas bien distinguibles: una de aprendizaje y la otra de definición. Cuando hay un conocimiento previo del dominio de la aplicación la primera fase se convierte en un proceso confirmatorio.

Los modelos que se utilizan en esta estrategia son:

- 1) un modelo léxico, LEL (Léxico Extendido del Lenguaje) [16], el cual describe el vocabulario utilizado en el dominio de la aplicación,
- 2) un modelo organizacional que describe los procesos actuales del negocio utilizando [17], denominado Escenarios Actuales,
- 3) un modelo organizacional que describe los procesos del negocio proyectados en base al sistema de software a desarrollar [18] [19], denominado Escenarios Futuros.

Los Escenarios Futuros tienen empotrados los requisitos del software. El documento de especificación de requisitos de software, ERS, se puede obtener a partir de estos Escenarios Futuros. El fin de la estrategia utilizada es alcanzar una profunda comprensión del dominio de la aplicación para definir una solución óptima para los clientes/usuarios, a través de un sistema de software.

Desde el punto de vista epistemológico el presente proyecto se puede catalogar como una investigación esencialmente exploratoria, en la que se aspira a detectar: i) los principales factores que contribuyen a definir los mecanismos que permitirían construir rastros entre los diferentes modelos del proceso de requisitos, ii) los requisitos de una herramienta de apoyo al proceso de requisitos y iii) evidencias sobre las debilidades del proceso. Es de esperar que estos resultados puedan luego ser contrastados en investigaciones confirmatorias y que además permitan establecer las bases que permitan diseñar una o más herramientas para facilitar la tarea de construcción y manipulación de los rastros entre modelos.

Si bien se ha procurado y se procurará mantener la máxima generalidad posible, los resultados del proyecto estarán afectados por la estrategia utilizada, consistente en el estudio de un caso. Estas limitaciones están atemperadas por la notoria amplitud del caso y por la continua revisión de los resultados obtenidos, en el marco de la biblioteca de casos estudiados previamente, la que supera largamente la cantidad de 100.

3. RESULTADOS OBTENIDOS/ESPERADOS

Con los trabajos preliminares que sirven de antecedentes al presente proyecto se han podido detectar algunas regularidades que posiblemente se transformen en conclusiones. Estas conclusiones permitirán elaborar hipótesis a ser consideradas en futuros proyectos, los que tendrán carácter de confirmatorios.

Concretamente, hasta el momento se ha descubierto que:

- 1) El vocabulario utilizado en el Universo de Discurso contiene una cantidad importante de símbolos compuestos por frases construidas por sustantivos compuestos, sustantivos adjetivados y frases verbales entre otros. Usualmente la cantidad de estos símbolos supera ampliamente la cantidad de símbolos constituidos por palabras aisladas.
- 2) Las frases verbales suelen estar asociadas semánticamente con objetos. Sin embargo, la forma resultante de estas frases verbales hace que su uso sea prácticamente antagónica con la combinación en una sola sentencia de la frase verbal y el objeto. En otras palabras, una oración suele ser perfectamente no ambigua en su contexto, pero omitir uno de los dos (el verbo o el objeto) o empotrar el objeto en el verbo. Este fenómeno hace que un ingeniero de requisitos, que no ha recibido indicaciones precisas al respecto, tienda a omitir

uno de los dos símbolos. Es importante tener en claro que ninguna de las heurísticas utilizadas hasta el momento mencionan este fenómeno. En un recuento sin validez estadística se ha encontrado que aproximadamente se omiten entre el 30 y el 40% de los símbolos involucrados en esta relación (verbos y objetos). Usualmente se registran más los objetos que los verbos, en una relación groseramente aproximada de 4 a 1. Claramente la existencia de estas omisiones hará que algunos aspectos de las características del Universo de Discurso no se perciban y esto hará que se omitan Escenarios Actuales y posiblemente también Escenarios Futuros. Se desconoce la importancia de estas omisiones las que posiblemente sean atemperadas, en alguna medida, a lo largo de todo el proceso, ya que se consulta el Universo de Discurso en varias actividades del proceso de requisitos, especialmente durante la validación de los diferentes modelos.

- 3) Parece ser que los símbolos compuestos provenientes de la concatenación de sustantivos también ocultan otros símbolos, especialmente aquellos pertenecientes a jerarquías taxonómicas. Sin embargo aún no se ha tenido oportunidad de tener una apreciación más concreta del fenómeno.
- 4) Se ha verificado, también sin comprobación cuantitativa, que la creación automática o semi-automática de rastros se verá facilitada si se procesa todo el conocimiento elicitado de una misma fuente de información ‘en bloque’, significando con esto que sería conveniente agotar el registro del conocimiento capturado de una misma fuente de información, antes de proceder a considerar otra fuente

de información. En esta última afirmación se está haciendo un uso abusivo de ‘fuente de información’ ya que se pretende incluir también la información que se extrae de un modelo para construir otro. En particular en lo que se refiere al LEL:

- a) Ocurre que aunque más de una fuente de información contribuya a completar la información correspondiente a un determinado ítem, esto hace que sea preferible que la gestión de los incrementos se produzca en cada símbolo del LEL y no en las fuentes de información. Esto haría que se incremente la importancia de la actividad de verificación.
- b) Una conclusión aparentemente muy importante surge del hecho que se estima que será necesario modificar la forma de construir el LEL, en virtud que la actividad ‘creación de la lista de símbolos candidatos’ recomendada en la heurística actual es contraproducente, tanto para el registro de rastros como para la reducción de las omisiones.

Todos estos resultados preliminares son sin lugar a duda valiosos porque de confirmarse permitirán elaborar heurísticas más apropiadas, pero su mera observación crítica hace evidente que los mismos constituyen un conjunto semánticamente poco conexo de patrones. Se espera que estos patrones se constituyan en una etapa intermedia en la detección de reglas más abarcativas. Este es el núcleo del presente proyecto.

4. FORMACIÓN DE RECURSOS HUMANOS

En el proyecto de la UNLaM participan cuatro investigadores. En el proyecto de la UNO participan en este tema otros dos investigadores, uno de ellos en formación.

La línea de investigación presentada aquí

es parte directa de las tesis de maestría de la Lic. Renata Guatelli y de las tesis de doctorado de la Mg. Gladys Kaplan y la Ing. Andrea Vera, y colabora con la tesis de doctorado de la Lic. María Pepe.

5. BIBLIOGRAFÍA

- [1] Gotel, O., C., Z., Finkelstein, ACW (1994) An analysis of the requirements traceability problema, ICRE'94. First IEEE International Conference on Requirements Engineering. IEEE Computer Society Press, Colorado Springs, pp. 94-101.
- [2] Palmer J., D., (1997) Traceability. Software Engineering, M. Dorfman y R.H. Thayer Editores. IEEE Computer Society Press. 1996. Pág. 266 a 276. Reimpreso en: Software Requirements Engineering. R.H. Thayer y M. Dorfman Editores. IEEE Computer Society Press. 2º Edición. Los Alamitos, CA. pp. 364-374.
- [3] Pinheiro F., A., C., (2004) Requirements Traceability. Perspectives on Software Requirements. Kluwer Academic Publishers. Capítulo 5, pp. 91-113.
- [4] Cleland-Huang, J., Chang, C., K., Christensen, M., (2013) Event-based traceability for managing evolutionary change. IEEE Trans. Softw. Eng. Vol. 29. N° 9, pp. 796-810.
- [5] De Lucia, A., Fasano, F., Oliveto, R., Tortora, G., (2007) Recovering Traceability Links in Software Artifact Management Systems using Information Retrieval Methods, ACM Transactions on Software Engineering and Methodology. Vol.16. N° 4, pp. 13.1-13.50.
- [6] Pruski, P, Lohar, S., Goss, W., et al (2015) TiQi: answering unstructured natural language trace queries, Requirements Engineering, Vol. 20, Nro. 215, pp. 215-232.
- [7] Mahmoud, A., Niu, N., (2015) On the role of semantics in automated requirements tracing. Requirements Engineering, Vol. 20, Nro. 281, pp. 281-300.
- [8] Leite, J., C., S., P., Doorn, J. H., Kaplan, G., Hadad, G., D., S., Ridao, M., N., (2004) Defining System Context using Scenarios, Perspectives on Software Requirements, Kluwer Academic Publishers, cap.8, pp. 169-199.
- [9] Doorn, J., H., Ridao, M., N., (2003) Completitud en Glosarios: Un estudio experimental, en WER'03, International Workshop on Requirements Engineering, pp. 317-328.
- [10] Ridao, M., N., Doorn, J., H., (2006) Estimación de Completitud en Modelos de Requisitos Basados en Lenguaje Natural, WER'06, Internacional Workshop on Requirements Engineering, pp. 146-157.
- [11] Doorn, J., H., Ridao, M., N., (2008) Completeness Concerns in Requirement Engineering”, en “Encyclopedia of Information Science and Technology, Second Edition”.
- [12] Hadad, D., S., Litvak, C., S., Doorn, J., H., Ridao, M., N., (2015) “Dealing with Completeness in Requirements Engineering”, en “Encyclopedia of Information Science and Technology, Third Edition”.
- [13] Loucopoulos, P., Karakostas, V. (1995) System Requirements Engineering. McGraw-Hill, Londres.
- [14] Rolland, C., Souveyet, C., Ben Achour, C., (1998) Guiding Goal Modeling Using Scenarios, IEEE TSE, Vol. 24, Nro. 12, pp. 1055-1071.
- [15] Neill, CJ., Laplante, PA. (2003) Requirements Engineering: The State of the Practice, IEEE Software, Vol. 20, Nro. 6, pp. 40-45.
- [16] Leite, JCSP, Franco, APM, (1993) A Strategy for Conceptual Model Acquisition, IEEE First International Symposium on Requirements Engineering, RE'93, IEEE Computer Society Press, pp. 243-246, Los Alamitos, EEUU.
- [17] Leite, JCSP, Hadad, GDS, Doorn, JH, Kaplan, GN (2000) A Scenario Construction Process. Requirements Engineering Journal. Vol.5, N° 1, pp. 38-61.
- [18] Hadad GDS (2008) Uso de Escenarios en la Derivación de Software. Tesis Doctoral, Facultad de Ciencias Exactas, Universidad Nacional de La Plata. Argentina. 2008.
- [19] Hadad, GDS, Doorn, JH, Ridao, MN, Kaplan GN (2009) Facilitando la asignación de Prioridades a los Requisitos, WER'09 – 12th Workshop on Requirements Engineering, ISBN: 978-956-319-941-3, Universidad Técnica Federico Santa María, pp. 75-84. Valparaíso. Chile.