

Desagregación de la información en los escenarios futuros

Gladys Kaplan¹, David Tua², Gabriel Blanco¹

¹Departamento de Ingeniería e Investigaciones Tecnológicas. Universidad Nacional de La Matanza. Argentina

²Programa de Maestría en Ingeniería en Sistemas de Información. Facultad Regional Buenos Aires. Universidad Tecnológica Nacional. Argentina
gkaplan@unlam.edu.ar, dtua@frba.utn.edu.ar

RESUMEN

Los escenarios, como la mayoría de los modelos de la Ingeniería de Requisitos, utilizan el lenguaje natural para asegurar una buena comunicación con los clientes-usuarios. El proceso de construcción de escenarios aprovecha las ventajas del lenguaje natural y genera mecanismos para controlar algunas de sus debilidades. Estas últimas son mitigadas con el uso de un glosario, la sintaxis, entre otras. Una vez completos los escenarios, se vuelven a controlar en la Actividad Verificar. Si bien la verificación es exhaustiva, omite analizar la riqueza semántica de las expresiones, las que pueden esconder conocimiento relevante del contexto. En el presente proyecto de investigación se analiza la granularidad de las expresiones utilizadas fundamentalmente en la descripción de los episodios, las restricciones y las excepciones, donde puede existir soterramiento de información que debe ser desagregada. Se ha comprobado empíricamente que los diferentes niveles de detalle utilizados en los escenarios perjudican la comprensión de la propuesta de solución y es una de las causas de omisiones o malos entendidos en los requisitos del software.

Palabras clave: Ingeniería de Requisitos, Escenarios, Lenguaje Natural, Completitud.

CONTEXTO

La línea de investigación que se presenta se encuadra dentro del proyecto de investigación “Aspectos no funcionales en los procesos de requisitos” de la Universidad Nacional de La Matanza (UNLaM), del cual se desprende una tesis de Maestría de la

Escuela de Posgrado de la Universidad Tecnológica Nacional Facultad Regional Buenos Aires (UTN-FRBA).

1. INTRODUCCIÓN

La especificación de requisitos de software (ERS) debe ser, entre otros atributos, lo más completa posible, tanto en términos de negociación contractual como de documentación técnica. Alcanzar este objetivo es necesario para asegurar la construcción de un sistema de software que satisfaga las necesidades del cliente. Como es conocido, el costo de corrección de errores y omisiones varía de acuerdo al momento en el cual se lo detecta [1] [2], por lo tanto, es de suma importancia asegurar la calidad de los requisitos del software lo más temprano posible.

El proceso de requisitos basado en escenarios [3], en el que se basa la presente investigación, ha avanzado de diferentes maneras en algunas problemáticas asociadas al uso del lenguaje natural (LN) [4]. En el proceso de requisitos se utilizan básicamente dos modelos, el Léxico Extendido del Lenguaje (LEL) [5] [6] y los escenarios [7]. La construcción de un glosario como primera actividad en el proceso es fundamental para asegurar la comunicación. El LEL contiene el vocabulario propio del dominio de la aplicación y es utilizado en todas las descripciones para mejorar la precisión y disminuir la ambigüedad [8].

Los escenarios son narrativas en LN con un mínimo de formalismo, representan las situaciones del contexto. Como ya se mencionó, tanto la sintaxis como la utilización de un glosario mejoran la calidad de estas descripciones.

La diferencia sustancial entre los escenarios actuales y los futuros es su punto de vista [9]. La semejanza o disparidad entre un conjunto de escenarios y el otro depende de la cantidad de cambios que se espera en el contexto futuro. A menor cantidad de cambios, por ejemplo una informatización, existe una gran similitud entre ambos conjuntos de escenarios. A medida que se incorporan cambios en los procesos del negocio, por ejemplo, en una reingeniería de procesos del negocio, las diferencias son cada vez mayores.

<p>Título : ASIGNAR FECHA DE ANÁLISIS</p> <p>Objetivo: definir el momento de analizar</p> <p>Contexto:</p> <p>Ubicación Geográfica: Oficina técnica</p> <p>Ubicación Temporal:</p> <p>Precondición: debe haber un insumo con etiqueta amarilla</p> <p>Recursos: cronograma de trabajo 1, insumo, fecha de análisis</p> <p>Actores: jefe de control de calidad, Sistema de Software</p> <p>Episodios:</p> <ol style="list-style-type: none"> 1. el jefe de control de calidad ingresa fecha probable de análisis 2. el Sistema de Software busca los insumos sin fecha de análisis 3. #el Sistema de Software muestra el cronograma de trabajo 1 de la fecha ingresada y los insumos ordenados por prioridad de análisis 4. el Sistema de Software muestra la disponibilidad de cada técnico # 5. Si no hay disponibilidad de técnicos en esa fecha determinada entonces REASIGNAR FECHA DE ANALISIS 6. el jefe de control de calidad determina la fecha de análisis 7. ingresa en el cronograma de trabajo 1 el insumo y el técnico seleccionado <p>Excepciones:</p> <p>Cuando existen conflictos por igual prioridad de análisis el jefe de control de calidad determina el orden según su criterio.</p>
--

Fig.1 – Ejemplo de un escenario futuro

Como puede observarse en la Fig.1, los escenarios son representaciones con una gran riqueza en conocimiento. Esto se logra en parte, concediendo la mayor libertad posible en la expresividad de las descripciones, pero deben existir algunas restricciones para asegurar la consistencia de la información. De esta manera, el modelo de escenarios presenta una sintaxis simple, pensada con dos objetivos fundamentales:

- 1) No entorpecer la construcción de cada escenario.

- 2) Ser transparente para el lector.

En la Fig.2 se puede observar una parte del modelo de escenarios que define la sintaxis utilizada en los episodios, restricciones y excepciones, foco del presente proyecto.

<p>Sintaxis (usando BNF parcial):</p> <pre> <episodios> ::= <serie grupo> <serie episodios> <serie grupo> ::= <grupo> <grupo> < grupo no secuencial> <serie grupo> <grupo> <grupo> ::= <grupo secuencial> < grupo no secuencial> <grupo secuencial> ::= <sentencia básica> <grupo secuencial > <sentencia básica> <grupo no secuencial> ::= # <serie episodios> # <serie episodios> ::= <sentencia básica> <sentencia básica> <serie episodios> <sentencia básica> <sentencia básica> ::= <sentencia simple> <sentencia condicional> <sentencia optativa> <sentencia simple> ::= <sentencia episodio> CR <sentencia condicional> ::= Si <condición entonces <sentencia episodio> CR <sentencia optativa> ::= [<sentencia episodio>] CR donde <sentencia episodio> se describe: (([Actor Recurso] + Verbo + Predicado) ([Actor Recurso] + [Verbo] + Título)) + {Restricción} Excepciones: generalmente reflejan la falta o mal funcionamiento de un recurso. Una excepción impide el cumplimiento del objetivo del Escenario. Eventualmente se puede indicar el tratamiento de la excepción a través de un Escenario. Sintaxis: Causa [(Solución)] donde Causa es: Frase ([Sujeto Actor Recurso] + Verbo + Predicado) donde Solución es: Título </pre>
--

Fig.2 – Sintaxis de los episodios

Para asegurar la calidad de los escenarios, no ha sido suficiente la definición de una sintaxis particular, la utilización del LEL, los patrones de descripción [10] y la heurística de construcción, también se debió incorporar un proceso de verificación que garantice la consistencia de toda la información representada.

En la Tabla 1 se puede observar en un estudio comparativo, que existen diferentes técnicas de verificación dentro de la IR que aplican en diferente momento del proceso.

Documentos	Técnicas de Verificación	Aplican a LN
ERS	45%	56%
Previos a la ERS	55%	33%

Tabla 1 – Técnicas de verificación en la IR

El proceso de requisitos en el cual se basa el presente proyecto, se encuentra en el 55% de las técnicas para documentos previos a la ERS y dentro del 33% que aplican al LN. Entre las técnicas más utilizadas para la verificación se encuentran las Inspecciones [11] [12], las cuales son utilizadas en este proceso para detectar inconsistencias, tanto sintácticas como semánticas. A continuación, se describe el proceso de verificación existente, el cual se ejecuta con formularios de inspección [13].

Pese a la gran flexibilidad con que se pueden escribir los escenarios, éstos son enormemente ricos en cuanto a su posibilidad de contrastar unos contra otros y contra el LEL, es así que se pueden definir dos tipos de consistencias posibles aplicables a un conjunto de escenarios:

- I. Consistencia interna de escenarios
 - I.1 Verificación sintáctica
 - I.2 Interrelación con el LEL
 - I.3 Interrelación entre componentes
- II. Consistencia entre escenarios
 - II.1 Relación entre escenarios
 - II.2 Superposición entre los escenarios
 - II.3 Cubrimiento del LEL

La Consistencia interna de escenarios se ocupa de contrastar las distintas partes constitutivas de los mismos. La Consistencia entre escenarios se ocupa de analizar la integridad del conjunto de escenarios. En la Fig. 3, se esquematizan ambos procesos.

Consistencia interna de escenarios: se ocupa de analizar cada escenario en forma individual. El proceso de consistencia interna de escenarios utiliza para su desarrollo los

siguientes elementos adicionales: el modelo de escenarios y el LEL. Al finalizar el proceso, se obtiene un conjunto de escenarios mejorados y un grupo de dudas que se deben resolver en el UdeD, a partir de las cuales se genera una nueva versión de escenarios y eventualmente un nuevo proceso de consistencia interna de escenarios.

Como se muestra en la Fig.3, el análisis de cada escenario comprende la verificación sintáctica, la interrelación con el LEL y la interrelación de los componentes. La verificación sintáctica comprueba que cada componente haya sido correctamente escrito.

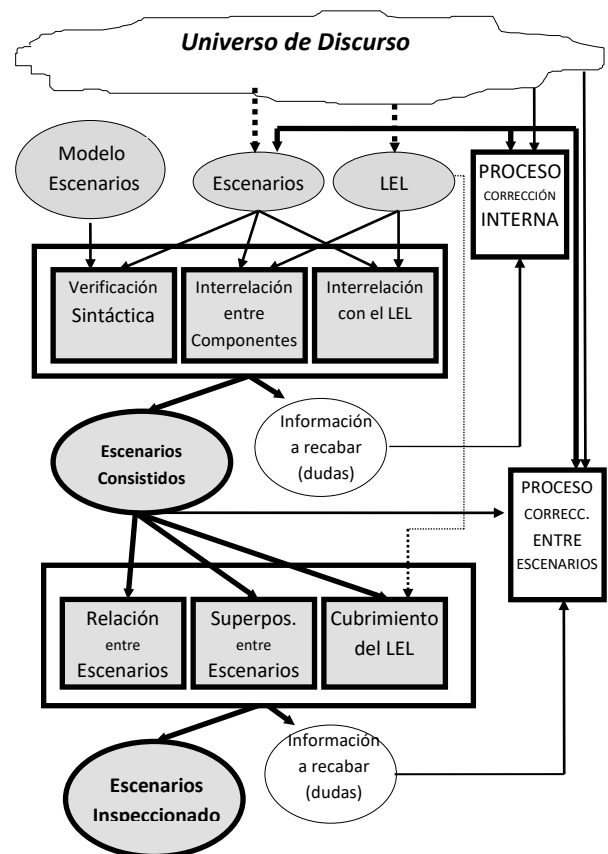


Fig.3 – Inspección de escenarios

La interrelación con el LEL verifica que los símbolos del LEL estén correctamente utilizados y que toda frase destacada como símbolo del LEL efectivamente sea parte del mismo. La interrelación de componentes verifica entre otras cosas, que todos los actores de la lista de actores cumplan un rol en algún episodio y que todo sujeto de los episodios esté en la lista de actores.

Consistencia entre escenarios: este proceso debe realizarse luego de finalizado el proceso de consistencia interna de escenarios. Al finalizar el proceso, se obtiene un conjunto de escenarios mejorados y un grupo de dudas que se deben resolver en el macro sistema, a partir de las cuales se genera una nueva versión de escenarios y eventualmente un nuevo proceso de consistencia interna de escenarios o de consistencia entre escenarios.

2. LÍNEAS DE INVESTIGACIÓN Y DESARROLLO

La verificación existente de escenarios, cubre algunas debilidades del propio proceso de construcción y otras relacionadas con el uso del lenguaje natural. Ninguna de estas inspecciones controla la granularidad de la información presente en cada componente del escenario. La presente línea de investigación se encuadra dentro de la consistencia interna de escenarios (ver Fig.3) pero en este caso, desde una perspectiva cualitativa. El objetivo es obtener un conjunto de escenarios más homogéneos y completos. Para lograrlo es necesario desagregar algunas expresiones para identificar nuevo conocimiento del contexto y detectar omisiones y faltantes de información que no fueron identificadas por las inspecciones existentes.

Se espera clasificar los tipos de soterramiento de conocimiento, como ser la hiperonimia, la polisemia, etc. que aparecen en general en los verbos que no están respaldados por el LEL y generar una heurística que detecte las expresiones que requieran ser desagregadas.

Entre los temas a analizar son:

- 1) Diferencia de granularidad en Episodios, Restricciones y Excepciones.
- 2) Diferencia de granularidad en los componentes Actores y Recursos.
- 3) Inconsistencia entre un episodio y el episodio anterior y posterior.
- 4) Inconsistencia entre un sub escenario con el episodio anterior y el siguiente.
- 5) Inconsistencia entre un episodio y su restricción.
- 6) Inconsistencia entre la excepción y el objetivo del escenario.

- 7) Sobre o sub cubrimiento del objetivo del escenario en el conjunto de episodios de un escenario.

Desagregar la información de los escenarios permite incrementar la comprensión de cada situación del contexto, detectando en algunos casos, otras situaciones involucradas que no habían sido visibilizadas. En el caso de los escenarios actuales, bajar la granularidad de la información, mejora la visión del contexto en estudio, mientras que para los escenarios futuros, asegura la validación de la propuesta para el nuevo sistema de software. La desagregación de conocimiento en los episodios mejora cuantitativa y cualitativamente la obtención de los requisitos funcionales, ya que estos se encuentran empotrados en los episodios. La desagregación en las restricciones y las excepciones puede aumentar la identificación de los requisitos no funcionales.

Al desagregar se puede identificar información faltante como sobrante. Incluir información faltante, incrementa la completitud de los escenarios. Eliminar información sobrante, mejora la legibilidad. En este último caso, puede existir una retroalimentación al LEL, ya que cierta información que aparece en los escenarios corresponde al glosario. De esta manera, la mejora en la granularidad de los escenarios puede repercutir en la completitud del LEL.

3. RESULTADOS OBTENIDOS/ESPERADOS

En un estudio empírico de aproximadamente 20 casos de estudio, se pudo observar que la presencia de diferentes niveles de detalle depende en gran medida del conocimiento previo que tiene el ingeniero de requisitos acerca del contexto en estudio, observando que cuanto mayor es su conocimiento del dominio, mayor es la tendencia a obviar detalles. También se detectó una relación con las fuentes de información utilizadas, ya que durante una entrevista con personal jerárquico la tendencia fue abstraer conocimiento, utilizando expresiones genéricas propias de su rol en la organización. Mientras que con el personal

operativo, en la mayoría de los casos, apareció un mayor nivel de detalle. No sucedió lo mismo con la literatura del dominio, las cuales mantienen una cierta regularidad. También es importante resaltar la importancia de la disponibilidad del usuario para disipar todas las dudas que la literatura del dominio no pueda aclarar. Finalmente, la cultura del ingeniero de requisitos se ve reflejada en el estilo de redacción de los escenarios, esto sucede por la ausencia de un vocabulario mínimo que regule todo aquello que no está definido en el LEL. Se ha comprobado que algunas expresiones parecen encerrar otros conceptos, pero al analizarlas con mayor profundidad resulta en un problema de redacción. En algunos casos, es detectado por la verificación existente, donde se analiza la estructura sintáctica de cada componente. En otros casos, el texto es confuso por una mala interpretación del contexto o por falta de información. Los problemas de redacción, a pesar de existir una sintaxis para cada caso, son muy comunes en descripciones en lenguaje natural y dificultan seriamente la consistencia de los escenarios y la comprensión del problema. Muchos de estos casos se resuelven simplemente con reemplazar parte del texto.

También se detectó que un mejor nivel de detalle genera la aparición de estados y jerarquías del contexto en los escenarios, que suelen quedar ocultos en las descripciones de los episodios, principalmente cuando la granularidad es alta. Esto mejora significativamente la precisión de la descripción de cada situación.

4. FORMACIÓN DE RECURSOS HUMANOS

Esta línea de investigación es parte directa de la tesis de Maestría del Ing. David Tua, en la Escuela de Posgrado de la UTN FRBA.

5. BIBLIOGRAFÍA

- [1] Boehm, B. W., & Papaccio, P. N. (1988), "Understanding and controlling software costs", *IEEE transactions on software engineering*, 14(10), 1462-1477.
- [2] Sharif, Bushra & Shoab, A & Khan, & Bhatti, Muhammad. (2012), "Measuring the Impact of Changing Requirements on software Project Cost: An Empirical Investigation", *International Journal, Computer Science*. Capítulo 9.
- [3] Leite, J., C., S., P., Doorn, J. H., Kaplan, G., Hadad, G., D., S., Ridao, M., N., (2004) "Defining System Context using Scenarios" in *Perspectives on Software Requirements*, Kluwer Academic Publishers, cap.8, pp. 169-199.
- [4] Vallez M. y Pedraza-Jimenez R. (2007), "El Procesamiento del Lenguaje Natural en la Recuperación de Información Textual y áreas afines". *Hipertext.net*, núm. 5. <https://www.upf.edu/hipertextnet/numero-5/pln.html>.
- [5] Leite J.C.S.P., Franco, A.P.M., (1990), "O Uso de Hipertexto na Elicitação de Linguagens da Aplicação", *Anais de IV Simpósio Brasileiro de Engenharia de Software*, SBC, pp. 134-149.
- [6] Hadad, G.D.S., Doorn J.H., Kaplan G.N. (2008), "Creating Software System Context Glossaries", in *Encyclopedia of Information Science and Technology*, Second edition, Mehdi Khosrow-Pour (ed.), Idea Publishing.
- [7] Leite, J. C. S., Hadad, G. D. S., Doorn, J. H., & Kaplan, G. N. (2000), "A scenario construction process", *Requirements Engineering*, 5(1), 38-61.
- [8] Berry, D. and Erik Kamsties (2004), "Ambiguity in Requirements Specification" in *Perspectives on Software Requirements*, Kluwer Academic Publishers, cap.2, pp. 7-39.
- [9] Doorn J.H., Hadad G.D.S., Kaplan G.N. (2002), "Comprendiendo el Universo de Discurso Futuro", *Workshop en Ingeniería de Requisitos*, Valencia-España.
- [10] Ridao, M., Doorn, J., Leite, J.C.S.P., "Incorporación de patrones al proceso de construcción de Escenarios" (2001), *Workshop on Requirements Engineering (WER)*, Buenos Aires, Argentina.
- [11] Fanmuy, G., Fraga, A., & Llorens, J. (2012), "Requirements verification in the industry" in *Complex systems design & management* (pp. 145-160). Springer, Berlin, Heidelberg.
- [12] Leite, J.C.S.P., Doorn, J.H, Hadad, G.D.S., Kaplan, G.N. (2003), "Using Scenario Inspections on Different Scenarios Representations", *Monografias em Ciência da Computação*, Departamento de Informática, PUC-Rio, N.33/03.
- [13] Leite, J.C.S.P., Doorn, J.H., Hadad, G.D.S., Kaplan, G.N. (2005), "Scenario Inspections", *Requirements Engineering Journal*, Vol.10, N° 1, SpringerVerlag London., Gran Bretaña, pp.1-21.