

Evolución de los Factores Situacionales durante el Proceso de Requisitos

Viviana A. Ledesma¹, Graciela D.S. Hadad^{1,2}, Jorge H. Doorn^{1,2}, Juan P. Mighetti¹,
Nicolás A. Bedetti¹, María Celia Elizalde²

¹DIIT, Universidad Nacional de la Matanza

²Escuela de Informática, Universidad Nacional del Oeste

{vledesma, ghadad}@unlam.edu.ar, jdoorn@exa.unicen.edu.ar, {jmighetti, nbedetti}@unlam.edu.ar,
melizalde@uno.edu.ar

RESUMEN

La literatura presenta una diversidad de procesos de software, que dependen en gran medida del tipo de software a construir y de la envergadura del mismo. La selección del proceso más adecuado y la adaptación a un contexto específico suele ser a su vez fuertemente dependiente de la experiencia de los desarrolladores. Sin embargo, condicionar el análisis de los factores situacionales de esta manera es una restricción innecesaria. Por otra parte, muchos de estos factores impactan en el proceso de software desde sus etapas iniciales. Es por ello que es beneficioso realizar una adaptación lo más temprana posible, en función de factores observables en el contexto. Es decir, considerar los factores situacionales desde el proceso de requisitos mismo. A partir de varios casos reales donde se aplicó un proceso de requisitos, se inició la evaluación de dichos factores, observándose que algunos de ellos son difíciles de establecer con precisión en los primeros pasos del proceso o que por su naturaleza cambian a lo largo del proceso. Conocer qué factores son más proclives a evolucionar por mejora de la estimación inicial o por cambios genuinos permitirá una mejor adaptación del proceso de requisitos.

Palabras clave: Ingeniería de Requisitos, Proceso de Requisitos, Ingeniería de Métodos Situacional, Variabilidad de Procesos.

CONTEXTO

La línea de investigación que se presenta es parte de los proyectos de investigación “Adaptación Dinámica de un Proceso de

Requisitos” del programa CYTMA2 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

Implementar una gestión adecuada de los procesos permite a las organizaciones no solo reducir esfuerzos, sino además obtener ventajas competitivas frente a otras organizaciones. Esto no resulta sencillo especialmente cuando se trabaja en ambientes extremadamente dinámicos, los cuales conllevan un alto nivel de variabilidad. Es por ello que los constantes cambios que surgen en dichos entornos crean la necesidad de adaptar continuamente estos procesos. Adaptarse y soportar estos cambios resulta clave para el éxito de las organizaciones [1].

El análisis de variabilidad de procesos involucra establecer qué partes varían (puntos de variación), las causas de esa variación, las alternativas posibles de resolución (flujos variantes) y el mecanismo de resolución de la variabilidad (aplicando operaciones de adición, eliminación, reemplazo o parametrización de partes del proceso) [2, 3].

Es así que la variabilidad puede interpretarse como planear o anticiparse al cambio, en lugar de improvisar cada vez que surge una necesidad o una situación no prevista [3].

Existe un gran número de trabajos de investigación relacionados al análisis de la variabilidad, que ofrecen numerosas propuestas para dominios específicos. Un

campo de estudio es la representación de la variabilidad en los Modelos de Proceso de Negocio [2, 4]. Bajo este enfoque el análisis de la variabilidad consiste en la definición de caminos alternativos de ejecución en un flujo de trabajo, donde es posible indicar qué partes del proceso de negocio permanecen abiertos al cambio o no totalmente definidas, soportando diferentes versiones del mismo proceso, dependiendo del uso proyectado o el contexto de ejecución [5]. En las Líneas de Producto Software también se aplica la gestión de variabilidad [6]. Una técnica que la pone en práctica es la aproximación Base-Variación-Resolución, que propone la construcción de un modelo base, un modelo de variación y un modelo de resolución, para definir de forma explícita las partes del modelo que son fijas, las que pueden cambiar, y las condiciones que harán que éstas cambien [7]. Basándose en la visión de Líneas de Producto Software, se da sustento al concepto de Líneas de Proceso Software [2, 8], facilitando el ajuste sistemático del proceso, incluyendo la reutilización y evolución de forma planificada.

Por lo anterior, se hace evidente que la variabilidad es un factor clave que aplica a muchos aspectos del desarrollo de software en general [9]. En ese sentido, se han propuesto distintas técnicas para modelar la variabilidad que van desde diagramas de actividad de UML [10], hasta otros más específicos tales como la notación para el Modelado de Procesos de Negocio [11], la Especificación de Recursos Reutilizables [12] que define de manera estándar la información asociada a un activo o recurso, incluyendo las guías necesarias para facilitar la manipulación y reutilización del mismo, o el Metamodelo de Ingeniería de Procesos de Software (SPEM) [13] que especifica un lenguaje para modelar familias de procesos de software.

Desde otra visión del modelado de procesos, la Ingeniería de Métodos se enfoca en el diseño de métodos mediante la composición de fragmentos o bloques de proceso existentes, con el fin de mejorar la productividad de los mismos [14]. A partir de esta disciplina, surgió la Ingeniería de

Métodos Situacional (IMS), enfocada en el diseño de métodos de desarrollo de software adaptados a situaciones particulares [15, 16]. La situación se describe a través de la estimación de un conjunto de factores, en base a los cuales se determina la adaptación a realizar del proceso de software [17]. Estos factores muestran tanto peculiaridades del contexto de aplicación como del proyecto de software [18]. Por lo tanto, un proceso se construye para una situación dada a partir de la combinación de bloques de proceso predefinidos y en función de los factores situacionales identificados, donde existen bloques comunes a cualquier situación y otros variantes [16].

La IMS no solo considera el desarrollo de componentes de proceso sino también componentes de producto, e incluso componentes que ensamblan tanto proceso como producto (bloques mixtos) [19]. La ventaja que conlleva el enfoque de uso separado de bloques de proceso y bloques de producto sobre los bloques mixtos, es que el primero brinda una mayor flexibilidad en el proceso de adaptación, ya que varios bloques de proceso podrían asociarse a los mismos bloques de producto [15, 19].

En general, las técnicas que aplican la IMS coinciden en los siguientes pasos para la construcción del nuevo método:

- Caracterizar el proyecto, a partir de definir la situación.
- Seleccionar los fragmentos existentes que conformarán el método.
- Ensamblar los fragmentos de modo tal que se obtenga el método para ese proyecto específico.

En el proceso de IMS propuesto en [16], el ingeniero de métodos selecciona los bloques del proceso base y construye un proceso específico, en función de los factores situacionales y las guías de construcción existentes.

Han surgido algunas propuestas desde la IMS que abordan el diseño y mejora de procesos de Ingeniería de Requisitos (IR), adaptándolos a las características particulares de un proyecto. Algunas propuestas están orientadas a definir un proceso de IR

utilizando frameworks, ya sea que existan en el mercado o propuestos por los autores, partiendo de una base de conocimiento sobre actividades y técnicas de IR, que sirve de apoyo para obtener un proceso acorde a las necesidades del proyecto [20, 21]. Varias propuestas se enfocan exclusivamente en la selección de las técnicas de elicitación que mejor se adapten al contexto situacional [22, 23]. Otros autores presentan modelos para adaptar procesos de IR para proyectos de determinada naturaleza o alcance, tales como desarrollo de software global o métodos ágiles, tomando algunos principios de la IMS para flexibilizar el proceso, incluyendo otros atributos contextuales [24, 25]. En otros trabajos [26, 27] se han aplicado algunos principios de la IMS para definir procesos de IR adaptables, mediante el uso de componentes modulares existentes, incluso combinando nociones de Variabilidad de Procesos y de Líneas de Producto Software.

En estos trabajos, se aplican en mayor o menor medida algunos conceptos de la IMS, proponiendo factores situacionales que son evaluados para definir el contexto, donde algunos enfoques se basan en la reutilización de componentes para conformar el nuevo proceso de IR. Sin embargo, en general no presentan propuestas de adaptación dinámica del proceso. Una aproximación a ello es propuesto en [28], donde se enfatiza que el conocimiento acerca del dominio del problema y del dominio del proyecto se va adquiriendo a medida que avanza el proceso de IR, con lo cual en cada iteración se puede mejorar la actividad de elicitación y de esta forma se mejoraría el producto final. En este sentido, Rolland [19] menciona que las características que describen una situación pueden cambiar a lo largo del proceso, pudiendo ser necesario readaptar el proceso inicialmente construido para esa situación.

2. LÍNEAS DE INVESTIGACIÓN Y DESARROLLO

A lo largo de dos décadas en varios proyectos de investigación, se ha desarrollado un proceso de IR basado en modelos en lenguaje natural [29], que ha sido difundido

en cursos de grado y posgrado, y puesto en práctica en diversos proyectos de software de mediana a gran envergadura en la industria. Aún cuando este proceso puede considerarse suficientemente maduro, todavía requiere mejoras en algunos aspectos. Una mejora apunta a la flexibilización y optimización del proceso, por lo que se propone mejorar la aplicación de la IMS incorporando nociones de análisis de variabilidad de procesos, y utilizando el proceso existente y las estrategias ya desarrolladas para su adaptación estática, como base para encarar su adaptación dinámica. Esto involucra tomar decisiones referidas a qué artefactos de requisitos construir, qué actividades del proceso realizar y qué técnicas específicas aplicar en cada actividad, de manera de lograr una mayor productividad del proceso, y procurando una mejor calidad de los requisitos del software, considerando la posibilidad que los factores situacionales hayan evolucionado o hayan sido estimados incorrectamente. Entender las decisiones adoptadas en la aplicación de una variante específica del proceso de IR permitirá establecer mejoras en adaptaciones posteriores.

Manejar alternativas en el proceso de IR implica la existencia de bloques de proceso comunes independientes de la adaptación, bloques de proceso alternativos y otros bloques de proceso con alguna variabilidad interna menor. Se asume que la variabilidad en los procesos también podrá influir en los productos, pudiendo entonces tenerse modelos alternativos (por ejemplo, diferentes estructuras para el documento de especificación de requisitos) o familia de modelos (por ejemplo, un conjunto de escenarios vinculado a un glosario del universo de discurso, o vinculado a un glosario que describe términos del sistema de software, o sin vínculos a ningún glosario).

Asimismo, ocurre con frecuencia que las condiciones iniciales de un proyecto e incluso el contexto de aplicación se alteran, pudiéndose necesitar una nueva adaptación del proceso [19]. Es por ello que se propone una adaptación continua del proceso mediante

el monitoreo de los factores que pueden afectarlo en determinados hitos, proveyendo guías precisas para su reajuste.

3. RESULTADOS OBTENIDOS/ESPERADOS

A continuación se enumeran los resultados ya alcanzados: i) se ha identificado un conjunto amplio de 25 factores situacionales con sus interacciones y tipo de evolución, incluyendo factores no contemplados inicialmente relativos al desarrollo de software global y a la calidad esperada de los requisitos; ii) se han determinado los puntos de variación del proceso de requisitos y los factores que impactan en cada punto; iii) se han definido preliminarmente los bloques de proceso necesarios para componer el proceso; y iv) se han establecido las reglas de adaptación del proceso en cada punto de variación, las que permiten seleccionar los bloques que conformarán el proceso para la situación específica. Con base en estas definiciones, se diseñó una versión preliminar de los mecanismos de adaptación dinámica del proceso de requisitos, donde para cada proyecto de software se debe estimar un valor para cada factor situacional, el grado de certeza esperado en su estimación “a priori” y la posibilidad de reevaluarlo en etapas intermedias del proceso de requisitos. Estas definiciones han quedado parcialmente reflejadas en [30].

Se ha iniciado un estudio de la evolución de los factores situacionales durante la aplicación del proceso de IR, disponiéndose de 35 casos de estudio desarrollados entre 2015 y 2017. Se está desarrollando una herramienta de apoyo para la carga y posterior análisis de los cambios sufridos por los factores situacionales al inicio y al final del proceso de requisitos, combinado con el grado de confiabilidad asumido para el factor. En una etapa muy temprana del análisis, se ha notado que ciertos factores, como complejidad del contexto, grado de reingeniería del proceso de negocio, conflicto de intereses de usuarios y envergadura del proyecto, deben ser observados con especial atención durante el proceso de IR pues suelen

cambiar, mientras que factores como novedad del contexto y nivel de rotación del equipo de desarrollo suelen ser percibidos desde el inicio con mayor precisión.

En las próximas etapas de investigación, se profundizará el análisis de la evolución que sufren los factores situacionales a lo largo del proceso de requisitos desde su estimación inicial a su valoración final. De esta manera se intentará identificar el impacto que valoraciones dudosas en ciertos factores pueden tener en el proceso de requisitos adaptado. Se propone establecer alguna heurística que permita alcanzar un valor más preciso desde el inicio de la planeación de dicho proceso.

Por otro lado, se llevará a cabo la puesta a prueba de la adaptación del proceso de IR en aquellas situaciones consideradas más usuales (dada la diversidad de situaciones posibles) con el fin de comprobar el nivel de impacto alcanzado en la productividad del proceso.

4. FORMACIÓN DE RECURSOS HUMANOS

En el proyecto de UNLaM participan tres investigadores y un alumno becario. En el proyecto de UNO participan en este tema otros dos investigadores, uno en formación.

La línea de investigación presentada aquí es parte directa de la tesis de maestría “Estrategia de Requisitos adaptable según factores de situación” que está desarrollando la Ing. Ledesma en UNLaM. El Ing. Mighetti finalizó su tesis de maestría “Mitigación de amenazas a requisitos en el desarrollo global de software usando LEL y Escenarios” en UNLaM, faltando presentarla para su defensa; esta tesis ha permitido identificar factores situacionales no considerados inicialmente.

5. BIBLIOGRAFÍA

- [1] Wheelen, T.L., Hunger, J.D. (2012) Strategic Management and Business Policy: Toward Global Sustainability, 13° ed., Pearson.
- [2] Schnieders, A., Puhlmann, F. (2006) Variability Mechanisms in E-Business Process Families. Intl Conference on Business Information Systems, pp. 583-601.
- [3] Galster, M., Weyns, D., Tofan, D., Michalik, B., Avgeriou, P. (2014) Variability in Software Systems - A Systematic Literature Review. IEEE

- Transactions on Software Engineering, 40, 3, pp.282-306.
- [4] Groefsema, H., Bulanov, P., Aiello, M. (2011) Declarative Enhancement Framework for Business Processes. Intl Conference on Service-Oriented Computing, pp.495-504. New York: Springer.
- [5] Santos, E., Castro, J., Sánchez, J., Pastor, O. (2010) A Goal-Oriented Approach for Variability in BPMN. 13th Workshop on Requirements Engineering, pp. 17-28.
- [6] Pol'la, M., Buccella, A., Cechich, A., Arias, M. (2014) Un modelo de metadatos para la gestión de la variabilidad en líneas de productos de software. 43 JAIIO, Buenos Aires.
- [7] Haugen, Ø., Øgård, O. (2014) BVR – Better Variability Results. Intl Conference on System Analysis and Modeling: Models and Reusability. SAM 2014, Lecture Notes in Computer Science, vol. 8769, pp. 1-15. Springer.
- [8] Simidchieva, B., Clarke, L., Osterweil, L. (2007) Representing Process Variation with a Process Family. Software Process Dynamics and Agility. ICSP 2007. Lecture Notes in Computer Science, vol. 4470, pp. 109-120. Springer, Berlín.
- [9] Chen, L., Babar, M. (2010) Variability Management in Software Product Lines: An Investigation of Contemporary Industrial Challenges. 14th Intl Conference on Software Product Lines: going beyond, pp. 166-180. Springer-Verlag Berlin.
- [10] Razavian, M., Khosravi, R. (2008) Modeling Variability in Business Process Models Using UML. 5th Intl Conference on Information Technology: New Generations. EEUU: IEEE.
- [11] Delgado, A., Calegari, D. (2017) BPMN 2.0 Based Modeling and Customization of Variants in Business Process Families. XLIII CLEI, Córdoba.
- [12] Pacini, K., Braga, R. (2015) An Approach for Reusing Software Process Elements based on Reusable Asset Specification: A Software Product Line Case Study. 10th Intl Conference on Software Engineering Advances, pp. 200-206. Barcelona.
- [13] Martínez-Ruiz, T., García, F., Piattini, M., Munch, J. (2011) Modelling Software Process Variability: An Empirical Study. IET Software, 5,2, pp.172-187.
- [14] Brinkkemper, S. (1996) Method Engineering: Engineering of Information Systems Development Methods and Tools. Information and Software Technology, 38, 4, pp. 275-280.
- [15] Henderson-Sellers, B., Ralyté, J. (2010) Situational Method Engineering: State-of-the-Art Review. Journal of Universal Computer Science, 16, 3, pp. 424-478.
- [16] Henderson-Sellers, B., Ralyté, J., Agerfalk, P., Rossi, M. (2014) Situational Method Engineering, Springer-Verlag Berlin Heidelberg.
- [17] Khan, H.H., bin Mahrin, M., bt Chuprat, S. (2014) Factors for Tailoring Requirement Engineering Process: A Review. Intl Journal of Software Engineering and Technology, 1,1, pp.7-18.
- [18] Bucher, T., Klesse, M., Kurpjuweit, S., Winter, R. (2007) Situational Method Engineering. Situational method engineering: fundamentals and experiences, Springer US, pp. 33-48.
- [19] Rolland, C. (2008) Method engineering: towards methods as services. Making Globally Distributed Software Development a Success Story, Springer Berlin, pp. 10-11.
- [20] Zowghi, D., Firesmith, D., Henderson-Sellers, B. (2005) Using the OPEN Process Framework to Produce a Situation-Specific Requirements Engineering Method. 1st Intl Workshop on Situational Requirements Engineering Processes, pp.59-74, Paris.
- [21] Jiang, L., Eberlein, A. (2008) A Framework for Requirements Engineering Process Development (FRERE). 19th Australian Conference on Software Engineering.
- [22] Hickey, A., Davis, A. (2003) Elicitation technique selection: how do experts do it? 11th IEEE Intl Requirements Engineering Conference, pp.169-178.
- [23] Carrizo, D., Dieste, O., Juristo, N. (2008) Study of elicitation techniques adequacy. 11th Workshop on Requirements Engineering, pp. 104-114.
- [24] Bakhat, K.A., Sarwar, A.A., Motla, Y.H.B., Akhtar, M.C. (2015) A Situational Requirement Engineering Model for an Agile Process. Bahria University Journal of Information & Communication Technology, 8, 1, pp. 21-26.
- [25] Khan, H., bin Mahrin, M., Mali, M. (2016) Situational Requirement Engineering Framework for Global Software Development: Formulation and Design. Bahria University Journal of Information & Communication Technologies, 9, 1, pp. 74-84.
- [26] Coulin, C., Zowghi, D., Sahraoui, A. (2006) A Situational Method Engineering Approach to Requirements Elicitation Workshops in the Software Development Process. Software Process: Improvement and Practice, 11, 5, pp. 451-464.
- [27] Jafarinezhad, O., Ramsin, R. (2012) Development of Situational Requirements Engineering Processes: A Process Factory Approach. 36th IEEE Intl Conference on Computer Software and Applications, pp.279-288.
- [28] Hickey, A.M., Davis, A.M. (2004) A Unified Model of Requirements Elicitation. Journal of Management Information Systems, 20, 4, pp.65-84.
- [29] Leite, J., Doorn, J., Kaplan, G., Hadad, G., Ridao, M. (2004) Defining System Context using Scenarios. Perspectives on Software Requirements, Kluwer Academic Publishers, cap.8, pp.169-199.
- [30] Hadad, G.D.S., Doorn, J.H., Ledesma, V.A. (2018) Dynamic Situational Adaptation of a Requirements Engineering Process. Encyclopedia of Information Science and Technology, 4º ed., IGI Global, EEUU, cap. 646, pp. 7422-7434.