Hacia una automatización de los procesos de desarrollo de software

Marcela Daniele, Marcelo Ariel Uva, Paola Martelloto
{*marcela,uva,pmartelloto*}*@dc.exa.unrc.edu.ar*
Dpto. de Computación-Facultad de Cs. Exactas Fco-Qcas y Naturales
Universidad Nacional de Río Cuarto

Resumen

El Proceso Unificado es una metodología de desarrollo de software que propone la construcción de un sistema con un proceso iterativo e incremental, centrado en la arquitectura y dirigido por casos de uso. En cada etapa del proceso se definen y refinan las funcionalidades del sistema, con el fin de lograr un producto de calidad. En todo proyecto de software, un alto porcentaje de sus funcionalidades presentan similar comportamiento, como son: "Alta", "Baja", "Modificación" y "Consulta" (ABMC). Esto hace posible plantear una solución que permita independizarse del elemento particular, y generalice el comportamiento descrito para cualquier entidad. Las plantillas genéricas para la descripción, análisis y diseño de casos de uso proveen una solución única a esta categoría de problemas. Este trabajo propone el desarrollo de una herramienta de ingeniería que implementará las plantillas genéricas definidas [2, 3] para cada etapa, automatizando las etapas del ciclo de vida de desarrollo de un software según el Proceso Unificado [1]. La herramienta reducirá el tiempo de producción de ABMC, destinando mayores recursos a funcionalidades que aportan mayor valor agregado al sistema.

Contexto

La línea de investigación presentada en este trabajo se desarrolla en el marco de los proyectos "Definición y Uso de Plantillas Genéricas para la Descripción de Casos de Uso" y "Evolución de Plantillas Genéricas para la descripción de Casos de Uso a Plantillas Genéricas para Análisis y Diseño", presentados en la Universidad Nacional de Río Cuarto. Desde el año 2005, un grupo de docentes e investigadores del área Ingeniería de Software de las carreras de ciencias de la computación del Dpto. de Computación, Facultad de Ciencias Exactas, Físico-Químicas y Naturales de la UNRC, vienen trabajando en dichos proyectos.

Palabras Clave: *Proceso Unificado, Automatización de Casos de Uso, Plantillas Genéricas.*

1. Introducción

La Ingeniería de software (IS) ofrece métodos y técnicas para desarrollar y mantener software de calidad [4]. Las metodologías de desarrollo son las encargadas de caracterizar las fases principales del desarrollo, administrar el progreso y mantenimiento, permitiendo estimar recursos, definir puntos de control y monitorear el avance del proyecto, entre otras. En el modelado orientado a objetos (MOO), un sistema es visto como un conjunto de objetos que interactúan unos con otros a través de los servicios que prestan. El objetivo del MOO es identificar los objetos que existen en el dominio del problema, especificar la información que encapsulan y los servicios que prestan, e identificar las relaciones existentes entre ellos, para soportar los servicios deseados por el usuario. El MOO ha tenido mucho auge en los últimos tiempos dado que hacen más fácil construir y mantener los sistemas. El Proceso Unificado (PU) [1] es una metodología de desarrollo de software que propone la construcción de sistemas haciendo uso de un proceso iterativo e incremental, centrado en la arquitectura y dirigido por casos de uso (CU). En cada una de las etapas del proceso se definen y refinan los casos de uso del sistema, con el fin de lograr un producto de calidad. En la actualidad el Proceso Unificado es una metodología ampliamente utilizada por las empresas de desarrollo de software, con las diversas variantes que éstas le introducen. En los trabajos finales de los alumnos de las carreras de computación de la UNRC donde desarrollan un sistema de software a un cliente real, el PU es la metodología más elegida. Analizando una gran cantidad de trabajos se ha podido determinar el gran volumen de documentación que se genera en cada etapa del ciclo de vida y el costo que requiere mantenerla consistente. Por otro lado, se puede observar que en estos proyectos, más de un 50 % de los CU del sistema identificados en cada proyecto son del tipo ABMC. Además, salvo ciertas funcionalidades específicas de cada proyecto, la mayoría de los CU responden a la misma lógica operativa, como por ejemplo "Alta producto" y "Alta cliente". Las plantillas genéricas [2, 3] de análisis y diseño dan una solución óptima a este tipo de problemas. Con el uso de las plantillas se reduce el volumen de documentación, se logran soluciones homogéneas para el mismo tipo de problema, favoreciendo tanto al diseñador y programador, como al usuario final que ve y maneja al sistema con una apariencia y comportamiento similar en todas sus funcionalidades. Al contar con plantillas para definir los casos de uso en la primera etapa de Captura de Requerimientos, y luego en su evolución para las etapas de Análisis y Diseño, es lógico pensar que quien utilice estas plantillas también intentará hacer implementaciones de las mismas cada vez más generalizadas, es decir, que puedan ser reusadas en otros proyectos de software.

Siguiendo en esta misma línea investigativa se propone la automatización de las etapas del ciclo de vida del PU para los ABMC. Para ello se está trabajando en el desarro-

llo de una meta-herramienta de ingeniería que tomará como entrada el "Modelo de Domino" [1] del problema y la descripción de los CU a través de las plantillas genéricas. Luego, la herramienta deberá realizar la transición por todas las etapas del ciclo de vida del PU hasta llegar a la implementación, generando toda la documentación (diagramas UML) correspondiente. La herramienta disminuirá los tiempos planificados para la construcción de los ABMC, posibilitando destinar mayores recursos a otros CU que aportan el valor agregado al sistema.

2. Implementación de la popuesta

Como ya se introdujo, en la mayoría de los proyectos de software, más del 50 % de las funcionalidades a implementar son ABMC, las cuales se vuelven repetitivas, debido a que los desarrollos de ABMC no poseen diferencias sustanciales. Por ej.el desarrollo de la funcionalidad "Alta de producto" no debería diferir operativamente de "Alta de cliente". La herramienta propuesta en este trabajo surge de la necesidad de automatizar la generación de los ABMC dentro del ciclo de vida del PU. Para ello se tiene planificado el desarrollo de una serie de módulos que permitan simplificar el mantenimiento de los programas, mejorar y estandarizar la documentación, aumentar la portabilidad de las aplicaciones, facilitar la reutilización de componentes software, permitir un desarrollo, entre otros aspectos. Los componenetes con los que contará la herramienta son los siguientes:

Módulo de Importación: Este componente permitirá la adquisición, en formato XML [5], de un Modelo de Dominio del sistema [1]. El modelo de dominio consistirá en un diagrama de clases UML [6]. El componente identificará y extraerá clases, atributos, relaciones y toda la información útil para la definición y construcción de AMBC. El formato XML ha sido seleccionado para la adquisición y exportación de información en todos los componentes.

Módulo de Edición: Este componente proporcionará las herramientas gráficas necesarias para la selección de la configuración de los ABMC a generar, y al mismo tiempo se proveerán las especificaciones necesarias. Es en este punto, donde el desarrollador seleccionará las plantillas que va a utilizar, junto con todos los datos requeridos para el registro de las mismas.

Módulo Constructor: Este componente será el encargado de instanciar las plantillas genéricas y construir los artefactos de software asociados a la implementación del AMBC.

Módulo de Documentación: Recopilará la información ingresada por el usuario y la generada en cada etapa, la organizará y la presentará en un documento final. El documentador deberá estar alerta durante todo el proceso de desarrollo, manteniendo siempre la consistencia de los documentos producidos.

3. Resultados obtenidos / esperados:

Desde el año 2005, se trabaja en los proyectos "Definición y uso de Plantillas Genéricas para la descripción de Casos de Uso" y "Evolución de Plantillas Genéricas para la descripción de Casos de Uso a Plantillas para Análisis y Diseño", con el objeto de lograr soluciones genéricas a problemas similares, como lo son los ABMC. Además, la aplicación de estas plantillas posibilita la reducción de los tiempos de producción de funcionalidades similares en el desarrollo de un producto de software.
Estas propuestas cubrieron las tres primeras etapas (descripción, análisis y diseño de CU) en el ciclo de vida de desarrollo. Actualmente, se trabaja en la construcción de una metaherramienta ingenieril que asista al desarrollador en la generación de ABMC siguiendo la metodología de desarrollo del PU, cubriendo las etapas de implementación y documentación para el tipo de funcionalidades planteadas.

4. Formación de Recursos Humanos

Durante el desarrollo de esta línea de investigación han logrado obtner el título Magister en Ingeniería de Software dos integrantes del grupo de trabajo. También, se han formado ayudantes de segunda en las asignaturas de Análisis y Diseño de Sistemas e Ingeniería de Software. Finalmente, un grupo de tres alumnos están realizando su trabajo final de fin de carrera de Analista en Computación, entre otros trabajos.

5. Conclusiones y trabajos futuros

La aplicación de las plantillas genéricas para la descripción, análisis y diseño de casos de uso, proporcionan las bases para la generación automática de sistemas basados en el PU. En el mercado, exiten algunas herramientas que facilitan el desarrollo de sistemas, por ejemplo: Genexus, que generalmente, no están respaldadas por metodologías de desarrollo adecuadas para la construcción y la evolución del software. Es muy común que el mantenimiento de los sistemas producidos con este tipo de herramientas sea muy costoso, resultando más conveniente desechar el trabajo realizado y re-comenzar todo desde cero. Como la herramienta propuesta implementa las plantillas de diseño y las mismas utilizan patrones de diseño [7] para su definición, ello permite asegurar que el diseño y código obtenidos a partir de la aplicación de esta herramienta, contarán con las características de calidad deseadas para cualquier producto de software. Además, la herramienta permitirá la optimización de los tiempos de producción de un proyecto de desarrollo, la facilidad de uso del software por parte del usuario final, ya que el código producido tendrá la misma apariencia y un comportamiento homogéneo. También, facilitará el mantenimiento, ya que se deben realizar las modificaciones considerando sólo el tipo de problema

al que corresponde, abstrayéndose del dominio particular.
Por otro lado, el equipo de desarrollo se ve librado de la realización de CU repetitivos disminuyendo la probabilidad de cometer errores.
En trabajos futuros se pretende extender la herramienta para que pueda generar ABMC siguiento los principios de otras metodología de desarrollo. Otra extensión planificada es la incorporación de la etapa de prueba o testing dentro de la herramienta.

Referencias

[1] I. Jacobson, G. Booch, J. Rumbaugh. El Proceso Unificado de Desarrollo de Software. Addison-Wesley, 2000.

[2] M. Daniele, D. Romero. “Evolución de Plantillas Genéricas para la descripción de Casos de Uso a Plantillas Genéricas para Análisis y Diseño”. VII Workshop de Investigadores en Ciencias de la Computación, 2005.

[3] M. Daniele, D. Romero. “Extendiendo las plantillas genéricas para la definición de casos de uso con un framework genérico distribuido", III Congreso de Tecnología en Educación y Educación en Tecnología (TE/ET’08), Bahía Blanca, Arg, 2008.

[4] C. Ghezzi, M. Jazayeri, D. Mandrioli. Fundamentals of Software Engineering. Prentice Hall, 1991.

[5] Johnny Brochard. “XML Conceptos e Implementación". Barcelona. Ediciones Software SL, 2001.

[6] Booch G., Rumbaugh J., Jacobson I., The Unified Modeling Language. 2.0. Addison Wesley, Second Edition, 2005.

[7] Gamma Erich, Helm Richard, Johnson Ralph, Vlissides John. "Design Patterns: Elements of Reusable Object-Oriented Software". Addison-Wesley, 1995.