

GENERACIÓN DE APLICACIONES MEDIANTE LENGUAJES ESPECIFICOS DE DOMINIO

1. A.Cortez, C.Naveda

1. Consejo de Investigaciones (CIUDA) – UDA.

2. Instituto de Investigaciones – Facultad de Ciencias Administrativas y Sociales.
UDA.

3.cortezalberto@gmail.com, claudia_naveda@hotmail.com

Resumen

En los últimos años ha tomado relevancia la necesidad de mejorar el proceso de desarrollo de software. Surgieron paradigmas como MDE (por sus siglas en inglés, Model Driven Engineering) [2], que promueven el uso eficiente de modelos en el desarrollo de software. Evidentemente el modelo se vuelve el foco de atención. Un modelo puede representarse y especificarse a través de una notación visual (a través de símbolos o graficas) o por medio de textos o escritos. Si se reduce la complejidad del modelo, delimitándolo a un ámbito particular, se puede representar utilizando la metodología DSM (Domain-Specific Modeling por sus siglas en inglés), para modelado específico de dominio. Y se puede utilizar para su representación el Lenguaje Específico de Dominio, por sus siglas en ingles DSL (Domain Specific Language). El modelado específico incluye la generación automática

de código y la creación de código ejecutable desde los modelos. Recurriendo a estas herramientas, las aplicaciones finales de usuario se pueden generar a partir de especificaciones de alto nivel, los modelos. DSM se diferencia de tentativas anteriores para la generación del código, al reducirse la complejidad y trabajar sobre un dominio.

Palabras clave: MDE, Modelos, DSL.

Contexto

El presente trabajo articula se enmarca dentro de las actividades en Investigación y Desarrollo que promueve el Consejo de Investigaciones (CIUDA) de la Universidad del Aconcagua. En este contexto , se pretende profundizar y desarrollar los estudios en el marco de la línea de investigación “Herramientas para ingeniería de software: Generación automática de código multiplataforma”.

Introducción

El propósito del Modelado Específico de Dominio (DSM) [4,5,10] es crear modelos, utilizando un lenguaje enfocado y especializado en un dominio. Dicho lenguaje es conocido como Lenguaje Específico de Dominio (DSL) [6,7]. Se incluye en este marco la generación automática de código ejecutable directamente desde los modelos. En consecuencia las aplicaciones finales se conciben a partir de especificaciones de alto nivel. Así, se libera al desarrollador de las tareas de codificación y mantenimiento del código fuente, y se incrementa significativamente su productividad. La metodología propone la formalización de modelos a través su descripción gráfica o textual y un conjunto de reglas que lo caracterizan.

Se puede lograr mayor eficiencia en esta derivación si la cantidad de aspectos considerados en el problema a tratar es finito, lo que conforma el “dominio específico” de los modelos.

La metodología precisa la creación de tres elementos principales que serán la herramienta fundamental:

un lenguaje de dominio específico formal a través del cual se definen las soluciones,
 un generador que transforme los modelos a código ejecutable, y
 un framework de dominio que sirva como base al código generado en el punto anterior,

para que la transformación sea más sencilla y reusable.

Para crear un DSL es necesario un marco teórico brindado por el metamodelado. El metamodelado permite describir los elementos de un modelo. Un lenguaje como DSL está compuesto por:

- Sintaxis abstracta
 - Define los diferentes elementos del lenguaje y las reglas que establecen cómo pueden ser combinados
- Sintaxis concreta
 - Define cómo los elementos del lenguaje aparece en una notación visual o textual utilizable por las personas
- Semántica
 - Define el significado del lenguaje

En conclusión, el objetivo de esta investigación es lograr un generador de aplicaciones basado en modelos con el uso herramientas DSL.

Líneas de investigación y desarrollo

En el marco de este proyecto se plantea el estudio de :

DSM como metodología para la generación de código automático. El análisis de los elementos a implementar en DSM para el desarrollo de software de manera automatizada.

La validez de los lenguajes específicos de dominio como herramienta para la construcción de modelos a partir del marco teórico brindado por DSM.

El estudio de herramientas que configuren un entorno para la creación de lenguajes específicos de dominio.

Resultados y Objetivos

Como resultado de aplicar los conceptos investigados se obtuvo un generador de modelos producido en Eclipse. Se eligió Eclipse como la herramienta más adecuada para la implementación de los conceptos teóricos, después de un cuidadoso estudio. Es

un entorno multiplataforma y de código abierto. Y un punto considerado en la elección de esta herramienta, es su interoperatividad con otras herramientas.

Se creó un ejemplo sencillo a los efectos de asimilar y aplicar los conceptos de DSL. El ejemplo contiene sólo cuatro elementos que detallan una clase dentro de un proyecto: proyecto, clase, relación, operación y atributo. El primer paso fue producir un metamodelo reducido al ejemplo, y de esta manera definir la sintaxis abstracta. El metamodelo se elaboró creando un metamodelo Ecore a través del plugin EMF [3,8]. La figura 1 muestra una vista del modelo logrado a través del editor visual.

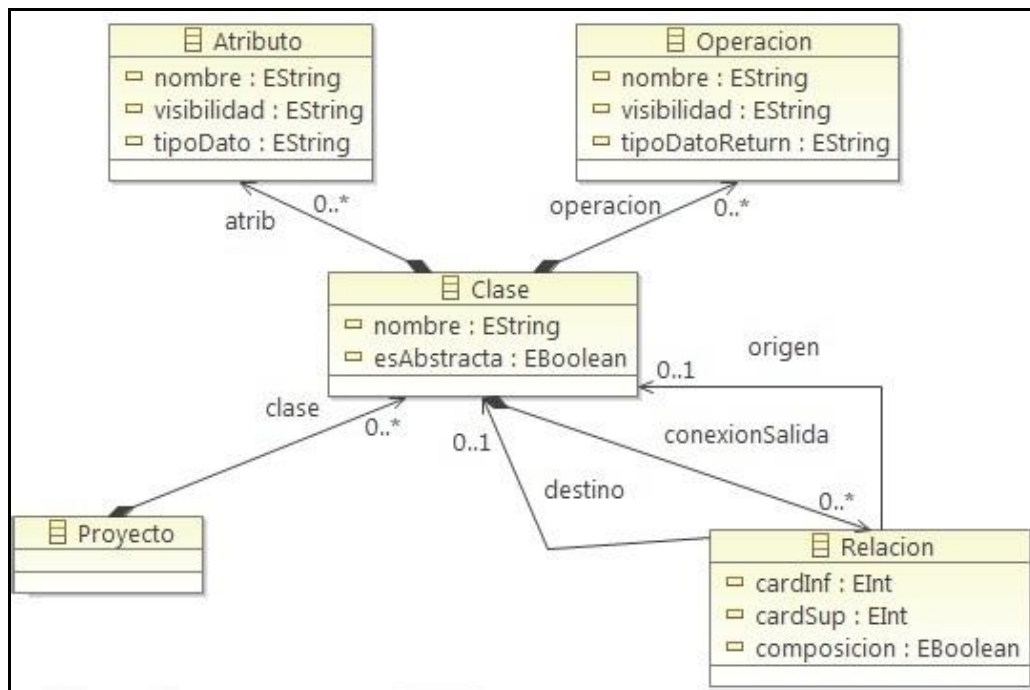


Figura 1 Metamodelo Ecore

En este caso se optó por implementar un generador gráfico utilizando el plugin GMF

[9] para implementar la sintaxis concreta. Se determinaron tres tipos de elementos:

elementos gráficos (archivo gmfgraph), elementos de la paleta de herramientas (archivo gmftool) y elementos de mapeo entre los anteriores (archivo gmfmap). A continuación se muestra en la figura 2 la paleta de herramientas generada en el archivo gmftool. De esta manera se elaboró un diseño para una herramienta de modelado. En la figura 3 puede verse una muestra abreviada de la utilización de la herramienta. En este caso se definieron dos clases (persona y domicilio) con sus correspondientes operaciones y

atributos, unidas por una relación.

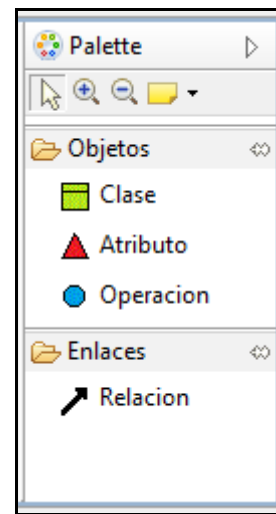


Figura 2 Paleta de herramientas

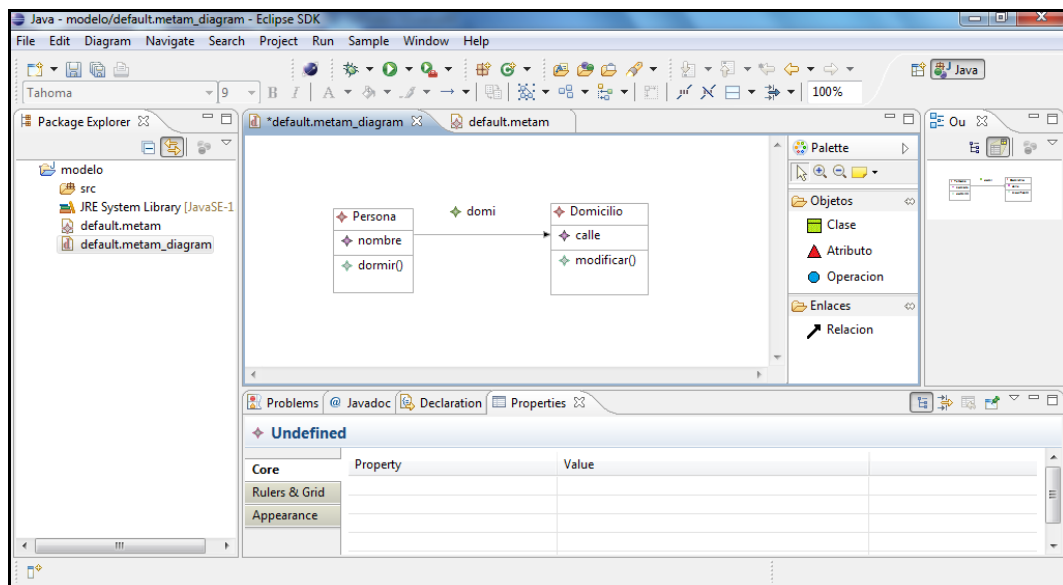


Figura 3 Ejemplo de modelado

Se está trabajando en el desarrollo de un metamodelo completo que contenga las especificaciones para la construcción de un generador integral. Este generador será la base para producir código automático desde el modelo. Como resultado se producirá código para :

el lenguaje php y su mecanismo de persistencia en mysql; el framework Java Server Faces y la API de persistencia JPA. La creación y aplicación de la herramienta muestra la versatilidad de las técnicas DSL para el desarrollo de software.

Formación de Recursos Humanos

En el marco de este proyecto se está desarrollando una investigación en el Consejo de Investigaciones de la Universidad del Aconcagua. También se está trabajando en esta línea de investigación en el Instituto de Investigaciones de Facultad de Ciencias Administrativas de la Universidad del Aconcagua. Adicionalmente se está desarrollando una tesis de grado en la Universidad Mendoza.

Referencias

- [1] Andino Luciano Omar, Ruiz Germán Esteban. (2009). Análisis y uso de los frameworks de Eclipse para la definición de DSLs. Universidad Nacional de La Plata.
- [2] Trask Bruce y Roman Angel. (2011). Applying Model Driven Engineering Technologies in the Creation of Domain Specific Modeling Languages. ACM/IEEE 14th International Conference on Model Driven Engineering Languages and Systems
- [3] Steinberg Dave , Budinsky Frank , Paternostro Marcelo y Merks Ed. EMF: Eclipse Modeling Framework, 2nd Edition. Addison Wesley Professional.
- [4] Lan Cao, Balasubramaniam Ramesh, Matti Rossi. (2009) . Are Domain-Specific Models Easier to Maintain Than UML Models?, IEEE Software, vol. 26, no. 4, pp. 19-21.
- [5] Bibbo L.M. , Garcia D. Garcia y Pons C. 2008. A Domain Specific Language for the Development of Collaborative Systems. IEEE International Conference of the Chilean Computer Science Society SCCC 2008.
- [6] Fowler, Martin.2010. Domain-Specific Languages (Addison-Wesley Signature Series Fowler).Addison-Wesley Professional
- [7] DSM Forum. DSM Case Studies and Examples .(2011). Disponible en <http://www.dsmforum.org/>
- [8] EMF.Eclipse Foundation.(2000-2012). Disponible en <http://www.eclipse.com/emf/>.
- [9] GMF. Eclipse Foundation. (2000-2012). Disponible en <http://www.eclipse.com/gmf/>
- [10] Atzmon Hen-Tov, David H. Lorenz, Assaf Pinhasi, Lior Schachter,(2009) ModelTalk: When Everything Is a Domain-Specific Language, IEEE Software, vol. 26, no. 4, pp. 39-46.