

# Perfiles UML para definición de Patrones de Diseño

Ana Garis, Daniel Riesco, Germán Montejano  
Departamento de Informática, Universidad Nacional de San Luis  
San Luis, CP 5700, Argentina

## RESUMEN

Los Patrones de Diseño ayudan a los desarrolladores de software a resolver problemas de diseño orientado a objetos que repetidamente ocurren. Cuando estos adaptan patrones a soluciones particulares, muchas veces elaboran modelos usando UML. Sin embargo, a menudo, UML no es lo suficientemente expresivo para describir patrones apropiadamente.

Los perfiles UML (en inglés “UML Profiles”) permiten extender la sintaxis y la semántica UML para modelar elementos de dominios particulares.

Así como los perfiles UML extienden su vocabulario y los patrones de diseño definen para los diseñadores un vocabulario común; también es posible usar perfiles para definir un vocabulario de patrones en UML. De esta manera, los perfiles pueden ser usados no solamente para dominios específicos, sino también para resolver problemas particulares en diferentes dominios.

**Palabras Claves:** Perfiles UML, OCL, Patrones de Diseño, Programación Orientada a Objetos, Ingeniería de Software

## 1. ANTECEDENTES

El perfil es un mecanismo definido por UML (en inglés, Unified Modeling Language) [1] para extender y adaptar UML a una plataforma o dominio particular. Incluye tres elementos: *estereotipos*, *valores etiquetados* y *restricciones*. Los estereotipos extienden el vocabulario UML y es posible asociarle valores etiquetados y restricciones. Si bien, el lenguaje natural es usado para definir restricciones, debido a su precisión, OCL (por sus siglas en inglés, Object Constraint Language) es uno de los más adecuados.

Las bondades ofrecidas por los perfiles pueden ser aprovechadas en la transformación de modelos a otros modelos o a código (facilitando la tarea a los diseñadores de software). En los últimos años el concepto de perfil ha servido para cubrir distintos dominios; tales como, tiempo real, procesos de negocio, y otros.

Cuando los desarrolladores de software modelan soluciones usando UML, muchas veces encuentran que UML no es lo suficientemente expresivo para describir patrones apropiadamente.

Numerosos trabajos de investigación se ocuparon de la descripción de patrones con diferentes enfoques. Algunos trabajos estudiaron el uso de UML para definir y documentar patrones. Por ejemplo, Le Guennec, Sunyé y Jézéquel en [4] modificaron el metamodelo 1.3 para permitir colaboraciones en el meta-nivel y restricciones OCL, en lugar de colaboraciones parametrizadas. La idea era mejorar el modelo de colaboración y las plantillas UML para una mejor definición de los roles de los patrones.

La OMG en [9] introdujo la noción de “Paquete de Patrón de Negocio” (consideró que las reglas que gobiernan los conceptos de negocio pueden ser representados con un patrón con restricciones) para definir un patrón, y “Ligamiento de Patrón de Negocio” para aplicar un patrón.

Fontoura y Lucena en [5] extendieron UML representando una clase de patrones de diseño llamados “patrones de configuración”. La extensión a UML fue desarrollada expresando un diagrama abstracto que permite mostrar diferentes formas de implementación; y representando, además, un patrón de instanciación.

Una técnica de especificación para patrones fue propuesta por France, Kim y Song en [8]. Definieron un lenguaje de metamodelado (usando UML y restricciones OCL) para especificar perspectivas de patrones; tales como estructura estática e iteraciones. Una especificación de patrón define una familia de modelos UML en términos de roles.

Otros trabajos, como [6] y [7], estuvieron menos orientados a la especificación de patrones para documentación, pero mostraron otras interesantes características. Sanada y Adams en [6] extendieron UML para soportar patrones de diseño en diagramas de clases. Dang y Yang en [7] presentaron un perfil UML para una mejor visualización de patrones de diseño en diagramas de clases.

Barotto y Demonte en [10] propusieron una definición de patrones de diseño a través del metamodelo UML. Utilizaron estereotipos con restricciones OCL para especificar los elementos que componen un patrón. Sin embargo, no utilizaron la noción de “perfil UML” definido en UML 2.0 [3], sino conceptos anteriores a dicha versión.

## 2. EL USO DE PERFILES UML PARA DEFINICIÓN DE PATRONES DE DISEÑO

Nuestra línea de investigación esta orientada a la definición de *Patrones de Diseño con Perfiles UML*. Se analizan las ventajas de los perfiles para definir, documentar y visualizar patrones de diseño. Al definir un perfil por cada patrón se irá conformando una librería de patrones especificados con perfiles UML. Además se pretende mostrar que las herramientas UML son suficientes para introducir perfiles de patrones, sin implementar una herramienta específica para patrones.

### Resultados Parciales

Para comenzar se seleccionaron los patrones diseño denominados “estructurales” de [2]. La experiencia inicial consistió en la definición del patrón “Composite”.

Se definieron tres estereotipos de clase: “Component”, “Leaf” and “Composite”; y un estereotipo de asociación, “Children” (ver Figura 1).

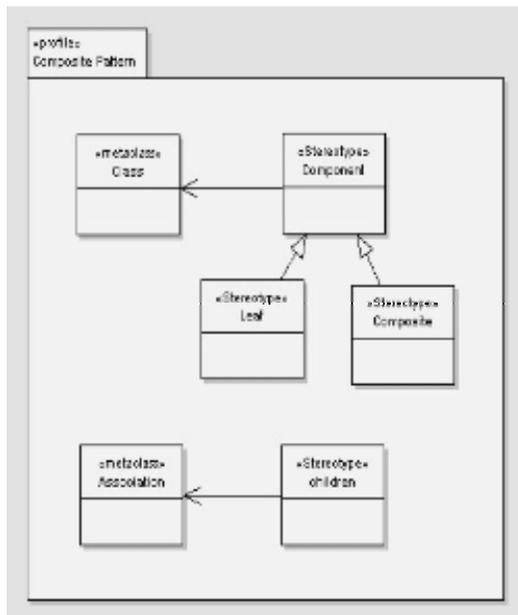


Figura 1: Definición de estereotipos del Patrón “Composite”

Fueron establecidas restricciones OCL sobre los estereotipos, la cuales son mostradas a continuación (el metamodelo UML es mostrado en la Figura 2 para entender mejor las especificaciones OCL).

### Children:(Core::Association)

inv:

```
self.connection->exists
(participant.isStereotyped (“Composite”) and
multiplicity.min=1 and
multiplicity.max=1) and
self.connection->forall (c1,c2|
c1.participant.isStereotyped (“Composite”)
```

and

```
c2.participant.isStereotyped (“Component”)
implies c1.aggregation=#composite and
c2.aggregation=#none )
```

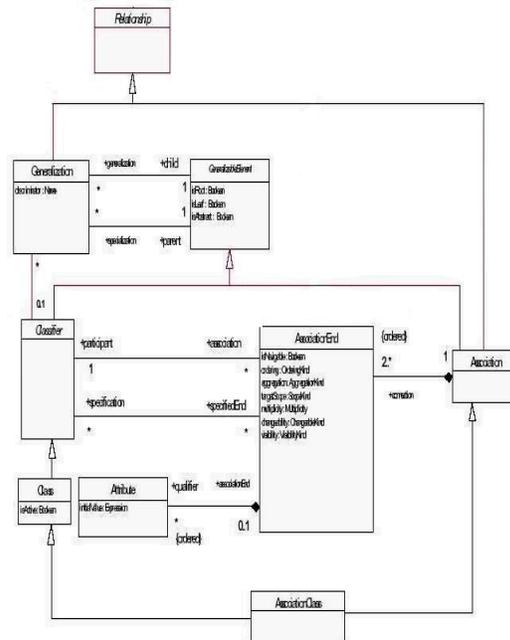


Figura 2: Core Package - Relationships

## 5. CONCLUSIONES Y TRABAJOS FUTUROS

Los perfiles UML pueden ser útiles para definir, documentar y visualizar patrones de diseño. Además cuentan con la ventaja de que si las herramientas UML permiten introducir perfiles de patrones, no es necesario implementar una herramienta específica para patrones.

El trabajo futuro esta dirigido a buscar información de herramientas UML que permitan definir perfiles UML. El análisis de las bondades de las diferentes herramientas, permitirá elaborar una propuesta general para la especificación de patrones usando herramientas UML.

Además se pretende mostrar que al usar perfiles para especificar patrones; el perfil esta siendo utilizado no solamente para definir un dominio específico (tal

como originariamente se pensó), sino también para definir un dominio general.

De la misma forma que con “Composite”, se realizará la especificación de otros patrones de diseño. Luego del análisis de las especificaciones resultantes, se podrá empezar a desarrollar una arquitectura para patrones usando perfiles UML.

## 6. REFERENCIAS

- [1] ISO/IEC, Unified Modeling Language (UML), Version 1.5, International Standard ISO/IEC 19501.
- [2] E. Gamma, R. Helm, R. Johnson, J. Vlissides, Design Patterns. Elements of Reusable Object-Oriented Software, Addison-Wesley. 1995.
- [3] UML 2.0, Infrastructure Specification, [http://www.omg.org/technology/ documents](http://www.omg.org/technology/documents), Agosto del 2004.
- [4] A. Le Guennec, G. Sunyé, J. Jézéquel, “Precise Modeling of Design Patterns”, In UML 2000, Vol. 1939 LNCS, 2000, pp. 482-496.
- [5] M. Fontoura, C. Lucena, “Extending UML to Improve the Representation of Design Patterns”, Journal of Object Technology, 2000.
- [6] Y. Sanada, R. Adams, “Representing Design Patterns and Frameworks in UML. Towards a Comprehensive Approach”, Journal of Object Technology, Vol.1(2), 2002, pp.143–154.
- [7] J. Dong, S. Yang, “Visualizing Design Patterns With A UML Profile”, IEEE Symposium on Human Centric Computing Languages and Environments (HCC 2003), 2003.
- [8] R. France, D. Kim, E. Song, “A UML-Based Pattern Specification Technique”, IEEE Transactions on Software Engineering, Vol.30(3), 2004, pp.193-206.
- [9] OMG, “UML Profile for Patterns Specification”, Parte del “UML Profile for Enterprise distributed Object. Computing (EDOC) Specification”, [http://www.omg.org/technology/ documents](http://www.omg.org/technology/documents), 2004.
- [10] V. Barotto, M. Demonte, D. Riesco. “Definición de Patrones de Diseño a través del metamodelo UML”. Tesis de Licenciatura en Ciencias de la Computación. Universidad Nacional de Río IV, Argentina. 2005.