

CC71P OBJETOS Y ASPECTOS

10 UD

Eric Tanter
Otoño 2006

Requisitos: CC51H

Objetivos:

Este curso se centra en la orientación a objetos (OOP) y su recién extensión, la programación por aspectos (AOP). El objetivo fundamental es entregar una visión clara del paradigma de OOP – incluyendo lenguajes OOP muy distintos de Java/C++/C# – y de los conceptos fundamentales que llevaron a la aparición de AOP, pasando por los temas de metaprogramación y arquitecturas reflexivas. Este curso plantea el contexto y la evolución de las ideas en el área que se encuentra al límite entre lenguajes de programación e ingeniería de software. Además provee una visión “de adentro” de los lenguajes OOP y AOP a través del estudio de sus intérpretes.

Temario:

- OOP
 - Diversidad de lenguajes OO
 - Lenguajes dinámicos con clases: Smalltalk (Squeak)
 - Lenguajes basados en prototipos: Self y ChitChat
 - Interpretación de lenguajes OO
- Metaprogramación y reflexión
 - Conceptos y evolución
 - Meta-arquitecturas de Smalltalk y Java
 - Extensiones reflexivas para Java: Reflex
- Programación por aspectos
 - Conceptos
 - Interpretación de lenguajes por aspectos
 - Herramientas actuales: AspectJ, Reflex
 - Composición de aspectos
 - Lenguajes (de aspectos) específicos a dominios (DSL)

Sistema de Evaluación:

La evaluación incluirá:

- Presentación/discusión de artículos
- Desarrollo de tareas de implementación
- Controles escritos de conocimiento

Referencias:

1. Tzilla Elrad, Robert E. Filman, and Atef Bader. Aspect-oriented programming. Communications of the ACM, 44(10), October 2001.
2. Adele Goldberg and David Robson. Smalltalk-80: The Language and its Implementation. Addison-Wesley, 1983.
3. Jan Hanneman and Gregor Kiczales. Design pattern implementation in Java and AspectJ. In Proceedings of OOPSLA 2002.
4. Gregor Kiczales, John Lamping, Anurag Mendhekar, Chris Maeda, Cristina V. Lopes, Jean-Marc Loingtier, and John Irwin. Aspect-oriented programming. Proceedings of ECOOP'97
5. Hidehiko Masuhara, Gregor Kiczales, and Chris Dutchyn. A compilation and optimization model for aspect-oriented programs. In G. Hedin, editor, Proceedings of CC 2003.
6. Fred Rivard. Smalltalk: a reflective language. In Gregor Kiczales, editor, Proceedings of Reflection'96).
7. Brian C. Smith. Reflection and semantics in a procedural language. Technical Report 272, MIT Laboratory of Computer Science, 1982.
8. David Ungar, Randall B. Smith. Self: The Power of Simplicity. Proceedings of OOPSLA'87
9. Chris Zimmermann. Advances in Object-Oriented Metalevel Architectures and Reflection. CRC Press, 1996.