

Aspect-Oriented Refactoring of Legacy Applications: An Evaluation

Dr. Sudipto Ghosh
Colorado State University

Date and Time: November 15, 2010 - 11:00am-12:00pm

Location: SEC 3437

Abstract:

The primary claimed benefits of aspect-oriented programming (AOP) are that it improves the understandability and maintainability of software applications by modularizing crosscutting concerns. Before there is widespread adoption of AOP, developers need further evidence of the actual benefits as well as costs. Applying AOP techniques to refactor legacy applications is one way to evaluate costs and benefits. We replace crosscutting concerns with aspects in three industrial applications to examine the effects on qualities that affect the maintainability of the applications. We study several revisions of each application, identifying crosscutting concerns in the initial revision, and also crosscutting concerns that are added in later revisions. Aspect-oriented refactoring reduced code size and improved both change locality and concern diffusion. Costs include the effort required for application refactoring and aspect creation, as well as a decrease in performance.

Biography:

Sudipto Ghosh is an Associate Professor of Computer Science at Colorado State University. He received the B.Tech degree from the Indian Institute of Technology, Kanpur, the M.S. degree from Iowa State University, and the Ph.D. degree from Purdue University. His teaching and research interests include modeling, designing and testing of object-oriented software, middleware technologies, and aspect-oriented and component-based software development. He is a member of the ACM and the IEEE Computer Society. He was a general co-chair of the ACM/IEEE 12th International Conference on Model Driven Engineering Languages and Systems in 2009. He was the program co-chair of the Third International Conference on Software Testing, Verification and Validation held in 2010. He is on the editorial boards of three journals: Information and Software Technology, Software Quality Journal, and the Journal of Software Testing, Verification, and Reliability.