

## **Adaptive and Evolvable Software Systems: Techniques, Tools, and Applications**

Jeff Gray

*Univ. of Alabama at Birmingham  
Dept. of Computer and  
Information Sciences  
Birmingham, Alabama USA  
gray(at)cis.uab.edu*

Raymond Klefstad

*Univ. of California, Irvine  
Dept. of Electrical Engineering  
and Computer Science  
Irvine, California USA  
klefstad(at)uci.edu*

Marjan Mernik

*University of Maribor  
Faculty of Electrical Engineering  
and Computer Science  
2000 Maribor, Slovenia  
marjan.mernik(at)uni-mb.si*

Mini-track URL: <http://www.cis.uab.edu/info/HICSS-AESS/>

A longstanding goal of software developers is to construct programs that are easily modified and extended. One example is the ideal that each change in a design decision would require only modifications to one module of a program. As demands for software increase, future requirements will necessitate new strategies for software development tools and techniques which better support adaptation and evolution.

Software's ability to adapt is actually partitioned among two stages: modifiability during development, and adaptation during execution. The first type of adaptation is concerned with design-time, or compile-time, techniques that permit the modification of the structure and function of a software representation in order to address changing stakeholder requirements. The first session of this mini-track contains papers that address this form of adaptation. Katrina Falkner et al., introduce work on separating policy mechanisms between components to support evolution. Hassan Gomaa and Diana Webber present four different approaches for modeling variability in product-lines.

A second type of adaptation occurs at run-time during the execution of the program. This type of adaptation refers to a system's ability to modify itself and to respond to changing conditions in its external environment. The second session of this mini-track begins with a paper by

Venkita Subramonian and Chris Gill, which illustrates generative techniques for configuring Quality of Service properties in componentized middleware. Fan Guochuang et al., highlight their work on using reflection to support evolution in web application servers. The contribution by Walter Cazzola and Dario Maggiorini describes work on middleware for mobile devices for adapting to location dependent services that are dynamically discovered. The paper by Carlos Varela, Kaoutar El Maghraoui and Travis Desell presents a framework and actor language for supporting autonomous load balancing.

The mini-track will appeal to those with interests in Generative Programming, Meta-programming and Reflection, Aspect-Oriented Software Development, Adaptive and Reflective Middleware, and Model-Driven approaches.

We received 17 full submissions. Six papers were accepted resulting in a 35% acceptance rate. The organizers would like to acknowledge the following individuals for their assistance in reviewing the submissions: Uwe Aßmann, Jean Bézivin, Fei Cao, Chia-Chu Chiang, Pascal Costanza, Aniruddha Gokhale, Ralf Lämmel, Seyed Masoud Sadjadi, Stan Sutton, Bedir Tekinerdogan, and Eric Wohlstadter.