The University of Alabama
Department of Computer Science
Colloquium Series Speaker

Ben Tyler
The Ohio State University

"Specifying and Monitoring Design Pattern Contracts"

Friday, February 25th
11:00 a.m., HO 108

Abstract:

Over the last decade, design patterns have fundamentally altered the way in which we think about large software systems. By providing descriptions of common solutions to problems that occur in varying contexts, design patterns act as a way to capture the collective wisdom and experience of the software community. Furthermore, they help software designers communicate their ideas in a more concise way by providing a common nomenclature for design solutions. Design patterns are usually described informally--and informal descriptions are essential--but they can also potentially lead to problems because different designers may have different interpretations of such descriptions. These differences can lead to systems whose behavior is not entirely predictable or understood well enough to allow for proper development, maintenance and evolution during its lifetime, undermining the overall reliability of the system. Precise descriptions of pattern behaviors can greatly reduce these problems by ensuring that design team members have a common understanding of how a pattern is being used, and how it is to behave within the system. Furthermore, precise descriptions give us a metric against which we can show whether the system is using the pattern correctly.

My current research addresses these concerns by focusing on ways in which we can formally specify design patterns using contracts. These pattern contracts provide requirements and constraints that must be followed when a designer uses a particular pattern, and specifies the resulting behavior that a properly implemented pattern will exhibit. In this talk I will present PCL, a Pattern Contract Language that is expressive enough to capture these essential requirements and the resulting expected behaviors of patterns. While PCL contracts are precise enough to resolve ambiguities present in most pattern descriptions, the formalism is tailored to preserve the flexibility of the pattern in question. My research has also addressed how we can build suitable software tools that allow us to show that such systems meet these contracts. In this talk, I will also present MonGen, a tool that generates monitoring code in the form of aspects (in the sense of aspect-oriented programming) from PCL contracts. I will illustrate these concepts and the use of PCL and MonGen with a simple case study.