

Information Science and Technology Institute Colloquium



Jonathan Aldrich
School of Computer Science
Carnegie Mellon University

“Access Permissions for Correct and Ubiquitous Concurrency”

Monday, November 8, 2010

10:00 - 11:00 AM

TA-3, Bldg. 1690, Room 102 (CNLS Conference Room)

Abstract: Leveraging concurrency has become essential for achieving adequate performance, whether in multicore chips, GPUs, or supercomputers. Given the complexity of modern software applications, however, supporting high developer productivity and good performance while avoiding race conditions and other concurrency-related errors has proved elusive. We propose Access Permissions as a way of capturing engineering design intent, including state transitions and aliasing among objects, and show how these permissions can be used to document concurrent abstractions, eliminate even high-level race conditions, and automatically parallelize applications. This approach allows developers to focus on domain issues and high-level design, while tools and the run-time system identify potential defects and extract greater levels of concurrency.

I will close by describing possible directions for future work, including permissions that support reasoning about locality in multiprocessor systems, verifying concurrent framework use (with Robert Bocchino), and parallel programming models based on neurobiological primitives (with Cyrus Omar).

Biography: Jonathan Aldrich is Associate Professor of Computer Science at Carnegie Mellon University. He is the director of CMU's undergraduate minor program in Software Engineering, and teaches courses in software engineering and program analysis for quality and security. Dr. Aldrich joined the CMU faculty after completing a Ph.D. at the University of Washington and a B.S. at Caltech.

Aldrich's research attacks software engineering problems using programming language techniques. Topics he has addressed include verifying architectural structure, modular formal reasoning about code, and API protocol safety. For his work on software architecture, Aldrich received a 2006 NSF CAREER award and the 2007 Dahl-Nygaard Junior Prize, given annually for a significant technical contribution to object-oriented programming. Aldrich's current work focuses on support for typestate-oriented programming and concurrency in the design of the Plaid language.