



Grace: Safe Multithreaded Programming for C/C++.

Prof. Emery Berger

University of Massachusetts, Amherst

Sala de Grados • 27 de mayo de 2009 • 15: 00

entrada libre hasta completar el aforo

resumen:

The shift from single to multiple core architectures means that programmers must write concurrent, multithreaded programs in order to increase application performance. Unfortunately, multithreaded applications are susceptible to numerous errors, including deadlocks, race conditions, atomicity violations, and order violations. These errors are notoriously difficult for programmers to debug.

This talk presents Grace, a software-only runtime system designed to enforce the correctness of a class of multithreaded programs: those based on fork-join parallelism. By turning threads into processes, leveraging virtual memory protection, and imposing a sequential commit protocol, Grace provides programmers with the appearance of deterministic, sequential execution, while taking advantage of available processing cores to run code concurrently and efficiently. Experimental results demonstrate Grace's effectiveness: with modest code changes across a suite of computationally-intensive benchmarks (1--16 lines), Grace can achieve high scalability and performance while enforcing correct execution.

sobre Emery Berger:

Emery Berger is an Associate Professor in the Department of Computer Science at the University of Massachusetts, Amherst.

He graduated with a Ph.D. in Computer Science from the University of Texas at Austin in 2002. His research spans programming languages, runtime systems, and operating systems, with a particular focus on systems that transparently improve reliability and performance. He is the creator of various widely-used software systems including Hoard, a fast and scalable memory manager that accelerates multithreaded applications. His honors include a Microsoft Research Fellowship (2001), an NSF CAREER Award (2003), a Lilly Teaching Fellowship (2006), and a Best Paper Award at FAST 2007. Professor Berger currently serves as an Associate Editor of the ACM Transactions on Programming Languages and Systems.