



Andrew Gill

University of Kansas

Friday, March 24, 2017

3:00 PM

Lindley Hall, Rm. 102

Remote Procedure Call bundling using Monads and Applicative Functors

Abstract: Monads and applicative functors are two ways Haskell programmers use to bundle effectful primitives into effectful program fragments. In this talk, we investigate using monads and applicative functors to bundle remote effectful primitives, specifically for amortizing the cost of remote communications. We look at three ways of maximizing the bundling of primitives, drawing from the remote monad design pattern and Facebook's Haxl system, and provide a taxonomy of mechanisms for amortization, with examples. The result of this research is that monadic fragments can be efficiently bundled into packets, almost for free, when given a user-supplied packet transportation mechanism, and the primitives obey some simple pre- and post-conditions.

Biography: Andrew (Andy) Gill is an Associate Professor of Electrical Engineering and Computer Science at the University of Kansas, and holds the John E. and Winifred E. Sharp Teaching Professorship. He was born and educated in Scotland, and has spent his professional career in the United States. Gill received his Ph.D. from the University of Glasgow in 1996, then spent three years in industry as a compiler developer (Hewlett Packard, Metrowerks, Semantic Designs), and a year in academia as a principal project scientist (Oregon Graduate Institute). In 2000, he co-founded Galois, a technology transfer company that used language technologies to create trustworthiness in critical systems. In 2008 he returned to academia and research, joining the University of Kansas and the Information and Telecommunication Technology Center based in Lawrence, KS.

Since returning to academia, Gill has targeted the application areas of telemetry and signal processing, specializing in generating high performance circuits from specifications. His research interests include optimization, domain-specific languages, language design, debugging, and dependability. The long-term goal of his research is to offer engineers and practitioners the opportunity to write clear and high-level executable specifications that can realistically be compiled into efficient implementations.



**SCHOOL OF INFORMATICS
AND COMPUTING**

INDIANA UNIVERSITY

Bloomington