

Adjoint School, ACT2019

Mark Hopkins

Relevant background

I have a basic understanding of, and am reasonable comfortable with, the content of MacLane (though no doubt a little rusty). I gave a conference presentation last year to programmers about the basic idea of and inter-expressibility of the different universal constructions, from initial object to Kan extension. I've had some exposure to higher categories and topos theory, though not in any great depth. My PhD work touched on braided categories, though unfortunately this didn't make it into the thesis.

PhD

I completed my PhD on quantum groups in 2007. The main result is that we recover a particular quantum group (the q -Yangian) as a subalgebra inside a limit of centralizers of matrix algebras.

Order of project preference

- Simplifying quantum circuits using the ZX-calculus
- Traversal optics and profunctors
- Partial evaluations, the bar construction, and second-order stochastic dominance
- Toward a mathematical foundation for autopoiesis
- Complexity classes, computation, and Turing categories
- Formal and experimental methods to reason about dialogue and discourse using categorical models of vector spaces

Attendance

Barring any sudden crisis at work, I should have no issues in coming to Oxford.

Motivation

I work as a functional programmer in Haskell (or a dialect thereof). I use lenses and have been interested in the question of how they can be specified categorically since I first met them. So I'm quite drawn to the profunctor optics topic.

Outside of work, I've been reading Coecke and Kissinger's "Picturing Quantum Processes", which I'm finding really fascinating. This is the main reason for my interest in the quantum circuits project.

Career-wise, I'm interested in participating in areas where computation and mathematics (particularly algebra or category theory) meet. Having the chance to participate in the school would be an enormous boost towards developing the right skills to do so meaningfully.

Mark Hopkins

Mobile	0451 158 602
Email	markjohnhopkins@gmail.com
Web	mjhopkins.github.io,
Github	github.com/mjhopkins

Personal Profile

Quality-focused software developer who enjoys the challenge of deeply understanding problem domains and crafting clear, elegant, resilient solutions. Passionate about functional programming and type theory.

Education

2000-2007	PhD (Mathematics) University of Sydney <i>Quantum affine algebras: quantum Sylvester theorem, skew modules and centralizer construction</i>
1996-2000	BSc (Pure Mathematics), 1st Class Honours University of Sydney

Employment History

2018 - Present	Digital Asset <i>Software Engineer</i> Digital Asset provides a permissioned digital ledger and associated smart contract language (DAML, a variant of Haskell). The company has teams devoted to language and compiler engineering, the ledger, formal verification, as well as application code. My work involves both writing smart contracts in DAML as well as supporting Scala code.
2014 - 2018	Commonwealth Bank of Australia <i>Senior Software Engineer</i> I worked on a number of projects: <ul style="list-style-type: none">• Tooling to provision and manage Kubernetes resources (Scala).• A HipChat chatbot that retrieves and updates alert data from alerta (Haskell).• A store for machine learning feature data backed by HBase (Scala).• A system for running repeatable, cached machine learning pipelines (Haskell).• An analytics toolset for event data stored in Hadoop, providing operations such as windowed join (Scala).
2013 - 2014	Packetloop/Arbor Networks <i>Senior Software Developer</i>

Big data security analytics: processing streamed packet capture data to discover attacks, which were analysed and fed to a live UI.

- Improved performance by fine-tuning sketch implementations
- Introduced Kafka
- Ported existing codebase from Java/Ruby to clean, functional Scala

Technologies: Storm, Kafka, Elasticsearch, Cassandra, scalaz, probabilistic data structures

**2008 -
2013**

Tyro Payments
Software Developer

Tyro is a small bank providing EFTPOS services to merchants. I helped develop and maintain their Java codebase.

- Automated, extensible solution for loading data from large MasterCard data files.
- Created a library for writing web application tests.
- Wrote a tool that tests backwards compatibility of database migrations.
- Designed and developed a tool for managing upgrades to a fleet of terminals (still in use).
- Helped add EMV support to terminal software (a large finite state machine)
- Key management using DUKPT and hardware HSMs.
- Financial message processing (ISO 8583)

Talks

■ All of Basic Category Theory

LambdaJam 2018, May 2018

A top-down view of the major universal constructions in category theory, from initial objects to Kan extensions.

■ Conditional Contexts

ScalaSyd, November 2017

Porting Haskell's ifcxt to Scala.

■ Monadic matching mishaps

ScalaSyd, July 2017

An examination of pattern matching within monadic expressions in Haskell, Idris, Purescript and Scala.

Please see mjhoppkins.github.io for details of earlier talks.

Other

I organise and help run the following:

- Sydney Paper Club – a monthly meetup where we read and discuss a computer science paper
- Sydney Type Theory – a weekly type theory reading group (currently reading the Homotopy Type Theory book)
- a weekly “Haskell Guild” meeting at work