

# Timothy A. K. Zakian

## CONTACT INFORMATION

Timothy Zakian  
Keble College, Parks Road  
Oxford OX1 3PG, United Kingdom

Phone: (+44)7572 661295 (UK)  
Email: [timzakian@gmail.com](mailto:timzakian@gmail.com)  
WWW: [cs.ox.ac.uk/people/timothy.zakian](http://cs.ox.ac.uk/people/timothy.zakian)  
Citizenship: U.S. Citizen

## EDUCATION

- University of Oxford** 2015-Present  
DPhil (PhD) Student in Computer Science
- Indiana University – Bloomington** 2011-2015  
Bachelor of Science with Highest Distinction (GPA: 3.98/4.0, 4.0 in major)
- Major: Computer Science with a Specialization in Programming Languages
  - Minor: Mathematics

## PUBLICATIONS

- T. L. McDonell, **T. A. K. Zakian**, M. Cimini, and R. R. Newton. Ghostbuster: A Tool for Simplifying and Converting GADTs. In *Proc. of the 21st International Conference on Functional Programming (ICFP 2016)*. [Link](#)
- C.S. Zakian, **T.A.K. Zakian**, A. Kulkarni, B. Chamith, and R.R. Newton. Concurrent Cilk: Lazy Promotion from Tasks to Threads in C/C++. In *Proc. of the 28th International Workshop on Languages and Compilers for Parallel Computing (LCPC 2015)*. [Link](#)

## EMPLOYMENT

### Cray Inc.

- Intern – Chapel Compiler Group May 2014 – Aug. 2014  
I extended my work from my previous internship, adding support for exposing and taking advantage of parallel and distributed file systems in Chapel. I also added support for a “pyCurl” type interface in Chapel and worked on the specification and implementation of various file plugins for the Chapel language.
- Intern - Chapel Compiler Group May 2013 - Aug. 2013  
In this internship:
  - I added support for in-memory and distributed MapReduce-type operations in Chapel.
  - I added an I/O interface to the Hadoop Distributed File System.
  - I added the ability to easily extend the runtime I/O system in Chapel to allow easy interfacing with, and use of other types of filesystems in Chapel.
  - I created a generic record reader/parser for the I/O interface in Chapel.
  - I tested, evaluated and gave feedback on features in the Chapel compiler - Specifically as it applies to the Foreign Function Interface, Regex support, LLVM support and parallel I/O features.

### Indiana University

- Undergraduate Instructor – Compilers Jan. 2014 – May 2014  
I ported the previous course infrastructure to a new infrastructure and created a grading framework, language definitions, and solutions for each assignment (18 assignments, ~5000 source lines of code). I held regular office hours, weekly code reviews, maintained course infrastructure, and helped grade homework and exams. I also filled in where necessary to ensure a successful course.
- Undergraduate Instructor - Discrete Math for CS Jan. 2014-May 2014  
I held regular office hours, graded homework, and helped lead discussion sections for the class. In this capacity, I also helped come up with problems for homework sets.
- Undergraduate Instructor - Intro to Programming Languages Spring and Fall 2013  
I provided feedback to the professor and suggestions for new problems for both homework and exams as well as course related infrastructure. I also graded weekly homework and exams, held regular office hours, and led weekly lab sections.

## AWARDS & HONORS

### Clarendon Scholarship

Spring 2015

Awarded full international scholarship to pursue a DPhil at the University of Oxford, awarded to fewer than 2% of all admitted students.

### Internal Wells Scholarship

Spring 2014

Sole recipient of the Internal Herman B Wells Scholarship for 2014. Finalists and recipients are selected from faculty nominations from the top 1% of all undergraduates at Indiana University.

### Marie S. Wilcox Scholarship in Mathematics

Spring 2013

Awarded to students who demonstrate a deep understanding and appreciation for Math, and who maintain a

record of high academic achievement. Nominated by both faculty members whose classes I took in 2012-2013 academic year

**Thelma Abell Prize in Mathematics**

**Spring 2012**

Awarded annually to outstanding undergraduates with high academic potential. Nominated by all four faculty members whose classes I took in 2011-2012 academic year.

**First Place, Indiana Collegiate Mathematics Competition**

**Spring 2012**

Out of twenty-eight teams.

**Scholarship Recipient, Supercomputing**

**Fall 2014**

(Experiencing HPC for Undergraduates) One of 26 undergraduates chosen to receive scholarship to attend SC '14

**Scholarship Recipient, (PLMW/POPL)**

**Spring 2014**

**Founders Scholar**

**2011-2012;2012-2013;2013-2014;2014-2015**

**Deans List, School of Informatics and Computing**

**2014;2015**

**Scholar of Highest Distinction**

**Fall 2011; Spring 2012**

**Mcammom OEM Scholarship**

**2011-2015**

**Indiana University Prestige Scholarship**

**2011 - 2015**

**PROJECTS**

- **Wrote two fully functional compilers from a subset of Scheme to x86 assembly.** This included parsing, defunctionalization, closure conversion, constant-folding and dead-code elimination, instruction-selection, register-allocation, code-generation and more. [Link](#)
- **Wrote an asteroids-type game for the STM32-F4 microcontroller.** This involved optimizing graphics, interrupt-handling for concurrent audio-visual playing (IRQs), input from physical devices (Wii nunchucks), and GPIO. [Link](#)
- **Helped write an embedded traffic counter for the Indiana DNR** as part of a class project. This was then deployed at McCormick's Creek State Park. [News article](#).
- **Wrote a probabilistic programming language in Scheme** incorporating a novel inference technique that combines Hamiltonian Monte-Carlo methods with standard MCMC inference. [Link](#)
- **Wrote a parallel implementation of the Karatusuba algorithm** in Haskell utilizing SIMD and multiple data-parallel libraries and constructs. [Link](#)

**SOFTWARE SKILLS AND GRADUATE CLASSES TAKEN**

- **Languages** (in order of familiarity): Haskell, C/C++, Scheme, Chapel, Java, OCaml, Scala
- **Operating Systems and Tools:** Linux, Unix (mac), ChibiOS, Vim, Git, Emacs, IntelliJ IDEA
- **Graduate Classes Taken:** Concurrency; Categories, Proofs, and Processes; Metaprogramming; Parallelism in Functional Programming Languages; Dependent and Homotopic Type Theory; Probabilistic Programming Languages; Programming Language Foundations; Topology; Introduction to Programming Languages; Compilers; Analysis I/II; Algebra I/II

**Misc**

Authorized to work in the U.S. without restrictions. (U.S. Citizen)

Biked across the U.S. in the summers of 2011 and 2012