

# Dr. Edd Barrett

Web: [eddbarrett.co.uk](http://eddbarrett.co.uk)

Email: [edd@theunixzoo.co.uk](mailto:edd@theunixzoo.co.uk)

Telephone: (on request)

Address: (on request)

An experienced, confident and open-minded programmer and computer scientist.

## Education

---

### 2014 PHD IN COMPUTER SCIENCE FROM UNIVERSITY OF KENT

Thesis titled “Range Analysis of Binaries with Decision Procedures” investigates the applicability of the Boolean satisfiability problem and linear programming to static program analysis and reverse engineering of low-level binary code.

### 2009 BSC HONS. COMPUTING FROM BOURNEMOUTH UNIVERSITY

First class with honours. Dissertation titled “3c: A JIT Compiler with LLVM” describes the implementation of a JIT for a dynamically typed object-oriented programming language using LLVM.

### 2002 A- AND AS-LEVELS FROM QUEEN MARY’S COLLEGE, BASINGTOSKE

Including Computing, Physics, Electronics and Mathematics.

## Employment

---

### 2013– RESEARCH ASSOCIATE, KING’S COLLEGE LONDON

Research on programming language composition. Our team implements and evaluates meta-tracing programming language VMs which can execute a mix of programming languages and fast.

### 2009–2013 PART TIME TEACHER AT UNIVERSITY OF KENT

Teaching C, UNIX, logic programming and Java at undergraduate/masters level.

### 2009–2013 OUTREACH PROJECT AT UNIVERSITY OF KENT

Teaching programming fundamentals to schools (age 13-16) using Lego robotics.

### 2007–2008 BOURNEMOUTH UNIVERSITY

Placement year as a part of undergraduate studies. UNIX systems administration (Solaris, Linux, OpenBSD) and in-house development.

### 2003–2005 TECTONICS LTD.

Maintenance of legacy BASIC stock management system, re-implementation of existing stock management system (web-based), programming industrial CNC routers for kitchen manufacturing, systems administration and IT support.

## Skills

---

**Programming Languages** I am an experienced programmer. I am currently most fluent in Python, C, Java, PHP and Prolog. I have toyed with OCaml, Lua, Ruby, PHP, Javascript etc. I have worked on implementations (VMs) for a handful of different languages including Python, PHP and Prolog. I can learn new programming languages and paradigms on demand.

**Software Engineering** I believe that correct software engineering practice can help deliver better quality software. I use version control (git, Subversion, Mercurial, CVS) on a daily basis. When I encounter software defects, I always file a bug upstream. I have experience with test-driven development.

**Computer Science** The nature of my PhD was theoretical. I have a solid understanding of abstract interpretation, optimisation problems and propositional logic. I understand the scientific method, especially with regards to performance benchmarking. I have taught computer science topics at undergraduate and masters level.

**Systems Administration** I have many years of experience with UNIX systems. Aside from maintaining my own personal systems (including desktops, web and mail servers), I have also had experience administering UNIX systems in a University environment.

**Organisation** I have worked in a small team to organise Barcamp Canterbury 2012–2014. I was the publicity chair for the Dynamic Languages Symposium in 2014. During my undergraduate degree I co-founded a monthly-meeting UNIX user group. Each summer I work voluntarily on folk festival task forces and stewarding teams.

## Software Contributions

---

**OpenBSD project** Developer for OpenBSD (an open-source UNIX distribution) since October 2009. I work mostly on the ports tree, which involves patching and packaging third party software. I've worked a little in the kernel on touch-screen drivers (`ums(4)`) and also in userland on `vmd(8)` and `top(1)`.

**Unipycation** A composed RPython VM capable of executing a mix of Python and Prolog code, allowing (fast) logic programming in Python programs. I also implemented the same composition upon the JVM and upon C for comparative purposes.

**PyHyp** A composed RPython VM capable of executing a mix of Python and PHP code. The approach is the same as for Unipycation, but PyHyp was designed to have a higher degree of language interoperability. For example, cross-language scoping is possible under this VM.

**Dgen/SDL project** Implemented (for fun) a debugger for an open-source games console emulator written in C/C++. This allowed me to reverse engineer the Motorola m68k code for a Sega game.

## Selected Publications

---

- E. Barrett, C. F. Bolz, L. Diekmann, L. Tratt. Fine-grained Language Composition: A Case Study. *European Conference on Object-Oriented Programming (ECOOP)*, July 2016.
- E. Barrett, C. F. Bolz, L. Tratt. Approaches to Interpreter Composition. *Computer Languages, Systems, and Structures*.
- E. Barrett, C. F. Bolz, L. Tratt. Unipycation: A Case Study in Cross-language Tracing. *Virtual Machines and Intermediate Languages (VMIL)*, October 2013.
- E. Barrett. Range Analysis of Binaries with Decision Procedures. *PhD thesis.*, March 2014.
- E. Barrett, A. M. King. Range Analysis of Binaries with Minimal Effort. *Formal Methods in Critical Systems (FMICS)*, August 2012.

## Interests

---

My research interests include: reverse engineering, programming language design/implementation, benchmarking, program analysis, abstract interpretation, mathematical optimisation and Boolean satisfiability.

I am a (casual) electronics hobbyist, and tinker with microcontrollers (e.g. Arduino). I enjoy reading sci-fi and sometimes make home brew. I also enjoy walking and camping in the great outdoors. I go rock climbing once a week. I repair and collect old microcomputers and games consoles.

## References

---

Dr. Andy King (PhD. Supervisor)  
School of Computing  
Cornwallis Buildings  
University of Kent  
Canterbury  
England  
CT2 7NZ  
Tel: +44 (0)1227 827911  
a.m.king@kent.ac.uk

Laurence Tratt (PI for Post Doc.)  
S1.01, Strand Building,  
Strand Campus,  
King's College London,  
Strand, London,  
England,  
WC2R 2LS  
Tel: +44 (0)02078 487034  
laurence.tratt@kcl.ac.uk