

# MICHAEL WALKER

## Experience

---

### GoCARDLESS

#### Software Development Engineer

Mar 2022–now

- Designed and led implementation of a new **event sourcing** approach to our core payment processing logic.
- Worked on improving scalability of a key part of our API:
  - Designed a **table partitioning** strategy to remove expensive queries, reducing peak API latency by 95%.
  - Worked with product, support, and marketing colleagues to co-ordinate a breaking change to the availability of historic data over the API with users.
- Assisted with incident triage and response, including contributing to a working group to improve our practices.

### GOVERNMENT DIGITAL SERVICE

#### Senior Developer

Jun 2019–Feb 2022

- Tech lead on a small multidisciplinary **agile** team doing discovery work into GOV.UK Accounts and Personalisation, where I:
  - Worked with product and delivery managers to prioritise and delegate work.
  - Implemented authentication and authorisation with **OAuth / OpenID Connect**.
  - Established good practices like **continuous deployment** and **service level objectives**.
  - Proposed and built consensus for large changes to the GOV.UK stack.
  - Worked with architects and our CDN provider to plan how to transition GOV.UK from a mostly static and very cacheable website to being much more dynamic.
- Prototyped and then productionised a machine learning pipeline for search result ranking using **Amazon SageMaker**.
- Lead an upgrade from **Elasticsearch 5** to **Elasticsearch 6**.
- Worked with external pentesters, assisting their work and triaging issues.
- Worked with performance analysts to plan and implement **A/B tests**.
- Line managed a mid-level developer.

#### Developer

Apr 2018–Jun 2019

- Made various improvements to the GOV.UK stack: for performance, technical debt, and architecture debt. Mostly **Ruby** and **Rails** or **Sinatra**, some **Python**, various types of database (e.g. **PostgreSQL** and **MySQL**). All running on **Linux**.
- Lead initial experiments into **load testing**.
- Planned and implemented an upgrade from **Elasticsearch 2** to **Elasticsearch 5**.
- Worked with **AWS** infrastructure using **Terraform** and **Puppet**.
- Gave regular support to teams which did not merit a full-time developer.

### OVERLEAF

#### Software Engineer (Part Time)

Jan–Mar 2018

- Maintained legacy **Rails** and **Java / JGit** services during the merger with ShareLaTeX.

#### Software Engineering Intern

Jul–Sep 2017

- Fixed security issues, ranging from **CSRF** and **XSS** vulnerabilities to a bug in Heroku's router.
- Ported a large **Rails 4** application to **Rails 5**.
- Designed and implemented a distributed message bus using **Node** and **Redis**.

London — England

[mike@barrucadu.co.uk](mailto:mike@barrucadu.co.uk) • [www.barrucadu.co.uk](http://www.barrucadu.co.uk)

github: [barrucadu](#) • linkedin: [barrucadu](#) • twitter: [barrucadu](#)

## **PUSHER**

Software Engineering Intern

May–Aug 2016

- Contributed to the productionisation of a prototype low-latency distributed message bus using **Go** and **Raft consensus**.

## **COREFILING**

Software Engineering Intern

Jul–Sep 2014

- Refactored a **Java** in-house wiki program, fixing numerous long-standing bugs.

## **Education**

---

### **UNIVERSITY OF YORK**

Ph.D in Computer Science

2014–2019

My thesis “Revealing Behaviours of Concurrent Functional Programs by Systematic Testing” examined the deterministic testing of concurrent programs with shared memory and message passing. Supervised by Colin Runciman, and examined by Simon Peyton Jones and Ana Cavalcanti.

M.Eng in Computer Systems and Software Engineering

2010–2014

My dissertation on the formal verification of stop-the-world garbage collectors received first-class honours. Overall, I achieved a 2:1.

## **Publications**

---

### **Revealing Behaviours of Concurrent Functional Programs by Systematic Testing**

Ph.D thesis. University of York, 2018.

### **Cheap Remarks about Concurrent Programs**

Michael Walker and Colin Runciman, 2018. In *ACM SIGPLAN Symposium on Functional and Logic Programming*.

doi: [10.1007/978-3-319-90686-7\\_17](https://doi.org/10.1007/978-3-319-90686-7_17)

### **Déjà Fu: A Concurrency Testing Library for Haskell**

Michael Walker and Colin Runciman, 2015. In *ACM SIGPLAN Symposium on Haskell*.

doi: [10.1145/2887747.2804306](https://doi.org/10.1145/2887747.2804306)

*PDFs and BibTeX entries are available on [my website](#).*

## **Open Source**

---

Déjà Fu

2015–now

A library for testing concurrent **Haskell** programs, developed as part of my Ph.D thesis. It has had some commercial users.

github: [barrucadu/dejafu](https://github.com/barrucadu/dejafu)

London — England

[mike@barrucadu.co.uk](mailto:mike@barrucadu.co.uk) • [www.barrucadu.co.uk](http://www.barrucadu.co.uk)

github: [barrucadu](https://github.com/barrucadu) • linkedin: [barrucadu](https://www.linkedin.com/company/barrucadu) • twitter: [barrucadu](https://twitter.com/barrucadu)