

Reid Douglas McKenzie

Cell: +1 (908) 514-0232

Email: me@arrdem.com

Blog: arrdem.com

GitHub, Twitter, IRC: arrdem

ABOUT

I am a software engineer with a soft spot for research looking for opportunities to scale my impact with mentorship and architecture responsibilities on growing teams.

WORK HISTORY

Eng. #1 **December 2021 - Present**
Divebell Inc.

Full-stack engineer on a data protection product; from core feature development to infrastructure management, UI development, feasibility study and customer support.

Sr. Software Engineer **January 2019 - September 2021**
Twitter Inc., Core Infrastructure Provisioning Automation

- **Cloud migration planning** - Drove a cloud migration efforts for Twitter's server fleet source of truth, enabling expansion of Twitter serving into new cloud regions. Identified extensive dependencies within legacy stacks as blockers to cloud migrations, necessitating decoupling and re-alignment with cloud patterns in the legacy stack to achieve desirable outcomes.
- **Kubernetes enablement** - Facilitated adoption of Kubernetes on-prem by reducing time-to-provision from weeks to in minutes by delivering a Metal³ Kubernetes Cluster API provider targeting the legacy datacenter management stack.
- **Third datacenter buildout** - Worked to plan & manage risk in the deployment of Twitter's third data center. Developed, socialized and drove a backhauling plan for bootstrap and validated it on foundational services to replace untested bring-up procedures. Delivered foundational capabilities with minimal engineering effort months ahead of schedule.
- **Disaster management** - Responded to Twitter's first whole compute platform outage, involving the loss of all services in a facility. Lead the authoring of a facility bootstrapping plan, involving ordered service bringup spanning 11 peer teams. Enabled recovery of the facility and restoration of traffic over the next 36 hours, rather than the weeks feared in the early hours of the outage. The plan has become the basis of recovery plans for Twitter's planned outage testing.

Software Engineer **December 2017 - January 2019**
Funding Circle Inc., Accounting Engineering Team

- **Ledger reliability testing** - Demonstrated that Kafka and Funding Circle's prototype ledger had the promised reliability and consistency characteristics leveraging the [jepsen](#) distributed systems test framework.
- **Ledger traffic testing** - Worked with business partners to model traffic from core business workloads and estimate predict future demands. Developed traffic generation based on this model, and a benchmarking tool specialized to queued system dynamics. Conclusively demonstrated that the candidate ledger could not sustain production objectives as originally designed.
- **Ledger re-architecture** - Defined, implement and evaluate a replacement ledger architecture which could meet business production requirements by adopting, heavily evaluating and installing backstops for Kafka 1.0's "exactly once" message processing semantics. Leveraged the performance testing tools to show that the transactional ledger exceeded business requirements handily.
- **Ledger migration** - Worked to develop a deployment plan for executing and validating a one-time cutover over from the previous ledger system to the new accounting stack. Successfully executed the migration, delivering on Funding Circle's three year business objective of decommissioning the legacy ledger system.

Site Reliability Engineer II **May 2016 - November 2017**
Twitter Inc., Core Infrastructure Services SRE

- **Durable functions** - Worked to reduce fleet management toil and human intervention to recover failed automation by designing, developing & deploying a durable functions engine specialized to capturing error conditions and describing complex retrying. Successfully unified three legacy systems and increased the reliability of key fleet operations from sub-50% to 95% using this tool.
- **Fleet allocation** - Partnered with the asset management group to improve server fleet utilization; deploying an auditable requirements based server allocator to replace slow, error-prone spreadsheet based workflows.
- **Core operations** - Participated development and oncall support for tier-0 services including Puppet, DNS, Kerberos, NTP and other infrastructure.

Software Engineering Intern **May - August 2015**
Factual Inc., Geopulse Audience Team

- Developed, debugged & ran two new customer data evaluations
- Documented & validated a prototype Apache Solr location search system
- Explored [statically typed Clojure compilation](#), improving performance 5-20%

Independent Contractor **May - August 2014**
Google Inc., for Clojure Google Summer of Code

- Designed & implemented [an ahead of time compiler](#) for Clojure
- Implemented static use set & reach set analysis, tree shaking & λ lifting
- Produced a 15% speedup on “typical” programs & 24% on benchmarks

Software Development Intern **June - August, December 2013**
Calxeda Inc. SOCMAN embedded OS development team

- Developed & implemented a new system-wide logging mechanism
- Implemented a timing-sensitive PHI lane driver workaround
- Refactored an internal OpenIPMI fork to resolve & upstream changes

EDUCATION **Bachelor of Science & Arts** Univ. Texas at Austin **May 2016**
Majored in Computer Science through the Turing Scholars honors program.

- PUBLICATIONS** [“Lightweight Tool Coordination”](#), ICSE 2013, TOPI workshop
- Compared historical models of version control systems
 - Presented a model of software development in terms of path expressions
 - Showed that each VCS model could be written as a path expr or CFG
 - Argued for the use of path expressions to enforce project stability
 - Presented a use case of applying testing requirements to the git-fow model

- SELECTED PROJECTS**
- arrdem.datalog**, <https://pypi.org/project/arrdem.datalog/>
- Full-up Python implementation of naive recursive datalog
 - Pluggable storage interface to enable experimentation with caching strategies
 - ‘Easy’ API attempts to hide internal implementation concerns from users
 - Formally defined parser grammar enables load/store of human noted datasets
 - Features an [interactive shell](#)

- source**, <https://git.arrdem.com/arrdem/source>
- Customized Bazel monorepo to encapsulate my Python projects
 - Enabled unified testing, integration and toolchain development
 - Considerably accelerated iteration on datalog and other projects
 - Lead to the development of a custom zipapp tool, [Zapp!](#)

- Homelab**,
- Designed, deployed & automated a home “datacenter”
 - Leveraged Ansible to manage configuration & deployment
 - Developed bare-metal provisioning automation
 - Migrated <https://www.arrdem.com> and other properties to on-prem