Caner Derici

Software Engineer II – PhD Candidate (abd), Computer Science

dericilab.live @ caner@dericilab.live canerderici cderici extended (CV) UT, USA

Technical Skills

Areas of expertise: Distributed Systems, Compilers & Programming Languages, Machine Learning

Languages: Go, Python, C/C++, Racket/Scheme, Java, SQL/NoSQL

Cloud: Kubernetes, AWS, GCE, Terraform, LXD, Docker

General: REST, gRPC, DQLite, MongoDB, PostgreSQL, Git, CI/CD, GitHub Actions, Jenkins

Experience

Canonical USA — Remote, US

Software Engineer II (L4), Enterprise Cloud Engineering, Juju team. ***2021 – 2024***

Distributed orchestration for large scale cloud workloads. Primarily in Go, and Python. See Juju below for details.

Indiana University — IN, US

Course Instructor, Teaching & Research Assistant ***2015 – 2021***

I taught compilers, principles of programming languages, domain specific languages, and data structures & algorithms. Researched runtime performance of JIT compiled VMs for functional languages.

Asseco SEE Group

Software Engineer ***2012-2013***

I developed and tested virtual point-of-sale applications in Java. Used Tomcat, Spring, Mercurial, Jira.

Selected Projects

Juju

A large scale distributed orchestration engine for managing cloud workloads on any infrastructure (Kubernetes or otherwise) across various cloud providers (e.g., AWS, GCE). Juju is written in Go, and has more than a million lines of code. I was a core maintainer in a team of 14 engineers.

- I architected full-stack distributed components, tackled reliability, fault tolerance, back-pressure handling on the eventually consistent back-end.
- I doubled the user base of client libraries python-libjuju, terraform juju provider).
- I helped transition the data model from NoSQL MongoDB to relational DQLite (e.g., a sample PR).
- I helped redesign the facade-based RPC API. I also developed a REST API as an alternative with OpenAPI.
- I owned deliverables, maintained release cadence, participated in roadmap planning, coordinated cross-team work, mentored junior engineers, and took part in hiring.

Terraform Juju Provider

A Terraform provider that enables integration with Juju while managing Terraform environments. I implemented new resources and features (e.g., manual provisioning on AWS), migrated the provider from the sdk2 to the provider framework (e.g., sample PR), and maintained release cadence of new versions. All in Go.

Pycket: A meta-tracing JIT compiler for self-hosting Racket

PhD thesis project. I developed and maintained Pycket for more than five years. I designed the compiler to bootstrap the whole Racket language on a meta-tracing JIT compiler back-end. I helped design a new IR (linklets, see publications) to make Racket run-time more portable. I developed performance analysis tools and formalisms to improve performance and reusability of the meta-traces in the JIT. I implemented run-time optimizations, data structures and run-time primitives.

Rax: A full-stack Racket to x86_64 nanopass compiler

I implemented all the passes (e.g., closure conversion, register allocation, code-gen, etc.), along with garbage collection. I developed optimizations, such as inlining, loop-invariant code motion, and proper tail-calls.

HazirCevap (Witty): A closed domain question answering system for high school students

Government funded large scale question answering system. MSc thesis on NLP and Machine Learning. I led R&D team (3 faculties, 4 grad students). I developed a Hidden Markov random field model for question analysis, and relevance metrics for information retrieval and response generation (see publications). Full stack in Python.

Education

PhD (abd), Indiana University, Computer Science, Programming Languages **2015 – 2021**

Optimizing VM run-times for dynamic languages on a meta-tracing JIT compiler.

MSc, Boğaziçi University, Computer Science, Machine Learning, Natural Language Processing **2012 – 2015**

BSc, Bilgi University, Computer Science **2005 – 2010**

Selected Publications

- Flatt M., Derici C. Dybvig R. K., Keep A. et. al. "Rebuilding racket on chez scheme (experience report)", ICFP'19
- Derici C. et. al. "A closed-domain question answering framework using reliable resources to assist students" Natural Language Engineering'18
- Derici C. et. al. "Question analysis for a closed domain question answering system", CICLING'15
- Derici C. et. al. "Rule-based focus extraction in Turkish question answering systems", SIU'14
- Başar R. E., Derici C., and Şenol Ç. "World With Web: A compiler from world applications to JavaScript". Technical Report, Scheme and Functional Programming Workshop'09

Awards & Scholarships

- Scholarship and award for a project on teaching natural languages to hearing impaired, 2014.
- Full Scholarship for PhD, 2015-2020
- Full Scholarship for MSc, 2012
- Full Scholarship for BSc, 2005-2010