

Hayley Patton

hayley@applied-langua.ge — <https://github.com/no-defun-allowed> — Canberra, Australia

PROJECTS

Common Lisp implementation (2020–)

I contribute to the Steel Bank Common Lisp implementation, most notably having upstreamed a parallel mark-region garbage collector (released in SBCL 2.3.8) which both improves collector throughput and reduces memory usage. I was awarded the [Google Open Source Peer Bonus](#) for the garbage collector.

I refactored and documented the *hash tables* module of the SICL Common Lisp implementation in May 2020, ported Abseil's `flat_hash_map` to Common Lisp in February 2021, and then was employed to write much of the *compiler backend* of SICL from April to October 2021, including *register allocation* and emitting x86-64 assembly.

Object-capability programming systems (2022–)

I work on the Utena interactive object-capability programming system, by designing the *abstract machine* of the system, and implementing the machine in Common Lisp and Standard ML. I wrote an optimising compiler from Utena to a typed assembly language based on AArch64 for my honours project.

I was employed to work on the Newspeak programming system over 2023–2024, in which I implemented WYSIWYG editing and reduced the size of images exported from Newspeak.

Regular expression compilation (2020–2023)

one-more-re-nightmare compiles regular expressions to Common Lisp code for fast text matching. I wrote optimisation passes which generate simpler automata, and accelerated searching of prefix literals by generating *single instruction-multiple data* code.

Replicated object system (2018–2021)

I designed and implemented the *replicated object system* Netfarm from 2018 to 2021 for creating portable and modular decentralised applications. It includes a Kademlia-based *distributed hash table*, work distribution for efficient retrieval of new objects using fine-grained locking which I modelled in TLA^+ , and a content moderation scheme involving *collaborative filtering*.

PUBLICATIONS

Hayley Patton. Parallel garbage collection for SBCL. In *Proceedings of the 16th European Lisp Symposium (ELS'23)*, 2023. <https://zenodo.org/record/7816398>

Hayley Patton. A replicated object system. In *Proceedings of the 14th European Lisp Symposium (ELS'21)*, 2021. <https://zenodo.org/record/4712699>

EDUCATION

Bachelor of Computing (Honours), first class, Australian National University (2024)

My honours thesis *Reinvestigating Typed Assembly Language for the Garbage Collection Interface* covers how to extend a typed assembly language to ensure that compiled code follows its contract with high-performance garbage collectors and was supervised by Steve Blackburn. I took the Parallel Systems (*MPI*, *OpenMP* and *CUDA*), Advanced Topics in Formal Methods and Programming Languages (*Isabelle/HOL*), and Document Analysis electives.

Bachelor of Computer Science with Distinction, RMIT University (2021–2023)

Studied *C++*, *Erlang*, *Java*, *Python*, and *SQL* (with *MySQL* and *Postgres*). My capstone project was exhaustive search of a theoretical game, using SIMD for fast equivalence detection and a GPU cluster programmed with *OpenCL* for the remaining search.

SKILLS

Most proficient as a *Common Lisp*, *C* and *Java* programmer (from most to least), and also familiar with *Erlang*, *Standard ML* and *Python*. Infrequent designer of parallel and concurrent algorithms. Works with *Linux* and *Git*. Typesets documents with $\text{\LaTeX}$. Can play keyboards and guitar if somehow necessary.