

Evan Ovadia

Lead Compiler Engineer
& Language Designer
verdagen.dev

 verdagen_li24@verdagen.dev
 (805) 443-0074
 <https://github.com/verdagen>

Vale Programming Language

Jan 2013 -
Present

- Lead for the [Vale Programming Language](#), a statically typed, ahead-of-time compiled, 100% safe, native language for bringing safety, speed, and flexibility to all programmers.
- Implemented the Vale compiler, 130,000 lines of C++, Scala, and Vale, built on LLVM.
- Created two novel memory safety approaches: [generational references](#) and [region borrowing](#).
- Created [Higher RAI](#), a technique that uses linear types to guarantee future calls.
- Regularly published on [Languages ∩ Architecture](#), [consistently hitting the front page of Hacker News](#).
- Interviewed by [Developer Voices](#) and [Handmade Seattle](#) on compilers and languages.
- Successfully implemented a myriad of complex language features:
 - Full generics, with a separate instantiation phase
 - Polymorphic lambdas, both templated and generic
 - Const generics
 - [Generational references](#) for memory safety
 - [Fearless FFI](#) which prevents accidental memory unsafety in FFI calls
 - [Region borrowing](#), enabling zero-cost memory safety without borrow checking
 - [Perfect replayability](#), to record and perfectly reproduce any run of the program
 - Full module system
 - Weak references without needing reference counting
 - Universal function call syntax, without the need for prefixes
 - Patterns and destructuring of complex objects

Google

Senior Software Engineer

Nov 2019 -
Aug 2021

Google Chat

- Solved an entire class of race conditions from collaborative updating of messages by specifying and enforcing a source of truth for all edits.
- Re-architected the central Message object and led a team in implementing it for each layer of the stack.
- Designed a future-proof Annotation object, for component-like attaching of arbitrary structured data.

Jun 2016 -
Nov 2019

Google Earth

- Architected and led the implementation of the Earth State System. Wrote State Guidelines and Best Practices, and gave a talk at the Geo Summit.
- Designed client-side crypto-shredding of locally cached cloud documents.
- Architected the document synchronization layer, which has grown to be a third of all Earth common code.
- Brought integration testing to the Earth webapp, trained and led the team in writing webdrivers.
- Implemented Chrome LocalFileSystem support, to save users' documents in the browser.

Feb 2014 -
Jun 2016

Google MyMaps

- Designed and implemented the Address Geocoder for the GMEPS backend.
- Designed changing columns' types, to solve a fundamental user-visible sorting problem, with a 36-page doc exploring options, involving load testing, flume jobs, and benchmarking.
- Maintainer of Candyshop, MyMaps' user-generated content discovery platform.
- Designed and implemented the mobile Candyshop site.

Education

California Polytechnic State University

San Luis Obispo, CA

Class of '13 | Masters | [Computer Science](#)
Bachelors | [Software Engineering](#)
Graduate Major GPA: 3.97

Projects, Skills, Interests

- Program architecture, with skills in EC, ECS, functional programming, modern OO, valuing readability, simplicity, flexibility, testability, and performance.
- Language guru, with skills in C++, Rust, Scala, SQL, C#, Java, Kotlin, Go, CSS, HTML, XSLT, PHP, Scheme, Javascript, Typescript, Swift, Objective-C, Visual Basic, and more.
- [Weird Event Finder](#), a web crawler that uses LLMs and retrieval-augmented generation to find interesting yearly events.
- [Incendian Falls](#), the time-traveling roguelike game where the player works with their past selves in combat to overcome enemies and challenges. Featured on [Roguelike Radio](#).
 - Made Chronobase, a declarative language for a journaling database that compiles to C#.
 - Made a system to tile arbitrary groups of polygons.
- [Ember Deep](#), a roguelike game with pentagonally tiled terrain.
- [Geomancer](#), a CRDT-based online collaborative editor for pentagonally-tiled levels.
- Founder and director of Google Humans vs Zombies, a yearly week-long game where hundreds of googlers try to infect humans, stun zombies, and accomplish various missions around the Mountain View and Sunnyvale campuses.
 - Organized >100 volunteers and trained them in web programming, logistics, advertising, game design, safety procedures and flexible planning.
 - Worked with organizations within Google to fund and distribute supplies to players.
 - Led a team of 16 developers to make the PlayHVZ game coordination webapp and mobile apps.
- Led the creation of [Helios](#), the online, multiplayer, 3D RTS game from the ground up with OpenGL and C++. Wrote an optimized pathfinding algorithm, 3D rounded-cylinder intersection code, network synchronization, discrete-step-less game simulation, realtime in-game sanity checking, a configurable shader framework, on-demand and predictive resource loading, a field-of-view algorithm, a fully-threaded non-polling networking library, a vector library, a parametrics-based animation engine, and an allocation tracker.