Akari

A two-level language for code generation, hosted in Rust

Project StarFall

June 2026
Field of Dreams Studio

1 What Akari Is

The big picture

A language with two layers.
Akari has a small inner language for doing math and logic, and an outer template language for generating code in any other language.

The inner language is a lambda calculus.
A lambda calculus is a tiny mathematical language whose only building block is a function (something that takes an input and gives back an output). Everything else is built out of functions.

The outer language is a template engine.
Whatever sits outside its markers is treated as ordinary text in your target language (HTML, SQL, Rust, anything). Whatever sits inside the markers is a small inner-language program that gets computed and pasted in.

A worked example: rendering HTML in Hotaru

A template (`profile.html`) is a normal HTML file with small holes inside the `-[ ... ]-` markers. The holes get filled in.

```
<h1>Hello, -[ user.name ]-!</h1>
<ul>
  -[ for item user.items ]-
    <li>-[ item ]-</li>
  -[ endfor ]-
</ul>
```

A one-line call from Rust feeds the template with real data.

```
akari_render!("profile.html", user = user_data)
```

The same trick works for any target language. Replace HTML with SQL, TypeScript, or Rust source, and the engine still fills in the holes.

2 How Akari Works

Three modules, one system

Expr: the inner language (in flight).
A small lambda-calculus engine. Anything you compute in Akari is built out of Expr.

Template and codegen: the outer language.
The engine that walks a template, fills its holes by running Expr, and prints source code for any target language.

Value and extension: the bridge to Rust.
Lets a host Rust program hand data into Akari and read results back out. The same role a bridge plays between any two languages.

Why lambda calculus for the inner language

It is tiny.
Lambda calculus has only one move: take an input, return an output. That single rule is enough to build everything else.

Functions fit together naturally.
You can pass a function as an argument, return one from another, or chain them. Most languages have to add this. Lambda calculus has it from day one.

A starting point that can grow.
Once the small core is right, we can add a type system or other safety guarantees later, without throwing away what is built.

3 Current Achievements

What ships today

The Value system.
A flexible data type for JSON-shaped data, with reading, writing, and arithmetic.

The typemap extensions.
A small toolkit for carrying arbitrary side data around. Faster than the popular `http::Extensions` library that solves the same problem.

The template engine.
HTML templating with inheritance and loops, used in production through the Hotaru web framework.

Embedded support.
Runs on tiny embedded devices and inside web browsers, where heavier libraries cannot.

Hotaru: a real user, today

What Hotaru is.
A Rust web framework, a toolkit for building websites, also developed inside Field of Dreams Studio.

How Hotaru uses Akari.
Every HTML page that Hotaru serves is rendered by Akari's template engine, through a one-line call written `akari_render!`.

Why this matters.
Akari is not just a plan. The template engine is already powering real websites. The work ahead is to deepen the language, not to prove it can work.

4 What Is Left

The work ahead

Build the inner language.
Pin down the syntax, write the program that runs it, design how programs sit in memory.

Redesign the template engine.
Replace the current quick version inside markers with the new inner language. Make it work for any target language.

Tighten the bridge to Rust.
Smoother conversion between Rust's types, Akari's Value system, and inner-language expressions.

Polish.
An interactive prompt, online documentation, examples in more target languages, a guide for first-time contributors.

5 How You Can Help

Project StarFall and Field of Dreams Studio

The Field of Dreams Studio (FDS)
FDS is a non-profit game development team founded on February 2, 2017, and rebuilt on March 30, 2022. Its core objective is to create a platform that does not rush for short-term gains, is relatively relaxed, and allows members to fully unleash their creativity for creation.

The Akari Lang and Project-SF
The Project-StarFall is a project under the FDS that focus on Minecraft RPG development as well as website maintainence. The Akari language is a project under the Project-StarFall.

The Akari Lang project started in early 2025, and is still developing now.
We also run projects like Minecraft GalGames, where the main Project-StarFall started in 2020.

How we work

Open by default.
The repository is on GitHub and the discussion happens in the open.

Live and asynchronous together.
A weekly meeting in real time, with code review the rest of the time at everyone's own pace.

Onboarding.
New members are paired with an existing member for their first few weeks.

Decisions in writing.
Decisions are documented so newcomers can join the conversation, not just the code.

Five ways to contribute

Pick what fits your skills and your time. You can switch roles later.

- **Core development.** Write Rust code: build the inner language, the template engine, the bridge to Rust.
- **Language theory.** Help shape the inner language. Think about correctness, types, and the guarantees the language should give its users.
- **Documentation.** Write tutorials, examples, a guide for first-time contributors.
- **Social and community.** Run the website, post updates, welcome new members, organise small events.
- **Use and feedback.** Build something with Akari (or Hotaru) and tell us what hurts.

What you take with you

A real open-source project on your name.
Not a homework repository, and not a one-semester sprint. You may need to spend years on it.

Mentorship.
From people who have shipped Rust libraries and who run a small language project.

Peers who stay around.
Project StarFall members collaborate across projects for years.

A window into real maintenance.
A first look at how serious software is actually designed, built, and maintained.

How to join

Reach out through any of these channels. No Rust experience is required, only the willingness to learn and to stick around.

- Sign-up form:
  `https://forms.office.com/Pages/ResponsePage.aspx?id=DQSIkWdsW0yxEjajBLZtrQAAAAAAAAAAAAMAAC6BwJ5UQ0lQUzdMTjhGR1g3SElLTFdHQUlJV0hFMS4u`
- Email: `redstone@fds.moe`
- QQ: 590328937
- Discord: `https://discord.gg/Y6b9KRUCux`

To browse first, see fds.moe or the repository at github.com/Field-of-Dreams-Studio/akari.