Building a
Proof Assistant

Favonia

Today's focus

Assistants based on dependent type theory

Coq, Agda, Lean, ...

Proof assistant is like a compiler, but...

Interactivity is prime focus

Code generation: less so

IDE is not a afterthought

RedPRL Team

Tulip S. Amalie

Carlo Angiuli

Evan Cavallo

Favonia

Matthew McQuaid

Reed Mullanix

Jonathan Sterling

RedPRL   redtt   cooltt   algaett

Build Yet⁴ Another Proof Assistant

NbE* — check whether two terms are equal

*Normalization by Evaluation

Refiner — check whether a term is well-typed

Kernel — the "trusted" part

Elaborator — surface syntax to core syntax

Driver — imports, definitions, etc.

Parser — strings to surface syntax

NbE*
Refiner
Kernel

Elaborator
Driver
Parser

yuujinchou: lexical scoping

asai: compiler diagnostics

bantorra: library management

base utility

algaeff: algebraic effects
bwd: backward lists

type-theory specific

kado: cofibrations in Cartesian cubical type theory
mugen: universe levels

Dream

Prototype good proof assistants for new type theories in no time

yuujinchou

namespaces and lexical scoping

Expressiveness: all features in common PLs
Explicit sequencing: unambiguous name modifiers
Implicit namespaces: easy definition injection
Typo detection
Small kernel
Effects and handlers!

used by cooltt,
polytt, forester

```
open M
  using (a)
  renaming (a to b)
  hiding (b)
```

Agda code: what would happen?

asai

compiler diagnostics

Great Unicode support: no column numbers

Multiple backends: terminal, GitHub, LSP*...

*currently not handling UTF-16

Backtraces

Multi-span highlighting

Effects and handlers!

What go into a diagnostic?

Should we end the program now? emit v.s. fatal

How serious the message is to users? error or warning?

Short, Google-able code E0001

Backtraces

Additional messages / multi-span marking

bantorra

library management

Recursive POSIX-style mounting

A DSL for library resultion

    Built-in routers: local dir, git, index, ...

    Built-in combinators: dispatch, rewrite, ...
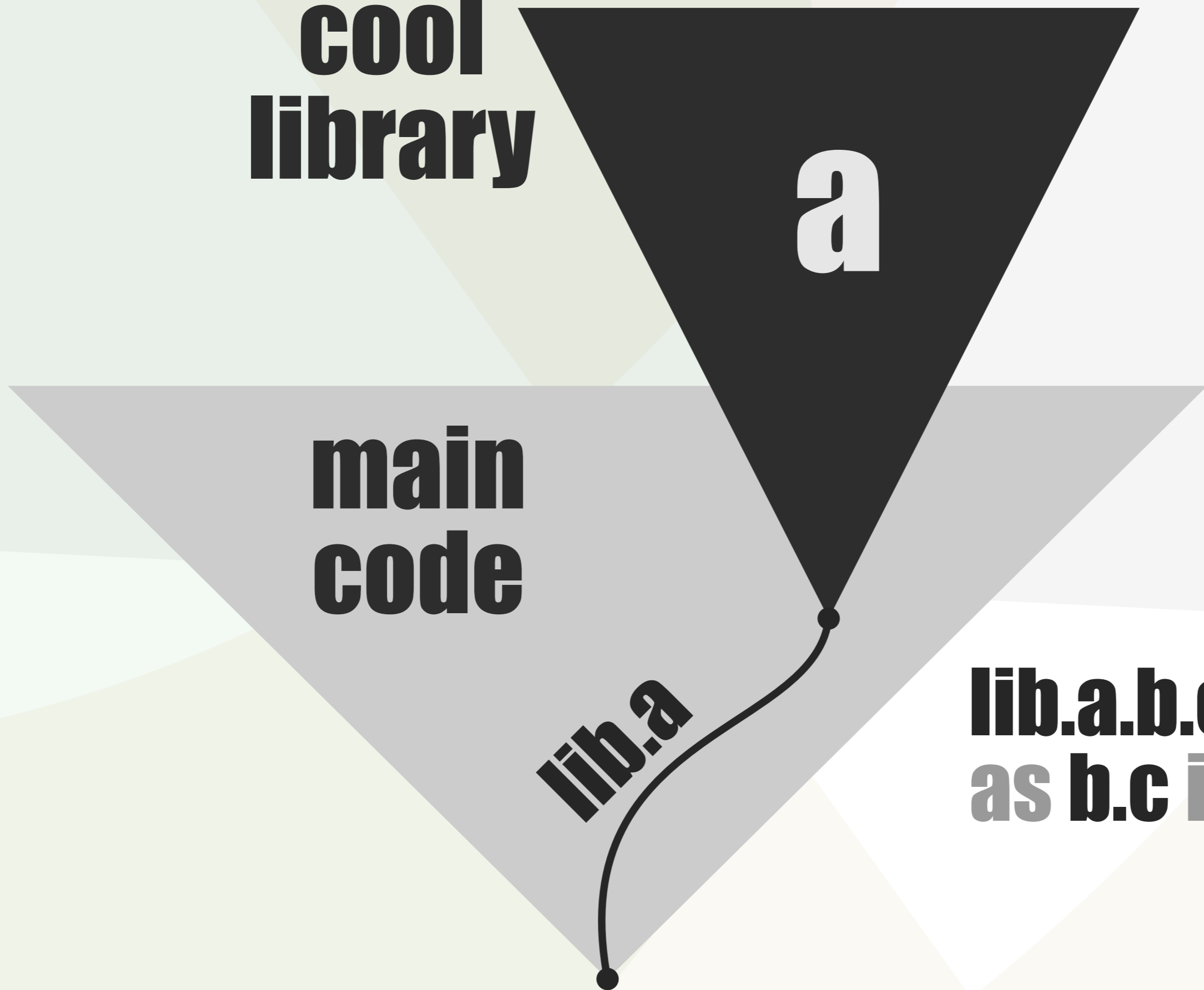
Thread safety

Effects and handlers!

cool
library

a

main
code

lib.a

lib.a.b.c understood
as **b.c** in the library

Minimum Version Selection

Use the minimum versions to satisfy version constraints

Simple, no SAT solving
Reproducible with a stable version index

used by Go

kado
cofibrations

Heavily optimized; used in cooltt

Used to implement unfolding control

This part will be handled by a new library, yomiko, soon

mugen

Negative levels: each universe contains a smaller one

Rational levels: always a level between any two levels

Fractal levels: embedding itself between any two levels

More to Come

Records
Pretty Printing
Inductive Types
Controlled Unfolding - yomiko
...

Website: redprl.org

Available on OPAM: algaeff asai bwd
mugen yuujinchou

Development: github.com/RedPRL

Website: `redprl.org`

Available on OPAM: `algaeff asai bwd mugen yuujinchou`

Development: `github.com/RedPRL`

Consider these packages next time!