Prism

Ruby World 2023

Kevin Newton

What is prism?

Overview

Overview

Prism is a new Ruby parser built from scratch with the following goals:

Overview

Prism is a new Ruby parser built from scratch with the following goals:

- Compatibility - must match CRuby semantics exactly in order to merge

Overview

Prism is a new Ruby parser built from scratch with the following goals:

- Compatibility - must match CRuby semantics exactly in order to merge
- Maintainability - must be as maintainable as possible

Overview

Prism is a new Ruby parser built from scratch with the following goals:

- Compatibility - must match CRuby semantics exactly in order to merge
- Maintainability - must be as maintainable as possible
- Error tolerance - must be able to recover from errors and continue parsing

Overview

Prism is a new Ruby parser built from scratch with the following goals:

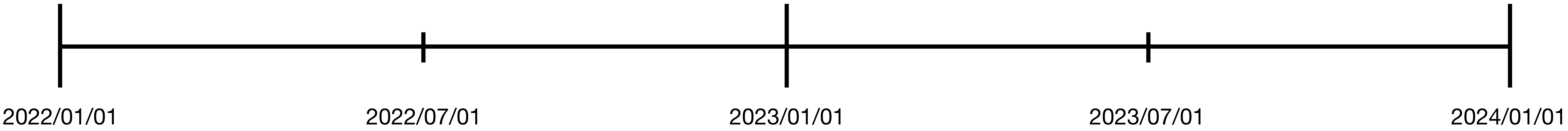
- Compatibility - must match CRuby semantics exactly in order to merge
- Maintainability - must be as maintainable as possible
- Error tolerance - must be able to recover from errors and continue parsing
- Portability - must be able to be run from any ruby platform

Overview

Prism is a new Ruby parser built from scratch with the following goals:

- Compatibility - must match CRuby semantics exactly in order to merge
- Maintainability - must be as maintainable as possible
- Error tolerance - must be able to recover from errors and continue parsing
- Portability - must be able to be run from any ruby platform
- Performance - must be as fast and memory efficient as possible

Timeline



2022/01/01

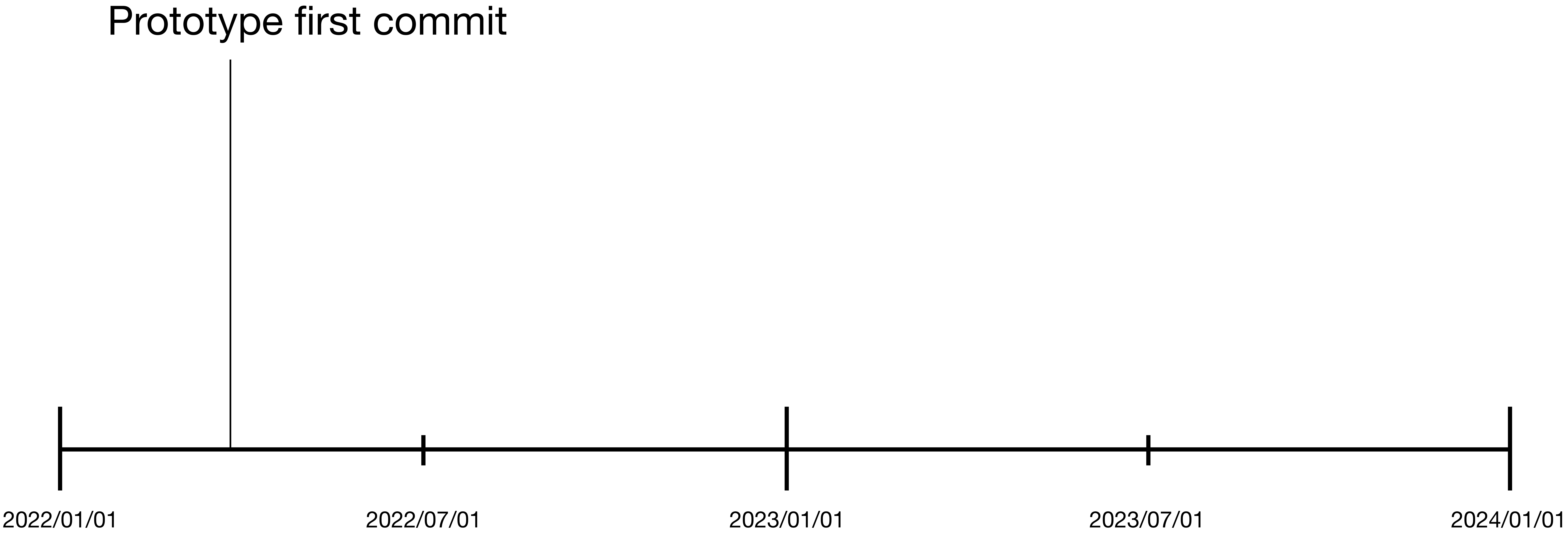
2022/07/01

2023/01/01

2023/07/01

2024/01/01

Timeline



Prototype first commit

2022/01/01

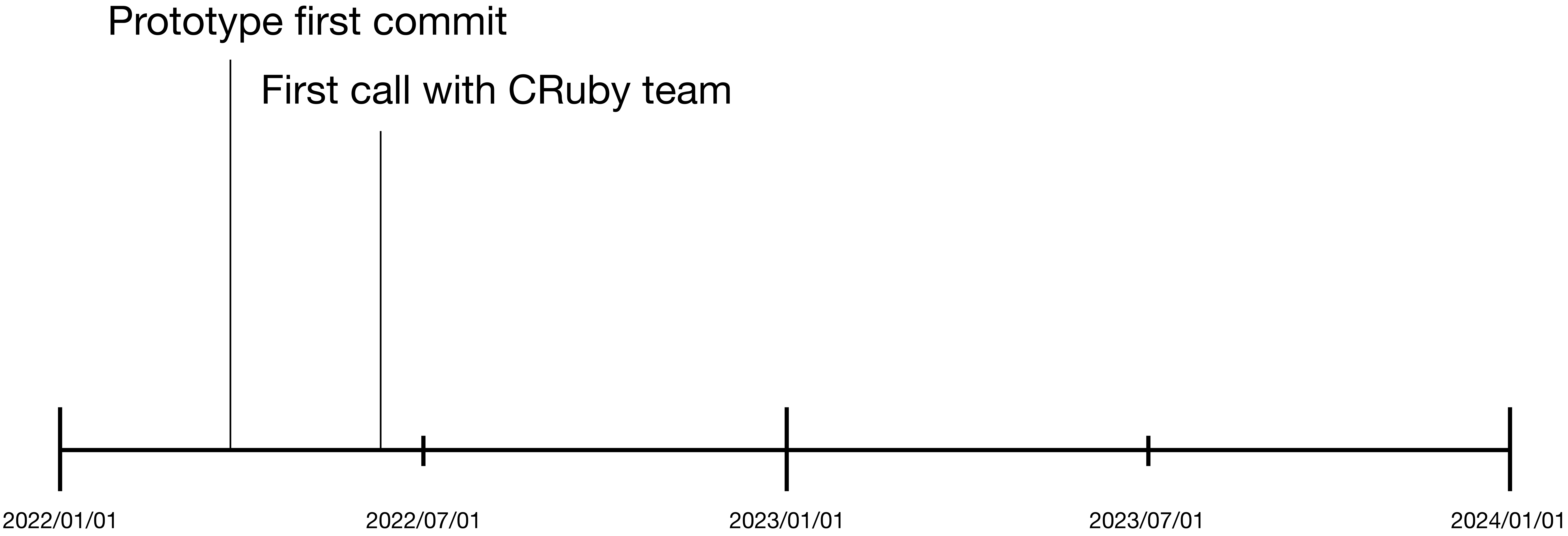
2022/07/01

2023/01/01

2023/07/01

2024/01/01

Timeline



Prototype first commit

First call with CRuby team

2022/01/01

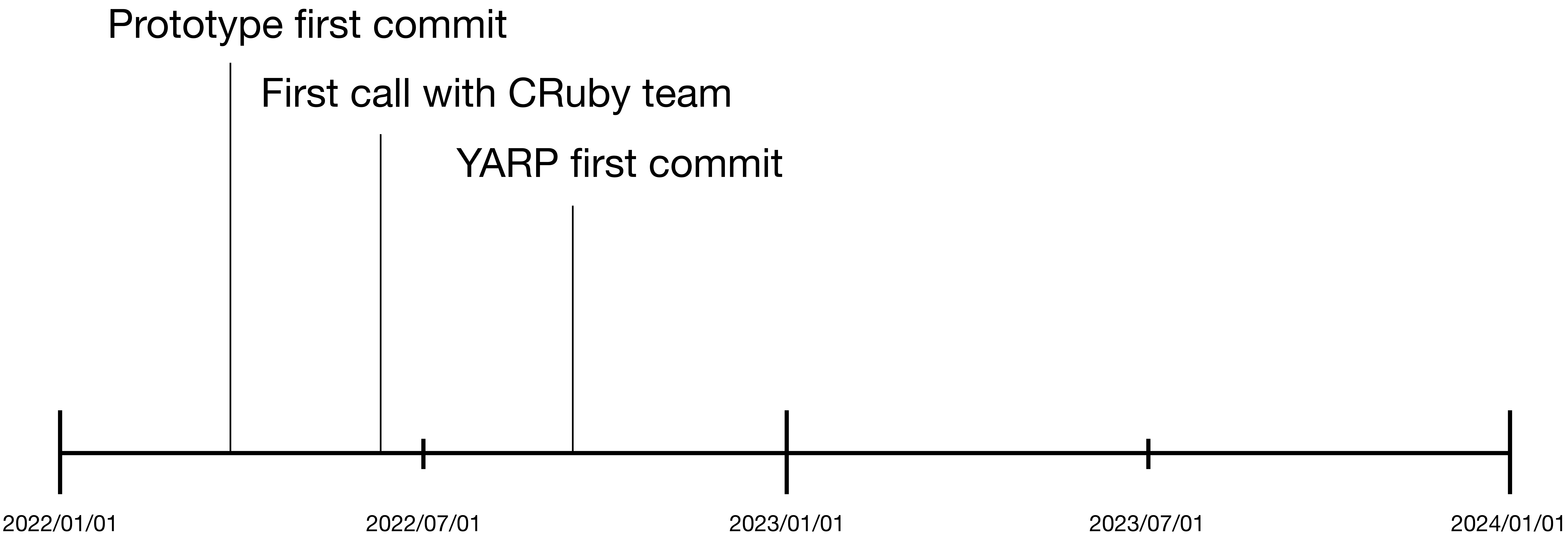
2022/07/01

2023/01/01

2023/07/01

2024/01/01

Timeline

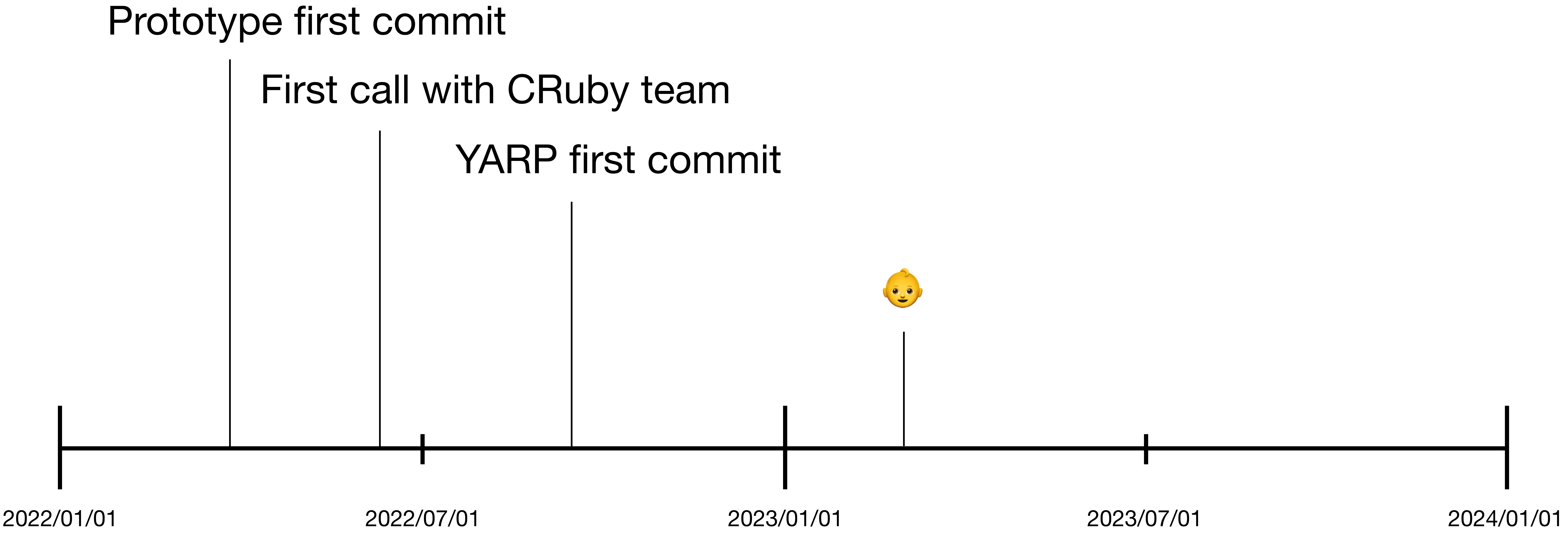


Prototype first commit

First call with CRuby team

YARP first commit

Timeline



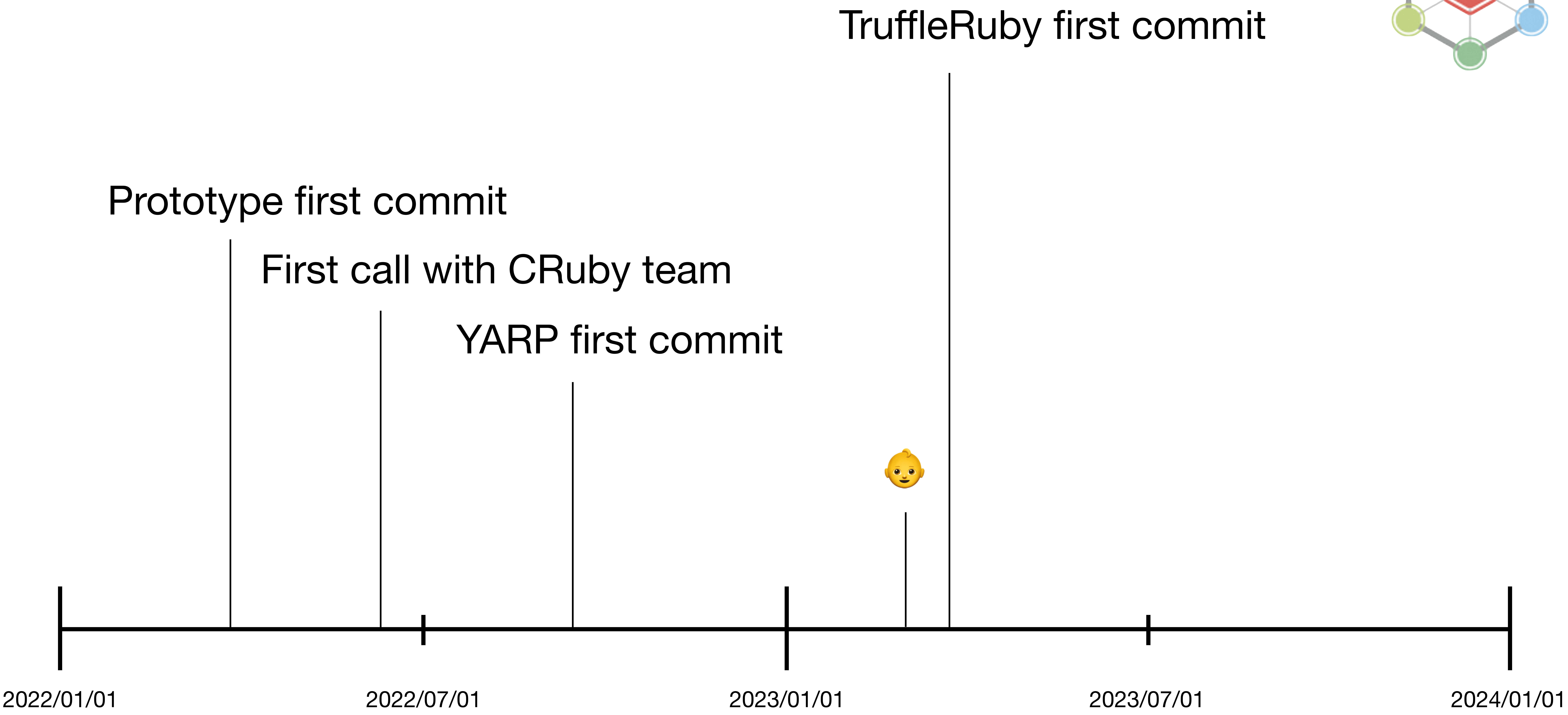
Prototype first commit

First call with CRuby team

YARP first commit

👶

Timeline



TruffleRuby first commit

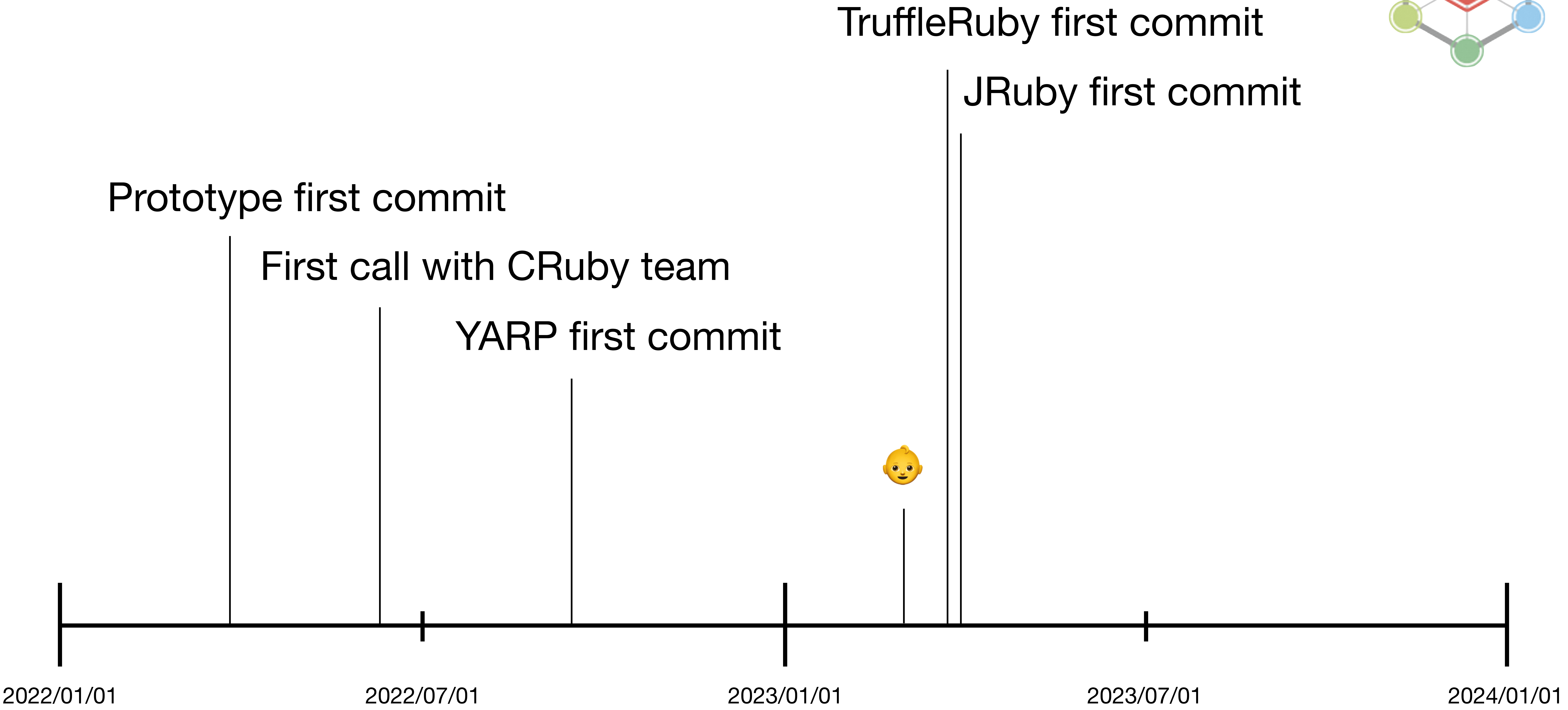
Prototype first commit

First call with CRuby team

YARP first commit

👶

Timeline



TruffleRuby first commit

JRuby first commit

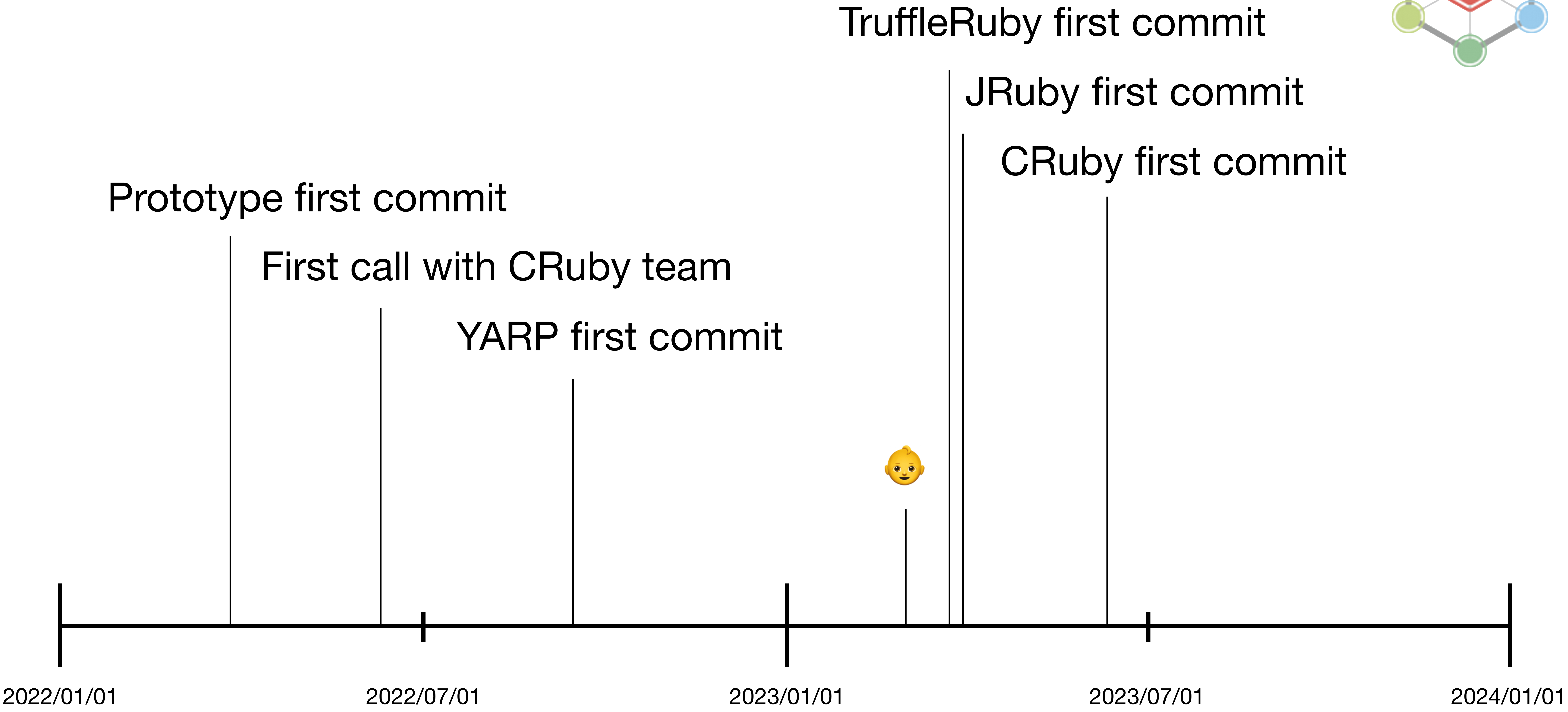
Prototype first commit

First call with CRuby team

YARP first commit

👶

Timeline



Prototype first commit

First call with CRuby team

YARP first commit

👶

TruffleRuby first commit

JRuby first commit

CRuby first commit

2022/01/01

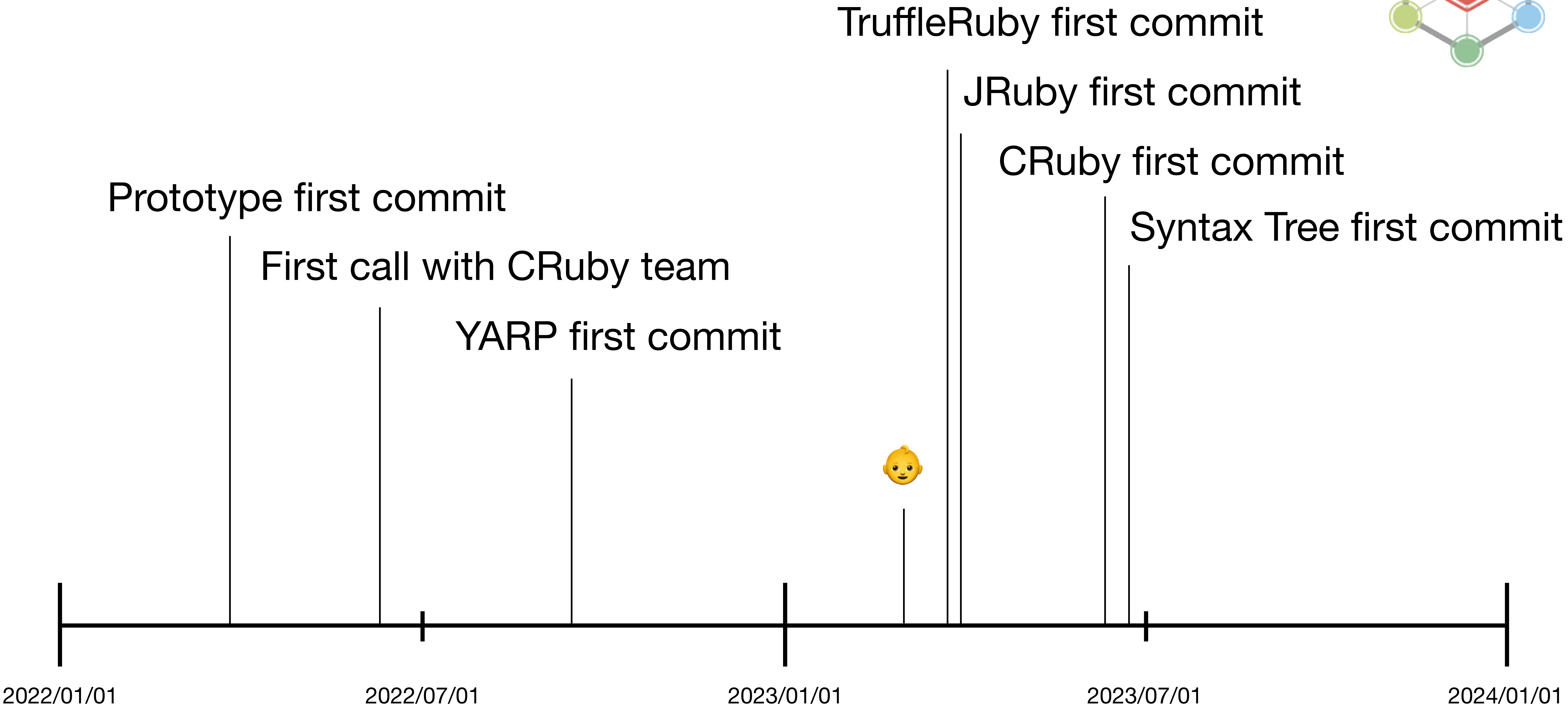
2022/07/01

2023/01/01

2023/07/01

2024/01/01

Timeline



Prototype first commit

First call with CRuby team

YARP first commit

👶

TruffleRuby first commit

JRuby first commit

CRuby first commit

Syntax Tree first commit

2022/01/01

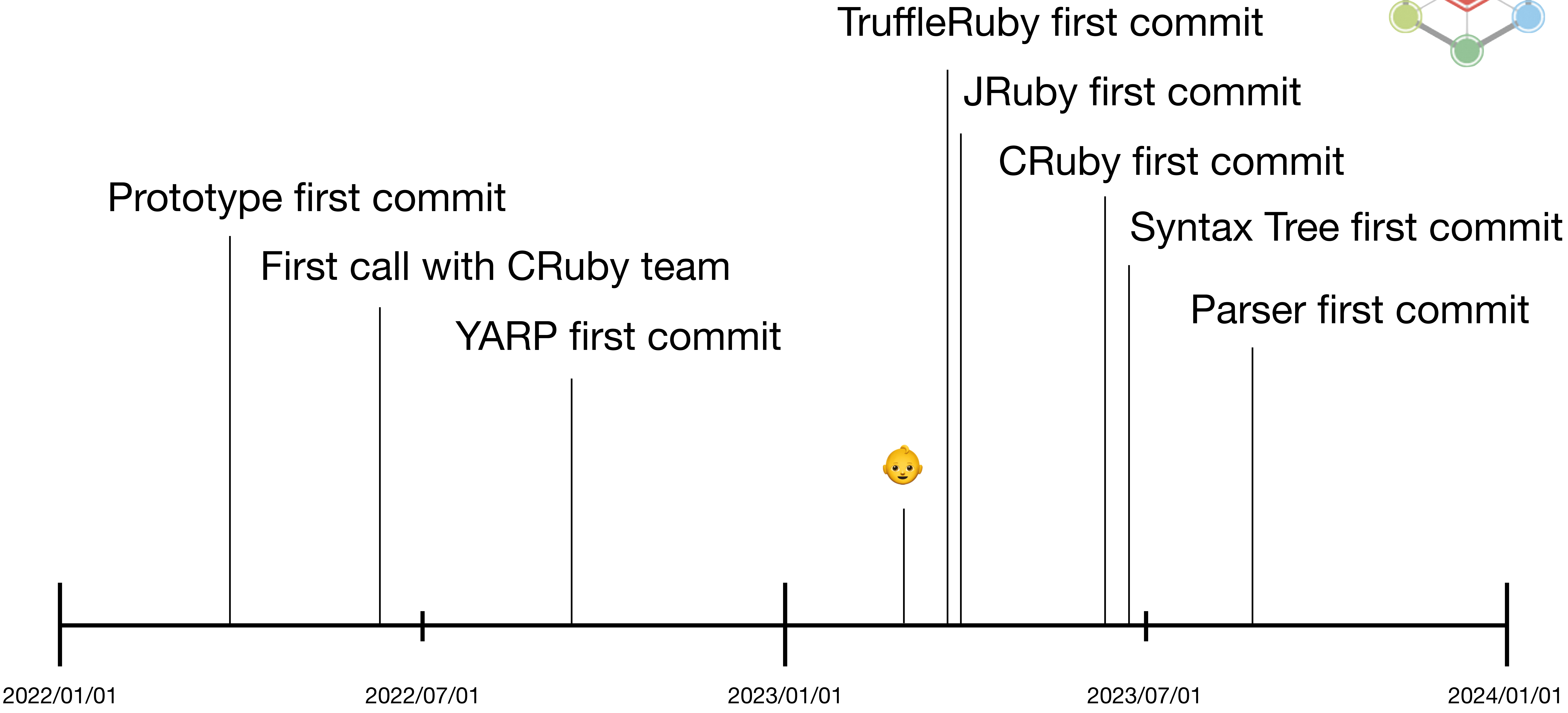
2022/07/01

2023/01/01

2023/07/01

2024/01/01

Timeline



Prototype first commit

First call with CRuby team

YARP first commit

👶

TruffleRuby first commit

JRuby first commit

CRuby first commit

Syntax Tree first commit

Parser first commit

2022/01/01

2022/07/01

2023/01/01

2023/07/01

2024/01/01

Timeline

Prototype first commit

First call with CRuby team

YARP first commit

👶

TruffleRuby first commit

JRuby first commit

CRuby first commit

Syntax Tree first commit

Parser first commit

Rename to prism

2022/01/01

2022/07/01

2023/01/01

2023/07/01

2024/01/01

How was it built?

Design

Design

Approach

Design

Approach

- Comparative analysis with existing trees

Design

Approach

- Comparative analysis with existing trees
- "Semantic" versus "syntactic" trees

Design

Approach

- Comparative analysis with existing trees
- "Semantic" versus "syntactic" trees
- Provide for all consumers

Design

Implementations

Design

Implementations

- Make compilation as simple as possible

Design

Implementations

- Make compilation as simple as possible
- Never have to check child to compile current node

Design

Implementations

- Make compilation as simple as possible
- Never have to check child to compile current node
- Cover as much common functionality as possible

Design

Tooling

Design

Tooling

- Named fields and known types

Design

Tooling

- Named fields and known types
- Nodes with enough information to round-trip

Design

Tooling

- Named fields and known types
- Nodes with enough information to round-trip
- Location information for as much as possible

Design

Maintenance

Design

Maintenance

- Template as much as possible

Design

Maintenance

- Template as much as possible
- Provide as much documentation as possible

Design

Maintenance

- Template as much as possible
- Provide as much documentation as possible
- Build up as big of a test suite as possible

Challenges

Challenges

- Escaping - required to properly find string-like terminators

Challenges

- Escaping - required to properly find string-like terminators
- Regular expressions - required because of named capture groups

Challenges

- Escaping - required to properly find string-like terminators
- Regular expressions - required because of named capture groups
- Encoding support - required to properly find string-like terminators

Challenges

- Escaping - required to properly find string-like terminators
- Regular expressions - required because of named capture groups
- Encoding support - required to properly find string-like terminators
- Heredocs - very complicated semantics, can be almost anywhere in source

Testing

Testing

- Lexer testing - testing against ripper lex

Testing

- Lexer testing - testing against ripper lex
- Snapshot testing - protect from regressions, check parser structure

Testing

- Lexer testing - testing against ripper lex
- Snapshot testing - protect from regressions, check parser structure
- Brute-force testing - all escape sequences in all contexts

Testing

- Lexer testing - testing against ripper lex
- Snapshot testing - protect from regressions, check parser structure
- Brute-force testing - all escape sequences in all contexts
- Fuzzing - random input, check for crashes

Testing

- Lexer testing - testing against ripper lex
- Snapshot testing - protect from regressions, check parser structure
- Brute-force testing - all escape sequences in all contexts
- Fuzzing - random input, check for crashes
- Grammar-based fuzzing - random input, check for crashes and parse errors

Integrations

Integrations

- C - CRuby, Sorbet, Artichoke, Natalie, C/C++/Rust/Zig tooling

Integrations

- C - CRuby, Sorbet, Artichoke, Natalie, C/C++/Rust/Zig tooling
- Ruby - Natalie, tooling

Integrations

- C - CRuby, Sorbet, Artichoke, Natalie, C/C++/Rust/Zig tooling
- Ruby - Natalie, tooling
- Serialization - JRuby, TruffleRuby, JavaScript tooling

Why build this?

Fractured ecosystem

Ruby parsers

Fractured ecosystem

Ruby parsers

- ruby/ruby/parse.y

Fractured ecosystem

Ruby parsers

- ruby/ruby/parse.y
- ruby/ruby/ext/ripper/parse.y

Fractured ecosystem

Ruby parsers

- ruby/ruby/parse.y
- ruby/ruby/ext/ripper/parse.y
- lib-ruby-parser/lib-ruby-parser

Fractured ecosystem

Ruby parsers

- ruby/ruby/parse.y
- ruby/ruby/ext/ripper/parse.y
- lib-ruby-parser/lib-ruby-parser
- mruby/mruby

Fractured ecosystem

Ruby parsers

- ruby/ruby/parse.y
- ruby/ruby/ext/ripper/parse.y
- lib-ruby-parser/lib-ruby-parser
- mruby/mruby
- typedruby/typedruby, sorbet/sorbet

Fractured ecosystem

Ruby parsers

- ruby/ruby/parse.y
- ruby/ruby/ext/ripper/parse.y
- lib-ruby-parser/lib-ruby-parser
- mruby/mruby
- typedruby/typedruby, sorbet/sorbet
- whitequark/parser

Fractured ecosystem

Ruby parsers

- ruby/ruby/parse.y
- ruby/ruby/ext/ripper/parse.y
- lib-ruby-parser/lib-ruby-parser
- mruby/mruby
- typedruby/typedruby, sorbet/sorbet
- whitequark/parser
- seattlerb/parser

Fractured ecosystem

Ruby parsers

- ruby/ruby/parse.y
- ruby/ruby/ext/ripper/parse.y
- lib-ruby-parser/lib-ruby-parser
- mruby/mruby
- typedruby/typedruby, sorbet/sorbet
- whitequark/parser
- seattlerb/parser
- jruby/jruby

Fractured ecosystem

Ruby parsers

- ruby/ruby/parse.y
- ruby/ruby/ext/ripper/parse.y
- lib-ruby-parser/lib-ruby-parser
- mruby/mruby
- typedruby/typedruby, sorbet/sorbet
- whitequark/parser
- seattlerb/parser
- jruby/jruby
- oracle/truffleruby

Fractured ecosystem

Ruby parsers

- ruby/ruby/parse.y
- ruby/ruby/ext/ripper/parse.y
- lib-ruby-parser/lib-ruby-parser
- mruby/mruby
- typedruby/typedruby, sorbet/sorbet
- whitequark/parser
- seattlerb/parser
- jruby/jruby
- oracle/truffleruby
- tree-sitter/tree-sitter-ruby

Fractured ecosystem

Ruby parsers

- ruby/ruby/parse.y
- ruby/ruby/ext/ripper/parse.y
- lib-ruby-parser/lib-ruby-parser
- mruby/mruby
- typedruby/typedruby, sorbet/sorbet
- whitequark/parser
- seattlerb/parser
- jruby/jruby
- oracle/truffleruby
- tree-sitter/tree-sitter-ruby
- sisshiki1969/ruruby

Fractured ecosystem

Ruby parsers

- ruby/ruby/parse.y
- ruby/ruby/ext/ripper/parse.y
- lib-ruby-parser/lib-ruby-parser
- mruby/mruby
- typedruby/typedruby, sorbet/sorbet
- whitequark/parser
- seattlerb/parser
- jruby/jruby
- oracle/truffleruby
- tree-sitter/tree-sitter-ruby
- sisshiki1969/ruruby
- natalie-lang/natalie

Fractured ecosystem

Ruby parsers

- ruby/ruby/parse.y
- ruby/ruby/ext/ripper/parse.y
- lib-ruby-parser/lib-ruby-parser
- mruby/mruby
- typedruby/typedruby, sorbet/sorbet
- whitequark/parser
- seattlerb/parser
- jruby/jruby
- oracle/truffleruby
- tree-sitter/tree-sitter-ruby
- sisshiki1969/ruruby
- natalie-lang/natalie
- ruby/prism

Fractured ecosystem

Ruby syntax trees

Fractured ecosystem

Ruby syntax trees

- NODE, RubyVM::AbstractSyntaxTree

Fractured ecosystem

Ruby syntax trees

- NODE, RubyVM::AbstractSyntaxTree
- Ripper::SexpBuilder{,PP}

Fractured ecosystem

Ruby syntax trees

- NODE, RubyVM::AbstractSyntaxTree
- Ripper::SexpBuilder{,PP}
- ruby-syntax-tree/syntax_tree

Fractured ecosystem

Ruby syntax trees

- NODE, RubyVM::AbstractSyntaxTree
- Ripper::SexpBuilder{,PP}
- ruby-syntax-tree/syntax_tree
- lib-ruby-parser/lib-ruby-parser

Fractured ecosystem

Ruby syntax trees

- NODE, RubyVM::AbstractSyntaxTree
- Ripper::SexpBuilder{,PP}
- ruby-syntax-tree/syntax_tree
- lib-ruby-parser/lib-ruby-parser
- mruby/mruby

Fractured ecosystem

Ruby syntax trees

- NODE, RubyVM::AbstractSyntaxTree
- Ripper::SexpBuilder{,PP}
- ruby-syntax-tree/syntax_tree
- lib-ruby-parser/lib-ruby-parser
- mruby/mruby
- typedruby/typedruby, sorbet/sorbet

Fractured ecosystem

Ruby syntax trees

- NODE, RubyVM::AbstractSyntaxTree
- Ripper::SexpBuilder{,PP}
- ruby-syntax-tree/syntax_tree
- lib-ruby-parser/lib-ruby-parser
- mruby/mruby
- typedruby/typedruby, sorbet/sorbet
- whitequark/parser, rubocop/rubocop-ast

Fractured ecosystem

Ruby syntax trees

- NODE, RubyVM::AbstractSyntaxTree
- Ripper::SexpBuilder{,PP}
- ruby-syntax-tree/syntax_tree
- lib-ruby-parser/lib-ruby-parser
- mruby/mruby
- typedruby/typedruby, sorbet/sorbet
- whitequark/parser, rubocop/rubocop-ast
- seattlerb/sexp_processor

Fractured ecosystem

Ruby syntax trees

- NODE, RubyVM::AbstractSyntaxTree
- Ripper::SexpBuilder{,PP}
- ruby-syntax-tree/syntax_tree
- lib-ruby-parser/lib-ruby-parser
- mruby/mruby
- typedruby/typedruby, sorbet/sorbet
- whitequark/parser, rubocop/rubocop-ast
- seattlerb/sexp_processor
- org.jruby.ast

Fractured ecosystem

Ruby syntax trees

- NODE, RubyVM::AbstractSyntaxTree
- Ripper::SexpBuilder{,PP}
- ruby-syntax-tree/syntax_tree
- lib-ruby-parser/lib-ruby-parser
- mruby/mruby
- typedruby/typedruby, sorbet/sorbet
- whitequark/parser, rubocop/rubocop-ast
- seattlerb/sexp_processor
- org.jruby.ast
- Truffle

Fractured ecosystem

Ruby syntax trees

- NODE, RubyVM::AbstractSyntaxTree
- Ripper::SexpBuilder{,PP}
- ruby-syntax-tree/syntax_tree
- lib-ruby-parser/lib-ruby-parser
- mruby/mruby
- typedruby/typedruby, sorbet/sorbet
- whitequark/parser, rubocop/rubocop-ast
- seattlerb/sexp_processor
- org.jruby.ast
- Truffle
- tree-sitter/tree-sitter

Fractured ecosystem

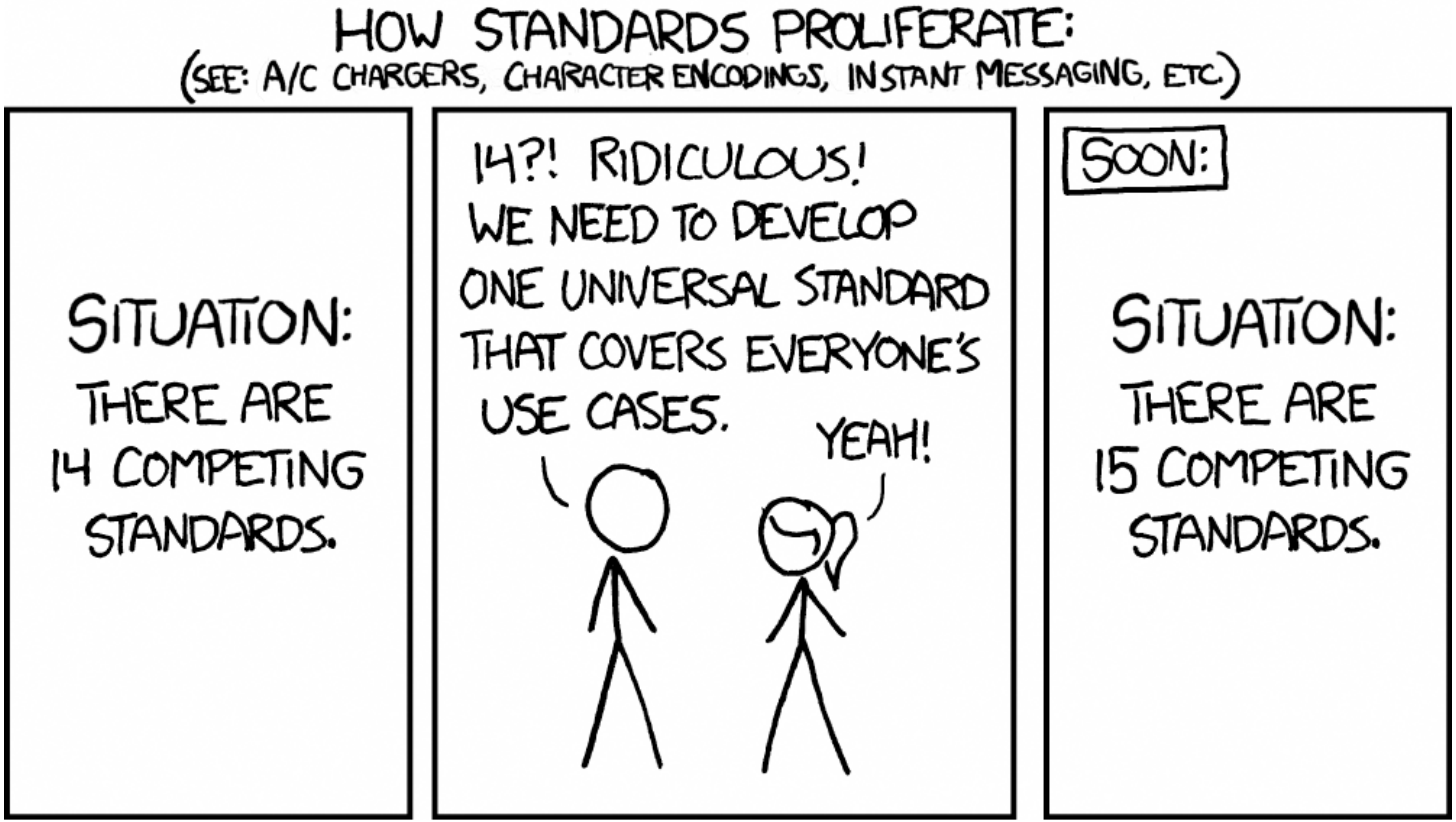
Ruby syntax trees

- NODE, RubyVM::AbstractSyntaxTree
- Ripper::SexpBuilder{,PP}
- ruby-syntax-tree/syntax_tree
- lib-ruby-parser/lib-ruby-parser
- mruby/mruby
- typedruby/typedruby, sorbet/sorbet
- whitequark/parser, rubocop/rubocop-ast
- seattlerb/sexp_processor
- org.jruby.ast
- Truffle
- tree-sitter/tree-sitter
- sisshiki1969/ruruby

Fractured ecosystem

Ruby syntax trees

- NODE, RubyVM::AbstractSyntaxTree
- Ripper::SexpBuilder{,PP}
- ruby-syntax-tree/syntax_tree
- lib-ruby-parser/lib-ruby-parser
- mruby/mruby
- typedruby/typedruby, sorbet/sorbet
- whitequark/parser, rubocop/rubocop-ast
- seattlerb/sexp_processor
- org.jruby.ast
- Truffle
- tree-sitter/tree-sitter
- sisshiki1969/ruruby
- natalie-lang/natalie


HOW STANDARDS PROLIFERATE:
(SEE: A/C CHARGERS, CHARACTER ENCODINGS, INSTANT MESSAGING, ETC.)
SITUATION: THERE ARE 14 COMPETING STANDARDS.
14?! RIDICULOUS! WE NEED TO DEVELOP ONE UNIVERSAL STANDARD THAT COVERS EVERYONE'S USE CASES.
YEAH!
SOON:
SITUATION: THERE ARE 15 COMPETING STANDARDS.

Fractured ecosystem

Ruby syntax trees

- NODE, RubyVM::AbstractSyntaxTree
- Ripper::SexpBuilder{,PP}
- ruby-syntax-tree/syntax_tree
- lib-ruby-parser/lib-ruby-parser
- mruby/mruby
- typedruby/typedruby, sorbet/sorbet
- whitequark/parser, rubocop/rubocop-ast
- seattlerb/sexp_processor
- org.jruby.ast
- Truffle
- tree-sitter/tree-sitter
- sisshiki1969/ruruby
- natalie-lang/natalie

Fractured ecosystem

Ruby syntax trees

- NODE, RubyVM::AbstractSyntaxTree
- Ripper::SexpBuilder{,PP}
- ruby-syntax-tree/syntax_tree
- lib-ruby-parser/lib-ruby-parser
- mruby/mruby
- typedruby/typedruby, sorbet/sorbet
- whitequark/parser, rubocop/rubocop-ast
- seattlerb/sexp_processor
- org.jruby.ast
- Truffle
- tree-sitter/tree-sitter
- sisshiki1969/ruruby
- natalie-lang/natalie
- ruby/prism

Fractured ecosystem

Ruby syntax trees

- ~~NODE, RubyVM::AbstractSyntaxTree~~
- Ripper::SexpBuilder{,PP}
- ruby-syntax-tree/syntax_tree
- lib-ruby-parser/lib-ruby-parser
- mruby/mruby
- typedruby/typedruby, sorbet/sorbet
- whitequark/parser, rubocop/rubocop-ast
- seattlerb/sexp_processor
- org.jruby.ast
- Truffle
- tree-sitter/tree-sitter
- sisshiki1969/ruruby
- natalie-lang/natalie
- ruby/prism

Fractured ecosystem

Ruby syntax trees

- ~~NODE, RubyVM::AbstractSyntaxTree~~
- ~~Ripper::SexpBuilder{,PP}~~
- ruby-syntax-tree/syntax_tree
- lib-ruby-parser/lib-ruby-parser
- mruby/mruby
- typedruby/typedruby, sorbet/sorbet
- whitequark/parser, rubocop/rubocop-ast
- seattlerb/sexp_processor
- org.jruby.ast
- Truffle
- tree-sitter/tree-sitter
- sisshiki1969/ruruby
- natalie-lang/natalie
- ruby/prism

Fractured ecosystem

Ruby syntax trees

- ~~NODE, RubyVM::AbstractSyntaxTree~~
- ~~Ripper::SexpBuilder{,PP}~~
- ~~ruby-syntax-tree/syntax_tree~~
- lib-ruby-parser/lib-ruby-parser
- mruby/mruby
- typedruby/typedruby, sorbet/sorbet
- whitequark/parser, rubocop/rubocop-ast
- seattlerb/sexp_processor
- org.jruby.ast
- Truffle
- tree-sitter/tree-sitter
- sisshiki1969/ruruby
- natalie-lang/natalie
- ruby/prism

Fractured ecosystem

Ruby syntax trees

- ~~NODE, RubyVM::AbstractSyntaxTree~~
- ~~Ripper::SexpBuilder{,PP}~~
- ~~ruby-syntax-tree/syntax_tree~~
- lib-ruby-parser/lib-ruby-parser
- mruby/mruby
- ~~typedruby/typedruby, sorbet/sorbet~~
- whitequark/parser, rubocop/rubocop-ast
- seattlerb/sexp_processor
- org.jruby.ast
- Truffle
- tree-sitter/tree-sitter
- sisshiki1969/ruruby
- natalie-lang/natalie
- ruby/prism

Fractured ecosystem

Ruby syntax trees

- ~~NODE, RubyVM::AbstractSyntaxTree~~
- ~~Ripper::SexpBuilder{,PP}~~
- ~~ruby-syntax-tree/syntax_tree~~
- lib-ruby-parser/lib-ruby-parser
- mruby/mruby
- ~~typedruby/typedruby, sorbet/sorbet~~
- ~~whitequark/parser, rubocop/rubocop-ast~~
- seattlerb/sexp_processor
- org.jruby.ast
- Truffle
- tree-sitter/tree-sitter
- sisshiki1969/ruruby
- natalie-lang/natalie
- ruby/prism

Fractured ecosystem

Ruby syntax trees

- ~~NODE, RubyVM::AbstractSyntaxTree~~
- ~~Ripper::SexpBuilder{,PP}~~
- ~~ruby-syntax-tree/syntax_tree~~
- lib-ruby-parser/lib-ruby-parser
- mruby/mruby
- ~~typedruby/typedruby, sorbet/sorbet~~
- ~~whitequark/parser, rubocop/rubocop-ast~~
- ~~seattlerb/sexp_processor~~
- org.jruby.ast
- Truffle
- tree-sitter/tree-sitter
- sisshiki1969/ruruby
- natalie-lang/natalie
- ruby/prism

Fractured ecosystem

Ruby syntax trees

- ~~NODE, RubyVM::AbstractSyntaxTree~~
- ~~Ripper::SexpBuilder{,PP}~~
- ~~ruby-syntax-tree/syntax_tree~~
- lib-ruby-parser/lib-ruby-parser
- mruby/mruby
- ~~typedruby/typedruby, sorbet/sorbet~~
- ~~whitequark/parser, rubocop/rubocop-ast~~
- ~~seattlerb/sexp_processor~~
- ~~org.jruby.ast~~
- Truffle
- tree-sitter/tree-sitter
- sisshiki1969/ruruby
- natalie-lang/natalie
- ruby/prism

Fractured ecosystem

Ruby syntax trees

- ~~NODE, RubyVM::AbstractSyntaxTree~~
- ~~Ripper::SexpBuilder{,PP}~~
- ~~ruby-syntax-tree/syntax_tree~~
- lib-ruby-parser/lib-ruby-parser
- mruby/mruby
- ~~typedruby/typedruby, sorbet/sorbet~~
- ~~whitequark/parser, rubocop/rubocop-ast~~
- ~~seattlerb/sexp_processor~~
- ~~org.jruby.ast~~
- ~~Truffle~~
- tree-sitter/tree-sitter
- sisshiki1969/ruruby
- natalie-lang/natalie
- ruby/prism

Fractured ecosystem

Ruby syntax trees

- ~~NODE, RubyVM::AbstractSyntaxTree~~
- ~~Ripper::SexpBuilder{,PP}~~
- ~~ruby-syntax-tree/syntax_tree~~
- lib-ruby-parser/lib-ruby-parser
- mruby/mruby
- ~~typedruby/typedruby, sorbet/sorbet~~
- ~~whitequark/parser, rubocop/rubocop-ast~~
- ~~seattlerb/sexp_processor~~
- ~~org.jruby.ast~~
- ~~Truffle~~
- tree-sitter/tree-sitter
- sisshiki1969/ruruby
- ~~natalie-lang/natalie~~
- ruby/prism

Fractured ecosystem

Ruby syntax trees

- lib-ruby-parser/lib-ruby-parser
- mruby/mruby

- tree-sitter/tree-sitter
- sisshiki1969/ruruby

- ruby/prism

Fractured ecosystem

Ruby syntax trees

- lib-ruby-parser/lib-ruby-parser
- mruby/mruby
- tree-sitter/tree-sitter
- sisshiki1969/ruruby
- ruby/prism

~~Fractured~~ ecosystem

Ruby syntax trees

- lib-ruby-parser/lib-ruby-parser
- mruby/mruby
- tree-sitter/tree-sitter
- sisshiki1969/ruruby
- ruby/prism

ecosystem

Ruby syntax trees

- lib-ruby-parser/lib-ruby-parser
- mruby/mruby
- tree-sitter/tree-sitter
- sisshiki1969/ruruby
- ruby/prism

Separate understandings

Separate understandings

- Every tool represents Ruby differently

Separate understandings

- Every tool represents Ruby differently
- Every tool requires contributors to relearn

Separate understandings

- Every tool represents Ruby differently
- Every tool requires contributors to relearn
- None of this knowledge translates to contributing to CRuby

Maintenance burden

Maintenance burden

- Every parser has to update every time new syntax is introduced

Maintenance burden

- Every parser has to update every time new syntax is introduced
- Very minor changes have very major consequences

Maintenance burden

- Every parser has to update every time new syntax is introduced
- Very minor changes have very major consequences
- Helping Ruby can inadvertently hurt Ruby tooling

What is the future of prism?

The future of prism

The future of prism

- Improved error tolerance and messaging

The future of prism

- Improved error tolerance and messaging
- Performance enhancements (SIMD instructions, arena allocation, etc.)

The future of prism

- Improved error tolerance and messaging
- Performance enhancements (SIMD instructions, arena allocation, etc.)
- Multi-version support

The future of prism

- Improved error tolerance and messaging
- Performance enhancements (SIMD instructions, arena allocation, etc.)
- Multi-version support
- Further integrations into the ecosystem

The future of prism

- Improved error tolerance and messaging
- Performance enhancements (SIMD instructions, arena allocation, etc.)
- Multi-version support
- Further integrations into the ecosystem
- Improved Ruby library for external tools

The future of prism

- Improved error tolerance and messaging
- Performance enhancements (SIMD instructions, arena allocation, etc.)
- Multi-version support
- Further integrations into the ecosystem
- Improved Ruby library for external tools
- Contributor community around a single tool

Thank you!