Should we rewrite OpenVox server in plain ruby?

more about Ruby concurrency than you ever wanted to know

Marcus

2026-02-03

table of contents

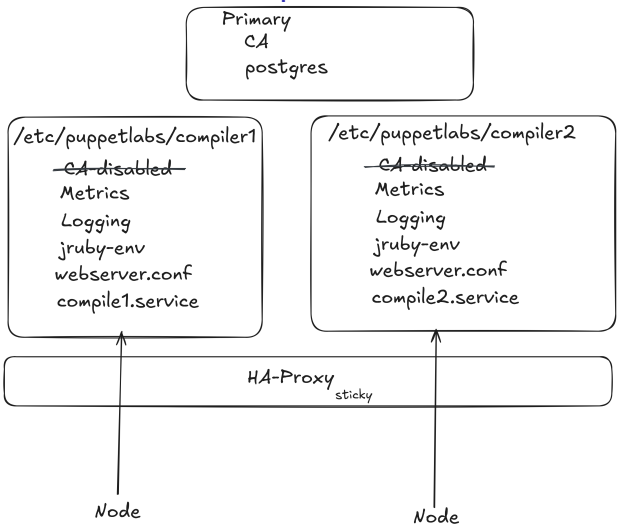
confessions

- ~~compiler OpenVox server~~
- ~~ruby programmer~~
- ~~architect~~
- rambling guy

spoiler

- I: No! pain points vs benefits
- Ben: Is this your “lets ditch the JVM presentation?”
- random: Yes, absoluteley
- Martin: Yes, given the proper architecture

spoiler


Primary
CA
postgres
/etc/puppetlabs/compiler1
CA disabled
Metrics
Logging
jruby-env
webserver.conf
compile1.service
/etc/puppetlabs/compiler2
CA disabled
Metrics
Logging
jruby-env
webserver.conf
compile2.service
HA-Proxy
sticky
Node
Node

survey

- ruby hello world
- ruby4
- process vs thread
- actor pattern
- jruby?
- debuged JVM

requirements

- 100 CPU, 1 TByte RAM
- 1 Primary, 10 Compiler, 10_000 Nodes

theory and examples

- coroutines - Ruby Fibers - Haskell lazy evalulation
- processes
- threads
- actors

Coroutines

enumerator Demo (1)

Processes

parallel demo (2)


PASSPORT CONTROL
PASSPORT CONTROL

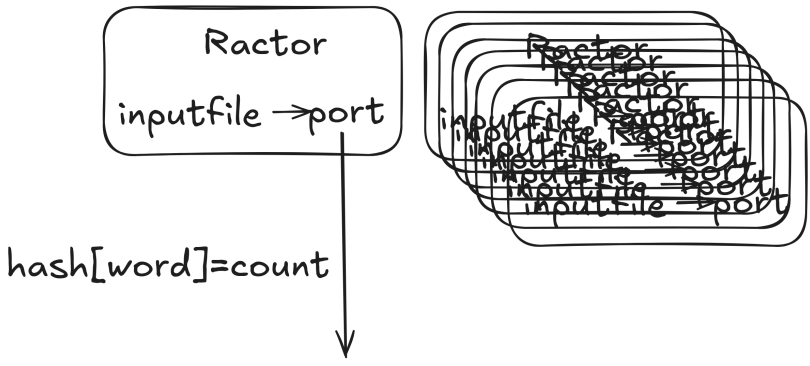
Threads

crunch demo (3)

Actors

Demo Ractor word count (4)

Ractor word count



implementation introduction

- Interpreter (MRI, jruby)
- Libaries (parallel, concurrent)

every possible interpreter

RVM Demo (6)

Crunch demo (3)

MRI

Yukihiro Matsumoto - matz
Matz Ruby Interpreter (MRI)

- ~~threads~~
- Ractors
- ~~concurrent-ruby~~
- YJIT
- Ruby::Box `https://rorindia.com/blog/ruby-box-the-game-changing-isolation-feature-in-ruby-4-0`

JRuby

- mixin java libraries (7)
- language standard 3.1
- crunch.rb: threads, fast (3)

JVM limits(8)

JVM limits cont.

- max 32 JRubies
- max 8 GByte per Alloc
- max 2 GByte reserved code cache

```
JAVA_ARGS="-XX:ReservedCodeCacheSize=2048m
```

source: https://dev.to/betadots/scaling-puppet-infrastructure-3p2o#performance-tuning-single-node

Speed comparison

	C	MRI-4 no thread	MRI-4 JIT no thread	Truffle-33 threads	MRI-4 JIT ractor	JRuby-10 threads
seconds	0.092	16.72	10.527	0.691	2.487	3.463
slowdown	1	181.7	114.4	7.5	27.0	37.6

Library summary

- Parallel
- Concurrent

https://github.com/ruby-concurrency/concurrent-ruby/blob/master/lib/concurrent-ruby/concurrent/hash.rb#L17-L23

```
HashImplementation = case
  when Concurrent.on_cruby?
    # Hash is not fully thread-safe on CRuby, see
    # https://bugs.ruby-lang.org/issues/19237
    # https://github.com/ruby/ruby/commit/ffd52412ab
    # https://github.com/ruby-concurrency/concurrent-ruby/issu
    # So we will need to add synchronization here (similar to
    ::Hash
```

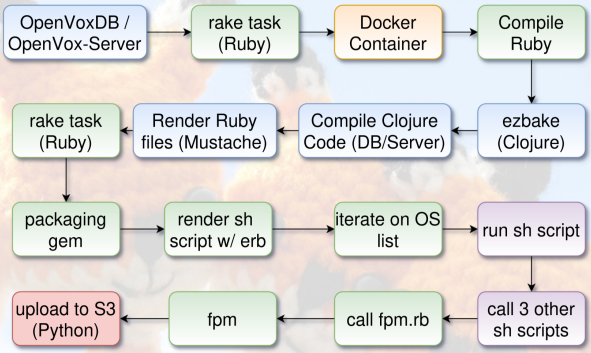
current architecture – rumors

- ezbake vs nick/tim `https://clojars.org/search?q=org.openvoxproject`
- perforce: compiled software easier to sell
- jruby performs better
- clojure + jruby
  `https://github.com/OpenVoxProject/openvox-server/blob/main/src/clj/puppetlabs/puppetserver/certificate_authority.clj`

OpenVoxProject

Where are we?

ezbake



Config Management Camp

Building OpenVox Server E2E

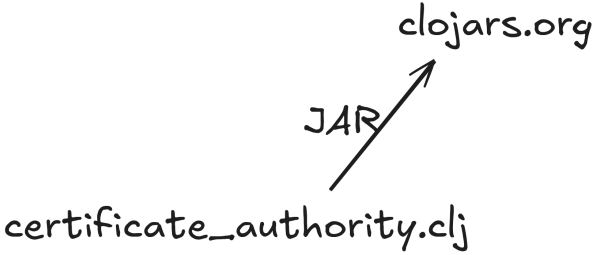
certificate_authority.clj

Building OpenVox Server E2E

JAR

certificate_authority.clj

Building OpenVox Server E2E



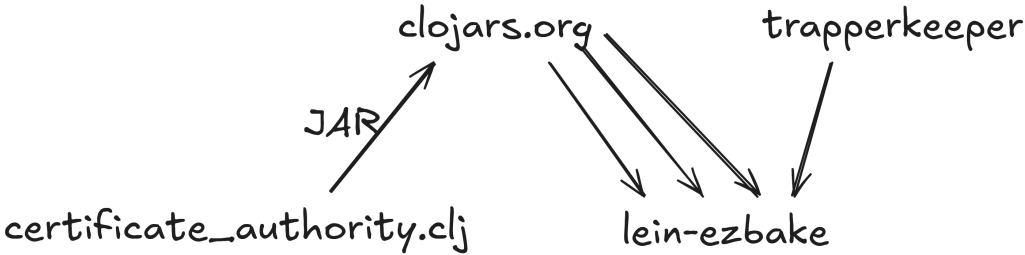
Building OpenVox Server E2E

clojars.org

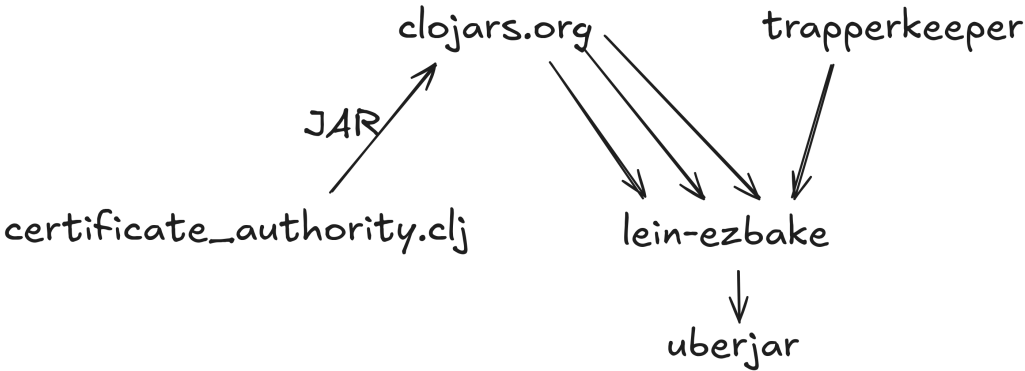
trapperkeeper

certificate_authority.clj

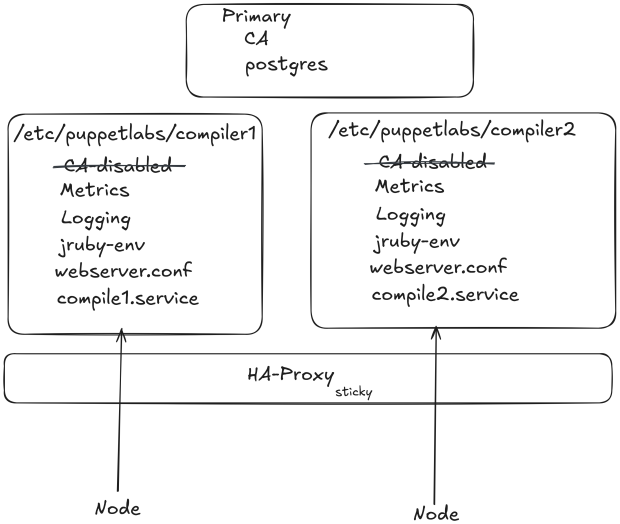
Building OpenVox Server E2E



Building OpenVox Server E2E



target architecture MRI



additional features

- template rendering time
- ~~ERB~~
- ~~Scope~~
- ~~EzBake~~
- agent facts parallel
- agent facts disable
- MRI box scope

summary

- open community discussion
- jruby pain points
- clojure pain points
- few developers - domain specific knowledeg
- ruby is portable, libraries are not
- emphasis on scalability/speed?

sources and additional links

- https://dev.to/betadots/scaling-puppet-infrastructure-3p2o#performance-tuning-single-node
- https://github.com/ruby-concurrency/concurrent-ruby/blob/master/lib/concurrent-ruby/concurrent/hash.rb#L17-L23
- https://github.com/OpenVoxProject/ezbake/tree/dev
- https://clojars.org/search?q=org.openvoxproject