Rails Girls

WHAT GEM
WHY GEM
HOW GEM

OR

SHARING YOUR CODE AS RUBYGEMS

AUSTIN ZIEGLER

GITHUB.COM/HALOSTATUE

RUBYIST SINCE 2002

19+ GEMS

19+ GEMS

114M+ DOWNLOADS

WHAT IS A GEM?

DOWNLOADABLE RUBY CODE

YOUR CODE

+

A GEMSPEC

GEMSPEC = GEM METADATA

```ruby
Gem::Specification.new do |spec|
  s.name          = "marlowe"
  s.version       = "1.0.1"
  s.authors       = ["Trevor Oke", "Kinetic Cafe ALDO Team"]
  s.email         = ["toke@kineticcafe.com", "dev@kineticcafe.com"]
  s.description   = %q{Provides a correlation …}
  s.summary       = %q{Correlation IDs for Rails and other Rack applications}
  s.homepage      = "https://github.com/KineticCafe/marlowe"
  s.license       = "MIT"

  # …
end
```

WHEN PACKAGED INSTALL WITH

```
gem install marlowe
```

WHY MAKE A GEM?

REUSE YOUR OWN CODE

KINETIC CAFE
6 PRIVATE GEMS

SHARED CODE FOR ALL APPS

NOT PUBLISHED
ON RUBYGEMS.ORG

GEMINABOX
+ OPENSHIFT
+ RUBYGEMS-QUICKSTART
= FREE PRIVATE GEM SERVER

OR

USE GEMFURY

SHARE YOUR CODE

OTHER PEOPLE MIGHT FIND YOUR CODE USEFUL

PUBLISHED ON RUBYGEMS.ORG

OPEN SOURCE

HOW TO MAKE A GEM?

GEMS ARE EASY

CREATE GEMSPEC

BUILD GEM

RELEASE GEM

HARDEST PART:

HARDEST PART: THE IDEA

HARDEST PART:
THE IDEA

HARDEST PART: THE NAME

THE BEST GEMS ARE FROM WORKING CODE

TOOLS FOR BUILDING GEMS

TOOL: BUNDLER

bundler gem GEM

WHY BUNDLER?

- UBIQUITOUS
- YOU ALREADY HAVE IT

```
bundler gem GEM [OPTIONS]
-b, [--bin], [--no-bin]
```

```
bundler gem GEM [OPTIONS]
-b, [--bin], [--no-bin]
```

GENERATE A BINARY FOR YOUR LIBRARY.

```
bundler gem GEM [OPTIONS]
[--coc], [--no-coc]
```

```
bundler gem GEM [OPTIONS]
[--coc], [--no-coc]
```

GENERATE A CODE OF CONDUCT FILE.

```
bundler gem GEM [OPTIONS]
[--ext], [--no-ext]
```

```
bundler gem GEM [OPTIONS]
[--ext], [--no-ext]
```

GENERATE THE BOILERPLATE FOR C EXTENSION CODE

```
bundler gem GEM [OPTIONS]
[--mit], [--no-mit]
```

```
bundler gem GEM [OPTIONS]
[--mit], [--no-mit]
```

GENERATE AN MIT LICENSE FILE.

```
bundler gem GEM [OPTIONS]
-t, [--test=rspec]
```

```
bundler gem GEM [OPTIONS]
-t, [--test=rspec]
```

GENERATE A TEST DIRECTORY FOR YOUR LIBRARY.

EITHER RSPEC OR MINITEST.

WHY NOT BUNDLER?

- MOST USEFUL FOR THE SIMPLEST GEMS
- NOT DRY
- QUESTIONABLE DEFAULTS

TOOL: HOE

```
gem install hoe
sow GEM
```

WHY HOE

- BUILT FOR GEM DEVELOPERS
- FULL DEVELOPMENT WORKFLOW. VERY DRY
- PLUG-IN BASED
- USES A MANIFEST

SOW GEM

sow --style TEMPLATE GEM

- TEMPLATES ARE IN `~/.hoe_template`.
- ONLY ONE BY DEFAULT. COPIED WHEN FIRST RUN.
- QUIK MAY BE BETTER THAN SOW. BUT USES HOE.

FULL DEVELOPMENT WORKFLOW BUILT AROUND RAKE

FULL DEVELOPMENT WORKFLOW

`rake` (TESTS)

`rake docs`

`rake dcov` (DOC COVERAGE)

`rake package` (BUILDS)

FULL DEVELOPMENT WORKFLOW

`rake release VERSION=x.y.z`

- WILL NOT RELEASE UNLESS `VERSION=x.y.z` MATCHES YOUR GEM VERSION

FULL DEVELOPMENT WORKFLOW

ALSO HAS TASKS FOR ANNOUNCEMENTS, BLOG POSTS, ETC.

VERY DRY

PARSES README AND HISTORY FILES

FINDS `VERSION` IN YOUR SOURCE CODE

PLUG-IN BASED
LOTS OF PLUG-INS

ADD OTHERWISE MISSING FEATURES:

- `hoe-doofus` (RELEASE CHECKLIST)
- `hoe-gemspec2` (CREATE `GEM.gemspec`)
- `hoe-git` (CHANGELOG, MANIFEST, AND TAG MANAGEMENT)
- `hoe-travis` (TRAVIS-CI BUILD SUPPORT)
- `hoe-geminabox` (PRIVATE GEM SERVER SUPPORT)
- MORE...

WHY NOT HOE?

- EXISTING TEMPLATES ARE IDIOSYNCRATIC.
- SHARING TEMPLATES IS HARD.
- MOST USEFUL FOR SIMPLE-TO-MEDIUM COMPLEXITY GEMS.
- VERY OPINIONATED.

NOTE: MORE TOOLS

- JEWELER
- RUBYGEMS-TASKS

LET'S

RELEASE

A

GEM

(DEMO)

Q&A

FURTHER WATCHING

PLUCK IT

ADAM CUPPY

RUBYCONF 2015

SPLIT YOUR CODE INTO MICRO-LIBRARIES RELEASED AS RUBYGEMS.

PROBLEM
RUBYCONF

SEVEN HABITS OF HIGHLY EFFECTIVE RUBYGEMS

MAT BROWN

SEVEN GOOD IDEAS FOR MAKING AND MAINTAINING GOOD RUBYGEMS. I ATTENDED THIS TALK AND IT'S WORTH A LISTEN.

4.2.5
Breaking change
New feature
Bugfix
Semantic Versioning
RUBYCONF

HOW DOES BUNDLER WORK, ANYWAY?

ANDRE ARKO

HOW DOES BUNDLER WORK? WHY DO WE NEED `bundle exec`?

only absolute paths?

probably fine too 🙈