Ember OSF: A Demo App

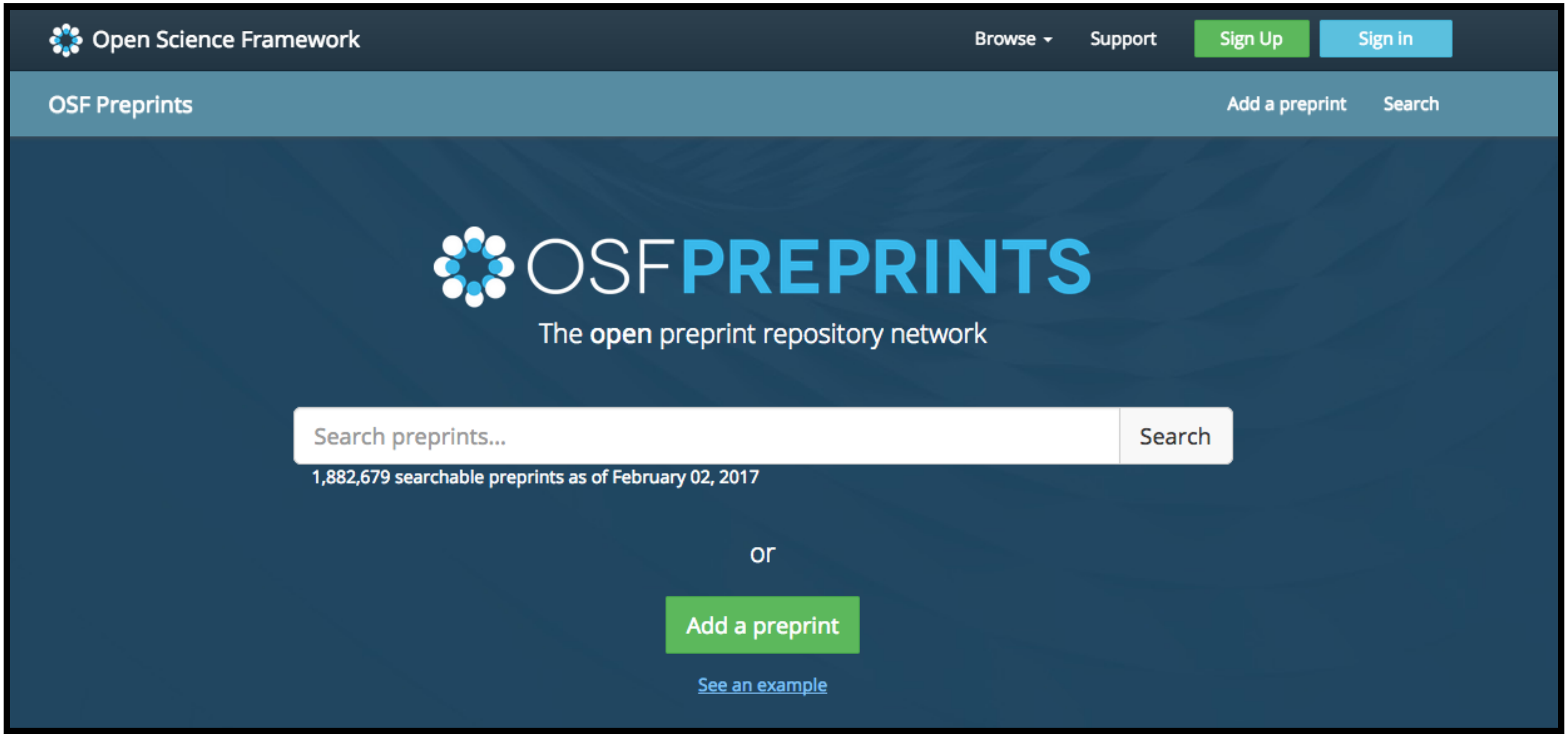
Or, ”how to eat your own dogfood”

Andy Boughton / Center for Open Science

First, the addon

Addon goals

- Help build ember apps that use OSF functionality, fast
- Be usable by COS developers and external collaborators alike
- Highly (re)usable, flexible components that demonstrate best practices



The **open** preprint repository network

1,882,679 searchable preprints as of February 02, 2017

Dogfooding

> *"...a slang term used to reference a scenario in which a company uses its own product to test and promote the product.[1]"*

1. Definition
2. In 2013: "25% of Yahoo Employees used Yahoo Mail"

How are we doing?

Inside COS we have started to use this for many projects

This is not easy to get started! We should learn from, and improve, the setup experience- can't tell partners to "just copy the preprints repo".

Need to go beyond the "bundle of components" for a true project starter kit.

Pain Points I

We are actively working to fix some of the challenges:

- Installation- should support NPM / ember install
- Secret settings
- baked-in config

```
"ember-osf": "git+https://github.com/CenterForOpenScience/ember-osf.git#ac254f
```

Pain Points II

Some challenges remain open items, and still require manual boilerplate to resolve. (especially UI related assets that not everyone needs)

- Users need to manually install certain bower assets
- ...And manually import key files in various places
- Useful features aren't always well known; the docs have room to improve

A path forward

- Don't be shy about breaking out smaller packages. (But remember left-pad)
  - Models (just talk to the server)
  - Components (if you also need UI)
- Use automated addon mechanisms for importing code
  - Addons can control what happens during build
  - Blueprints can control events after install
- Know what's possible! https://ember-cli.com/api/

Uh, so what?

https://github.com/abought/demo-ember-osf

Batteries included

Don't just document all the steps- help people get started.

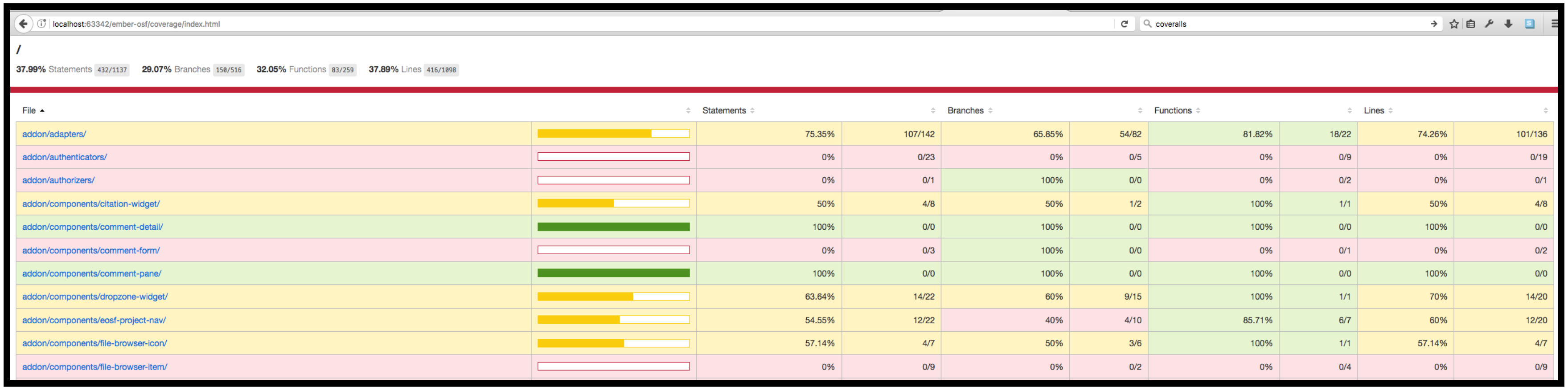
- Boilerplate: LICENSE, README, make the addon work
- Tools: Yarn, Sass, NVM, YUIDoc
- Best practices: Travis, linters, code coverage, Sentry. Gitignore your secrets and standardize your commands
- Nice pages: dynamic page titles, styling
- Demo tricky stuff: login, authenticated requests, (someday) pagination
- Push the envelope: Automatic accessibility testing

Thinking collectively

- Focus on shared problems and shared solutions
- Coordinate releases across repos
- Clear path to deprecation- don't remove stuff blindly
- Best practices: unit tests, docs, and code coverage

Code coverage: Knowing where to test

Ember encourages granular tests. Code coverage tells you which ones to write.



/

37.99% Statements 432/1137 **29.07%** Branches 150/516 **32.05%** Functions 83/259 **37.89%** Lines 416/1098

File ▲		Statements		Branches		Functions		Lines	
addon/adapters/		75.35%	107/142	65.85%	54/82	81.82%	18/22	74.26%	101/136
addon/authenticators/		0%	0/23	0%	0/5	0%	0/9	0%	0/19
addon/authorizers/		0%	0/1	100%	0/0	0%	0/2	0%	0/1
addon/components/citation-widget/		50%	4/8	50%	1/2	100%	1/1	50%	4/8
addon/components/comment-detail/		100%	0/0	100%	0/0	100%	0/0	100%	0/0
addon/components/comment-form/		0%	0/3	100%	0/0	0%	0/1	0%	0/2
addon/components/comment-pane/		100%	0/0	100%	0/0	100%	0/0	100%	0/0
addon/components/dropzone-widget/		63.64%	14/22	60%	9/15	100%	1/1	70%	14/20
addon/components/eosf-project-nav/		54.55%	12/22	40%	4/10	85.71%	6/7	60%	12/20
addon/components/file-browser-icon/		57.14%	4/7	50%	3/6	100%	1/1	57.14%	4/7
addon/components/file-browser-item/		0%	0/9	0%	0/2	0%	0/4	0%	0/9

Further reading

Yes, there are docs. Really.

http://centerforopenscience.github.io/ember-osf/