Adding Sparkle to Social Coding: An Empirical Study of Repository Badges in the npm Ecosystem

Asher Trockman, Shurui Zhou, Christian Kästner, Bogdan Vasilescu

ICSE ’18, May 27-June 3, 2018, Gothenburg, Sweden
Async utilities for node and the browser  

http://caolan.github.io/async/

JavaScript  async  callbacks

1,629 commits  11 branches  72 releases  206 contributors  MIT

Async is a utility module which provides straight-forward, powerful functions for working with asynchronous JavaScript. Although originally designed for use with Node.js and installable via npm install async, it can also be used directly in the browser.
Key features: Transparency & signaling

Users

Popular repositories

- breakfast-repo
  - a collection of videos, recordings, and podcast...
  - 208 ⭐
- x86-kernel
  - a simple x86 kernel, extended with Rust
  - 48 ⭐
- ashleywilliams.github.io
  - hi, i'm ashley. nice to meet you.
  - 37 ⭐
- jsconf-2015-deck
  - deck for jsconf2015 talk, "if you wish to learn e...
  - 32 ⭐
- ratpack
  - sinatra boilerplate using activerecord, sqlite, a...
  - 32 ⭐

Repositories contributed to

- npm/docs
  - The place where all the npm docs live.
  - 44 ⭐
- mozilla/publish.webmaker.org
  - The teach.org publishing service for goggles a...
  - 2 ⭐
- ashleywilliams.github.io
  - hi, i'm ashley. nice to meet you.
  - 37 ⭐
- jsconf-2015-deck
  - deck for jsconf2015 talk, "if you wish to learn e...
  - 32 ⭐
- ratpack
  - sinatra boilerplate using activerecord, sqlite, a...
  - 32 ⭐

Summary of pull requests, issues opened, and commits. Learn how we count contributions.

- Contributions in the last year: 1,886 total
- Longest streak: 37 days
  - October 7 – November 12
- Current streak: 7 days
  - January 18 – January 24

Projects
Key features: Transparency & signaling

Simplified HTTP request client.

2,199 commits 17 branches 134 releases 270 contributors

Popular repositories
- breakfast-repo: a collection of videos, recordings, and podcast...
- x86-kernel: a simple x86 kernel, extended with Rust
- ashleywilliams.github.io: hi, i’m ashley. nice to meet you.
- jsconf-2015-deck: deck for jsconf2015 talk, "if you wish to learn e...
- ratpack: sinatra boilerplate using activerecord, sqlite, a...

Repositories contributed to
- npm/docs: The place where all the npm docs live.
- mozilla/publish.webmaker.org: The teach.org publishing service for goggles a...
- ashleywilliams.github.io: hi, i’m ashley. nice to meet you.
- jsconf-2015-deck: deck for jsconf2015 talk, "if you wish to learn e...
- ratpack: sinatra boilerplate using activerecord, sqlite, a...

Public contributions

Contributions in the last year
- 1,886 total
- Longest streak
- 37 days
- October 7 – November 12
- Current streak
- 7 days
- January 18 – January 24

Summary of pull requests, issues opened, and commits. Learn how we count contributions.

Projects

Users

Followers 38 Starred 15 Following

Follow

Follow, Follow

376

Key features: Transparency & signaling

Users

Projects

Organizations

Carnegie Mellon University
Badges are Reliable Signals

- **build** passing of the presence of **tests**
- **dependencies** up to date of **up-to-date** and secure **dependencies**
- **coverage** 94% of the presence of **tests in pull requests**
- **downloads** 654/month of **popularity**
Mixed methods study

Survey

- 32 maintainers, 57 contributors
- Maintainers:
  - What do you intend to signal?
  - What effects do you expect?
- Contributors:
  - What do badges tell you?

Repository Mining

- 294,941 npm packages
- Mined badge adoptions/removals from README files
- Measured proxies for code quality, test suite quality, popularity, dependency freshness, …
Popular Badges in npm

- build passing
- release v2.1.1
- dependencies up to date
- coverage 94%
- downloads 654/month
- code climate 4.0
- license BSD
- code style standard
- git join chat

Percent of packages

- 0% - 10%
- 10% - 20%
- 20% - 30%
- 30% - 40%
Popular Badges in npm

- **build**: passing
- **release**: v2.1.1
- **dependencies**: up to date
- **coverage**: 94%
- **downloads**: 654/month
- **code climate**: 4.0
- **license**: BSD
- **code style**: standard
- **gitter**: join chat

Percent of packages
Popular Badges in npm

- build passing
- release v2.1.1
- dependencies up to date
- coverage 94%
- downloads 654/month
- code climate 4.0
- license BSD
- code style standard
- jitter join chat

Percent of packages

10%  20%  30%
Popular Badges in npm

- build: passing
- release: v2.1.1
- dependencies: up to date
- coverage: 94%
- downloads: 654/month
- code climate: 4.0
- license: BSD
- code style: standard
- git: join chat

Percent of packages
What do developers expect from badges?

- 32 Maintainers
  - What do you intend to signal?
  - What effects do you expect?
- 57 Contributors
  - What do badges tell you?

“welcoming contributions”
“expectations of contribution quality”
“dedicated to offering support”
“reduced chances of conflicting versions of dependencies”

“indicator of product quality”
Analysis

Correlation
If all you saw was the badge, how much would that tell you?

Regression Analysis
How much more does the badge tell you, relative to existing signals?

Time Series Analysis
How do things change after adding the badge?
If all you saw was the badge, how much would that tell you?

How much more does the badge tell you, relative to existing signals?

How do things change after adding the badge?
Step 1: Correlation

Signals of fresh dependencies

- **Based on survey:** The adoption of dependency management badges correlates with fresher dependencies
- **Freshness metric:** lower is better
  - (More up-to-date deps.)

**Result:** Dep. badges correlate with fresher dependencies
Analysis

Correlation
If all you saw was the badge, how much would that tell you?

Regression Analysis
How much more does the badge tell you, relative to existing signals?

Time Series Analysis
How do things change after adding the badge?
Step 2: Regression Analysis

Signals of fresh dependencies

- **Based on survey:** The adoption of dependency management badges correlates with fresher dependencies
  - **Freshness metric:** *lower* is better
  - (More up-to-date deps.)

**Result:** Dep. badges are *the best* signals of fresh dependencies

### Data

<table>
<thead>
<tr>
<th>Coeffs (Err.)</th>
<th>LR Chisq</th>
<th>Coeffs (Err.)</th>
<th>LR Chisq</th>
</tr>
</thead>
<tbody>
<tr>
<td>(Inter.)</td>
<td>3.54 (0.03)***</td>
<td>3.50 (0.03)***</td>
<td></td>
</tr>
<tr>
<td>Dep.</td>
<td>-1.78 (0.01)***</td>
<td>32077.8***</td>
<td>-1.79 (0.01)***</td>
</tr>
<tr>
<td>RDep.</td>
<td>0.22 (0.01)***</td>
<td>610.3***</td>
<td>0.21 (0.01)***</td>
</tr>
<tr>
<td>Stars</td>
<td>-0.08 (0.00)***</td>
<td>301.4***</td>
<td>-0.09 (0.00)***</td>
</tr>
<tr>
<td>Contr.</td>
<td>-0.24 (0.01)***</td>
<td>500.5***</td>
<td>-0.25 (0.01)***</td>
</tr>
<tr>
<td>lastU</td>
<td>-0.65 (0.01)***</td>
<td>12080.9***</td>
<td>-0.64 (0.01)***</td>
</tr>
<tr>
<td>dependencies</td>
<td>up to date</td>
<td>0.24 (0.03)***</td>
<td>116.1***</td>
</tr>
<tr>
<td>npm</td>
<td>v1.1.0</td>
<td>0.11 (0.02)***</td>
<td>48.3***</td>
</tr>
<tr>
<td>dependencies</td>
<td>up to date</td>
<td>-0.05 (0.04)</td>
<td>1.9</td>
</tr>
<tr>
<td>hasOther</td>
<td></td>
<td>0.01 (0.01)</td>
<td></td>
</tr>
</tbody>
</table>

***p < 0.001, **p < 0.01, *p < 0.05
Analysis

Correlation
If all you saw was the badge, how much would that tell you?

Regression Analysis
How much more does the badge tell you, relative to existing signals?

Time Series Analysis
How do things change after adding the badge?
Step 2: Time Series Analysis

Signals of fresh dependencies

Before Badge

After Badge

10^2

10^1

Month index relative to badge

Time

Before Badge

After Badge

Adoption Month

Badge

dependencies up to date

Carnegie Mellon University
Step 2: Time Series Analysis

Signals of fresh dependencies

Before Badge

After Badge

Month index relative to badge

Before Badge

After Badge

Time
Step 2: Time Series Analysis

Signals of fresh dependencies

Before Badge

After Badge

Month index relative to badge

Freshness

$10^1$ $10^2$

Before Badge

After Badge

Adoption

Badge

Adoption Month

dependencies up to date
Step 2: Time Series Analysis

Signals of fresh dependencies

Before Badge

After Badge

Month index relative to badge

Freshness

Before Badge

After Badge

Decrease in Level

Decrease in Slope

Time
Step 2: Time Series Analysis

Signals of fresh dependencies

- **Based on survey:** The adoption of dependency management badges correlates with fresher dependencies

- **Freshness metric:** lower is better
  - (More up-to-date deps.)

**Result:** Dep. badges indicate improved dep. management practices
Signals of popularity

**Result:** Dep. badges are mostly reliable signals of popularity
Signals of popularity

Result: Dep. badges are mostly reliable signals of popularity
Signals of popularity

Result: Dep. badges are mostly reliable signals of popularity
Signals of test suite quality

Result: Build status/code coverage badges indicate a test suite
Signals of test suite quality

Result: Build status/code coverage badges indicate a test suite
Signals of test suite quality

Result: Build status/code coverage badges indicate a test suite
Result: Build status+code coverage badges indicate *more tests in PRs*
Signals of PR quality

Result: Build status + code coverage badges indicate more tests in PRs
Result: Build status + code coverage badges indicate more tests in PRs
Take-aways

Badges with **underlying analyses:**

- build: passing
- codacy: A
- dependencies: out of date
- vulnerabilities: 0
- code climate: 4.0
- coverage: 94%
- docs: standard
- issue resolution: 3 h

are **stronger predictors** than badges that merely state intentions or provide links:

- cdnjs: v3.2.1
- license: BSD
- PRs: welcome
- code style: standard
- git: join chat
- Patreon
- commitizen: friendly
- code style: standard

}{

**assessment signals**

}
Take-aways

When possible, design or choose the badge that takes the **most work**: 

- **slack** 6/160 > **slack join**

  - assessment signal
  - conventional signal
Take-aways

When possible, design or choose the badge that takes the **most work**:

- **assessment signal**: slack 6/160
- **conventional signal**: slack join
Signals of fresh dependencies

Step 2: Time Series Analysis

Key features: Transparency & signaling

Mixed methods study

Survey + Repository Mining

- 52 maintainers, 57 contributors
- Maintainers: What do you intend to signal?
- Contributors: What do badges tell you?

- 294,941 npm packages
- Mixed badge adoptions/removals from README files
- Measured proxies for code quality, test suite quality, popularity, dependency freshness...

Take-aways

When possible, design or choose the badge that takes the most work:

- slack 6/160
- slack took

Contact:
asher.trockman@gmail.com
https://cmustrudel.github.io

badges mostly reliable
http://cmustrudel.github.io/badges