

# 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



# Open-source is everywhere

## Open source:

- “Digital dark matter”
- Companies (2015 survey):
  - 78% run OSS
  - 66% build on OSS
- Huge economic impact:
  - Apache httpd server: \$7-10b



# “Social coding” is a catalyst

## Open source:

- “Digital dark matter”
- Companies (2015 survey):
  - 78% run OSS
  - 66% build on OSS
- Huge economic impact:
  - Apache httpd server: \$7-10b

## GitHub:



27 million  
people



76 million  
repositories



# “Social coding” is a catalyst

## Open source:

- “Digital dark matter”
- Companies (2015 survey):
  - 78% run OSS
  - 66% build on OSS
- Huge economic impact:
  - Apache httpd server: \$7-10b

## GitHub:



27 million people



76 million repositories

## Worldwide estimates:



22 million software dev's (2017)

# Key feature: Transparency

[request / request](#)

Watch 417 Star 18,384 Fork 2,196

Code Issues 578 Pull requests 52 Projects 0 Wiki Insights

Simplified HTTP request client.

2,199 commits 17 branches 134 releases 270 contributors Apache-2.0

Branch: master New pull request Create new file Upload files Find file Clone or download

mikeal committed on Sep 27, 2017 2.83.1 Latest commit 253c5e5 on Sep 27, 2017

File	Commit Message	Date
.github	small change to template wording	a year ago
examples	Adds example for Tor proxy	2 years ago
lib	refactor(lint): replace eslint with standard (#2579)	11 months ago
tests	lint fix, PR from pre-standard was merged with passing tests	10 months ago
.gitignore	Updating deps.	5 months ago
.travis.yml	Add Node.js v8 to Travis CI	7 months ago
CHANGELOG.md	Update changelog	11 months ago
CONTRIBUTING.md	Update contributing guidelines	2 years ago
LICENSE	Adding license information.	7 years ago

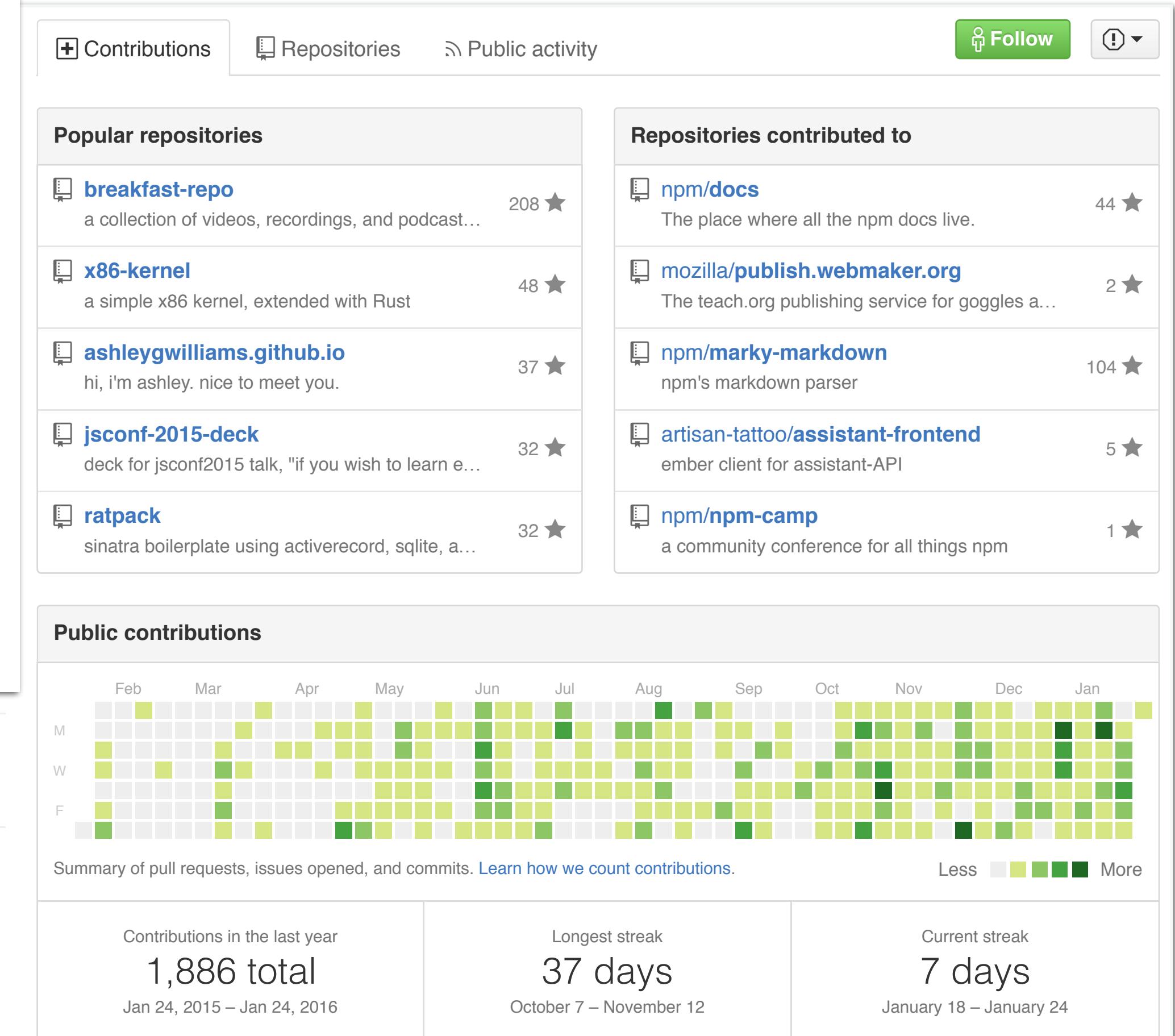
## Projects

776 Followers 38 Starred 15 Following

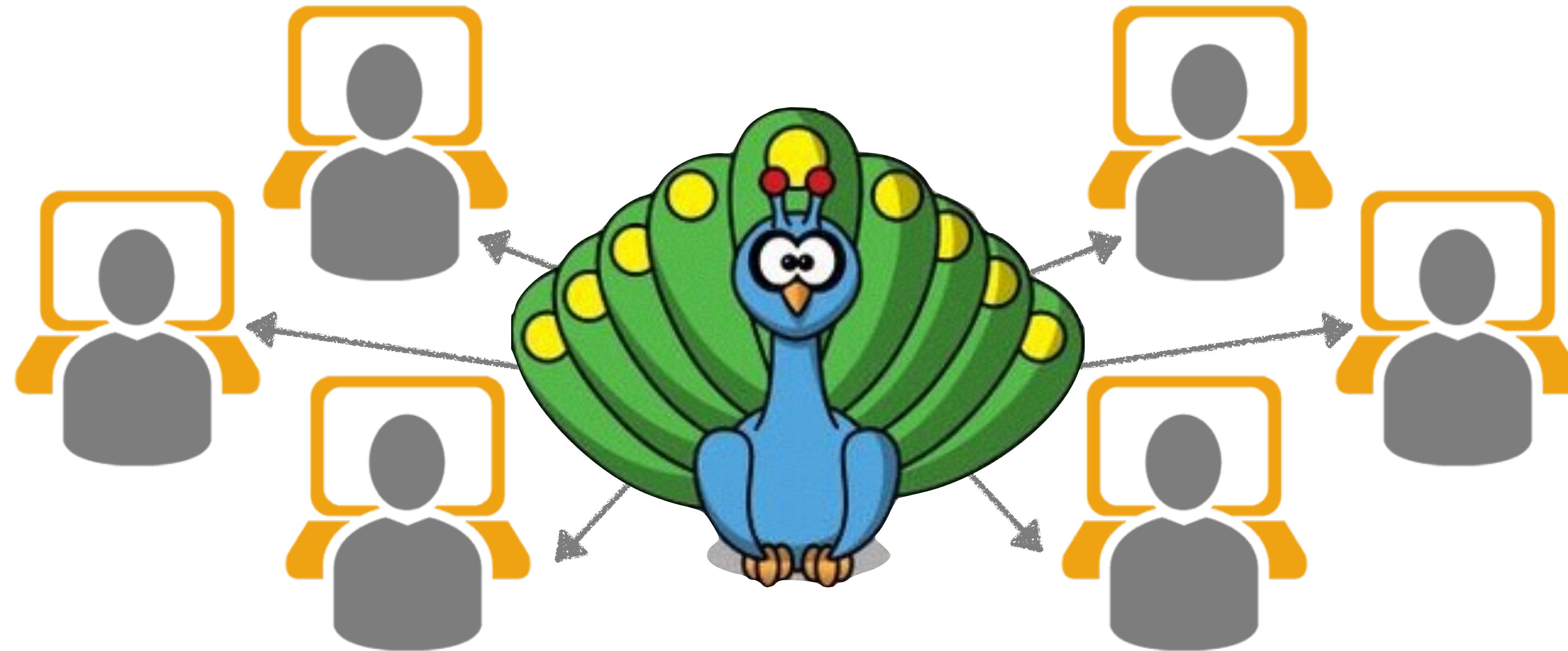
Organizations



## People



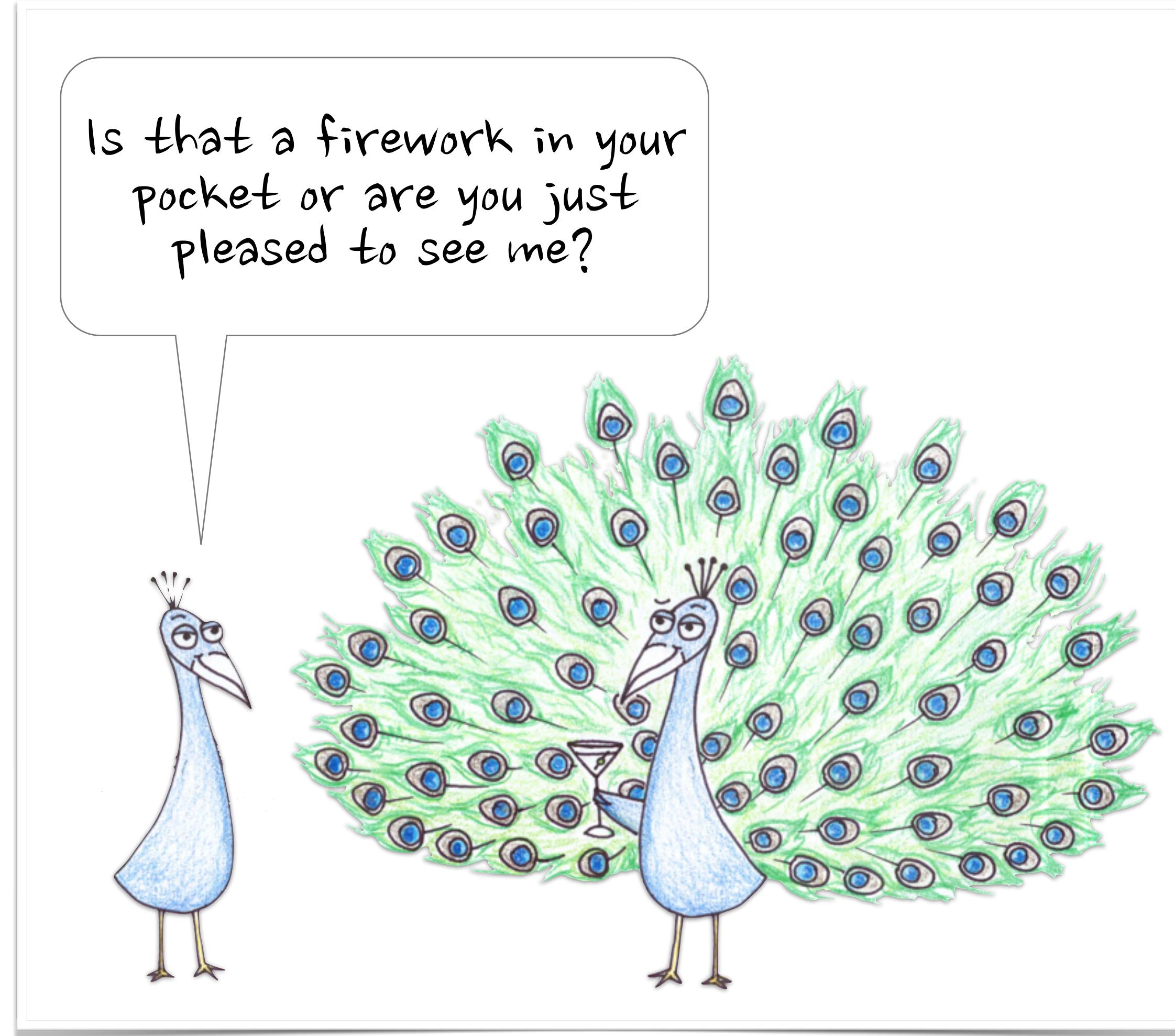
# The “social programmer” ... signals



- Assessing technical candidates on the social web  
A Capiluppi, A Serebrenik, L Singer. *IEEE Software* 2013
- Mutual assessment in the social programmer ecosystem: an empirical investigation of developer profile aggregators  
L Singer, F Figueira Filho, B Cleary, C Treude, MA Storey, K Schneider. *CSCW* 2013

- Impression formation in online peer production: activity traces and personal profiles in GitHub  
J Marlow, L Dabbish, J Herbsleb. *CSCW* 2013
- Activity traces and signals in software developer recruitment and hiring  
J Marlow, L Dabbish. *CSCW* 2013

# Today: Signaling with badges



# Badges are customizable signals

The screenshot shows the GitHub repository page for the 'request / request' project. A vertical double-headed arrow on the left side indicates the transition from 'Built-in (GitHub)' at the top to 'Custom' at the bottom.

**Built-in (GitHub):** This section shows the standard GitHub repository interface. It includes the GitHub logo, navigation links like 'This repository' and 'Search', and tabs for 'Code', 'Issues (523)', 'Pull requests (40)', 'Projects (0)', 'Wiki', and 'Insights'. At the top right, there are buttons for 'Watch' (395), 'Star' (16,836), 'Fork' (2,023), and notifications. Below the tabs, it displays metrics: 2,190 commits, 17 branches, 130 releases, 273 contributors, and Apache-2.0 licensing.

**Custom:** This section shows a modified version of the repository page where the built-in GitHub UI has been replaced by a custom design. The title 'Request - Simplified HTTP client' is prominently displayed. Below it is an 'npm' badge with the text 'npm install request' and '1,275 ★'. At the bottom, there are status indicators: 'build passing', 'coverage 92%', 'coverage 93%', 'dependencies up to date', 'vulnerabilities 0', and 'gitter join chat'.

# Types of badges



# Types of badges

- Quality assurance
  - Build status, test coverage, static analysis, ...

build passing

ember observer 8 / 10

 build passing

 bitHound 97

codacy A

coverage 53%

build passing

coverage 94%

code climate 4.0

build passing

docs 

# Types of badges

- Quality assurance
  - Build status, test coverage, static analysis, ...
- Dependency management
  - Version tracking, vulnerability tracking, ...

dependencies out of date

vulnerabilities 0

dependencies insecure

dependencies up to date

Greenkeeper enabled

# Types of badges

- Quality assurance
  - Build status, test coverage, static analysis, ...
- Dependency management
  - Version tracking, vulnerability tracking, ...
- Information
  - *npm* version, license, coding style, release strategy, commit message conventions, ...

npm v1.1.0

code style standard

license BSD

cdnjs v3.2.1

bower v3.1.4

IRC irc.freenode.net#unshift

commitizen friendly

version 4.2.1

release v2.1.1

made by Protocol Labs

code style standard

Deploy to Heroku

semantic-release

# Types of badges

- Quality assurance
  - Build status, test coverage, static analysis, ...
- Dependency management
  - Version tracking, vulnerability tracking, ...
- Information
  - *npm* version, license, coding style, release strategy, commit message conventions, ...
- Popularity
  - *npm* downloads, GitHub stats, Twitter, ...

 Star  4k

 downloads 654/month

 Follow  350

 Forks  847

# Types of badges

- Quality assurance
  - Build status, test coverage, static analysis, ...
- Dependency management
  - Version tracking, vulnerability tracking, ...
- Information
  - *npm* version, license, coding style, release strategy, commit message conventions, ...
- Popularity
  - *npm* downloads, GitHub stats, Twitter, ...
- Support
  - chat & collaboration, issue stats, ...

 codementor [GET HELP ▶](#)

issue resolution 3 h

slack [join](#)

IRC <irc.freenode.net#unshift>

gitter [join chat](#)

commitizen [friendly](#)

slack 6/160

# Types of badges

- Quality assurance
  - Build status, test coverage, static analysis, ...
- Dependency management
  - Version tracking, vulnerability tracking, ...
- Information
  - *npm* version, license, coding style, release strategy, commit message conventions, ...
- Popularity
  - *npm* downloads, GitHub stats, Twitter, ...
- Support
  - chat & collaboration, issue stats, ...
- Misc:
  - Paypal, donations, Gittip, ...

ember observer 8 / 10



tips \$3.64/week



made by Protocol Labs

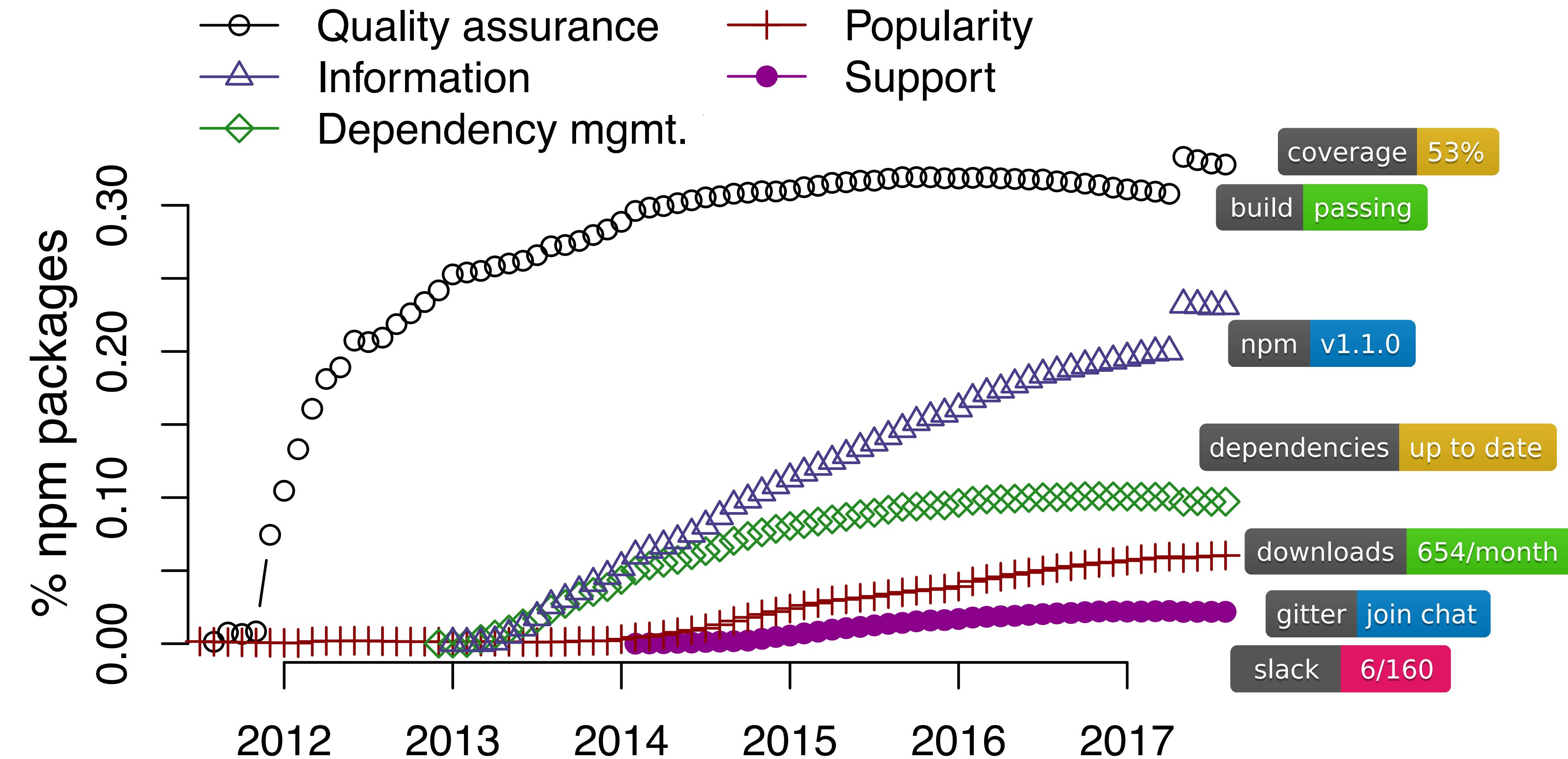
tips \$1.45/week



# Popular badges in



- Only few badges are broadly adopted
- Badges tend to be adopted in groups and hardly ever change



# Research questions

- How are badges used?
- What do they tell about a project?
- How much can you trust them?



# Signaling theory (Spence, 1973)

- Badges are signals:
  - reduce information asymmetry
- Conventional signals vs assessment signals
  - **assessment signals:** more costly to produce —> more reliable

# Signaling theory (Spence, 1973)

- Badges are signals:
  - reduce information asymmetry
- Conventional signals vs assessment signals
  - assessment signals: more costly to produce —> more reliable
- Badges vary widely in production cost
  - Expensive: coverage 94% vulnerabilities 0 dependencies up to date build passing
  - Cheap: release v2.1.1 npm v1.1.0 license BSD
  - No cost: code style standard PRs welcome

made by Protocol Labs

# Mixed methods study



+



# Mixed methods study

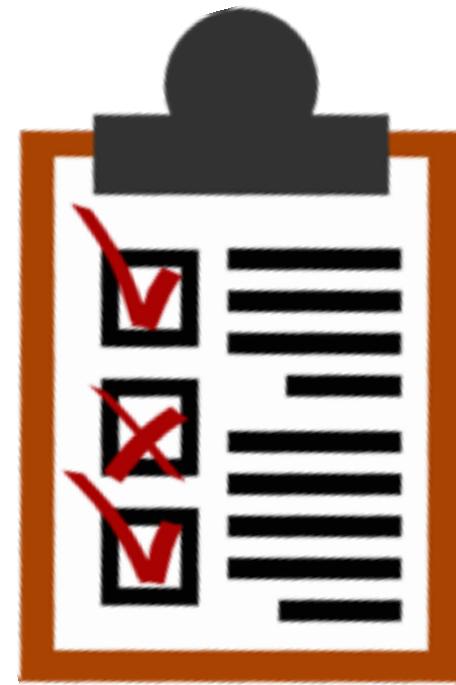


+



- 32 maintainers, 57 contributors  
(15% resp. rate)
- Maintainers:
  - What do you intend to signal?
  - What effects do you expect?
- Contributors:
  - What do badges tell you?

# Mixed methods study



+



- 32 maintainers, 57 contributors (15% resp. rate)
- Maintainers:
  - What do you intend to signal?
  - What effects do you expect?
- Contributors:
  - What do badges tell you?
- 294,941 *npm* packages
- Mined badge adoptions/removals from README files
- Measured proxies for code quality, test suite quality, popularity, dependency freshness, ...

# Analysis



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?

# Analysis



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

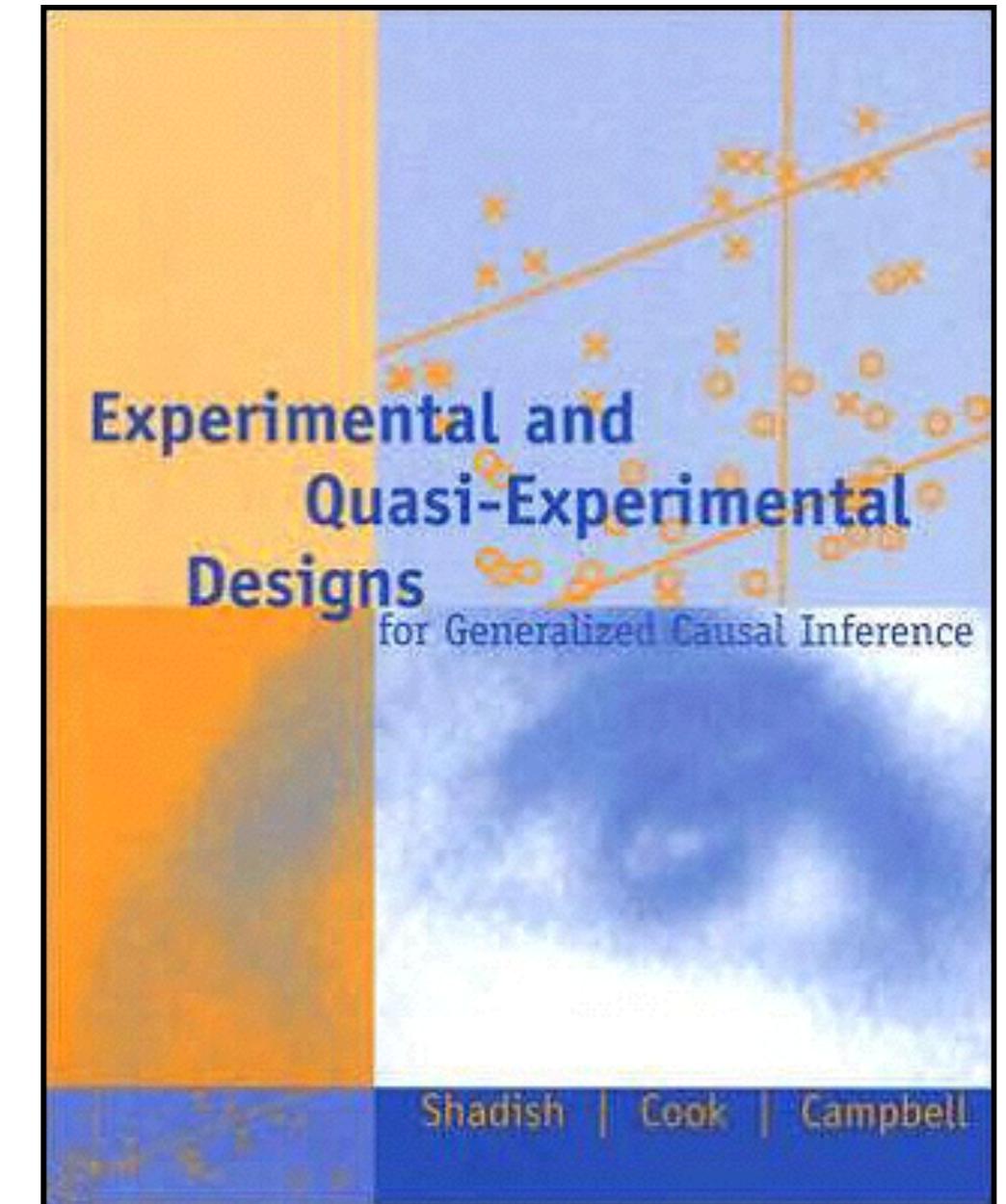
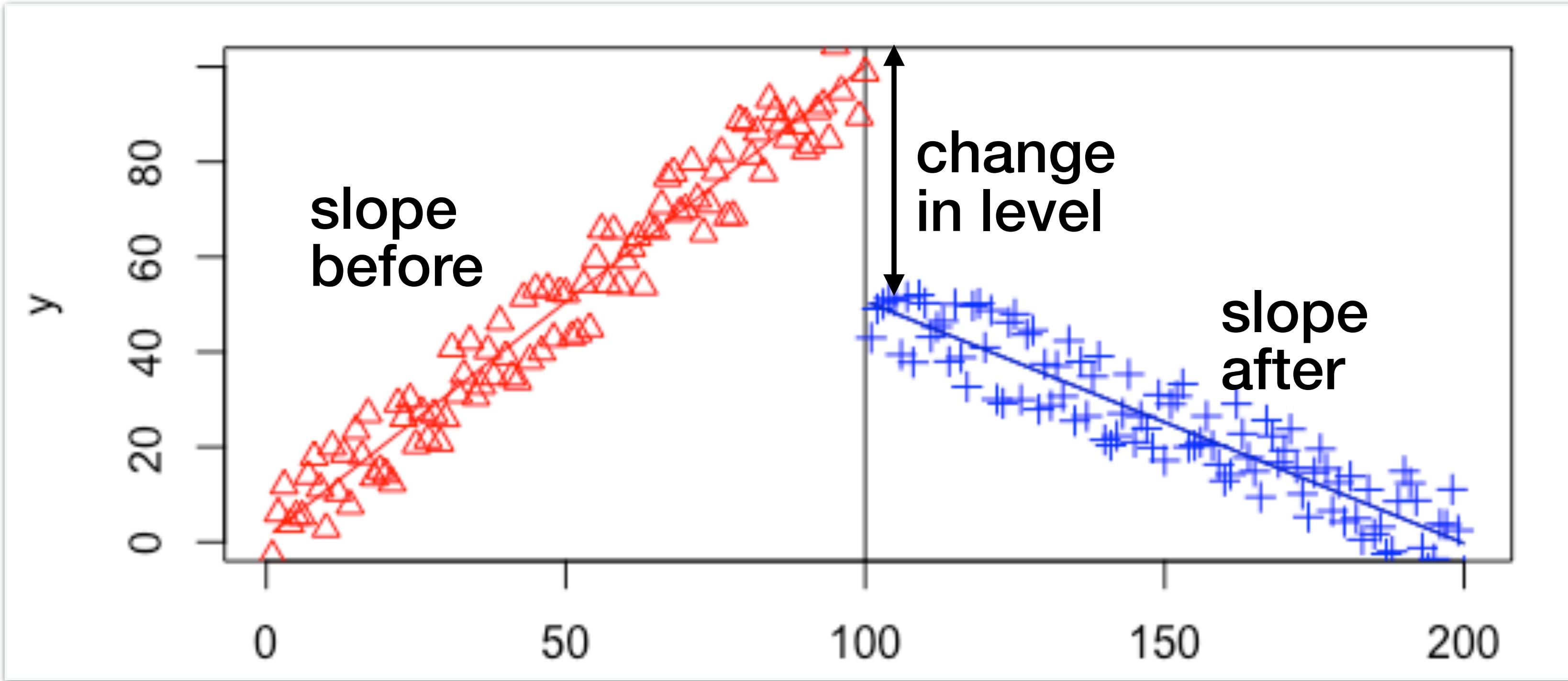


How much more does the badge tell you, relative to existing signals?

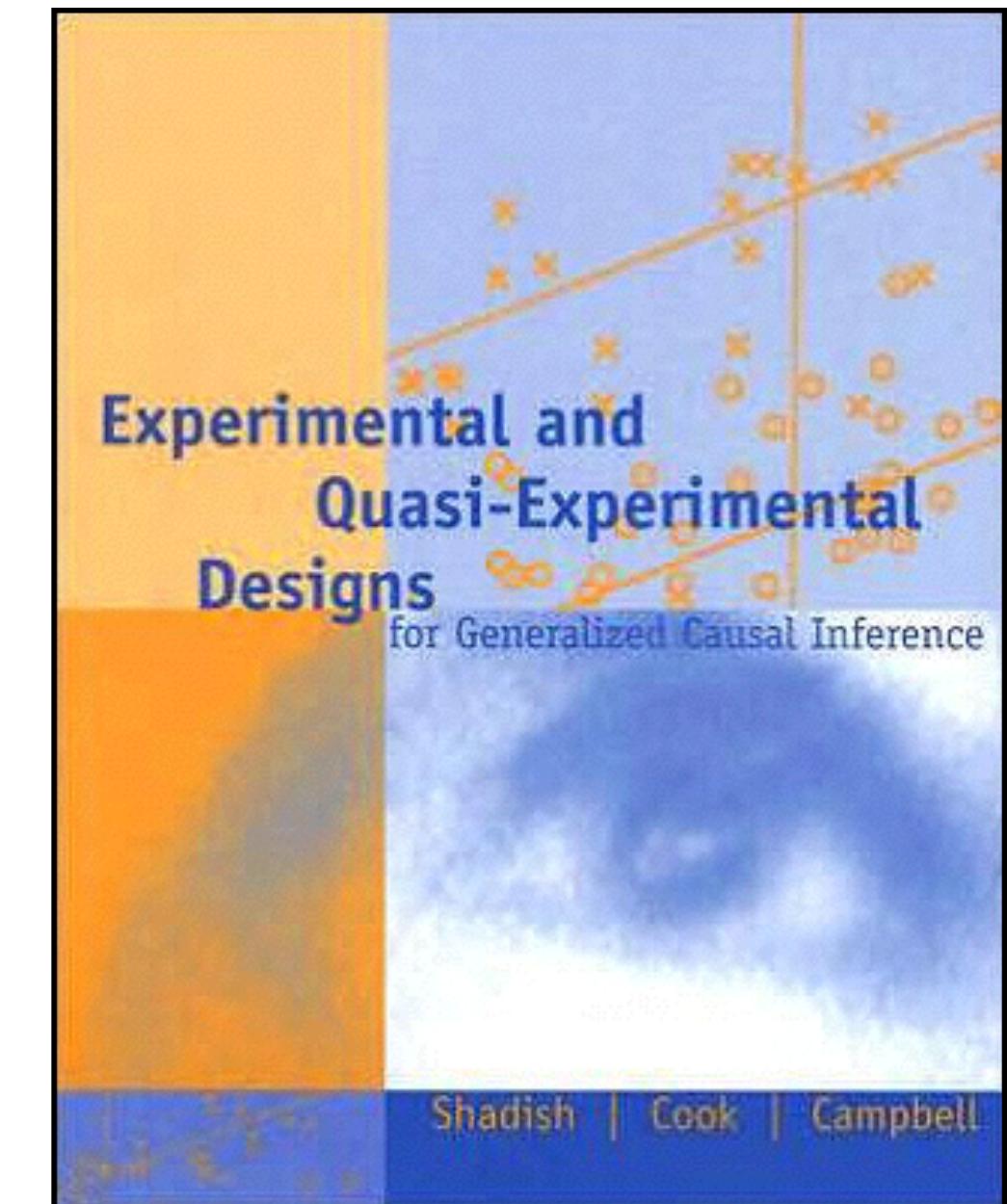
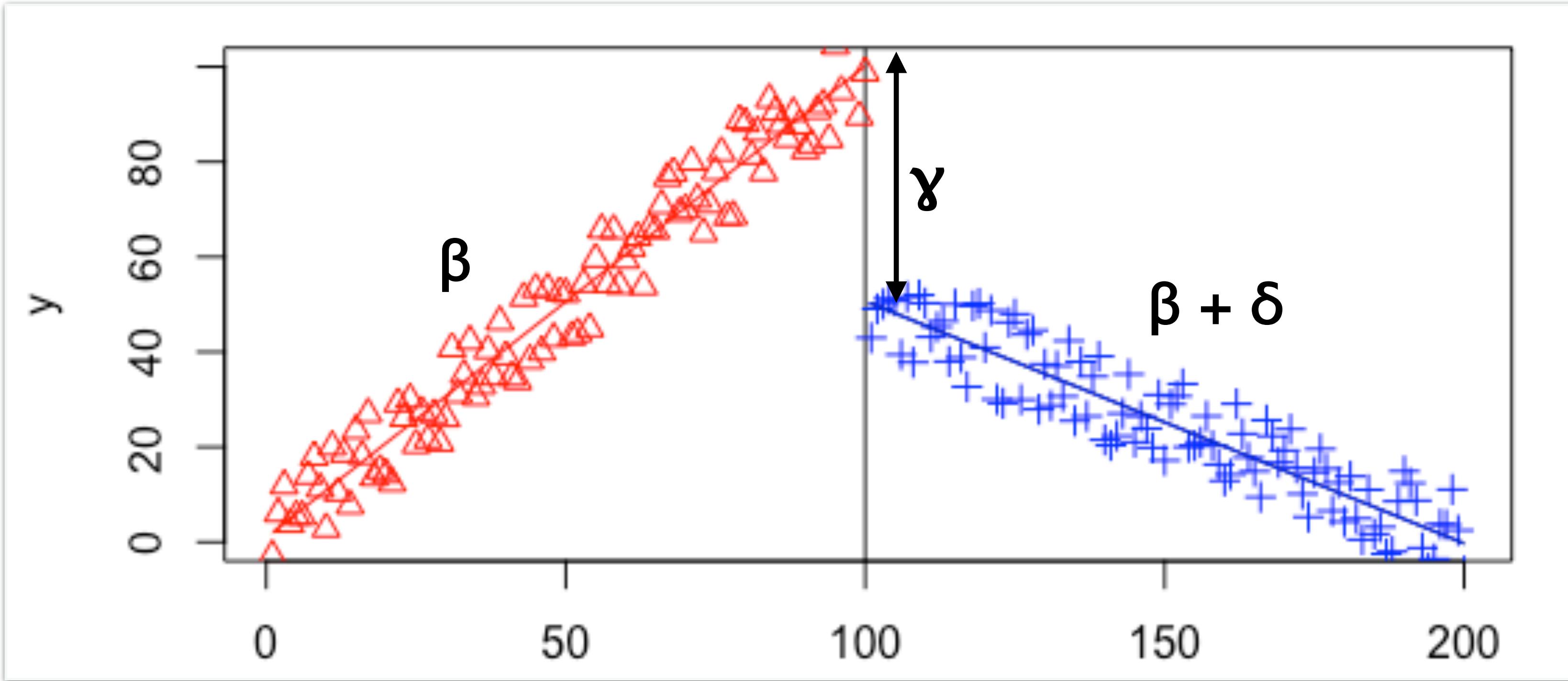


How do things change after adding the badge?

# Interrupted time series



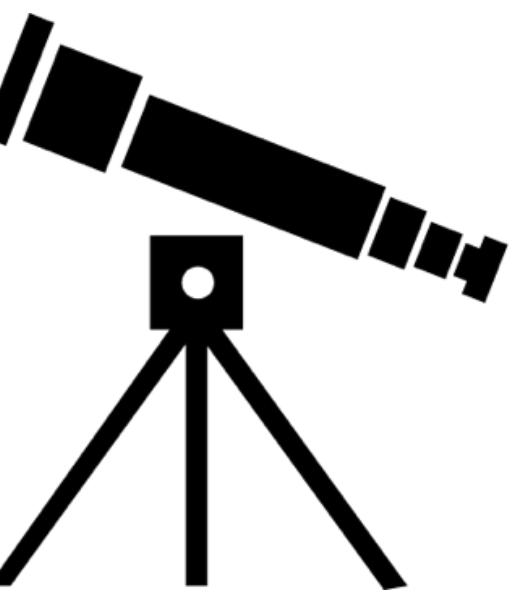
# Interrupted time series



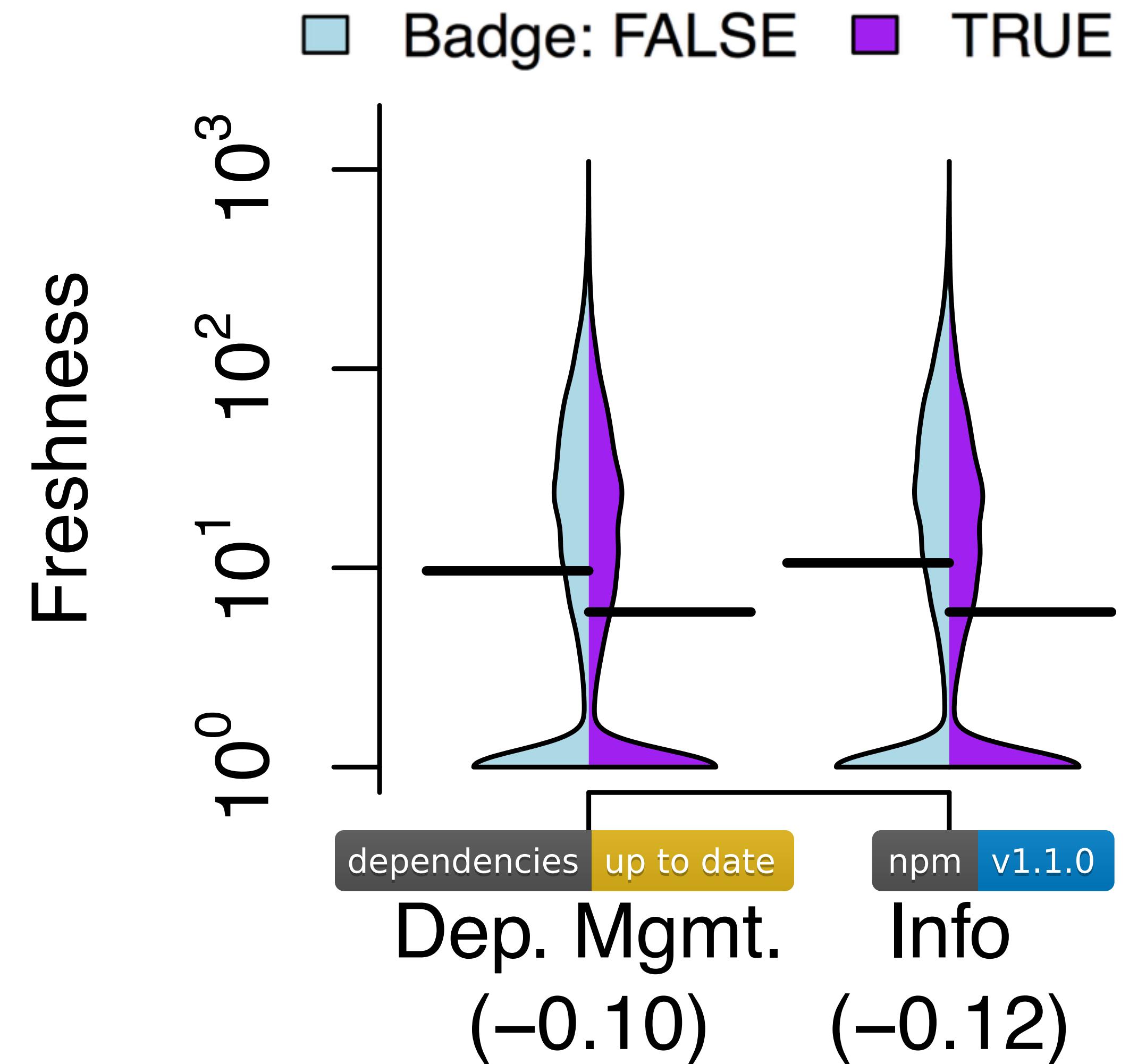
$$y_i = \alpha + \beta \cdot \text{time}_i + \gamma \cdot \text{intervention}_i + \delta \cdot \text{time\_after\_intervention}_i + \varepsilon_i$$

time:	1	2	3	...	...	100	101	102	...	...	200
time after intervention:	0	0	0	...	...	0	1	2	...	...	100
intervention:	F	F	F	...	...	T	T	T	...	...	T

# Signals of fresh dependencies



- Hyp: The adoption of dependency-management badges correlates with fresher dependencies
- Hyp: Information badges have no effect
- In aggregate: both badges correlate with having the quality



# Signals of fresh dependencies



- Hyp: The adoption of dependency-management badges correlates with fresher dependencies
- Hyp: Information badges have no effect
- Both badges add information beyond other readily observable signals

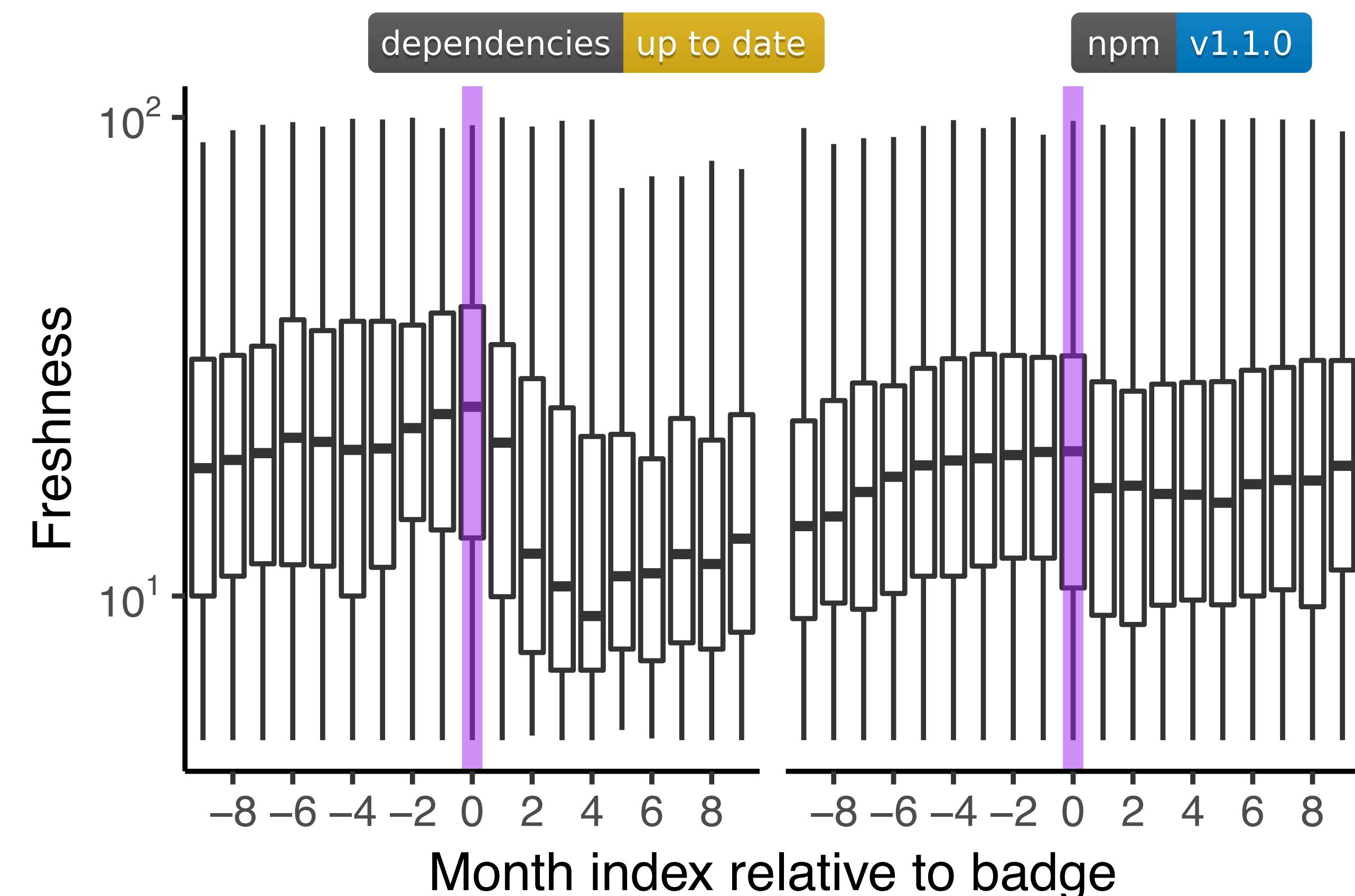
		Basic Model		Full Model	
		response: <i>freshness</i> = 0		response: <i>freshness</i> = 0	
		Coeffs (Err.)	LR Chisq	Coeffs (Err.)	LR Chisq
(Inter.)	3.54 (0.03)***			3.50 (0.03)***	
Dep.	-1.78 (0.01)***	32077.8***		-1.79 (0.01)***	32292.8***
RDep.	0.22 (0.01)***	610.3***		0.21 (0.01)***	560.6***
Stars	-0.08 (0.00)***	301.4***		-0.09 (0.00)***	311.2***
Contr.	-0.24 (0.01)***	500.5***		-0.25 (0.01)***	548.7***
lastU	-0.65 (0.01)***	12080.9***		-0.64 (0.01)***	11537.9***
dependencies up to date				0.24 (0.03)***	116.1***
npm v1.1.0				0.11 (0.02)***	48.3***
dependencies up to date : npm v1.1.0				-0.05 (0.04)	1.9
hasOther				0.01 (0.01)	

\*\*\*  $p < 0.001$ , \*\*  $p < 0.01$ , \*  $p < 0.05$

# Signals of fresh dependencies



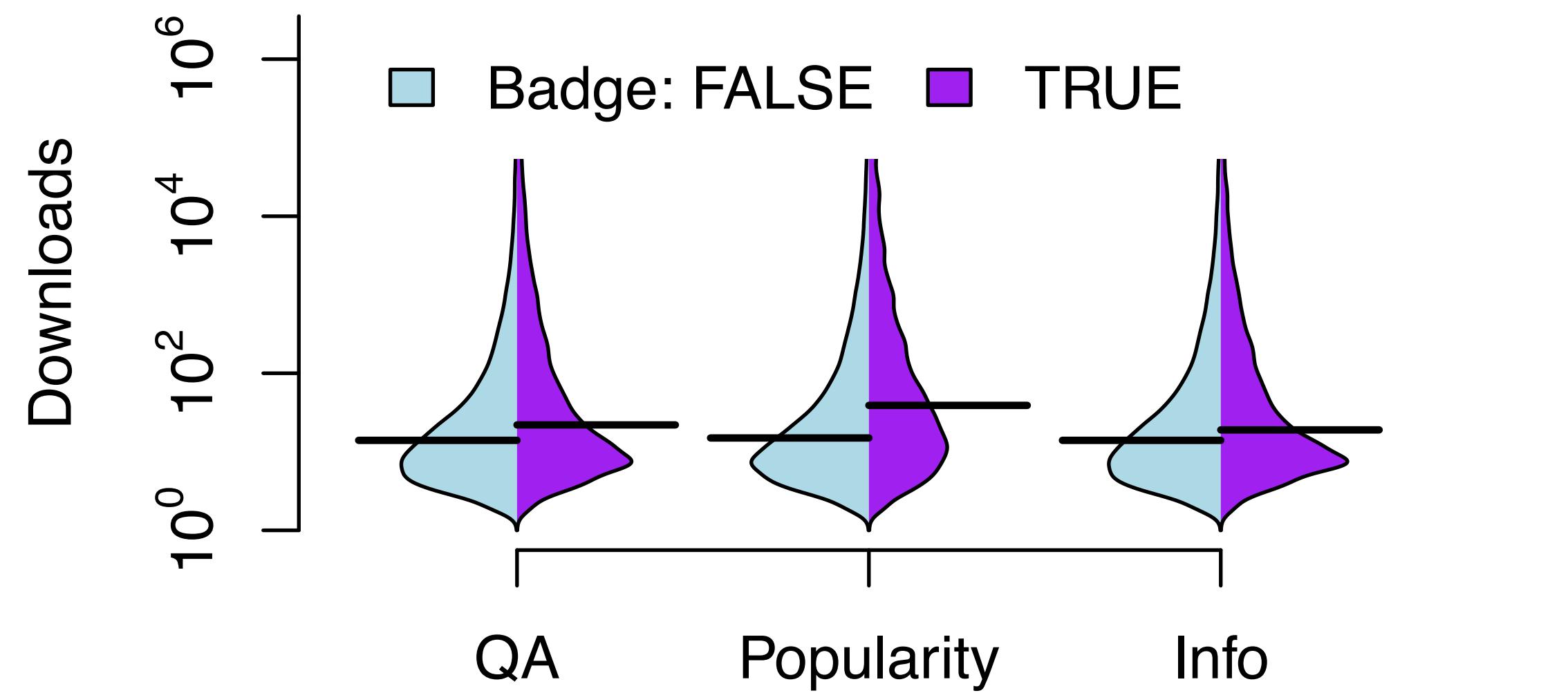
- Hyp: The adoption of dependency-management badges correlates with fresher dependencies
- Hyp: Information badges have no effect
- The adoption of DM badges is associated with a longer lasting effect



# Signals of popularity



- Hyp: The adoption of quality-assurance badges makes users more confident in a package and attracts more users
- Hyp: The adoption of popularity-related badges in popular packages correlates with more future downloads
- Packages with a badge tend to skew toward more downloads than packages without.



build passing

coverage 94%

downloads 654/month

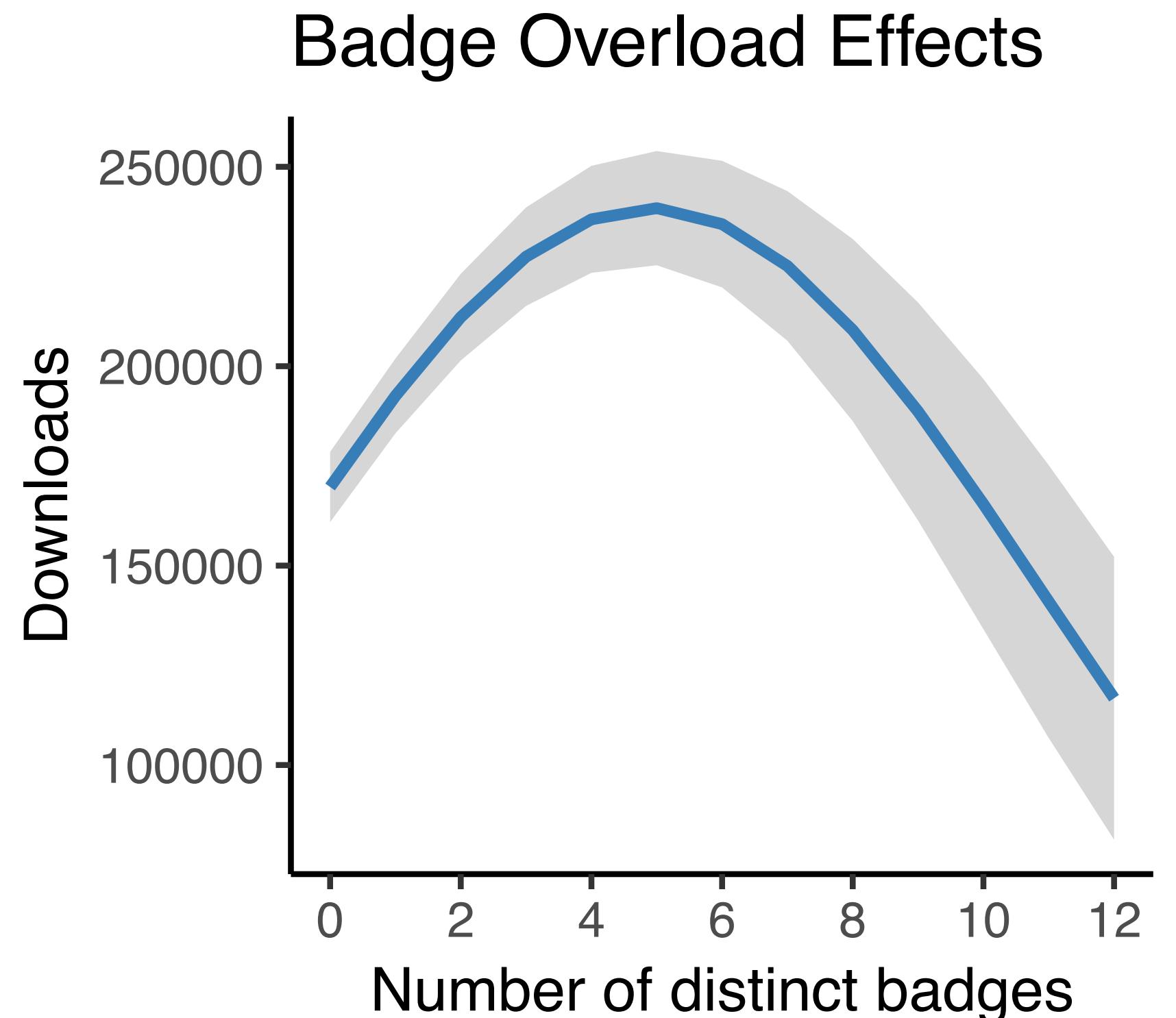
npm v1.1.0

*Code was “built with love” or “well written” by an “experienced developer” who pays “attention to quality”*

# Signals of popularity



- Hyp: The adoption of quality-assurance badges makes users more confident in a package and attracts more users
- Hyp: The adoption of popularity-related badges in popular packages correlates with more future downloads
- **Attractiveness wears off beyond five badges**

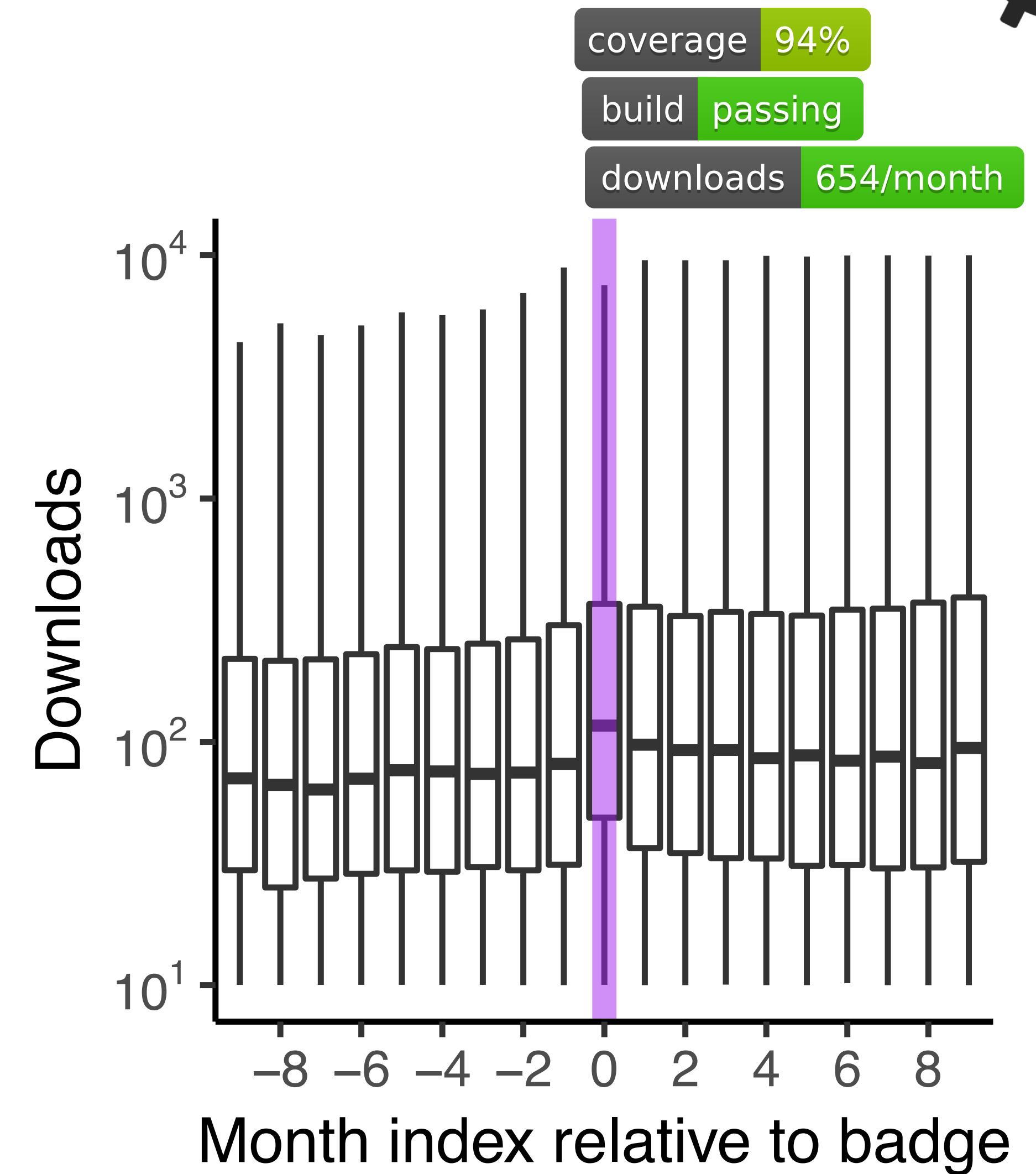


*“People tend to overwhelm visitors with too many (useless) badges, thus creating a contra effect and loosing the initial purpose of having useful information.”*

# Signals of popularity



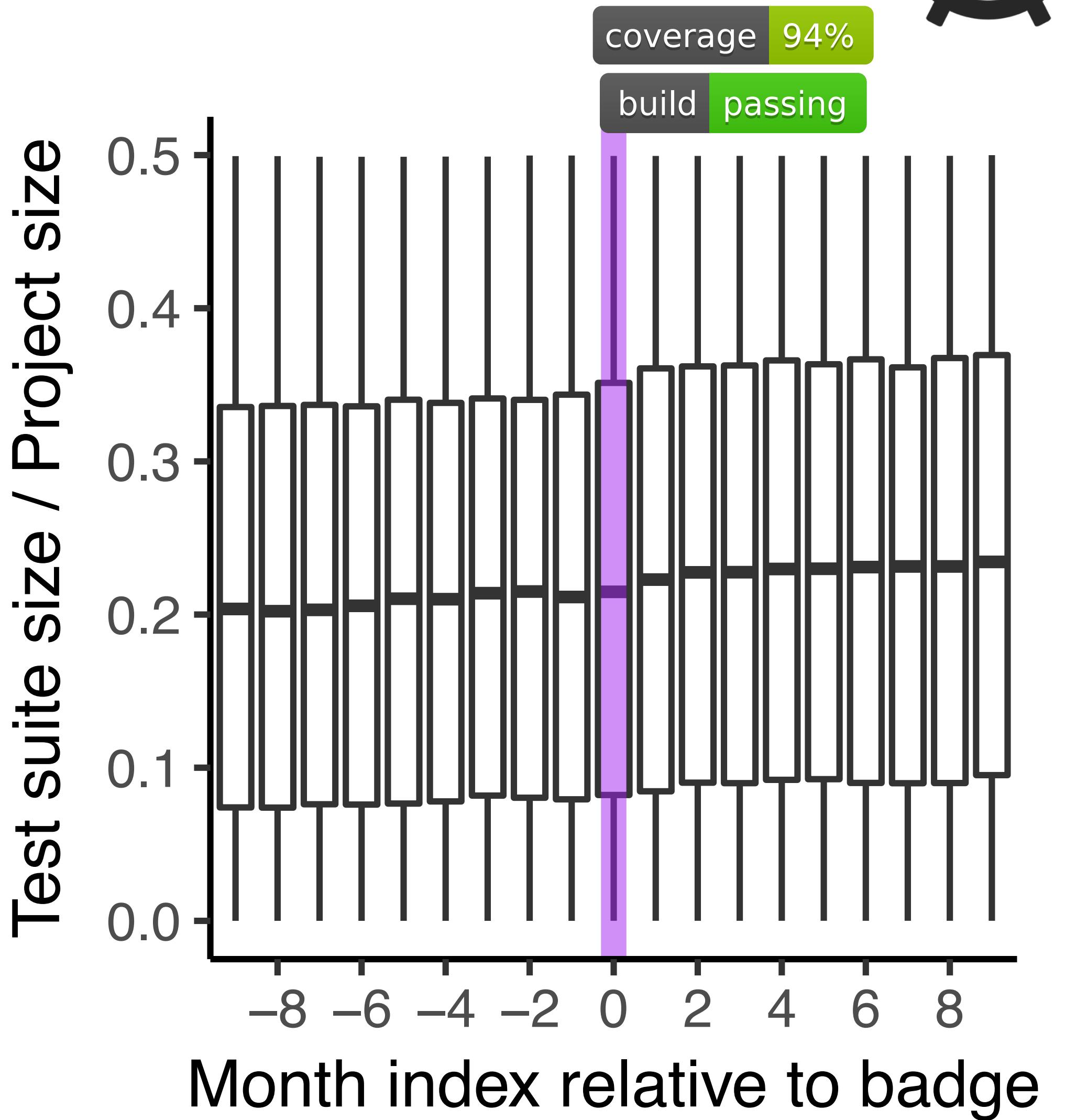
- Hyp: The adoption of quality-assurance badges makes users more confident in a package and attracts more users
- Hyp: The adoption of popularity-related badges in popular packages correlates with more future downloads
- Badge adoption correlates with a sudden popularity boost, but the acceleration is not sustained over time.



# Signals of test suite quality



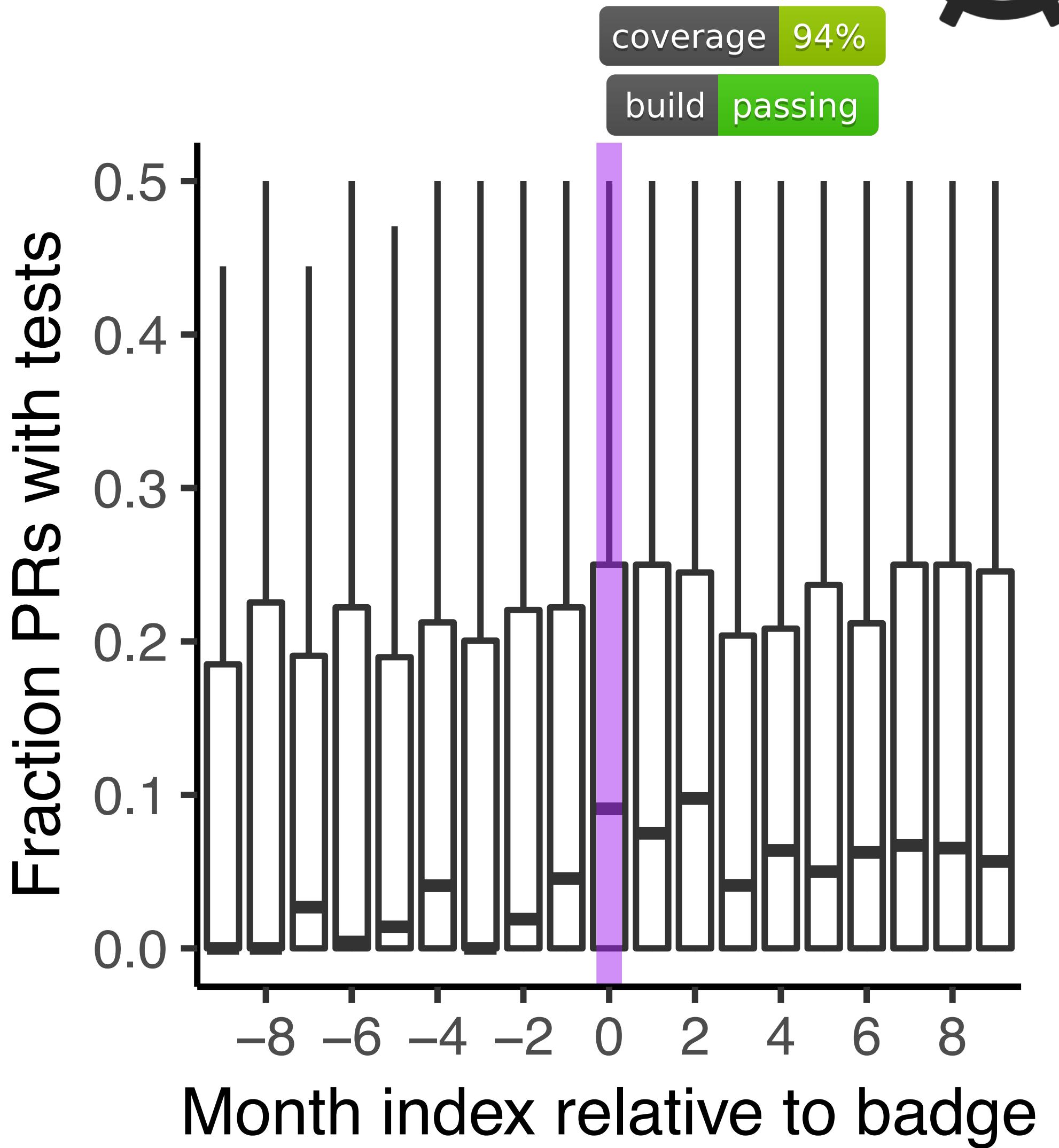
- Hyp: The adoption of quality-assurance badges correlates with other indicators of code quality (metric: test suite size).  
*Code was “built with love” or “well written” by an “experienced developer” who pays “attention to quality”*
- 18x higher odds of having any tests with QA badge; among those with tests, 18.3% larger test suite with QA badge.
- But, no change in trend



# Signals of PR quality



- Hyp: The adoption of a quality-assurance badge, and even more so of a coverage badge, encourages more external contributors to include tests.  
*"PRs with new functionality tend to include new tests, as not to decrease coverage."*
- Increase in the monthly fraction of PRs containing tests after adopting QA badge



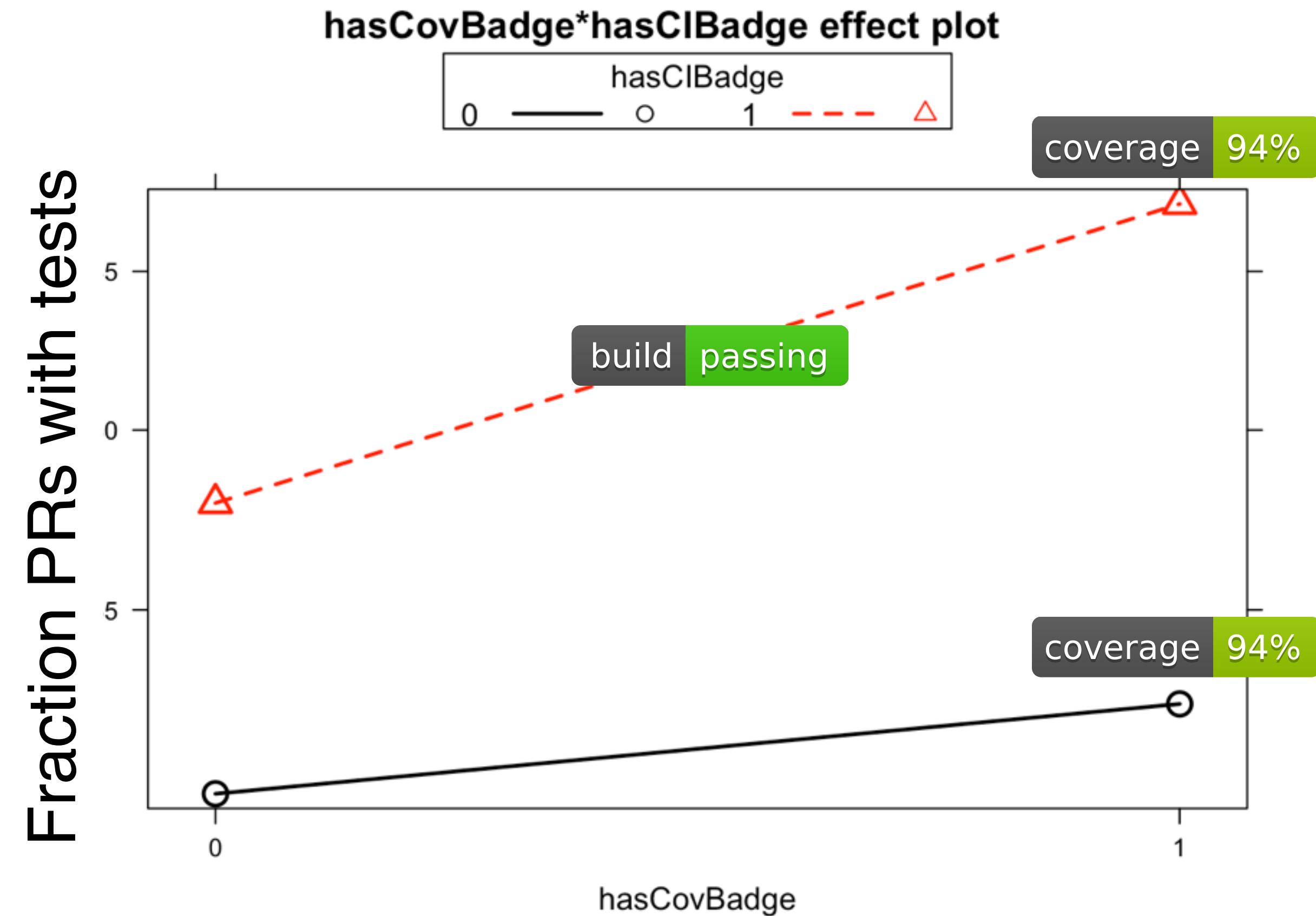
# Signals of PR quality



- Hyp: The adoption of a quality-assurance badge, and even more so of a coverage badge, encourages more external contributors to include tests.

*"PRs with new functionality tend to include new tests, as not to decrease coverage."*

- Coverage and CI badges interact, amplifying each other's effects.



# Gaming badges

seanmonstar / intel

Watch ▾

10

Star

191

Fork

29

Code

Issues 11

Pull requests 0

Projects 0

Wiki

Insights

I need more intel! <http://seanmonstar.github.io/intel/>

224 commits

3 branches

22 releases

11 contributors

MPL-2.0

Branch: master ▾

New pull request

Create new file

Upload files

Find file

Clone or download ▾

README.md

intel

build passing

coverage 100%

npm package 1.2.0

An abbreviation of intelligence. In this case, the acquirement of information.

# Gaming badges

seanmonstar / intel

Watch ▾ 10 Star 191 Fork 29

Code Issues 11 Pull requests 0 Projects 0 Wiki Insights

I need more intel! <http://seanmonstar.github.io/intel/>

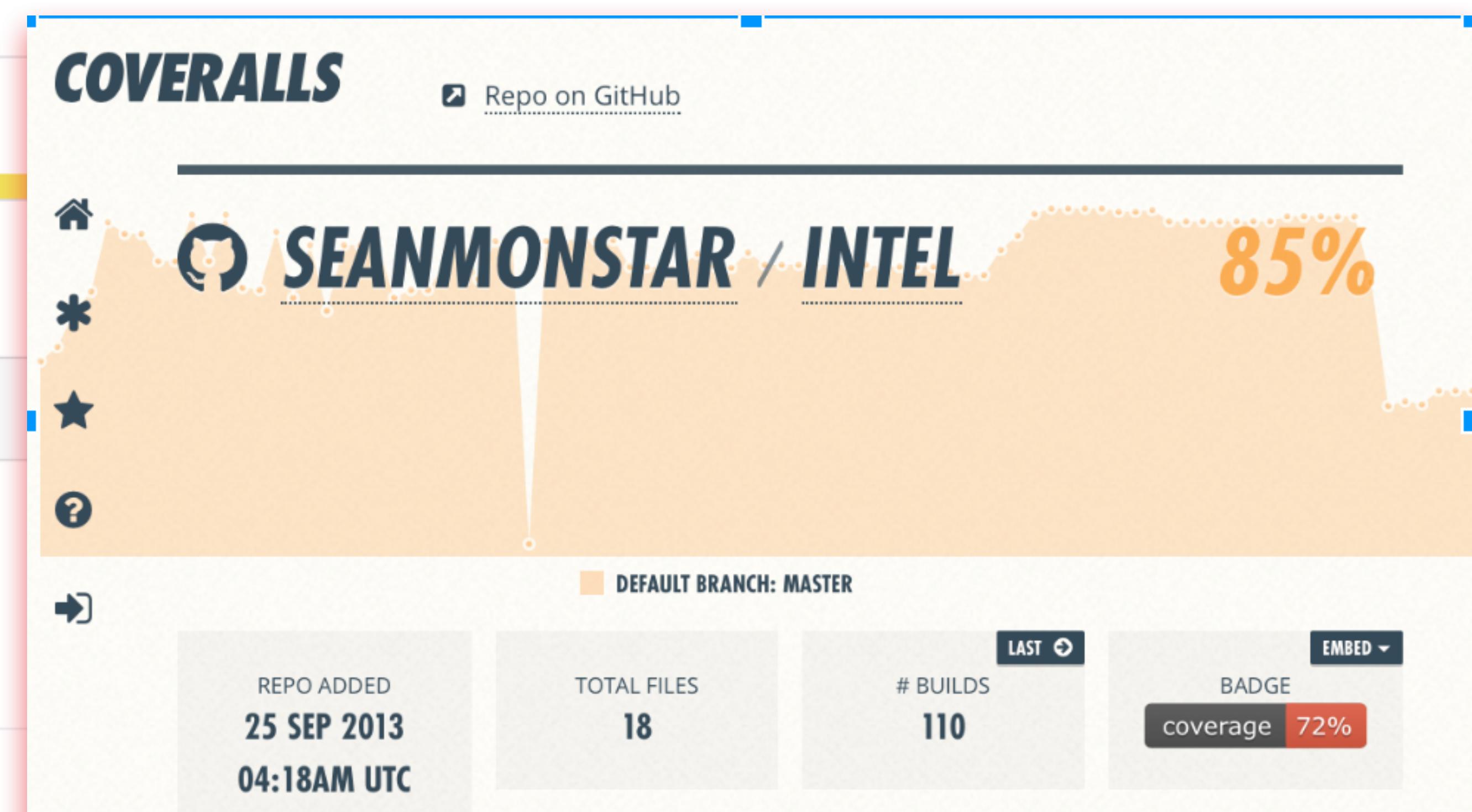
224 commits 3 branches

Branch: master New pull request

README.md

intel

build passing coverage 100% npm package 1.2.0



The screenshot shows a GitHub repository page for 'seanmonstar/intel'. A red box highlights the 'coverage' badge at the bottom left, which shows 100%. To the right, a large Coveralls badge indicates 85% coverage. The Coveralls interface includes a timeline chart, repository information (Repo on GitHub), and build statistics (Repo added: 25 SEP 2013, Total files: 18, # Builds: 110, Last build: 04:18AM UTC). There are also links for Embed and Badge.

An abbreviation of intelligence. In this case, the acquirement of information.

# Mixed methods study



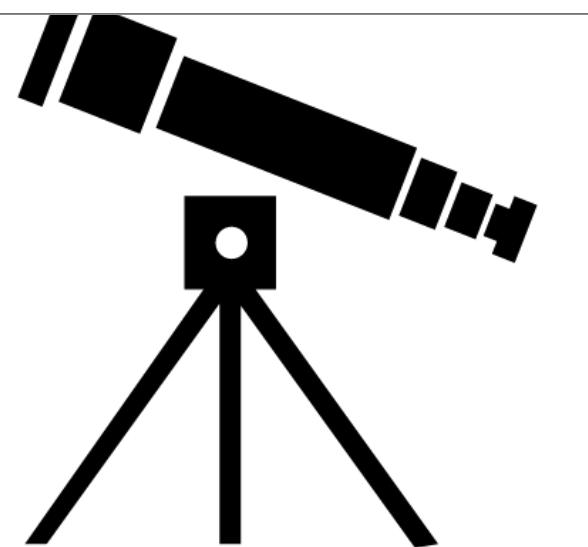
+



- 32 maintainers, 57 contributors (15% resp. rate)
- Maintainers:
  - What do you intend to signal?
  - What effects do you expect?
- Contributors:
  - What do badges tell you?
- 294,941 *npm* packages
- Mined badge adoptions/removals from README files
- Measured proxies for code quality, test suite quality, popularity, dependency freshness, ...

# Recap

## Analysis



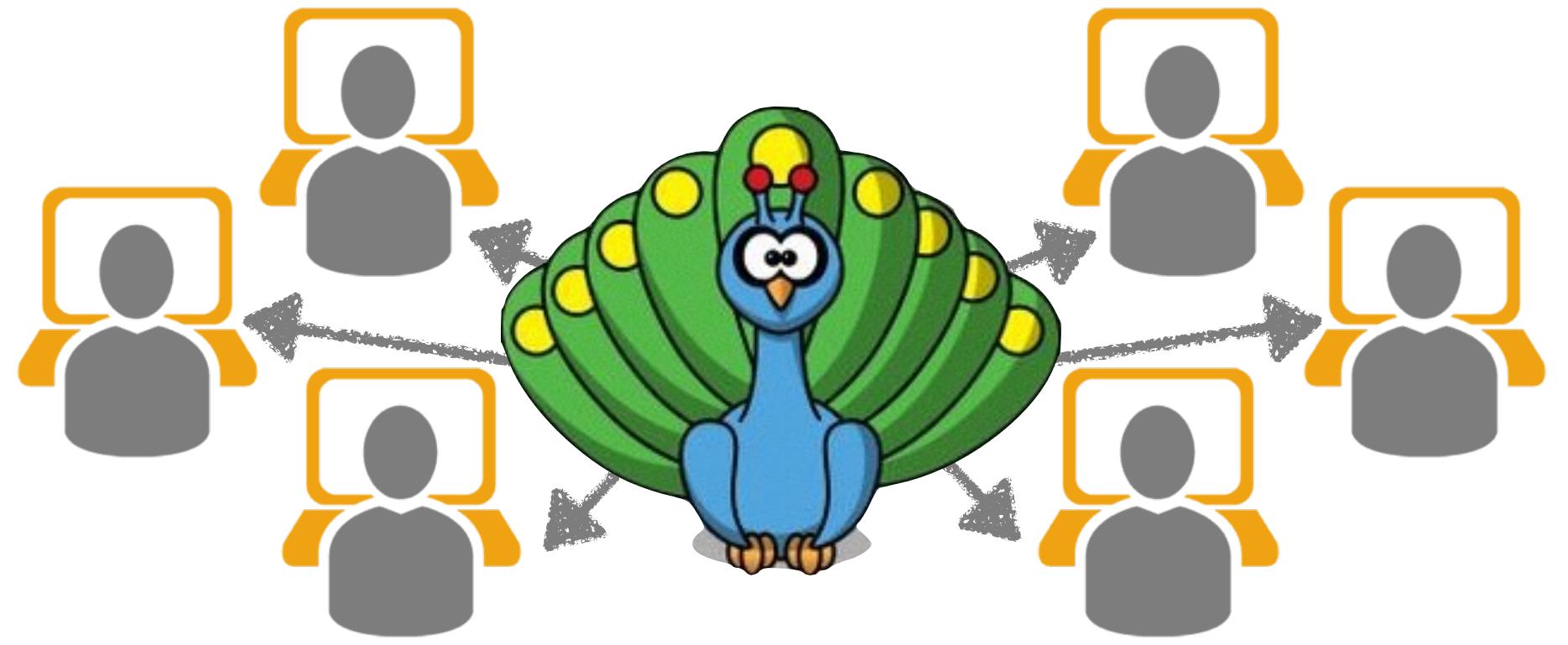
If all you saw was the badge, how much would that tell you?

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

How do things change after adding the badge?

# Take-aways (1)

- Many open source developers rely on, and respond to, signals
  - We add both qualitative and quantitative evidence for badges



# Take-aways (2)

- Harder to fake badges provide more reliable signals
  - As signaling theory predicts

build passing

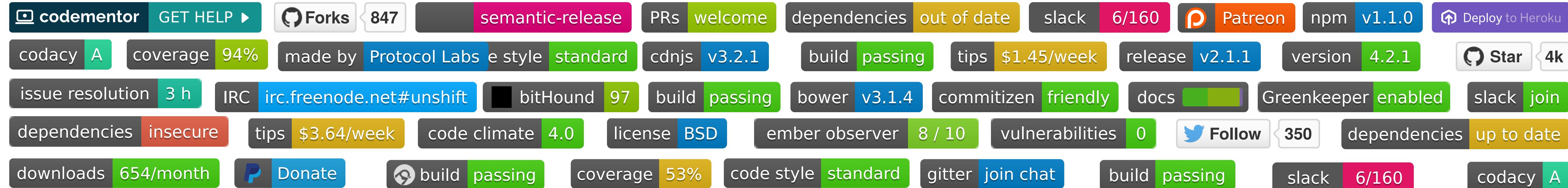
downloads 654/month

dependencies up to date

VS

npm v1.1.0

slack join



# Take-aways (2)

- Harder to fake badges provide more reliable signals
  - As signaling theory predicts
  - Redesign badges as assessment signals

code style standard

gitter join chat

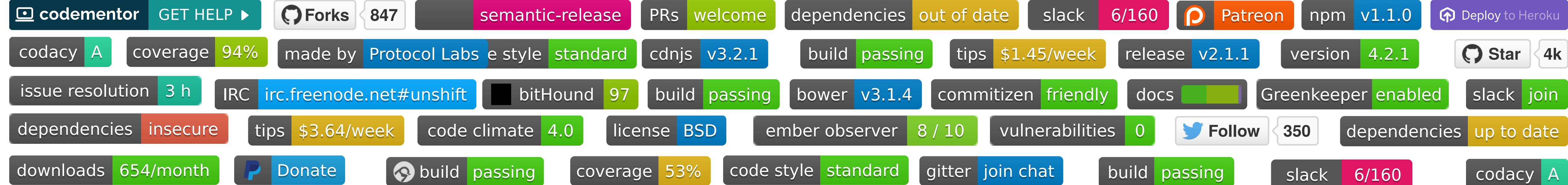
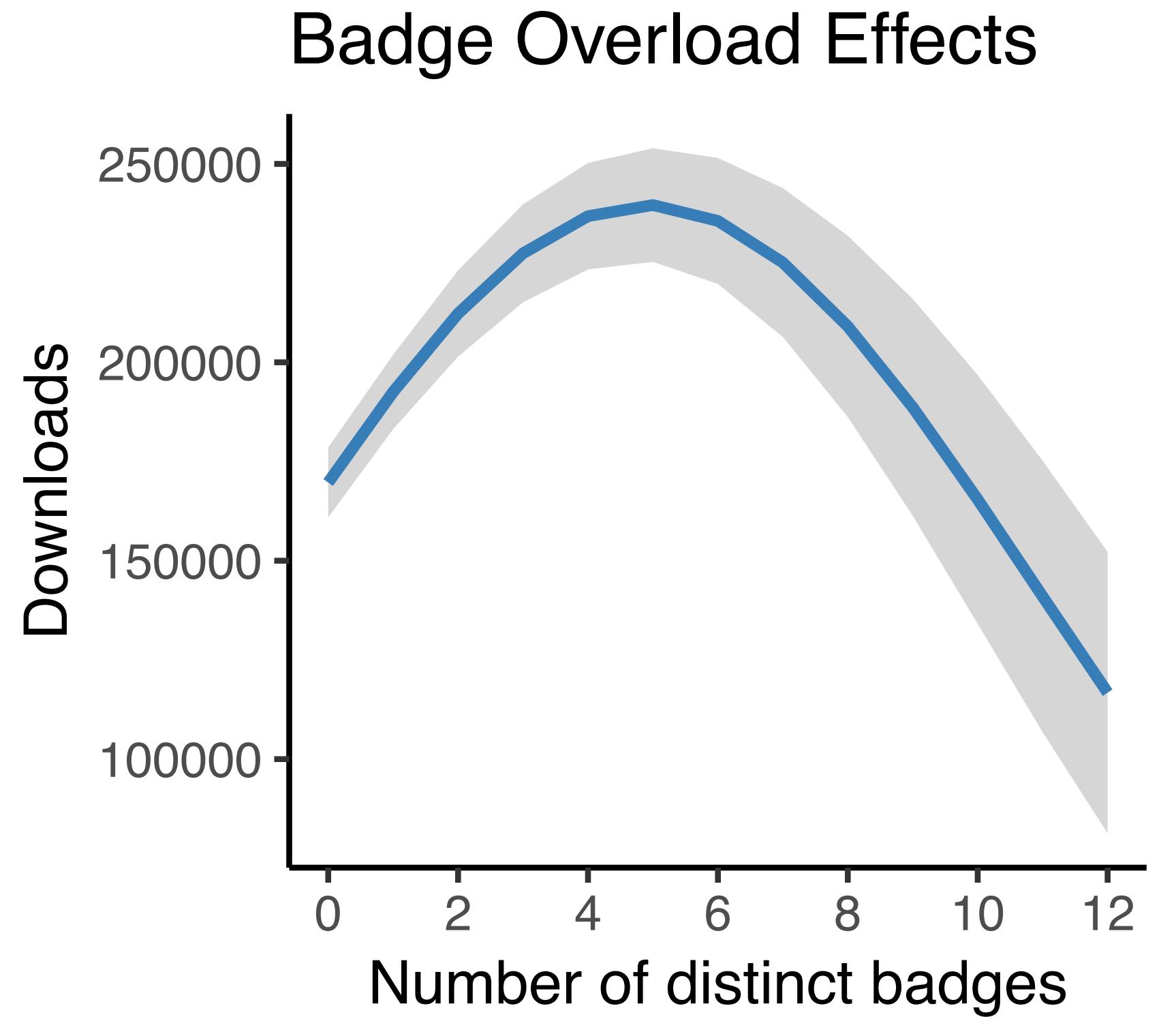
slack join

slack 6/160



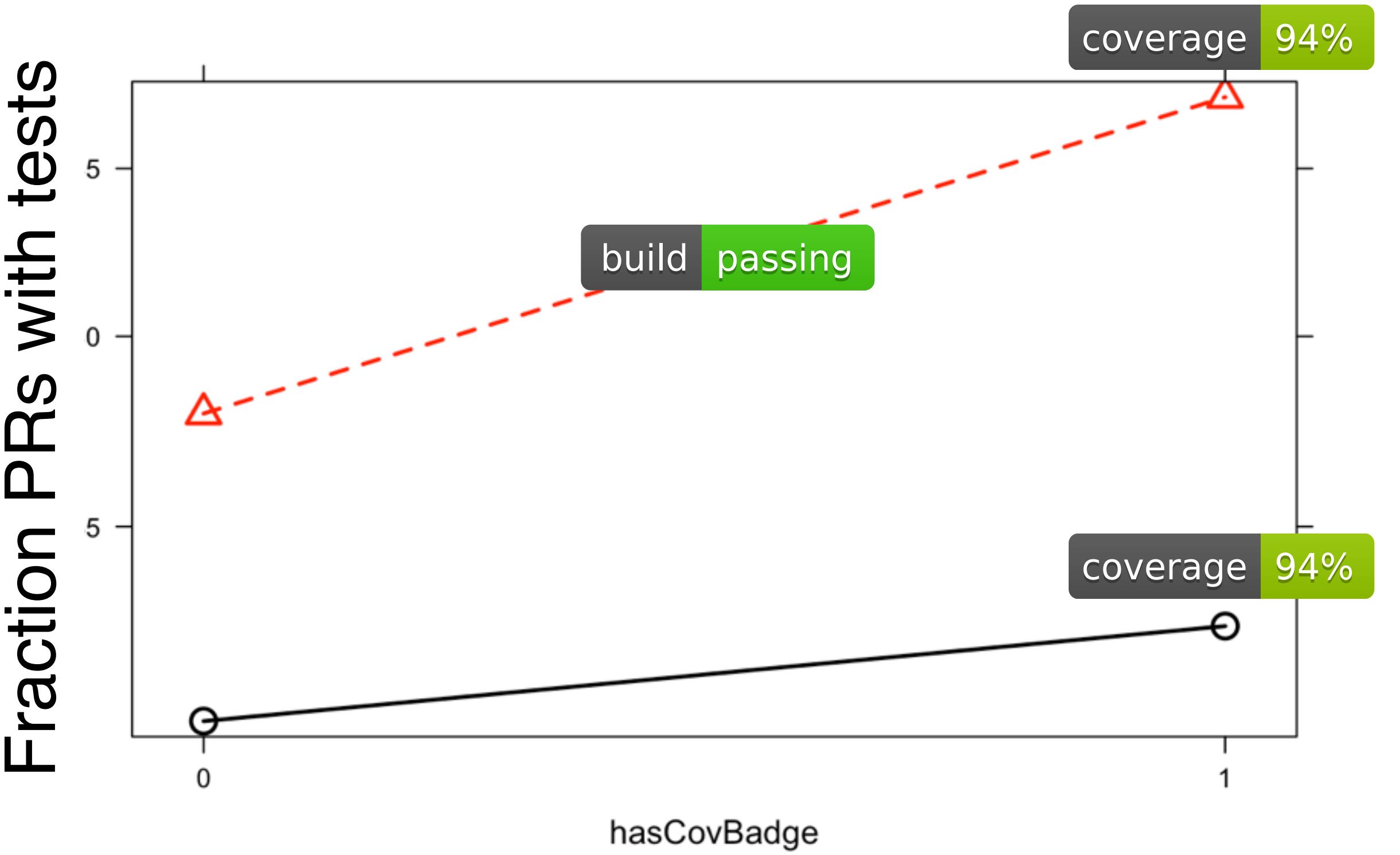
# Take-aways (3)

- Too much of a good thing



# Take-aways (4)

- Gamification effects



# Adding Sparkle to Social Coding

