

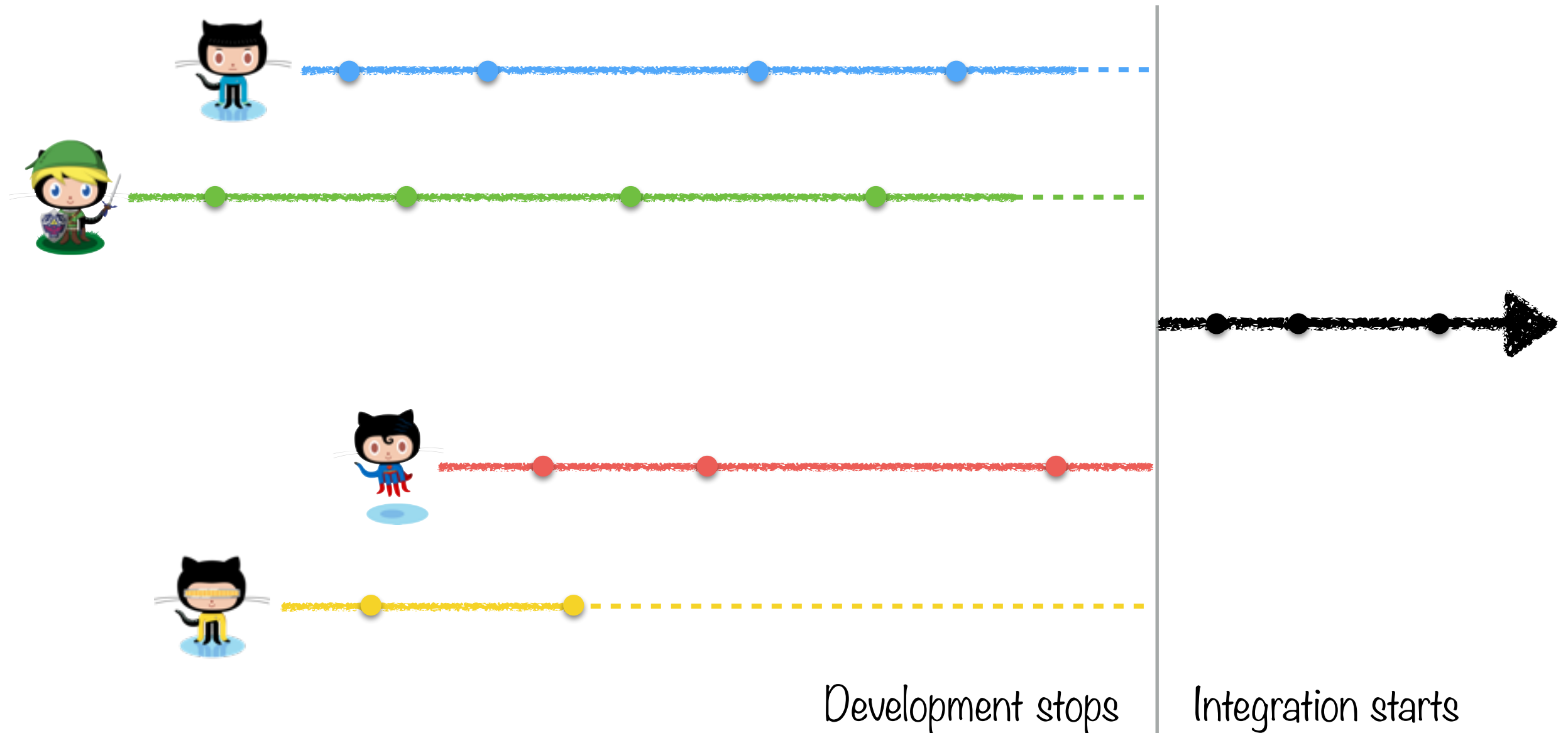


Continuous integration in a social-coding world

Bogdan Vasilescu, Stef van Schuylenburg, Jules Wulms,
Alexander Serebrenik, Mark G. J. van den Brand

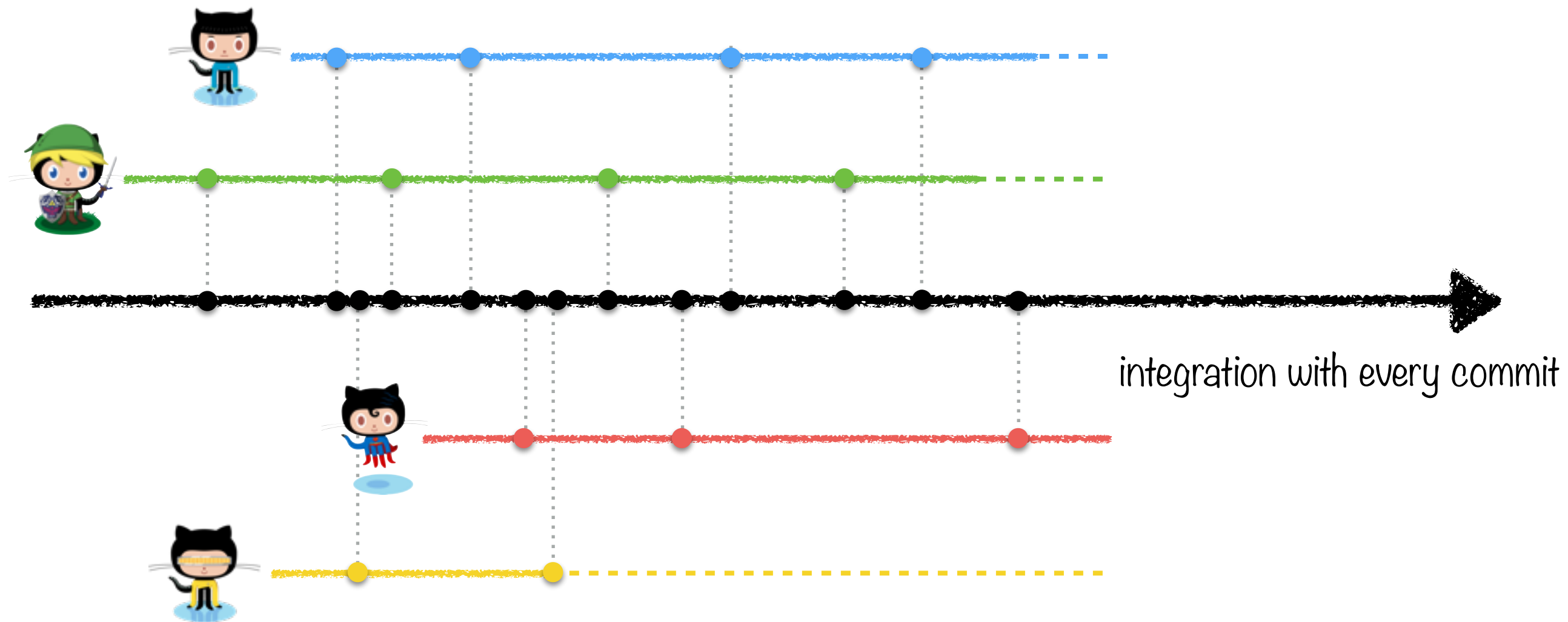
Integration

“[used to be] a long and unpredictable process” (Martin Fowler, 2000)



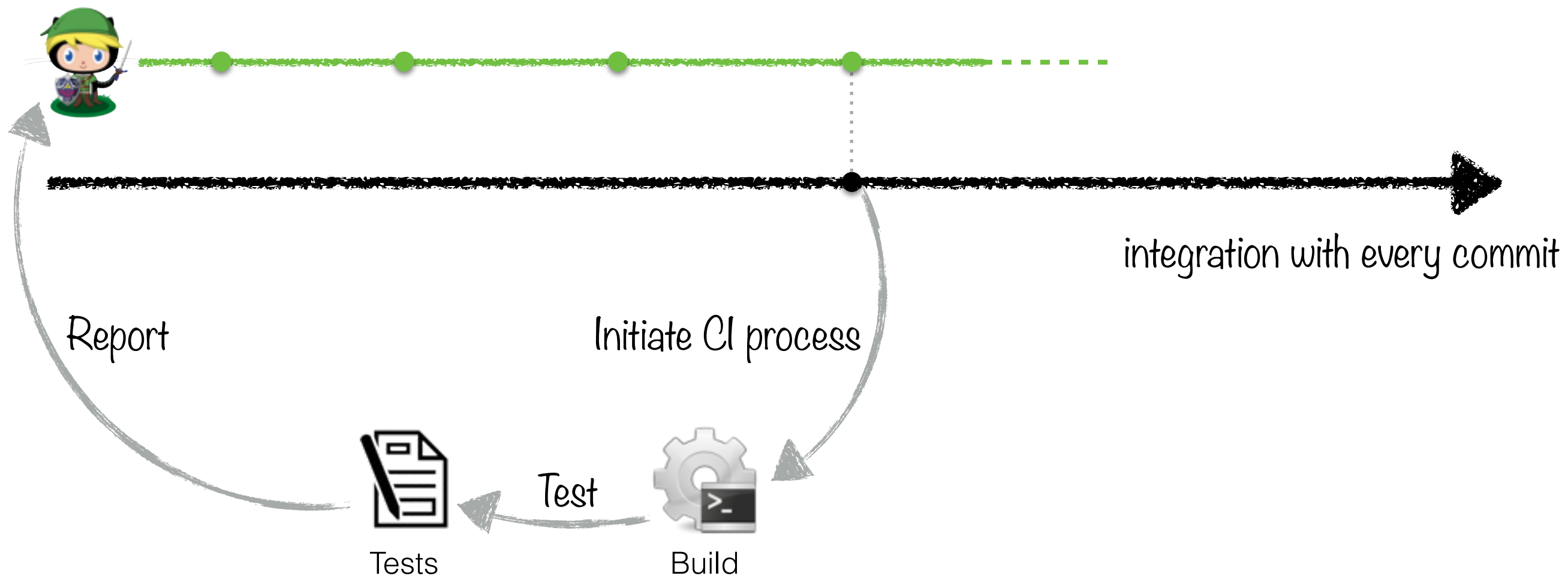
Continuous Integration

speeds up collaborative software development
by reducing integration problems



Continuous Integration

speeds up collaborative software development
by reducing integration problems





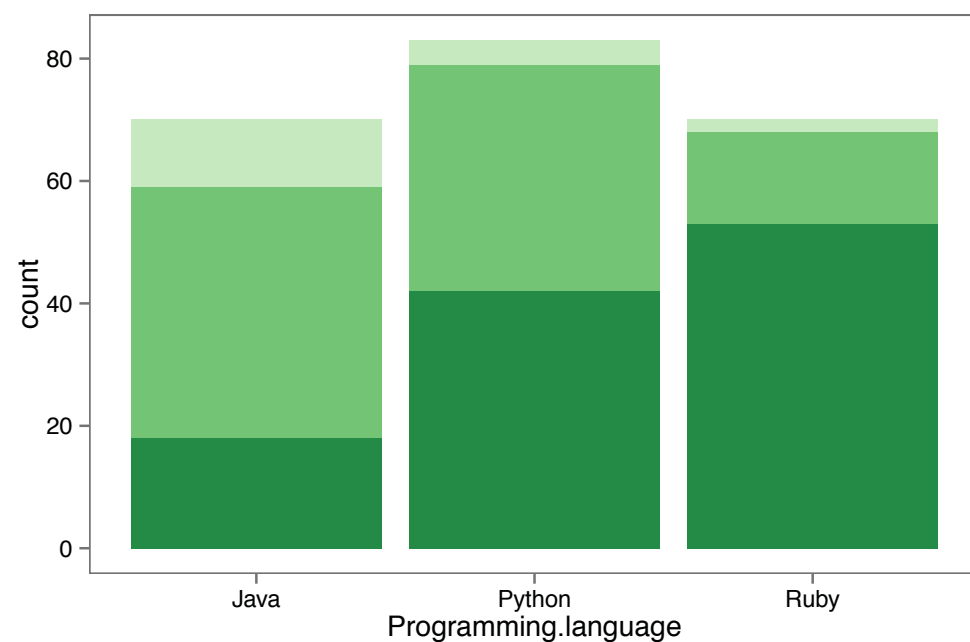
“There are **7.1M** people collaborating right now across **15.9M** repositories on GitHub. Developers from all around the world are building amazing things together. Their story is our story.”

<https://github.com/about/press>

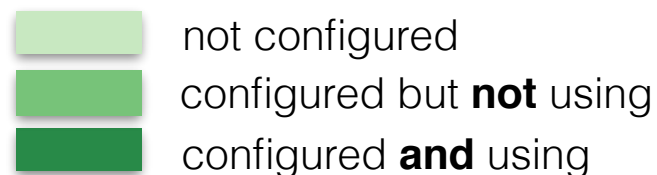
How are GitHub developers using CI?



How are GitHub developers using CI?



(1) Most projects are configured to use CI, but less than half actually do.



GitHub workflow

Contributing to IronCat's repository

Octocat



Jean-Luc
Picat



Spocktocat



IronCat

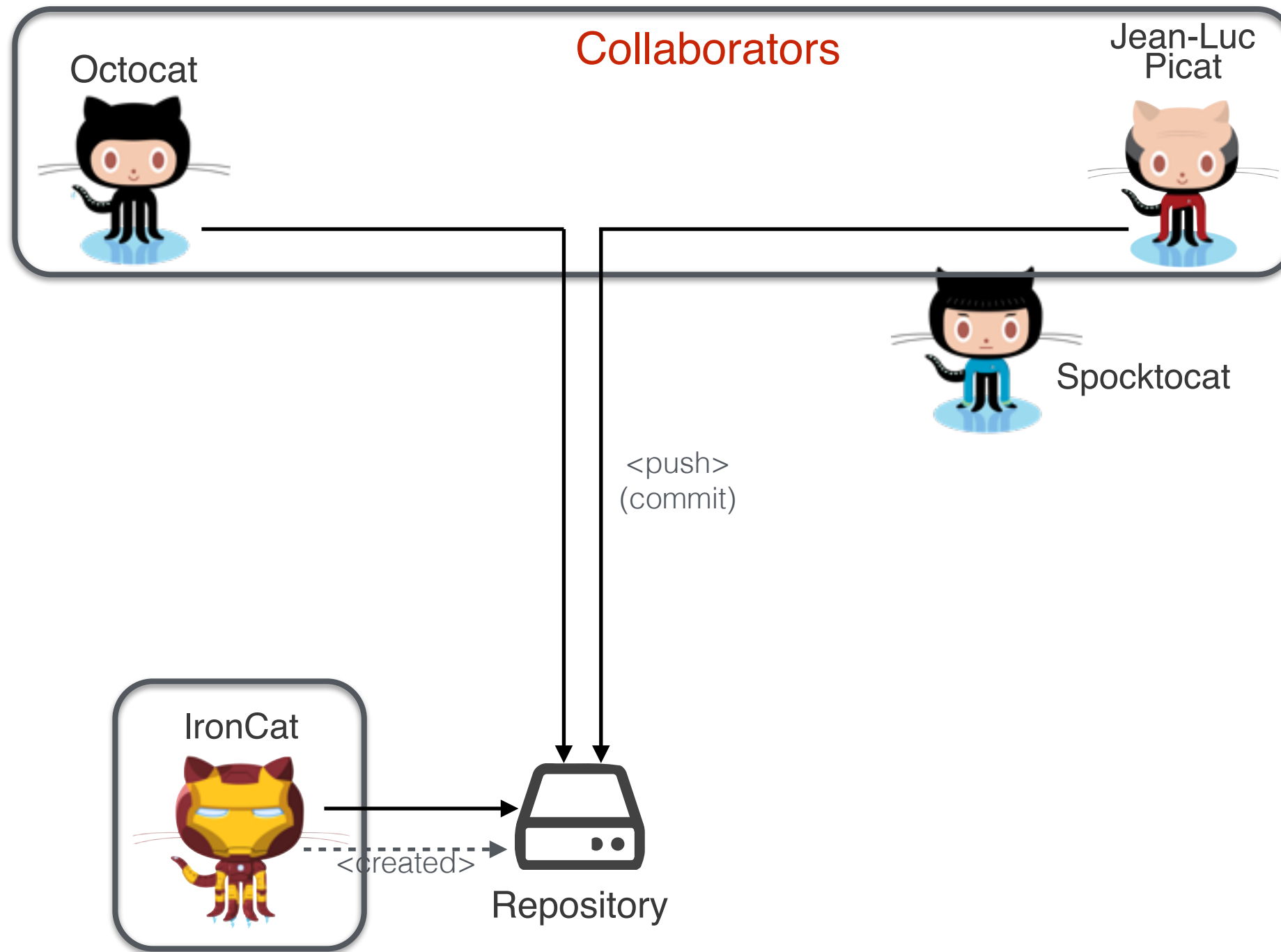


<created>

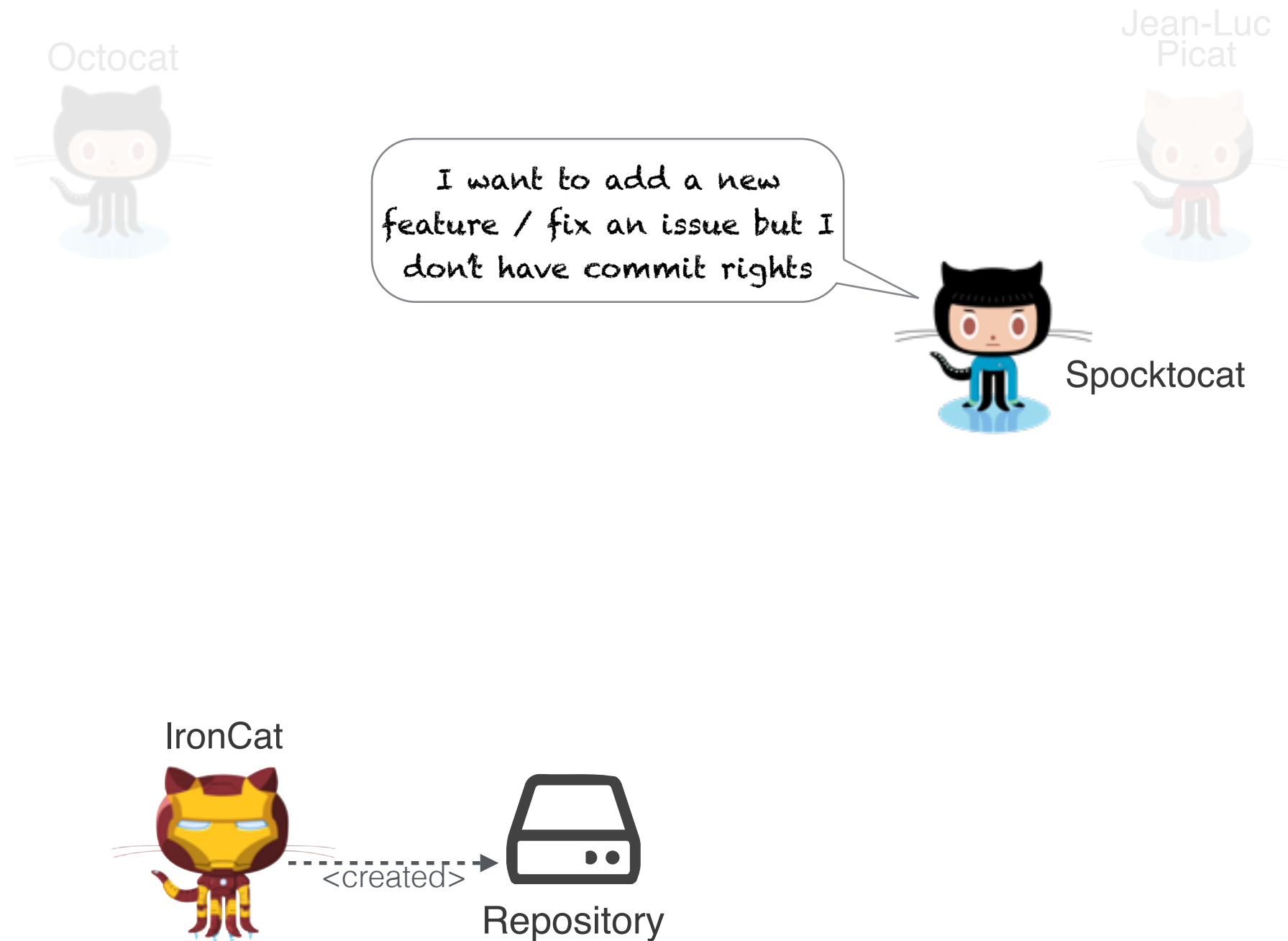


Repository

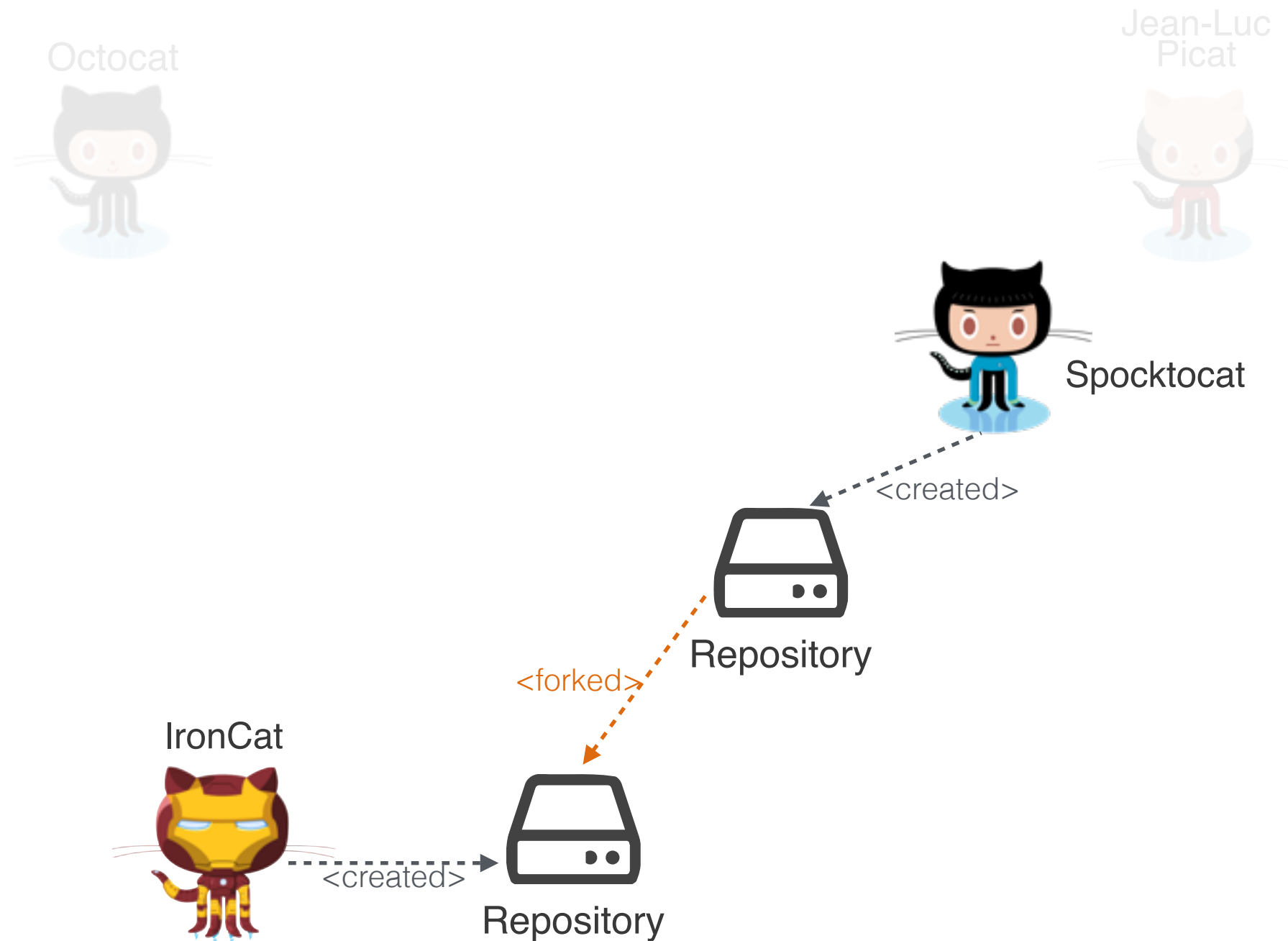
GitHub workflow



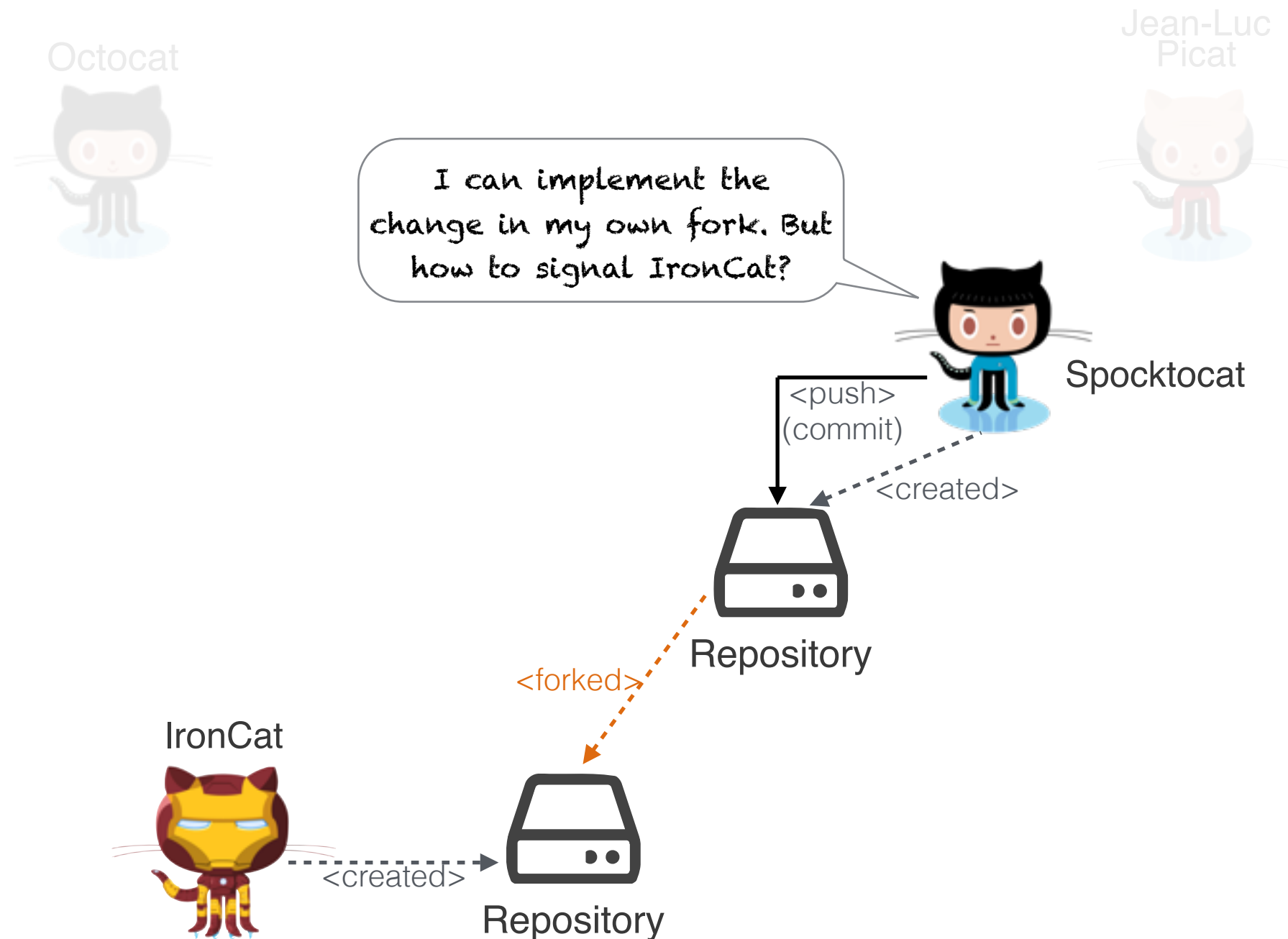
GitHub workflow



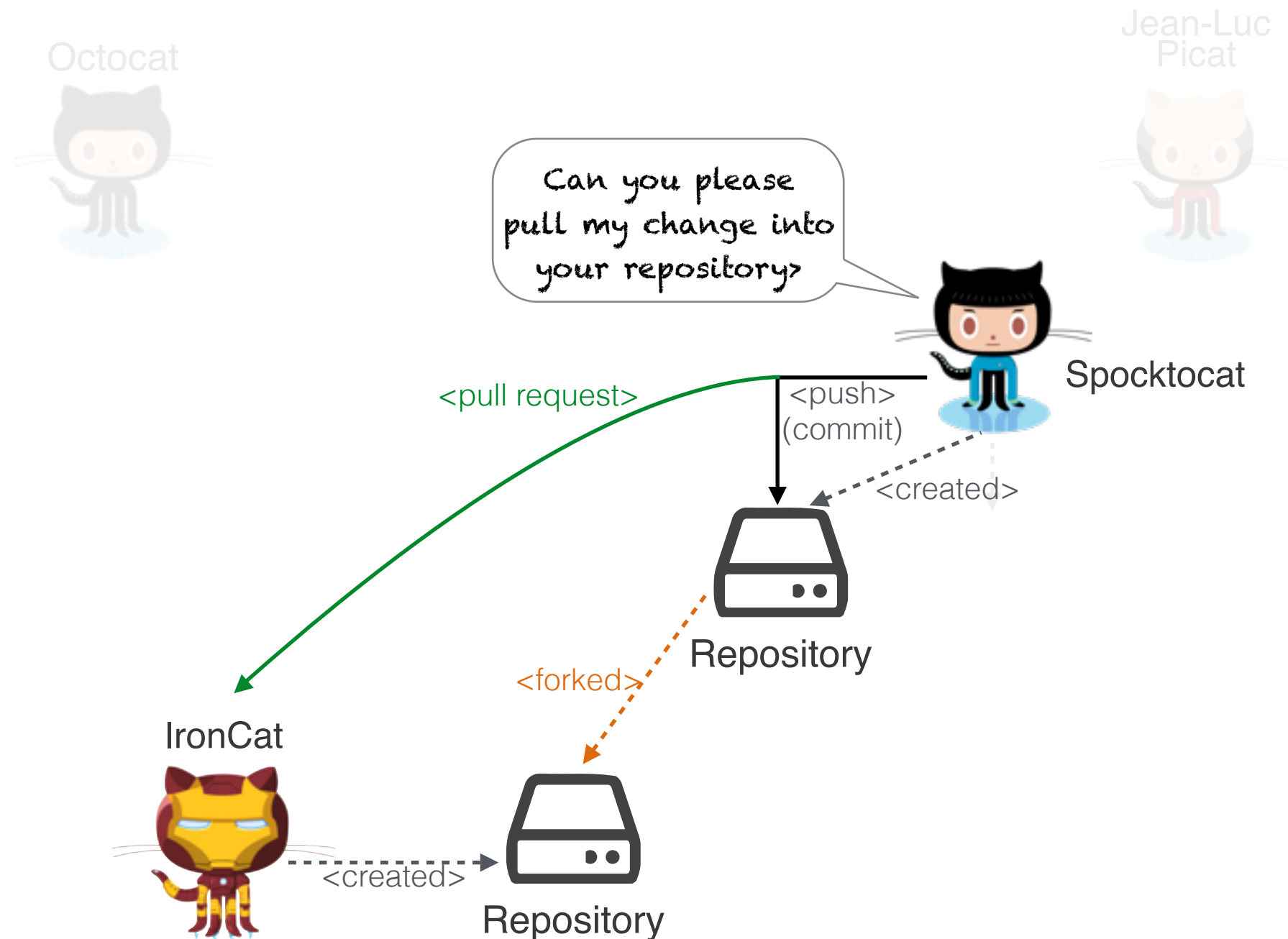
GitHub workflow



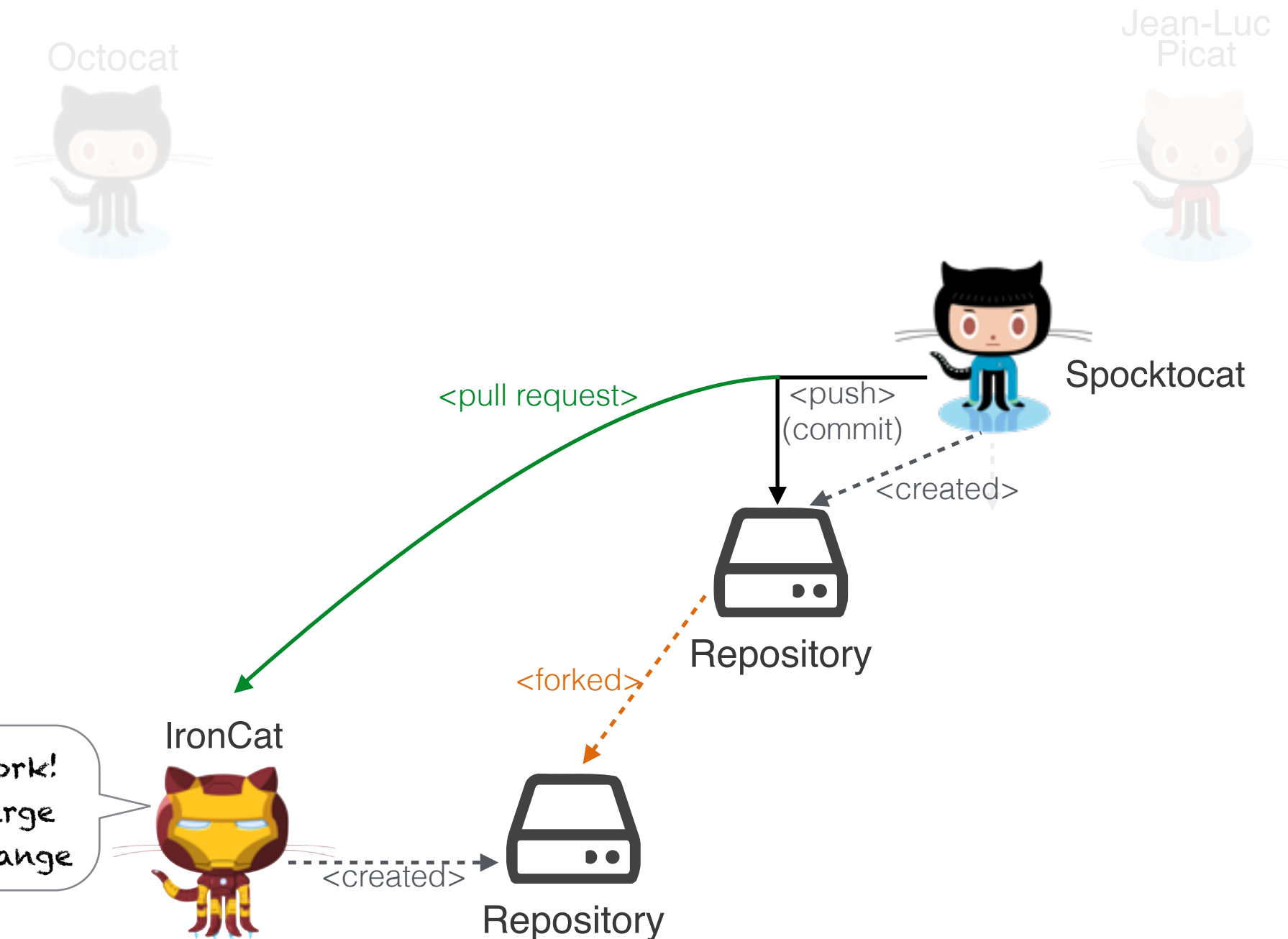
GitHub workflow



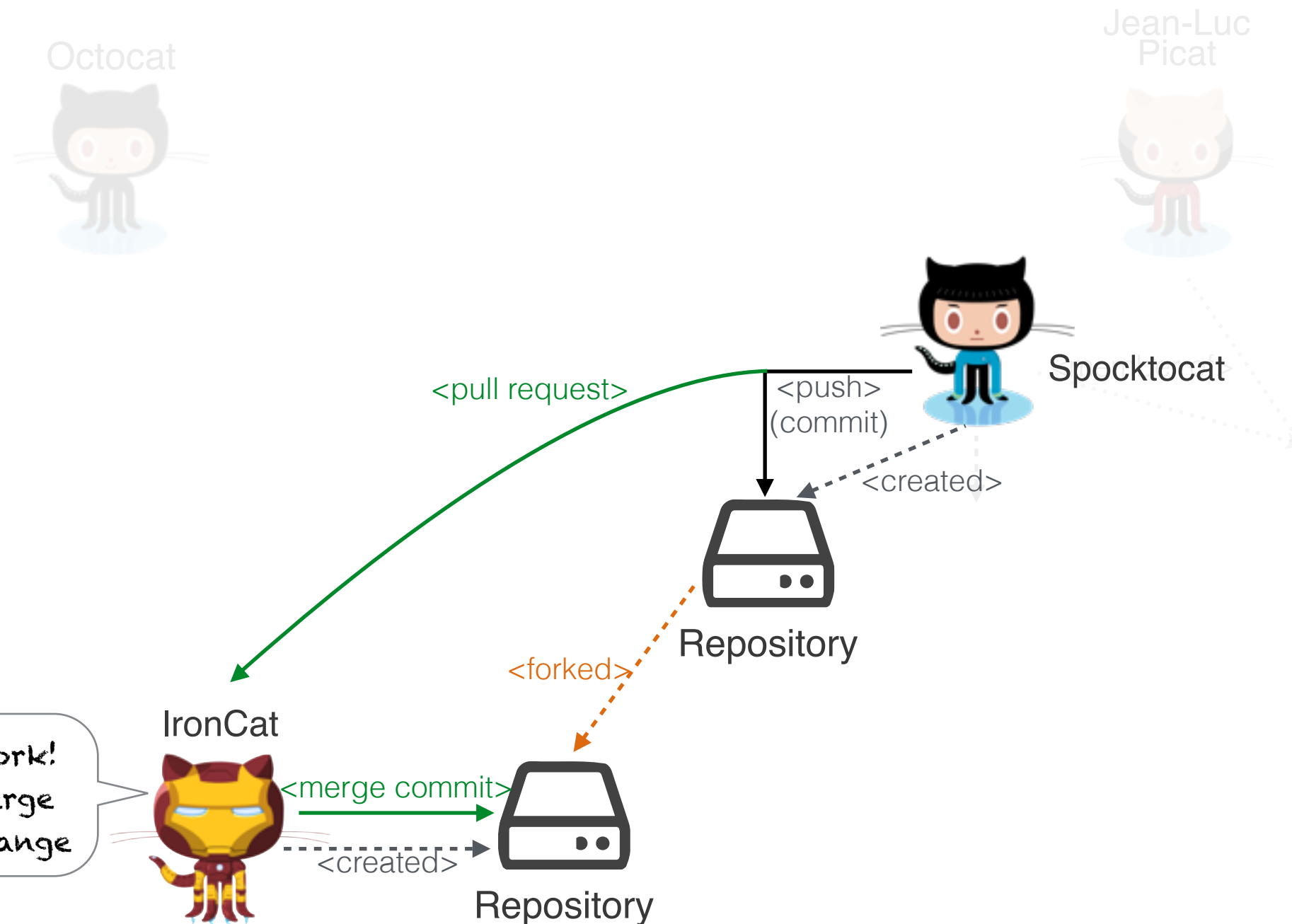
GitHub workflow



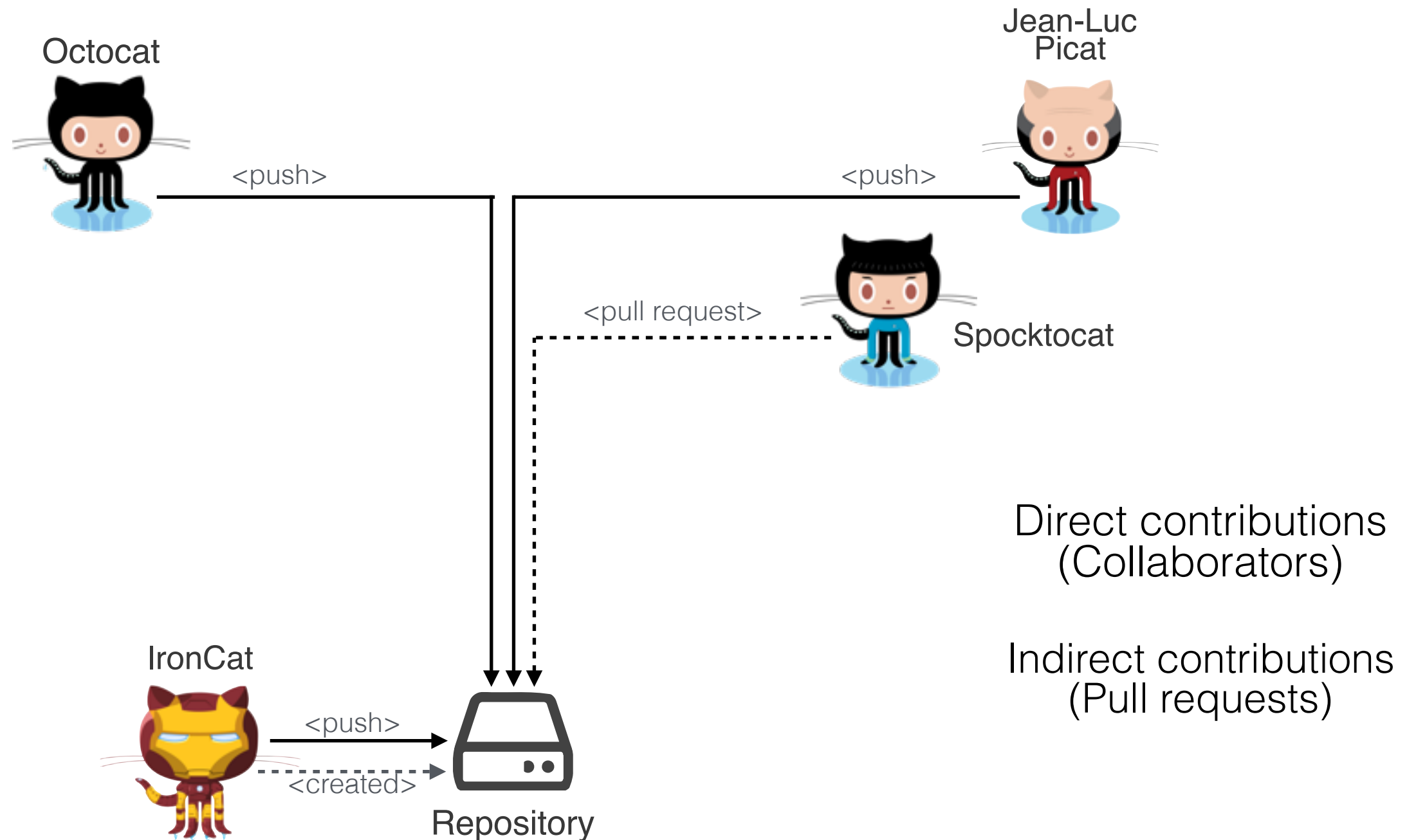
GitHub workflow



GitHub workflow



GitHub workflow





unprecedented low barrier to entry for
potential contributors

L. A. Dabbish, H. C. Stuart, J. Tsay, and J. D. Herbsleb, “**Social coding in GitHub: transparency and collaboration in an open software repository**,” in *CSCW 2012*

scalability challenges when integrating
(many) outside contributions

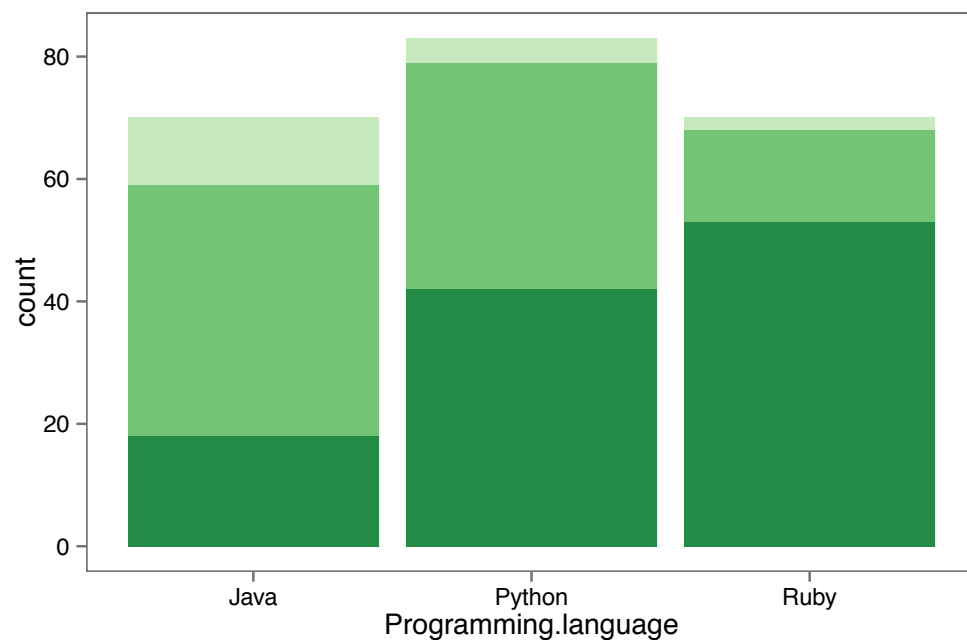


automated tests

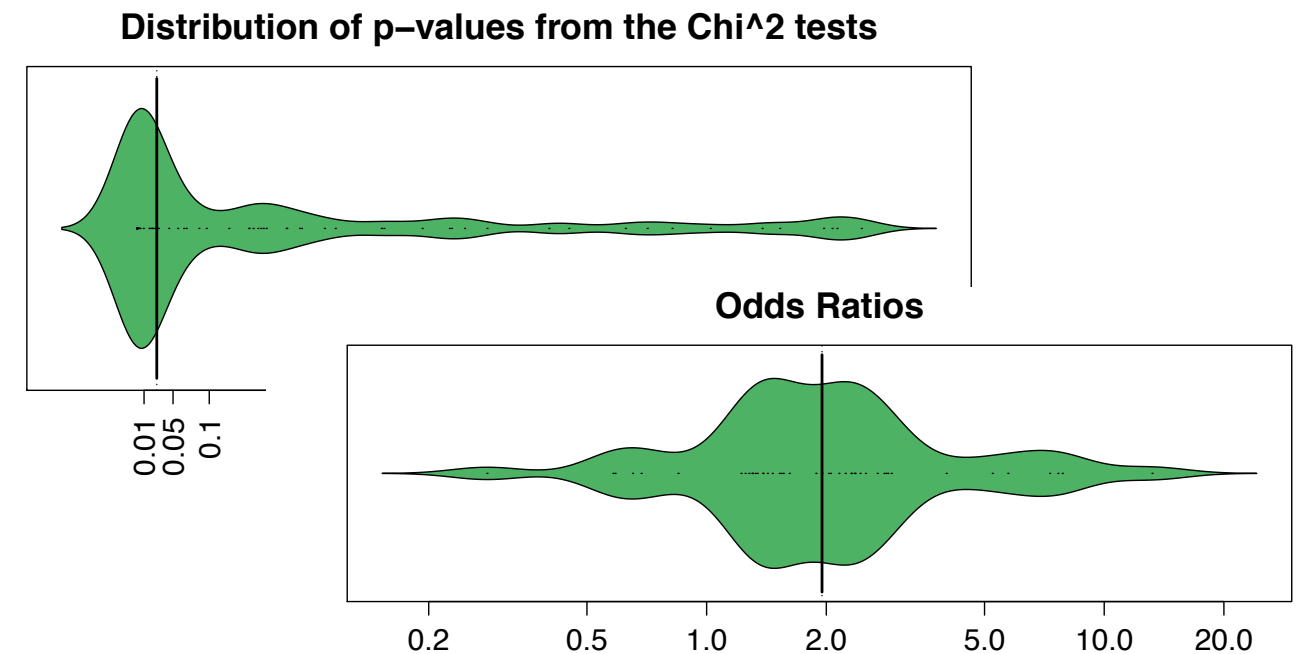
R. Pham, L. Singer, O. Liskin, K. Schneider et al., “**Creating a shared understanding of testing culture on a social coding site**,” in *ICSE 2013*



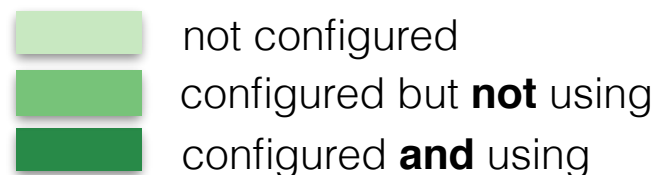
How are GitHub developers using CI?



(1) Most projects are configured to use CI, but less than half actually do.



(2) Pull requests are much more likely to result in successful builds than direct commits



GHTorrent sample

G. Gousios, “**The GHTorrent dataset and tool suite**,” in *MSR 2013*

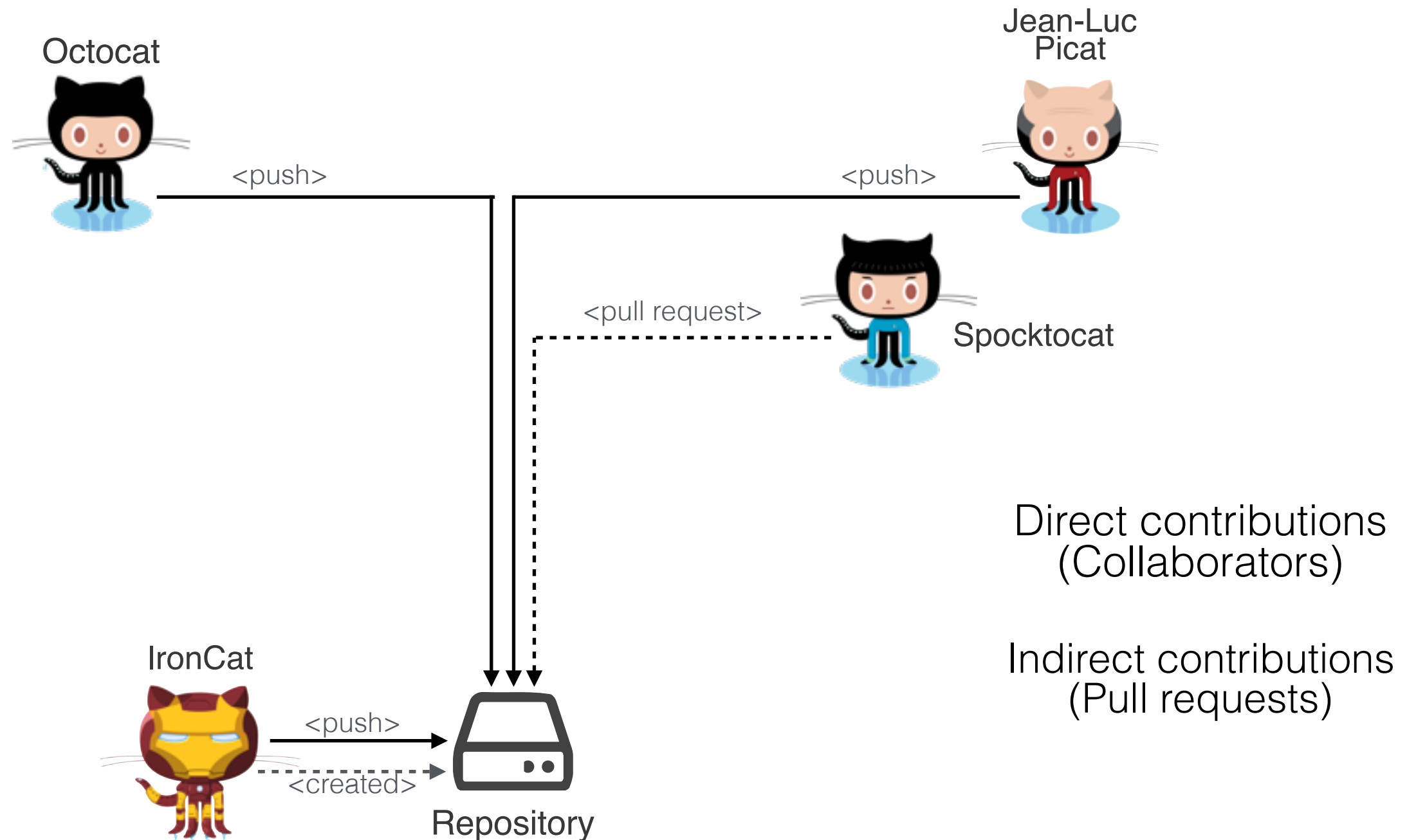
G. Gousios, B. Vasilescu, A. Serebrenik, A. Zaidman, “**Lean GHTorrent: GitHub data on demand**,” in *MSR 2014*

223 large and active repositories

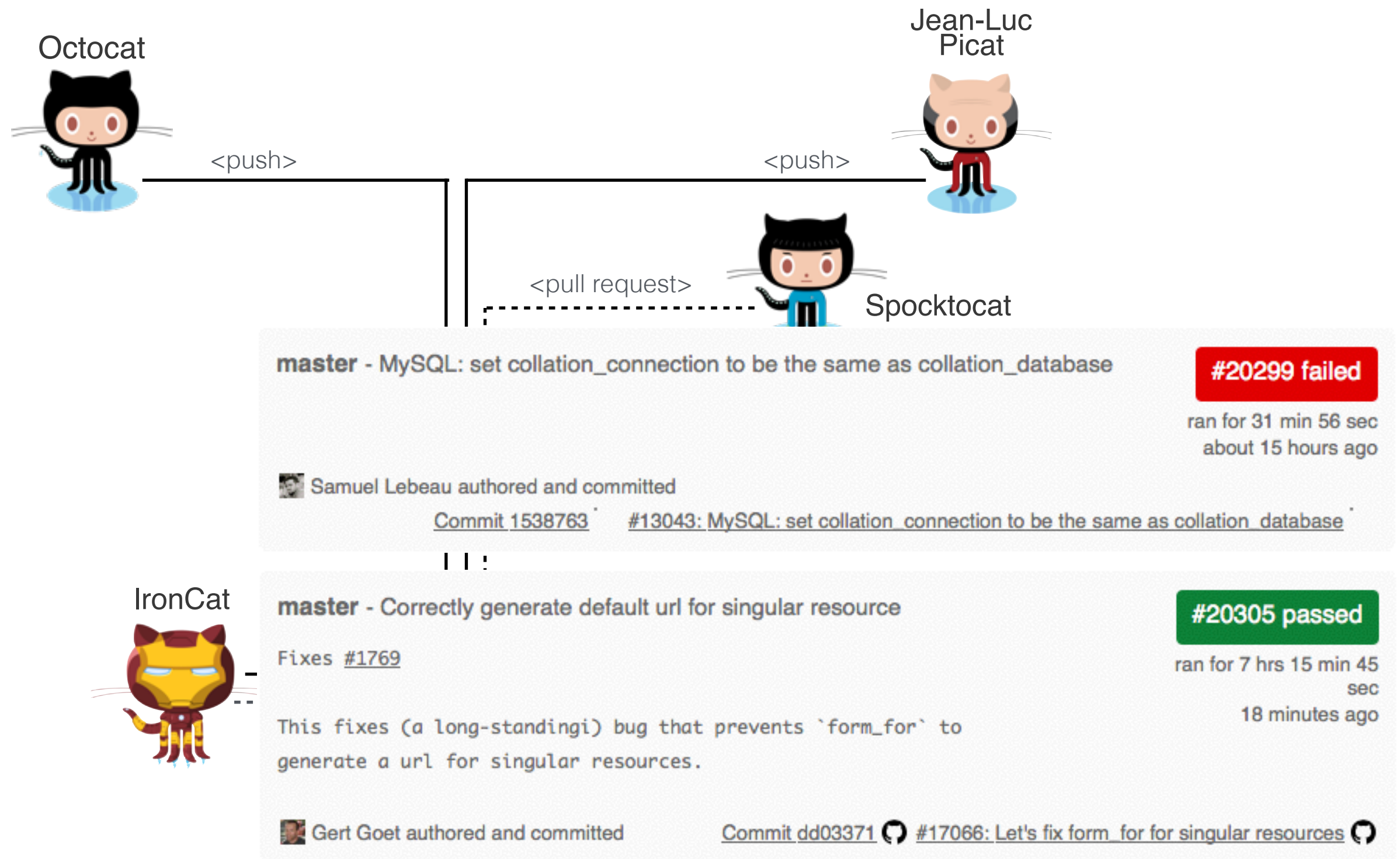
- * not forks
- * ≥ 1 year
- * both commits and pull requests
- * Java, Python, or Ruby
- * ≥ 10 changes last month
- * ≥ 10 contributors



GitHub + Travis-CI



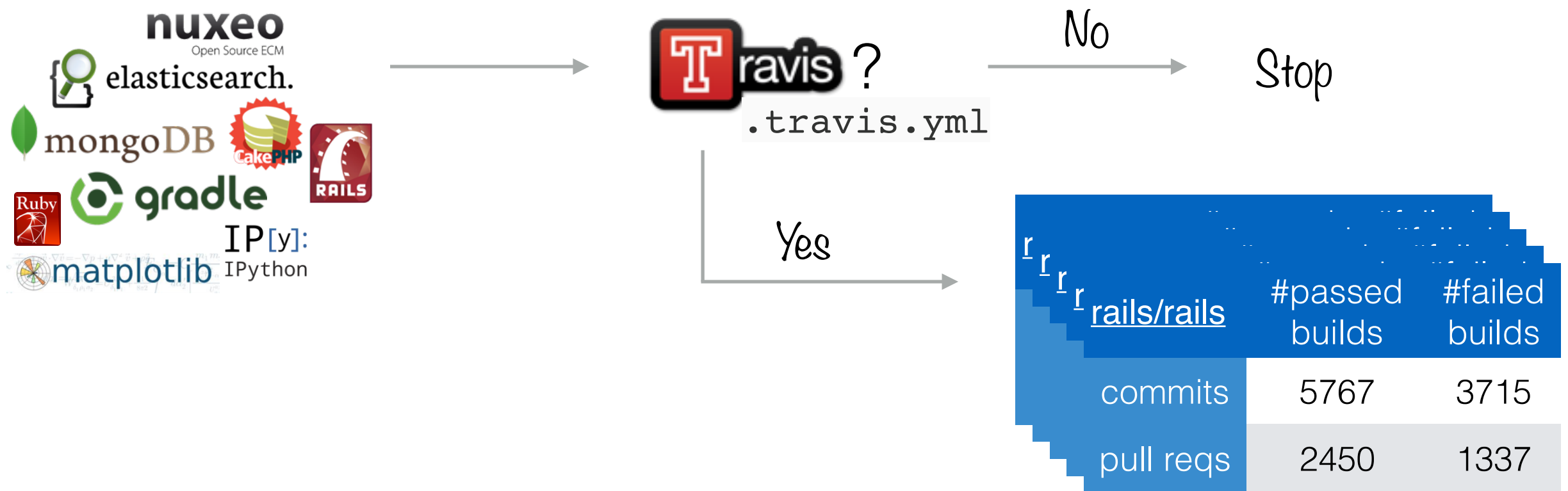
GitHub + Travis-CI



Methods and Data



Methods and Data



Methods and Data

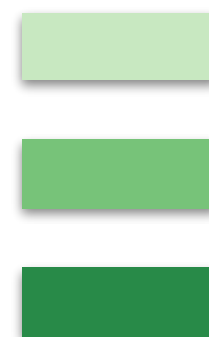
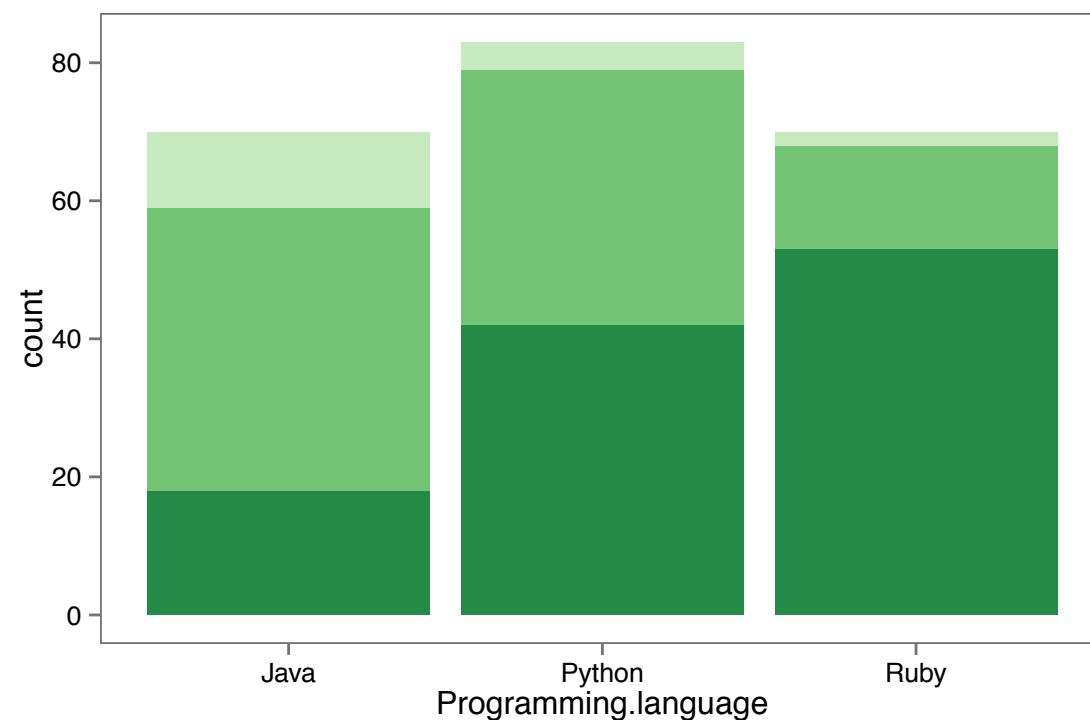


No

Stop

Yes

rails/rails		
	#passed builds	#failed builds
commits	5767	3715
pull reqs	2450	1337



not configured (17 / 223)

configured but **not** using (93 / 223)

configured **and** using (113 / 223)

Methods and Data



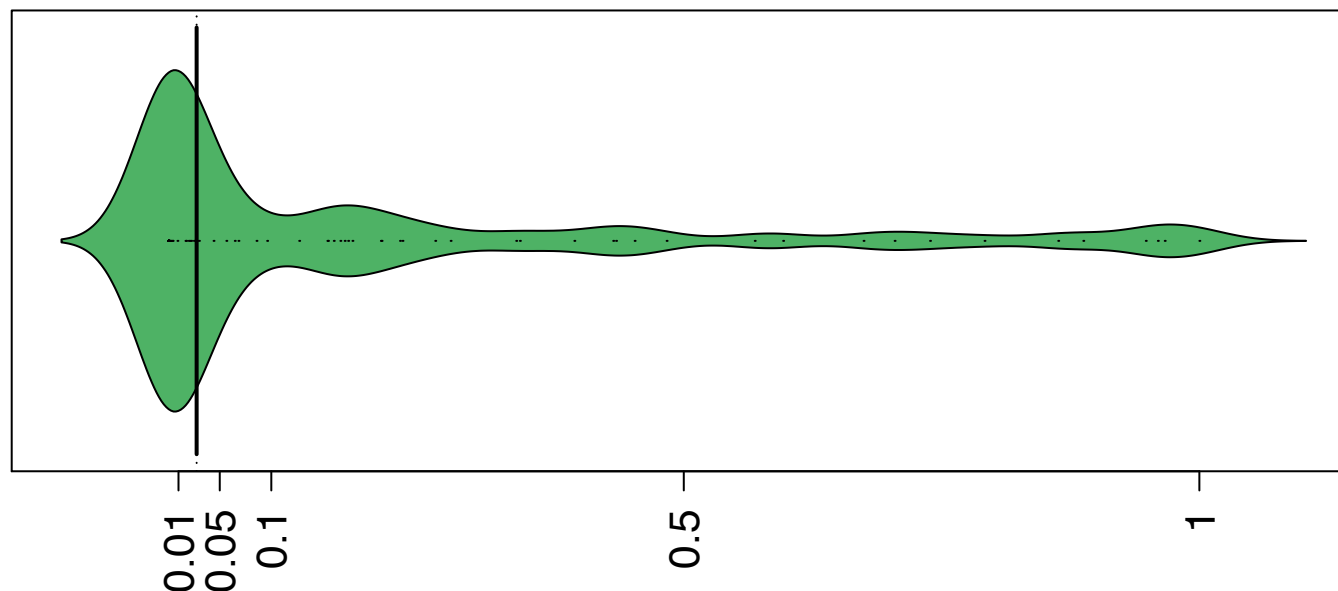
No

Stop

Yes

	#passed builds	#failed builds
rails/rails		
commits	5767	3715
pull reqs	2450	1337

Distribution of p-values from the Chi² tests



Does type matter for passed/failed?

χ^2 + Stouffer

Methods and Data



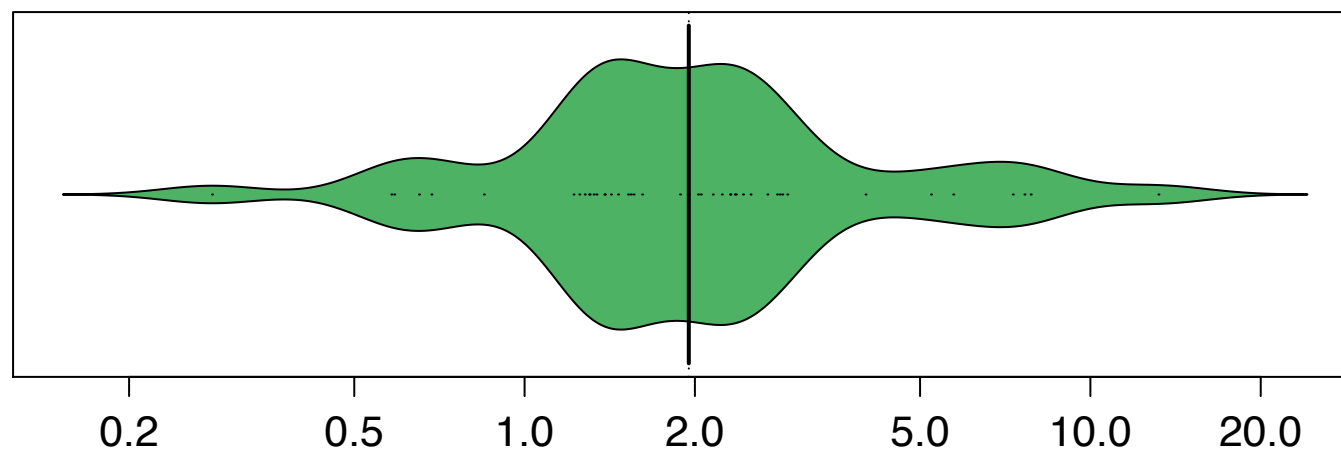
No

Stop

Yes

rails/rails		
	#passed builds	#failed builds
commits	5767	3715
pull reqs	2450	1337

Odds Ratios

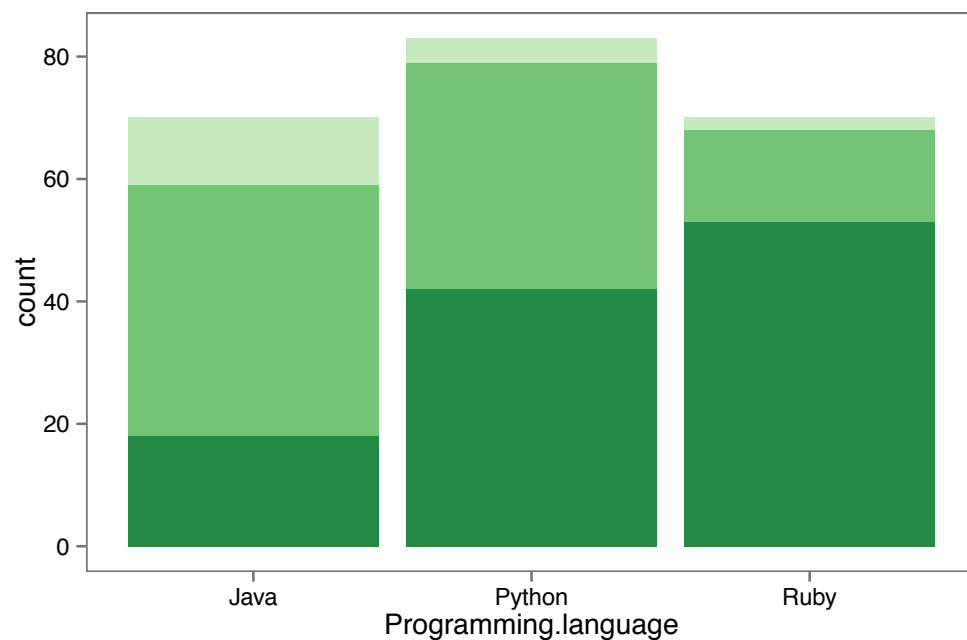


Which type
succeeds more
often?

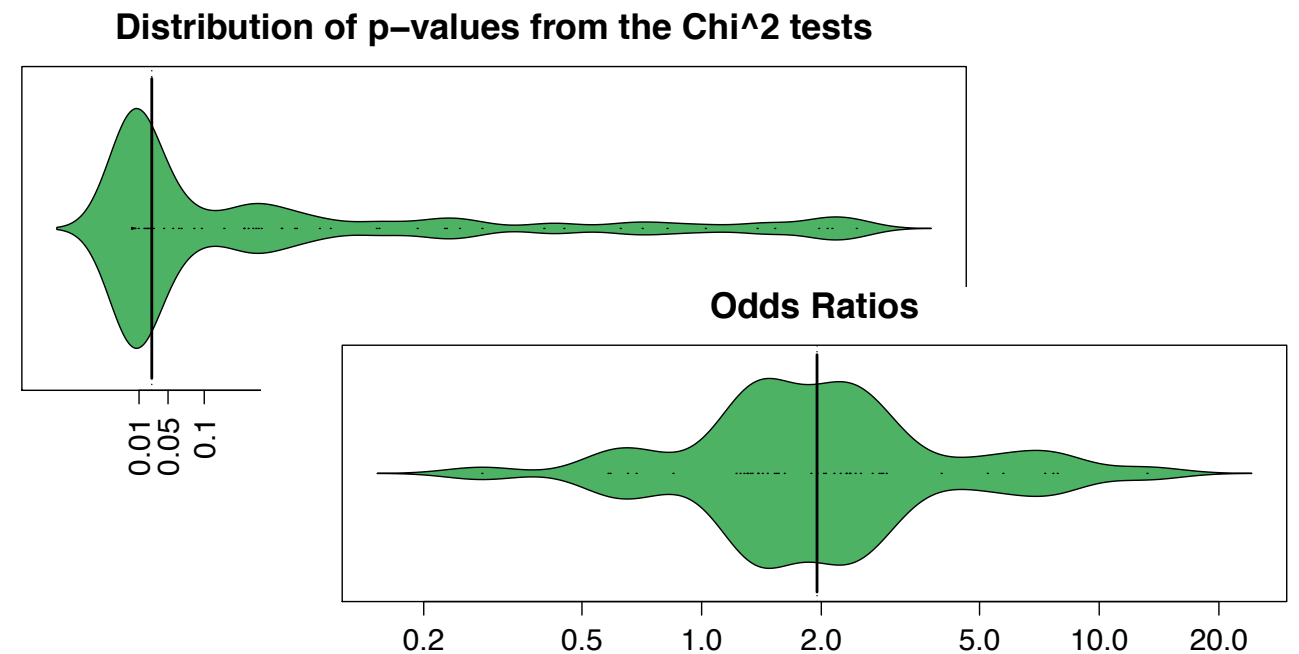
Odds ratios



How are GitHub developers using CI?



(1) Most projects are configured to use CI, but less than half actually do.



(2) Pull requests are much more likely to result in successful builds than direct commits