DALL·E 3 - “Networks of open-source software projects”

The Strength of Weak Ties in Open-Source Software Development Networks

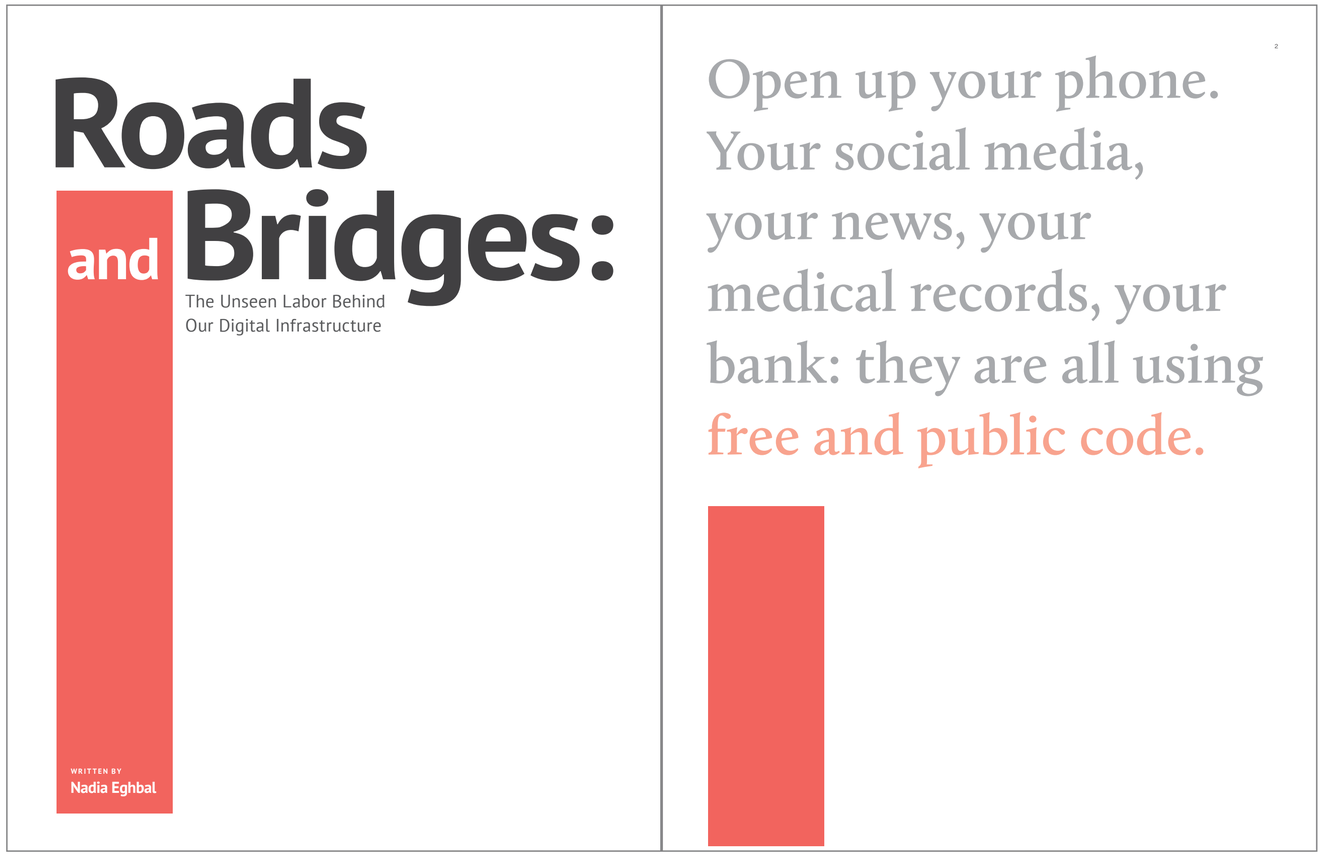
Bogdan Vasilescu
At U Zurich, January 9th, 2025

STRUDEL
SOCIO-TECHNICAL RESEARCH USING DATA EXCAVATION LAB

Carnegie Mellon University

S3D
Software and Societal Systems Department

Open source software has become digital infrastructure

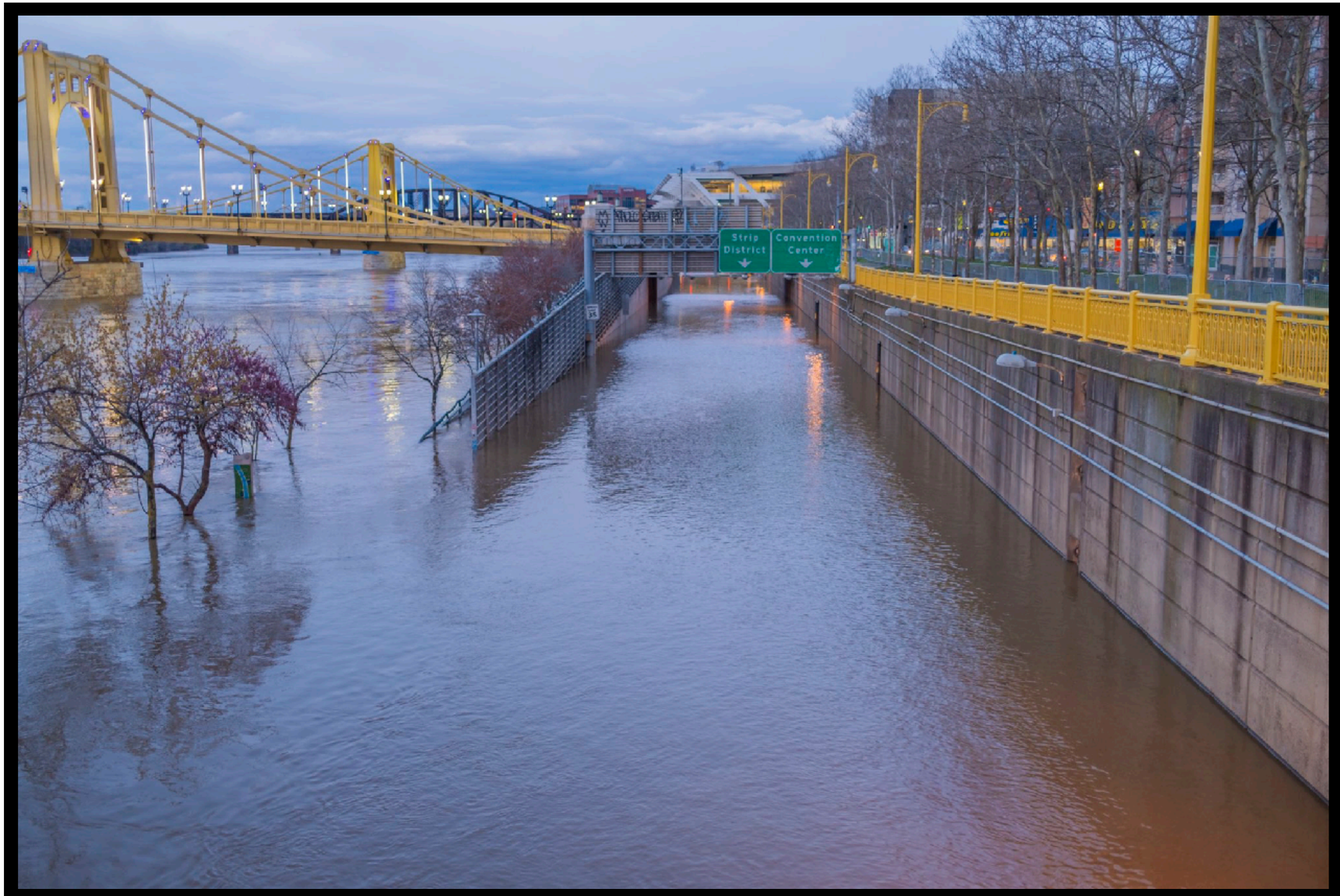


(Eghbal, 2016)

"A pool of programmers across the globe could recreate all of OSS that is being widely used for a cost of approximately $8.80 trillion."

(Hoffmann, Nagle, & Zhou, 2024)

Infrastructure needs regular upkeep and maintenance



Dave DiCello
@DaveDiCello

More images of the flooding in #pittsburgh from last night. The Allegheny River rose above the wall on the North Shore while the Bathtub downtown was filling up as well. The fountain at the Point is almost completely underwater as the rivers rose to 27'.

2:28 PM · Apr 4, 2024 · **69.3K** Views

Infrastructure needs regular upkeep and maintenance

Surreal images show city bus swallowed by sinkhole in downtown Pittsburgh

Authorities expect it will take hours to extract the bus as crews attempt to turn off power lines directly beneath the bus.

Authorities secure the scene after a Port Authority bus fell through a sinkhole in Pittsburgh on Oct. 28, 2019.

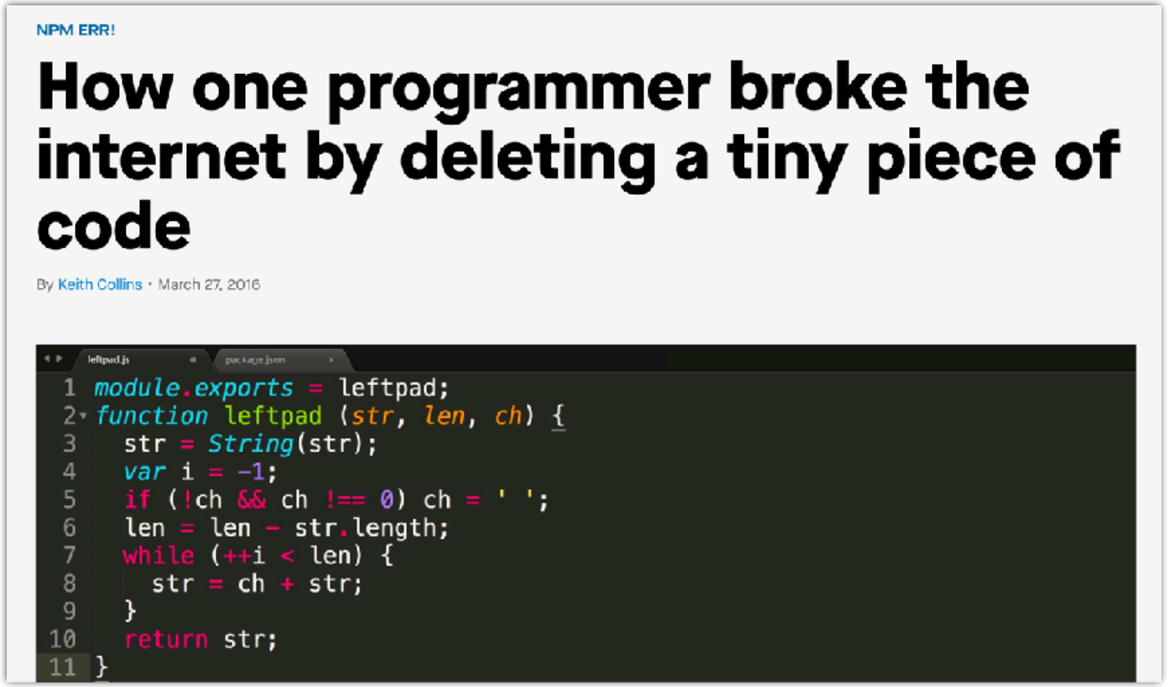
Darrell Sapp / Pittsburgh Post-Gazette via AP

Pittsburgh bridge collapses, drops city bus into ravine

BY GENE PUSKAR AND MARK SCOLFORO

Published 3:16 AM GMT+1, January 29, 2022

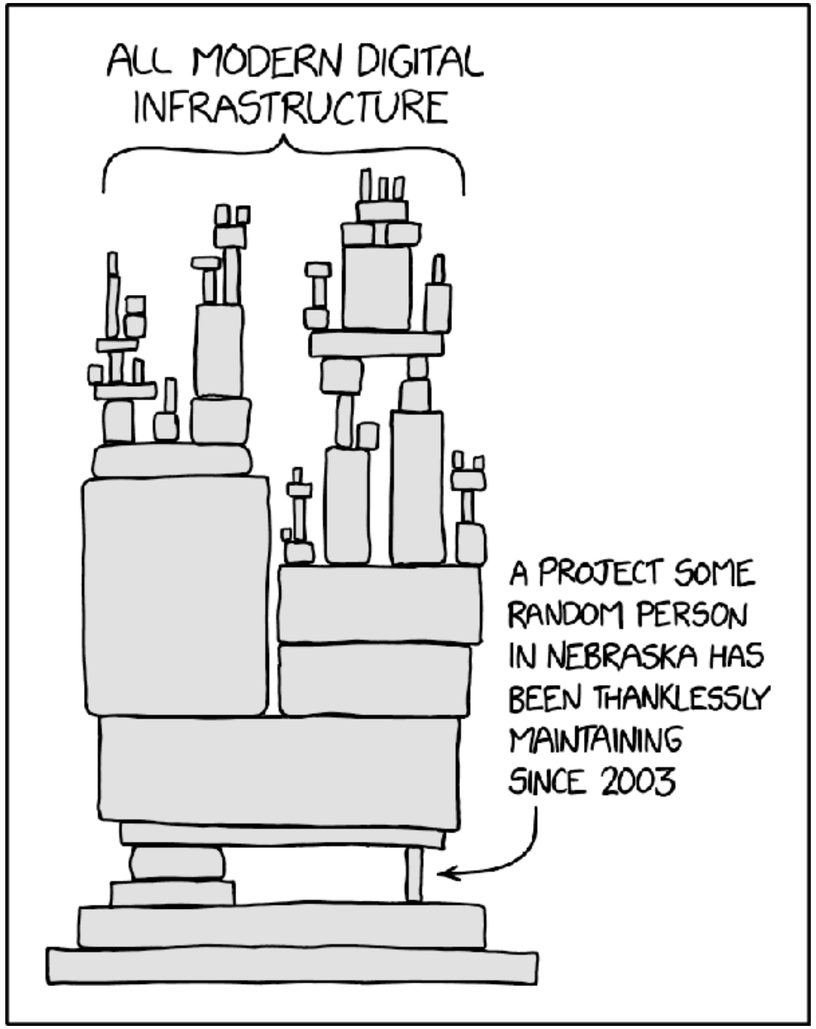
Like any infrastructure, open source software also needs regular upkeep and maintenance

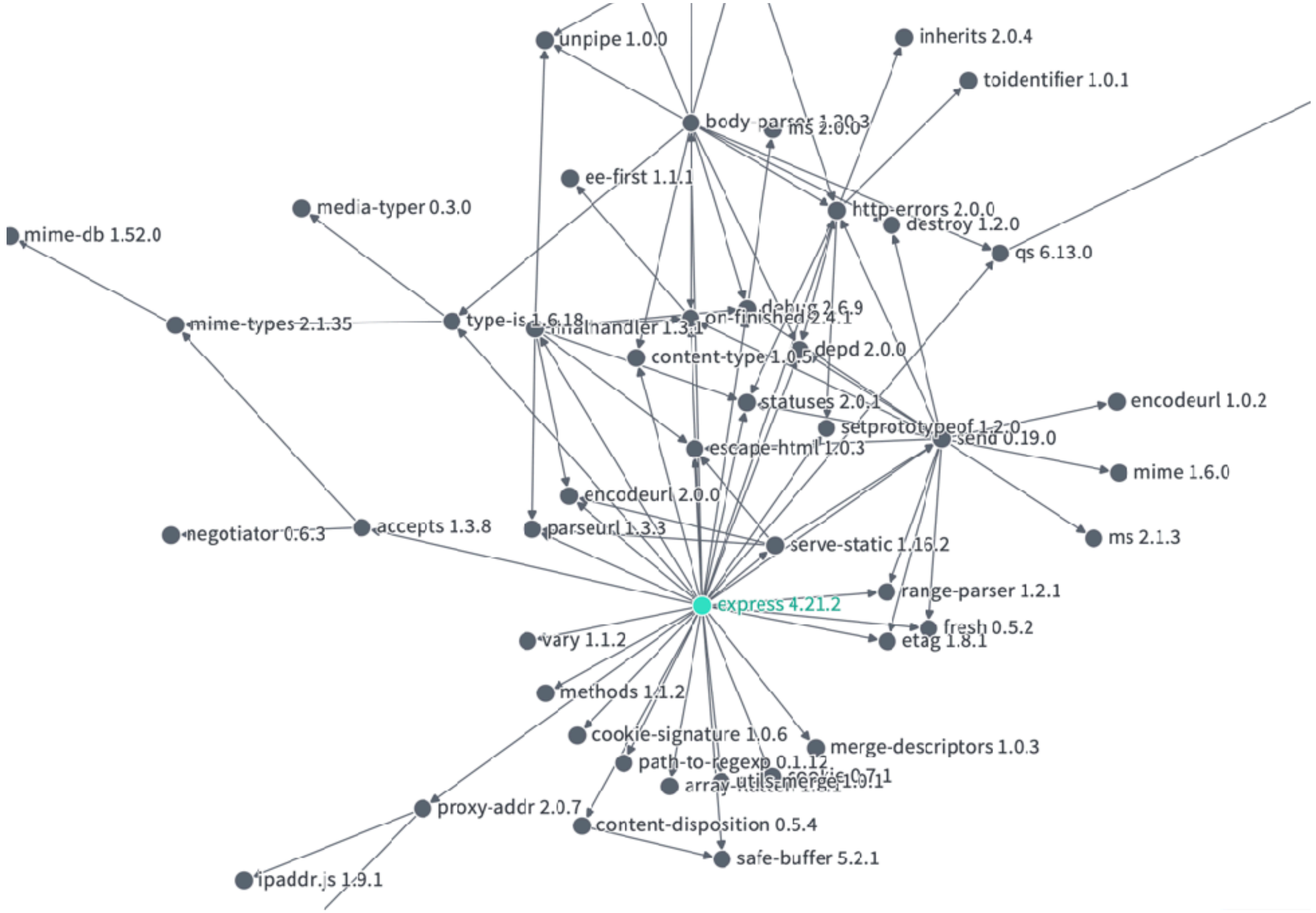


https://qz.com/646467/how-one-programmer-broke-the-internet-by-deleting-a-tiny-piece-of-code/

https://www.zdnet.com/article/equifax-confirms-apache-struts-flaw-it-failed-to-patch-was-to-blame-for-data-breach/

One thing stories like these have in common: The expose the heavily interconnected & interdependent nature of modern software





Dependency graph for the open-source framework express 4.21.2.
(Google Open Source Insights Project, 2021)

Sustaining open source is hard

STRIDEL sustainability research on …

Project practices

- CHASE 2023 (social media)
- ICSE 2020 (forking)
- FSE 2019 (forking)
- FSE 2018 (abandonment factors)

Funding models

- ICSE 2020 (donations)

Sunsetting

- FSE 2023
- ICSE 2025

(dealing with abandonment)

Attracting contributors

- ICSE 2022 (Twitter)
- MSR 2020 (Twitter)
- CSCW 2019 (signals)
- FSE 2015 (social connections)

Transparency and signaling

- FSE 2020 (diffusion of practices)
- CSCW 2019 (signals)
- ICSE 2018 (badges)

Stress, burnout, disengagement

- ICSE 2022 (toxicity theory)
- ICSE SEIS 2022 (toxicity vs pushback)
- ICSE NIER 2020 (toxic language)
- ICSE 2019 (overwork)
- OSS 2019 (dropout, survival analysis)

Diversity and inclusion

- CHI 2023 (ClimateCoach)
- ICSE SEIS 2023 (census)
- ICSE 2019 (social capital)
- CHI 2015 (gender & tenure)
- CHASE 2015 (survey)

Novelty and innovation

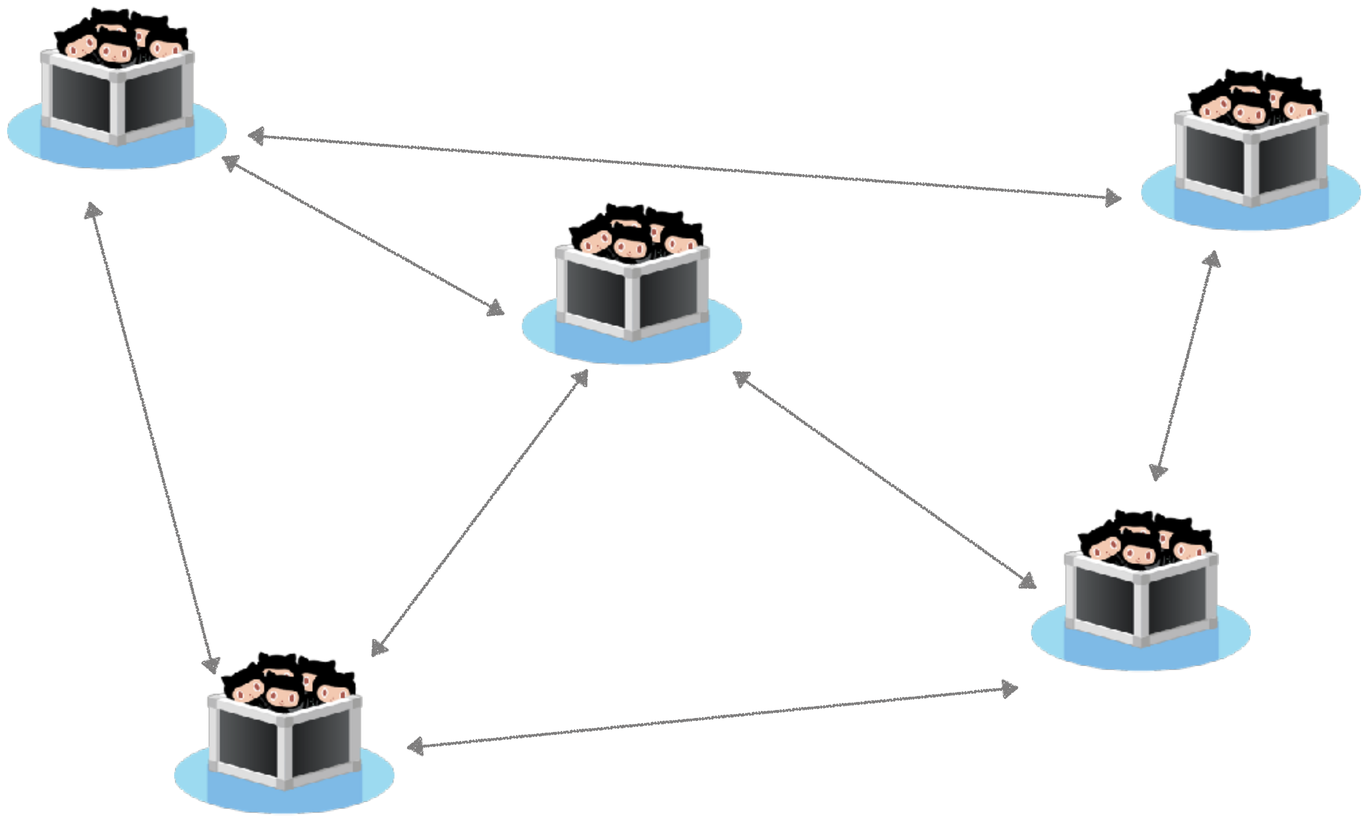
- ICSE 2024 (atypical combinations)

Network effects

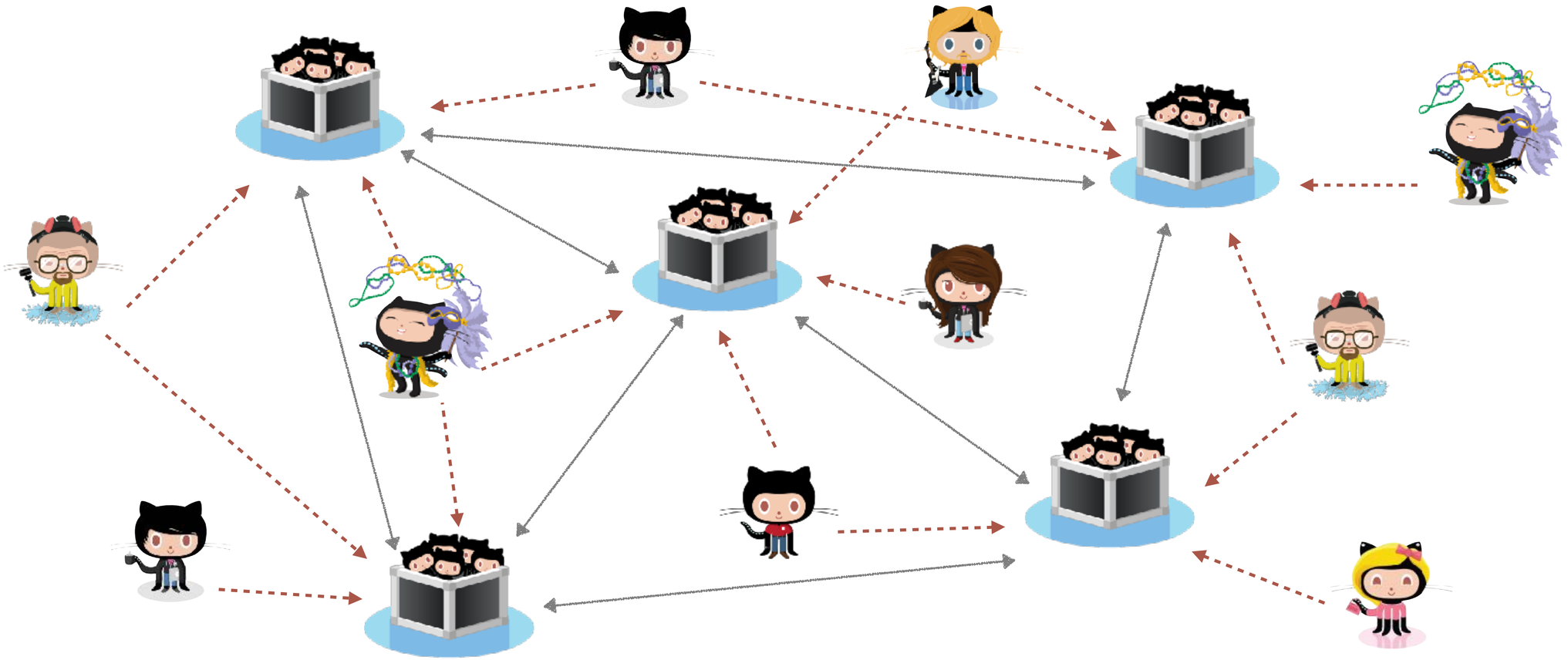
- ICSE 2024 (innovation)
- FSE 2023 (labor pools)
- ICSE 2022 (Twitter)
- FSE 2020 (diffusion of practices)
- ICSE 2019 (social capital)
- FSE 2018 (abandonment factors)

Today:
A look at
network effects

Contributors and projects form complex socio-technical networks!



Contributors and projects form complex socio-technical networks!



Let’s look at some concrete examples of network effects

The emergence of innovation

Social capital

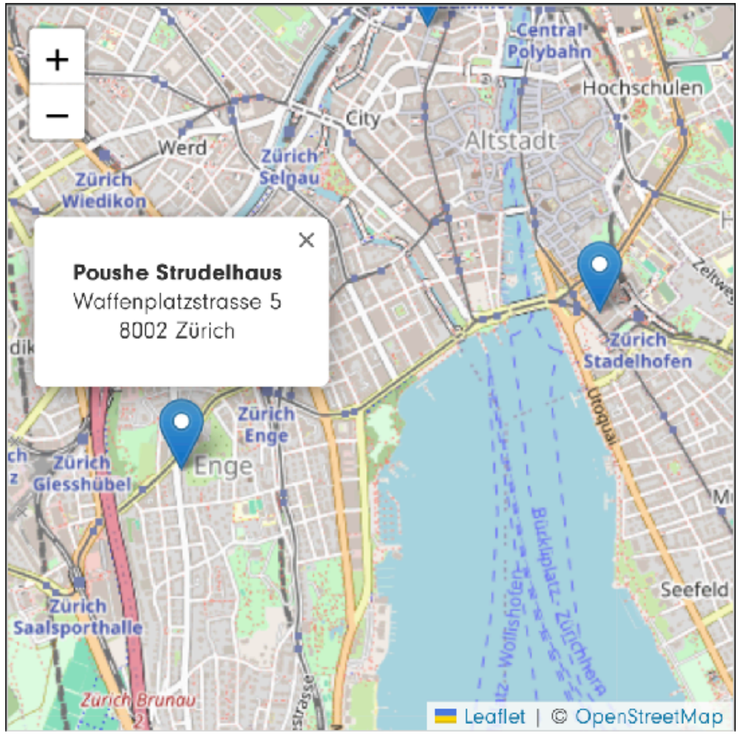
Social contagion

STRUDEL

Let’s look at some concrete examples of network effects

The emergence of innovation

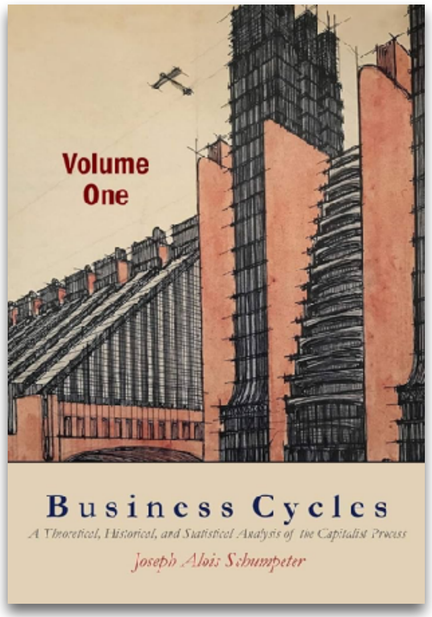
Apfel mit Dark Chocolate Toping (poushe.ch)



STRUDEL

First, what do we mean by innovation?

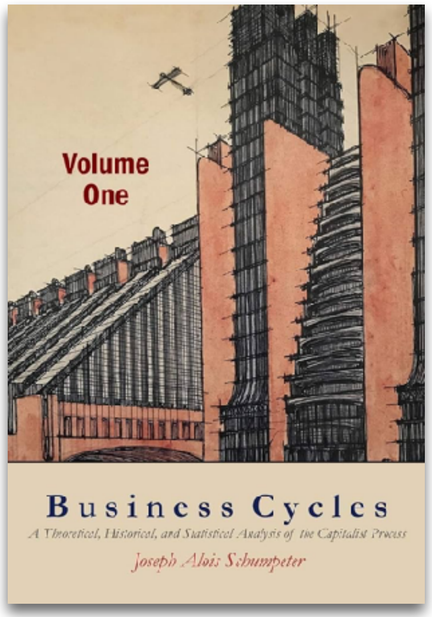
Key idea: Innovation as novel recombination



"[We may say] that innovation combines factors in a new way, or that it consists in carrying out new combinations."

(Schumpeter, 1939)

Key idea: Innovation as novel recombination



(Schumpeter, 1939)

"[We may say] that innovation combines factors in a new way, or that it consists in carrying out new combinations."

"… how scientists search for ideas is premised in part on the idea that teams can span scientific specialties, effectively combining knowledge that prompts scientific breakthroughs."

Atypical Combinations and Scientific Impact

Brian Uzzi,[1,2] Satyam Mukherjee,[1,2] Michael Stringer,[2,3] Ben Jones[1,4*]

Novelty is an essential feature of creative ideas, yet the building blocks of new ideas are often embodied in existing knowledge. From this perspective, balancing atypical knowledge with conventional knowledge may be critical to the link between innovativeness and impact. Our analysis of 17.9 million papers spanning all scientific fields suggests that science follows a nearly universal pattern: The highest-impact science is primarily grounded in exceptionally conventional combinations of prior work yet simultaneously features an intrusion of unusual combinations. Papers of this type were twice as likely to be highly cited works. Novel combinations of prior work are rare, yet teams are 37.7% more likely than solo authors to insert novel combinations into familiar knowledge domains.

Scientific enterprises are increasingly concerned that research within narrow boundaries is unlikely to be the source of the most fruitful ideas (*1*). Models of creativity emphasize that innovation is spurred through original combinations that spark new insights (*2–10*). Current interest in team science and how scientists search for ideas is premised in part on the idea that teams can span scientific specialties, effectively combining knowledge that prompts scientific breakthroughs (*11–15*).

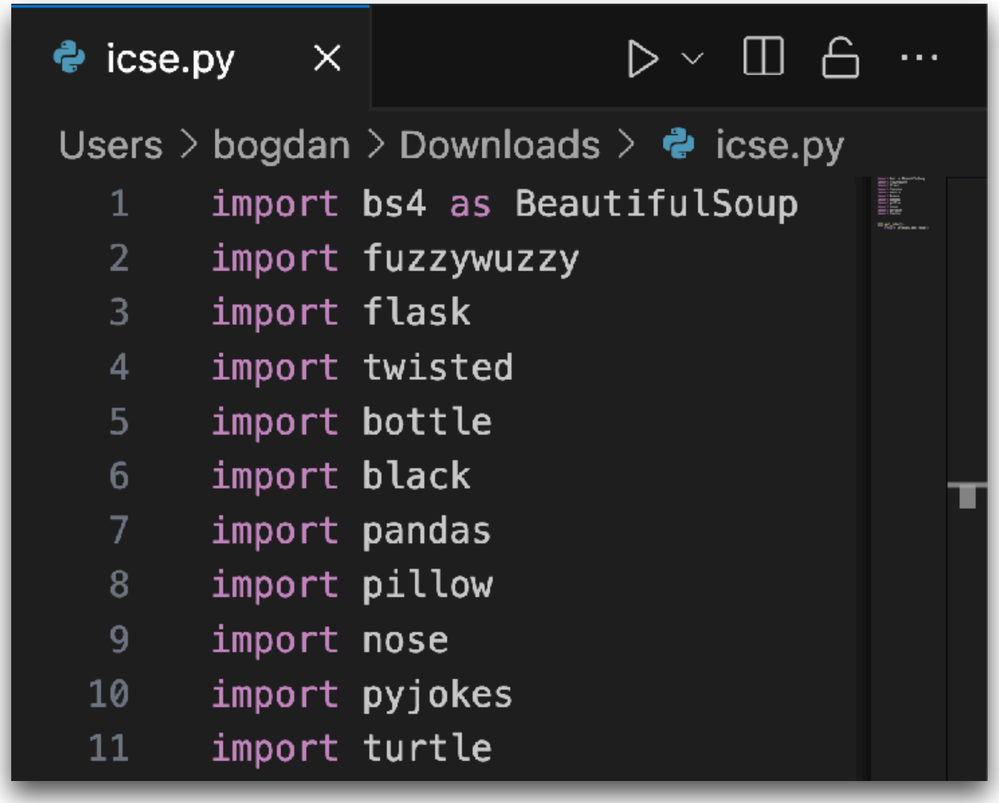
Yet the production and consumption of boundary-spanning ideas can also raise well-known challenges (*16–21*). If, as Einstein believed (*21*), individual scientists inevitably become narrower in their expertise as the body of scientific knowledge expands, then reaching effectively across boundaries may be increasingly challenging (*4*), especially given the difficulty of searching unfamiliar domains (*17, 18*). Moreover, novel ideas can be difficult to absorb (*19*) and communicate, leading scientists to intentionally display conventionality. In his *Principia*, Newton presented his laws of gravitation using accepted geometry rather than his newly developed calculus, despite the latter's importance in developing his insights (*22*). Similarly, Darwin devoted the first part of the *Origin of Species* to conventional, well-accepted knowledge about the selective breeding of dogs, cattle, and birds. From this viewpoint, the balance

[1]Kellogg School of Management, Northwestern University, 2001 Sheridan Road, Evanston, IL 60208, USA. [2]Northwestern Institute on Complex Systems, Northwestern University, 600 Foster, Evanston, IL 60208, USA. [3]Datascope Analytics, 180 West Adams Street, Chicago, IL 60603, USA. [4]National Bureau of Economic Research, 1050 Massachusetts Avenue, Cambridge, MA 02138, USA.

*Corresponding author. E-mail: bjones@kellogg.northwestern.edu

(Uzzi et al, 2013)

Software innovation as novel recombination of software libraries



Lots of combinations:

- **(twisted, bottle)**
- **(turtle, nose)**
- **(black, pandas)**
- **(fuzzywuzzy, pillow)**
- …

C(n,2) unique pairs of packages.

Some of these may be highly innovative because they are atypical.

Software innovation as novel recombination of software libraries

icse.py

Users > bogdan > Downloads > icse.py

```
1   import bs4 as BeautifulSoup
2   import fuzzywuzzy
3   import flask
4   import twisted
5   import bottle
6   import black
7   import pandas
8   import pillow
9   import nose
10  import pyjokes
11  import turtle
```

Lots of combinations:

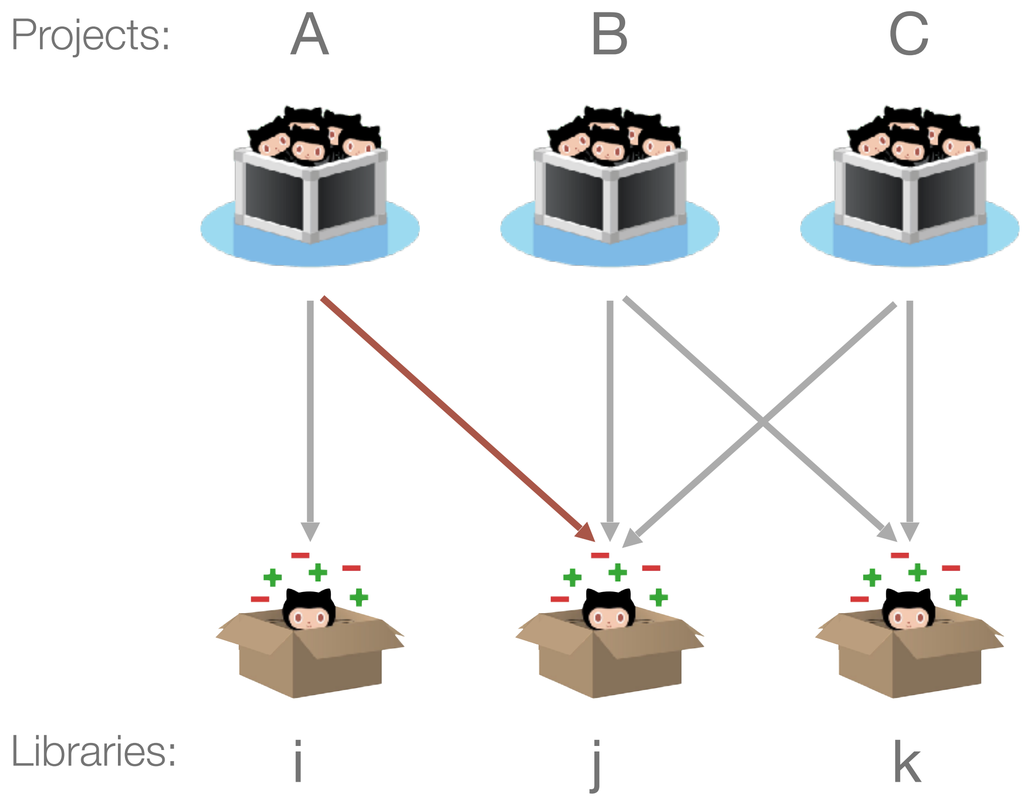
- **(twisted, bottle)**
- **(turtle, nose)**
- **(black, pandas)**
- **(fuzzywuzzy, pillow)**
- …

C(n,2) unique pairs of packages.

Dark chocolate + apple strudel is arguably innovative because it is atypical.

Key idea from network science: Comparison to null (random) model

Observed reality:

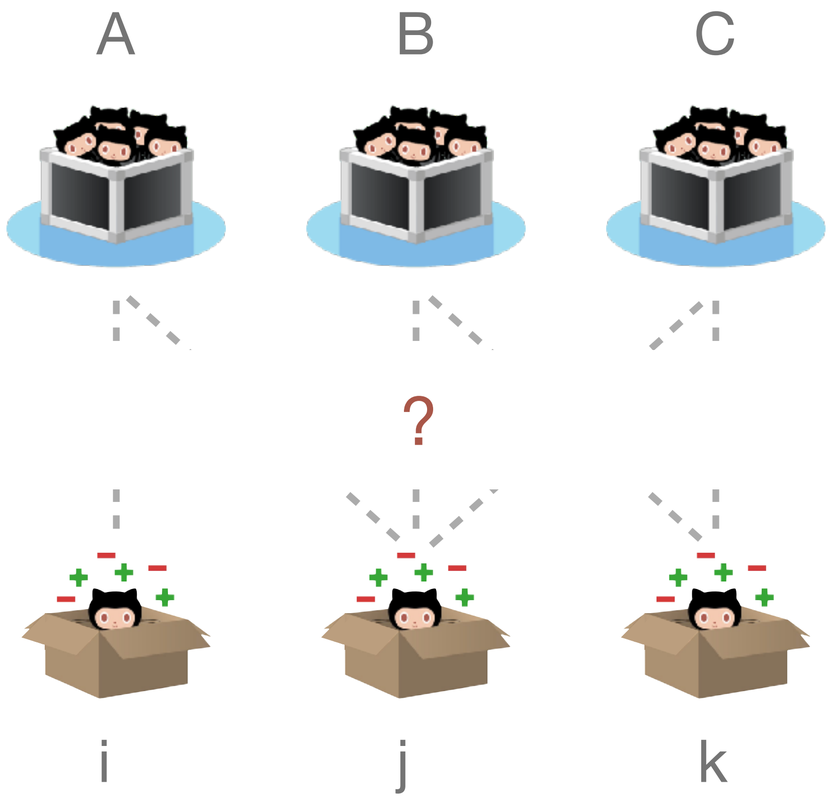


Project A adds a dependency on package j.
New combinations are formed, e.g., (i, j).

How atypical is (i, j)?

Key idea from network science: Comparison to null (random) model

Counterfactual:

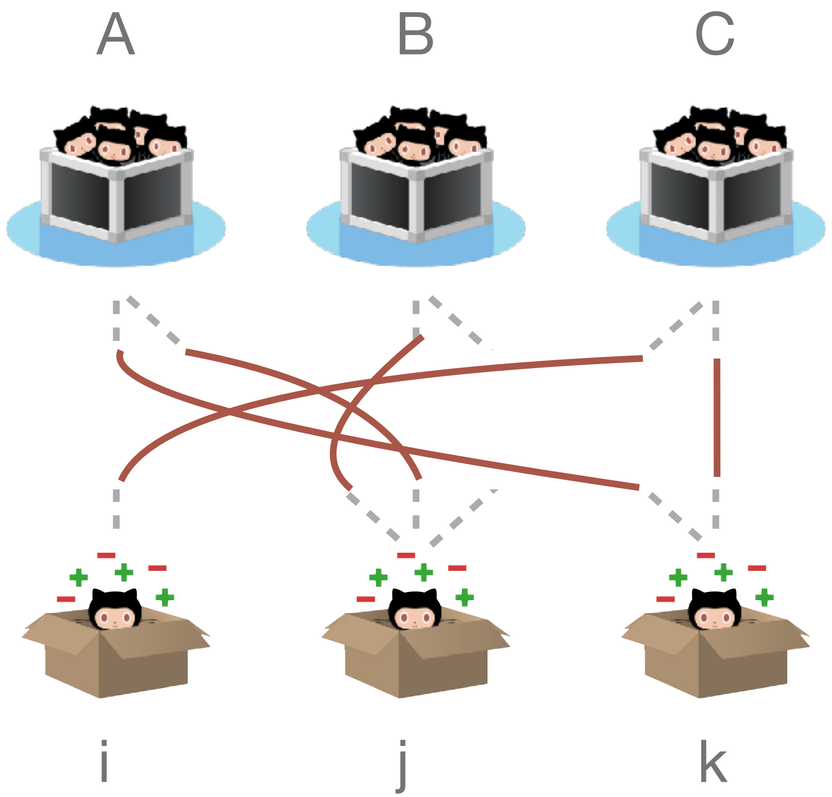


Preserve:

- all the projects
- all the libraries
- the distribution of imports per project
- the distribution of imports per library

Key idea from network science: Comparison to null (random) model

Counterfactual:



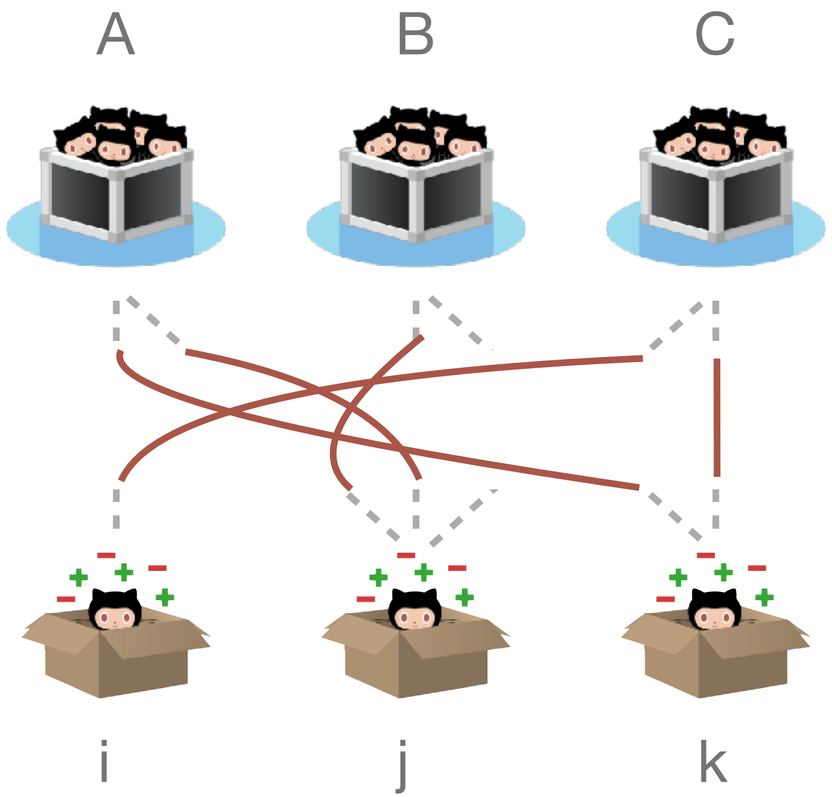
Preserve:

- all the projects
- all the libraries
- the distribution of imports per project
- the distribution of imports per library

But randomly rewire the network.

Key idea from network science: Comparison to null (random) model

Counterfactual:



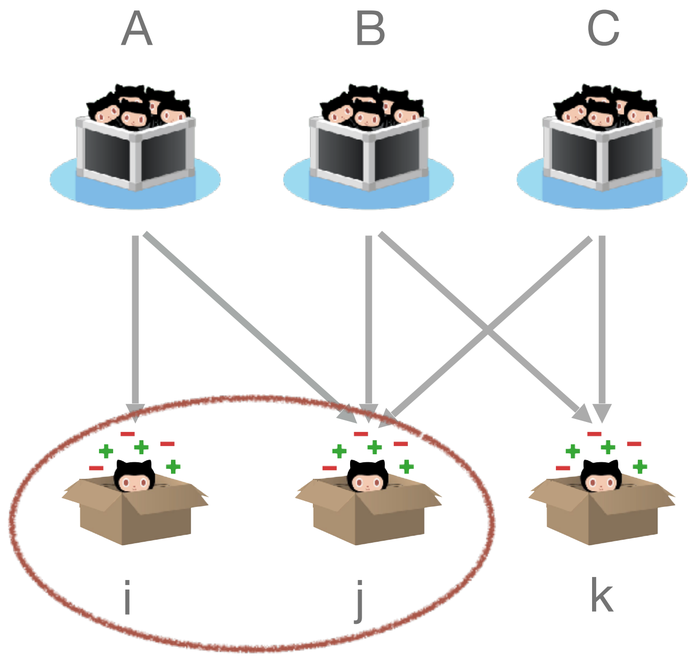
Preserve:

- all the projects
- all the libraries
- the distribution of imports per project
- the distribution of imports per library

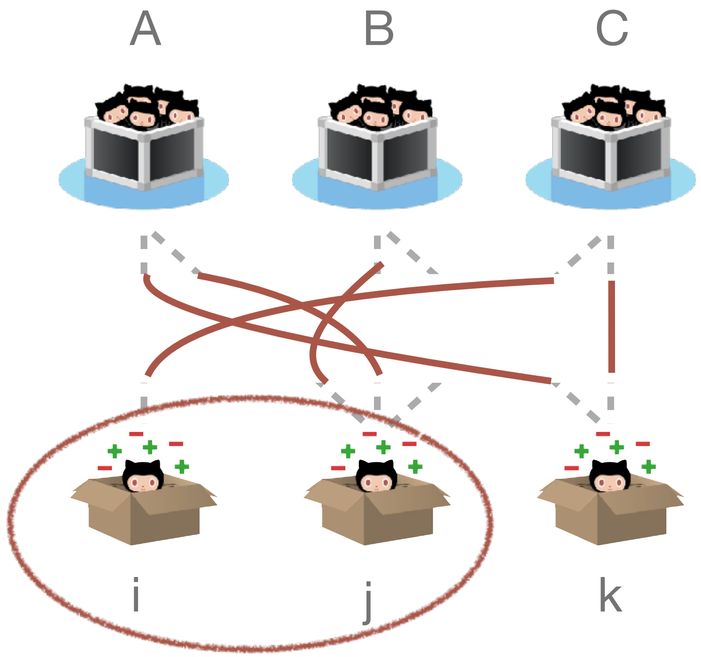
But randomly rewire the network.

And repeat many times.

This z-score estimates if two packages are used together more, less, or about as much as could be expected by chance.



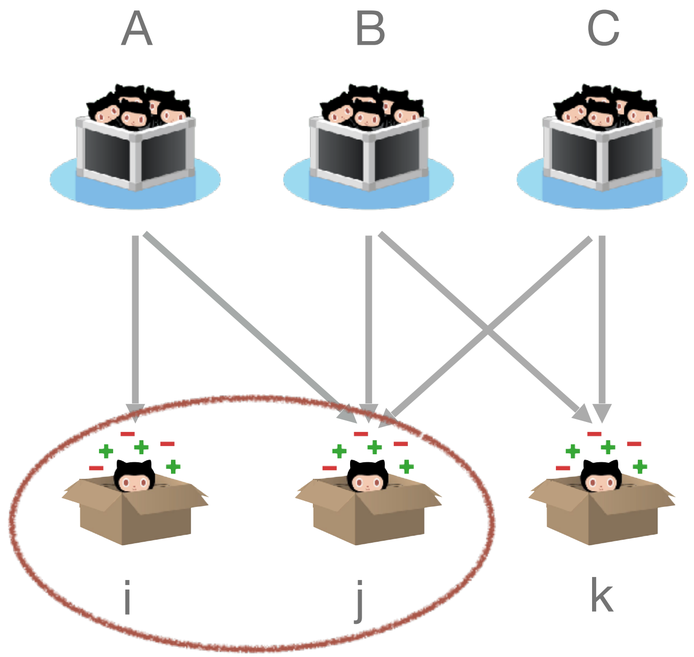
Observed number of times packages i and j appeared together until year t.



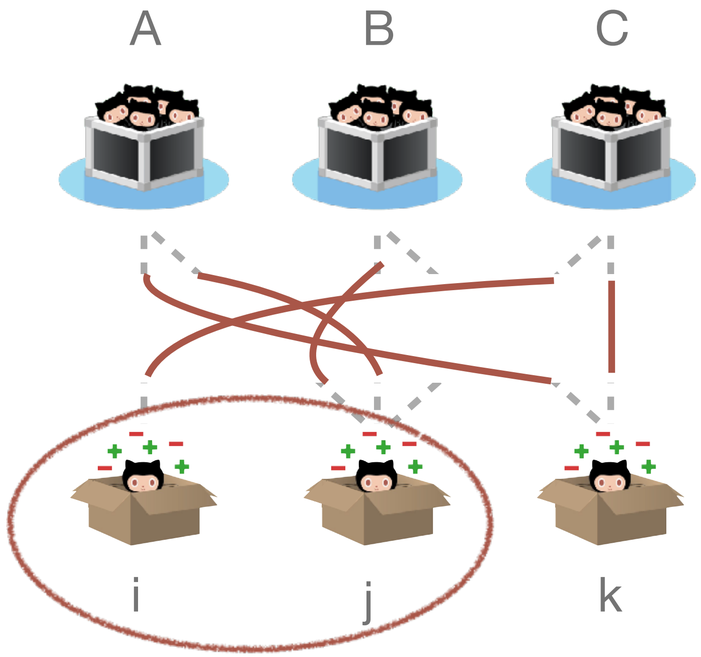
Average (i.e., expected) number of times packages i and j appeared together over N simulations.

$$z_{ijt} = (obs_{ijt} - exp_{ijt})/(\sigma_{ijt})$$

This z-score estimates if two packages are used together more, less, or about as much as could be expected by chance.



Observed number of times packages i and j appeared together until year t.



Average (i.e., expected) number of times packages i and j appeared together over N simulations.

low

high

⇒ atypical combination

$$z_{ijt} = (obs_{ijt} - exp_{ijt})/(\sigma_{ijt})$$

More details in our ICSE 2024 paper

Among others, atypical (novel) projects tend to have more GitHub stars, other factors held constant.

Novelty Begets Long-Term Popularity, But Curbs Participation

A Macroscopic View of the Python Open-Source Ecosystem

Hongbo Fang
hongbofa@cs.cmu.edu
Carnegie Mellon University
Pittsburgh, PA, USA

James Herbsleb
jdh@cs.cmu.edu
Carnegie Mellon University
Pittsburgh, PA, USA

Bogdan Vasilescu
vasilescu@cmu.edu
Carnegie Mellon University
Pittsburgh, PA, USA

ABSTRACT

Who creates the most innovative open-source software projects? And what fate do these projects tend to have? Building on a long history of research to understand innovation in business and other domains, as well as recent advances towards modeling innovation in scientific research from the science of science field, in this paper we adopt the analogy of innovation as emerging from the novel recombination of existing bits of knowledge. As such, we consider as innovative the software projects that recombine existing software libraries in novel ways, i.e., those built on top of *atypical combinations* of packages as extracted from import statements. We then report on a large-scale quantitative study of innovation in the Python open-source software ecosystem. Our results show that higher levels of innovativeness are statistically associated with higher GitHub star counts, i.e., novelty begets popularity. At the same time, we find that controlling for project size, the more innovative projects tend to involve smaller teams of contributors, as well as be at higher risk of becoming abandoned in the long term. We conclude that innovation and open source sustainability are closely related and, to some extent, antagonistic.

CCS CONCEPTS

• **Software and its engineering → Open source model**.

KEYWORDS

Open-source software, innovation

ACM Reference Format:
Hongbo Fang, James Herbsleb, and Bogdan Vasilescu. 2024. Novelty Begets Long-Term Popularity, But Curbs Participation: A Macroscopic View of the Python Open-Source Ecosystem. In *2024 IEEE/ACM 46th International Conference on Software Engineering (ICSE 2024), April 14–20, 2024, Lisbon, Portugal.* ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3597503.3608142

1 INTRODUCTION

It has long been recognized that open-source software development is an avenue for innovation and creative expression — "how creative a person feels when working on the project is the strongest and most pervasive driver" of participation in open source [29]. Unsurprisingly, we have seen an explosion in production of open-source software, especially in the last decade, with the proliferation of the social coding philosophy [10]. Nowadays, open-source software seems more popular than ever before [12], and it is hard to find any sectors of the economy that do not rely heavily on open-source infrastructure [15].

At the same time, with growing use and ubiquity of open source, there are growing concerns about the maintainability and sustainability of this digital infrastructure [21, 34, 38, 45]. It is not always without barriers for newcomers to join projects [33, 42], turnover rates for open-source contributors are high [27, 35], and even widely-used projects can end up being maintained by a single person or, sometimes, by no one at all [2, 9]. The insufficient maintenance of open-source projects can have disastrous consequences, as prominent security incidents like Heartbleed [47], the Equifax breach [7], and Log4Shell [23] have shown, just to name a few.

These days, significant attention is being paid by policy makers, practitioners, and researchers to understanding open source health and improving open source sustainability, with many open questions remaining around determinants of project success and failure, governance models, procurement and allocation of resources, and others. In this general context we focus on one important but poorly understood concept — innovation. While open source as a whole is a catalyst for innovation [14] (e.g., technology startup companies would have much slower start without access to open-source infrastructure) and understanding, and being able to identify, innovations in other fields has always been of great interest to investors, governments, etc, we know very little about how innovation emerges at the individual project level and what are its consequences, in either open-source or commercial software development.

Taking one step in this direction, in this paper we begin to study innovation in open source in the Schumpeterian tradition [40] of viewing innovation as emerging from the novel recombination of existing bits of knowledge, a typical perspective in the science of science field [18]. Operationalizing innovation at code level as a function of the libraries and packages a project imports (see Section 3.3) — projects built on top of more *atypical combinations* of libraries are considered to be more innovative — we find that in the Python open-source ecosystem projects with higher levels of innovation tend to be more popular on average, in terms of GitHub star counts. Stated differently, novelty begets popularity. At the same time, we find that controlling for project size, projects with higher levels of innovation tend to involve smaller teams and are more likely to become abandoned sooner, suggesting that the benefits of increased popularity also carry a cost of limiting the available labor pool [16] of potential maintainers of a code base that, on average, fewer people may be familiar with.

ICSE 2024, April 14–20, 2024, Lisbon, Portugal

ACM ISBN 979-8-4007-0217-4/24/04.
https://doi.org/10.1145/3597503.3608142

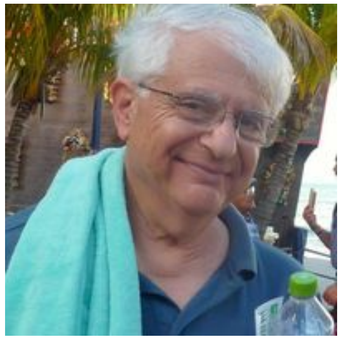
Next, how does innovation emerge?

Once upon a time, a PhD student at Harvard University was writing his dissertation …

Stanford
Sociology
SCHOOL OF HUMANITIES AND SCIENCES

Mark Granovetter

Joan Butler Ford Professor in the School of Humanities and Sciences; Professor of Sociology



A.B. Princeton University 1965
Modern European and American History
Ph.D. Harvard University 1970
Sociology

https://sociology.stanford.edu/people/mark-granovetter

The Strength of Weak Ties[1]

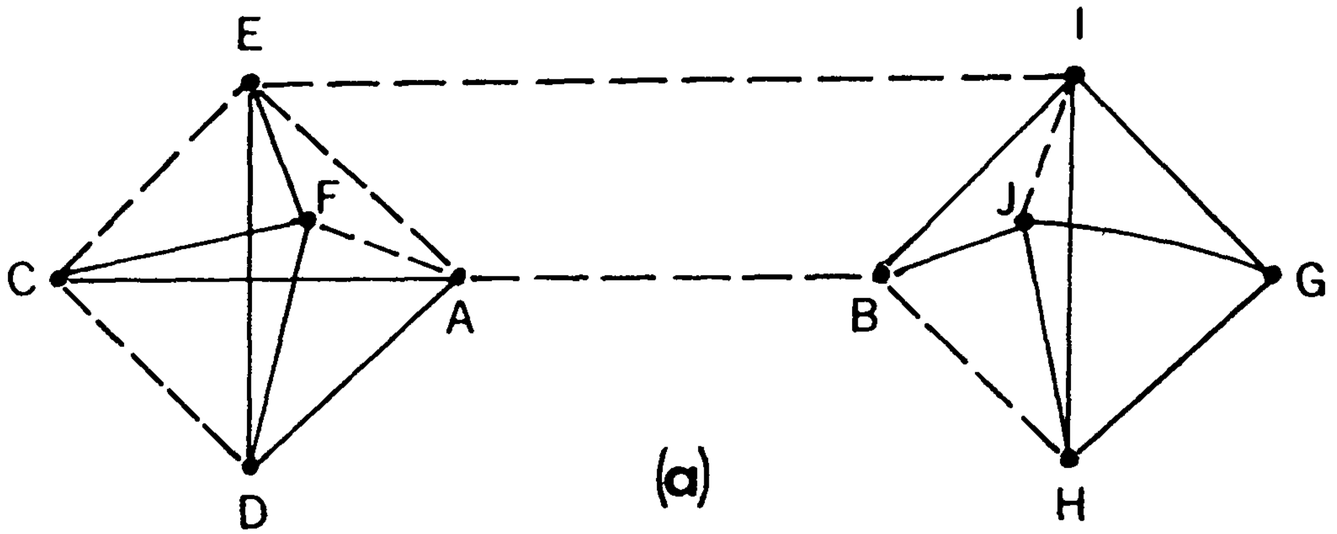
Mark S. Granovetter
Johns Hopkins University

Analysis of social networks is suggested as a tool for linking micro and macro levels of sociological theory. The procedure is illustrated by elaboration of the macro implications of one aspect of small-scale interaction: the strength of dyadic ties. It is argued that the degree of overlap of two individuals' friendship networks varies directly with the strength of their tie to one another. The impact of this principle on diffusion of influence and information, mobility opportunity, and community organization is explored. Stress is laid on the cohesive power of weak ties. Most network models deal, implicitly, with strong ties, thus confining their applicability to small, well-defined groups. Emphasis on weak ties lends itself to discussion of relations *between* groups and to analysis of segments of social structure not easily defined in terms of primary groups.

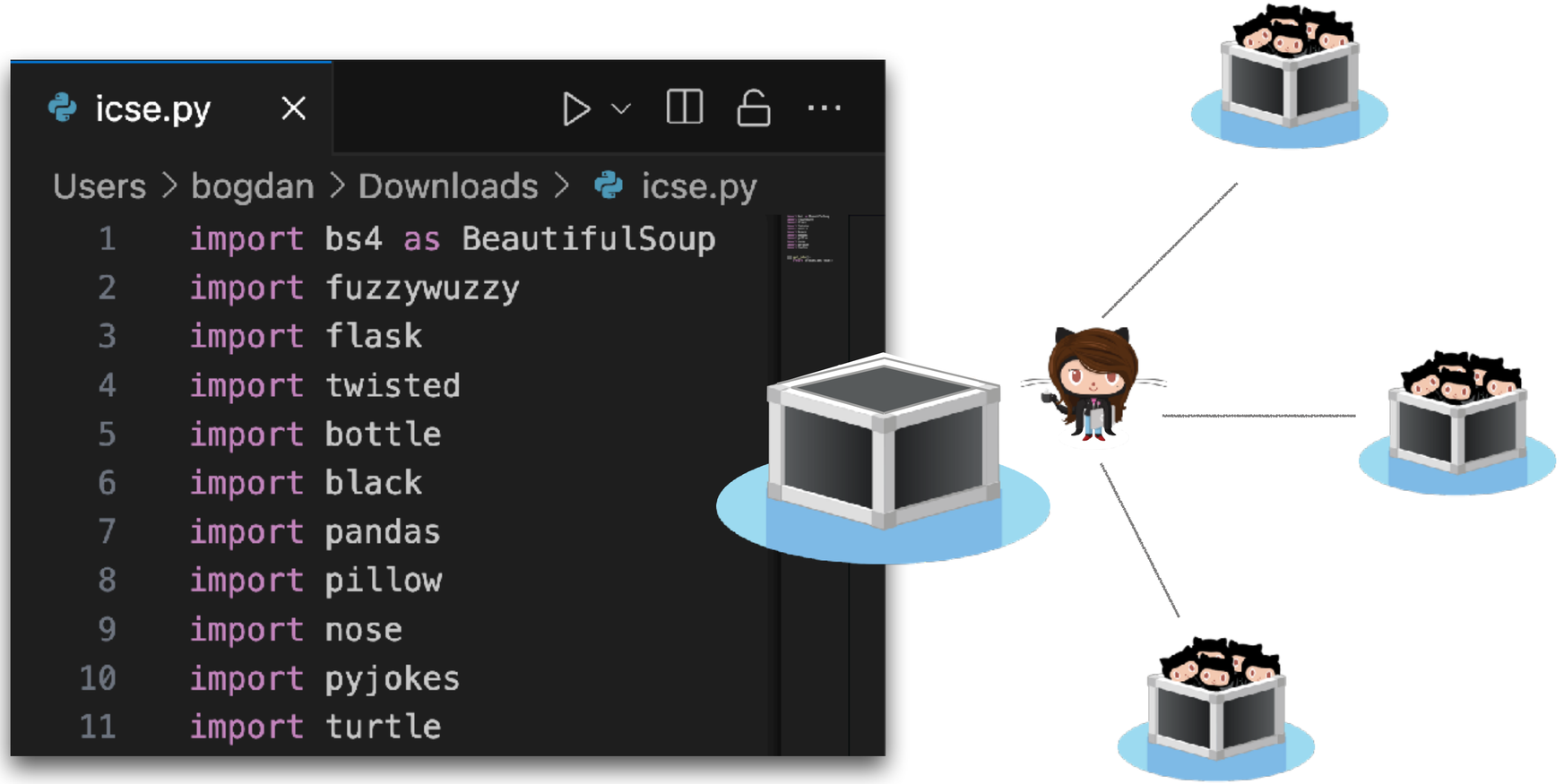
Weak ties are more effective in job searches because they act as bridges.

The majority of people found their jobs through acquaintances (weak ties) rather than close friends or family (strong ties).

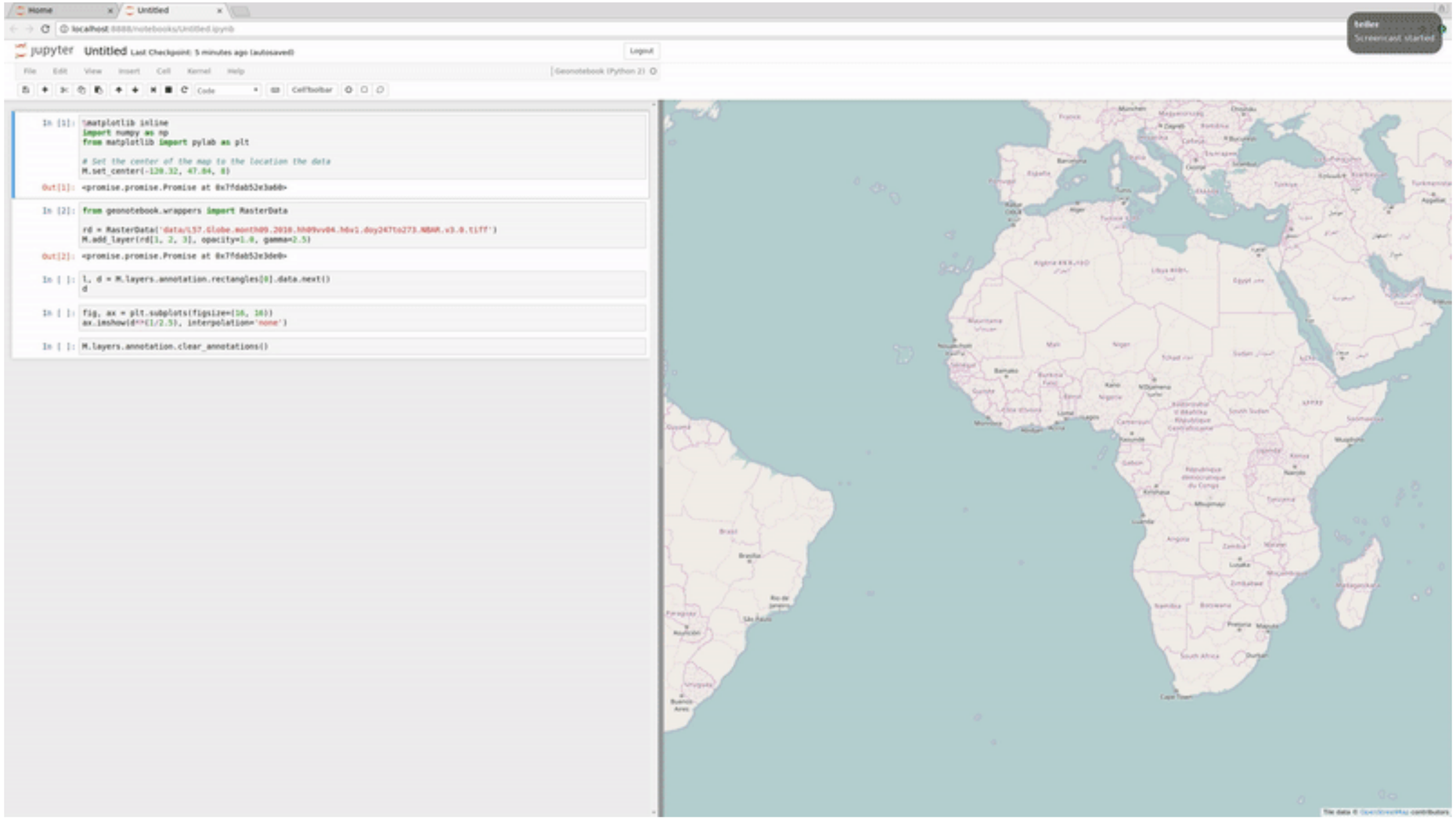
In a random sample of recent professional, technical, and managerial job changers living in a Boston suburb, I asked those who found a new job through contacts how often they *saw* the contact around the time that he passed on job information to them. I will use this as a measure of tie strength.[15] A natural a priori idea is that those with whom one has strong ties are more motivated to help with job information. Opposed to this greater motivation are the structural arguments I have been making: those to whom we are weakly tied are more likely to move in circles different from our own and will thus have access to information different from that which we receive.



Do OSS developers also find their new ideas through weak ties?

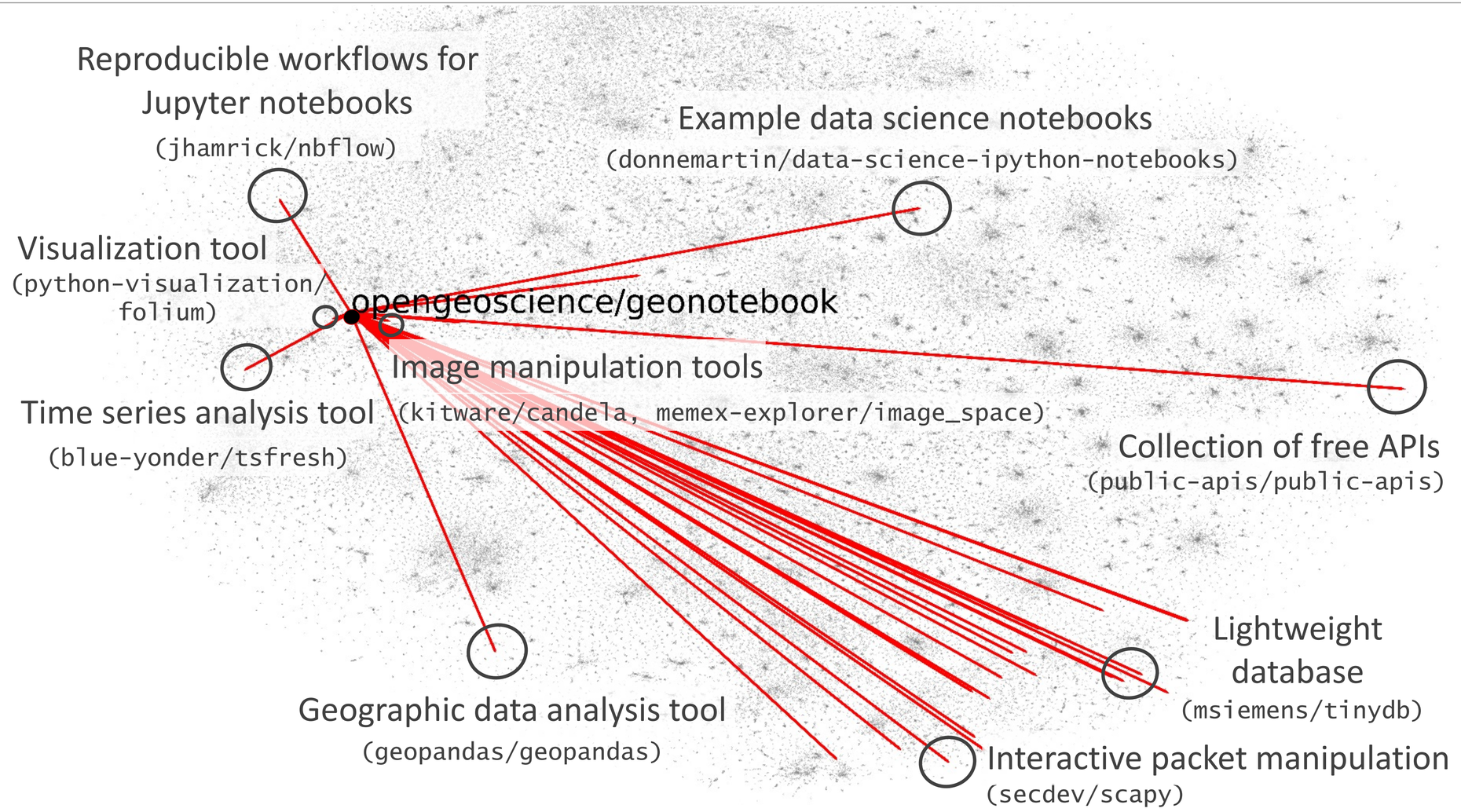


Do OSS developers also find their new ideas through weak ties?



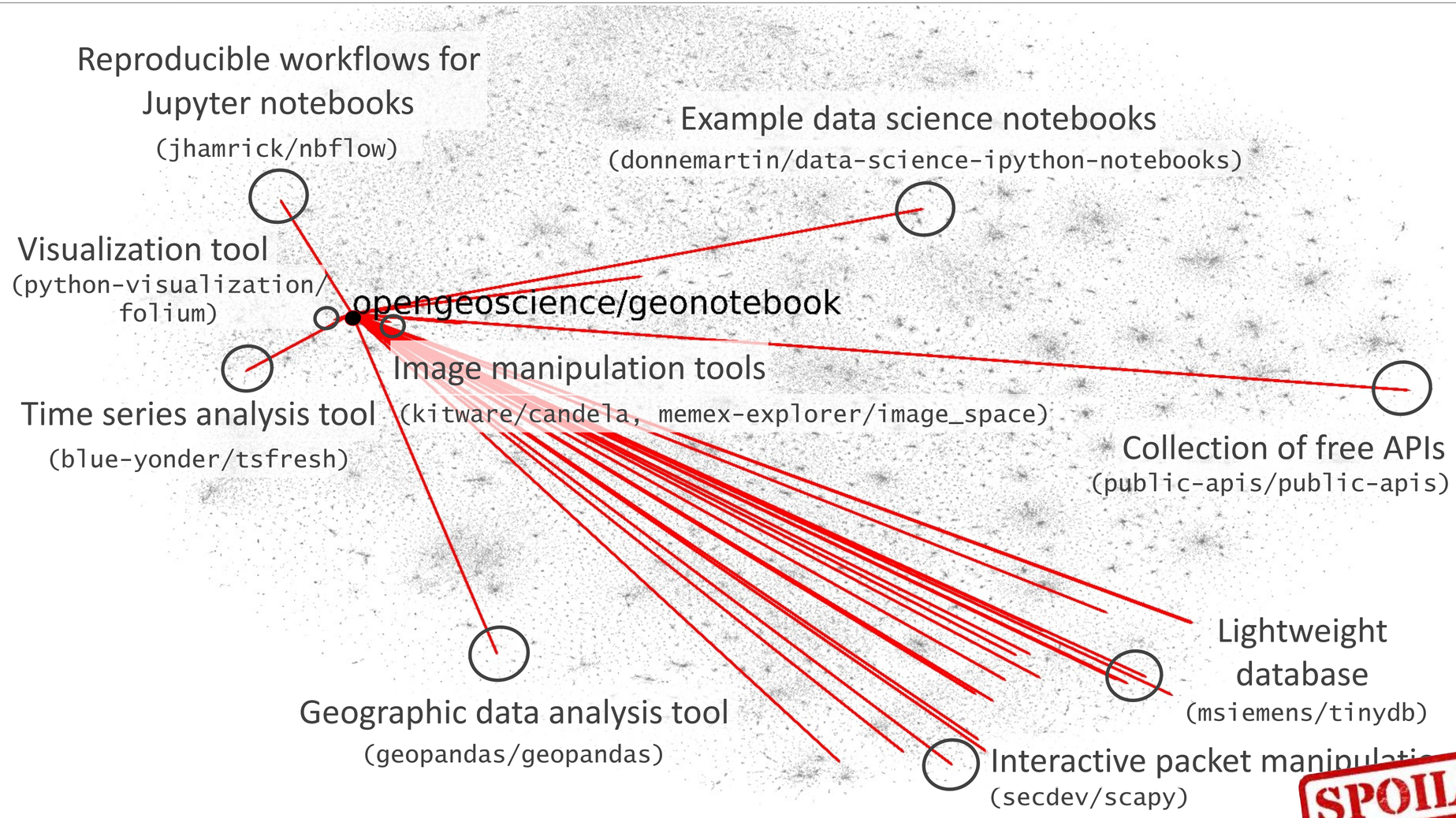
https://github.com/opengeoscience/geonotebook

Do OSS developers also find their new ideas through weak ties?



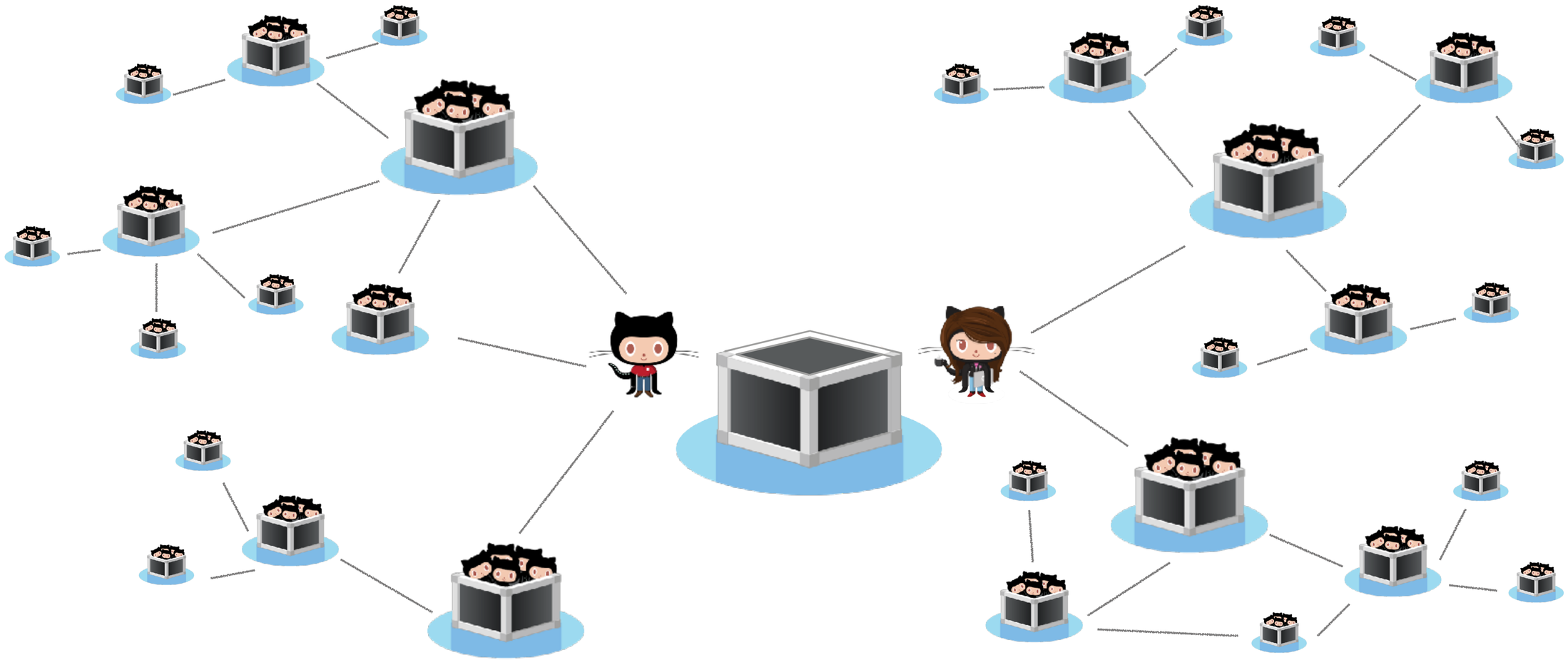
Anecdotally, yes

Do OSS developers also find their new ideas through weak ties?

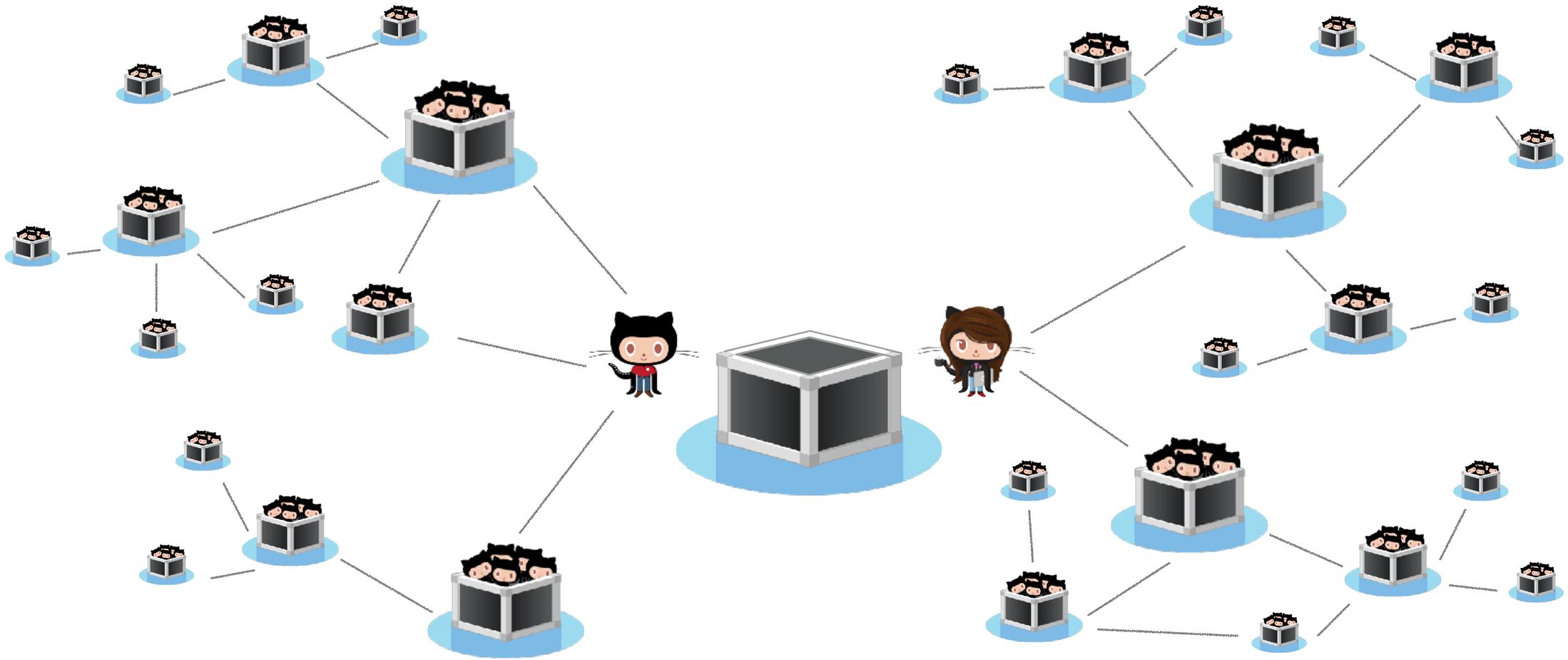


Amazingly, statistically also yes!

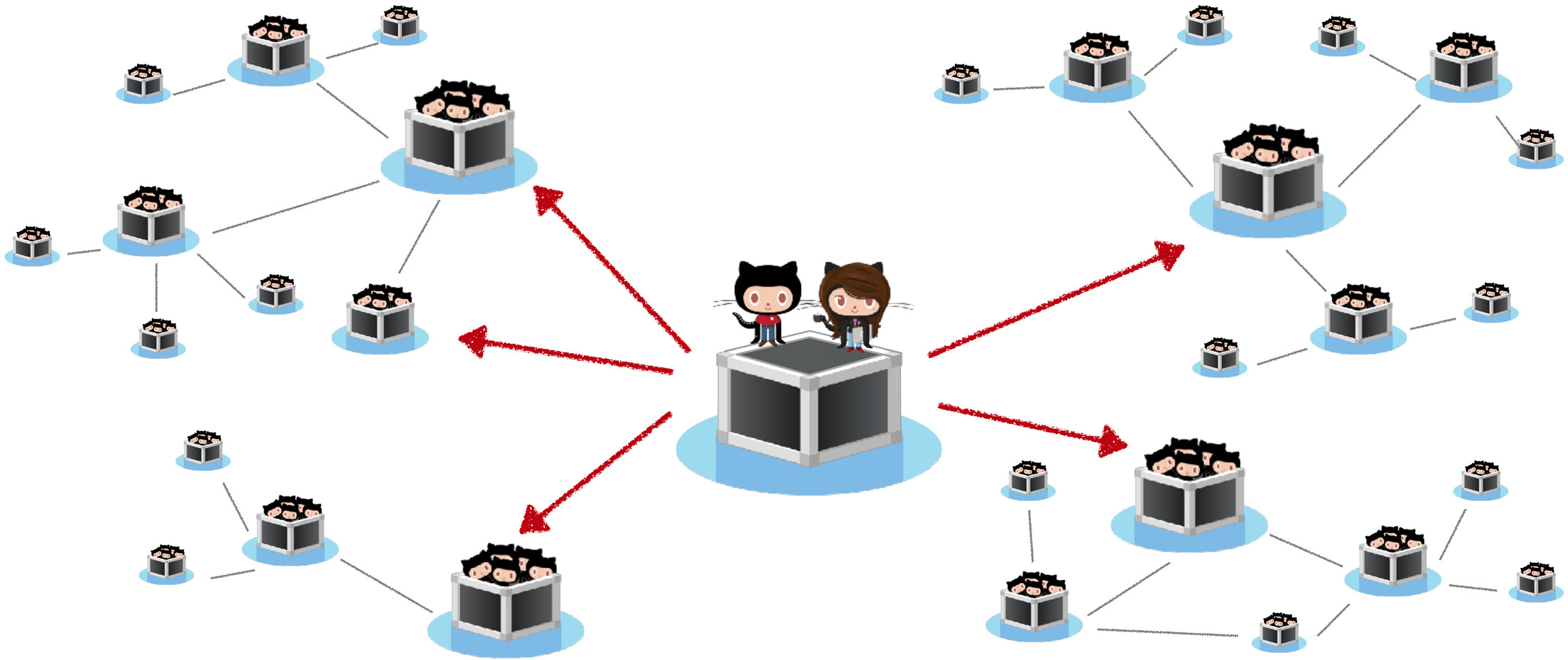
People interact with artifacts and with each other. This creates ties.



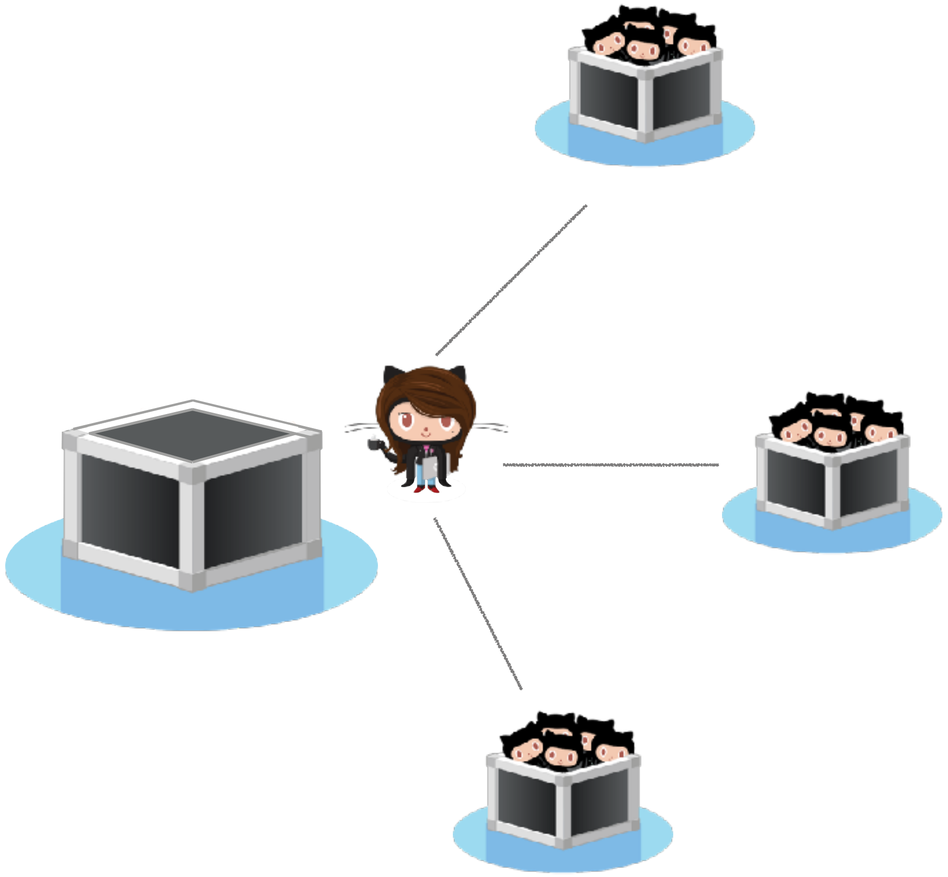
Hypothesis 1: The bigger developers' networks are, the better informed they are, and the more innovative their projects are.



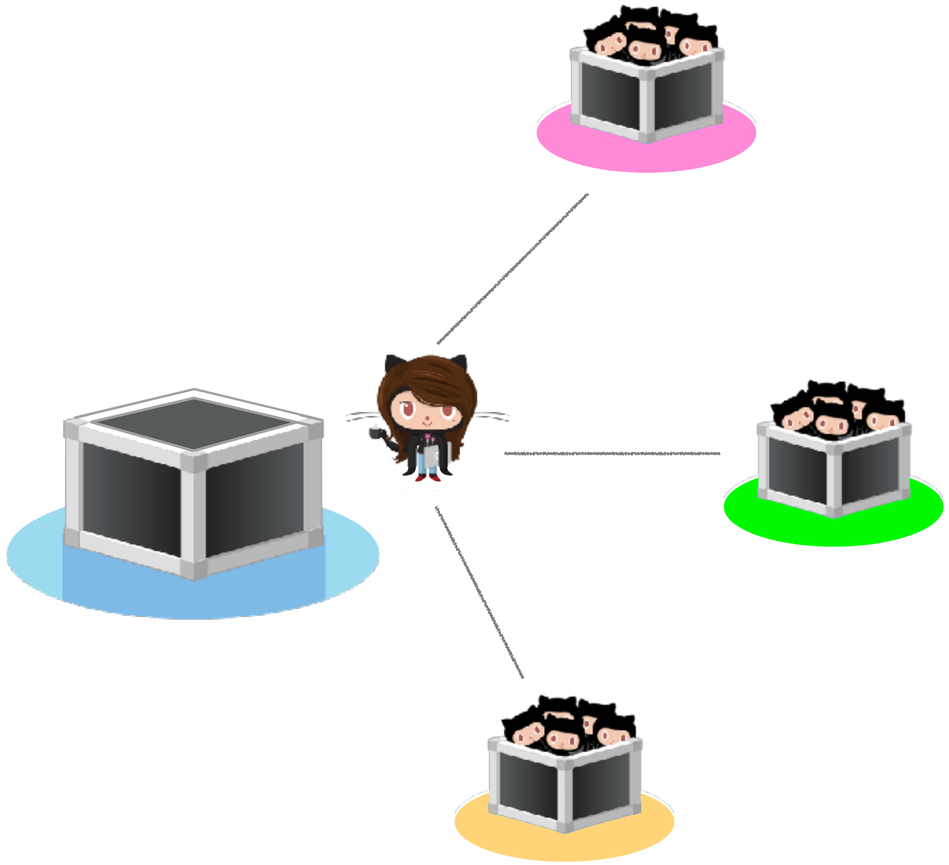
Measure: Out-degree centrality



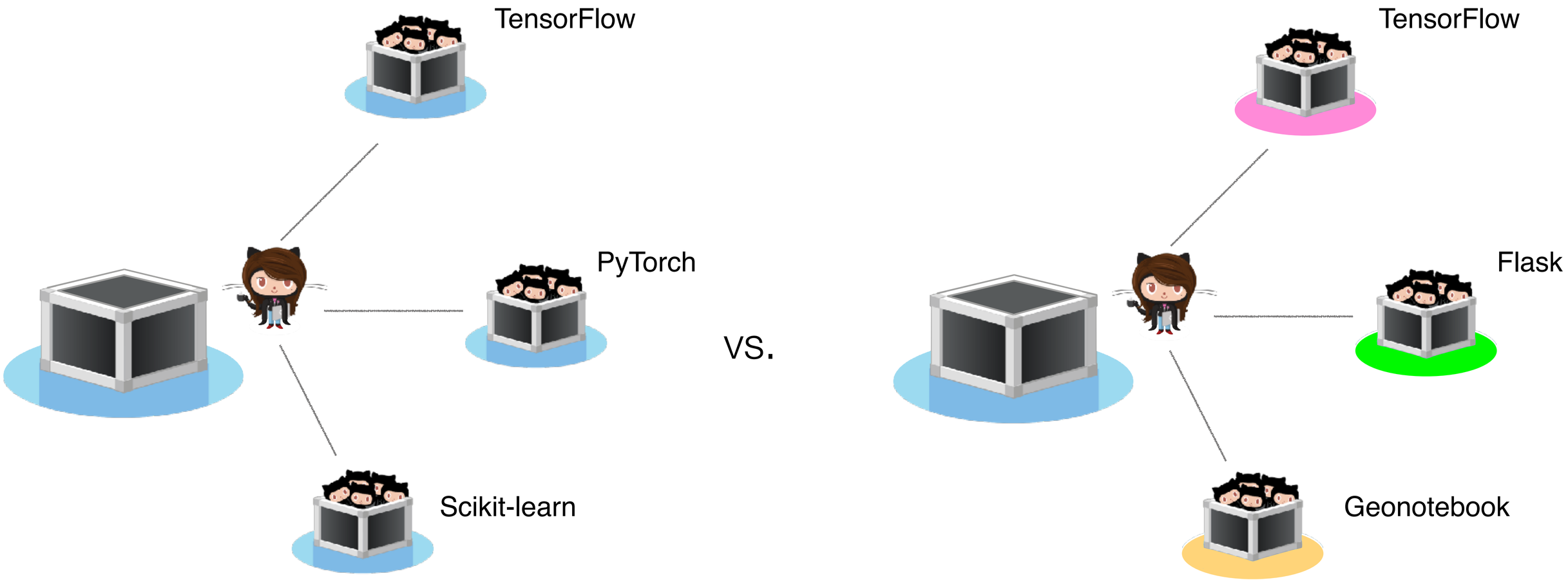
Hypothesis 2: The greater the informational diversity of developers' networks, the more innovative their projects are.



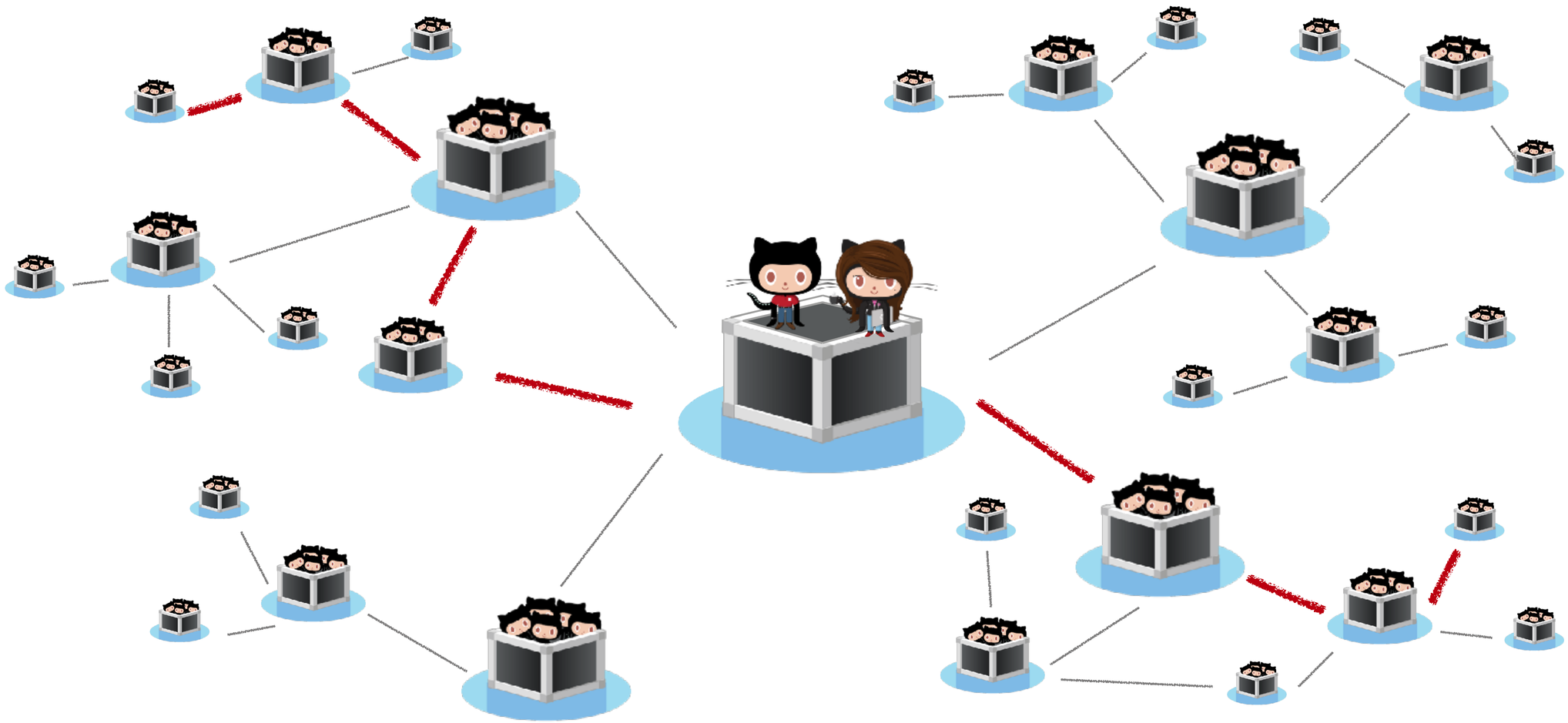
vs.



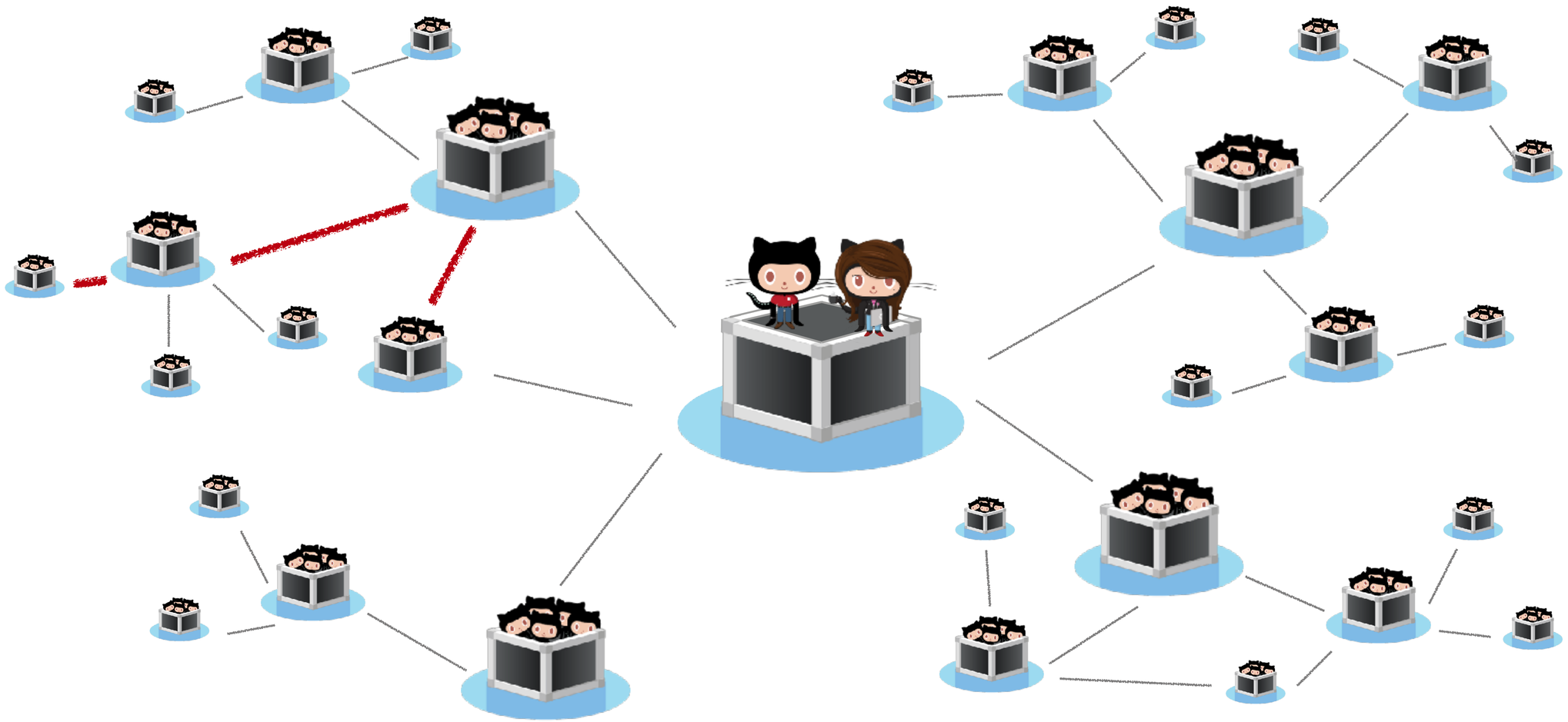
Intuition: Projects on the left have high overlap in the sets of people that interact with them. On the right — low overlap.



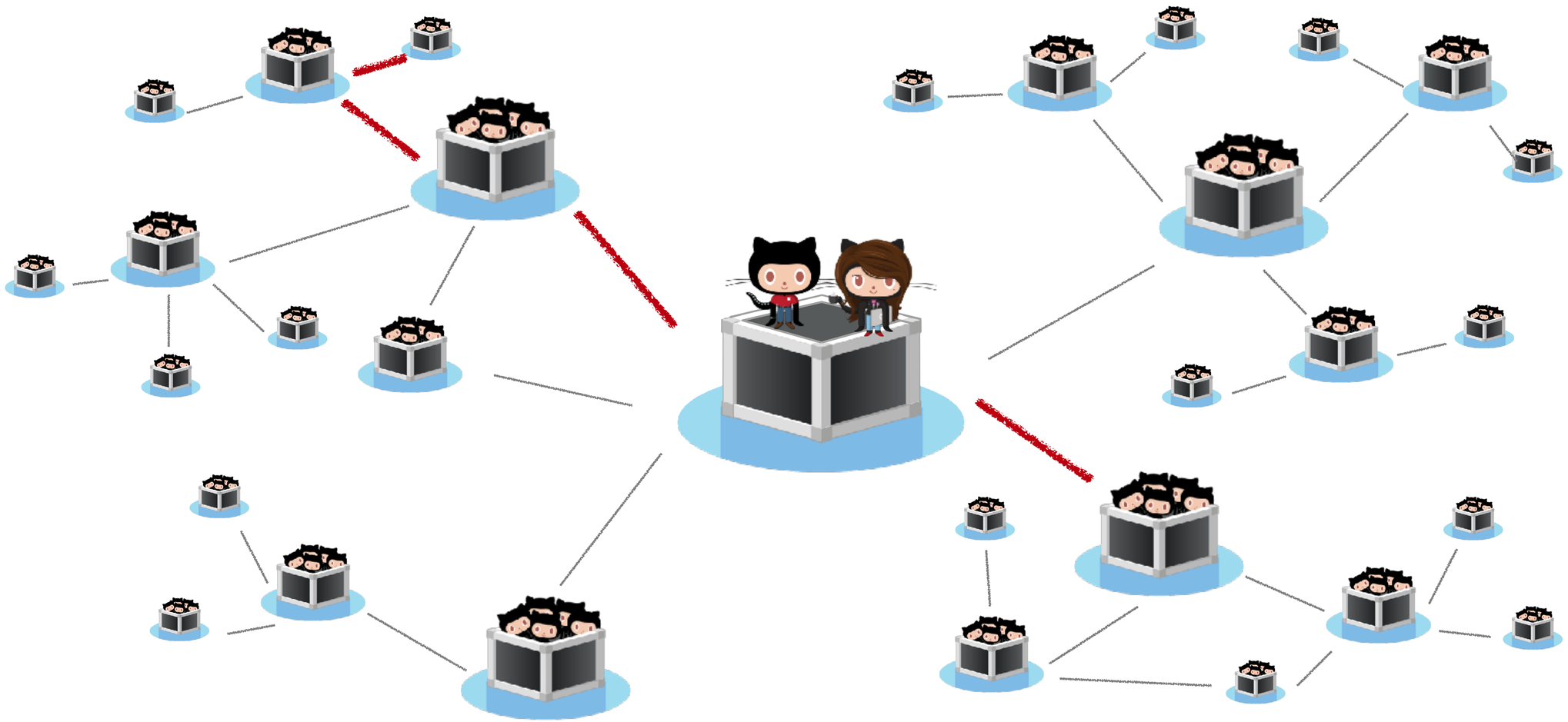
Measure: First, we generate Node2Vec embeddings for each project



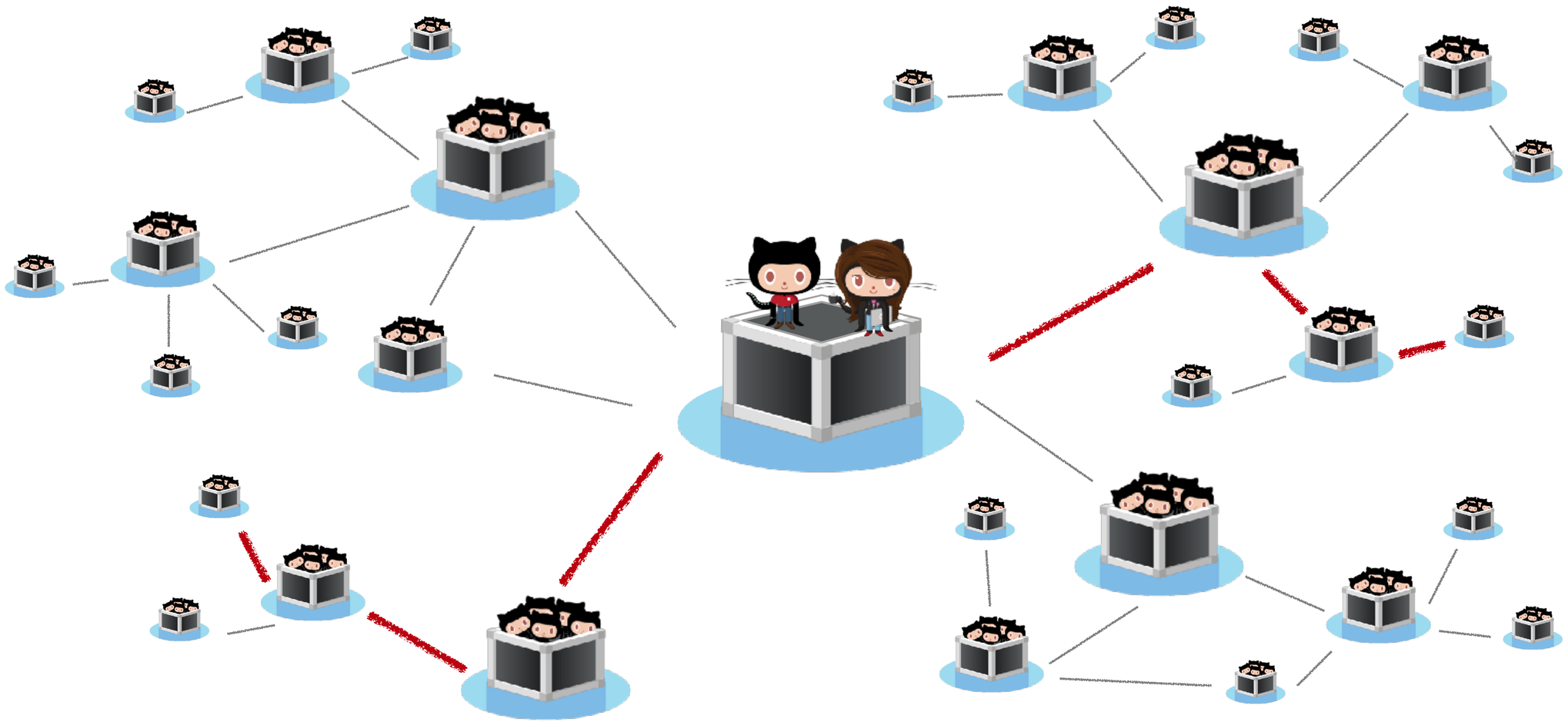
Measure: First, we generate Node2Vec embeddings for each project



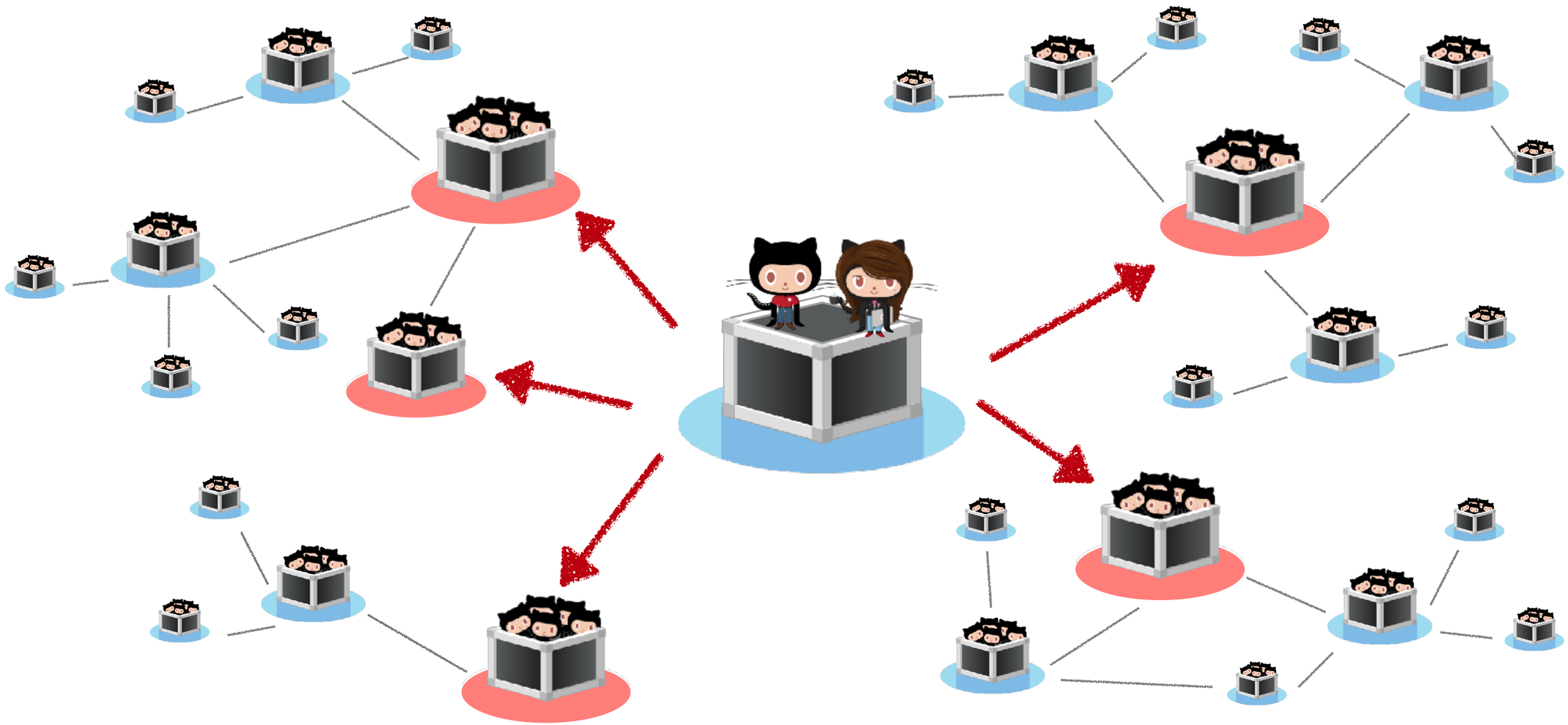
Measure: First, we generate Node2Vec embeddings for each project



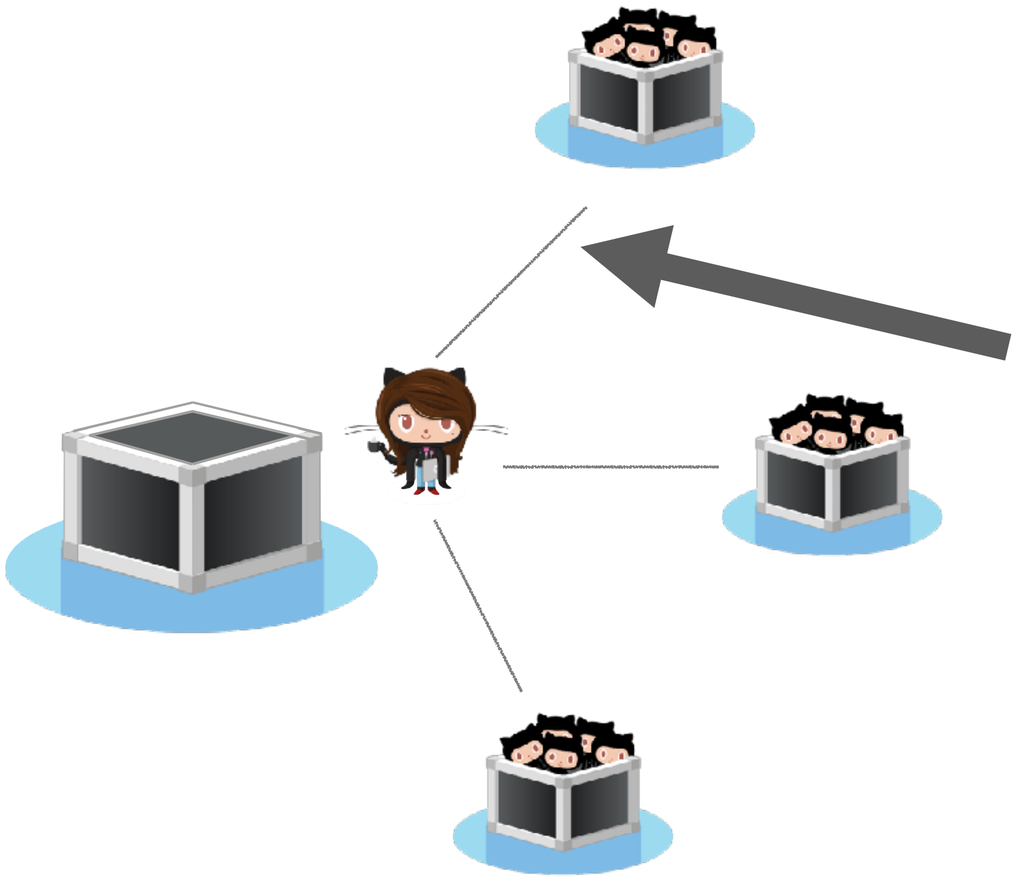
Measure: First, we generate Node2Vec embeddings for each project



Measure: Then, we compute the average pairwise distance (inverse cosine similarity) between a focal project's direct neighbors



From interactions to ties of varying strength



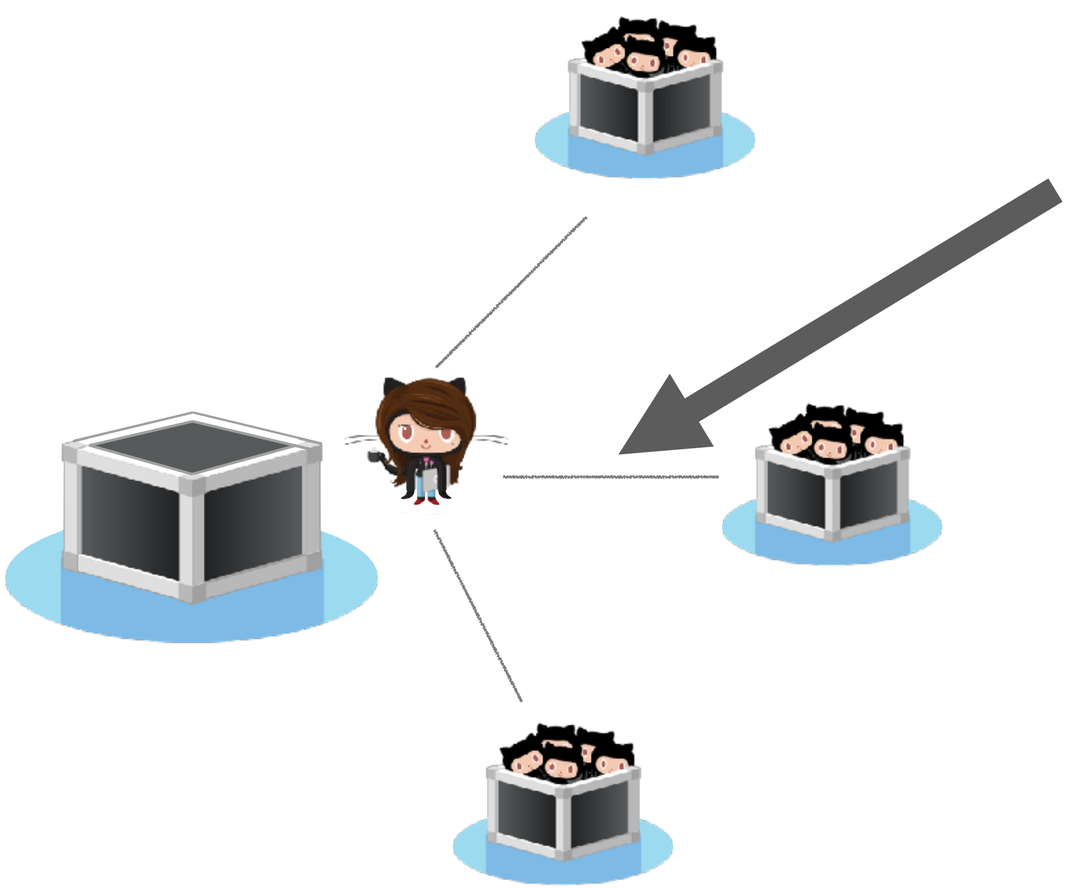
1 file changed +1 -1 lines changed

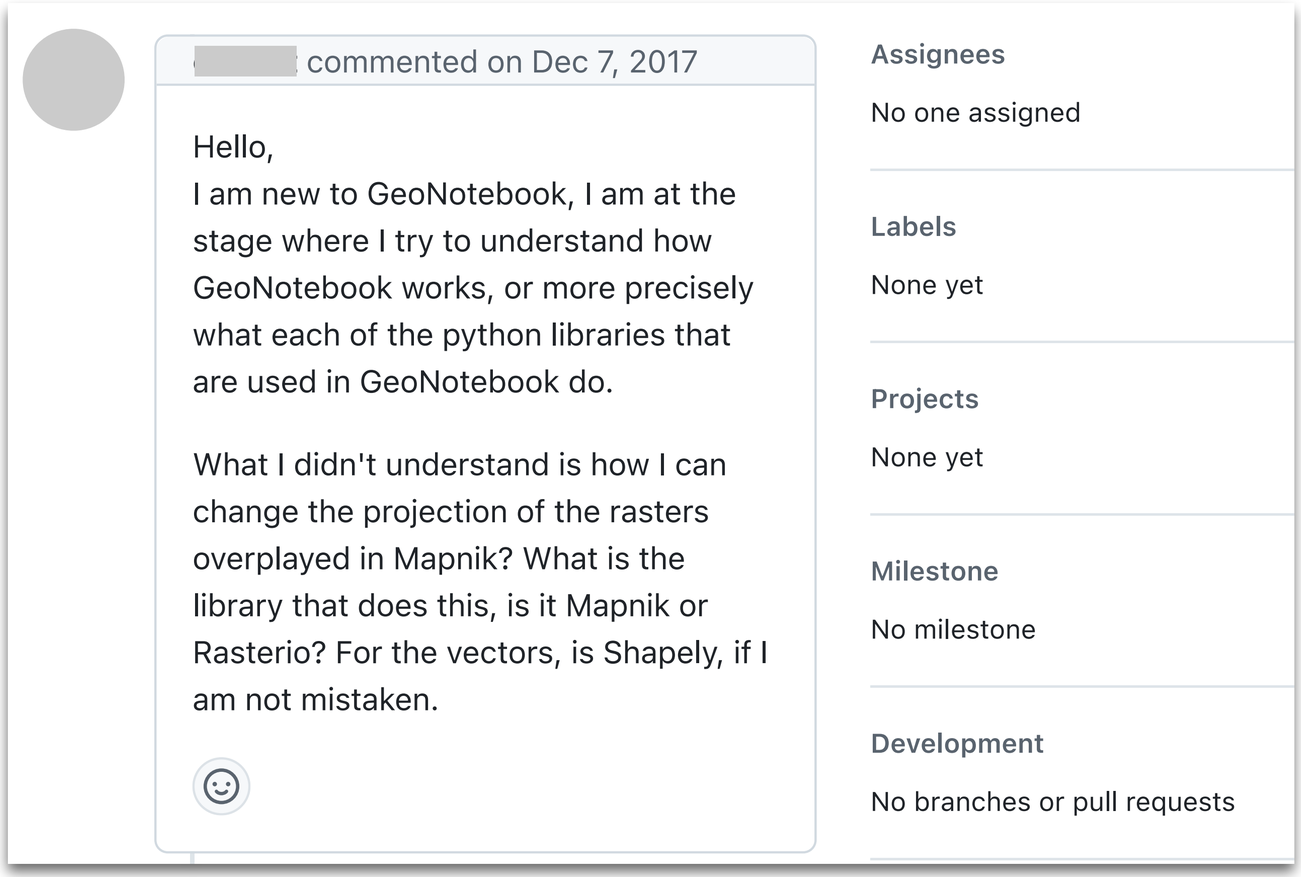
js/config/resolve.js +1 -1

```
@@ -1,6 +1,6 @@
1  var path = require('path');                    1  var path = require('path');
2                                                 2
3 - var renderer =                                3 + var renderer =
    process.env.GEONOTEBOOK_MAP_RENDERE             process.env.GEONOTEBOOK_MAP_RENDERE
    R || 'geojs';                                   R || 'ol';
4                                                 4
5  module.exports = {                             5  module.exports = {
6    alias: {                                     6    alias: {
```

Commits to the codebase
(relatively deep understanding of the codebase)

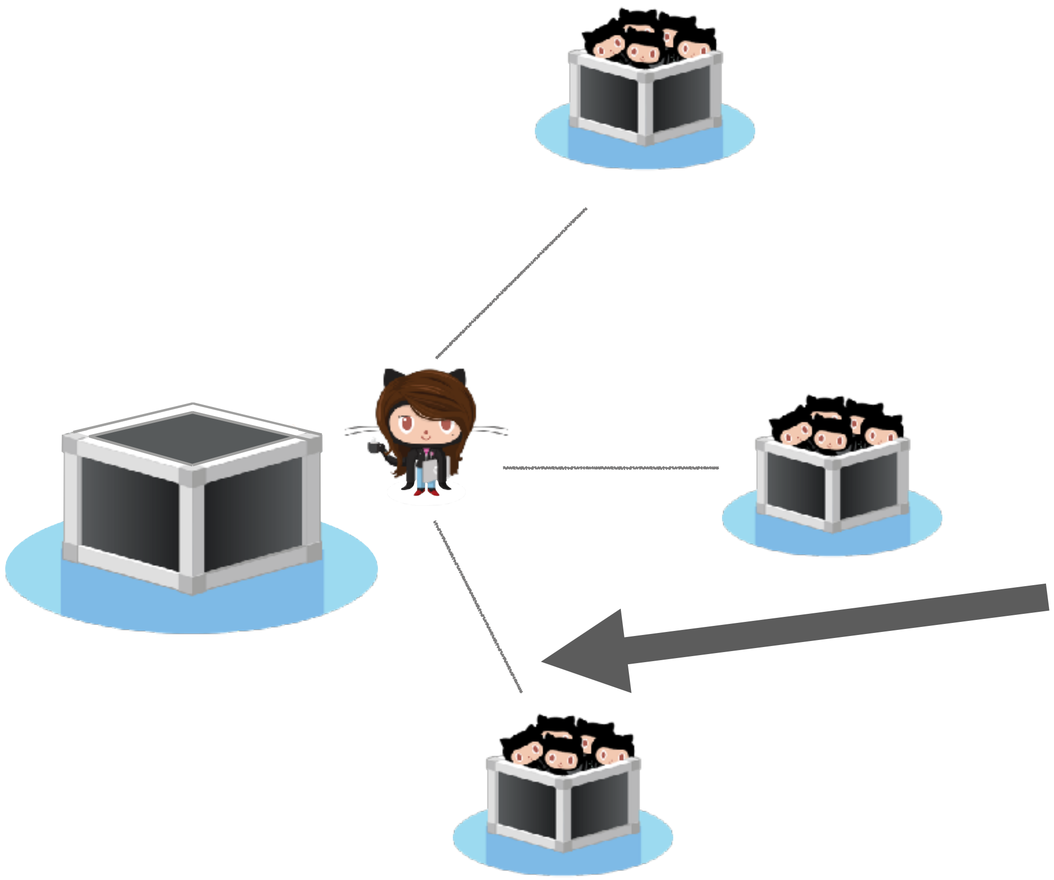
From interactions to ties of varying strength

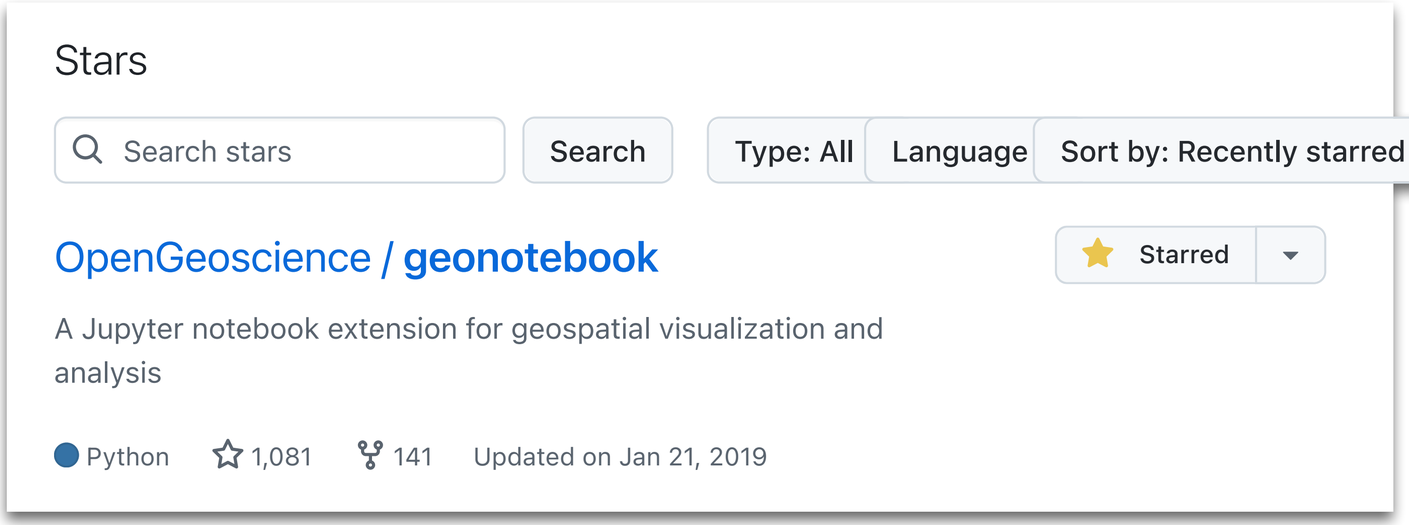




Issue reports
(some understanding of the project)

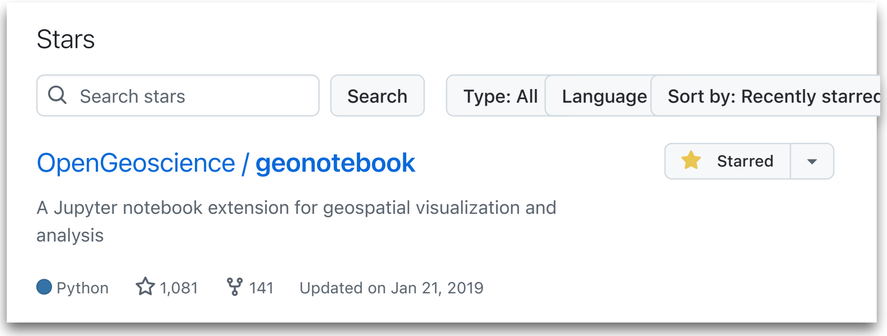
From interactions to ties of varying strength



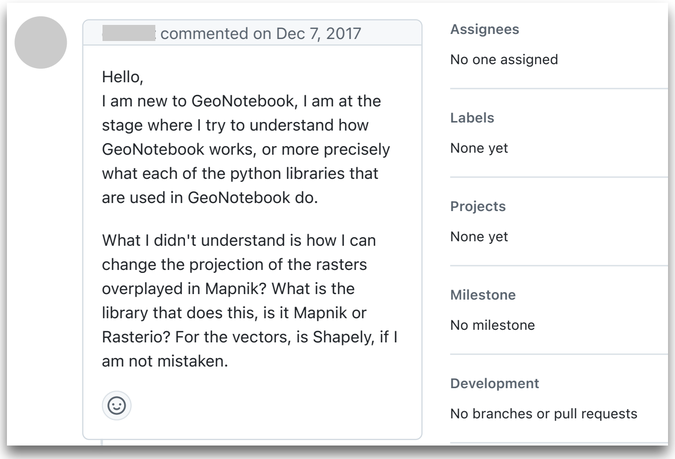


Stars
(awareness of the project)

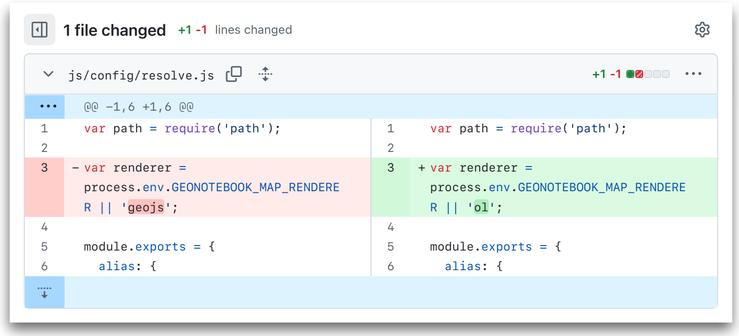
Many interactions are possible, these were just three examples.



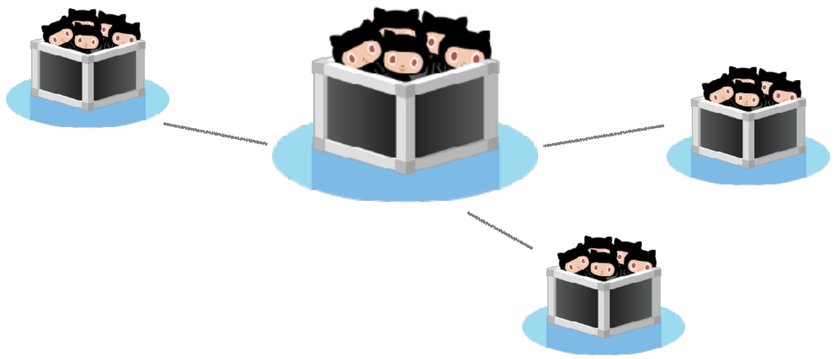
Stars



Issues



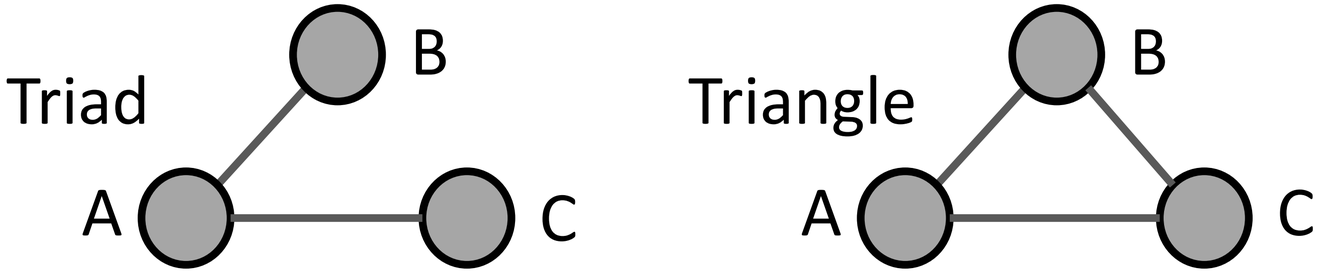
Commits



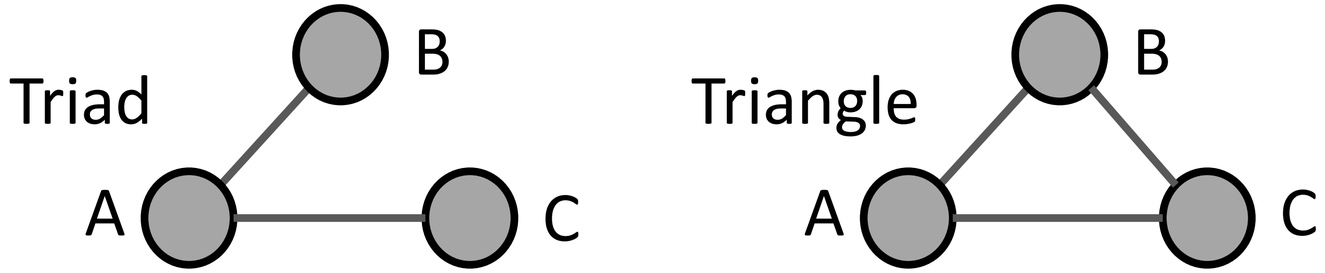
Weaker ties

Stronger ties

In strongly-tied social networks, triads are unlikely.



There is ~an order of magnitude (10×) difference in transitivity values between each pair of networks.

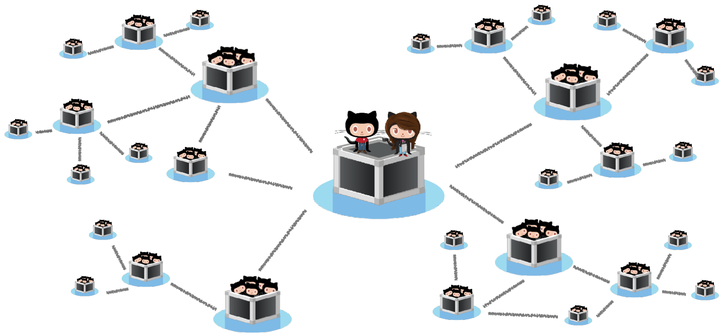


Interaction	**#Nodes**	**#Edges**	**Transitivity** $(\times 10^{-2})$
Commits	$763,062$	$1,926,978$	30.04
Issues	$278,945$	$727,255$	3.42
Stars	$480,394$	$3,658,543$	0.23

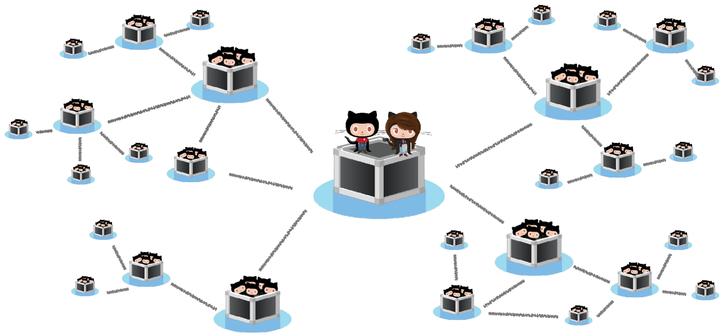
$$\text{Transitivity} = 3 * \text{N}_{\text{triangles}} / \text{N}_{\text{triads}}$$

Commits >> Issues >> Stars

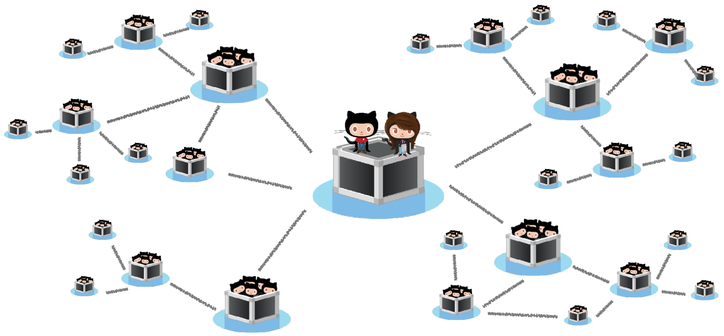
Now what?



Stars

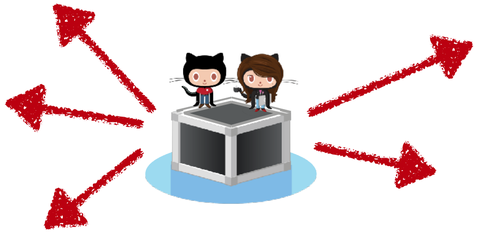


Issues

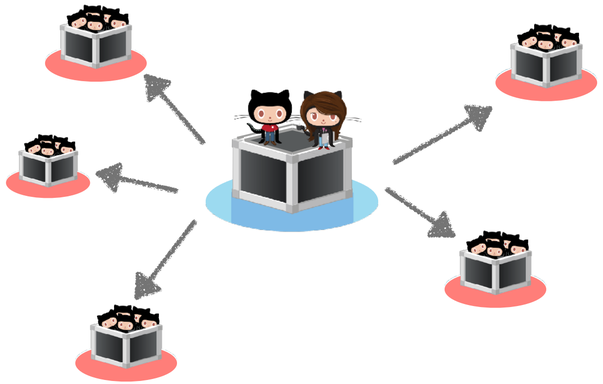


Commits

Out-degree centrality x 3?



Information diversity index x3?



The first two PCs cumulatively explain over 80% of the variance.

	Out-deg. centrality			Diversity index		
	PC1	**PC2**	**PC3**	**PC1**	**PC2**	**PC3**
D_{commit}	0.60	−0.45	0.67	0.63	−0.36	0.69
D_{issue}	0.61	−0.28	−0.74	0.64	−0.24	−0.72
D_{star}	0.52	0.85	0.11	0.43	0.90	0.08

PC1

Average volume of information available

Average diversity of the knowledge space

The first two PCs cumulatively explain over 80% of the variance.

	Out-deg. centrality			Diversity index		
	PC1	**PC2**	**PC3**	**PC1**	**PC2**	**PC3**
D_{commit}	0.60	−0.45	0.67	0.63	−0.36	0.69
D_{issue}	0.61	−0.28	−0.74	0.64	−0.24	−0.72
D_{star}	0.52	0.85	0.11	0.43	0.90	0.08

PC2

Where the connectivity comes from

Where the diversity comes from

(The strength of weak ties)

Hypothesis 3: The more the informational diversity can be attributed to weak ties, the more innovative the projects are.

	Out-deg. centrality			Diversity index		
	PC1	**PC2**	**PC3**	**PC1**	**PC2**	**PC3**
D_{commit}	0.60	−0.45	0.67	0.63	−0.36	0.69
D_{issue}	0.61	−0.28	−0.74	0.64	−0.24	−0.72
D_{star}	0.52	0.85	0.11	0.43	0.90	0.08

Where the diversity comes from

(The strength of weak ties)

Finally, the novelty regression:

- Hypothesis 1 (greater connectivity): weak/inconsistent effects
- Hypothesis 2 (greater info diversity): small but clear effects (25—75 percentile: 4% change in the distribution)
- Hypothesis 3 (strength of weak ties): clear effects, comparable size

	Model III	Model IV
Variables of interest		
Deg_{ave} $(\mathbf{H}_1)$		-0.002^{***}
		(0.001)
$Deg_{weakness}$		-0.005^{***}
		(0.001)
Div_{ave} $(\mathbf{H}_2)$	0.007^{***}	0.008^{***}
	(0.001)	(0.001)
$Div_{weakness}$ $(\mathbf{H}_3)$	0.005^{***}	0.007^{***}
	(0.001)	(0.001)
Observations	38,164	38,164

$^{*}p<0.05$; $^{**}p<0.01$; $^{***}p<0.001$

More details in our preprint

https://arxiv.org/pdf/2411.05646

Weak Ties Explain Open Source Innovation

Hongbo Fang*, Patrick Park*, James Evans†, James Herbsleb*, and Bogdan Vasilescu*

*Carnegie Mellon University, Pittsburgh, USA
fanghongdoublebo@gmail.com, {patpark, jim.herbsleb, vasilescu}@cmu.edu

†University of Chicago, Chicago, USA
jevans@uchicago.edu

***Abstract*—In a real-world social network, weak ties (reflecting low-intensity, infrequent interactions) act as bridges and connect people to different social circles, giving them access to diverse information and opportunities that are not available within one's immediate, close-knit vicinity. Weak ties can be crucial for creativity and innovation, as it introduces new ideas and approaches that people can then combine in novel ways, leading to innovative solutions and creative breakthroughs. Do weak ties facilitate creativity in software in similar ways?**

In this paper, we show that the answer is "yes." Concretely, we study the correlation between developers' knowledge acquisition through three distinct interaction networks on GitHub and the innovativeness of the projects they develop, across over 38,000 Python projects hosted on GitHub. Our findings suggest that the diversity of projects in which developers engage correlates positively with the innovativeness of their future project developments, whereas the volume of interactions exerts minimal influence. Notably, acquiring knowledge through weak interactions (e.g., starring) as opposed to strong ones (e.g., committing) emerges as a stronger predictor of future novelty.

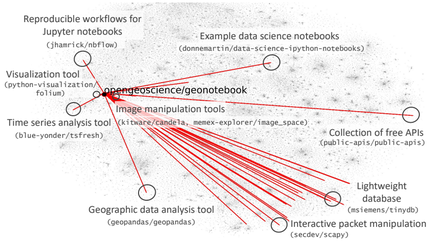


Fig. 1: t-SNE visualization of the embedding space for weak ties (details in Section III-C), depicting all the weak ties of the GeoNotebook Python project in our sample. We highlight some that seem influential for the design of the focal project.

I. Introduction

In the late 1960s and early 1970s, Mark Granovetter, then a PhD student at Harvard University, studied a random sample of men living in a Boston suburb who had recently changed jobs, to understand how they learned about their new job opportunities. His 1973 paper [1], which would go on to become the most cited work in the social sciences, had one key finding: that while a significant majority of people found their jobs through personal contacts rather than through formal channels like advertisements or employment agencies, these contacts were acquaintances (weak ties) rather than close friends or family (strong ties). Granovetter's resulting theory on the "strength of weak ties" explains how weak ties are more effective in job searches—they act as bridges and connect individuals to different social circles, giving access to diverse information and opportunities that are not available within one's immediate, close-knit network. The theory continues to accrue scholarly attention and large-scale confirmation in the digital age [2].

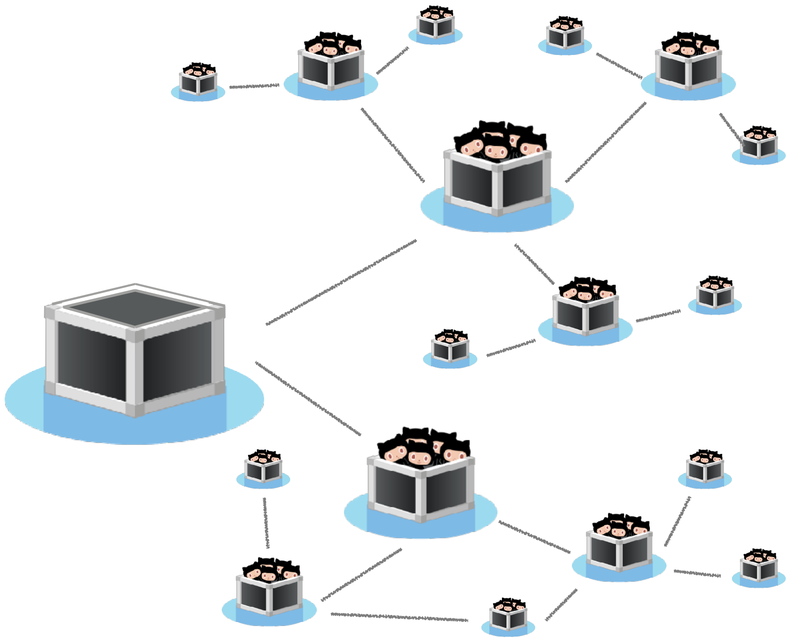
Beyond job searches, the theory had far-reaching implications. For example, weak ties impact, the diffusion of information, social mobility, community organization, and, the focus of this paper, creativity and innovation. The greater diversity of information enabled by weak ties can be crucial for creativity and innovation, as it introduces new ideas and approaches that might not be available within a close-knit group of strong ties, where information is more likely to be redundant. People can then combine these ideas in novel ways, leading to innovative solutions and creative breakthroughs [3]–[5]. The famous "water-cooler effect," that advocates of in-person work environments often quote, relies on the same mechanism—these casual encounters, which are typically with weak ties (work acquaintances rather than close colleagues), can lead to the exchange of diverse ideas and perspectives that spark creativity [6].

Do weak ties facilitate creativity in software in similar ways? More generally, how are structural characteristics of the information networks of software creators associated with the innovativeness of the software they will create? For example, is the software more innovative when it is created by developers with access to more diverse information networks?

In this paper we answer these questions quantitatively, through a large-scale empirical study of over 38,000 open-source Python projects hosted on GitHub. For every project in our sample, we start by reconstructing three project-to-project networks based on interactions (i.e., commits, issues, and GitHub star events) of its core developers with other projects in the past, spanning ties of varying strength. Using these networks we then compute proxies for the amount and diversity of

Main finding: Exposure to diverse ideas through weak ties predicts novel combinations of packages.

- Lurking on the GitHub platform seems to have quantifiable benefits. Redesign the Trending page?
- Automated project recommendation tools may be counterproductive?
- Well-informed but not necessarily highly active developers may also be experts at their craft?
- How to track and give credit to ideas?
- Surface-level vs deep-level diversity?
- AI-generated code: novel or regression to the mean?



Let’s look at some concrete examples of network effects

The emergence of innovation

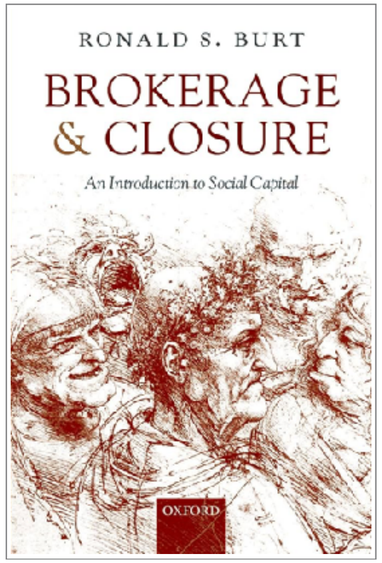
Social capital

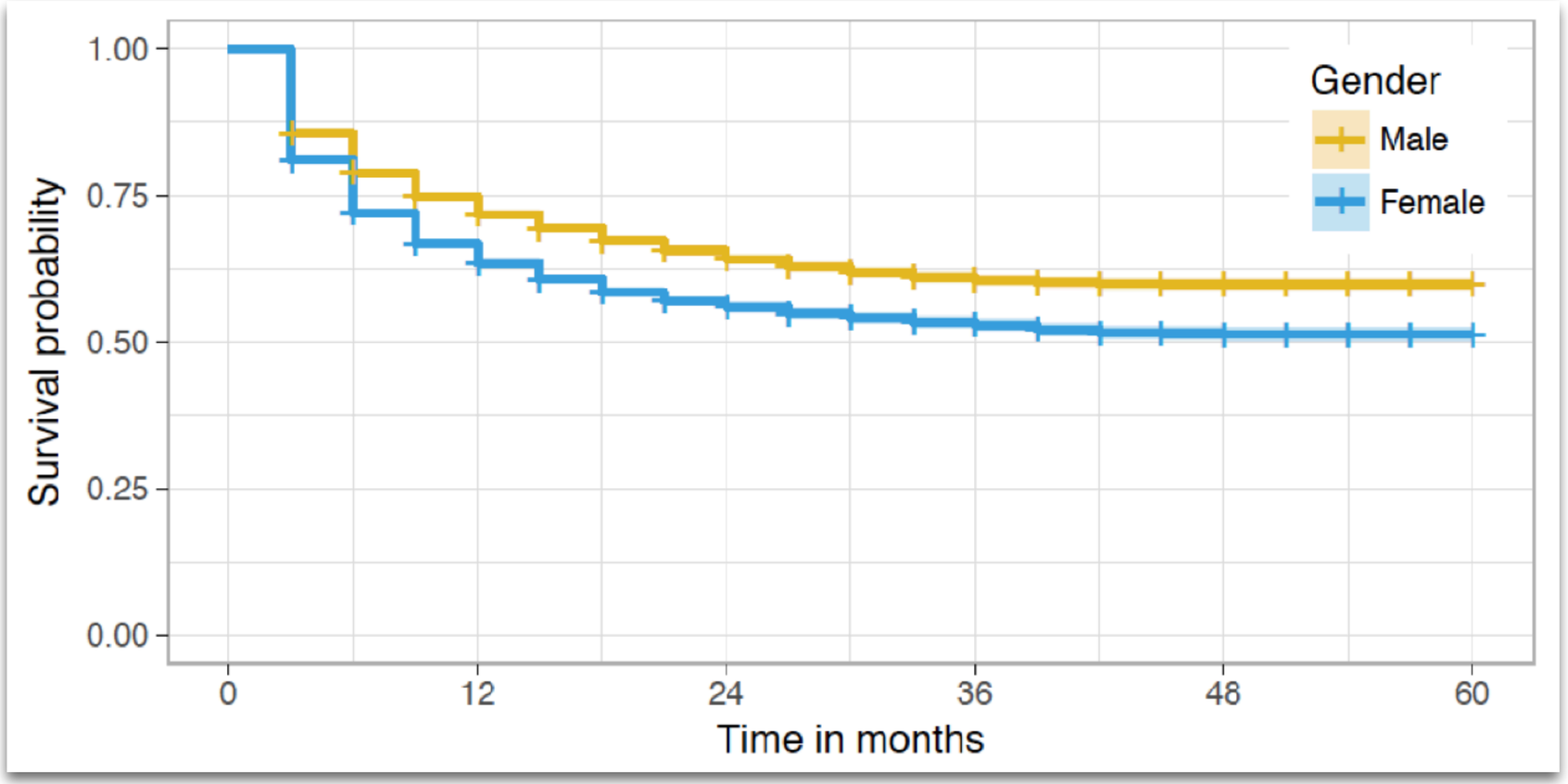
Social contagion

STRUDEL

Weak ties predict longer-term participation for women

Social capital mechanism





Qiu, Nolte, Brown, Serebrenik, and Vasilescu. “Going farther together: The impact of social capital on sustained participation in open source.” ICSE 2019 Distinguished Paper Award.

Weak ties predict the spread of tools

Diffusion of innovations mechanism

12 popular quality assurance tools

Continuous integration	*Dependency management*	*Code coverage reporters*	*Cross browser testers*
build passing	dependencies up to date	coverage 94%	Firefox 82 ✓ Chrome 86 ✓
Travis	David	Coveralls	Saucelabs
Circle	Bithound	Codeclimate	
Appveyor	Gemnasium	Codecov	
Codeship		Codacy	

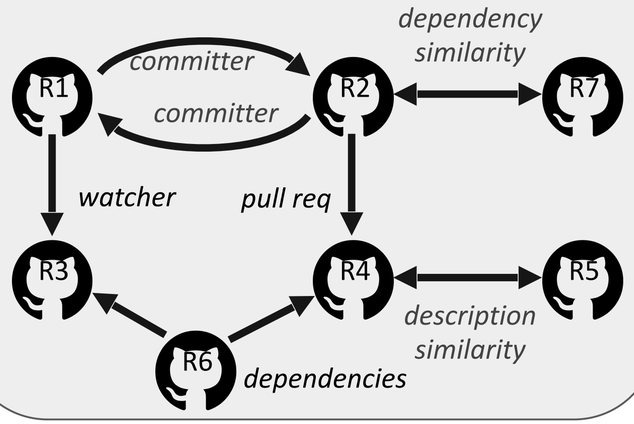
~86,000 npm package repositories

GitHub

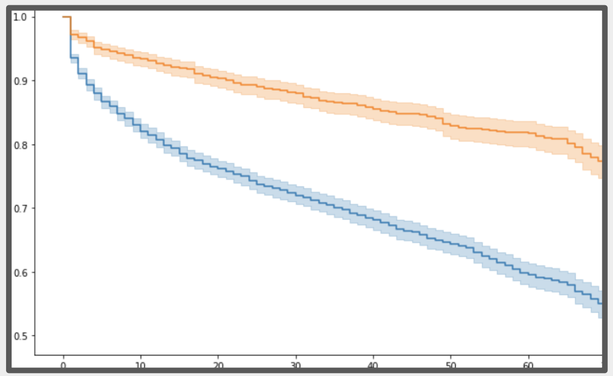
npm

For each tool:

Heterogeneous network



Hazard modeling (Cox regression)



Lamba, Trockman, Armanios, Kästner, Miller, and Vasilescu. “Heard it through the Gitvine: An empirical study of tool diffusion across the npm ecosystem. FSE 2020.

Acknowledgements


Courtney Miller


Anita Brown


Asher Trockman


Jim Herbsleb


Shurui Zhou

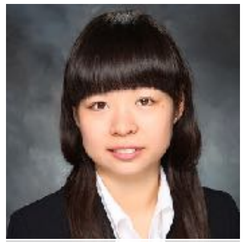

Hongbo Fang


Anita Sarma


Cassandra Overney


Audris Mockus


Alex Nolte


Sophie Qiu

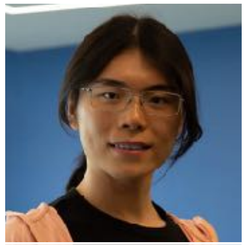

Alex Serebrenik


Marat Valiev


Laura Dabbish


Lily Li


Naveen Raman


Hao He


Christian Kästner


Hemank Lamba


Emerson Murphy-Hill

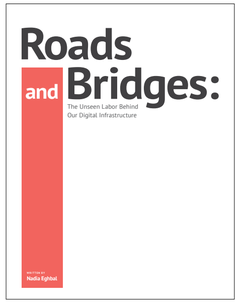
Google

Alfred P. Sloan FOUNDATION

National Science Foundation

FORDFOUNDATION

Like any infrastructure, it needs regular upkeep and maintenance

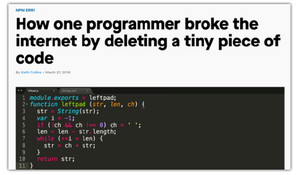


Everybody uses open source:

- Fortune 500 companies
- Major software companies
- Startups
- Government
- …

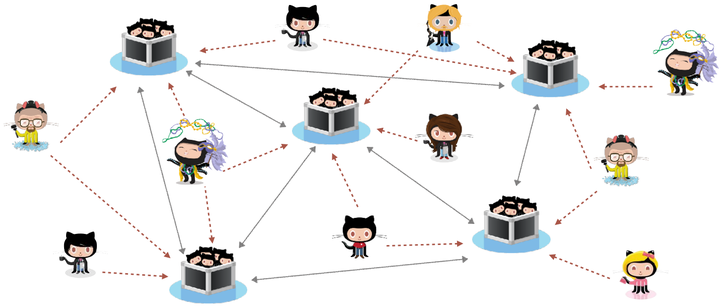
If undermaintained:

- Brittle supply chains
- Risks for downstream users
- Slows down innovation
- …



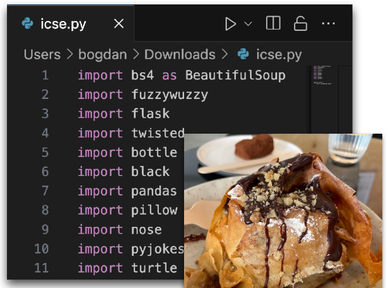
https://qz.com/646467/how-one-programmer-broke-the-internet-by-deleting-a-tiny-piece-of-code/

Contributors and projects form complex socio-technical networks!



Can we measure the network effects?

Software innovation as novel recombination of software libraries



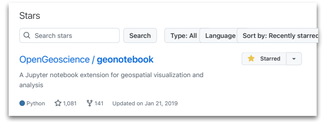
Lots of combinations:

- **(twisted, bottle)**
- **(turtle, nose)**
- **(black, pandas)**
- **(fuzzywuzzy, pillow)**
- …

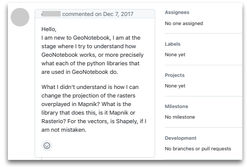
C(n,2) unique pairs of packages.

Dark chocolate + apple strudel is arguably innovative because it is atypical.

Exposure to diverse ideas through weak ties predicts novel combinations of packages.

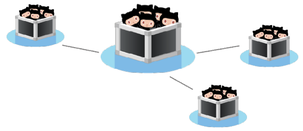


Stars



Issues

Commits



Weaker ties

Stronger ties

DALL·E 3 - “Networks of open-source software projects”

The Strength of Weak Ties in Open-Source Software Development Networks

Bogdan Vasilescu
At U Zurich, January 9th, 2025

STRUDEL
SOCIO-TECHNICAL RESEARCH USING DATA EXCAVATION LAB

Carnegie Mellon University

S3D
Software and Societal Systems Department

DALL·E 3 - “Networks of open-source software projects”

COUNT_SCRATCHULA

The Strength of Weak Ties in Open-Source Software Development Networks

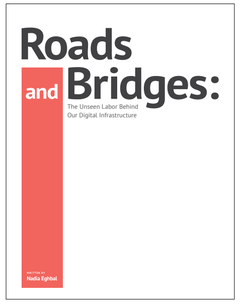
Bogdan Vasilescu
At U Zurich, January 9th, 2025

STRUDEL
SOCIO-TECHNICAL RESEARCH USING DATA EXCAVATION LAB

Carnegie Mellon University

S3D
Software and Societal Systems Department

Like any infrastructure, it needs regular upkeep and maintenance

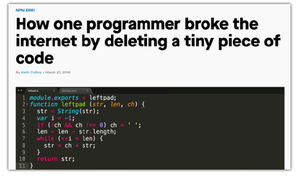


Everybody uses open source:

- Fortune 500 companies
- Major software companies
- Startups
- Government
- …

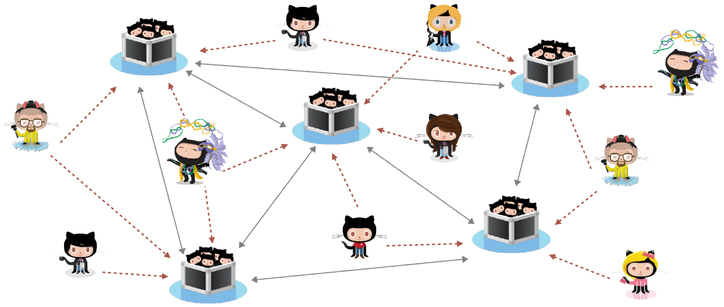
If undermaintained:

- Brittle supply chains
- Risks for downstream users
- Slows down innovation
- …



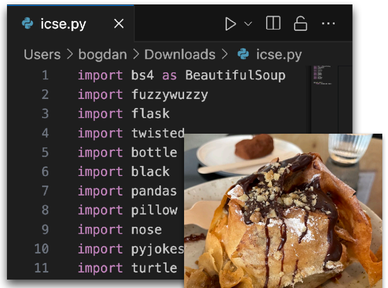
https://qz.com/646467/how-one-programmer-broke-the-internet-by-deleting-a-tiny-piece-of-code/

Contributors and projects form complex socio-technical networks!



Can we measure the network effects?

Software innovation as novel recombination of software libraries



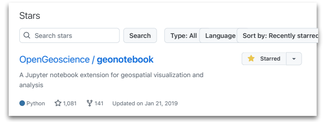
Lots of combinations:

- **(twisted, bottle)**
- **(turtle, nose)**
- **(black, pandas)**
- **(fuzzywuzzy, pillow)**
- …

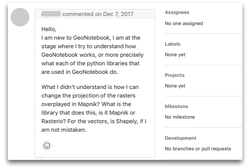
C(n,2) unique pairs of packages.

Dark chocolate + apple strudel is arguably innovative because it is atypical.

Exposure to diverse ideas through weak ties predicts novel combinations of packages.

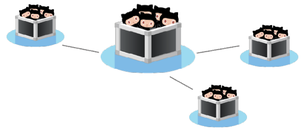


Stars



Issues

Commits



Weaker ties

Stronger ties

DALL·E 3 - “Networks of open-source software projects”

The Strength of Weak Ties in Open-Source Software Development Networks

Bogdan Vasilescu
At U Zurich, January 9th, 2025

STRUDEL
SOCIO-TECHNICAL RESEARCH USING DATA EXCAVATION LAB

Carnegie Mellon University

S3D
Software and Societal Systems Department