Team Discussions and Dynamics During DevOps Tool Adoptions in OSS Projects

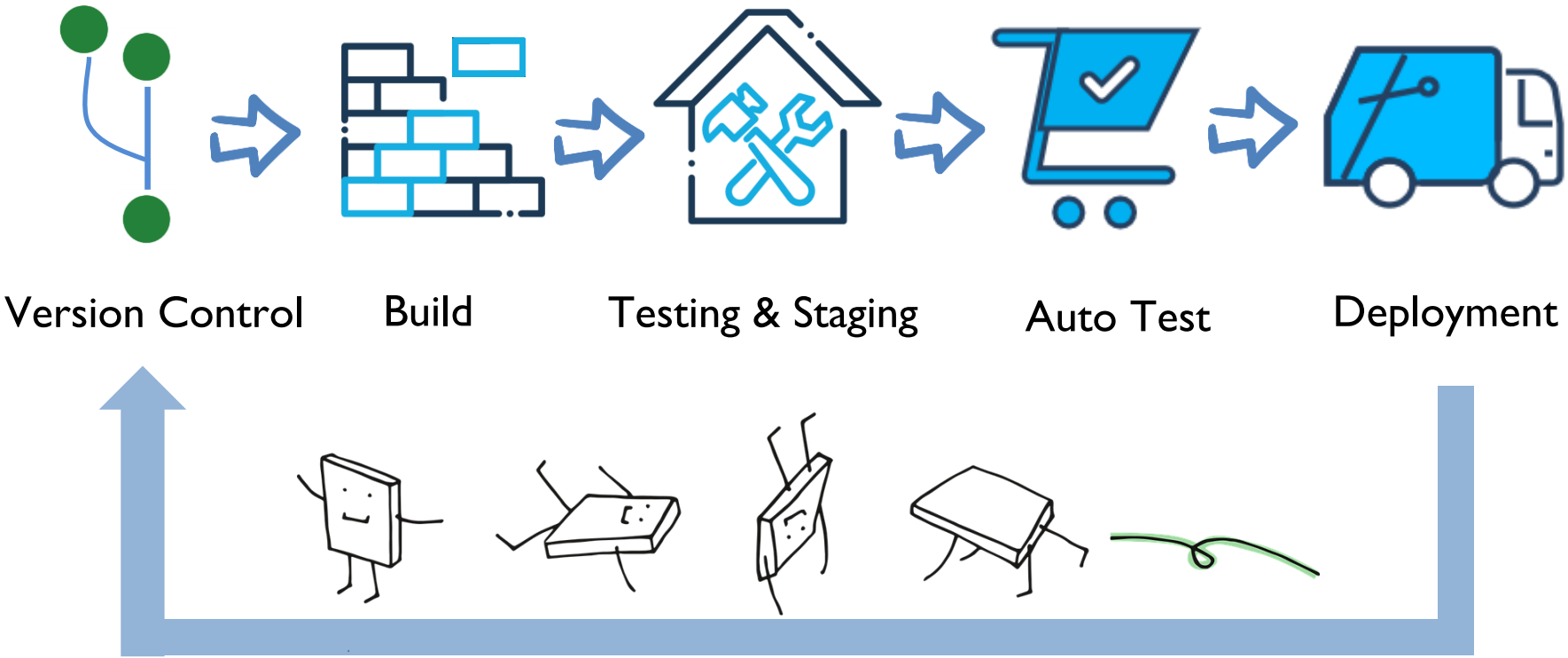
Likang Yin
DECAL and CS Department
University of California, Davis
lkyin@ucdavis.edu

Vladimir Filkov
DECAL and CS Department
University of California, Davis
vfilkov@ucdavis.edu

UCDAVIS
UNIVERSITY OF CALIFORNIA

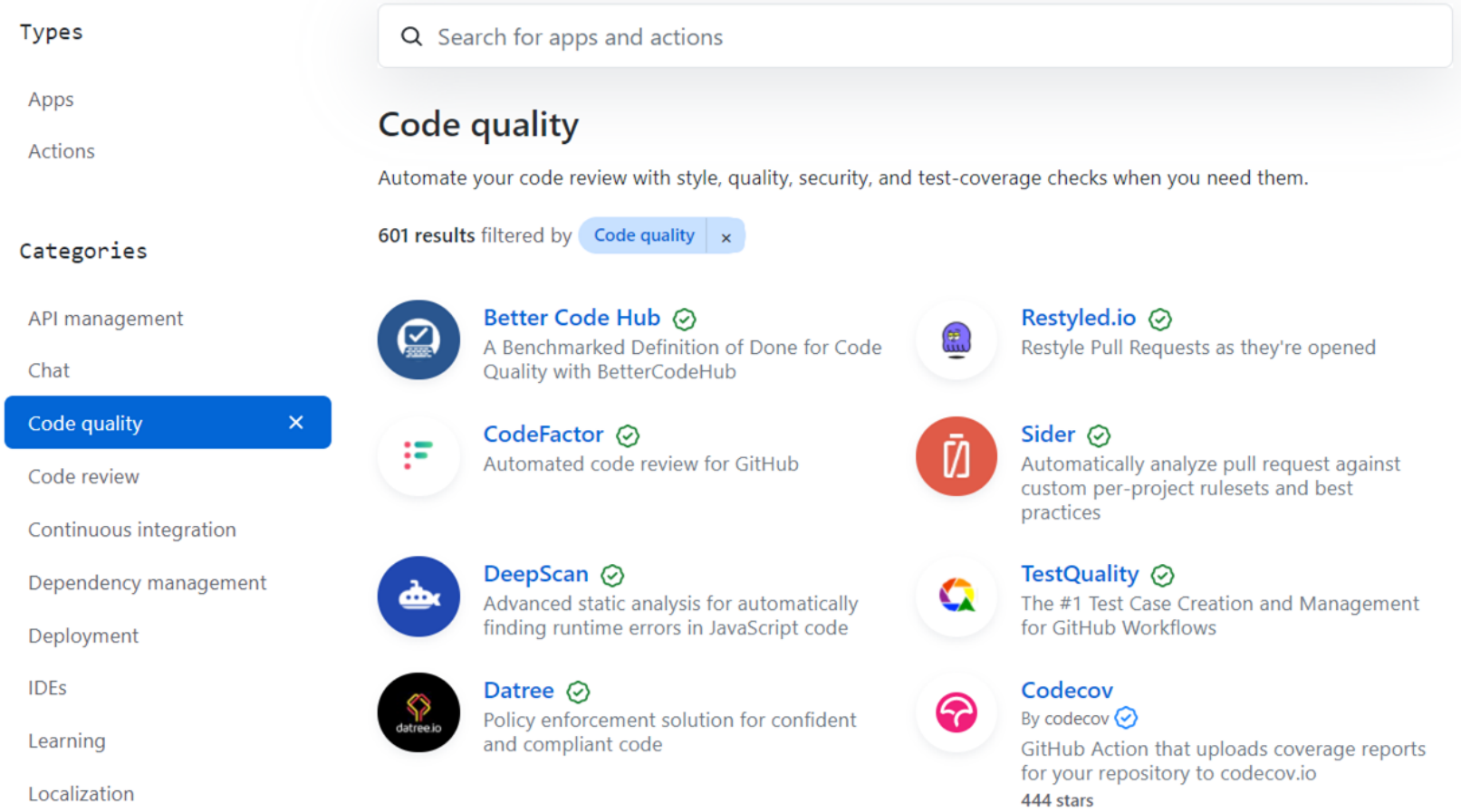
NSF

Continuous Integration Pipeline



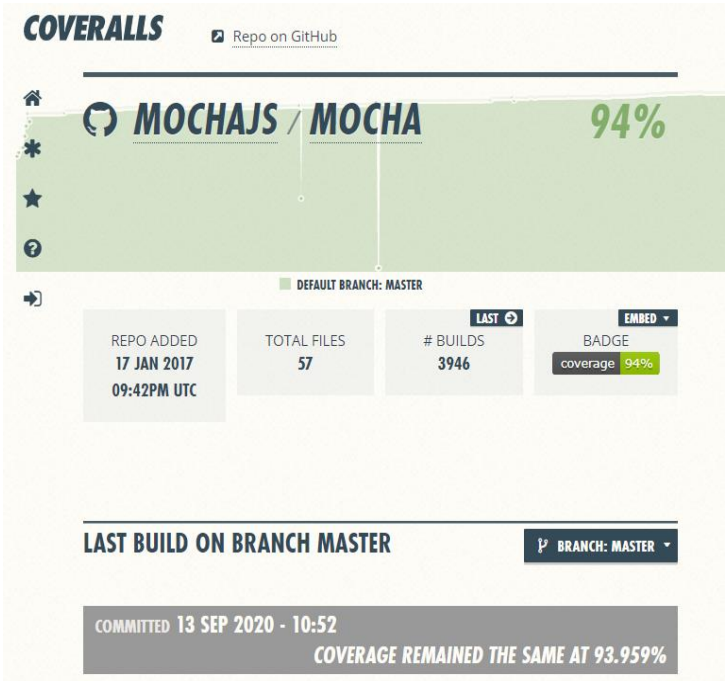
The pixel cell is from here: https://pixabay.com/users/manfredsteger-1848497/

Pipelines Use Tools, and There are Many

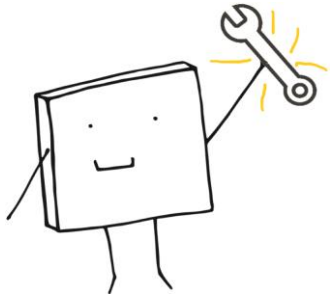


Badges Show The Tools Being Used

```
<p align="center"><a href="http://travis-ci.org/mochajs/mocha"><img
src="https://api.travis-ci.org/mochajs/mocha.svg?branch=master" alt="Build Status"></a>
<a href="https://coveralls.io/github/mochajs/mocha"><img
src="https://coveralls.io/repos/github/mochajs/mocha/badge.svg" alt="Coverage Status">
</a>  <a
```



Tool Adoptions in GitHub

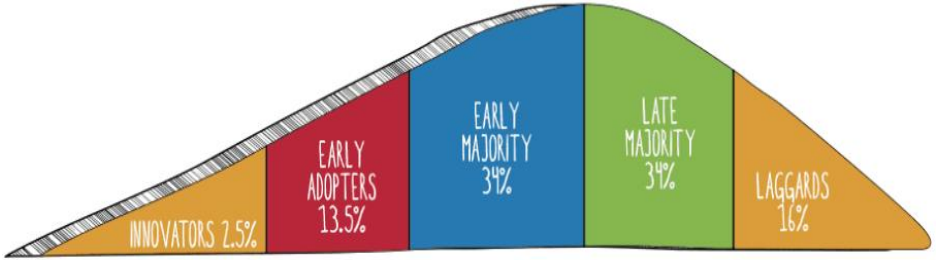


- Project teams adopt tools all the time.
- Many GitHub projects, especially the small and individual projects, do not use any tools.
- Adoptions are rarely preceded by meaningful discussions.
- Adopting a new tool can affect everyone in the team as they are all stakeholders, making adopting tools a complex process.

There exist issues in the adoption process, and we need to improve it.

What Drives Adoptions in Teams?

DIFFUSION OF INNOVATION MODEL

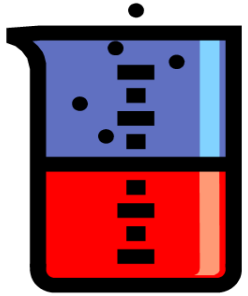


Diffusion as the process by which an innovation unfolds over time through communication channels among the population.

https://www.smartinsights.com/marketing-planning/marketing-models/diffusion-innovation-model/

Social Judgement Theory

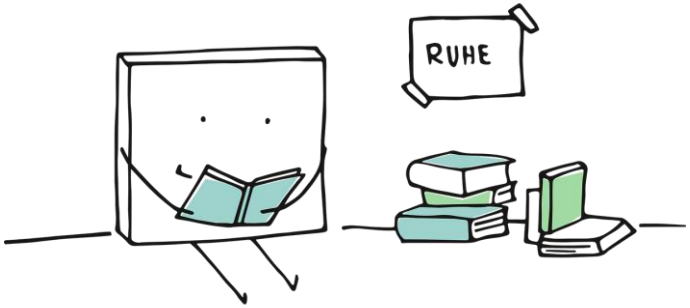
- Sherif & Sherif



When people are exposed to new ideas, they tend to consider their previously formed attitude, or anchor to which they compare the new idea. Therefore, people can react very differently.

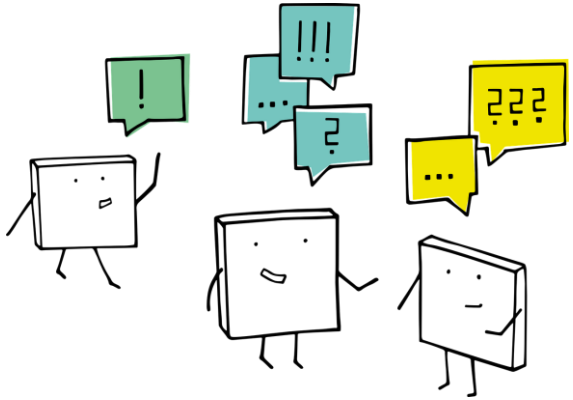
https://www.powershow.com/viewht/b7e63-ZDc1Z/Social_Judgement_Theory_powerpoint_ppt_presentation

Communication Means During Tool Adoptions

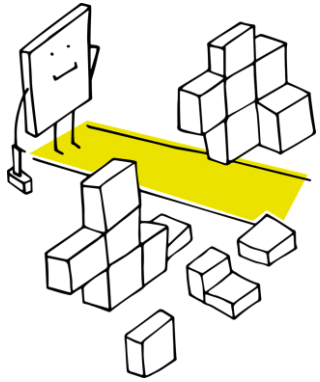


(a) Social and traditional media

(b) Work/group discussions/learning



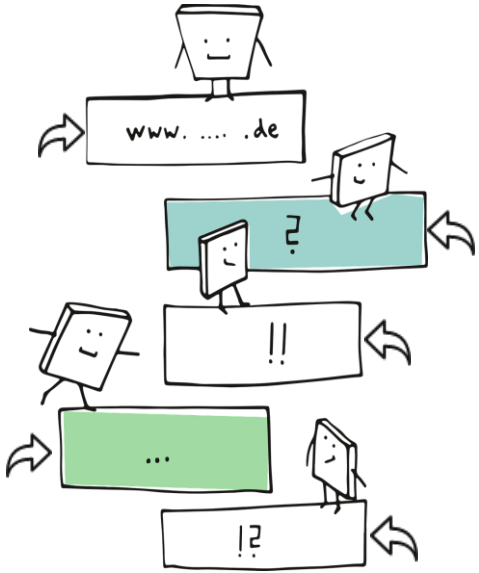
(c) One-on-one communication



(d) Other channels

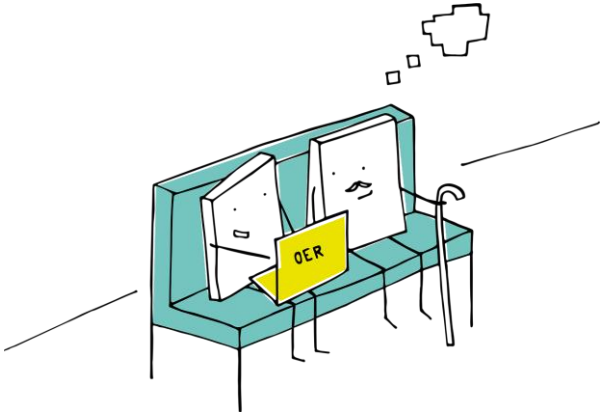
In OSS Projects We Can Track the Following

Team Discussions

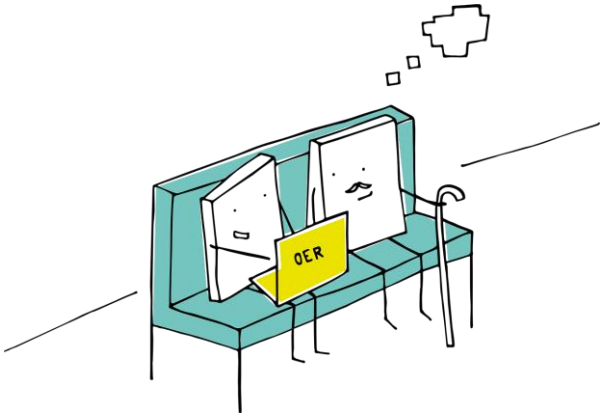


Prior Tool Exposure

Tenure Time



Experience and Tool Exposure



(a) New Devs vs. Senior Devs

(b) Devs w/ exposure v.s. Devs w/o exposure to tool

(c) Influencers

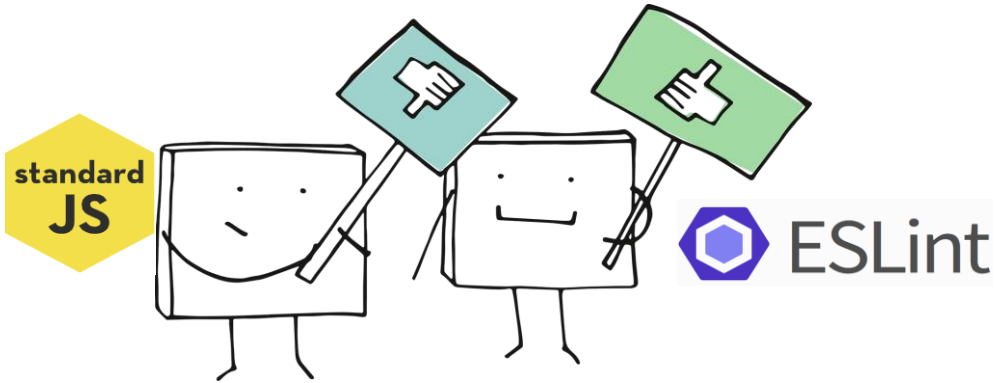
Research Questions

RQ1: What is the social dynamics during tool adoptions?

RQ2: How are people's sentiments evolving toward the tools?

RQ3: Are tool adoption events associated with influencers?

RQ4: What are the determinants of tool adoptions?



Research Approach

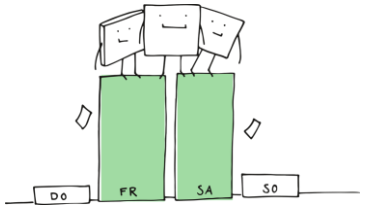
1. **Data collection:** 52,923 distinct GitHub projects with 96,176 tool adoptions.

2. **Data cleaning:** Only keep comments toward tools and remove emojis, URLs, code from comments.

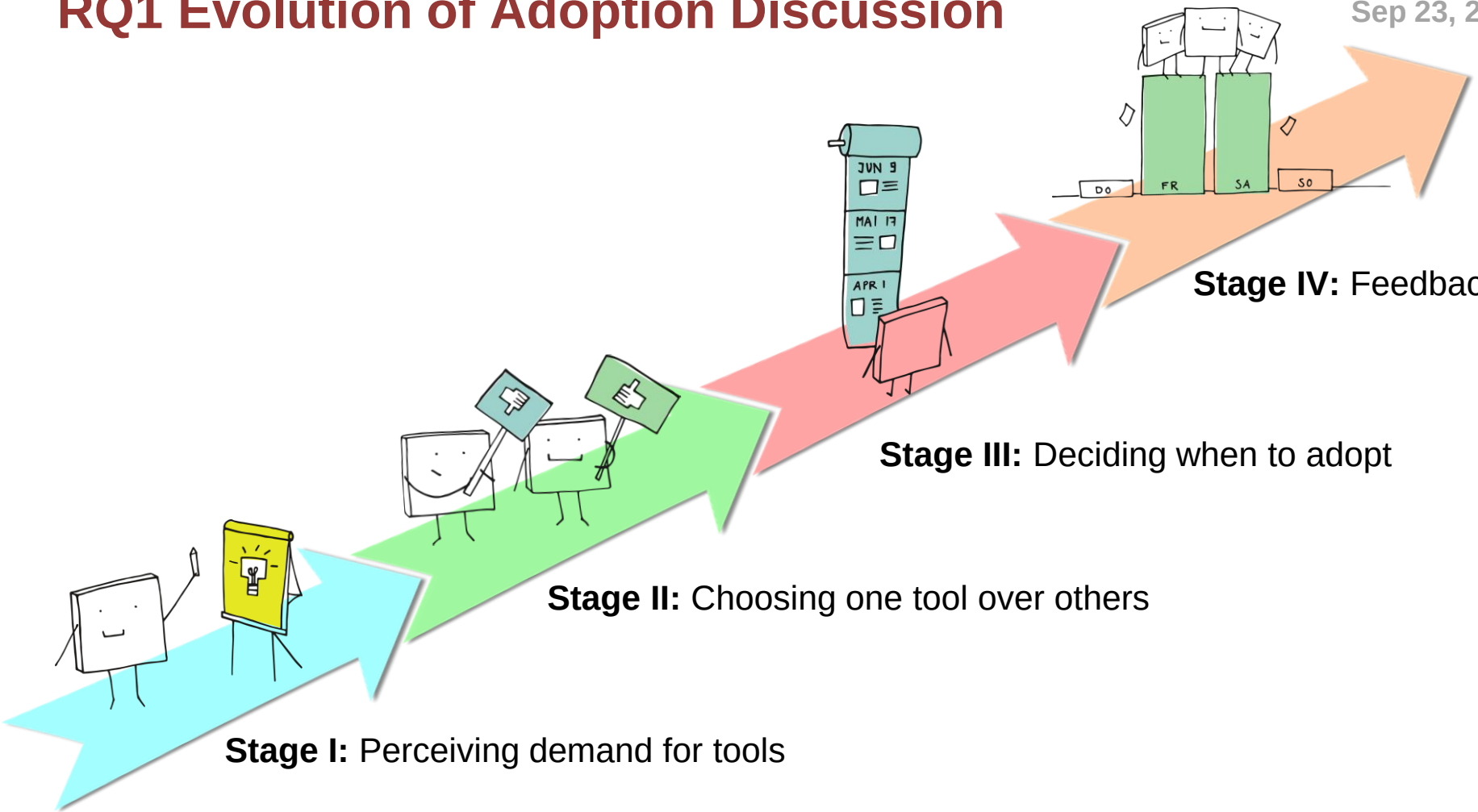
3. **Sentiment Analysis:** Senti4SD is used to label the sentiment of comments.

4. **Case Study:** Discussion evolution and influencers.

5. **Modeling:** Generalized Linear mixed-effects (hierarchical) models (GLMMs).



RQ1 Evolution of Adoption Discussion



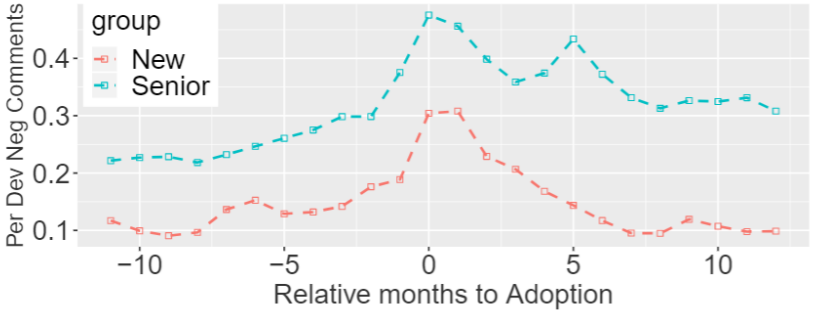
Stage I: Perceiving demand for tools

Stage II: Choosing one tool over others

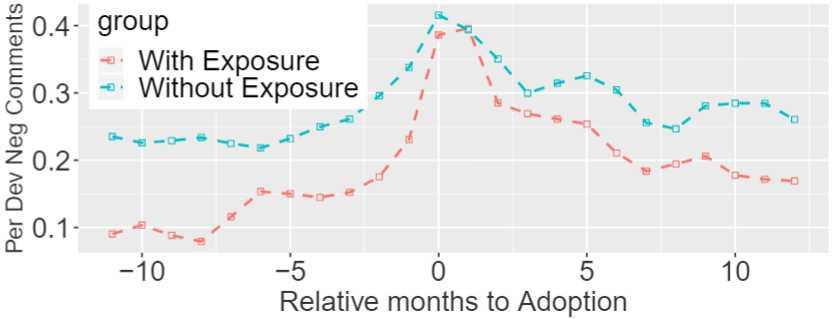
Stage III: Deciding when to adopt

Stage IV: Feedback

RQ2 Sentiment Dynamics During Tool Adoptions

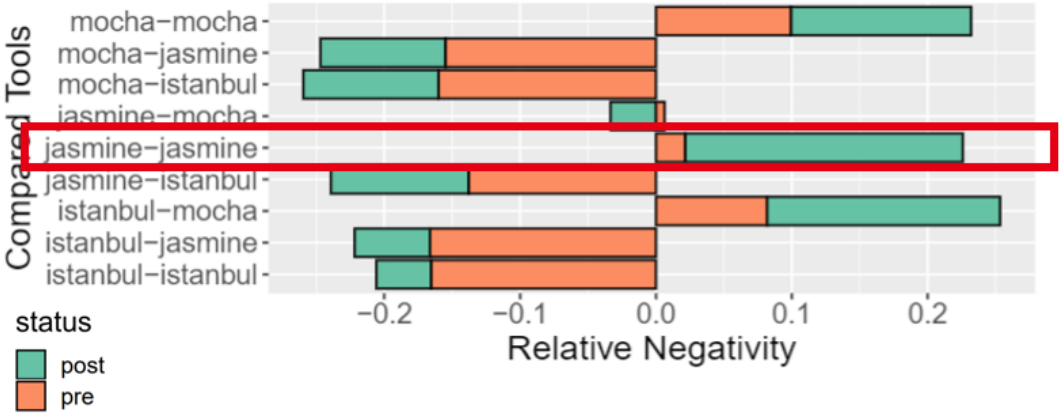


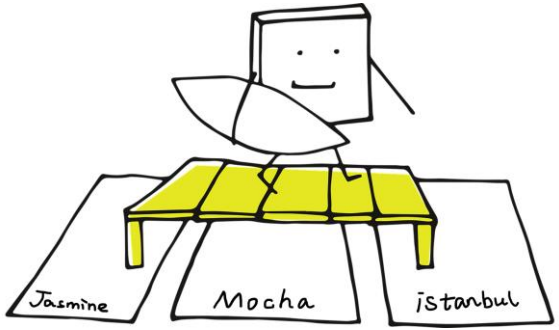
New developers, overall, seem to have lower negativity compared to the sentiment of senior developers.



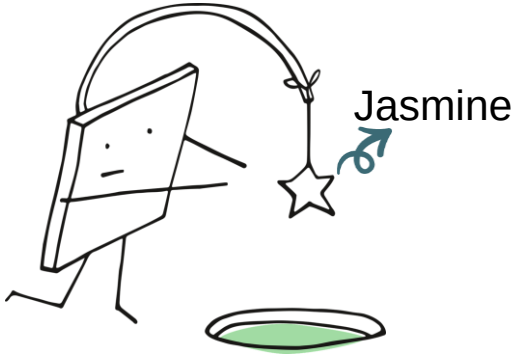
Developers with prior exposure seem to drop the negativity faster and remain a relatively lower negativity to the developers without exposure.

RQ2 Some tools are, different than expected...





After adopting one the alternative tools, the sentiment of that tool changes, which can become even more negative.



RQ3 Case Study on Influencers

With Strong Influencers

With a strong influencer In the project *testem/testem*, the following discussion transpired, on using a tool to automate testing, where <Dev 2> is the strong influencer in the team.

<Dev 1>: "... I think it would be convenient to render the page as template and pass there..."

<Dev 2> "Why do you need this? Please give more details about your use case."

<Dev 1> "... I use testem not just to run unit tests to see standard test report page, but as a watching tool that automatically reloads my web application when sources changed ..."

<Dev 2> "I get most of what you are saying. Are you using <tool 1>? The hash issue I think I need to rethink how to handle that ..."

<Dev 1> "Now I switched to <tool 2> (as it really more robust and convinient), I used <tool 1> as well it doesn't actually matter. And I use dependency management tool to load scripts ..."

Without Strong Influencers

Without a strong influencer In contrast, the project *bower/bower* encountered an issue when trying to use a tool for automated testing. A developer asked for help from the community, however, no one presented strong opinions in favor of continued use of this tool. One member suggested to not use <tool> anymore and switch to another tool.

<Dev 1>: "Should we have a common way to declare the tests for any component? For example, I'd specify <tool> and file in <file>. I'd then run <command>, which would open the <tool> page in a browser. Thoughts?"

<Dev 2>: "what we could support is something similar to <package name> which is common scripts specified in the <file> ... what do <Dev 3> and <Dev 4> think?"

<Dev 4>: "<Dev 5> has said it was a mistake to make them all globals. they should be triggered with <file> ..."

<Dev 6>: "... Any component with unit tests that I've written just ends up using a separate node module (like *<tool>*) to automate the test workflow. I'd be in favor of closing this."

RQ4 Data Modelling

Mining

sample projects

Responses

Tool Adoption

Discussion Length

Team-level Metrics

New

Involved

w/ Exposure

Positive

Negative

Controls

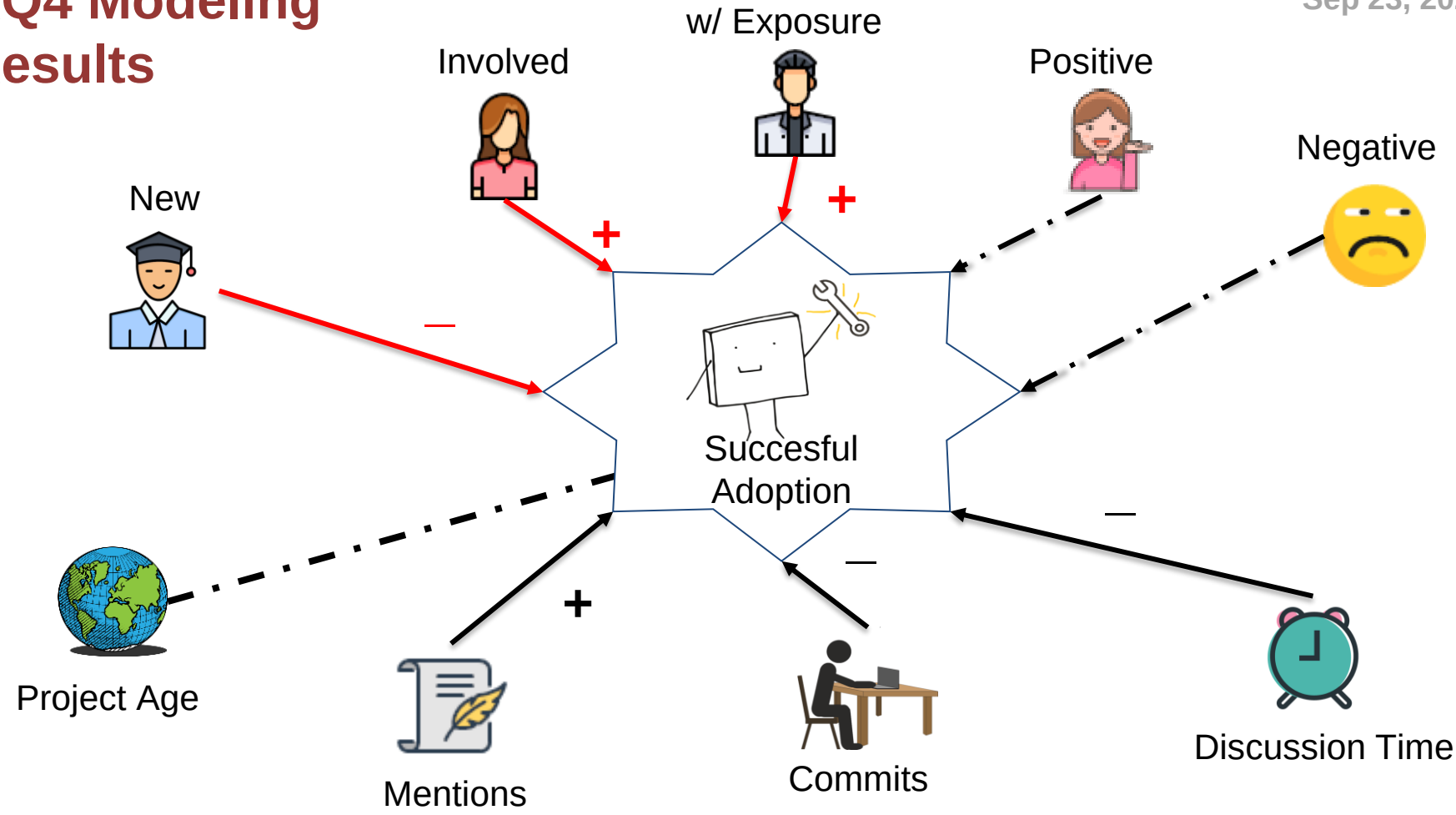
Project Age

Mentions

Commits

Tool

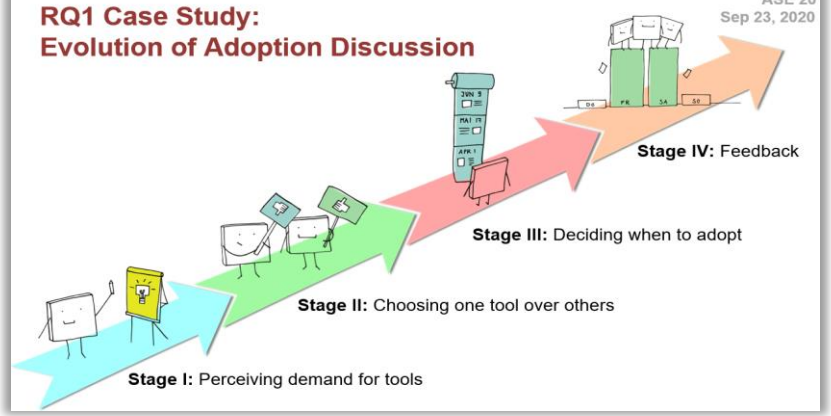
RQ4 Modeling Results



The Takeaways

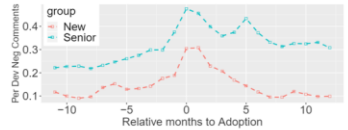
RQ1 Case Study: Evolution of Adoption Discussion

ASE'20
Sep 23, 2020

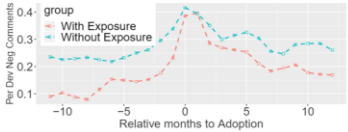


RQ2: Sentiment Dynamics During Tool Adoptions

ASE'20
Sep 23, 2020



New developers, overall, seem to have lower negativity compared to the sentiment of senior developers.



Developers with prior exposure seem to drop the negativity faster and remain a relatively lower negativity to the developers without exposure.

RQ3 Case Study on Influencers

ASE'20
Sep 23, 2020

With Strong Influencers

With a strong influencer In the project *testem/testem*, the following discussion transpired, on using a tool to automate testing, where <Dev 2> is the strong influencer in the team.

<Dev 1>: "... I think it would be convenient to render the page as template and pass there..."

<Dev 2> "Why do you need this? Please give more details about your use case."

<Dev 1> "... I use testem not just to run unit tests to see standard test report page, but as a watching tool that automatically reloads my web application when sources changed ..."

<Dev 2> "I get most of what you are saying. Are you using <tool 1>? The hash issue I think I need to rethink how to handle that ..."

<Dev 1> "Now I switched to <tool 2> (as it really more robust and convinient), I used <tool 1> as well it doesn't actually matter. And I use dependency management tool to load scripts ..."

Without Strong Influencers

Without a strong influencer In contrast, the project *bower/bower* encountered an issue when trying to use a tool for automated testing. A developer asked for help from the community, however, no one presented strong opinions in favor of continued use of this tool. One member suggested to not use <tool> anymore and switch to another tool.

<Dev 1>: "Should we have a common way to declare the tests for any component? For example, I'd specify <tool> and file in <file>. I'd then run <command>, which would open the <tool> page in a browser. Thoughts?"

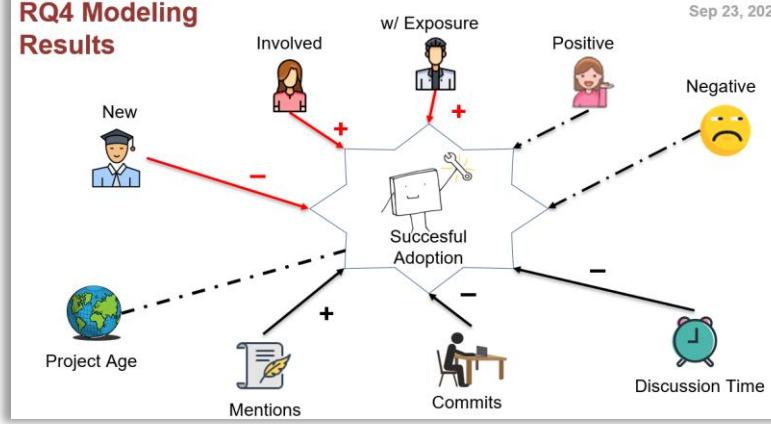
<Dev 2>: "what we could support is something similar to <package name> which is common scripts specified in the <file> ... what do <Dev 3> and <Dev 4> think?"

<Dev 4>: "<Dev 5> has said it was a mistake to make them all globals. they should be triggered with <file> ..."

<Dev 6>: "... Any component with unit tests that I've written just ends up using a separate node module (like *<tool>*) to automate the test workflow. I'd be in favor of closing this."

RQ4 Modeling Results

ASE'20
Sep 23, 2020



Thank you!