

# The GitHub Graph: Characterizing Open-Source Collaboration Patterns

Emilie Ma

(<https://emilie.ma> + hello@emilie.ma)

*In collaboration with Professors Cleidson de Souza (Universidade Federal do Pará), Dongwook Yoon (UBC), & Ivan Beschastnikh (UBC)*

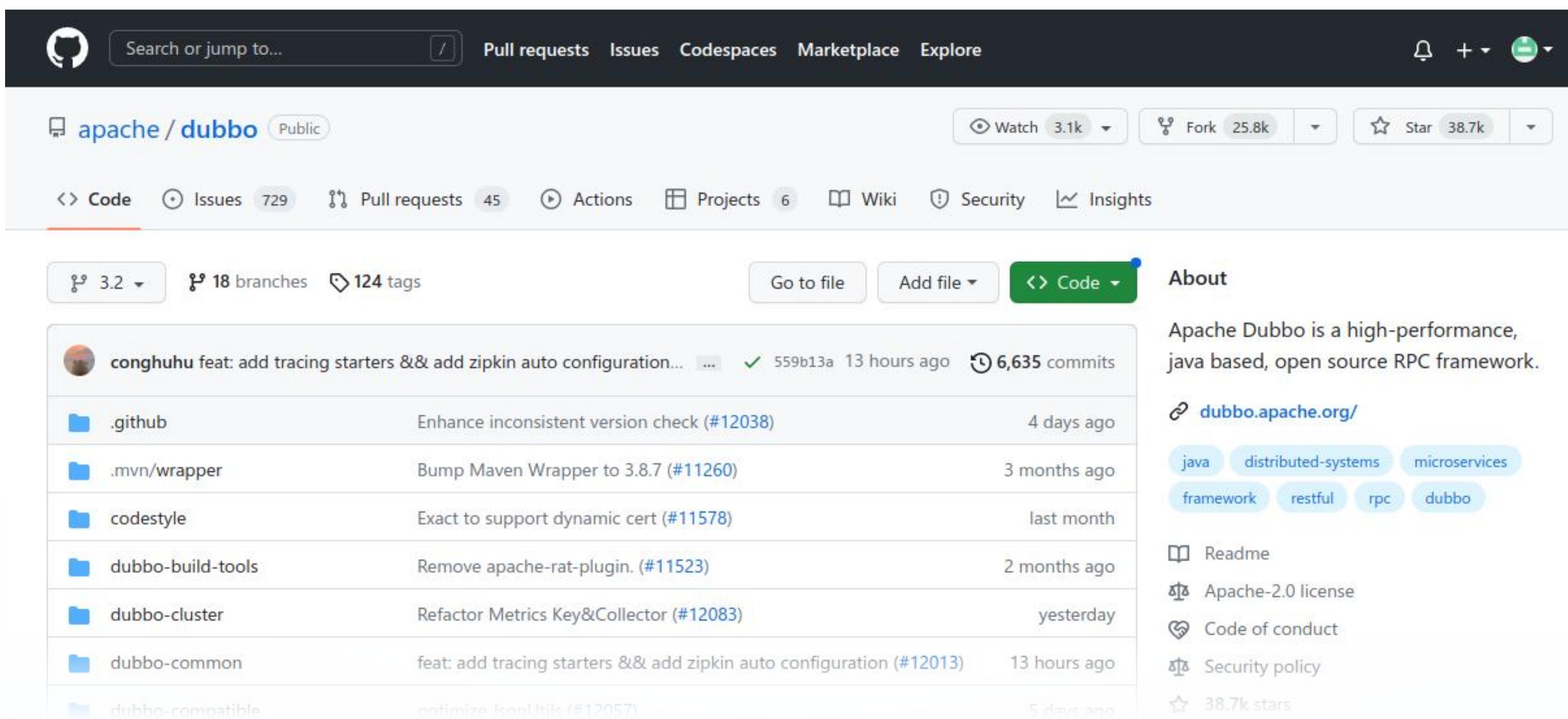
*Follow along at [emilie.ma/ossna2024](https://emilie.ma/ossna2024)*

# In this talk:

- novel collaboration patterns we observed in OSS
- PR-Issue Graph attributes + implications
- **takeaway:** new outlook on managing OSS, know how to identify workflow types and their implications

*Follow along at [emilie.ma/ossna2024](http://emilie.ma/ossna2024)*

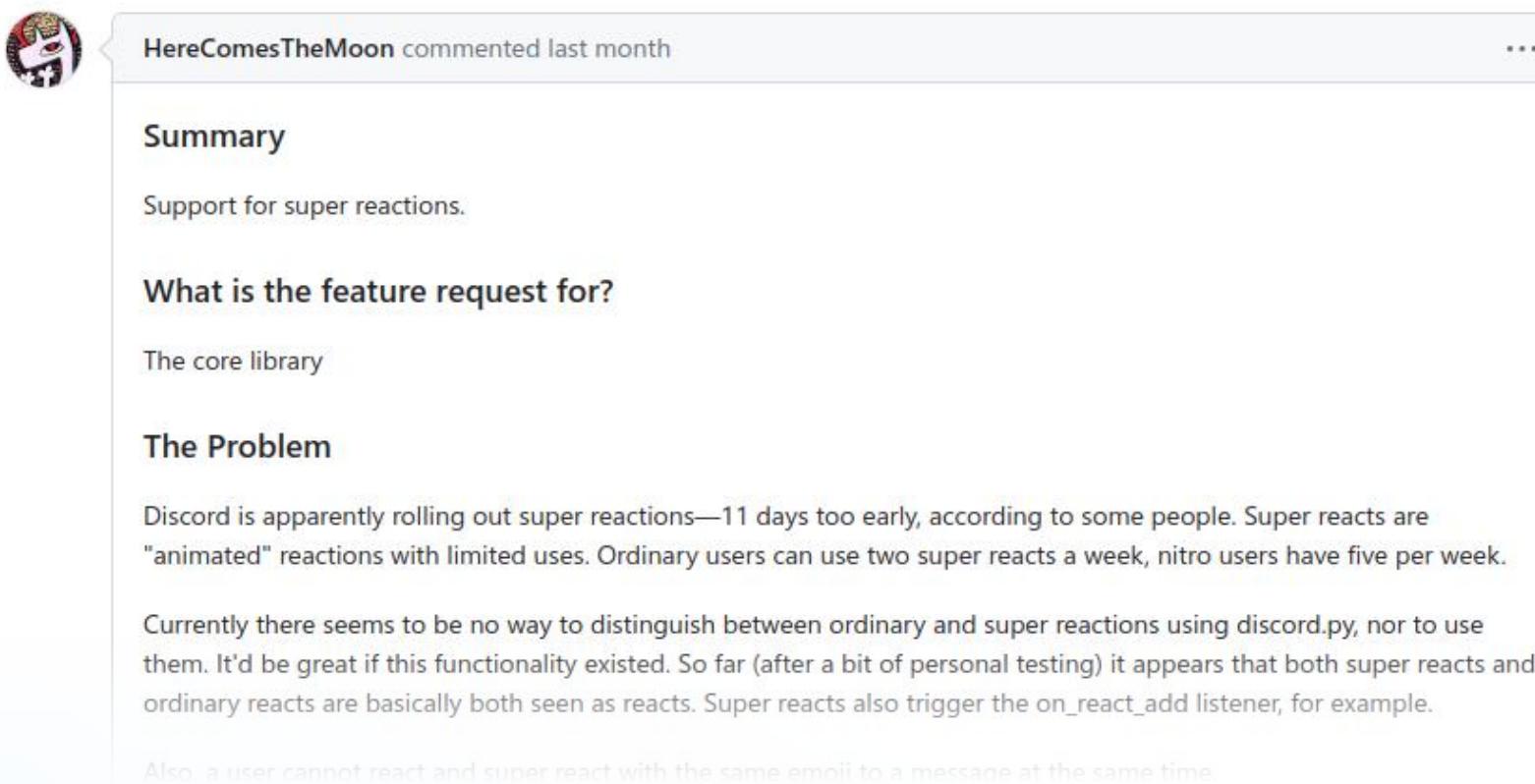
# GitHub: “Where the world builds software”



# Issues: bug reports, feature requests

Support for super reacts #9309

 Open HereComesTheMoon opened this issue last month · 1 comment

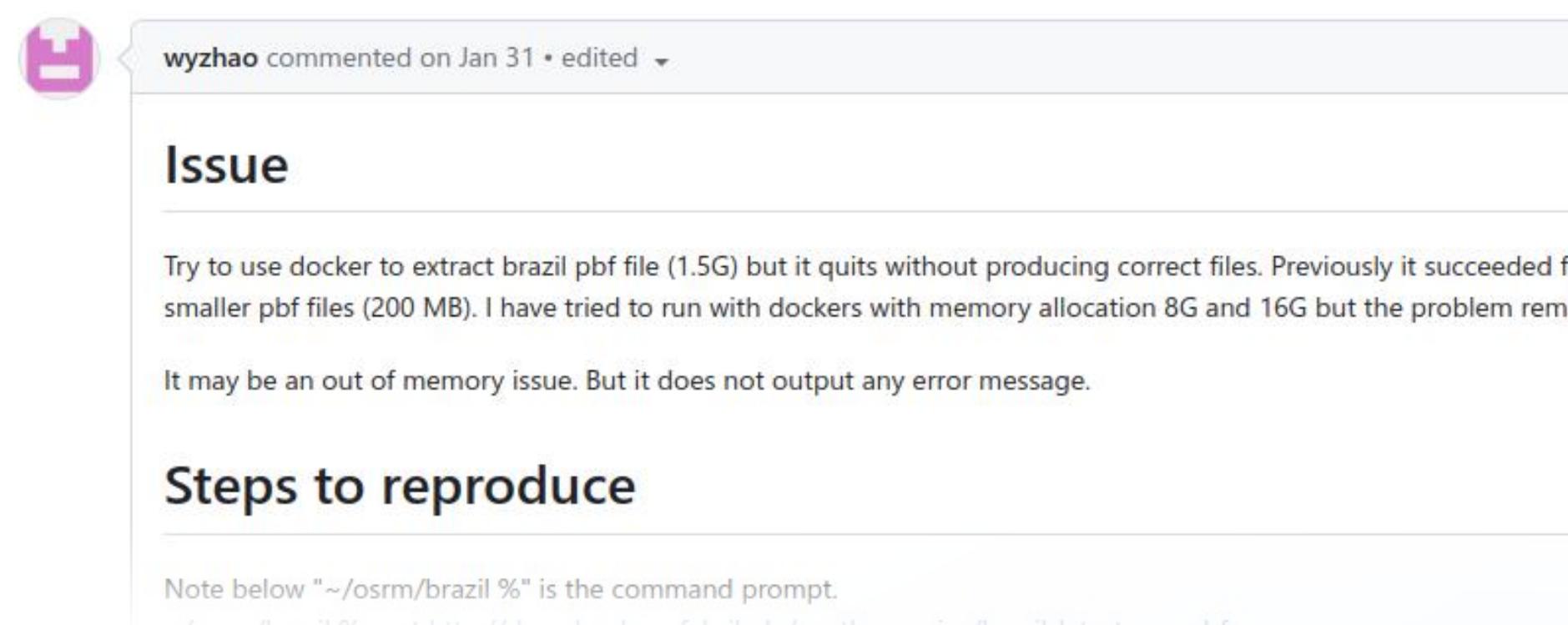


The screenshot shows a GitHub issue page for pull request #9309. At the top, there's a green 'Open' button and a note that 'HereComesTheMoon opened this issue last month · 1 comment'. Below this, a comment from 'HereComesTheMoon' is shown, dated 'commented last month'. The comment body contains sections for 'Summary' ('Support for super reactions.'), 'What is the feature request for?' ('The core library'), and 'The Problem' ('Discord is apparently rolling out super reactions—11 days too early, according to some people. Super reacts are "animated" reactions with limited uses. Ordinary users can use two super reacts a week, nitro users have five per week. Currently there seems to be no way to distinguish between ordinary and super reactions using discord.py, nor to use them. It'd be great if this functionality existed. So far (after a bit of personal testing) it appears that both super reacts and ordinary reacts are basically both seen as reacts. Super reacts also trigger the on\_react\_add listener, for example.'). There's also a note at the bottom: 'Also, a user cannot react and super react with the same emoji to a message at the same time.'

Rapptz/discord.py

osrm-extract failed to extract a large pbf file #6533

 Closed wyzhao opened this issue on Jan 31 · 8 comments



The screenshot shows a GitHub issue page for pull request #6533. At the top, there's a purple 'Closed' button and a note that 'wyzhao opened this issue on Jan 31 · 8 comments'. Below this, a comment from 'wyzhao' is shown, dated 'commented on Jan 31 · edited'. The comment body contains sections for 'Issue' ('Try to use docker to extract brazil pbf file (1.5G) but it quits without producing correct files. Previously it succeeded for smaller pbf files (200 MB). I have tried to run with dockers with memory allocation 8G and 16G but the problem remains. It may be an out of memory issue. But it does not output any error message.') and 'Steps to reproduce' ('Note below "~/osrm/brazil %" is the command prompt.  
~/osrm/brazil % wget http://download.geofabrik.de/south-america/brazil-latest.osm.pbf').

Project-OSRM/osrm-backend

# Issues: bug reports, feature requests

Support for super reacts #9309  
unfixed → Open HereComesTheMoon opened this issue last month · 1 comment

The screenshot shows a GitHub issue page for pull request #9309. At the top, it says "Support for super reacts". Below that, there's a green "Open" button and a comment from "HereComesTheMoon" from last month. The issue has one comment. The main content area includes sections for "Summary" (Support for super reactions), "What is the feature request for?" (The core library), and "The Problem" (Discord is apparently rolling out super reactions—11 days too early, according to some people. Super reacts are "animated" reactions with limited uses. Ordinary users can use two super reacts a week, nitro users have five per week). It also notes that currently there seems to be no way to distinguish between ordinary and super reactions using discord.py.

Rapptz/discord.py

osrm-extract failed to extract a large pbf file #6533  
addressed → Closed wyzhao opened this issue on Jan 31 · 8 comments

The screenshot shows a GitHub issue page for pull request #6533. At the top, it says "osrm-extract failed to extract a large pbf file". Below that, there's a purple "Closed" button and a comment from "wyzhao" from Jan 31. The issue has 8 comments. The main content area includes sections for "Issue" (Try to use docker to extract brazil pbf file (1.5G) but it quits without producing correct files. Previously it succeeded for smaller pbf files (200 MB). I have tried to run with dockers with memory allocation 8G and 16G but the problem remains) and "Steps to reproduce" (Note below "~/osrm/brazil %" is the command prompt.  
~/osrm/brazil % wget http://download.geofabrik.de/south-america/brazil-latest.osm.pbf).

Project-OSRM/osrm-backend

# Pull Requests: work addressing an Issue

Fix: "Source Run" field in Model Version UI page points to wrong run model artifact in Run UI page #8156

Merged WeichenXu123 merged 10 commits into mlflow:master from WeichenXu123:bug-8150 3 days ago

Conversation 36 Commits 10 Checks 24 Files changed 4 +61 -3

WeichenXu123 commented 2 weeks ago • edited

Related Issues/PRs

#8150

What changes are proposed in this pull request?

Fix: "Source Run" field in Model Version UI page points to wrong run model artifact in Run UI page

Closes #8150

How is this patch tested?

Existing unit/integration tests  
 New unit/integration tests  
 Manual tests (describe details, including test results, below)

Reviewers

- harupy ✓
- dbczumar
- hubertzub-db ✓

Assignees

No one assigned

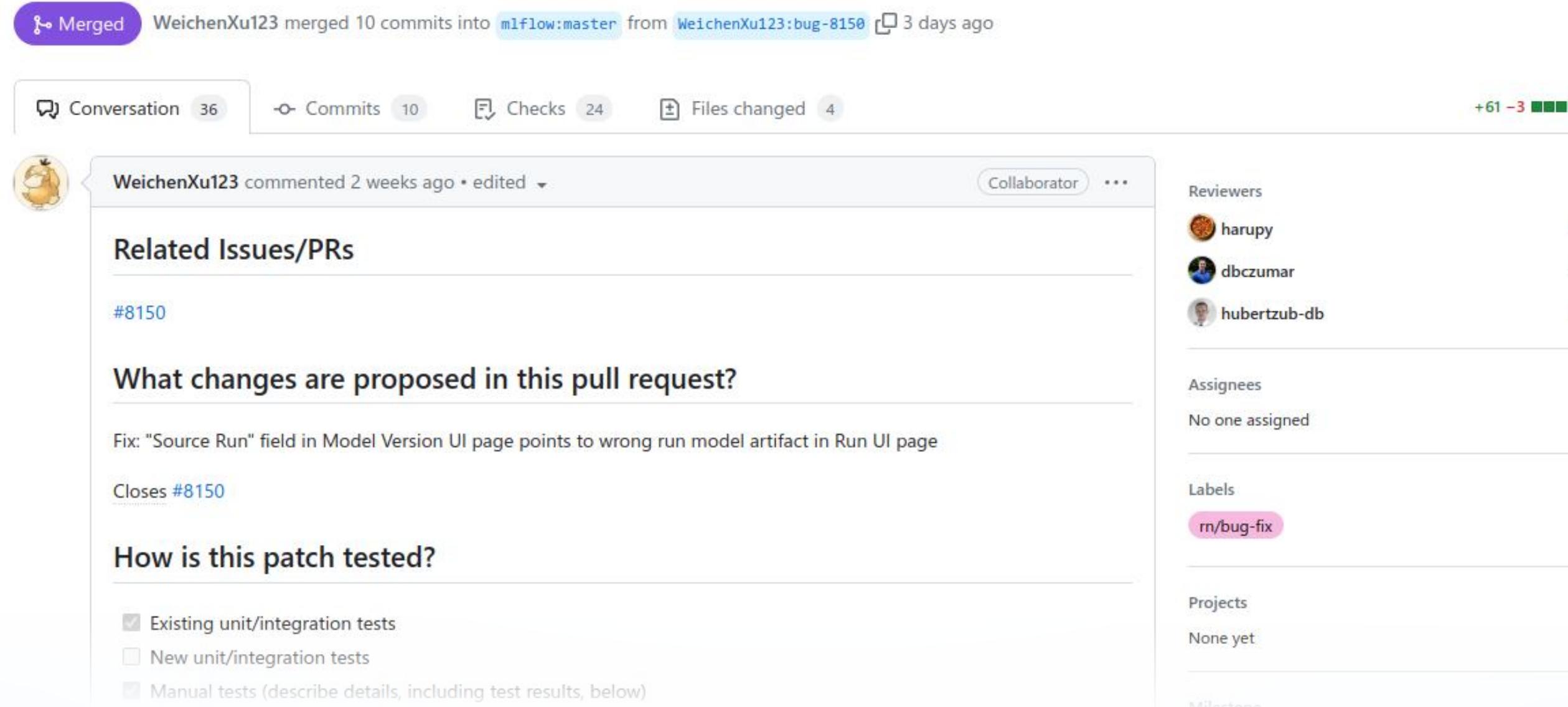
Labels

m/bug-fix

Projects

None yet

Milestone



mlflow/mlflow

# Pull Requests: work addressing an Issue

work was accepted →  
possible statuses:  
open, closed, merged

Fix: "Source Run" field in Model Version UI page points to wrong run model artifact in Run UI page #8156

Merged WeichenXu123 merged 10 commits into mlflow:master from WeichenXu123:bug-8150 3 days ago

Conversation 36 Commits 10 Checks 24 Files changed 4 +61 -3

WeichenXu123 commented 2 weeks ago • edited

Related Issues/PRs  
#8150

What changes are proposed in this pull request?  
Fix: "Source Run" field in Model Version UI page points to wrong run model artifact in Run UI page  
Closes #8150

How is this patch tested?  
 Existing unit/integration tests  
 New unit/integration tests  
 Manual tests (describe details, including test results, below)

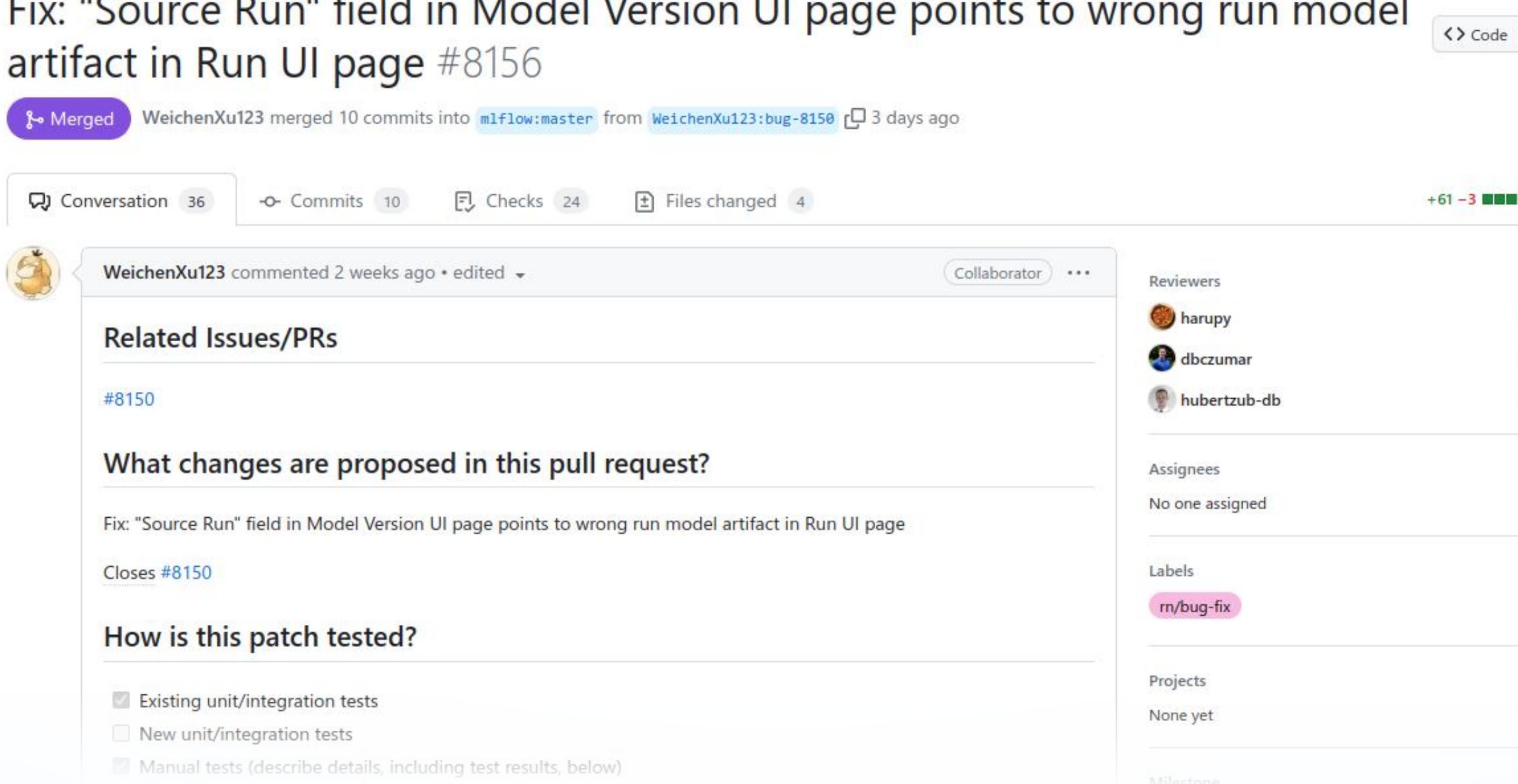
Reviewers  
harupy ✓  
dbczumar  
hubertzub-db ✓

Assignees  
No one assigned

Labels  
m/bug-fix

Projects  
None yet

Milestone



mlflow/mlflow

# Pull Requests: work addressing an Issue

work was accepted →  
possible statuses:  
open, closed, merged

Fix: "Source Run" field in Model Version UI page points to wrong run model artifact in Run UI page #8156

Merged WeichenXu123 merged 10 commits into mlflow:master from WeichenXu123:bug-8150 3 days ago

Conversation 36 Commits 10 Checks 24 Files changed 4 +61 -3

WeichenXu123 commented 2 weeks ago • edited

Related Issues/PRs  
#8150

What changes are proposed in this pull request?  
Fix: "Source Run" field in Model Version UI page points to wrong run model artifact in Run UI page

Closes #8150

How is this patch tested?  
 Existing unit/integration tests  
 New unit/integration tests  
 Manual tests (describe details, including test results, below)

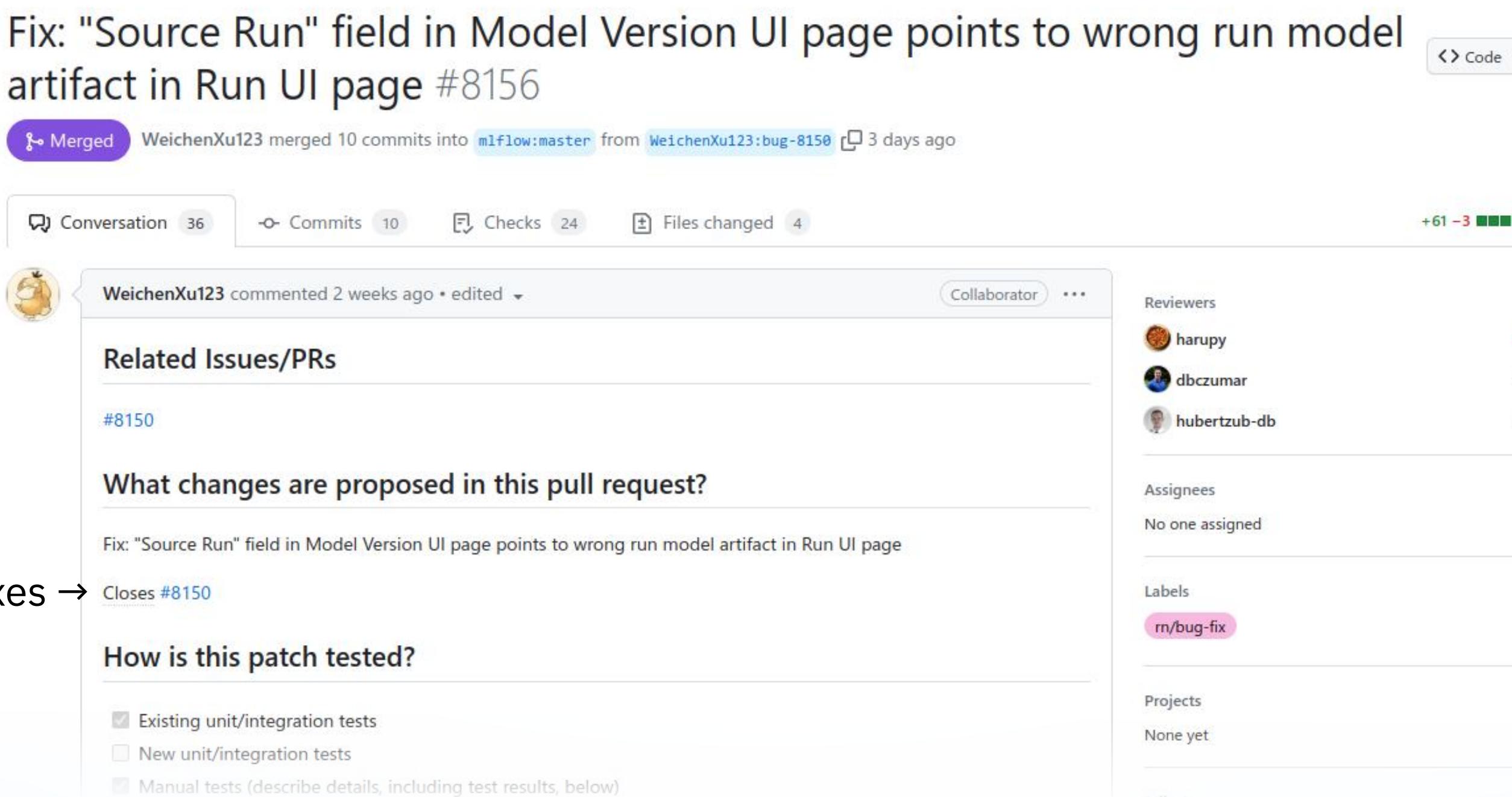
Reviewers  
harupy ✓  
dbczumar  
hubertzub-db ✓

Assignees  
No one assigned

Labels  
m/bug-fix

Projects  
None yet

Milestone



mlflow/mlflow

# Issue ↔ PR = A link!

Fix: "Source Run" field in Model Version UI page artifact in Run UI page #8156

Merged WeichenXu123 merged 10 commits into mlflow:master from WeichenXu123:bug-8150 3 d

Conversation 36 Commits 10 Checks 24 Files changed 4

WeichenXu123 commented 2 weeks ago • edited

**Related Issues/PRs**

#8150

**What changes are proposed in this pull request?**

Fix: "Source Run" field in Model Version UI page points to wrong run model artifact in Run UI pag

Closes #8150

**How is this patch tested?**

Existing unit/integration tests  
 New unit/integration tests  
 Manual tests (describe details, including test results, below)

[BUG] "Source Run" field in Model Version UI page points model artifact in Run UI page #8150

Closed amesar opened this issue 2 weeks ago • 4 comments • Fixed by #8156

amesar commented 2 weeks ago

**Willingness to contribute**

No. I cannot contribute a bug fix at this time.

**MLflow version**

2.2.2

**System information**

- OS Platform and Distribution (e.g., Linux Ubuntu 16.04):
- Python version:
- yarn version, if running the dev UI:

**Describe the problem**

If you have logged more than one model artifact in a run, the "Source Run" field of the Model Version UI page always links to the last model artifact in the Run UI page.

↑ PR 8156 explicitly linked to Issue 8150 ↑

# Issue ↔ PR = A link!

Fix: "Source Run" field in Model Version UI page artifact in Run UI page #8156

Merged WeichenXu123 merged 10 commits into mlflow:master from WeichenXu123:bug-8150 3 d

Conversation 36 Commits 10 Checks 24 Files changed 4

WeichenXu123 commented 2 weeks ago • edited

**Related Issues/PRs**

#8150

**What changes are proposed in this pull request?**

Fix: "Source Run" field in Model Version UI page points to wrong run model artifact in Run UI pag

keyword → Closes #8150

**How is this patch tested?**

Existing unit/integration tests  
 New unit/integration tests  
 Manual tests (describe details, including test results, below)

[BUG] "Source Run" field in Model Version UI page points model artifact in Run UI page #8150

Closed amesar opened this issue 2 weeks ago • 4 comments • Fixed by #8156

amesar commented 2 weeks ago

**Willingness to contribute**

No. I cannot contribute a bug fix at this time.

**MLflow version**

2.2.2

**System information**

- OS Platform and Distribution (e.g., Linux Ubuntu 16.04):
- Python version:
- yarn version, if running the dev UI:

**Describe the problem**

If you have logged more than one model artifact in a run, the "Source Run" field of the Model Version UI page always links to the last model artifact in the Run UI page.

↑ PR 8156 explicitly linked to Issue 8150 ↑

# Issue ↔ PR = A link!

Fix: "Source Run" field in Model Version UI page artifact in Run UI page #8156

Merged WeichenXu123 merged 10 commits into mlflow:master from WeichenXu123:bug-8150 3 d

Conversation 36 Commits 10 Checks 24 Files changed 4

WeichenXu123 commented 2 weeks ago • edited

Related Issues/PRs

#8150

What changes are proposed in this pull request?

Fix: "Source Run" field in Model Version UI page points to wrong run model artifact in Run UI pag

Closes #8150

How is this patch tested?

Existing unit/integration tests  
 New unit/integration tests  
 Manual tests (describe details, including test results, below)

[BUG] "Source Run" field in Model Version UI page points model artifact in Run UI page #8150

Closed amesar opened this issue 2 weeks ago • 4 comments • Fixed by #8156

WeichenXu123 commented 2 weeks ago

Willingness to contribute

No. I cannot contribute a bug fix at this time.

MLflow version

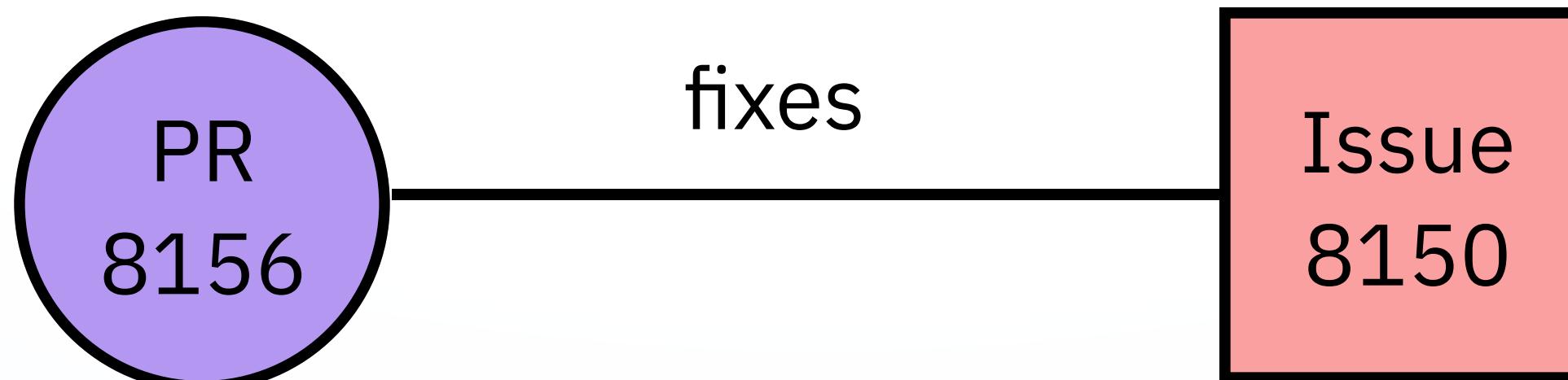
2.2.2

System information

- OS Platform and Distribution (e.g., Linux Ubuntu 16.04):
- Python version:
- yarn version, if running the dev UI:

Describe the problem

If you have logged more than one model artifact in a run, the "Source Run" field of the Model Version UI page always links to the last model artifact in the Run UI page.



## Node Status



Merged



Open



Closed

## Node Type



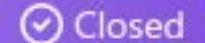
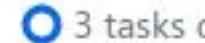
Pull Request



Issue

# Another explicit link: “duplicate”

message.stickers always return empty list #7186

 Closed  3 tasks done Poloin34 opened this issue on Jul 7, 2021 · 2 comments



Poloin34 commented on Jul 7, 2021

## Summary

as title, stickers not listed and no way to get them (even just if there's one)

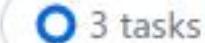


Rapptz commented on Jul 7, 2021

Duplicate of #7140. It's literally a few below this one. Search before opening an issue.



Missing Message.stickers #7140

 Closed  3 tasks done attendeeany opened this issue on Jun 30, 2021 · 6 comments



attendeeany commented on Jun 30, 2021

## Summary

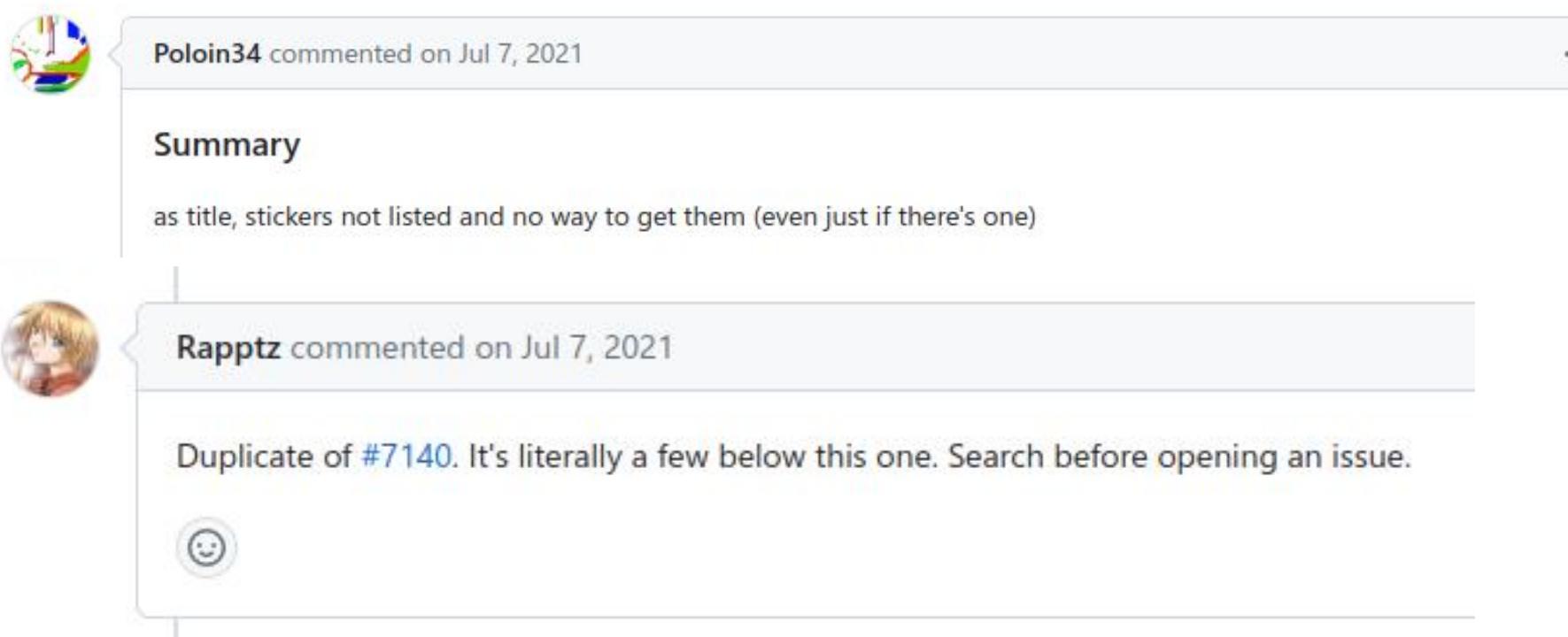
discord.Message.stickers is always empty

*Rapptz/discord.py*

# Another explicit link: “duplicate”

message.stickers always return empty list #7186

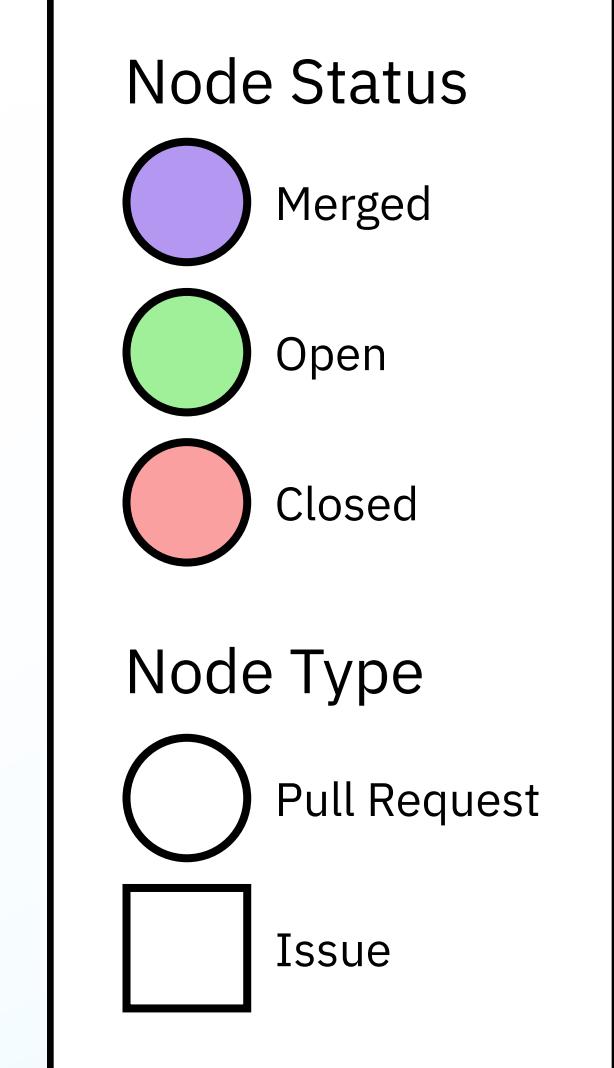
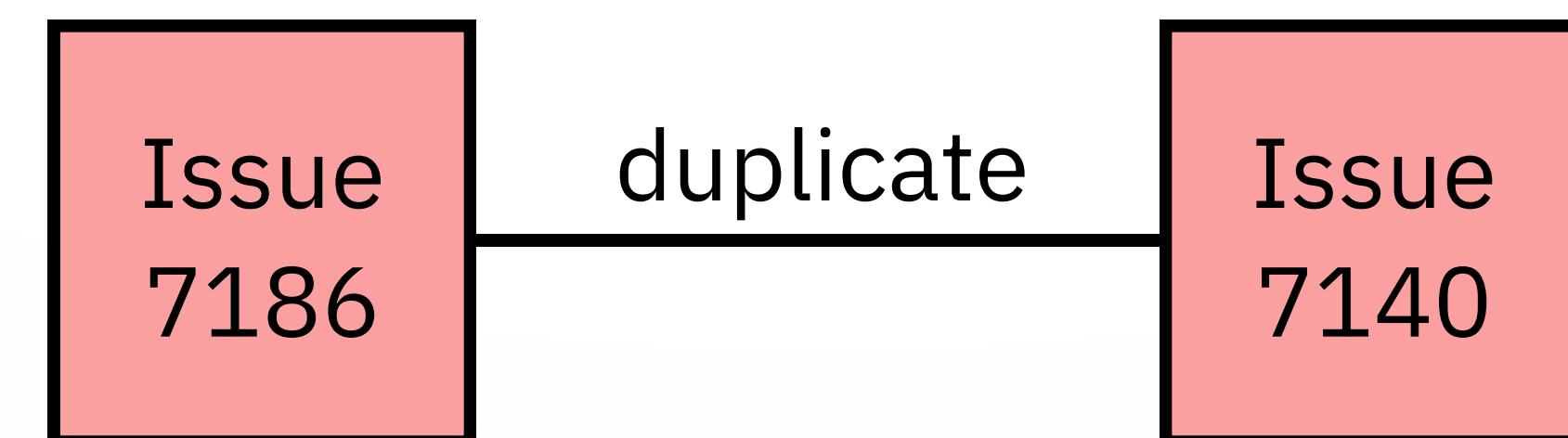
 Closed  3 tasks done Poloin34 opened this issue on Jul 7, 2021 · 2 comments



Rapptz/discord.py

Missing Message.stickers #7140

 Closed  3 tasks done attendeeany opened this issue on Jun 30, 2021 · 6 comments

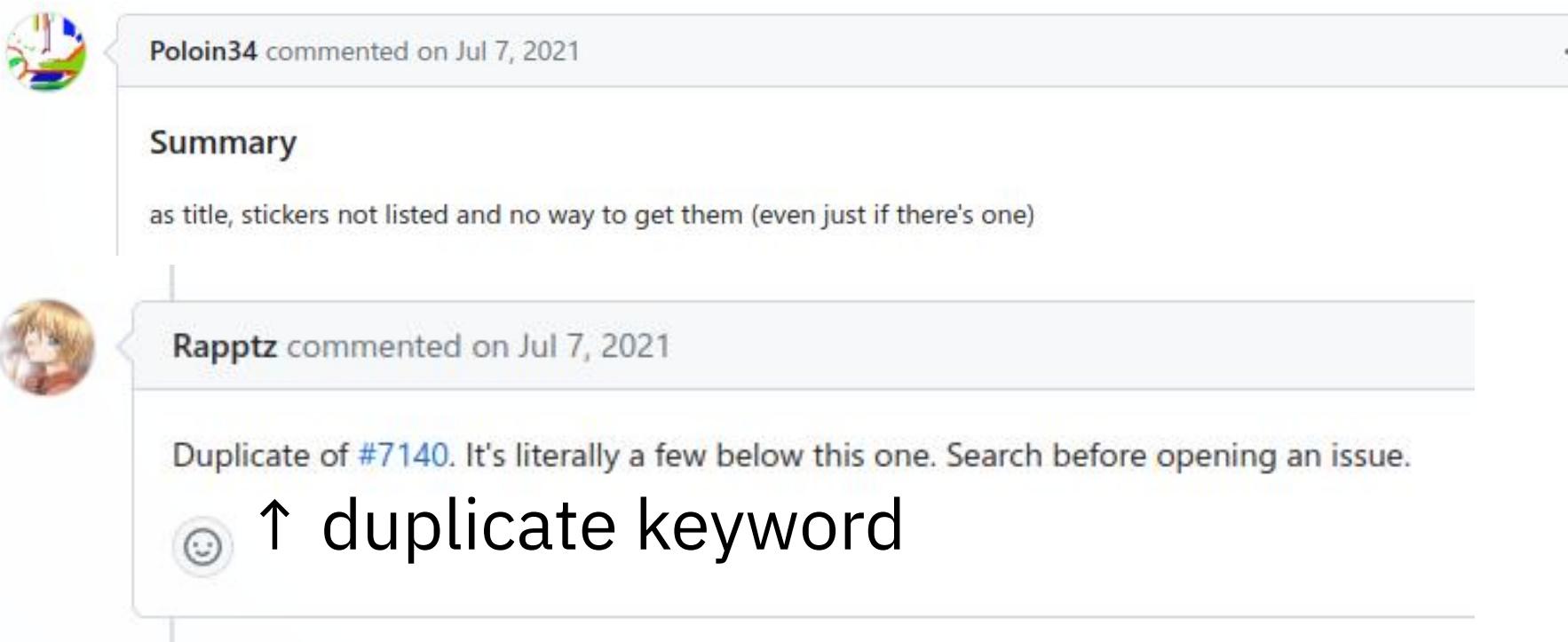


# Another explicit link: “duplicate”

message.stickers always return empty list #7186

Closed 3 tasks done Poloin34 opened this issue on Jul 7, 2021 · 2 comments

↑ created later

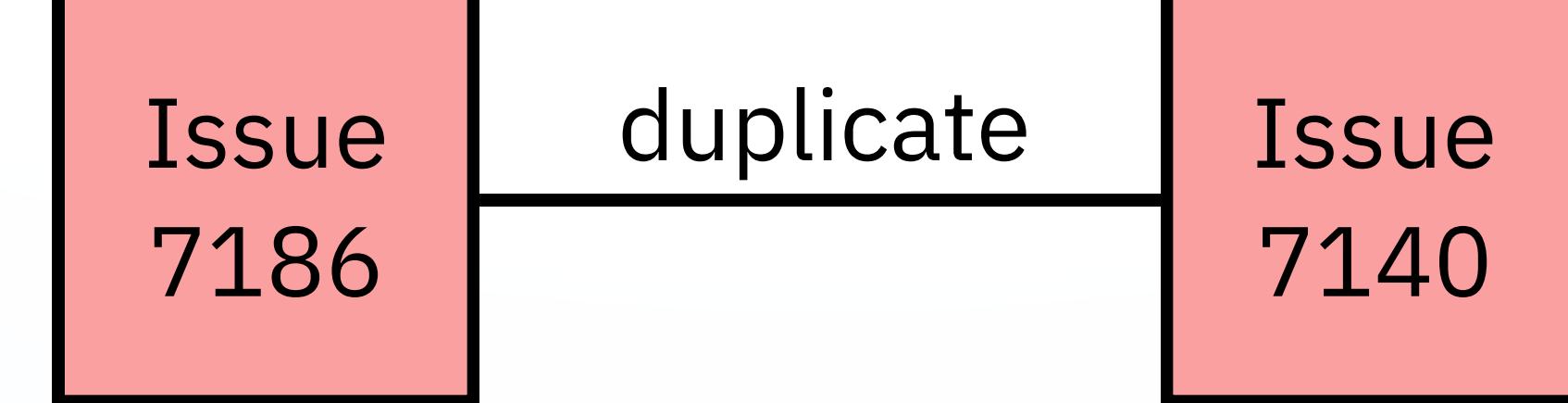


Rapptz/discord.py

Missing Message.stickers #7140

Closed 3 tasks done attendeeany opened this issue on Jun 30, 2021 · 6 comments

↑ created earlier

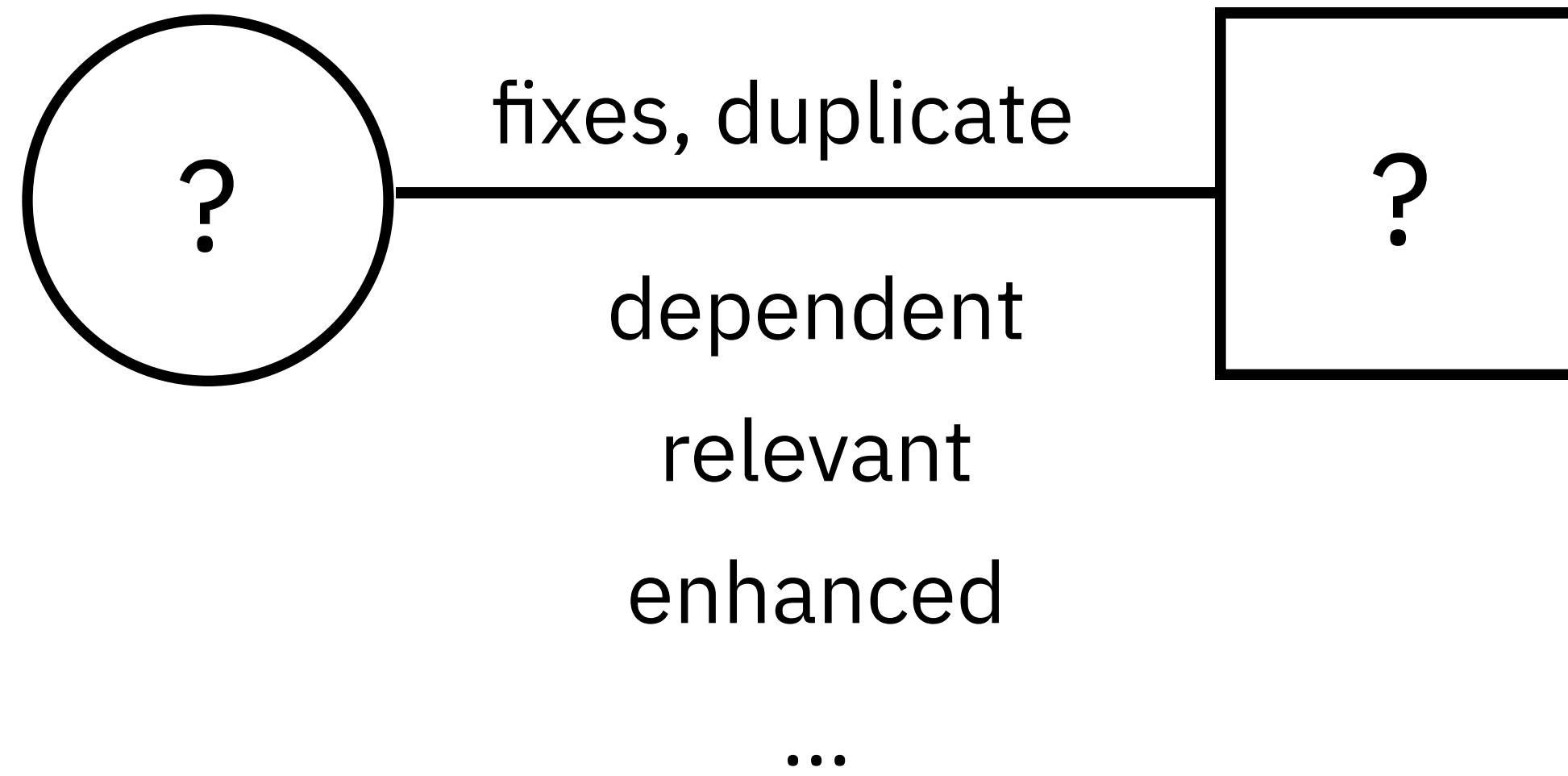


Node Status
<span style="background-color: purple; border-radius: 50%; width: 20px; height: 20px;"></span> Merged
<span style="background-color: green; border-radius: 50%; width: 20px; height: 20px;"></span> Open
<span style="background-color: red; border-radius: 50%; width: 20px; height: 20px;"></span> Closed

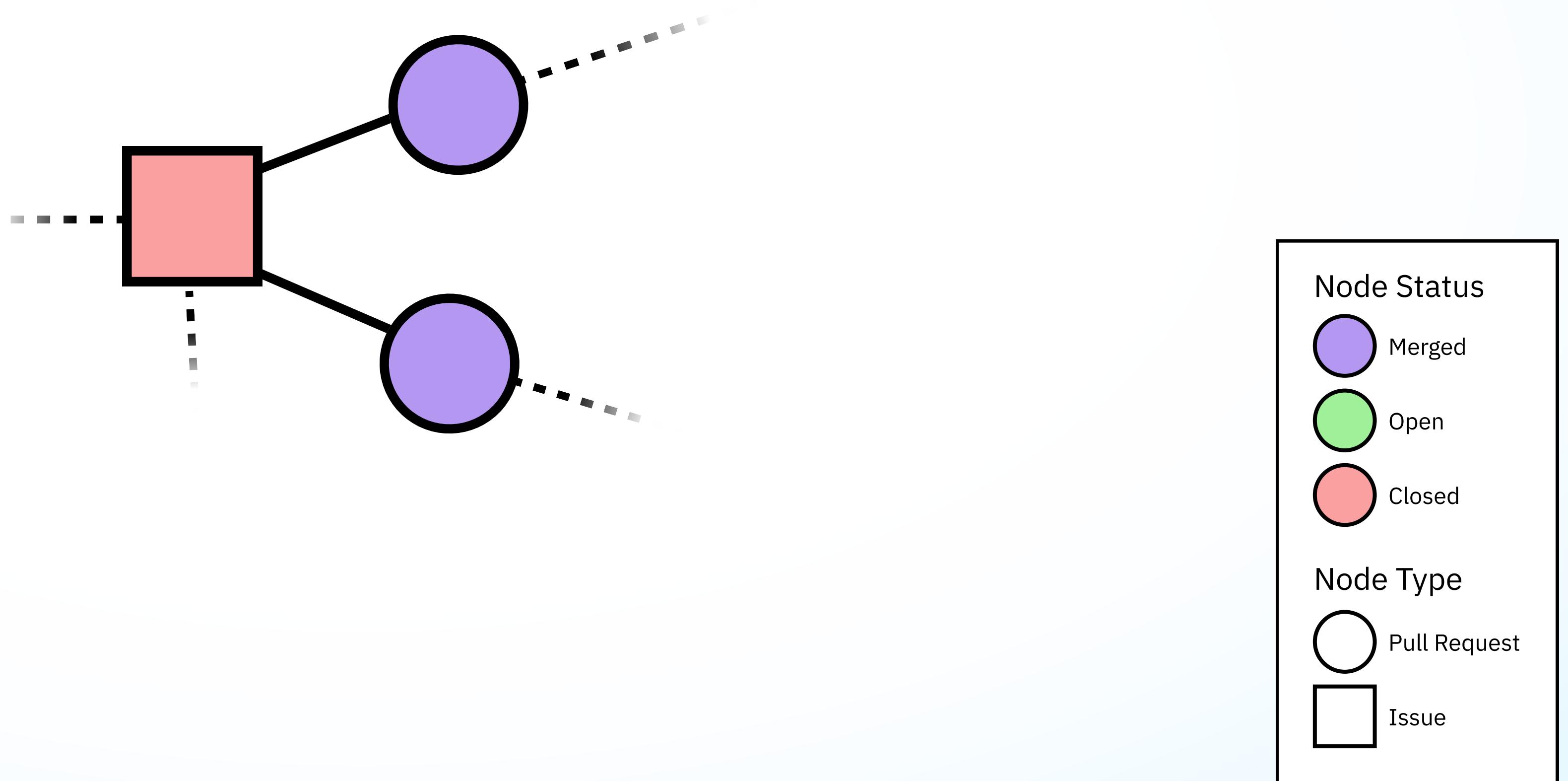
Node Type
<span style="border: 1px solid black; width: 20px; height: 20px; display: inline-block;"></span> Pull Request
<span style="border: 1px solid black; width: 20px; height: 20px; display: inline-block;"></span> Issue

# Prior work has focused on these links.

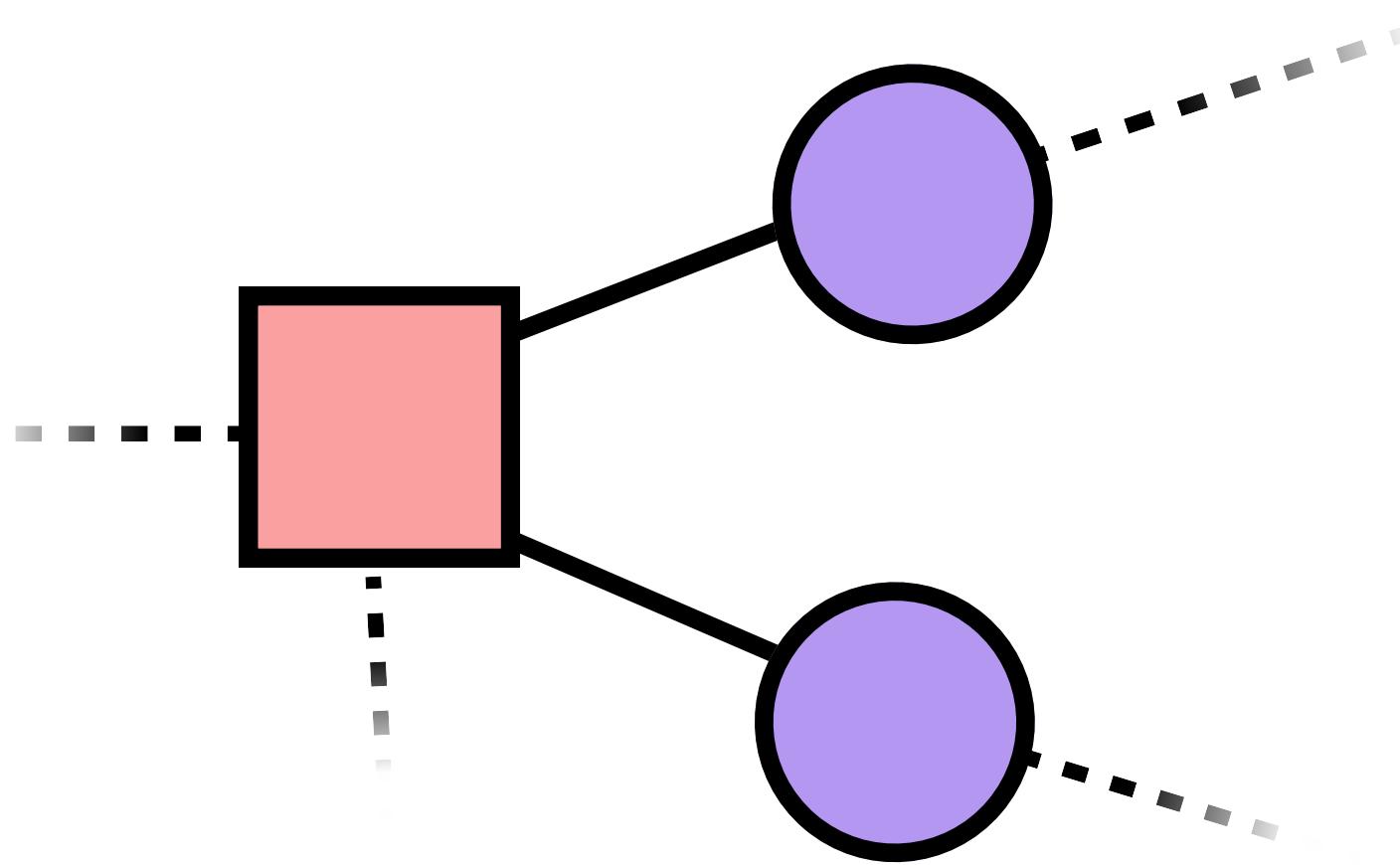


(“@alex, this fixes #9...”, Chopra et al., CSCW 2021;  
“How Are Issue Units Linked?...”, Li et al., APSEC 2018;  
“Understanding shared links and their intentions...”, Wang et al., ESE 2021;  
“The review linkage graph for code review analytics...”, Hirao et al., ESEC/FSE 2019;  
“Traceability Network Analysis...” Nicholson et al., AIRE 2020)

# Going beyond links: Graph Perspective



# Going beyond links: Graph Perspective

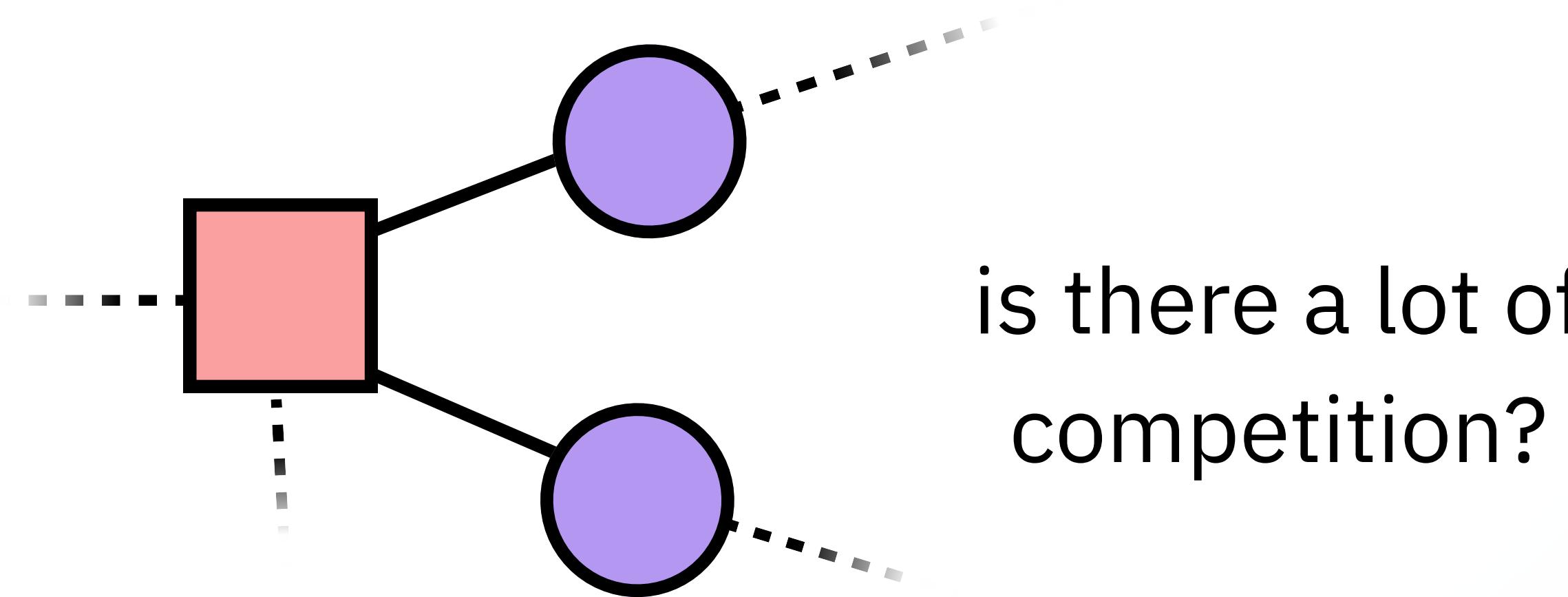


graph perspective = more context = richer  
understanding of work practices

Node Status	
	Merged
	Open
	Closed
Node Type	
	Pull Request
	Issue

# Going beyond links: Graph Perspective

do developers break  
their work up?

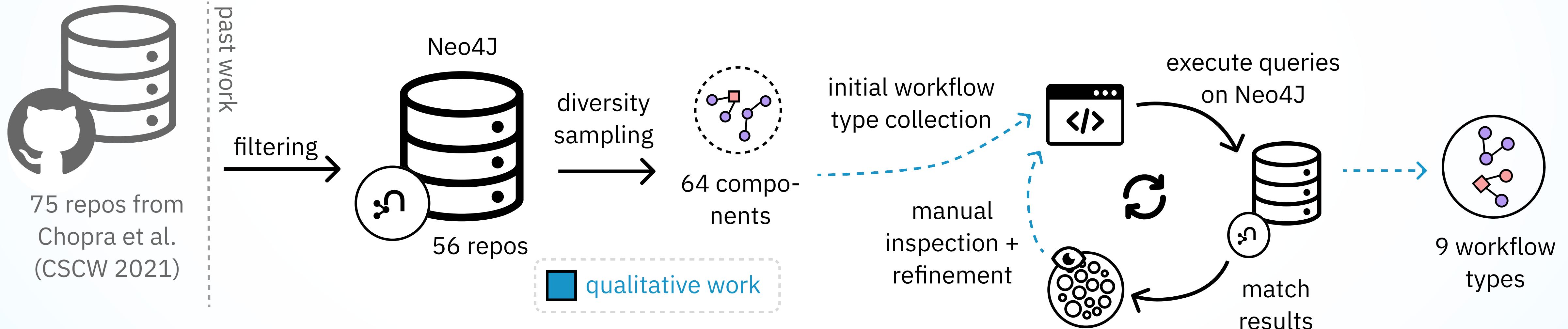


is there a lot of  
competition?

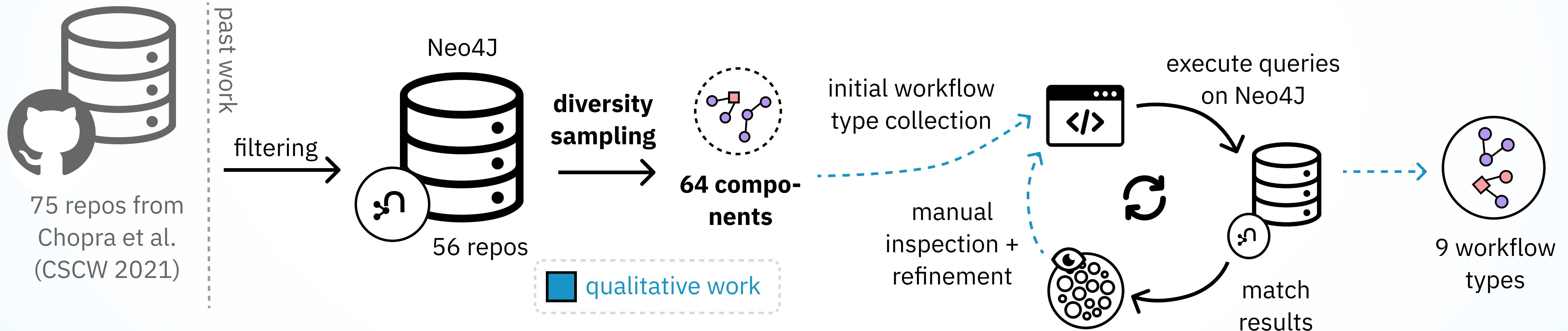
graph perspective = more context = richer  
understanding of work practices

Node Status	
	Merged
	Open
	Closed
Node Type	
	Pull Request
	Issue

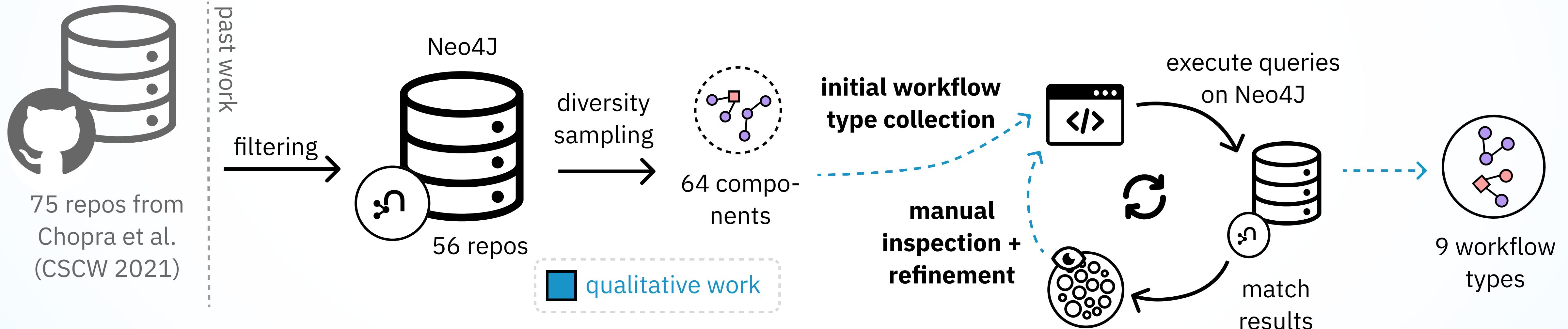
# Analyzing 90K+ Issues and PRs with Neo4j



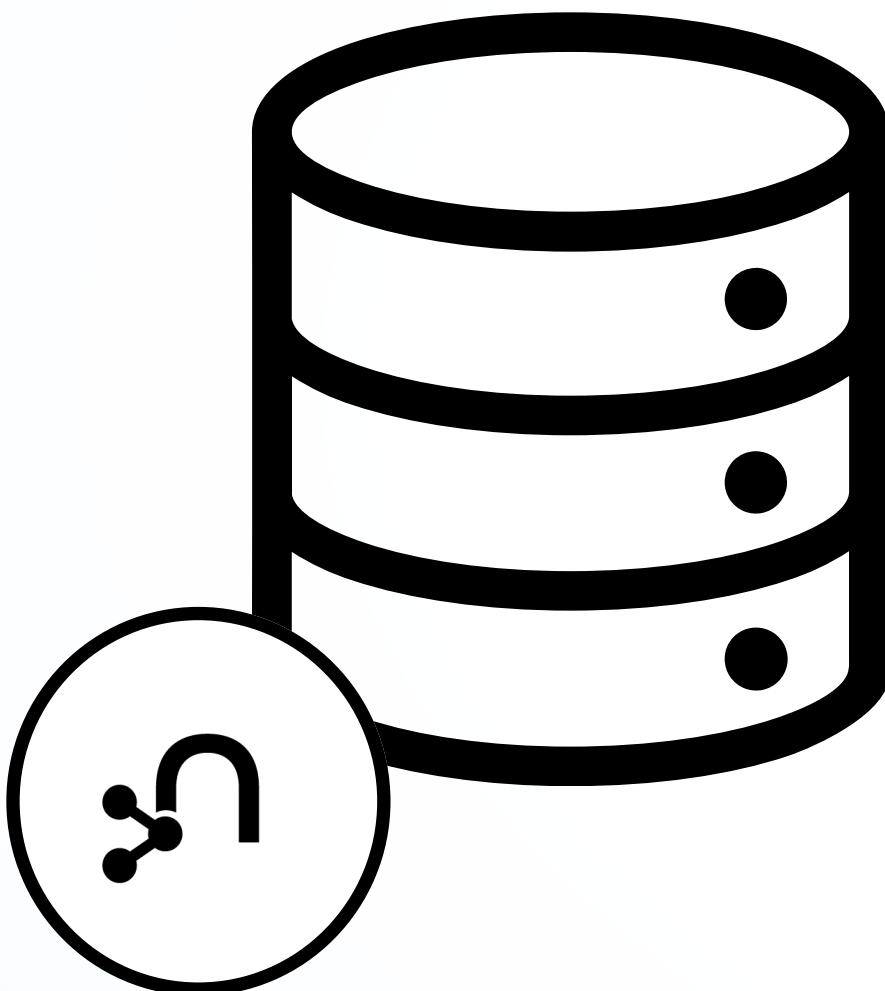
# Analyzing 90K+ Issues and PRs with Neo4j



# Analyzing 90K+ Issues and PRs with Neo4j



# Neo4j + Cypher Queries



Neo4j =  
graph DBMS

The screenshot shows the Neo4j browser interface. On the left, there is a code editor window containing a Cypher query:

```
1 //Competition
2 match (i:issue)-[r {labels:"fixes"}]-(pr:pull_request)
3 where i.status = "closed"
4 with i, collect(distinct pr) as pull_requests, collect (distinct pr.user) as users
5 where size([p_r in pull_requests where p_r.status="merged"]) = 1 and size([p_r in pull_requests where p_r.status="closed"]) >= 1 and size(pull_requests) >= 2 and
size(users) > 1
6 return i, pull_requests
```

To the right of the code editor is a graph visualization area showing a complex network of nodes and relationships. The nodes are colored green and purple, and the relationships are shown as dashed lines. Below the graph, the Neo4j interface displays an 'Overview' section with statistics:

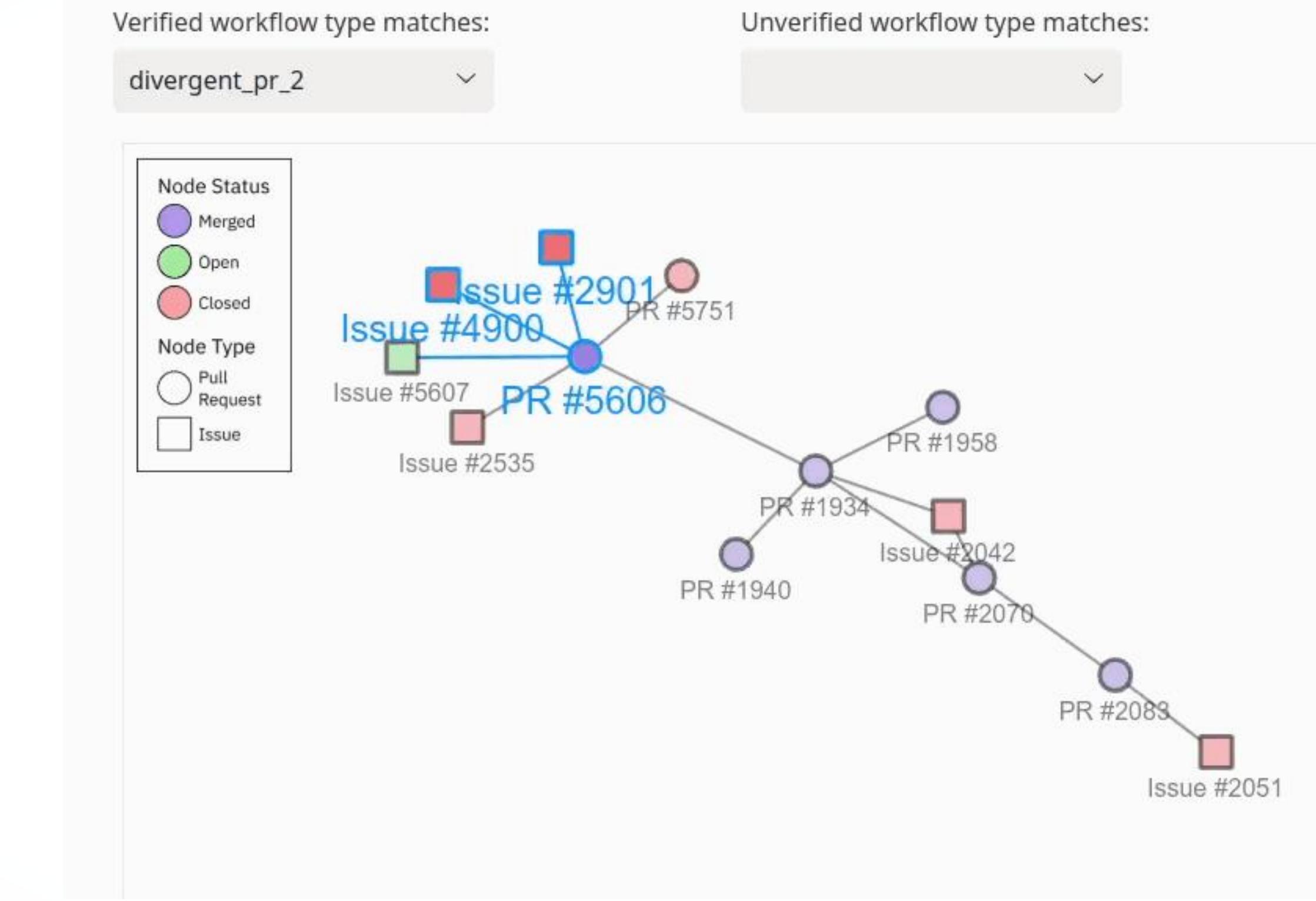
- Node labels:
  - \* (111)
  - Issue (36)
  - pull\_request (75)
- Relationship types:
  - \* (103)
  - RELATED (103)

At the bottom, it says "Displaying 111 nodes, 0 relationships."

↑ used for qualitative coding and iteration

# Workflows Explorer

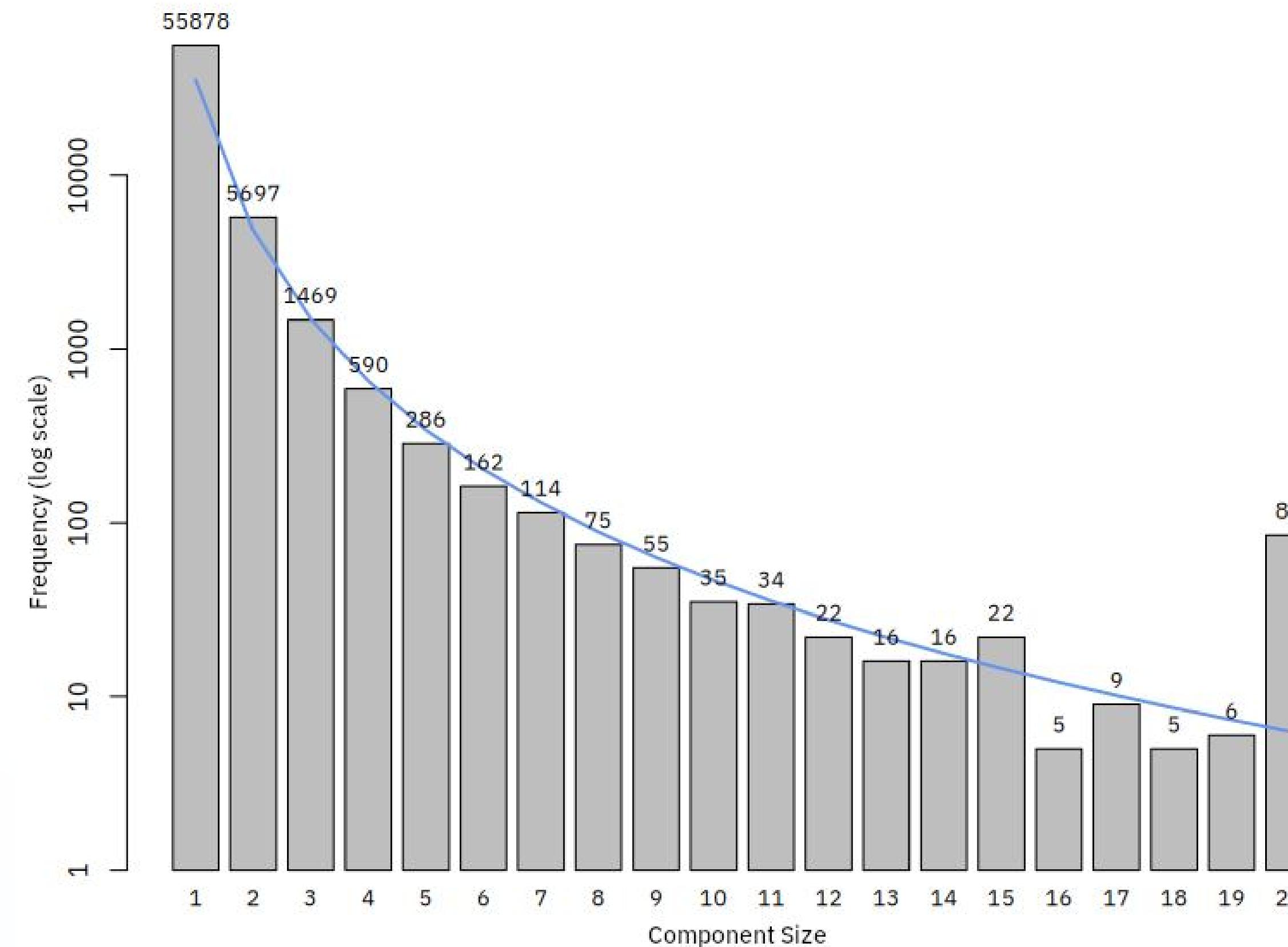
## mlflow/mlflow Workflow Type Matches



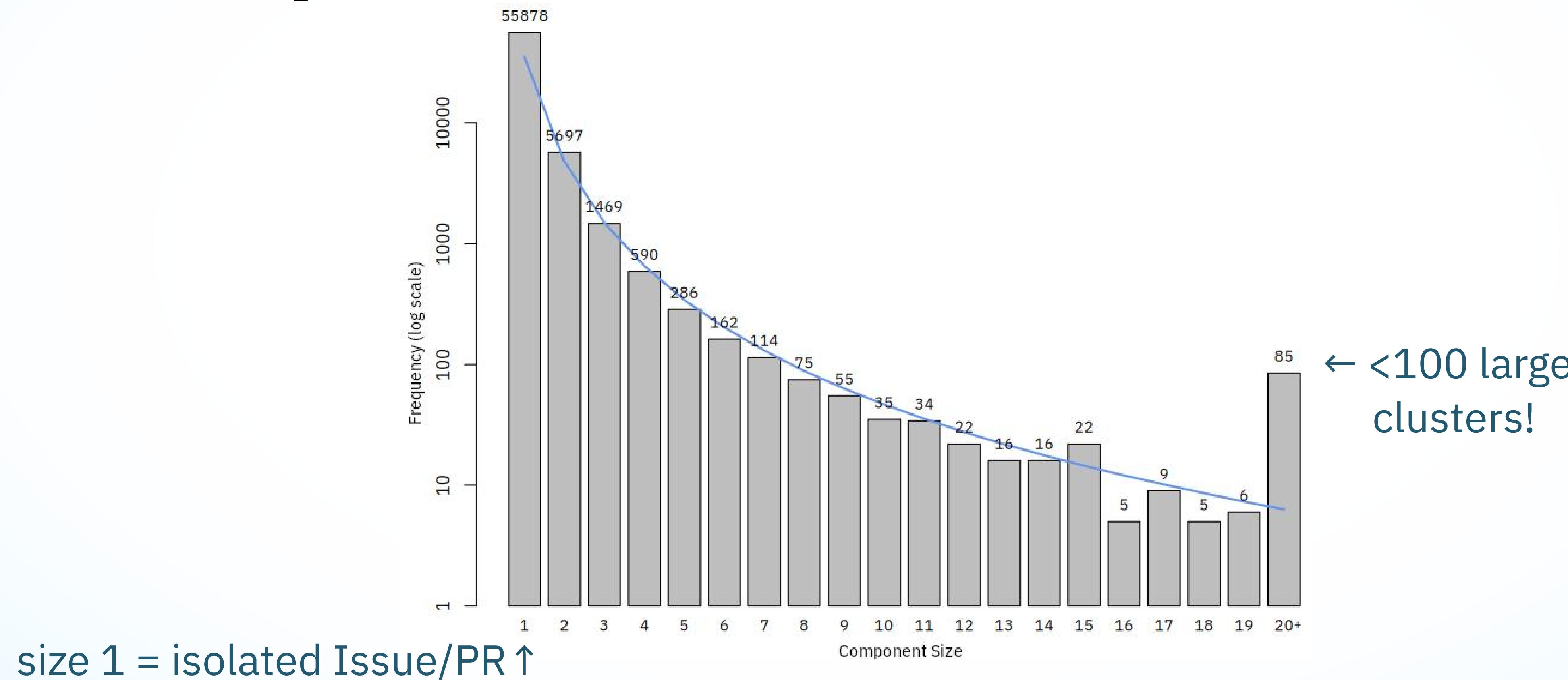
[ubcdllab.github.io/pr-issue-topology-project/](https://ubcdllab.github.io/pr-issue-topology-project/)

↑ our project + workflow type match visualization tool,  
used for conducting interviews

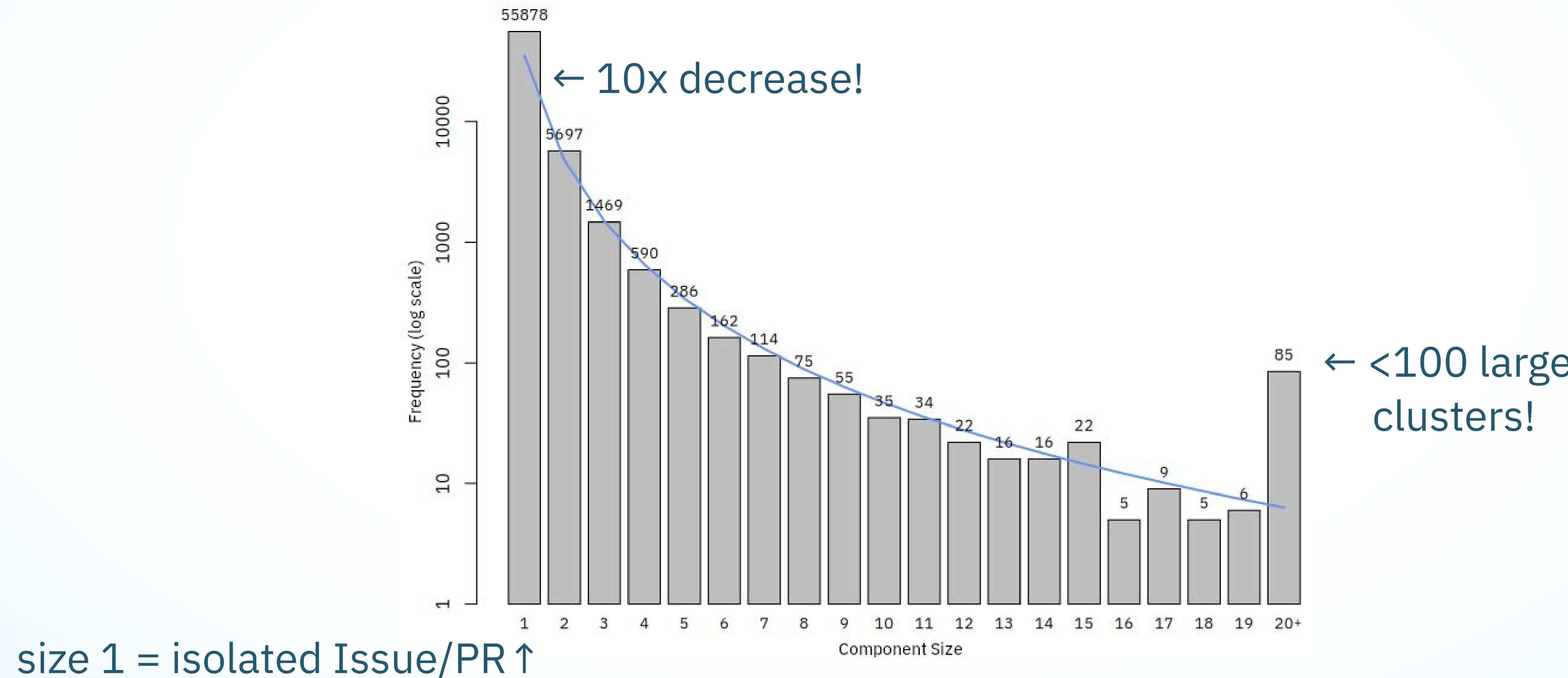
# Cluster sizes follow a power-law distribution.



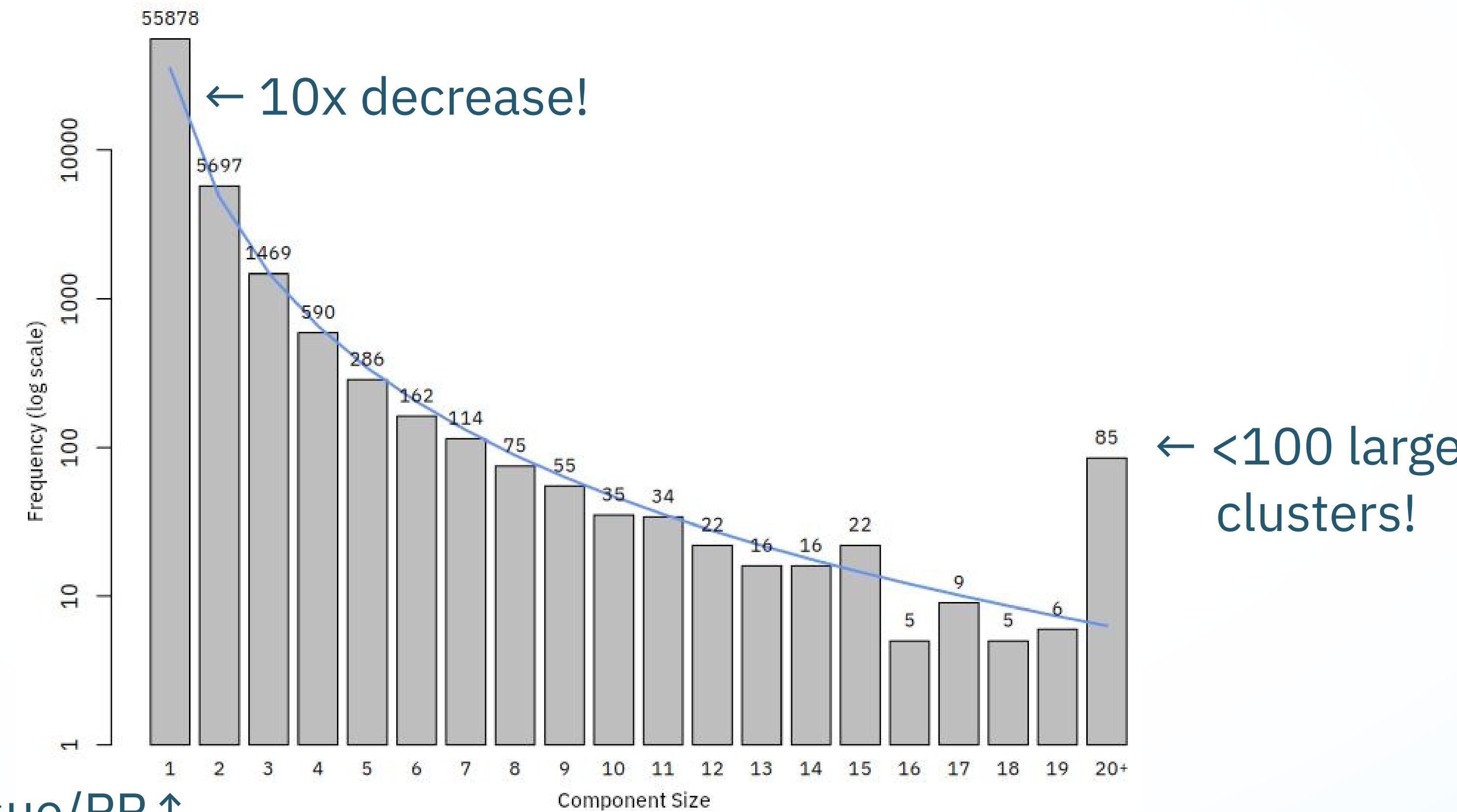
# Cluster sizes follow a power-law distribution.



# Cluster sizes follow a power-law distribution.

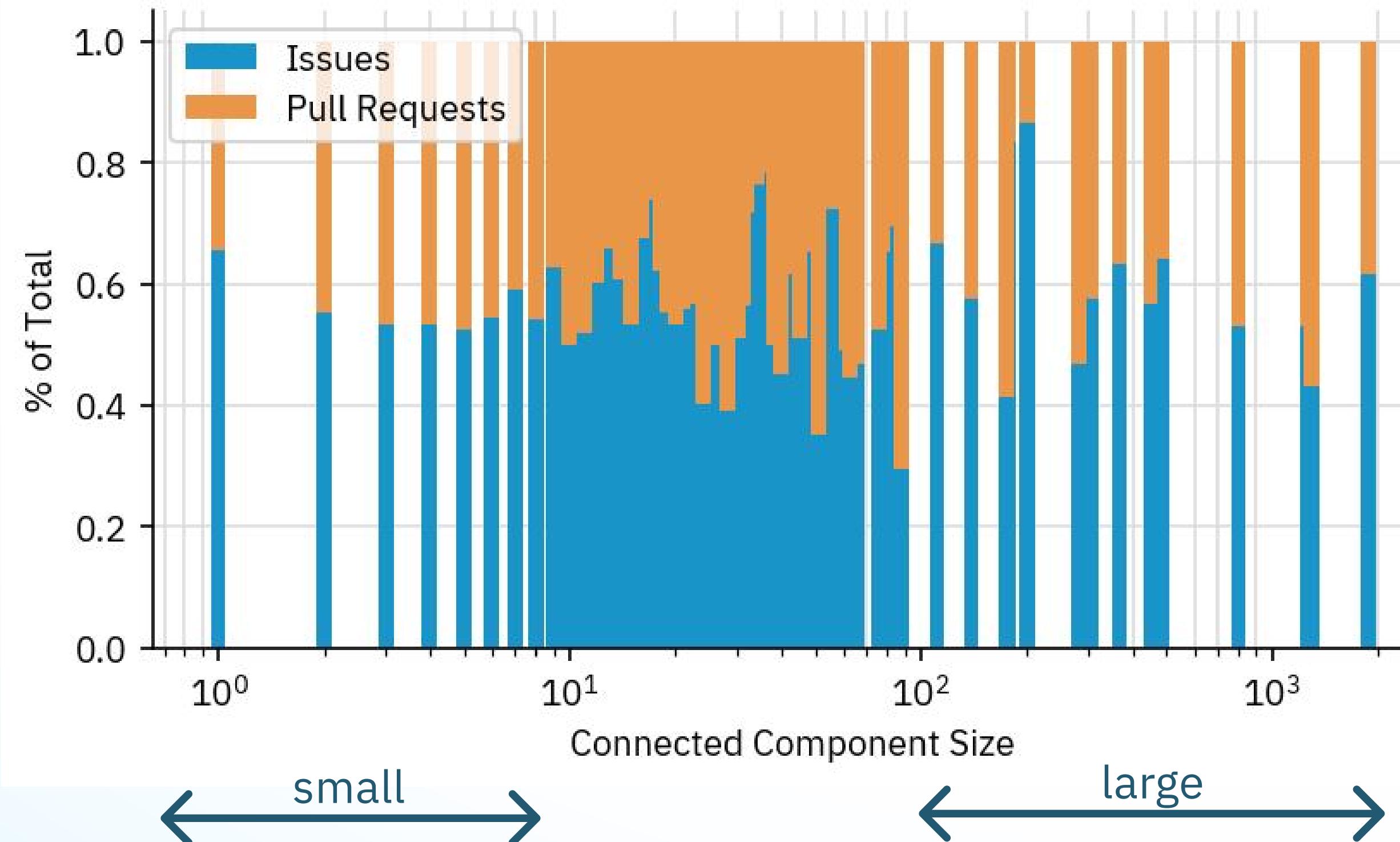


# Cluster sizes follow a power-law distribution.



Can help identify work to prioritize.

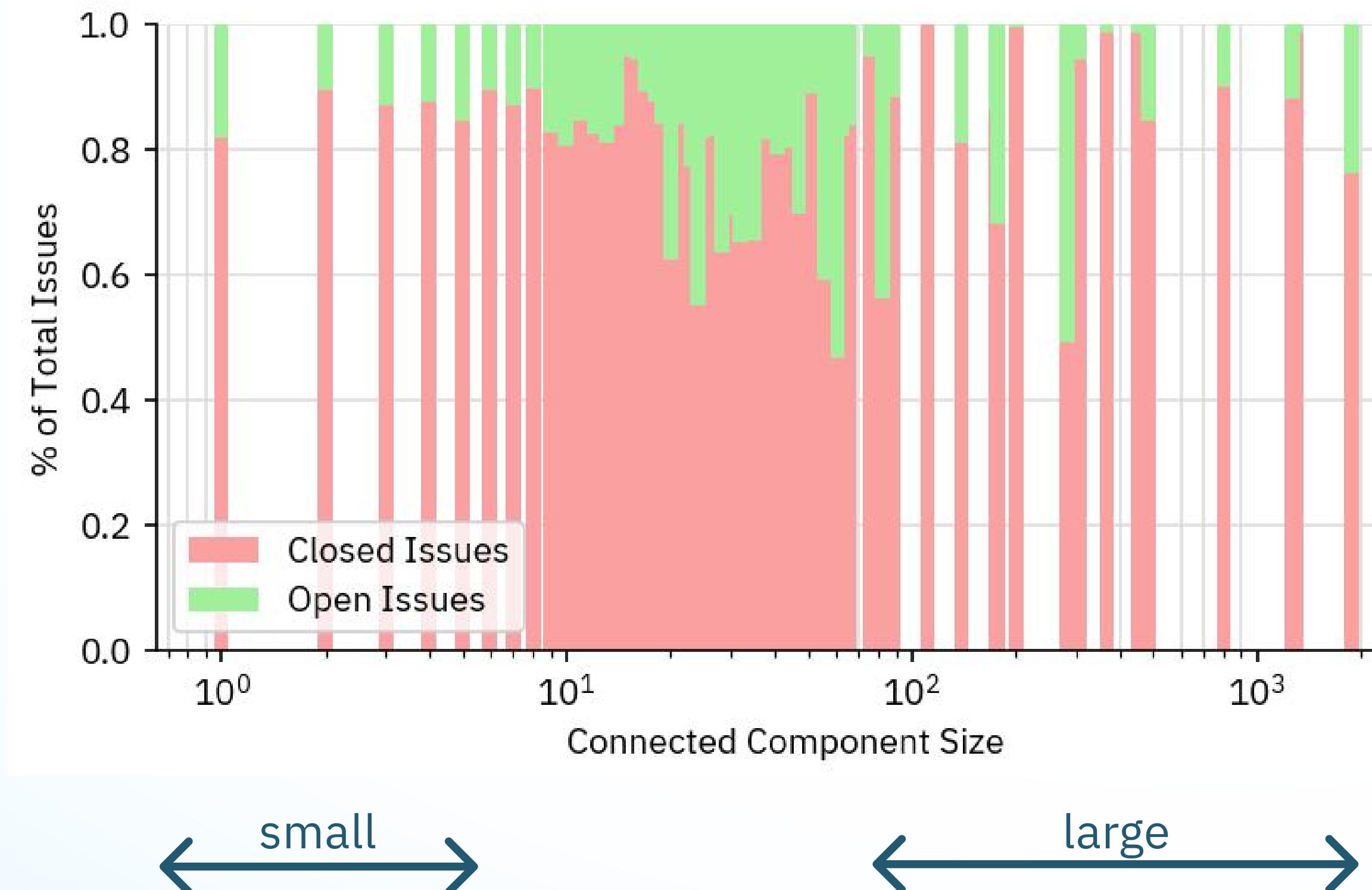
# Different cluster sizes, different types of work.



Small clusters have:

- More **Issues (blue)**

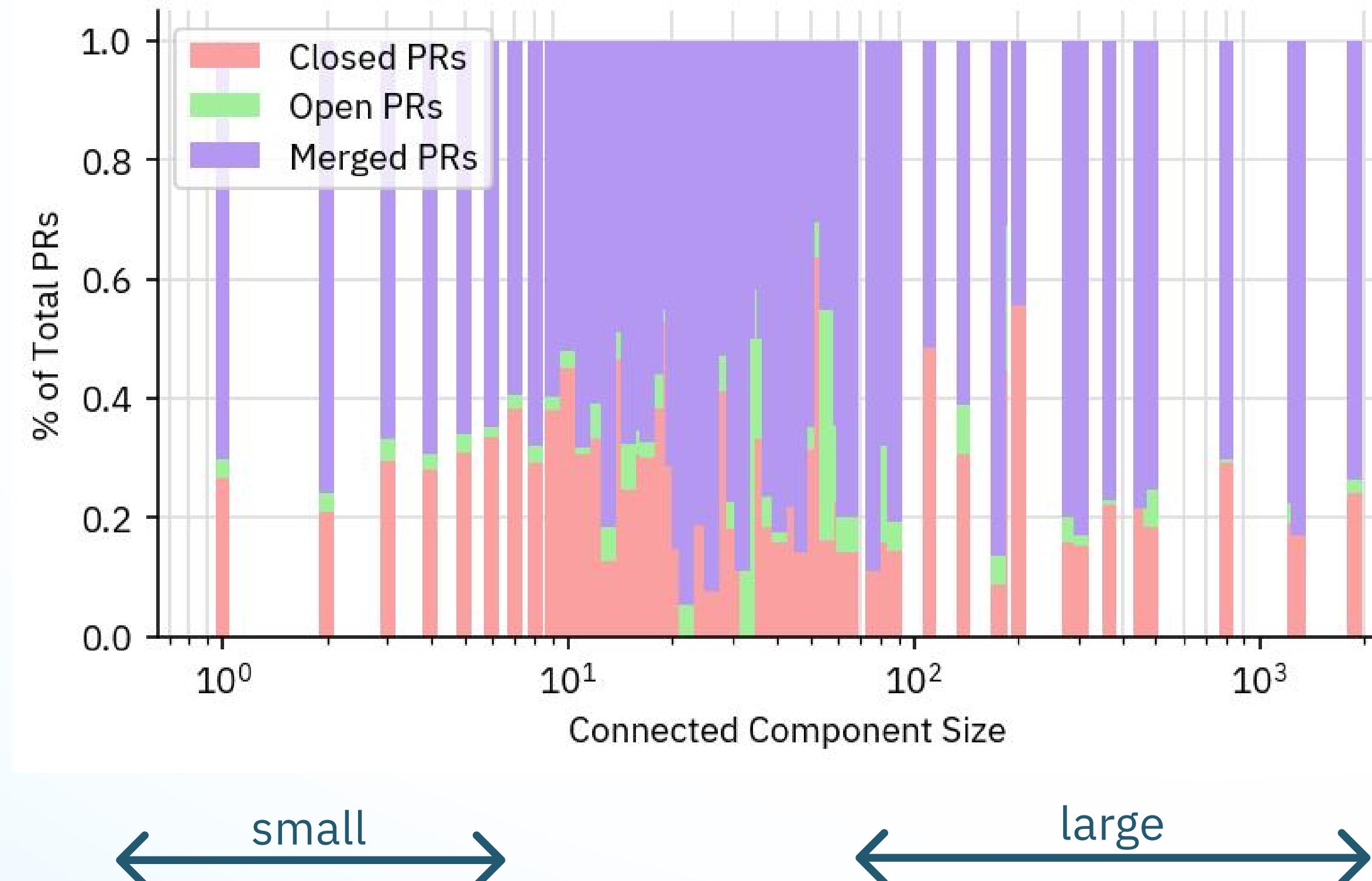
# Different cluster sizes, different types of work.



Small clusters have:

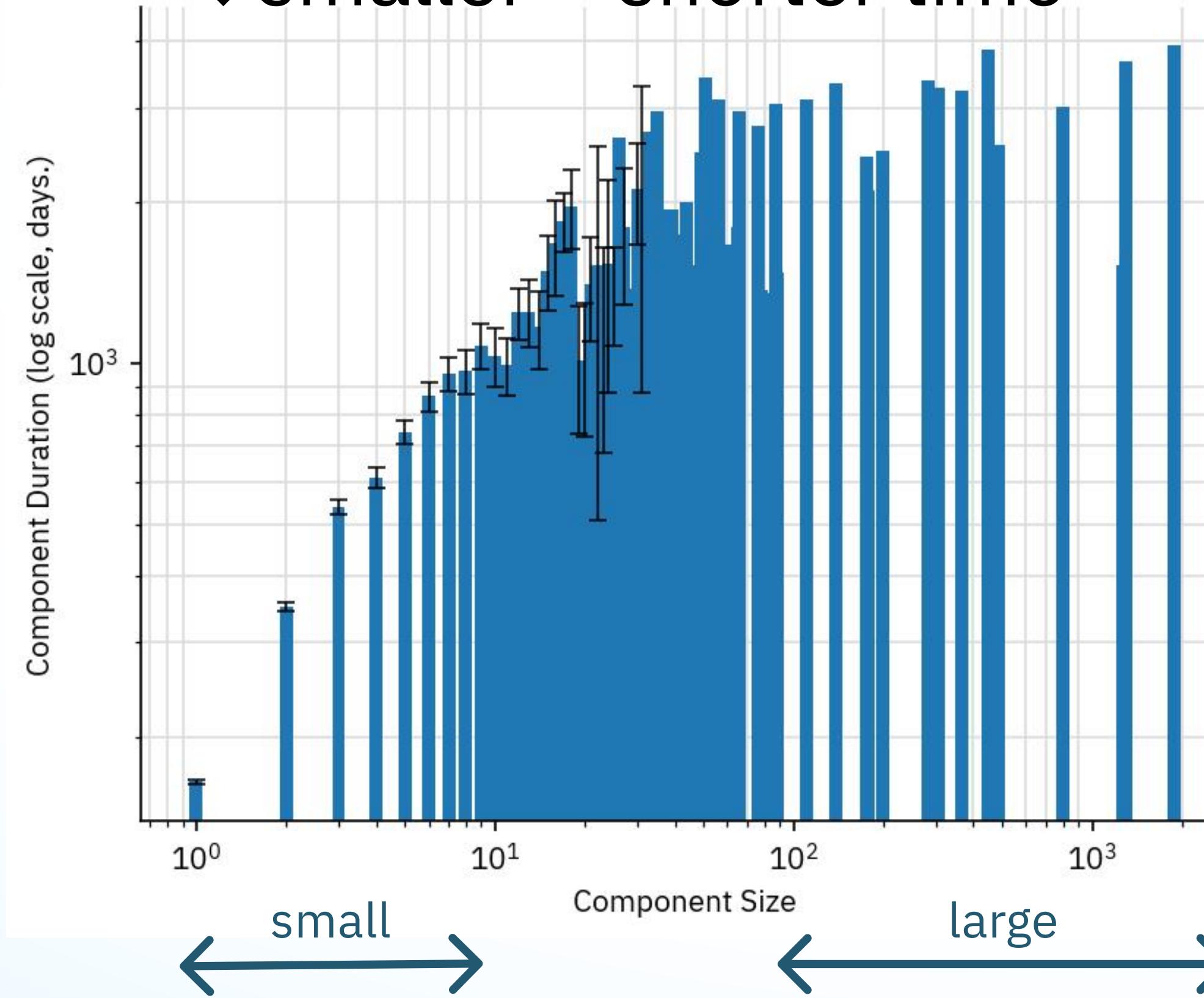
- More Issues
- Fewer **closed** Issues (red)

# Different cluster sizes, different types of work.



# Different cluster sizes, different types of work.

↓ smaller = shorter time



Small clusters have:

- More Issues
- Fewer closed Issues
- Fewer merged PRs
- Shorter timespan between first/last update

# Different cluster sizes, different types of work.

- |   |   |
|---|---|
| Big (>100 node) clusters have:  | Small (<10 node) clusters have:   |
| <ul style="list-style-type: none"><li>• Fewer Issues</li><li>• More closed Issues</li><li>• More merged PRs</li><li>• Longer timespan between first/last update</li><li>• = <b>long-term, mature work</b></li></ul> | <ul style="list-style-type: none"><li>• More Issues</li><li>• Fewer closed Issues</li><li>• Fewer merged PRs</li><li>• Shorter timespan between first/last update</li><li>• = <b>one-off contribution, WIPs</b></li></ul> |

# Examining a cluster

*From Rapptz/discord.py:*

Rewrite: please add a `invoked_parent` feature. #1874

Closed

ribosomerocker opened this issue on Feb 4, 2019 · 4 comments · Fixed by #6462

Adds an `invoked_parents` to context #6453

Closed

sudosnok wants to merge 3 commits into `Rapptz:master` from `sudosnok:patch/invoked-parents`



sudosnok commented on Feb 21, 2021

## Summary

This adds an `invoked_parents` property allowing easy access to the aliases of the parent  
(Group : str)

Closes #1874



Rapptz commented on Feb 21, 2021

I'm definitely not a fan of the way this implementation was done.

The way I imagined this attribute was similar to how `invoked_with` and `parents` are done  
attribute within the context. As it stands right now this walks over the view a second time!



Rapptz closed this on Feb 21, 2021

[commands] Add Context.`invoked_parents` #6462

Merged

Rapptz merged 2 commits into `Rapptz:master` from `SebbyLaw:feature/invoked-parents` on Feb 23, 2021



SebbyLaw commented on Feb 22, 2021

## Summary

Adds `Context.invoked_parents`, resolves #1874

Also resolves #6461 which I found during testing, and including the commit here because it's easier to merge.

This PR can be merged without squashing.



Conversation

0



Commits

2



Checks

0



Files changed

2

# Examining a cluster

*From Rapptz/discord.py:*

Rewrite: please add a `invoked_parent` feature. #1874

Closed

ribosomerocker opened this issue on Feb 4, 2019 · 4 comments · Fixed by #6462

Adds an `invoked_parents` to context #6453

Closed

sudosnok wants to merge 3 commits into `Rapptz:master` from `sudosnok:patch/invoked-parents`



sudosnok commented on Feb 21, 2021

## Summary

This adds an `invoked_parents` property allowing easy access to the aliases of the parent  
(Group : str)

Closes #1874

← “fixes” link type



Rapptz commented on Feb 21, 2021

I'm definitely not a fan of the way this implementation was done.

The way I imagined this attribute was similar to how `invoked_with` and `parents` are done  
attribute within the context. As it stands right now this walks over the view a second time!



Rapptz closed this on Feb 21, 2021

[commands] Add Context.`invoked_parents` #6462

Merged

Rapptz merged 2 commits into `Rapptz:master` from `SebbyLaw:feature/invoked-parents` on Feb 23, 2021



SebbyLaw commented on Feb 22, 2021

## Summary

Adds `Context.invoked_parents`, resolves #1874

Also resolves #6461 which I found during testing, and including the commit here because it's easier to merge.

This PR can be merged without squashing.

↓ another keyword

# Examining a cluster

*From Rapptz/discord.py:*

Rewrite: please add a `invoked_parent` feature. #1874

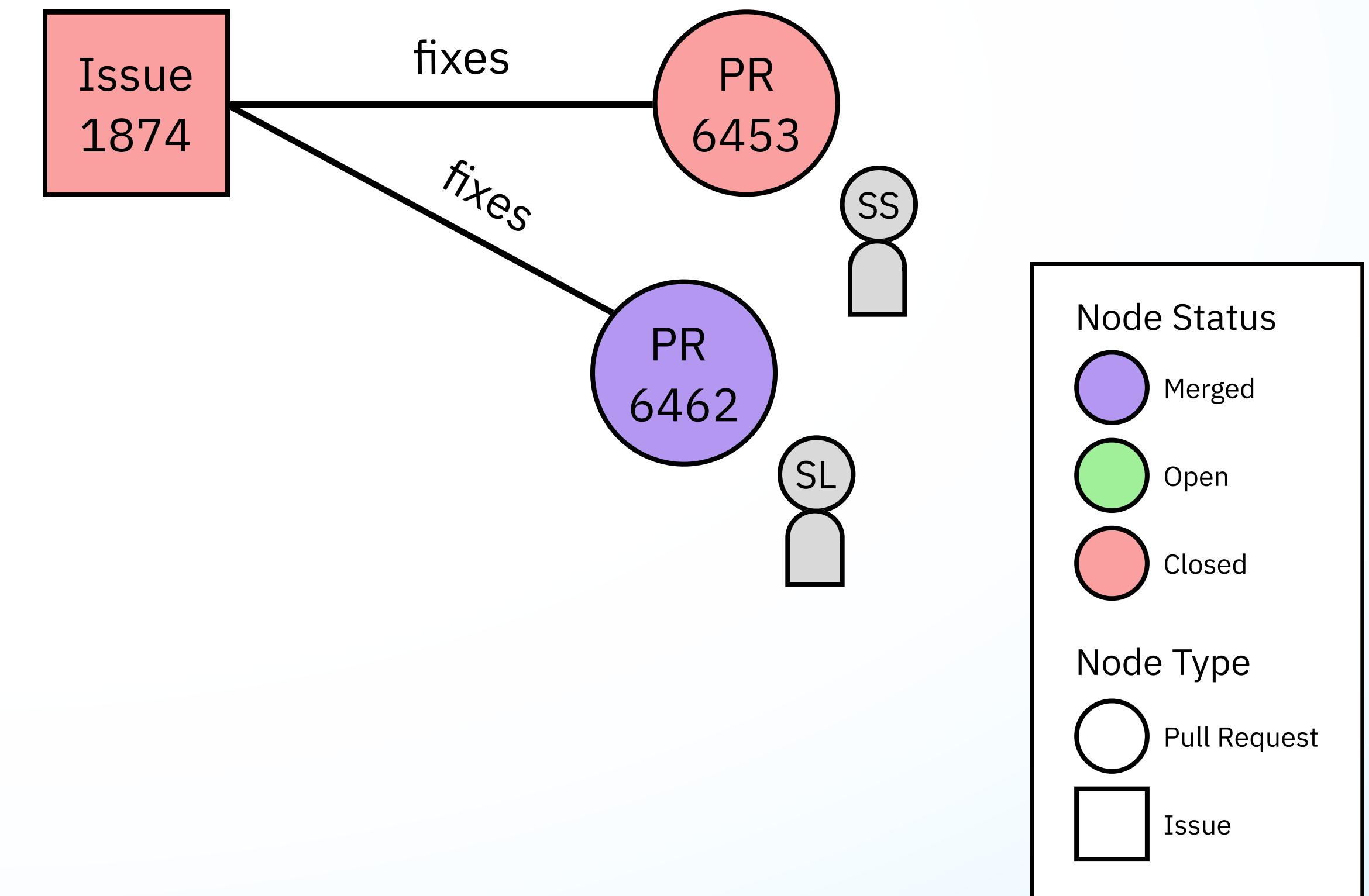
Closed ribosomerocker opened this issue on Feb 4, 2019 · 4 comments - Fixed by #6462

Adds an `invoked_parents` to context #6453

Closed sudosnok wants to merge 3 commits into `Rapptz:master` from `sudosnok:patch/invoked-parents`

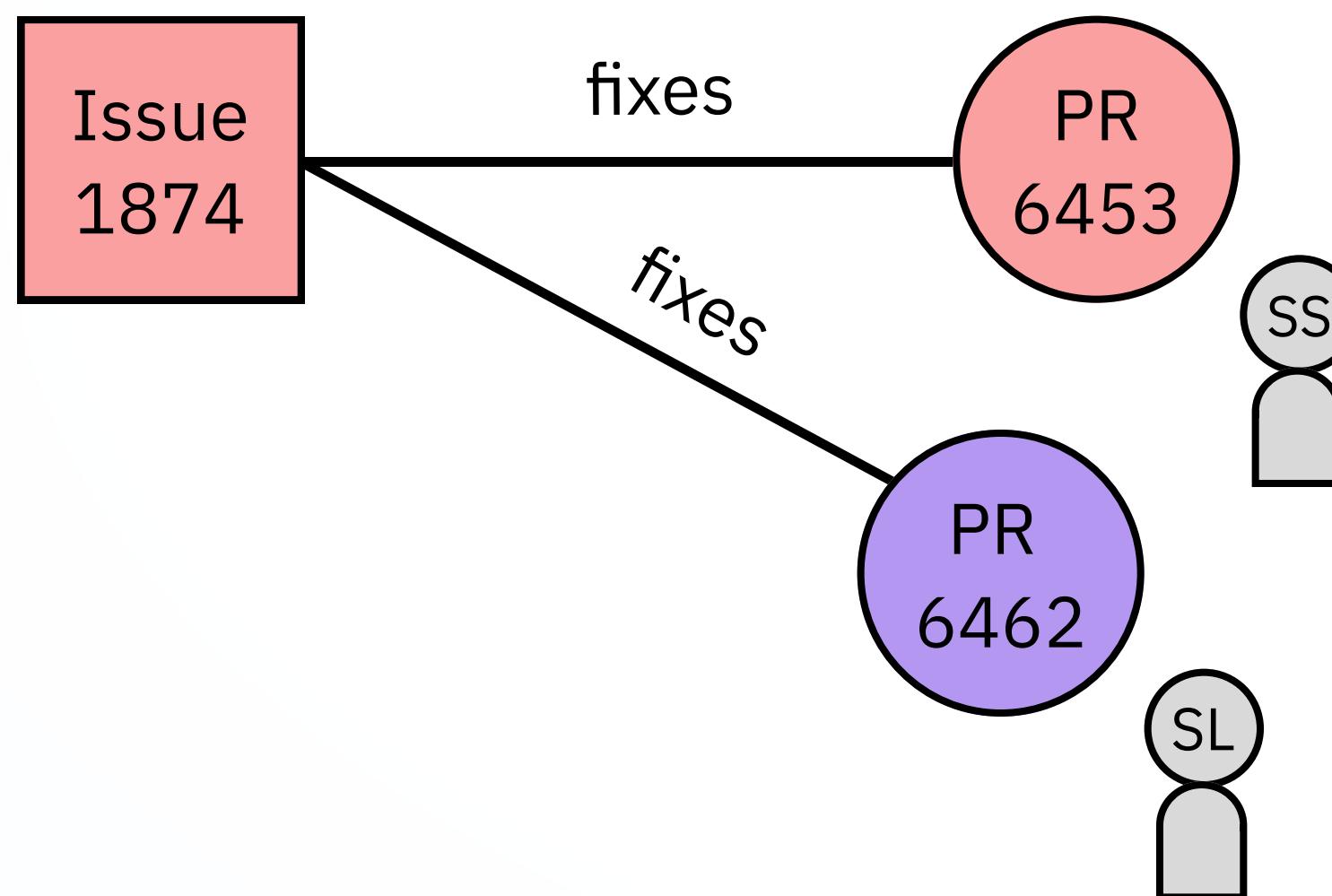
[commands] Add Context.invoked\_parents #6462

Merged Rapptz merged 2 commits into `Rapptz:master` from `SebbyLaw:feature/invoked-parents` on Feb 2

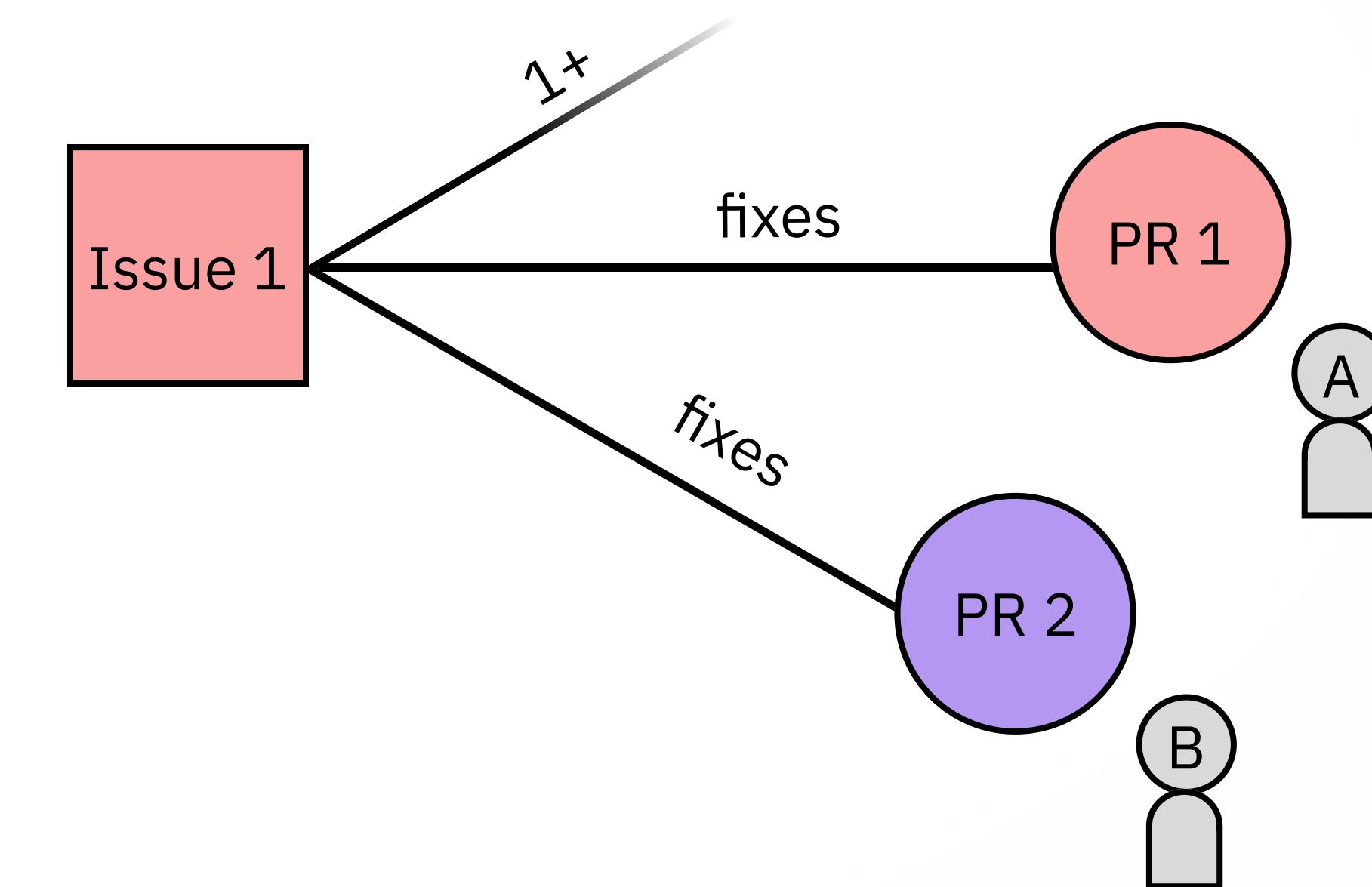


# Examining a cluster

*From Rapptz/discord.py:*

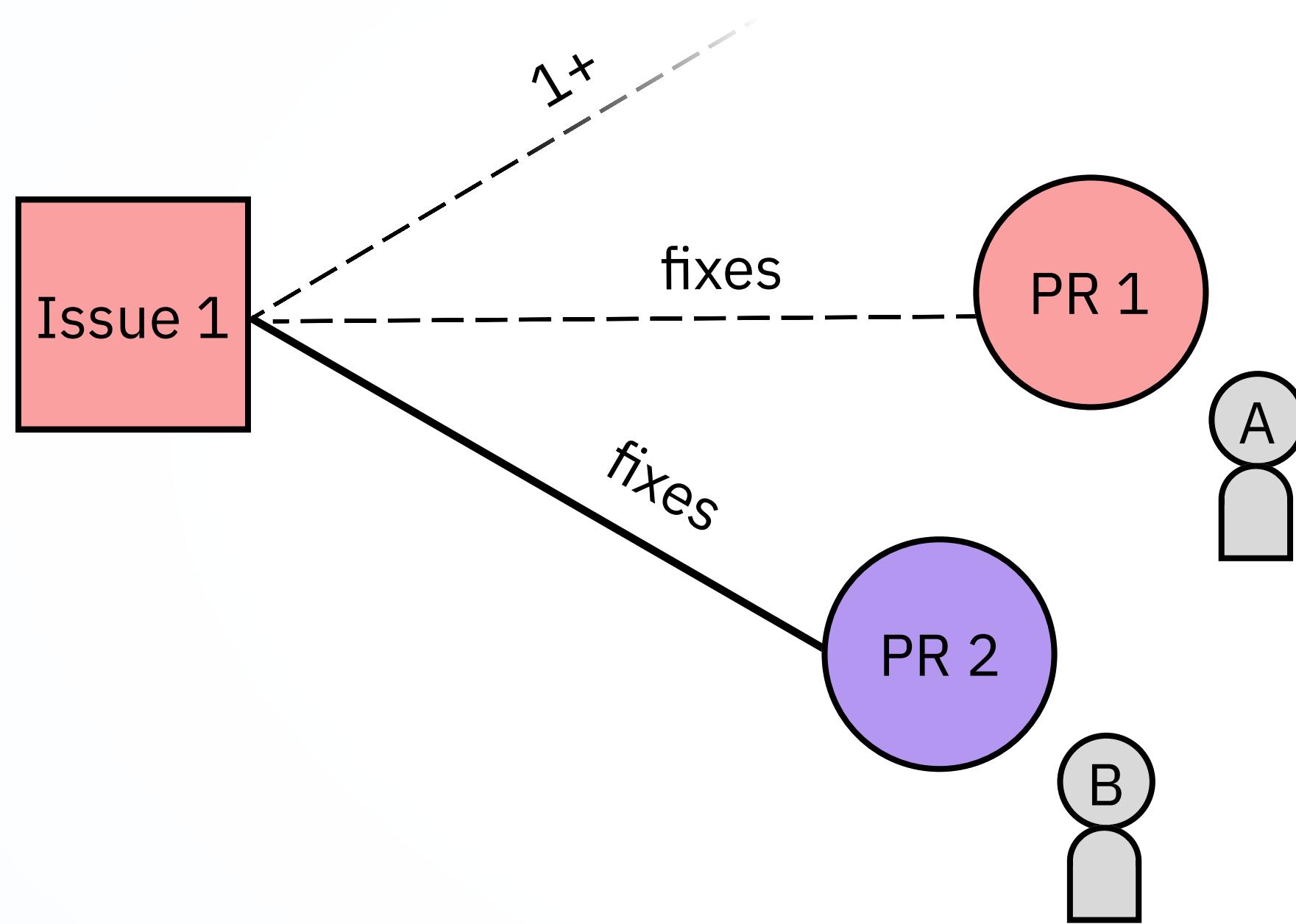


Competing PRs workflow type:



Node Status	
	Merged
	Open
	Closed
Node Type	
	Pull Request
	Issue

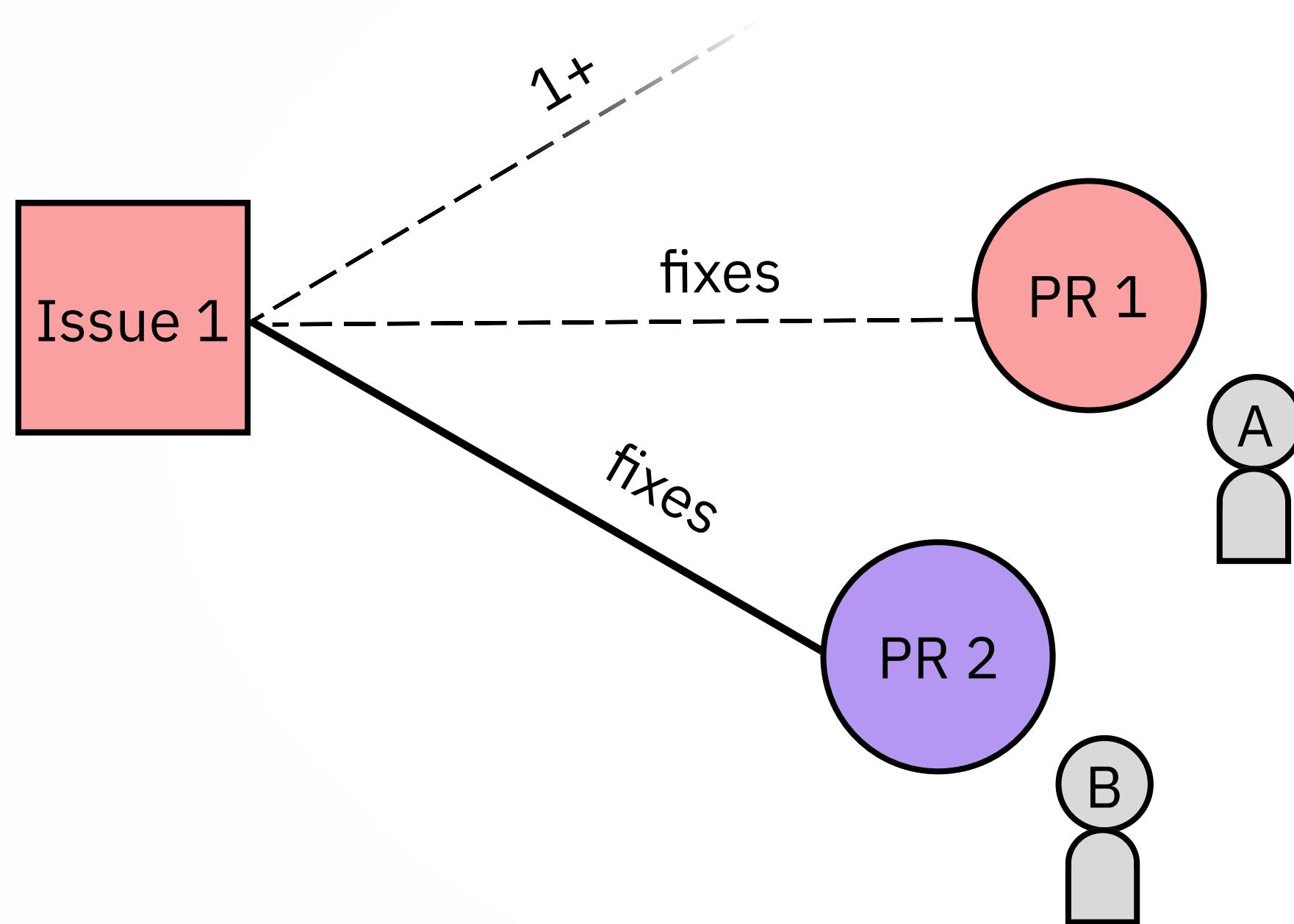
# Workflow type definitions



- associated with work practices

*Example: Competing PRs*

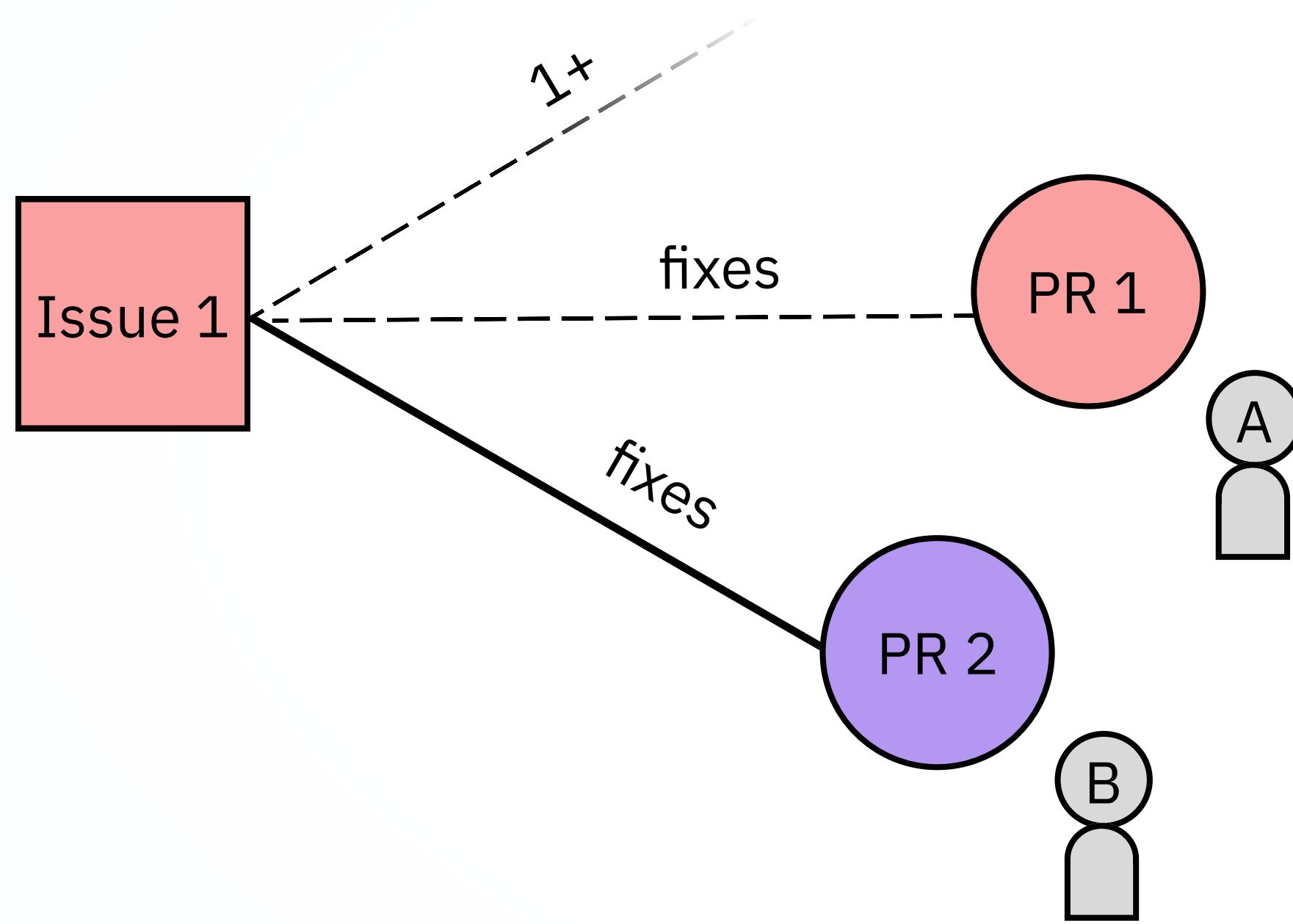
# Workflow type definitions



- associated with work practices
- has prototypical graph structure

*Example: Competing PRs*

# Workflow type definitions

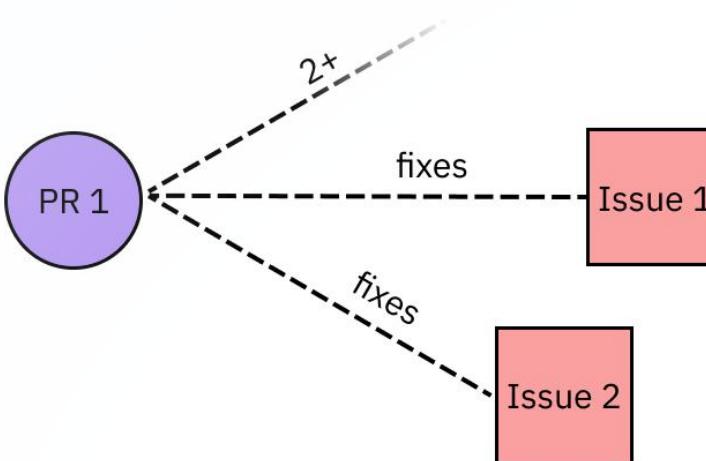


- associated with work practices
- has prototypical graph structure
- described as a combination of some constraints (e.g. an Issue/PR with some given status fixes another Issue/PR)

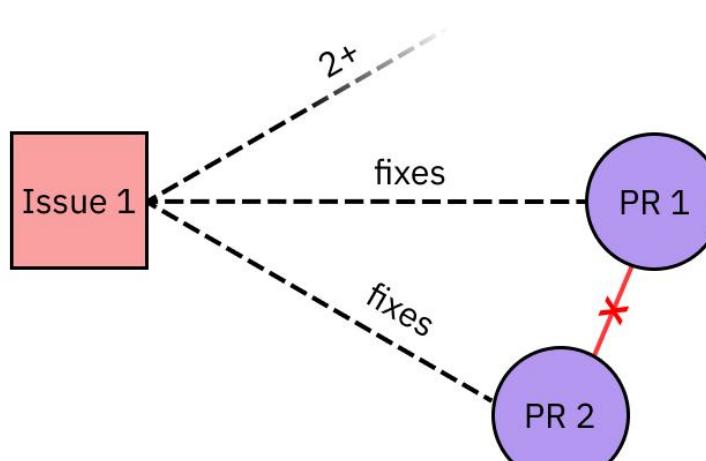
*Example: Competing PRs*

# Workflow type definitions

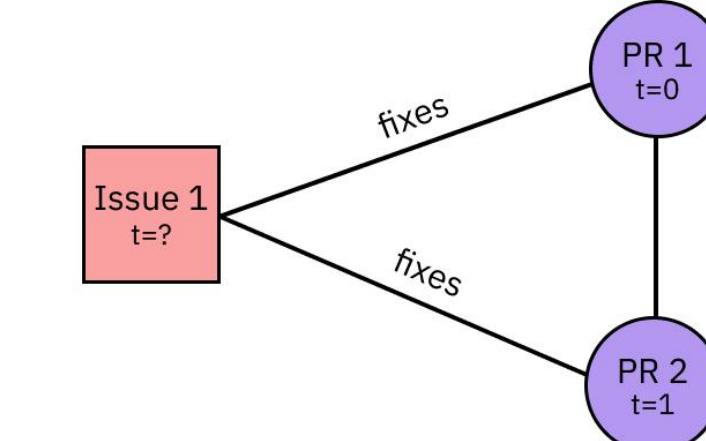
Divergent PR



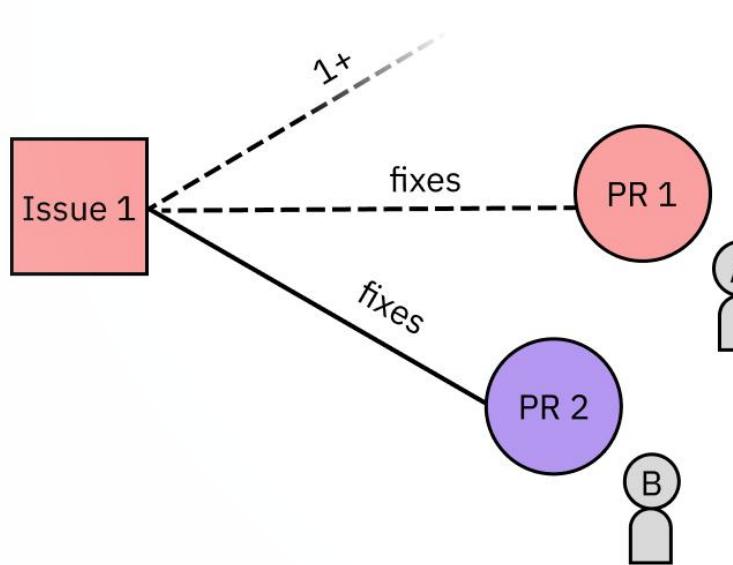
Decomposed Issue



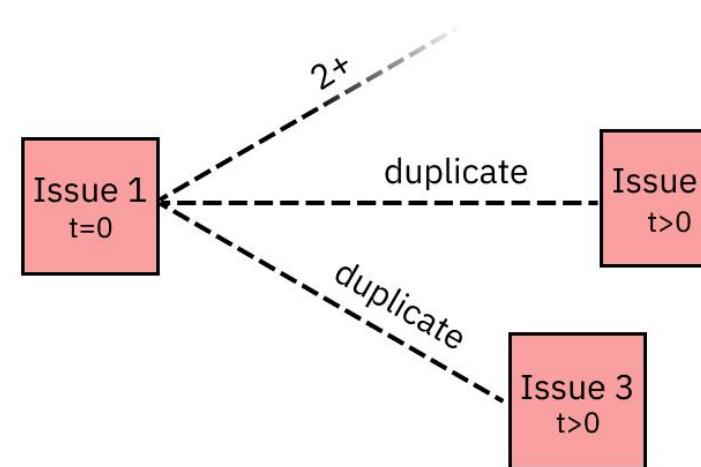
Extended PR



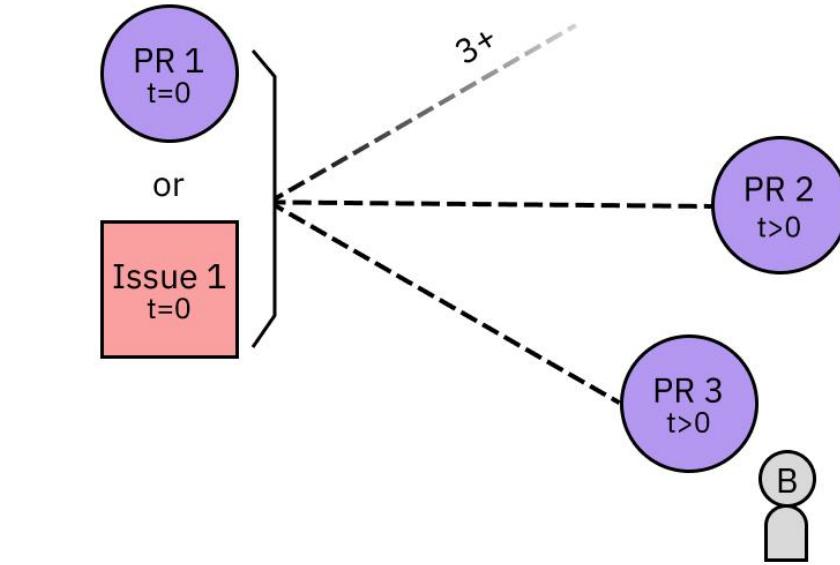
Competing PRs



Duplicate Issue Hub

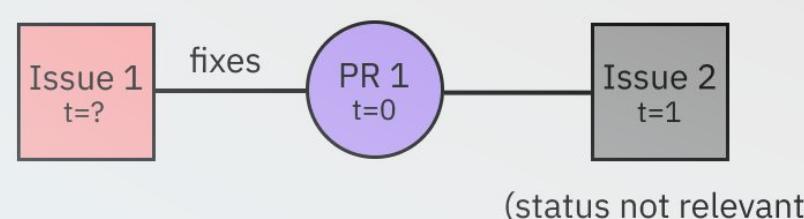


Integrating PR/Issue Hub

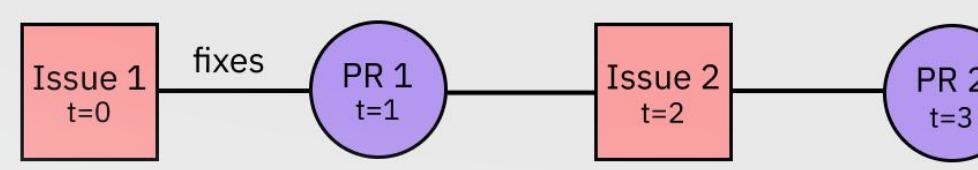


Consequence

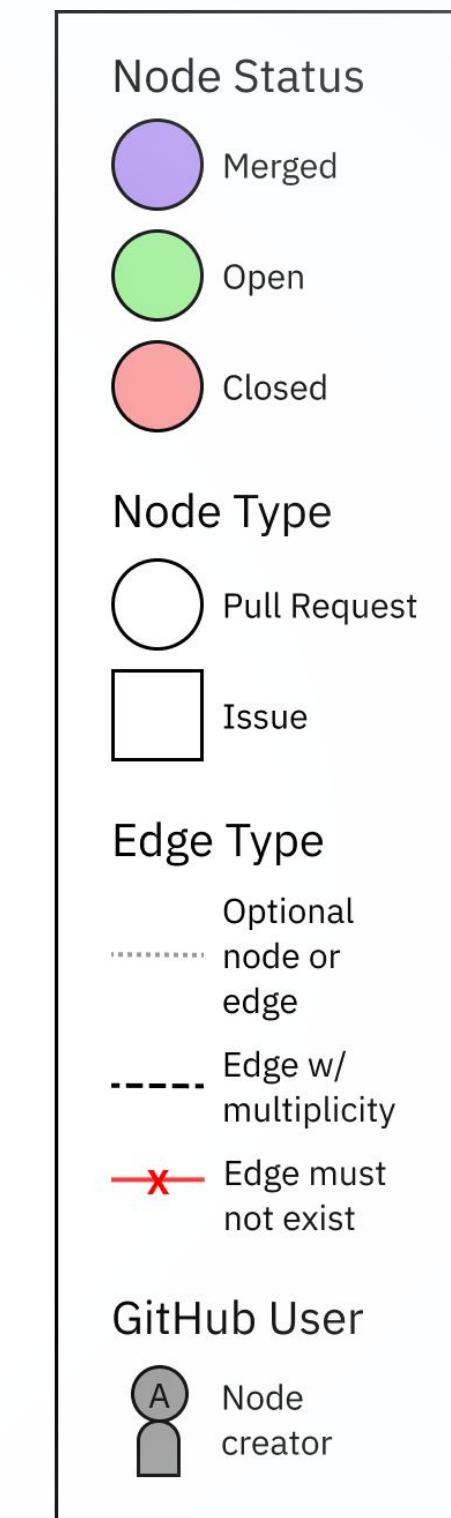
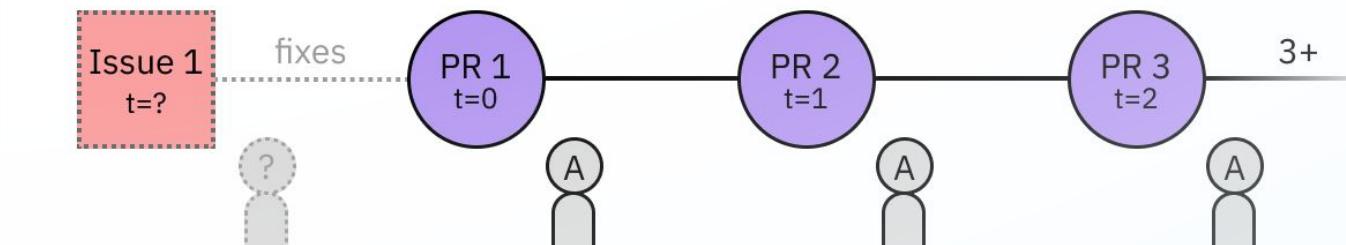
Consequent Issue



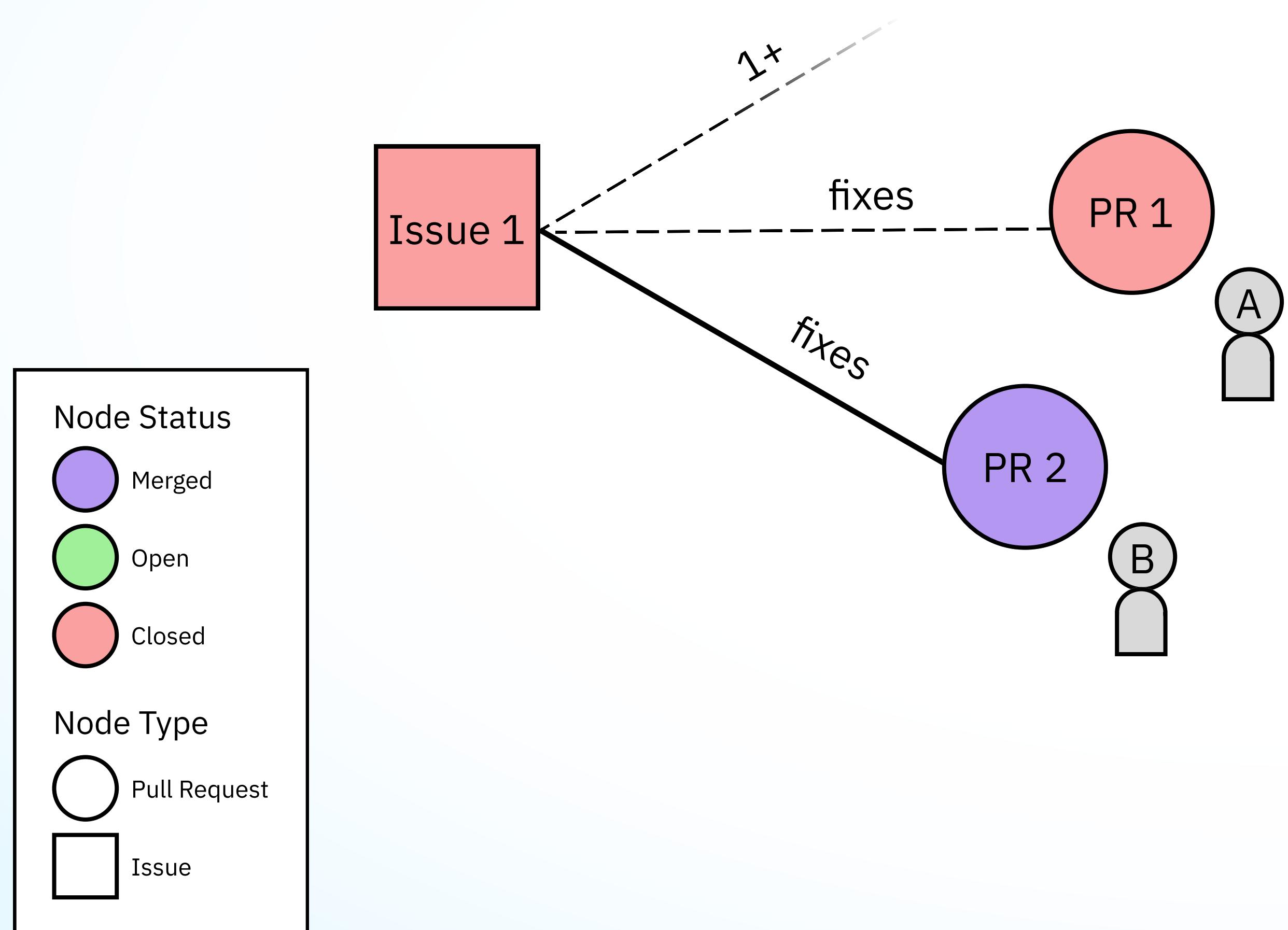
Consequent Issue-PR



Dependent PRs

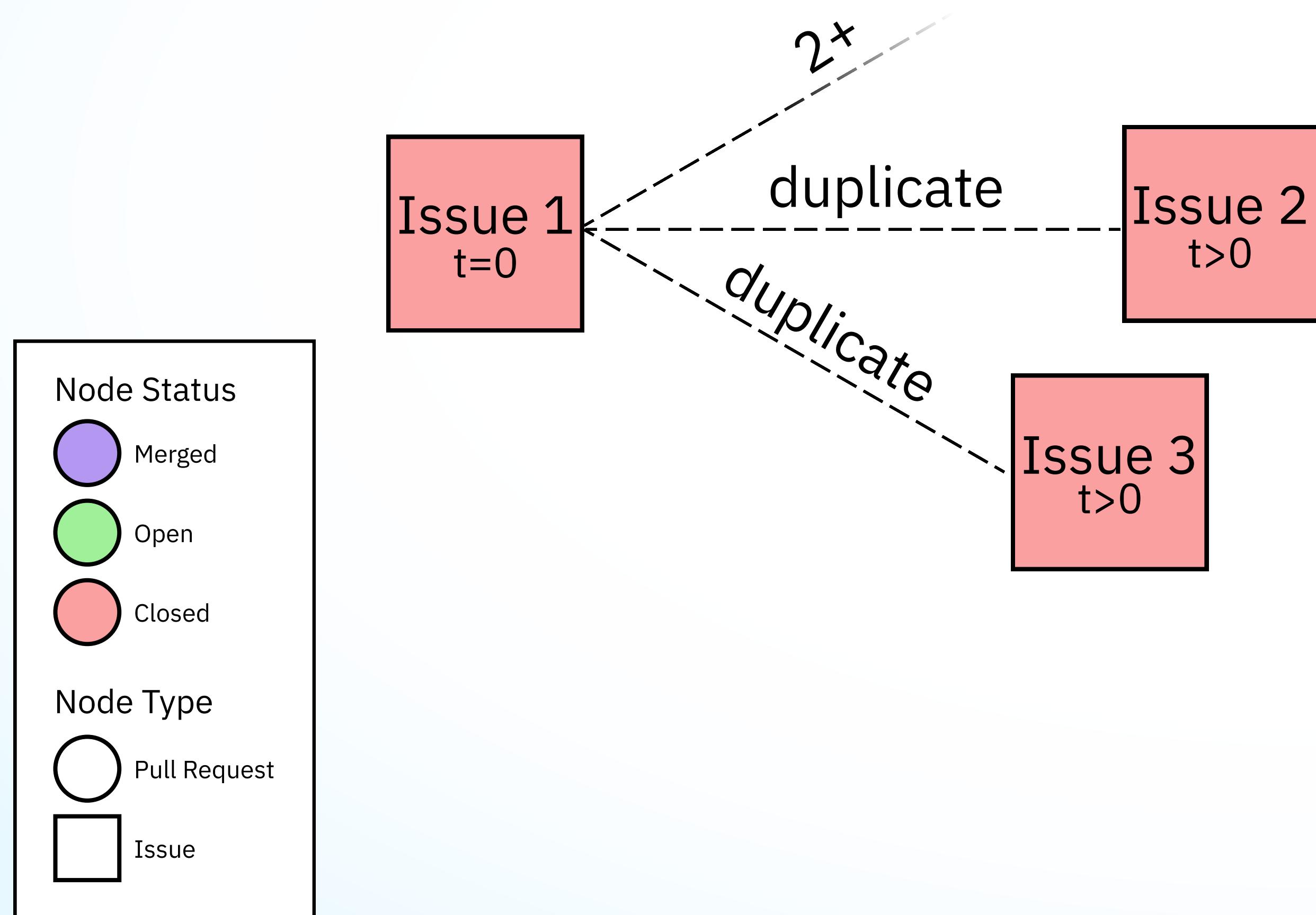


# Competing PRs



- one Issue connected to multiple PRs
- only one PR merged
- “wasted” work due to poor communication
- alternatively: finding the “best” solution

# Duplicate Issue Hub



- one Issue with multiple outgoing ‘duplicate’ Issue links
- contributors not being aware of other work
- additional maintenance burden
- not as frequent as you’d think!

# Duplicate Issue Hub

*From Rapptz/discord.py:*

Node Status	
	Merged
	Open
	Closed
Node Type	
	Pull Request
	Issue



# Integrating PR/Issue Hub

*From apache/dubbo:*

[WeeklyReport] Weekly report for Dubbo 2019-5-31 to 2019-6-7 #4266

 Closed dubbo-bot opened this issue on Jun 7, 2019 · 0 comments

## Repo Overview

### Basic data

Basic data shows how the watch, star, fork and contributors count changed in the passed week.

Watch	Star	Fork	Contributors
3217	27013 (1191)	15166 (197)	201 (11)

### Issues & PRs

Issues & PRs show the new/closed issues/pull requests count in the passed week.

New Issues	Closed Issues	New PR	Merged PR
20	16	25	24

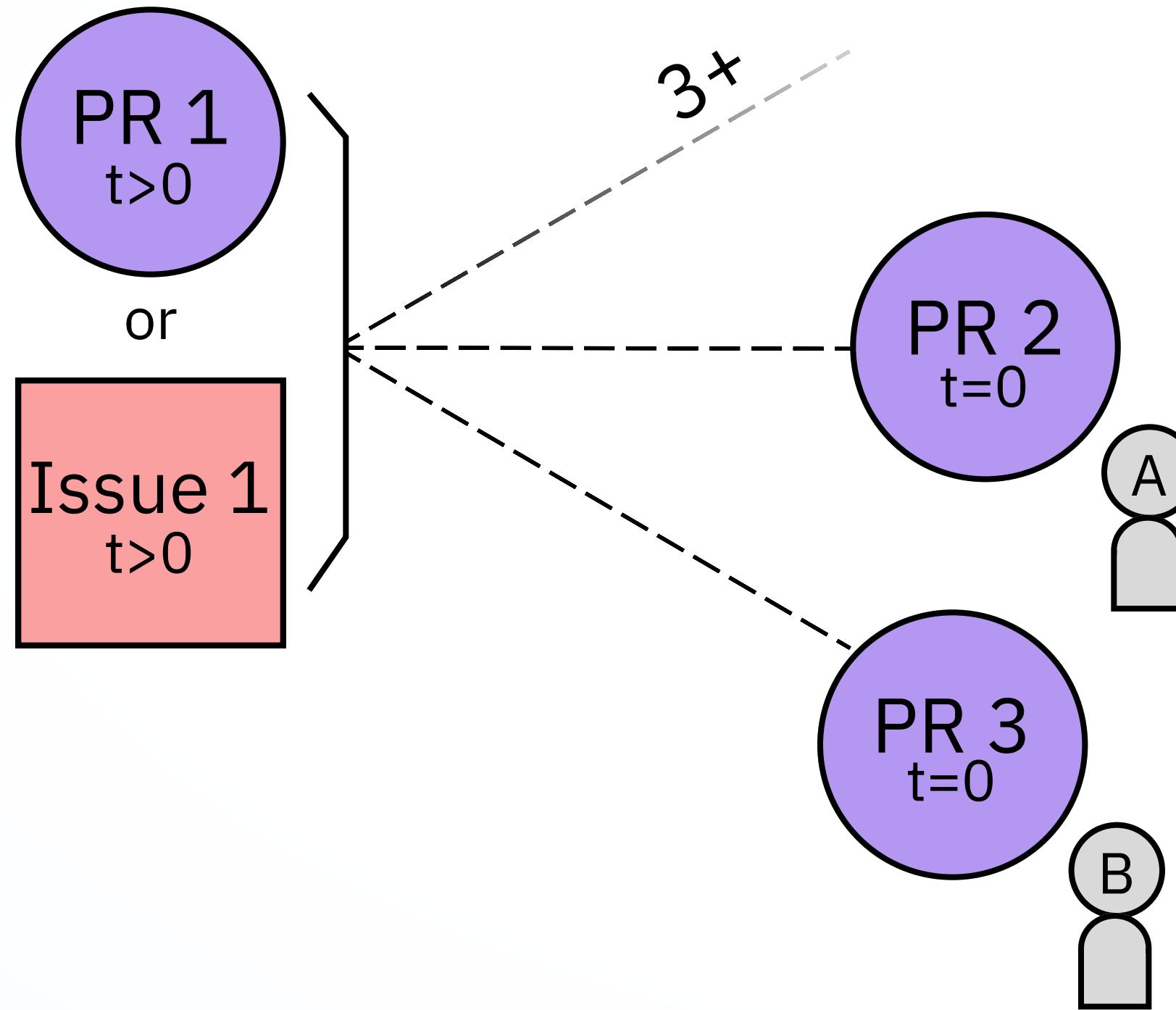
## PR Overview

Thanks to contributions from community, Dubbo team merged 24 pull requests.

- 2.7.2 release ([#4259](#))
- Default future from jeff-lv ([#4257](#))
- performance tuning: cache service key ([#4255](#))
- performance tuning: channel connect status ([#4254](#))
- Performance tuning: keep heartbeat updated in io thread to avoid race ([#4249](#))
- Applicationmodel revert ([#4245](#))
- getMethodparameter refactor ([#4244](#))
- delete useless code ([#4242](#))

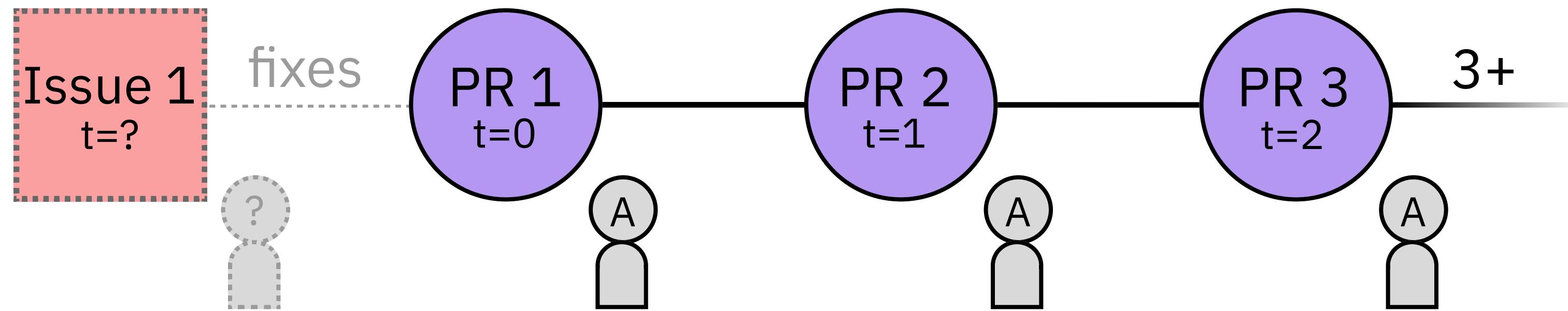
# Integrating PR/Issue Hub

Node Status	
Circle	Merged
Light Purple	Merged
Green	Open
Red	Closed
Node Type	
Circle	Pull Request
Box	Issue



- Issue or PR connected to many merged PRs
- central Issue/PR created after merged PRs
- release or documentation workflows

# Dependent PRs



Node Status



Merged



Open



Closed

Node Type



Pull Request



Issue

- stacked PRs workflow: breaking work up into small, shippable changes
  - series of PRs optionally starting with Issue
  - created by same (few) authors

# Divergent PR

*From grpc/grpc-web:*

Fixes #848: Fixed a bug where we can't pass the interceptors into the client constructor because of a type mismatch.

Fixes #868: Added `MethodDescriptor` to the exported types. Replace the deprecated `MethodInfo` with `MethodDescriptor`.

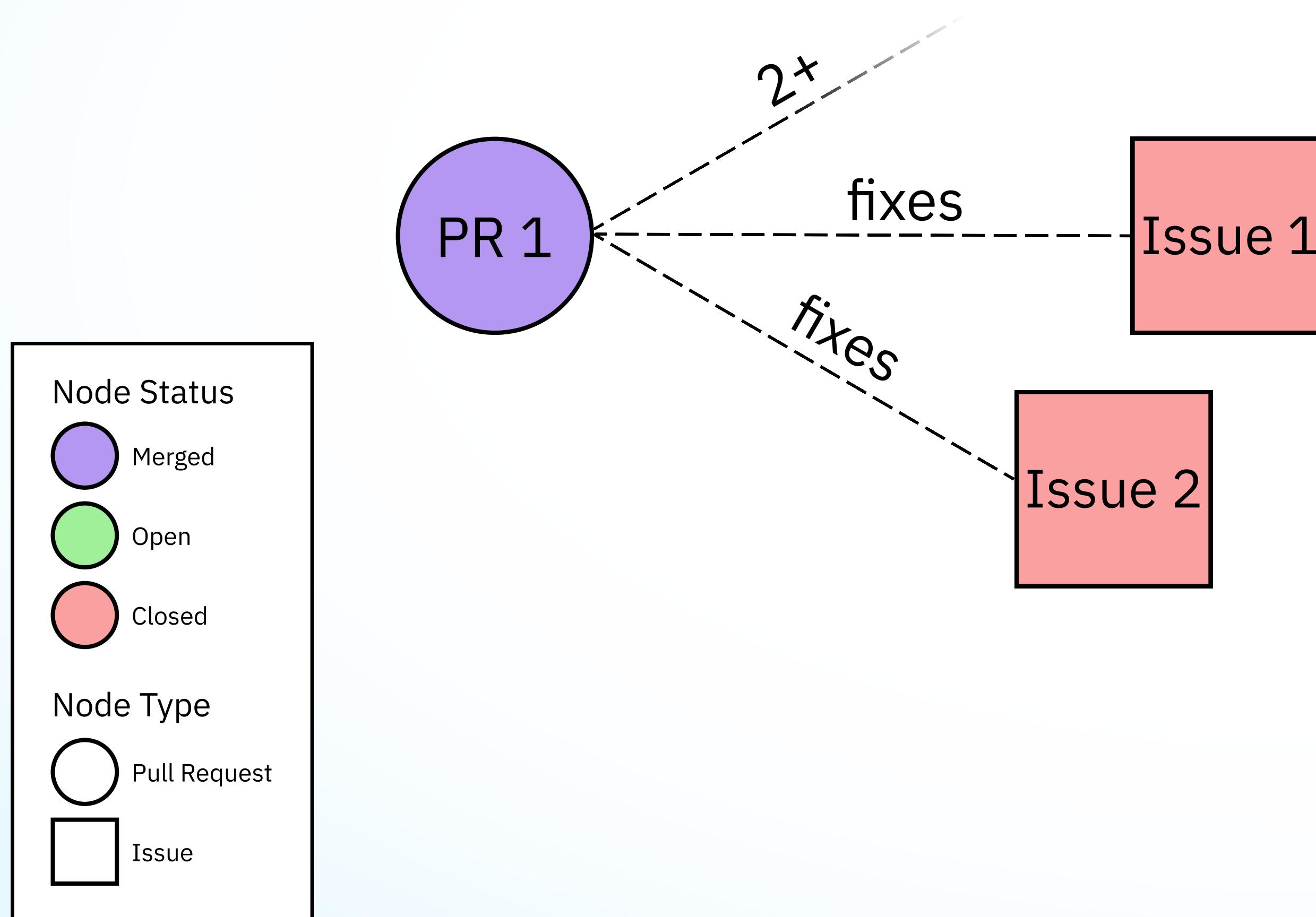
Fixes #877: Fixed issue where `UnaryResponse` symbols are being optimized away.

And in general added some missing classes to the exported types as well.

Added tests for both callback-based `StreamInterceptor` and promise-based `UnaryInterceptor`. And now execute `tsc` with `--strict`.

↑ negative, because unrelated changes in same PR = hard to review

# Divergent PR

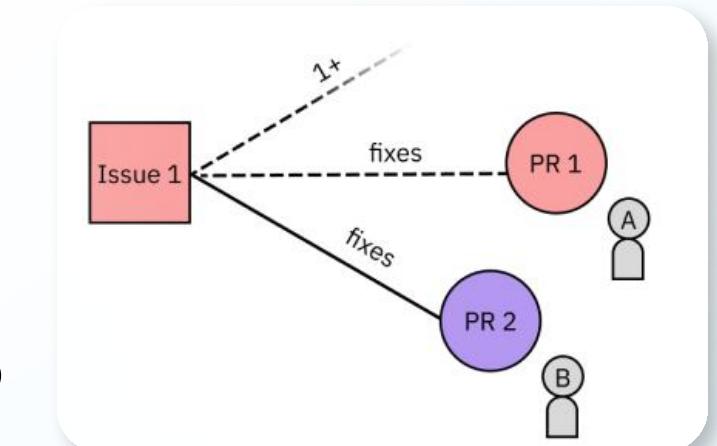
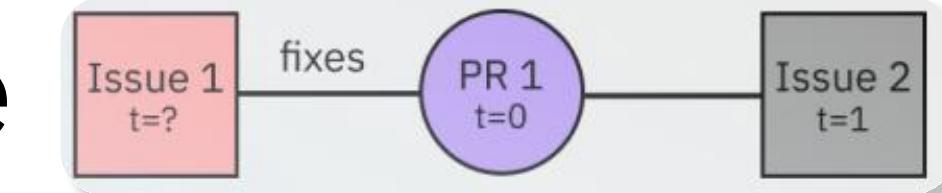


- merged PR fixing multiple Issues
- can be both positive or negative, depending if Issues are related and linked

# Some workflow types are more popular than others.

Workflow Type	Num. Matches
Competing PRs	15
Consequent Issue	210
Consequent Issue-PR	471
Decomposed Issue	91
Dependent PRs	52
Divergent PR	118
Duplicate Issue Hub	45
Extended PR	14
Integrating PR/Issue Hub	94

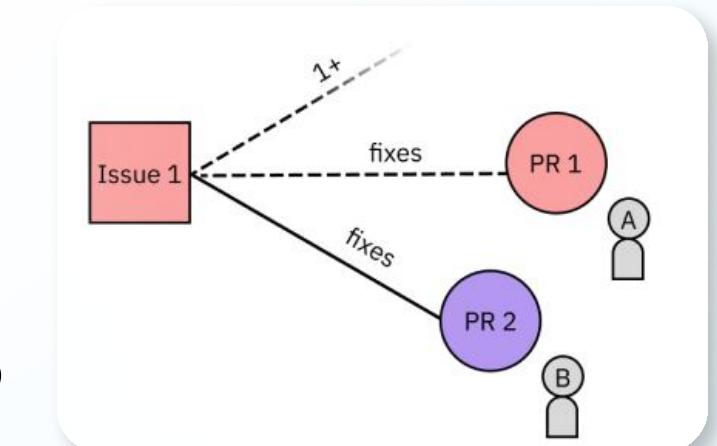
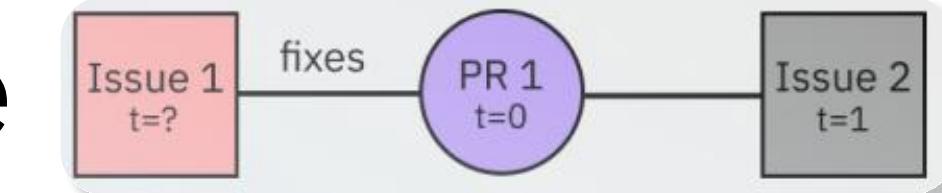
- Consequent Issue-PR most popular, followed by very similar Consequent Issue
- Very few Competing PRs or Extended PR matches → constraints or alternatives



# Some workflow types are more popular than others.

Workflow Type	Num. Matches
Competing PRs	15
<b>Consequent Issue</b>	<b>210</b>
<b>Consequent Issue-PR</b>	<b>471</b>
Decomposed Issue	91
Dependent PRs	52
Divergent PR	118
Duplicate Issue Hub	45
Extended PR	14
Integrating PR/Issue Hub	94

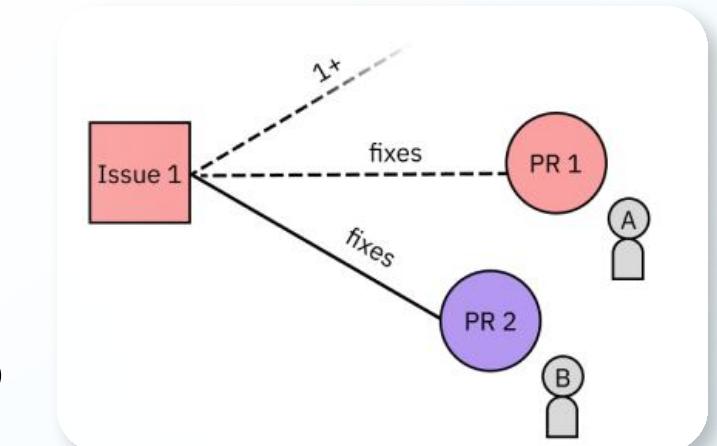
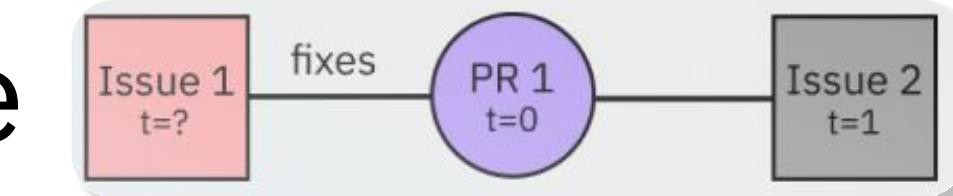
- Consequent Issue-PR most popular, followed by very similar Consequent Issue
- Very few Competing PRs or Extended PR matches → constraints or alternatives



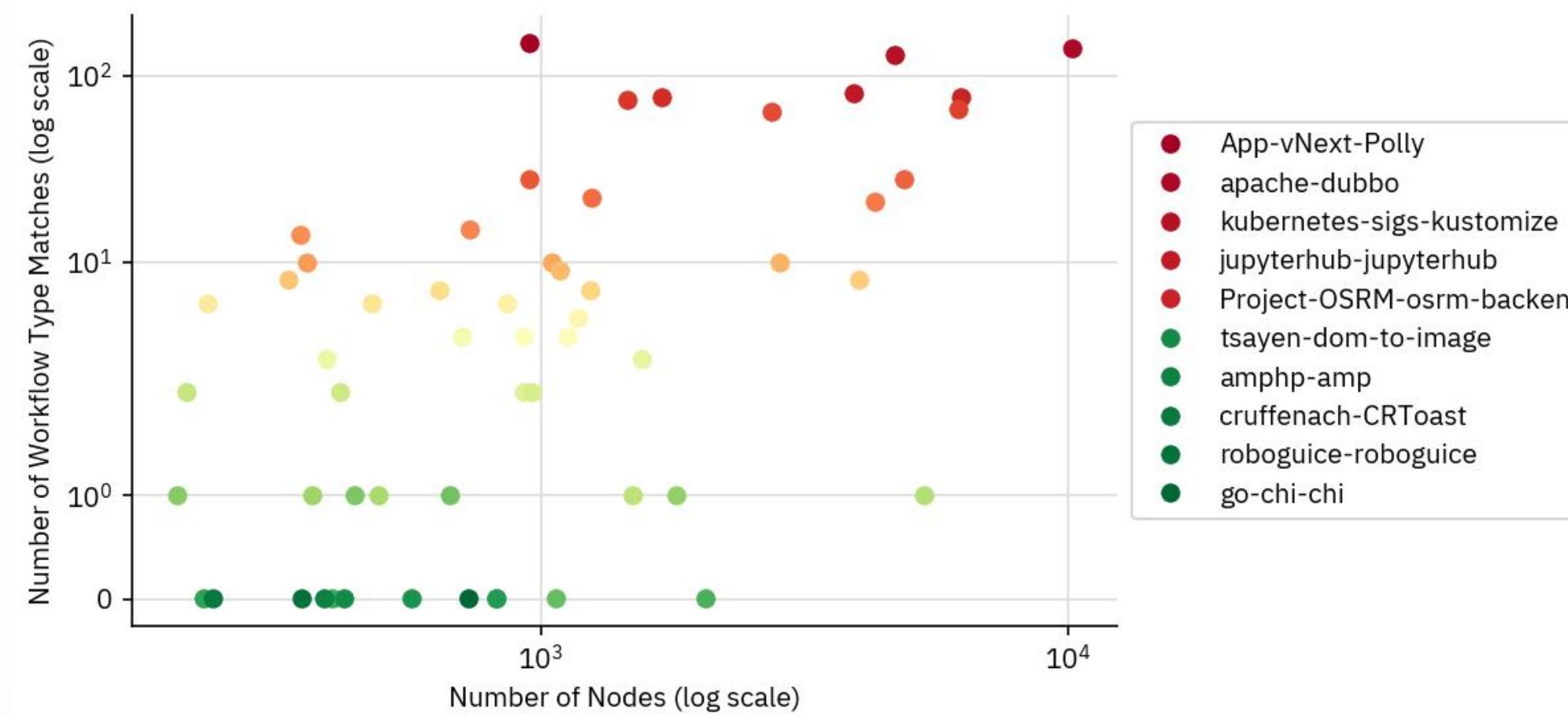
# Some workflow types are more popular than others.

Workflow Type	Num. Matches
Competing PRs	15
Consequent Issue	210
Consequent Issue-PR	471
Decomposed Issue	91
Dependent PRs	52
Divergent PR	118
Duplicate Issue Hub	45
Extended PR	14
Integrating PR/Issue Hub	94

- Consequent Issue-PR most popular, followed by very similar Consequent Issue
- Very few Competing PRs or Extended PR matches → constraints or alternatives



# Workflow types represent projects fairly well.



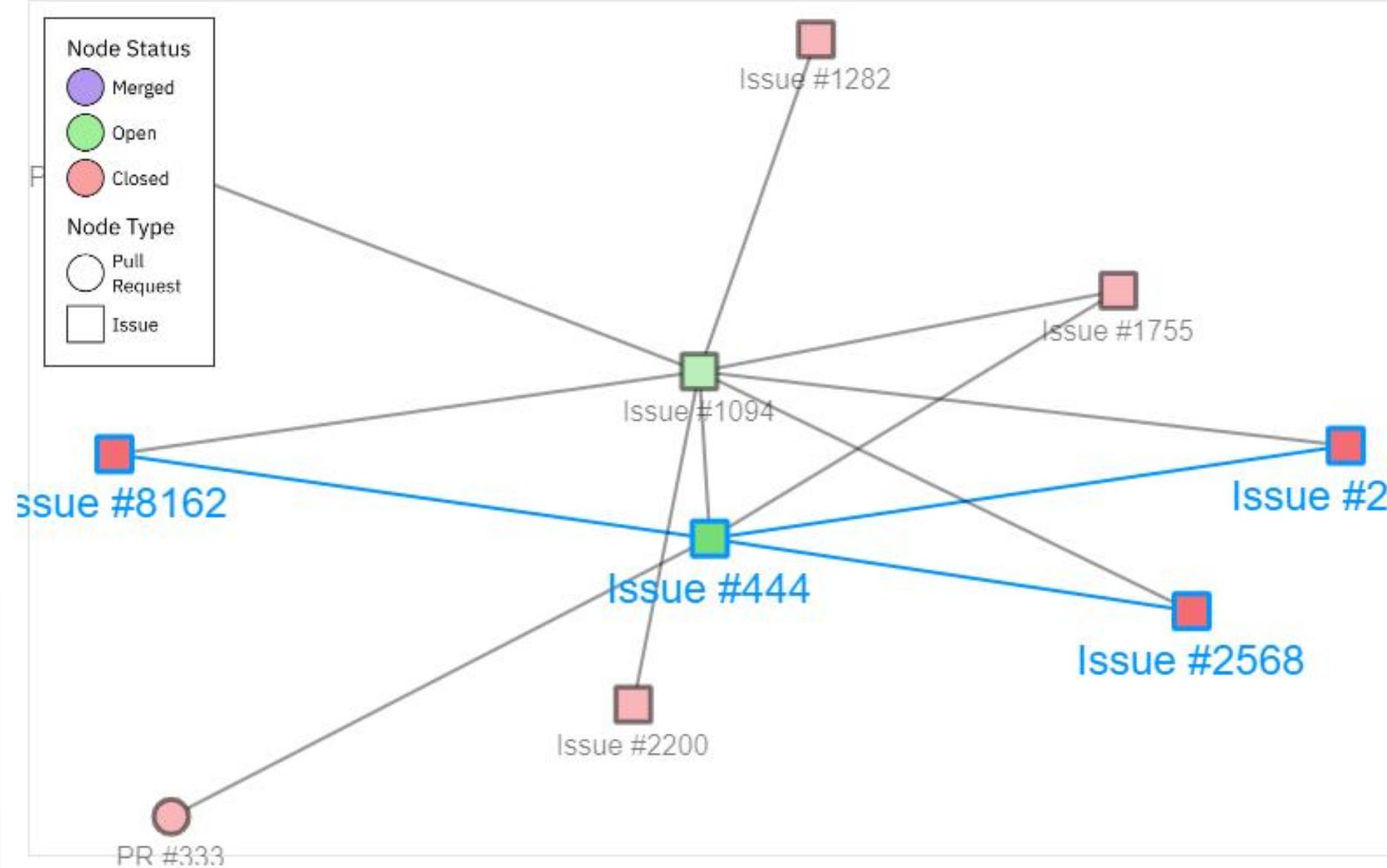
- 52% of nodes meeting basic criteria are in a match
- Most projects contain at least one match, though frequencies differ per repo → maturity
- 12 small projects contain no workflow type matches

# What did developers think?

## Rapptz/discord.py Workflow Type Matches

Verified workflow type matches:

duplicate\_issue\_hub\_3



- 6 interviews: 3 with maintainers of projects we studied; 3 with engineers using GitHub in a closed-source context
- centred around WorkflowsExplorer tool customized for their project

# What did developers think?

- Help make better decisions in development process
  - Duplicate Issue Hubs → change in documentation or bug reporting process
  - Divergent PR → prioritize review

# What did developers think?

- Help make better decisions in development process
  - Duplicate Issue Hubs → change in documentation or bug reporting process
  - Divergent PR → prioritize review
- Help understand inter-dependencies visually

# What did developers think?

- Help make better decisions in development process
  - Duplicate Issue Hubs → change in documentation or bug reporting process
  - Divergent PR → prioritize review
- Help understand inter-dependencies visually
- Are limited due to not examining contents of Issue or PR

# Implications

- Can inform prioritization

# Implications

- Can inform prioritization
- Can improve code review/PM practices

# Implications

- Can inform prioritization
- Can improve code review/PM practices
- Can identify areas of wasted work

# Implications

- Can inform prioritization
- Can improve code review/PM practices
- Can identify areas of wasted work
- Can extend duplicate/first issue detection tools (Hirao et al., 2019)

(“The review linkage graph for code review analytics...”, Hirao et al., ESEC/FSE 2019)

# Implications

- Can inform prioritization
- Can improve code review/PM practices
- Can identify areas of wasted work
- Can extend duplicate/first issue detection tools (Hirao et al., 2019)
- Next: talking with more maintainers like you!

(“The review linkage graph for code review analytics...”, Hirao et al., ESEC/FSE 2019)

# Conclusion

- Novel graph-based perspective
- Workflow type definitions and attributes
- Automatable queries and interactive explorer tools
- TL;DR: Grafana for collab health

Reach us at [cleidson.desouza@acm.org](mailto:cleidson.desouza@acm.org),  
reach me at [hello@emilie.ma](mailto:hello@emilie.ma).

*Slides available at [emilie.ma/ossna2024](http://emilie.ma/ossna2024)*