Almost There: A Study on Quasi-Contributors in Open Source Software Projects

Igor STEINMACHER @igorsteinmacher
Gustavo PINTO @gustavopinto
Igor WIESE @igorwiese
Marco GEROSA @gerosa_marco
Sustainable OSS projects
Sustainability of Open Source software communities beyond a fork: How and why has the LibreOffice project evolved?

Jonas Camaelason, Björn Lundell

Show more
Understanding Sustained Participation in Open Source Software Projects

Yulin Fang & Derrick Neufeld

Pages 9-50 | Published online 08 Dec 2014

Sustainability of Open Source software communities beyond a fork: How and why has the LibreOffice project evolved?

Jonas Camandelsson, Björn Lundell

Show more

Social Barriers Faced by Newcomers Placing Their First Contribution in Open Source Software Projects

An exploratory study of contribution barriers experienced by newcomers to open source software projects

Full Text: PDF

Authors: Igor Steinmacher, Igor Wiese, Ana Paula Chaves, Marco Aurelio Gerosa

Published in: Proceeding CSI-SE 2014 Proceedings
CrowdSourcing in Software Development
Pages 11-14

Why do newcomers abandon open source software projects?

View Document 16 Paper Citations 209 Full Text Views
Contributing to OSS

Many contributors succeed when they submit changes
Contributing to OSS

Some got some rejected patches
Contributing to OSS
Contributing to OSS

Quasi-contributors
Investigate **how** and **why** quasi-contributors fail
Research Questions

**RQ1.** How common are quasi-contributors and quasi-contributions?

**RQ2.** Why were the quasi-contributions not accepted?

**RQ3.** How do quasi-contributors perceive nonacceptance?
Sampling

angular
bitcoin
bootstrap
caffe
d3
django
docker
flask
jenkins
joomla!
jquery
kubernetes
laravel
mongo
opencv
rails
react
redis
scikit-learn
spring
tensorflow
Mixed-Method Approach

- Quasi-contributors: 5,138
- Integrators: 234

335 answers from quasi-contributors
21 answers from integrators

Manual analysis
263 PRs from quasi-contributors
Mixed-Method Approach

RQ1. How common are quasi-contributors and quasi-contributions?

RQ2. Why were the quasi-contributions not accepted?

RQ3. How do quasi-contributors perceive nonacceptance?

5,138 Quasi-contributors
234 Integrators

335 answers from quasi-contributors
21 answers from integrators

Manual analysis
263 PRs from quasi-contributors
RQ1. How common?

Considering all 21 projects:
→ 10,099 quasi-contributors
→ 14,623 actual contributors
RQ1. How common?

- ~85% → 1 attempt

![Diagram showing distribution of contributors by number of attempts.](image-url)
RQ2. Reasons for non-acceptance
Quasi-contributors' perspective

superseded/duplicated pull-request (52/335)
[Gousios et al., 2014; Zhou et al., 2018]

"The fix I submitted remained unmerged until someone else submitted the exact same fix ... the integrators accepted their (identical) fix and closed mine."

"Other pull-requests fixed the same issues as my pull-requests"

RQ2. Reasons for non-acceptance
Quasi-contributors' perspective

superseded/duplicated pull-request (52/335)
[Gousios et al., 2014; Zhou et al., 2018]

"The fix I submitted remained unmerged until someone else submitted the exact same fix ... the integrators accepted their (identical) fix and closed mine."

"Other pull-requests fixed the same issues as my pull-requests"

"when you add a new feature to the project, your vision can be out of tune with the vision of the project’s team, and this is natural."

"The project decided that was not a bug they wanted to provide a fix for"

mismatch between developer's and team's vision/opinion (47/335)
RQ2. Reasons for non-acceptance

**Quasi-contributors’ perspective**

- Lack of interest from integrators (37/335)
- "I did not receive answers"
- PR not needed/not relevant (25/335)
- wasn’t important enough to warrant merging
- The process is too onerous, and bureaucratic.
- Quasi-contributors also offered mea culpa:
  - → Not an optimal solution (20)
  - → Contributors' lack of experience/commitment (24)
RQ2. Reasons for non-acceptance

From integrators perspective:
- PR not needed/not relevant (10/21)
- guidelines not followed (9/21)

"Trivial PRs that are more trouble than they are worth"

"proposed changes do not have real value"

"Does not adhere to internal style or design guidance"
RQ2. Reasons for non-acceptance

Manual analysis
263 PRs from quasi-contributors

Superseded/duplicated
(32/263)

bbonev commented on Jun 18, 2013

I now see that similar change is already proposed by others. Sorry for the noise
RQ2. Reasons for non-acceptance

- Superseded/duplicated (32/263)
- Lack of experience/commitment (25/263)

Manual analysis
263 PRs from quasi-contributors
RQ2. Reasons for non-acceptance

Manual analysis
263 PRs from quasi-contributors

- Superseded/duplicated (32/263)
- Lack of experience/commitment (25/263)
- PR not needed/not relevant (22/263)
RQ2. Reasons for non-acceptance

Manual analysis
263 PRs from quasi-contributors

Superseded/duplicated (32/263)
Lack of experience/commitment (25/263)
PR not needed/not relevant (22/263)

Manual analysis results reverberate the quasi-contributors answers
RQ3. Quasi-contributors' perception

- Do you agree with your PR being unmerged?
  - Agree: 67.4%
  - Disagree: 32.6%

- Did the unmerged PR prevent or demotivate you to provide more PRs?
  - Agree: 30.3%
  - Disagree: 69.7%

- Were the comments in the PR constructive?
  - Agree: 88.8%
  - Disagree: 11.2%
RQ3. Quasi-contributors' perception

<table>
<thead>
<tr>
<th>Agreed with non-acceptance</th>
<th>Felt demotivated</th>
</tr>
</thead>
<tbody>
<tr>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>No</td>
</tr>
<tr>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>No</td>
</tr>
</tbody>
</table>
RQ1. How common are quasi-contributors and quasi-contributions?

RQ2. Why were the quasi-contributions not accepted?

RQ3. How do quasi-contributors perceive nonacceptance?
RQ1. How common are quasi-contributors and quasi-contributions?

RQ2. Why were the quasi-contributions not accepted?

RQ3. How do quasi-contributors perceive nonacceptance?
RQ1. How common are quasi-contributors and quasi-contributions?

RQ2. Why were the quasi-contributions not accepted?

RQ3. How do quasi-contributors perceive nonacceptance?

"The fix I submitted remained unmerged until someone else submitted the exact same fix ... the integrators accepted their (identical) fix and closed mine."

"When you add a new feature to the project, your vision can be out of tune with the vision of the project’s team, and this is natural."

"The project decided that was not a bug they wanted to provide a fix for."

Superseded/duplicated (32/263)

Lack of experience/commitment (25/263)

PR not needed/not relevant (22/263)
Disagreement \rightarrow Demotivation

**RQ1.** How common are quasi-contributors and quasi-contributions?

**RQ2.** Why were the quasi-contributions not accepted?

**RQ3.** How do quasi-contributors perceive nonacceptance?
RQ1. How common are quasi-contributors and quasi-contributions?

RQ2. Why were the quasi-contributions not accepted?

RQ3. How do quasi-contributors perceive nonacceptance?
Conclusions

- Quasi-contributors are rather common (70% of the actual contributors)

- Different reasons lead to non-acceptance
  - Duplicated/superseded PRs
  - Mismatch between developer's and team's vision
  - Developers' fault

- Non-acceptance may incur in demotivation (⅓ felt demotivated)

- Results can be useful to OSS maintainers, newcomers, and Educators
## RQ1. How common?

<table>
<thead>
<tr>
<th>Projects</th>
<th># Comments</th>
<th># Files</th>
<th># LoC Add</th>
<th># LoC Del</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>ES</td>
<td>PV</td>
<td>ES</td>
<td>PV</td>
</tr>
<tr>
<td>angular</td>
<td>-0.42</td>
<td>0.00</td>
<td>0.49</td>
<td>0.00</td>
</tr>
<tr>
<td>bitcoin</td>
<td>0.11</td>
<td>0.00</td>
<td>0.26</td>
<td>0.00</td>
</tr>
<tr>
<td>bootstrap</td>
<td>-0.41</td>
<td>0.00</td>
<td>0.35</td>
<td>0.00</td>
</tr>
<tr>
<td>caffe</td>
<td>0.16</td>
<td>0.00</td>
<td>0.01</td>
<td>0.72</td>
</tr>
<tr>
<td>d3</td>
<td>-0.35</td>
<td>0.00</td>
<td>0.01</td>
<td>0.38</td>
</tr>
<tr>
<td>django</td>
<td>-0.38</td>
<td>0.00</td>
<td>0.26</td>
<td>0.00</td>
</tr>
<tr>
<td>docker</td>
<td>0.00</td>
<td>0.87</td>
<td>0.22</td>
<td>0.00</td>
</tr>
<tr>
<td>flask</td>
<td>-0.38</td>
<td>0.00</td>
<td>0.00</td>
<td>0.98</td>
</tr>
<tr>
<td>jenkins</td>
<td>0.00</td>
<td>0.00</td>
<td>0.26</td>
<td>0.00</td>
</tr>
<tr>
<td>joomla!</td>
<td>0.03</td>
<td>0.21</td>
<td>0.38</td>
<td>0.00</td>
</tr>
<tr>
<td>jquery</td>
<td>-0.38</td>
<td>0.00</td>
<td>0.53</td>
<td>0.00</td>
</tr>
<tr>
<td>kubernetes</td>
<td>0.00</td>
<td>0.00</td>
<td>0.22</td>
<td>0.00</td>
</tr>
<tr>
<td>laravel</td>
<td>-0.36</td>
<td>0.00</td>
<td>0.02</td>
<td>0.35</td>
</tr>
<tr>
<td>mongo</td>
<td>-0.77</td>
<td>0.00</td>
<td>0.03</td>
<td>0.54</td>
</tr>
<tr>
<td>opencv</td>
<td>-0.24</td>
<td>0.00</td>
<td>0.18</td>
<td>0.00</td>
</tr>
<tr>
<td>rails</td>
<td>-0.28</td>
<td>0.00</td>
<td>0.13</td>
<td>0.00</td>
</tr>
<tr>
<td>react</td>
<td>-0.30</td>
<td>0.00</td>
<td>0.39</td>
<td>0.00</td>
</tr>
<tr>
<td>redis</td>
<td>0.20</td>
<td>0.00</td>
<td>0.00</td>
<td>0.92</td>
</tr>
<tr>
<td>scikit-learn</td>
<td>-0.07</td>
<td>0.08</td>
<td>0.12</td>
<td>0.00</td>
</tr>
<tr>
<td>spring</td>
<td>-0.20</td>
<td>0.00</td>
<td>0.29</td>
<td>0.00</td>
</tr>
<tr>
<td>tensorflow</td>
<td>-0.26</td>
<td>0.00</td>
<td>0.28</td>
<td>0.00</td>
</tr>
</tbody>
</table>
RQ1. How common?

Pull-requests by quasi-contributors are more discussed

→ in line with Tsay et al. (2014) - more discussed pull-requests are less likely to be accepted

RQ1. How common?

Pull-requests by quasi-contributors are more discussed
→ in line with Tsay et al. (2014) - more discussed pull-requests are less likely to be accepted

Quasi-contributors' pull-requests tend to be smaller