

# Responses to Editor and Reviewer Comments

No.	Comment	Authors' Response
<i>Review 1</i>		
1.1	The information presented is very interesting and has great potential. What I believe is missing is in the introduction and situating your research. Questions such as 1) Is OSS development struggling? 2) is OSS development missing key components? 3) what impact do OSS developers have on software development as an industry?	Thanks for your comment. We agree completely that there are many other aspects related to OSS development in general that we could touch upon. However, due to the focus of our study and the conference format's space restrictions, we particularly zoomed in on the problem of continued participation of developers and what drives their participation decisions. We therefore only highlighted the need for developers continuously participating in projects to keep them sustainable in the long run.
1.2	I believe that lessons could be taken from user experiences, marketing, etc. in some of your conclusions. The authors identified a niche that needs to be addressed but need to support why. In other words, what is the "So what?" behind this research. While space is limited, it would be helpful to begin with a development process where this research would have added to the process. Help the reader who has not participated in OSS imagine how your research could add to this topic.	Again, thanks for your input. As mentioned above, we tried to focus on the problem of continuous developer participation and briefly highlight previous research. Our main argument is that not only the social networks in OSS development are a potential influence on a developer's participation decision, but that also the technical network that results from package interdependencies plays a crucial role, which we indeed show in this paper. We agree that we could elaborate more on the OSS phenomenon, but unfortunately had to make cuts there due to the space limitations.
1.3	The references aren't formatted according to HICSS guidelines which is a major flaw. Authors sometimes spend hours just ensuring that this is correct. There were also other formatting issues.	We have taken great care to adhere to the formatting guidelines and template. As far as we are informed - and also stated in the provided template by HICSS - the reference format changed to APA this year, which we used in the submitted paper. If that is not the case, please let us know and we will happily change that. We would also like to know which other formatting issues you identified so that we can also address them accordingly.
1.4	This research has potential.	Thanks!
<i>Review 2</i>		
2.1	This paper clearly identifies the research gap in the open-source software ecosystem. To supplement research focusing on the social factors affecting developers' participation, this research applies a quantitative methodology with a large-scale dataset to test the technical relationship and connections with developers' participation.	Thanks for taking the time to review our paper!
2.2	If I understand correctly, I think the results are a little intuitive. Developers contribute to projects and packages related to their work (with high dependency). They would not be interested in packages and projects that are irrelevant to their work, though these packages have a high upstream or downstream dependency. I think the contribution is that this paper quantifies this assertion with a novel model.	We agree that the results, in hindsight, might seem intuitive. However, given the free-rider problem in OSS, we believe that developers contributing back to other projects they build upon cannot be taken for granted, and previous empirical results that support that are - as far as we know - not present.

No.	Comment	Authors' Response
2.3	The methods and model development are detailed. The discussion also acknowledges the non-novel findings and compares them in detail with prior literature. I would recommend acceptance, though I still have some concerns; however, I felt optimistic that the authors might resolve them.	Thanks for your kind words! We tried our best to address the issues that you raised below.
2.4	# Better explanation of upstreaming dependence.  What does it mean about dependencies of its own? Please elaborate more about it in the literature.	We agree that our wording for explaining the nature of upstream dependencies was not ideal. Hence, we rephrased these parts. Upstream dependencies arise if the focal package itself uses other packages. We hope that the explanation is now clearer.
2.5	#Little confused about the hypothesis statement.  In H1, packages with both upstream and downstream dependencies are more attractive than which objects.  It is hard to see the comparison object/entity. It seems like each package can be categorized into either upstream or downstream dependency. Is there a package with no dependency? Is no dependency the comparison object?  Recommendation: If there is another comparison object, please clarify it in the literature review and hypothesis statement.  Suppose the upstream and downstream are the comparison objects. In that case, the hypotheses are contradictory, a severe problem that needs to review the full results and discussion, which is not what I expected.	Thanks for your comment. We are not completely sure if we understand the issue you raise completely. First of all, packages can only be categorized into an upstream or downstream dependency from the perspective of a specific (focal) package. A package can be an upstream dependency of one package but also a downstream dependency of another.  Therefore, we looked at each package in our dataset and counted the number of its dependencies. Hence, the comparison is between each observed package. Of course, packages can have more or less up- or downstream dependencies. In our selected dataset, there are also packages with no upstream and/or no downstream dependencies.  Nevertheless, we rephrased hypothesis H1 to better match the statistical model and the statements that can be made based on the results.
2.6	In H2, there is no clear relationship between the two variables - packages and developers. Please specify the relationship (e.g., comparison, correlation, contrast, prediction, etc.)  I think the results also clearly show it is a comparison. For example, "developers are more likely to participate in a package if they also participate in another package that is a down- or upstream dependency of that specific package."  However, It is not clear about the comparison objects.	Again thanks for your comment. We rephrased hypothesis H2 to better match the statistical model and the possible propositions based on our findings.

No.	Comment	Authors' Response
3.1	This paper addresses how package dependencies in open-source projects can influence developers' participation. They propose and test four mechanisms by which such influence could occur. This topic is current and appropriate for the focus of the track and this mini-track.	Thanks for taking the time to review our paper!
3.2	<p data-bbox="304 362 1081 415">There does exist some prior work which looks at social networks of software dependencies in open source projects which is not yet cited in the submission.</p> <p data-bbox="304 456 1081 509">Consider this paper, which addresses the role of dependencies in the social networks of OSS:</p> <p data-bbox="304 521 1081 574">Zanella, Gianluca and Charles Zhechao Liu. "A Social Network Perspective on the Success of Open Source Software: The Case of R Packages." HICSS (2020).</p> <p data-bbox="304 615 478 634">Also these papers:</p> <p data-bbox="304 646 1081 727">Korkmaz, Gizem &amp; Kelling, Claire &amp; Robbins, Carol &amp; Keller, Sallie. (2018). Modeling the Impact of R Packages Using Dependency and Contributor Networks. 511-514. 10.1109/ASONAM.2018.8508255.</p> <p data-bbox="304 768 1081 930">Hazra, R., Aggarwal, H., Goyal, P., Mukherjee, A., Chakrabarti, S. (2021). Joint Autoregressive and Graph Models for Software and Developer Social Networks. In: Hiemstra, D., Moens, MF., Mothe, J., Perego, R., Potthast, M., Sebastiani, F. (eds) Advances in Information Retrieval. ECIR 2021. Lecture Notes in Computer Science(), vol 12656. Springer, Cham. <a href="https://doi.org/10.1007/978-3-030-72113-8_15">https://doi.org/10.1007/978-3-030-72113-8_15</a></p> <p data-bbox="304 979 1081 1084">M. Schwind, A. Schenk and M. Schneider, "A Tool for the Analysis of Social Networks in Collaborative Software Development," 2010 43rd Hawaii International Conference on System Sciences, 2010, pp. 1-10, doi: 10.1109/HICSS.2010.40.</p> <p data-bbox="304 1096 1014 1115">"...strengths and properties of collaboration ties between the developers..."</p> <p data-bbox="304 1164 1081 1352">The novelty of this paper and thus its potential contribution thus appear to lie in the specific application of stochastic actor-oriented models to the research context. The additional literature mentioned above conducts studies of dependency networks for a variety of reasons related to the aims of the authors: social networks, collaboration ties, project success and stability, for example. There is some additional novelty in the submission's consideration specifically of developer participation.</p>	<p data-bbox="1113 362 1890 415">Thanks for pointing us towards these additional studies on this topic. We looked at each of the suggested papers.</p> <p data-bbox="1113 427 1890 670">Based on our reading, the novelty and contribution of our paper is twofold. First, to the best of our knowledge, there is no study that looks at the influence of a project's (or package's) position and relations in the dependency network in a software ecosystem on developers' participation decisions. The suggested literature (as well as other similar publications) focus on the influence of a package's dependency network position on its success (Korkmaz et al., 2018; Zanella et al., 2020). The study by Schwind et al. (2010) looks at the internal dependency structure on a project-level. Here, we contribute by taking the package's environment (i.e., the software ecosystem) into account.</p> <p data-bbox="1113 682 1890 841">Second, as you also stated, another contribution is the application of SAOMs in this context, which is perfectly suited to investigate the participation choices of developers and thereby integrating the project's position and relations in the dependency network, even though the proposed method by Hazra et al. (2021) using autoregressive models looks interesting. However, SAOMs allows us to take the state of the network into account.</p>

- 3.3 I understand that space is limited; with that in mind, the theoretical development for the hypotheses is not yet fully developed. For hypothesis 1a, for example, the submission makes a face validity argument for the attractiveness of a project based on the quantity of upstream dependencies. This might be so, but the justification is very brief, and is related to but not established by the theoretical elements of the cited sources. Haefflinger 2008, for example, presents a highly granular look at why developers re-use code, which is related to but distinct from contributing to an OSS project. In Haefflinger's Table 3 (p. 26), H4b about coding fun and enjoyment (compare to the submission's "...focus on tasks they actually like to work on..." is not supported. Does this tend to support or undermine the authors' presentation of theory for H1A? The brevity of the theory itself may be at fault.

A similar approach is used for H1b regarding downstream dependencies. The literature establishes that reputation gains are important in participation. The hypothesis wants to extend that idea to the quantity of downstream dependencies; there is a face validity argument, but not one based on a theory that would indicate that the quantity of dependencies is actually a proxy for reputational stature for that project. I should add that I would tend to agree with that reasoning - IF the number of downstream dependencies were established as positively related to the importance and value of the package in the software ecosystem (citation?), then the reasoning appears sound.

I can't quite follow this reasoning from the submission:

This means that developers should not only participate in the originating project, but also in the newly created upstream dependency. Hence, we propose:

H2a: Packages attract developers from upstream dependencies.

The sequence presented: 1) developers are first working on an originating project, then 2) become interested in a new, upstream project, seems to imply that the H2a would be "Packages attract developers TO upstream dependencies". Or rather than a directional/sequential relationship, something like "Developers are attracted to participation both in an originating project, and to its upstream dependencies.

I understand that this is not the place to revisit fundamental components such as hypotheses all over again. Perhaps the presentation for theoretical development for H1B can be clarified/improved.

With the caveat that these conference papers do not have the breadth necessary for comprehensive theoretical development, the theoretical basis for the hypotheses might be stronger than seen so far in the submission.

Thanks for your comment. We agree that the development of the hypotheses is not as detailed as it could be, which unfortunately is because of the space restrictions.

Regarding the issue you raise related to Haefflinger et al. (2008), we are a little confused. It looks like you are referring to a different paper, because in the Management Science paper, there is for example no table 3. However, based on the findings from the conducted interviews, the paper explicitly states that the reuse of code allowed developers to focus on more rewarding tasks by evading the need for writing "mundane" code (see section 4.3 pp. 189-190). These findings would support our hypothesis H1a.

Regarding your critique on H1b, we agree that this is not based on theory. However, we tried to make a face-valid argument for why the number of downstream dependencies function as a proxy for the package's importance and usage frequency in the software ecosystem. If not for the space restrictions, we could make a stronger argument and elaborate more on this. Nevertheless, we added a half-sentence including a reference that points to the proxy of the number of a package's downstream dependencies for its user base.

Lastly, thanks for pointing out the confusion that H2a raises. We agree that the wording and presented sequence is not particularly precise. Hence, we rephrased the sentence and also the hypothesis to better match the statistical model and the possible propositions based on our findings.

No.	Comment	Authors' Response
3.4	Figure 1 is fascinating, if densely populated. There are so many edges depicted that, except at the edges, little information is conveyed by the gray lines. On the other hand, there are interesting cluster formations, especially towards the center. Graph layouts rely on some distance metric which drives the layout depiction. Could the authors supply the algorithm used? Fruchterman-Reingold? Something else?	Indeed, we used the Fruchterman-Reingold algorithm. We added a sentence after mentioning Figure 1 that provides the used algorithm.
3.5	The stochastic actor-oriented modeling approach is exciting, with powerful tools to address change processes in social networks of OSS projects. I notice from the submission's Table 3 that the vast majority of events were reflective of stasis ( $0 \rightarrow 0$ ). The hypotheses tend speak to characteristics of projects, not evolution. H1a and H1b posit that projects "are more attractive", implying a durable characteristic. H2a and H2b by contrast do use active-voice verbs relevant to processes, here potential changes: "Packages attract". The authors' interests, data, and methodology appear to be most closely related to the evolution/change aspects. Again, I don't want to dive into wholesale revision of hypotheses, necessarily. I like the stochastic actor-oriented modeling a lot, and I believe it's fruitful.	The amount of "zero" ties that stay "zeroes" only reflects the sparse characteristic of the network where many of the possible edges are not initiated. The amount of change in the network is reflected by the distance, which is the sum of all tie changes ( $(0 \rightarrow 1) + (1 \rightarrow 0)$ ), and the Jaccard index. Related to our hypothesis, we rephrased the wording of H2a and H2b to better match the statistical model and the statements that can be made based on the results.