Open Source, Open Science, and the Replication Crisis in HCI

Florian Echtler
Bauhaus-Universität Weimar
Weimar, Germany
florian.echtler@uni-weimar.de
Maximilian Häußler
Genomics Institute
University of California, Santa Cruz
Santa Cruz, CA, USA
max@soe.ucsc.edu

CHI'18 Extended Abstracts, April 21–26, 2018, Montréal, QC, Canada.

ACM ISBN 978-1-4503-5621-3/18/04 ...$15.00.
https://doi.org/10.1145/3170427.3188395

Abstract
The open-source model of software development is an established and widely used method that has been making inroads into several scientific disciplines which use software, thereby also helping much-needed efforts at replication of scientific results. However, our own discipline of HCI does not seem to follow this trend so far. We analyze the entire body of papers from CHI 2016 and CHI 2017 regarding open-source releases, and compare our results with the discipline of bioinformatics. Based on our comparison, we suggest future directions for publication practices in HCI in order to improve scientific rigor and replicability.

Author Keywords
open source; open science; replication; HCI

ACM Classification Keywords
H.5.m [Information interfaces and presentation (e.g., HCI)]: Miscellaneous

Introduction
Many research results are not reproducible [7] and may consequently be incorrect. The shock waves of this "replication crisis" can be felt across the sciences. As a result, empirical disciplines such as psychology and the life sciences are trying to adapt their analysis methods and publication requirements, e.g. by using open datasets where

possible and by requiring the release of raw data and analysis software. The entirety of these efforts is sometimes subsumed under the term "open science", as discussed by Ben Marwick [14]. While there are noticeable differences between individual disciplines regarding what is actually meant by "open data" as analyzed by Pasquetto et al. [16], a general agreement is that at least raw data and experimental descriptions as well as all software used for filtering, processing and analyzing the data should be included.

There is even some disagreement about what the terms "reproducibility" or "replicability" by themselves mean [9]. Generally, "reproducibility" is assumed to refer to the ability to re-run statistical analyses on the original raw data and arrive at the same conclusions, while "replicability" refers to the ability to re-execute the experiment, ideally under identical conditions, and thereby collect a new, raw dataset that remains comparable to the first one.

In this paper, we look at the extent by which the replication crisis affects a field like HCI that also involves a significant number of software artifacts. Of course, this is also related to the larger question of whether HCI itself can be classified as a "science", a "discipline" or something else entirely (for a more detailed look at this topic, we refer to Dix [5] and Reeves [18]). Independent of this philosophical discussion, others have asked similar questions about replicability before: for example, this was discussed at the RepliCHI series of panels, SIGs and workshops at CHI 2012 - 2014 [23, 25, 24]. Unfortunately, no further structured discussion of this topic seems to have taken place in the last years. Since 2016, SIGCHI recommends to provide supplementary material for ACM publications [15] in order to improve replicability. A related topic, focusing on transparent statistics, was also discussed as a SIG at CHI 2016 [10] and a workshop at CHI 2017 [11].

As HCI is centered on computing artifacts, a particularly interesting subset of this topic is the relation between open-source software/hardware and HCI research: how is it used, and how is it contributed to? How are efforts at replication hindered by lack of source code and data? How can the open-source model help with current questions about the publishing process? These are the questions we discuss in this paper.

Open Source in HCI

To get an impression of how prevalent the usage of and contribution to open-source software in HCI actually is, we performed an automated analysis of the entire body of extended abstracts and papers from the CHI 2016 and 2017 conferences.

Our initial attempt was based on searching the PDFs for links to popular open-source repositories such as Github, BitBucket or Sourceforge. However, a vast number of papers include links to third-party software, and the resulting amount of false positives was far greater than the actual amount of open-source releases related to the paper.

Consequently, we switched to a keyword-based approach: we first converted all PDF files into searchable plain-text files, and then automatically located all case-insensitive instances of the keywords "free software", "open(-|)source", "source(-|)code", "F(L)OSS"[1], and "supplement" within these files. Files mentioning one or more of these terms were then manually analyzed to see if they only mentioned the concept of open source (usually in conjunction with a library or piece of software used in the paper), or if they actually released source code to the public. Of course, it is possible that our search terms did miss some papers where

[1] Free, (Libre,) Open-Source Software

the authors used a different wording to refer to their released source code; however, we consider our keywords to be suitable to match a large majority of possible wordings. All source code and raw data for the analysis are available on GitHub at https://github.com/mmbuw/chi-paper-analysis.

The results of our analysis are shown in table 1. *Out of a total of 1027 papers, notes and extended abstracts published at CHI 2016, only **21** individual publications in total released source code to the public. For CHI 2017, the figures increase very slightly to 1057 total papers with **27** source code releases, although the total percentage still remains very low.* We acknowledge that some papers and extended abstracts are dealing with topics where no source code or similar artifacts were produced at all, such as workshop abstracts or ethnographic studies. Nevertheless, even if we conservatively assume that 20 % of all papers belong to these categories, the amount of projects that actually released source code remains exceedingly low when compared to the total of number of projects which have produced or modified source code in some way. Interestingly, 24 (2016) and 40 (2017) publications referred to and provided supplemental material, either in the ACM Digital Library or in another online repository; however, only four of these papers in 2016 and one in 2017 provided source code as part of the supplement.

Given the fact that nearly 5 times as many papers at least acknowledge and/or use open-source software in some way, this low number is even more depressing - these projects are "standing on the shoulders of giants" by building on the work of countless open-source developers, but at the same time deny others the same opportunity to build on their own work. While open-access policies are now often implemented for publicly-funded research, these policies usually only focus on the publications created, and not on the source code or datasets. An interesting and very recent addition to these policies which explicitly encourages open-source releases is the "Federal Open Licensing Playbook" [20]. In a similar manner, the open letter of the "Public Money, Public Code" initiative [8] is advocating that any software developed with public funds should also be available to the public as open-source.

Although there may be a wide range of reasons for this lack of releases (e.g. anxiety that one's own code is "not good enough" for the public [2], not wanting to provide future support for the software, social barriers when contributing to an existing project [19]), the most likely reason is that this practice is not encouraged by the community. Packaging a research project's codebase for release will undeniably create extra work for the authors, and this work is currently not rewarded. While individual, widely known research software packages such as R have many volunteer contributions, the motivations for these contributions are complex and often not long-term, as analyzed by Mair et al. [12].

In a certain sense, this is a "chicken-and-egg" problem: HCI, as a discipline, sometimes seems to be determined to "reinvent the wheel" time and again instead of building on existing work (also called "not-invented-here syndrome"). This may be part of a broader issue within HCI as identified by Marshall et al. [13]: citations to prior work are often also not reflected upon critically, and the content of the prior publication is not engaged with. Consequently, even if researchers do the extra work of releasing their projects as open source, they are unlikely to be reused by other researchers and quickly become forgotten. Seeing this trend, other researchers are not motivated to invest a significant amount of work for what is perceived as a fruitless effort, and even fewer projects actually get published in an open

	CHI 2016			CHI 2017			ISMB 2016	ISMB 2017
	Extended Abstracts	Papers/ Notes	Total	Extended Abstracts	Papers/ Notes	Total	Papers	Papers
Total	567	460	1027	458	599	1057	55	56
containing OS keywords	50	74	124	33	91	124	24	56
... supplemental material	2	22	24	4	36	40	-	-
releasing source code	12	10	22	6	21	27	20	49
Percentage	2.1%	2.0%	2.0%	1.3%	3.5%	2.6%	36.4%	87.5%

Table 1: Open-source in HCI and bioinformatics: mentions and releases

manner. As Ye and Kishida [26] as well as von Krogh et al. [21] have found, a successful open-source community requires both intrinsic (learning) and extrinsic (social) motivation for participants. If the extrinsic/social aspect is underrepresented (as it is currently the case in HCI), it is unlikely that such a community will thrive or even form in the first place..

Comparison to Bioinformatics

To put our results in context, we look to another research field with strong connections to software. Bioinformatics has a strong link to the "traditional" disciplines of chemistry, biology and medicine. Since medical data analysis can decide about life and death, and OECD countries spent 59 billion USD on life science research in one year [7], there is a lot of pressure to address reproducibility issues in these domains. An increasing number of journals request a public release of the underlying data and analysis source code as a precondition to publication. The basic rule for most prestigious journals with a high rejection rate is that data and source code must be made available to readers. Where these rules are not codified yet, there is some pressure by referees to ask for data publication. Some high-ranking journals require software release, such as *Nature*[2] and *Science*[3], some even under an open source license (*Genome Biology*[4]).

For purpose of our analysis, we have downloaded all articles accepted for the main bioinformatics conference, ISMB 2016 [1] and 2017 [4]. All manuscripts are available under an open access license and therefore on the FTP server of PubmedCentral[5], the repository of free life science manuscripts. For 2016, 24 articles mention the words "open source" or synonyms, as per the search terms described above, and 20 articles indicate availability of an open source code repository, on either GitHub, SourceForge or BitBucket. *For 2017, all 56 articles matched the keywords, and an impressive 49 out of 56 papers do indeed release their source code (we manually verified this exceptional number).* While it is difficult to directly compare these values with the ones we determined for CHI 2016/17 due

[2] http://www.nature.com/authors/policies/availability.html

[3] http://www.sciencemag.org/authors/science-editorial-policies#data-deposition

[4] https://genomebiology.biomedcentral.com/submission-guidelines/preparing-your-manuscript/software

[5] https://www.ncbi.nlm.nih.gov/pmc/

to the different venue, sample size, and prevalent publication strategy, it is safe to say that a far higher percentage of publications in bioinformatics are accompanied by an open-source release of the software used and created by the authors.

Counterarguments

Advocating for a stronger adoption of the open-source approach in HCI has been met with recurring counterarguments, which we would briefly like to address here.

"I don't have time to create an open-source project from my research."
We believe that even a five-minute open-source release which solely consists of a project snapshot and a license file is better than none at all. In addition, this is likely due to the aforementioned "chicken-and-egg" problem. If the community would value these contributions as equal to a published paper - or better yet, as integral part of the publication - researchers would not consider the time spent preparing such a contribution as wasted. If the established members of our HCI community lead by example, the required shift in perception can easily be achieved.

"Pushing for open-source will discourage companies from participating in HCI research."
In our opinion, this argument is easily invalidated by looking at the whole industry that has developed around open-source software. Companies in this space regularly release their software under a dual-license model in which non-commercial and research usage is covered by an open license, while commercial use requires an individual paid-for license. Additionally, even researchers at very large companies such as Microsoft have already published research projects as open-source - a recent example by A. Wilson is available as [22].

"A focus on open source disregards HCI contributions which do not create source code."
As mentioned above, we acknowledge that there is a wide spectrum of possible research contributions in the HCI space which do not necessarily involve writing source code. However, within the context of this paper, we use the term "open source" more loosely to refer to the "open source philosophy" of releasing one's tools in general, which we consider applicable to any research contribution - e.g. authors of an ethnographic study should also make their questionnaires, coding schemes, and other artifacts such as spreadsheets or statistical analysis scripts available to the public.

Open-Source Publishing?

Online collaboration tools such as `git` and GitHub have helped create a new variant of software development, particularly in the open-source context: collaborators create a personal clone of the entire software repository, a so-called *fork*, and work on that linked copy. From time to time, interesting local changes are then submitted to the original repository as a so-called *pull request* and integrated into the primary copy at the owner's discretion. Forks can pull from and push to multiple other repositories, forming a directed graph comparable to citation graphs as used in "traditional" publication analysis.[6]

This model is not just used purely for source code anymore, but increasingly also for documentation and other written content. *So why not use it for an entire publication, including the paper itself, the source code, the data files and all supplementary material?* These components thereby become equal and inextricably linked members of a larger research project. Source code stops being an

[6] [6] is an intentionally recursive citation designed as a test case for such citation graph analysis tools.

afterthought, to be released maybe sometime after the paper. Interested parties can immediately create a fork of the entire project, post comments or recommendations, integrate changes they consider necessary or useful across all components and submit a pull request back to the original authors. An example of such an "integrated" repository is available on GitHub at https://github.com/mmbuw/massive-mobile-multiplayer/tree/master/paper.

Just as with software-only projects, a specific snapshot of the research repository can be tagged as a release, a fixed instance of the project that can be referred to, i.e. "cited", by other papers or projects. One such specific instance might be the version of paper and software that was presented at a conference.

Some subsets of this proposal already exist in isolation. Services like Zenodo[7] offer the possibility to assign an immutable DOI to a specific release of a Github repository, thereby allowing "traditional" publications to cite the software they use. Hypothes.is[8] [17] allows users to post public annotations and comments to any website they visit, with a particular focus on scholarly articles. Distill[9] focuses on interactive articles that combine text and code to illustrate machine learning topics in particular, and ReScience[10] is a novel type of journal which exclusively publishes replication studies, to be submitted with all code and discussion on GitHub.

Discussion & Outlook

Of course, our proposed approach will open up a wide range of questions that cannot be easily answered. Some of these are of a technical nature, e.g. what happens if a repository URL needs to be changed? Where should such repositories be hosted? Other questions relate to the core of our current publishing model - e.g. how will authorship be determined in this new approach? Is anybody who fixed a typo in the original paper already a co-author for all future versions? How can double-blind peer review be integrated into this approach, e.g. as post-publication review? Do we even want to continue with the double-blind review model, or can we use this opportunity to try out different approaches?

On the other hand, other researchers within the HCI context have already considered alternative approaches to authorship, e.g. BD et al. at CHI 2016 [3]. Git, with its line-by-line tracking of authorship in text-based documents, could provide a basis for the kind of fine-grained attribution of individual contributions proposed in this paper.

Although we do not yet have definite answers for the questions posed above, we are convinced that in order to thrive as a discipline, HCI will have to adopt a more open stance. Our proposal is to encourage a mixed approach at first, in which traditionally-reviewed and -published papers are also published in an open-source repository along with the relevant data and source code, and marked as a release. If this approach gains traction, then it may offer a path to transforming the whole publishing model into a more open variant.

After CHI 2018, we plan to supplement our keyword-based analysis with a survey sent to all corresponding authors of papers and extended abstracts to gather more fine-grained feedback about what types of computing artifacts were generated during the course of the project, and whether they are available to the public in some way.

[7] https://zenodo.org/
[8] https://hypothes.is/
[9] https://distill.pub/about/
[10] https://rescience.github.io/about/

Acknowledgements

We would like to thank Alan Dix, Abram Handler, Matthew Kay, Henning Pohl, and Paul Strohmeier for their valuable feedback on earlier versions of this text, which has been integrated into the final version of this paper.

References

[1] Pierre Baldi and Teresa Przytycka. 2016. ISMB 2016 Proceedings. *Bioinformatics* 32, 12 (2016), i1–i2. DOI: http://dx.doi.org/10.1093/bioinformatics/btw326

[2] Nick Barnes. 2010. Publish Your Computer Code: It Is Good Enough. *Nature News* 467, 7317 (13 Oct. 2010), 753–753. DOI: http://dx.doi.org/10.1038/467753a

[3] AC BD, Christine Bauer, and Afsaneh Doryab. 2016. Solving the Battle of First-Authorship: Using Interactive Technology to Highlight Contributions. In *Proceedings of the 2016 CHI Conference Extended Abstracts on Human Factors in Computing Systems (CHI EA '16)*. ACM, New York, NY, USA, 609–620. DOI: http://dx.doi.org/10.1145/2851581.2892582

[4] Niko Beerenwinkel and Yana Bromberg. 2017. ISMB/ECCB 2017 proceedings. *Bioinformatics* 33, 14 (2017), i1–i2. DOI: http://dx.doi.org/10.1093/bioinformatics/btx321

[5] Alan Dix. 2010. Human-computer Interaction: A Stable Discipline, a Nascent Science, and the Growth of the Long Tail. *Interact. Comput.* 22, 1 (Jan. 2010), 13–27. DOI: http://dx.doi.org/10.1016/j.intcom.2009.11.007

[6] Florian Echtler and Maximilian Häußler. 2018. Open Source, Open Science, and the Replication Crisis in HCI. In *Proceedings of the 2018 CHI Conference Extended Abstracts on Human Factors in Computing Systems (CHI EA '18)*. ACM, New York, NY, USA. DOI: http://dx.doi.org/10.1145/3170427.3188395

[7] The Economist. 2013. Trouble at the Lab. (19 Oct. 2013). http://www.economist.com/news/briefing/21588057-scientists-think-science-self-correcting-alarming-degree-it-not-trouble.

[8] Free Software Foundation Europe (FSFE). 2017. Public Money, Public Code. (2017). https://publiccode.eu/

[9] Steven N. Goodman, Daniele Fanelli, and John P. A. Ioannidis. 2016. What does research reproducibility mean? *Science Translational Medicine* 8, 341 (2016). DOI: http://dx.doi.org/10.1126/scitranslmed.aaf5027

[10] Matthew Kay, Steve Haroz, Shion Guha, and Pierre Dragicevic. 2016. Special Interest Group on Transparent Statistics in HCI. In *Proceedings of the 2016 CHI Conference Extended Abstracts on Human Factors in Computing Systems (CHI EA '16)*. ACM, New York, NY, USA, 1081–1084. DOI: http://dx.doi.org/10.1145/2851581.2886442

[11] Matthew Kay, Steve Haroz, Shion Guha, Pierre Dragicevic, and Chat Wacharamanotham. 2017. Moving Transparent Statistics Forward at CHI. In *Proceedings of the 2017 CHI Conference Extended Abstracts on Human Factors in Computing Systems (CHI EA '17)*. ACM, New York, NY, USA, 534–541. DOI: http://dx.doi.org/10.1145/3027063.3027084

[12] Patrick Mair, Eva Hofmann, Kathrin Gruber, Reinhold Hatzinger, Achim Zeileis, and Kurt Hornik. 2015. Motivation, Values, and Work Design as Drivers of Participation in the R Open Source Project for Statistical Computing. *PNAS* 112, 48 (2015), 14788–14792. DOI: http://dx.doi.org/10.1073/pnas.1506047112

[13] Joe Marshall, Conor Linehan, Jocelyn Spence, and Stefan Rennick Egglestone. 2017. Throwaway Citation of Prior Work Creates Risk of Bad HCI Research. In *Proceedings of the 2017 CHI Conference Extended Abstracts on Human Factors in Computing Systems (CHI EA '17)*. ACM, New York, NY, USA, 827–836. DOI: http://dx.doi.org/10.1145/3027063.3052751

[14] Ben Marwick. 2015. How Computers Broke Science – and What We Can Do to Fix It. (9 Nov. 2015). http://theconversation.com/how-computers-broke-science-and-what-we-can-do-to-fix-it-49938.

[15] Dan Morris. 2015. Improving the replicability of HCI research with supplementary material. (23 Sept. 2015). http://sigchi.tumblr.com/post/129729411430/improving-the-replicability-of-hci-research-with.

[16] Irene V. Pasquetto, Ashley E. Sands, Peter T. Darch, and Christine L. Borgman. 2016. Open Data in Scientific Settings: From Policy to Practice. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (CHI '16)*. ACM, New York, NY, USA, 1585–1596. DOI: http://dx.doi.org/10.1145/2858036.2858543

[17] Jeffrey M. Perkel. 2015. Annotating the Scholarly Web. *Nature* 528, 7580 (2015), 153–154. DOI: http://dx.doi.org/10.1038/528153a

[18] Stuart Reeves. 2015. Human-computer Interaction As Science. In *Proceedings of The Fifth Decennial Aarhus Conference on Critical Alternatives (AA '15)*. Aarhus University Press, 73–84. DOI: http://dx.doi.org/10.7146/aahcc.v1i1.21296

[19] Igor Steinmacher, Tayana Conte, Marco Aurélio Gerosa, and David Redmiles. 2015. Social Barriers Faced by Newcomers Placing Their First Contribution in Open Source Software Projects. In *Proceedings of the 18th ACM Conference on Computer Supported Cooperative Work & Social Computing (CSCW '15)*. ACM, New York, NY, USA, 1379–1392. DOI: http://dx.doi.org/10.1145/2675133.2675215

[20] United States Department of State. 2017. Federal Open Licensing Playbook. https://eca.state.gov/files/bureau/open_licensing_playbook_final.pdf

[21] Georg Von Krogh, Stefan Haefliger, Sebastian Spaeth, and Martin W. Wallin. 2012. Carrots and Rainbows: Motivation and Social Practice in Open Source Software Development. *MIS Q.* 36, 2 (June 2012), 649–676. http://dl.acm.org/citation.cfm?id=2481639.2481654

[22] Andrew D. Wilson and Hrvoje Benko. 2017. Holograms Without Headsets: Projected Augmented Reality with the RoomAlive Toolkit. In *Proceedings of the 2017 CHI Conference Extended Abstracts on Human Factors in Computing Systems (CHI EA '17)*. ACM, New York, NY, USA, 425–428. DOI: http://dx.doi.org/10.1145/3027063.3050433

[23] Max Wilson, Wendy Mackay, Ed Chi, Michael Bernstein, and Jeffrey Nichols. 2012. RepliCHI SIG: From a Panel to a New Submission Venue for Replication. In *CHI '12 Extended Abstracts on Human Factors in Computing Systems (CHI EA '12)*. ACM, New York, NY, USA, 1185–1188. DOI: http://dx.doi.org/10.1145/2212776.2212419

[24] Max L. Wilson, Ed H. Chi, Stuart Reeves, and David Coyle. 2014. RepliCHI: The Workshop II. In *CHI '14 Extended Abstracts on Human Factors in Computing Systems (CHI EA '14)*. ACM, New York, NY, USA, 33–36. DOI: http://dx.doi.org/10.1145/2559206.2559233

[25] Max L. L. Wilson, Paul Resnick, David Coyle, and Ed H. Chi. 2013. RepliCHI: The Workshop. In *CHI '13 Extended Abstracts on Human Factors in Computing Systems (CHI EA '13)*. ACM, New York, NY, USA, 3159–3162. DOI: http://dx.doi.org/10.1145/2468356.2479636

[26] Yunwen Ye and Kouichi Kishida. 2003. Toward an Understanding of the Motivation Open Source Software Developers. In *Proceedings of the 25th International Conference on Software Engineering (ICSE '03)*. IEEE Computer Society, Washington, DC, USA, 419–429. http://dl.acm.org/citation.cfm?id=776816.776867