The Manubot Manifesto: envisioning the future of scholarly publishing

This manuscript (permalink) was automatically generated from manubot/manufesto@51b43e8 on October 22, 2019.

Authors

- **Daniel S. Himmelstein**
  0000-0002-3012-7446 · dhimmel · dhimmel
  Department of Systems Pharmacology and Translational Therapeutics, University of Pennsylvania, Philadelphia

Abstract

This document lays out a vision for the future of scholarly publishing. Specifically, we define features that an ideal platform for authoring & publishing should possess. Community contributions to edit this manuscript are welcome via GitHub.

Introduction

The process by which scholarly articles are written, reviewed, and published affects the pace of progress. Unfortunately, the process is outdated and, as such, predisposes research to be irreproducible, siloed, proprietary, poorly documented, and difficult to build upon.

The internet combined with a growing toolset of open source infrastructure presents us with an opportunity to reinvent publishing [1,2]. But how? Here we envision the functionality and principles that would define an ideal publishing platform. We presuppose that infrastructure for scholarly authoring and publishing should be tightly integrated. That in the future, the separated tasks of drafting, reviewing, revising, and publishing will be merge into a single process of iterative refinement called scholarly communication or “science”.

This document is referred to as the Manubot Manifesto (i.e. Manufesto). Manubot is a workflow and set of tools that allows users to write manuscripts openly on GitHub in the Markdown format, while providing a high level of automation, transparency, and opportunity for collaboration. Manubot implements many of the features described below; the rest are goals. Manubot aims to empower technical users with the ability to demonstrate, and in doing so define, the future of publishing.

Our hope is that our vision will be realized not only by Manubot, but by other tools for scholarly authoring and publishing. In the meantime, we must experiment, explore the pitfalls of various innovations, and establish best practices, such that when scholars are ready to migrate en masse, the path forward is clear.

Features

What are the ideal features of an authoring/publishing platform?

Libre content

Open access should be the default. The benefits of openly licensed scholarly literature are so immense that we should avoid any platform whose incentives are misaligned with the open licensing of manuscripts.

Low cost

Authoring and publishing should be free of charge. Article processing charges (APCs, and/or subscription revenue) do not fund the intellectual contributions or creation of manuscripts. Instead the revenue of publishers funds inefficient publishing platforms that can either be fully replaced by automation or the existing volunteer contributions of authors and reviewers. Nonetheless, the cost of publishing continues to rise [3]. There will always be some cost for compute, hosting, and archiving of scholarly communication, but these should be in the range of dollars per article not thousands [4]. Futhermore, several providers currently offer these services for free to public projects.

Journals do provide services that aren't entirely performed by unpaid volunteers. Examples include orchestrating peer review, performing copyediting, and helping disseminate and advertise an article. However, these services should not be a precondition to publication. Instead, they should be opt-in services that authors or funders solicit to improve a manuscript, establish its credibility, and expand its visibility. APCs could fall dramatically if publication were unbundled from the value-added services of journals, which could then be offered under à la carte pricing models [5].

Libre infrastructure

Authoring and publishing platforms should be open source and extensible. Innovation in publishing will accelerate when end-users can make enhancements as they see fit. Empower authors with full control of their publishing platforms. Scholars who are discontent with the current publishing system should have a direct way to change it. Thanks to open licensing, open platforms are protected against discontinuation: the ability of the community to fork the project also guards against stagnation and mistreatment of users.

Issue trackers for journals

Journals should have public issue trackers and accept community pull requests to improve their infrastructure. Journals are buggy and publishers currently make many mistakes. Even popular preprint servers do not have issue trackers that allow users to discuss bugs or suggest improvements. Currently, when there's a problem with a journal, the public can either email or tweet the journal. Neither method is well-suited to transparent, continuous public discussion that is critical to accountability and ensuring the best solutions are pursued.

Did a publisher submit incorrect metadata to Crossref or PubMed? Submit a pull request to fix the software that deposits the metadata, and fix not only your paper's metadata but also all other records with the same issue.

Instant publication

Publishing should be instant. Science can only advance as quickly as scientists can communicate. However, the publishing process currently imposes extreme delays, generally on the range of months to years, on scholarly communication [6,7]. Therefore, all manuscripts should be self-published instantly. By accepting self-published articles for submission, preprint servers and journals will impose delays only on their distribution of an article, but not on the underlying availability of its content. Systems that support instant publication distangle publication from evaluation, an essential change to accelerate communication [2].

Versioned publishing

Publishing should be versioned. Every coherent and self-sufficient changeset should create a new version with a persistent identifier. Old versions should remain accessible by permalink. Content-addressing, i.e. versioning by git commit hash, can ensure the integrity of content returned by a permalink. Specific versions can be tagged as major releases to provide larger manuscript checkpoints similar to preprint versions currently.

Platforms should provide easy ways of comparing different manuscript versions through rich diffs. For example, readers should be able to quickly evaluate how a manuscript evolved in response to community, author, or reviewer feedback. Currently, journals occasionally require authors to submit a tracked changes document, while most preprint servers don't provide diffs at all. Ideally, diffs would automatically be generated from two manuscript versions. Automated rich diffs have the benefits of saving authors extra effort, while simultaneously providing more trustworthy diffs to viewers that are not prone to human error or manipulation.

Early feedback

Public feedback should occur as early as possible. Feedback is most actionable while a study is in progress. Therefore, we should seek authoring platforms that expose manuscripts to public scrutiny as early as possible. By doing so, the community can provide peer review in a proactive rather than reactive setting.

Living manuscripts

Manuscripts should be living, even once published in a journal. Science is a continuous process. Versioning enables manuscripts to be continuously updated. At some point in a paper's life cycle, its authors may choose to stop accepting major revisions that change the findings of the study. However, revisions to clarify methods, fix typos, or cite overlooked prior work should continue into the future. No more errata or corrigenda as separate publications: just a new version with a note of who changed what and why!

Community contributions

Community members should be able to propose contributions to public manuscripts. Presently, independent teams work in private to produce siloed and redundant research. Online writing in a public venue reduces the hurdles to contribute to existing projects, tipping the scales towards global collaboration rather than fragmentation. For some studies, community members may only contribute small enhancements, like fixing typos. Other studies, may become massively open online papers (MOOPs) where geographically dispersed experts co-investigate at scale [8].

Contributions proposed by the community should be reviewed by manuscript maintainers — the individuals in charge of approving manuscript changes. Discussion and iterative refinement of the proposed change can proceed until the change is accepted. When community members provide substantive contributions, they should be offered formal coauthorship.

One forum

Changes requested by journals, either by editorial or peer review, should occur in the **same public forum** and system where all manuscript discussion and edits are performed. The formatting changes and copyedits performed by journals should receive the same transparency and scrutiny as those performed by the authors themselves. Since the provenance of every word in every sentence of a study must be tracked to have a complete record, it is imperative that journals are subject to the same standards of transparency as authors.

Post-publiation peer review

Post-publication peer review should be supported. Currently, we entrust the evaluation of a study to a few individuals who submit private feedback on a study to an editor. These peer reviewers have little incentive to perform a thorough and constructive review. Post-publication peer review enables anyone interested in the study — a much larger body of individuals — to submit feedback. Many times, a post publication reviewer will scrutinize a study more closely than a pre-publication reviewer, especially since they are more likely to investigate and attempt to replicate the methods.

We cannot let discussion on the merits of a study go to waste. Therefore, we need design systems to capture the offline feedback that currently represents the majority of intellectual input scholarly works receive. Therefore, publishing platforms should provide ample tools for post publication commenting.

Artifact integrity

Manuscript inputs should be **provenanced, attributable, and reproducible**. All artifacts — such as quantitative findings, figures, tables — should be traceable to their source. For example, if a manuscript reports a value of "3.5" that value should be inserted directly from the upstream analysis rather than hardcoded into the manuscript. By inserting analysis results directly, the integrity of artifacts is straightforward to establish. In other words, readers need not worry that the authors mistyped a crucial value or inserted an outdated version of a figure.

Furthermore, readers can go backwards from a value or figure to the source code that generated it. Assuming the analyses are open, readers can then modify the analysis and evaluate the resulting changes to the manuscript. Reproducibility is an important precursor to modifiability, which is where the large gains will occur as scholars can rerun previous experiments tweaked to their own specification.

One helpful practice is to reference upstream analyses using content-addressing. Content-addressing provides an efficient method for not only requesting a specific version of an analysis but also enforcing the exact source code and data used as inputs.

Transparent history

The history of a manuscript should be easily viewable. For a sentence or phrase, it should be easy to inspect its history: Who originally wrote it and when? What subsequent edits have been made and why?

The inability to attribute text to a specific author and changeset undermines the integrity of the whole manuscript. For example, if a specific statement comes under question, it is essential to know who wrote it, when, and why to re-evaluate its veracity. Furthermore, attribution helps keep authors honest.

Readers who want to know the source of a statement, in terms of its author or source analysis, shouldn't have to inquire. Instead the interface for viewing papers should immediately make clear the history of the statement. With this information the following interface becomes possible: click an author's name on a manuscript to highlight all the prose they created or modified. In addition, show which artifacts in the manuscript they contributed to either by creating the source code or performing commentary.

Public conversation

All **intellectual input into a manuscript should be preserved** and retrievable. Perhaps the most wasteful aspect of the current scholarly system is how the overwhelming majority of labor and conversation occur in transient channels and therefore does not get recorded into the scientific record. Therefore, it is often impossible to figure out why researchers did what they did. Mistakes are repeated; false leads remain undocumented.

When reading a manuscript, it should be easy to retrieve all relevant conversation, experimentation, and documentation relating to a specific passage. The power of linking manuscript sections to the associated intellectual inputs will be most powerful when project notebooks, discussion, and peer review are all open and preserved. As this begins to occur, we need a publication system that ensures these supplemental records are easily retrievable from the final product.

Dissuade misconduct

Academic misconduct should be infeasible. Misconduct thrives when perpetrators thinks there's a strong possibility it will go undetected. For manuscripts where neither provenance nor revisions are publicly tracked, the allure of misconduct is strongest. Alternatively, when research is fully transparent, misconduct is not a viable strategy.

Misconduct comes in many forms: misrepresenting results, manipulating data, lies of omission, mistreatment of others, claiming credit for others' work, and other unethical means of performing science. One commonality is that public scrutiny is often sufficient to dissuade such behavior. Therefore, transparency in research and publishing is essential.

Furthermore, we should look to technical solutions to tip practitioners towards honesty and ethical conduct. One possibility is timestamping manuscripts such that authors can prove that a given version existed at a given time. This makes attempts to retroactively revise previous work infeasible, because the pre-existing timestamps would become invalidated.

Authorship disputes can also be greatly diminished by tracking the history of manuscripts. Ambiguity in the historical record breeds disputes, turning miscommunication, misunderstandings, or interpersonal disputes into destructive conflicts. Alternatively, credit is silently deprived from the true creators of content, tending to favor established scientists to the detriment of early career or disadvantaged scientists. A transparent scholarly record means authorship can be precisely determined and displayed alongside research. Furthermore, timestamping makes retroactive tampering with the true authorship history infeasible.

Automate workflows

Publishing should be built on computational workflows that are error-free and automated. Presently journals and their submission systems require manual steps that silently introduce errors. In addition, the automated portions of the existing workflows are not adequately tested to ensure even basic formatting conversions are performed properly.

One salient example is how journals email authors a PDF proof to review, while imposing a mere 1 day deadline. However, the PDF contains unknown changes from the submitted document. Since the PDF is not easily machine-readable, it's infeasible to create a diff between the submitted document and the proof. Authors must scrutinize every word and character and compare it to the submission. Authors uncover a variety of mistakes ranging from introduced typos, editor mistakes, formatting issues, and other conversion fails. To suggest corrections, authors must annotate the PDF, itself a rickety endeavor. Yet, the journal ignores most of the author suggestions and publishes the error-ridden manuscript. Ironically, this was exactly our experience when publishing the software paper on Manubot in a journal. If only, the journal could have just published our meticulously formatted, machine-readable, standardized manuscript exactly as submitted!

None of this needs to be. By automating publishing with tested computational workflows, manuscripts can be published with lossless fidelity. When editorial changes are introduced, a clear record of those changes should make them easy for authors to review.

Deduplicate everything

Authoring should be DRY (don't repeat yourself). Information should be defined in one place, and one place only. For example, if an author's affiliation changes, there should be a single source field that gets updated. All other mentions of their affiliation should be automatically derived from this source, and automatically update as required.

Currently, the publishing process often requires information to be repeatedly defined, leading to stale content with no clearly correct, authoritative version. When submitting a manuscript to a journal, the journal should extract all required metadata from the submission, obviating the need to enter information into legacy submission workflow software such as Editorial Manager. By adopting DRY principles, we can ensure metadata and content is defined in the right place, such that submitting to multiple journals incurs no additional overhead in terms of repeatedly providing the same metadata.

Cite persistent identifiers

Citations should point to persistent identifiers (PIDs), such as DOIs, PubMed IDs, ISBNs, Wikidata IDs, URLs, etcetera. PIDs are the most succinct and stable method to unambiguously identify an external work. Currently, authors waste time retrieving metadata and formatting references, oftentimes relying on proprietary reference management software. In contrast, PIDs generally include sufficient metadata to fully populate the bibliography. And references can be automatically generated for thousands of existing bibliographic styles. Therefore by citing PIDs, authors reduce the busywork to minimum while ensuring citations remain unambiguous and machine-readable, supporting comprehensive open networks of scholarly citation.

One challenge of citation-by-PID currently is that metadata from databases such as Crossref or PubMed is sometimes incorrect. Therefore, when a user does manually update reference metadata, those updates should be shared. For example, the modifications could be automatically relayed to the publisher or reported to Crossref. In addition, community managed databases such as Wikidata can be updated. The end result is that only a single scholar must correct faulty metadata rather than all users in perpetuity.

Contextualize citations

Authors should have the ability to **annotate citations with context**. The citation graph is increasingly being used to curate, rank, and classify scholarly literature. However, not all citations are equal: some dispute the cited work, some affirm it, most do neither. Authors should have the ability to specify why they cite a particular work, preferably assigning a citation type from a standardized terminology [9].

Web-native interfaces

Article viewers and interfaces should take full advantage of the web. Web browsers are, and will continue to be, the primary means by which human readers access scholarly literature. Therefore, journals should adopt best practices for the web. This includes setting proper metadata such that articles can be properly indexed by search engines, shared on social media, and interpreted by accessibility tools (like text-to-speech utilities).

Furthermore, articles should be interactive. Hovering over a citation should bring up the full reference and highlight other instances of the same citation. Figures should be previewable in-line. Users should be able to go from a reference or figure to all of its citations/mentions. Sections should be collapsible. Cross-references between various elements should be linked and navigable.

While these features are intuitive, most journals apply sorely outdated interfaces. One contributing factor is that interfaces have not been portable between journals, due to their proprietary and nonmodular implementations. Open source frontends offer to bring modern interfaces to all journals that provide the underlying content in an interoperable way.

In the longterm, interfaces can provide access to not only a single article in an intuitive and interactive manner, but to the entire corpus of openly licensed literature. For example, it should be possible select an excerpt from one study and browse all other studies commenting on the same topic.

Machine readable

Manuscripts should be **machine readable**. While web browsers are the primary way in which humans now access literature, machines are quickly outpacing humans in the quest to collect, categorize, summarize, curate, and interpret literature. With the explosion in the amount of literature produced, machines are crucial to helping scholars find the right literature as well as build knowledge graphs that aid human investigation by scaling storage beyond the capacity of the brain. Machine-readable manuscripts should conform to standards and expose as much metadata as possible. In addition, manuscripts must be publicly accessible and openly licensed to be included in pre-processed text mining corpuses. Currently, there are large benefits to fulltext mining [10], but legal barriers require each research team to redo preprocessing, an often insurmountable barrier at scale.

Automate styling

Typesetting & styling should be automatic. Scholars writing a manuscript ought only to focus on the content. Aspects such as line spacing, margins, font types and sizing are orthogonal to content. Users should be able to transmit content (i.e. submitting to a journal or preprint server) without being burdened with formatting. Formatting can be automated. Publishing systems of the future will make it trivial to switch one style out for another style and for a journal to take a preprint and apply a branded style. Authors should have the ability to customize their style, but never should styling get in the way of producing and transmitting content [11].

Preserve & decentralize

Once published, **manuscripts should be preserved using persistent file hosting on decentralized networks**. The preservation of the scholarly record is too important to leave to a single centralized entity. Instead articles should be hosted by decentralized file storage networks, which allow entities all around the world to participate in hosting scholarly content. Therefore, access to articles won't shut off due to downtime or technical difficulties at the publisher's site [12]. Censorship will be more difficult, both in terms of preventing certain groups from either consuming scholarly literature or producing it. For example, it will be less feasible for political sanctions to prevent a population from participating in global scholarship [13,14,15].

References

1. **Reinventing Discovery**
Michael Nielsen
Princeton University Press (2011-12-31) https://doi.org/gfx2dm
DOI: 10.1515/9781400839452

2. **Scientific Utopia: I. Opening Scientific Communication**
Brian A. Nosek, Yoav Bar-Anan
Psychological Inquiry (2012-07) https://doi.org/gcsk27
DOI: 10.1080/1047840x.2012.692215

3. **Article Processing Charge Hyperinflation and Price Insensitivity: An Open Access Sequel to the Serials Crisis**
Shaun Yon-Seng Khoo
LIBER Quarterly (2019-05-09) https://doi.org/c6pc
DOI: 10.18352/lq.10280

4. **Cost models for running an online open journal**
Daniel S. Katz, Lorena A. Barba, Kyle E. Niemeyer, Arfon M. Smith
Journal of Open Source Software Blog (2019-06-04) https://blog.joss.theoj.org/2019/06/cost-models-for-running-an-online-open-journal

5. **Assessing the size of the affordability problem in scholarly publishing**
Alexander Grossmann, Björn Brembs
PeerJ (2019-06-18) https://doi.org/gf5ggz
DOI: 10.7287/peerj.preprints.27809v1

6. **The history of publishing delays**
Daniel Himmelstein
Satoshi Village (2016-02-10) https://blog.dhimmel.com/history-of-delays/

7. **Does it take too long to publish research?**
Kendall Powell
Nature (2016-02) https://doi.org/f3mn4t
DOI: 10.1038/530148a · PMID: 26863966

8. **Ten simple rules for researchers collaborating on Massively Open Online Papers (MOOPs)**
Jonathan Tennant, Natalia Z Bielczyk, Veronika Cheplygina, Bastian Greshake Tzovaras, Chris Hubertus Joseph Hartgerink, Johanna Havemann, Paola Masuzzo, Tobias Steiner
Center for Open Science (2019-07-02) https://doi.org/gf7k53
DOI: 10.31222/osf.io/et8ak

9. **FaBiO and CiTO: Ontologies for describing bibliographic resources and citations**
Silvio Peroni, David Shotton
Journal of Web Semantics (2012-12) https://doi.org/gf2tnn
DOI: 10.1016/j.websem.2012.08.001

10. **A comprehensive and quantitative comparison of text-mining in 15 million full-text articles versus their corresponding abstracts**
David Westergaard, Hans-Henrik Stærfeldt, Christian Tønsberg, Lars Juhl Jensen, Søren Brunak

PLOS Computational Biology (2018-02-15) https://doi.org/gcx747
DOI: 10.1371/journal.pcbi.1005962 · PMID: 29447159 · PMCID: PMC5831415

11. **Scientific sinkhole: The pernicious price of formatting**
Allana G. LeBlanc, Joel D. Barnes, Travis J. Saunders, Mark S. Tremblay, Jean-Philippe Chaput
PLOS ONE (2019-09-26) https://doi.org/gf84h5
DOI: 10.1371/journal.pone.0223116 · PMID: 31557272 · PMCID: PMC6763211

12. **Paywall Watch**http://www.paywallwatch.com/

13. **How US sanctions are crippling science in Iran**
Declan Butler
Nature (2019-09-24) https://doi.org/gf9r2n
DOI: 10.1038/d41586-019-02795-y · PMID: 31576034

14. **Update: In reversal, science publisher IEEE drops ban on using Huawei scientists as reviewers**
Jeffrey Mervis
Science (2019-06-03) https://doi.org/gf9r2p
DOI: 10.1126/science.aay2091

15. **Academic Censorship in China: The Case of The China Quarterly**
Mathew Y. H. Wong, Ying-ho Kwong
PS: Political Science & Politics (2019-01-07) https://doi.org/gf9r2m
DOI: 10.1017/s1049096518002093