

# Independent Publishing Based on the Values of Open Science

Axel Dürkop, Hamburg University of Technology (TUHH)

May 19, 2020

Open science is linked with many values such as accountability, integrity, trust, reproducibility and of course, openness. From the beginning of the research process, publishing is key to inform the community and society about the findings and the winding road that led to them. Saying this, accessibility must be mentioned as a necessary condition and another core value of open science.

To promote Open Science and its values, Hamburg University of Technology (TUHH)<sup>1</sup> and Hamburg State and University Library (SUB)<sup>2</sup> are developing an Open Access framework for digital publishing. We use open source tools that every scholar, library and publisher shall be able to use and adapt to his and her needs. The joint project of Modern Publishing is part of the program of Hamburg Open Science<sup>3</sup>.

## The Scholarly Writing and Publishing Framework

In our team we trust in the Unix philosophy<sup>4</sup> that says "One job, one tool." Like in Unix/Linux best practices we chain our tools together to build a modular pipeline that can be extended and configured as you like. Distinct parts can be replaced and adapted to ones needs. This modular architecture is open to participation and

---

<sup>1</sup> <https://www.tuhh.de/alt/tuhh/startpage.html>

<sup>2</sup> <https://www.sub.uni-hamburg.de/startseite.html>

<sup>3</sup> <https://openscience.hamburg.de/de/startseite-hamburg-open-science/>

<sup>4</sup> [https://en.wikipedia.org/wiki/Unix\\_philosophy](https://en.wikipedia.org/wiki/Unix_philosophy)

inspiration from the community. We call it a *Scholarly Writing and Publishing Framework*.

In short, the system can be described as: Authors write collaboratively in **Markdown**, a **GitLab** pipeline running with **Docker** converts the text with **pandoc/pandoc-scholar** to various output formats like PDF, HTML and XML. Reviewers annotate the article before submission by using **Hypothesis**. The article is submitted and published in an **OJS** instance and again can be annotated and discussed with Hypothesis..

Let's have a look at these tools in detail:

**Markdown.** In many open science and open source ecosystems the Markdown syntax for scientific writing has already become the default. Markdown can quickly be learned and can be found in many environments of the web. Markdown makes single-source publishing possible as it can be converted to many different formats with Static Site Generators<sup>5</sup> like Jekyll, Hugo or pandoc.<sup>6</sup>

**Pandoc.** Pandoc<sup>9</sup> is a converter for a large number of document types. It works on the command-line and converts Markdown to LaTeX, PDF, HTML, XML, office formats and various wiki dialects. To get an impression of its concept, you can try it online<sup>10</sup>. For our framework pandoc is the first choice as it meets the requirements for scientific writing including citations, footnotes, figures and reference management.

**pandoc-scholar.** For scholarly writing the correct mention of authors and affiliation is crucial. To equip pandoc with this requirement, Albert Krewinkel and Robert Winkler wrote pandoc-scholar<sup>11</sup> that we also use in our project (Krewinkel and Winkler 2017).<sup>12</sup> Krewinkel and Winkler also intend to reduce the expense of article processing with pandoc-scholar and can convert Markdown to PDF, LaTeX, HTML, JATS XML, epub and other formats. This way we can generate at the same time good-looking and machine readable versions of scientific articles from a single source.

---

<sup>5</sup> <https://www.staticgen.com/>

<sup>6</sup> Markdown e.g. is the way to write in Jupyter Notebooks<sup>7</sup> or document your projects on GitHub<sup>8</sup>. Grandesso (2018) also points to the advantages of writing and publishing with Markdown and inspired our work.

<sup>9</sup> <https://pandoc.org/>

<sup>10</sup> <https://pandoc.org/try/>

<sup>11</sup> <https://pandoc-scholar.github.io/>

<sup>12</sup> We are happy and lucky to have Albert Krewinkel in our team to further develop pandoc-scholar for the needs of the framework. All of his contributions will be open source. Public money, public code.

**GitLab.** Many scientists already appreciate collaboration on GitHub, a sharing platform for software, data and text. GitLab<sup>13</sup> is quite similar to GitHub, but open source software and free to host on a server of your choice. Besides the advantage of working together on code, data and text, GitLab excels with the feature of building digital artifacts with configurable pipelines<sup>14</sup>. For the *Scholarly Writing and Publishing Framework* we make extensive use of these pipelines with Docker.

**Docker**<sup>15</sup> is – in a nutshell – a technology that lets you build virtual machines as small as possible and as powerful as necessary. We use Docker to run pandoc/pandoc-scholar on Markdown files in GitLab pipelines to build scientific articles on the server, not on the authors' clients.

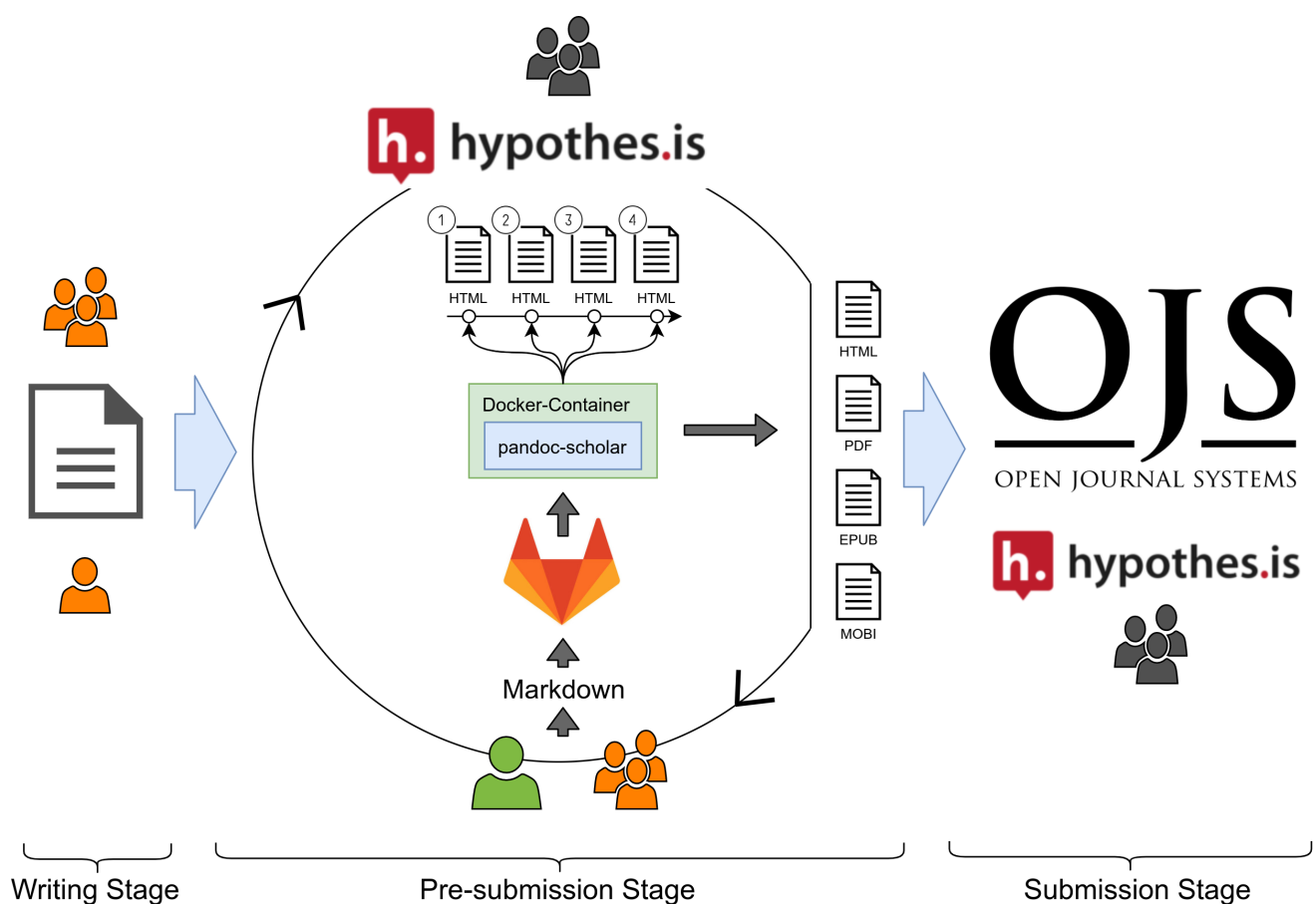


Figure 1: Architectural design of a socio-technical system for collaborative writing and publishing of scientific journal articles. Vertically, a GitLab pipeline produces HTML versions of the text based on branches. These versions can be (openly) peer-reviewed with Hypothesis. Horizontally, with every pipeline various output formats are generated that could immediately be published in OJS. Axel Dürkop, v0.4, 2019-06-11

<sup>13</sup> <https://docs.gitlab.com/ce/README.html>

<sup>14</sup> <https://docs.gitlab.com/ce/ci/pipelines/>

<sup>15</sup> <https://www.docker.com/>

**Hypothesis.** For (open) peer review of articles before and potentially also after submission we use the annotation tool Hypothesis<sup>16</sup>.

**Open Journal Systems.** An instance of Open Journal Systems (OJS)<sup>17</sup> is run by Hamburg University Press<sup>18</sup> at SUB. Our objective is to foster the foundation and migration of peer reviewed scholarly journals hosted by Hamburg University Press. We support scholars in experiencing state-of-the-art requirements for open access publishing such as ORCID or Crossref DOI, metrics, appropriate metadata as well as information on publishing ethics or good scholarly practise, but also services on indexing and sustainability. The API Magazine<sup>19</sup>, a student journal, is the first one published this year.

**Take a look at an example pipeline** In a file called `.gitlab-ci.yml` a GitLab pipeline can be configured. Here lies the source of power and creativity for the framework, as you can link software as you like to build what you need. To give you an example of a very simple pipeline configuration:

```
image: pandoc/latex

build:
  script:
    - pandoc draft.md -o draft.pdf
    - pandoc draft.md -o draft.html
  artifacts:
    paths:
      - "*.pdf"
      - "*.html"
```

The first line declares the Docker image to be used. Then, in the `script` part, any number of command lines can operate on the GitLab repository files. In this example, the file `draft.md` contains a scientific article that is converted by pandoc to PDF and HTML. The `artifacts` list enumerates all file types generated to be saved for further use before the Docker container is destroyed.

---

<sup>16</sup> <https://hypothes.is/>

<sup>17</sup> <https://pkp.sfu.ca/ojs/>

<sup>18</sup> <https://www.sub.uni-hamburg.de/service/publizieren/hamburg-university-press.html>

<sup>19</sup> <https://journals.sub.uni-hamburg.de/hup3/apimagazin>

With a configuration like this every change in `draft.md` leads to a new start of the pipeline to generate the desired artifacts. Thus, GitLab becomes a universal Content Management System (CMS) with great flexibility what to build. Artifacts then can be published with OJS.

## Potentials and challenges of the framework

In agreement with Herrmann (2003) we understand the *Scholarly Writing and Publishing Framework* as a socio-technical system. This helps us to keep in mind that complex technology has to be seen in the context of “[...] organisational, technical, educational and cultural structures and interactions” (Herrmann 2003, 60) and also points us to the needs and expectations of the users.

Speaking from a technical perspective, we see various potentials in this system. To mention some of the more important:

- We do not decide on the text editor to use. While Visual Studio Code<sup>20</sup> is a good choice for scientific writing with Markdown, Zettlr<sup>24</sup> could also be an editor you might like. A quick way to multi-author collaboration in a low-threshold Markdown writing environment can also be the browser tool HackMD<sup>25</sup> and its FLOSS sister CodiMD<sup>26</sup>.
- Based on the Docker image used, the pipeline can be configured to generate more complex artifacts like books (article in German)<sup>27</sup>, websites or even web applications that make use of the Markdown text. We use e.g. the framework also for our project blog<sup>28</sup> where we replace pandoc with Hugo<sup>29</sup> in the pipeline.
- With a complete FLOSS stack and software that is based on healthy developer communities we stay independent, modular and open for change.

---

<sup>20</sup> In our team we prefer the unbranded and telemetry-free FLOSS version VSCode<sup>21</sup>. Check out the extensions Pandoc Citer<sup>22</sup> and Markdown Preview Enhanced<sup>23</sup> for writing conveniently with Markdown.

<sup>24</sup> <https://www.zettlr.com/>

<sup>25</sup> <http://hackmd.io/>

<sup>26</sup> <https://github.com/codimd>

<sup>27</sup> <https://oa-pub.hos.tuhh.de/de/2019/08/19/kollaborativ-buecher-schreiben-mit-dem-gitlab-wiki/>

<sup>28</sup> <https://oa-pub.hos.tuhh.de/>

<sup>29</sup> <https://gohugo.io/>

Speaking from a social perspective, we also see various potentials and challenges in this system:

- Depending on your background, getting started with the framework means a more or less steep learning curve. We think it's a worthwhile investment spending the time, as we chain together concepts and tools that are for themselves helpful and valuable in other research and education contexts as well.
- Involving authors and editors in our work, we learn about the writing habits of authors and usability aspects of new publishing systems. The framework supports collaborative writing processes and open peer review, thus supporting open science in its core sense.
- We know that the characteristics of the framework heavily relate to software development culture. This is good to easily onboard writers that come from a natural or computer science background. For scholars from other disciplines like the humanities and social sciences we think writing texts in a software developer environment could flatten the learning curve also for these tools. Furthermore, it supports adopting to the new paradigm of opening up the research process.

By saying this and following Herrmann, we assume that developing the framework underlies the characteristics described as "reciprocal indispensability," "reciprocal forming" and "ubiquitous self-description" (Herrmann 2003, 63). In short this means, that the social and technical system are joined together like two sides of a piece of paper and influence each other in the process of development and usage.

## Outlook

The technical stack depicted here will be presented and released as a modular bundle and open source in the context of Open-Access-Days 2020<sup>30</sup>. Till the end of 2020 we will concentrate on optimizing the JATS XML output of pandoc/pandoc-scholar to fit the requirements of OJS. We also will join the discourse of using or developing a meta data set for authors and affiliations. Also, we are optimizing the templates, write a documentation for others to understand what we're doing and test the framework with authors and editors.

---

<sup>30</sup> <https://open-access.net/en/community/open-access-tage/open-access-tage-2020-bielefeld>

## About the Author

Axel Dürkop has his roots in philosophy and theatre. He works for Hamburg Open Online University (HOOU)<sup>31</sup> at the Institut für Technische Bildung und Hochschuldidaktik<sup>32</sup> at Hamburg University of Technology (TUHH)<sup>33</sup> shaping open education and designing resources. He is team leader and developer for the project Modern Publishing<sup>34</sup> at TUHH Library<sup>35</sup> in the program Hamburg Open Science<sup>36</sup>. Further information also on Axel's website<sup>37</sup>.

This article was first published under the same title May 19, 2020 on Generation Research, <https://doi.org/10.25815/kx86-sx58>, under CC BY-SA 4.0<sup>38</sup>.

## References

- Grandesso, Piero. 2018. "A Pandoc-Based Layout Workflow for Scholarly Journals." Personal blog. Piero G. March 23, 2018. <http://pierog.it/en/2018/03/markdown-workflow/>.
- Herrmann, Thomas. 2003. "Learning and Teaching in Socio-Technical Environments." In *Informatics and the Digital Society: Social, Ethical and Cognitive Issues*, 59–71. Boston, Massachusetts: Kluwer Academic Publishers. [https://doi.org/10.1007/978-0-387-35663-1\\_6](https://doi.org/10.1007/978-0-387-35663-1_6).
- Krewinkel, Albert, and Robert Winkler. 2017. "Formatting Open Science: Agilely Creating Multiple Document Formats for Academic Manuscripts with Pandoc Scholar." *PeerJ Computer Science* 3 (May): e112. <https://doi.org/10.7717/peerj-cs.112>.

---

<sup>31</sup> <https://www.hoou.de/>

<sup>32</sup> <https://itbh-hh.de/en/>

<sup>33</sup> <https://www.tuhh.de/alt/tuhh/startpage.html>

<sup>34</sup> <https://oa-pub.hos.tuhh.de/en/>

<sup>35</sup> <https://www.tub.tuhh.de/en/>

<sup>36</sup> <https://openscience.hamburg.de/de/startseite-hamburg-open-science/>

<sup>37</sup> <https://axel-duerkop.de/en/>

<sup>38</sup> <https://creativecommons.org/licenses/by-sa/4.0/>