

# Announcing Commonmeta

Martin Fenner 

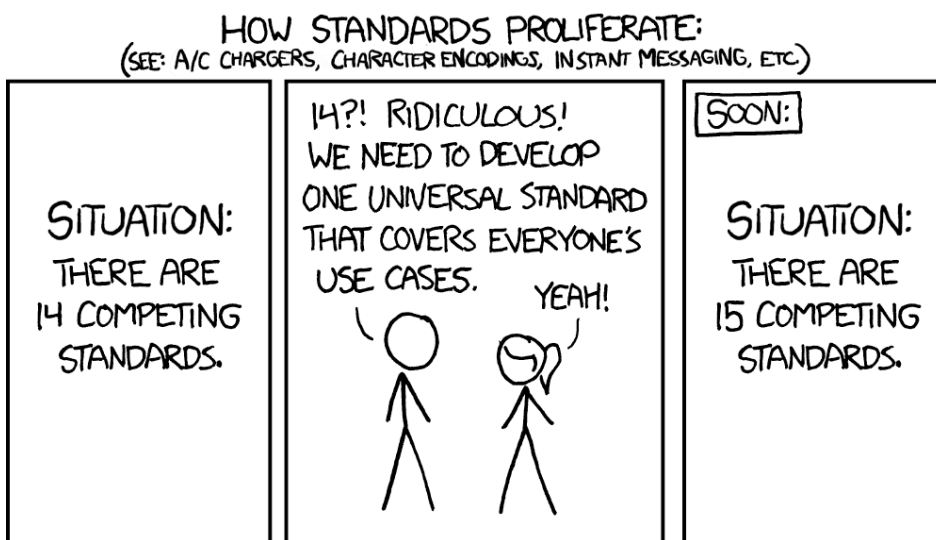
Published March 9, 2023

## Citation

Fenner, M. (2023). Announcing Commonmeta. *Front Matter*. <https://doi.org/10.53731/cp7apdj-jk5f471>

## Keywords

Metadata



## Copyright

Copyright © Martin Fenner 2023. Distributed under the terms of the [Creative Commons Attribution 4.0 International License](https://creativecommons.org/licenses/by/4.0/), which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

## Front Matter

This week I launched **Commonmeta**, a new scholarly metadata standard described at <https://commonmeta.org>. Commonmeta is the result of working on conversion tools for scholarly metadata for many years. One conclusion early on was that these conversions are many-to-many, so it becomes much easier to have an internal format that is the intermediate step for these conversions.

Commonmeta is inspired by two initiatives: [Codemeta](#) and [Commonmark](#). CodeMeta contributors are creating a minimal metadata schema for science software and code, in JSON and XML. The goal of CodeMeta is to create a concept vocabulary that can be used to standardize the exchange of software metadata across repositories and organizations. Commonmark is a strongly defined, highly compatible specification of Markdown, along with a suite of comprehensive tests to validate Markdown implementations against this specification.

These two specifications not only inspired the name but also the principles of how I want to see Commonmeta operate:

- driven by real-world implementations and not committees
- features that focus on what is common in existing implementations/formats
- a testable specification

The website goes into a little bit more detail about why I didn't pick any the existing standards but instead came up with a new metadata standard. This is a familiar pattern made famous by the XKCD comic shown above.

As I want this to be driven by real-world implementations and not committees, I also in the last few weeks launched [commonmeta-py](#), a Python implementation of the standard available on [PyPi](#). And in a few months, I hope to have tweaked the [Ruby Gem](#) that I originally wrote a few years ago to support Commonmeta as the internal format.

With testable specification, I mean both a JSON Schema to describe Commonmeta and many, many tests that validate the conversions with real-world data. The JSON Schema is available [here](#), and will become stable once it reaches version 1.0. `commonmeta-py` comes with lots of tests, but I hope to further improve the test coverage.

Please reach out to me if you want to help with Commonmeta, in particular, work on implementations in other languages, such as Javascript, PHP, or Java.