PyRDM: A library to facilitate the automated publication of software and data in computational science

Christian T. Jacobs[1], Alexandros Avdis[1], Gerard J. Gorman[1], Matthew D. Piggott[1]

[1] Department of Earth Science and Engineering, Imperial College London.

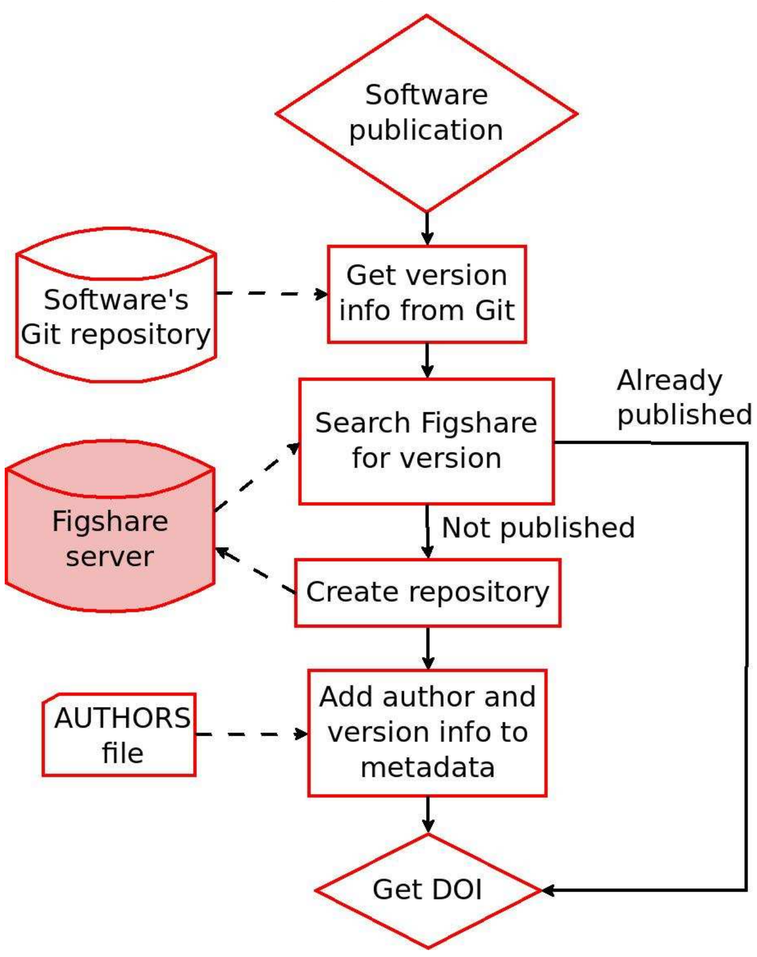
1. Overview

Computational science workflows are characterised by the interactions between both software and data. Collected data (e.g. bathymetry, atmospheric conditions) is frequently provided as input to scientific software (e.g. a numerical model). This software then produces 'output data' (e.g. oceanic flow fields) which is analysed by the researcher to yield scientific results. In order to achieve reproducible results or indeed recomputability of the raw data itself, the software, input data and output data should all be captured along with any provenance metadata. However, these components are often not published with the main findings in journal articles.

This work presents a new open-source library, called PyRDM (github.com/pyrdm), whose functionality aims to facilitate the sharing of software and data via online, persistent and citable repositories by introducing a large amount of automation into the curation process. This in turn can further motivate researchers and encourages re-use of research outputs through a more open and easy-to-use scientific workflow.

2. PyRDM functionality

The PyRDM library is able to automatically publish software source code and data to online repositories provided by Figshare, Zenodo and DSpace-based services. In return, these repositories yield a DOI or Handle for formal and proper citation.

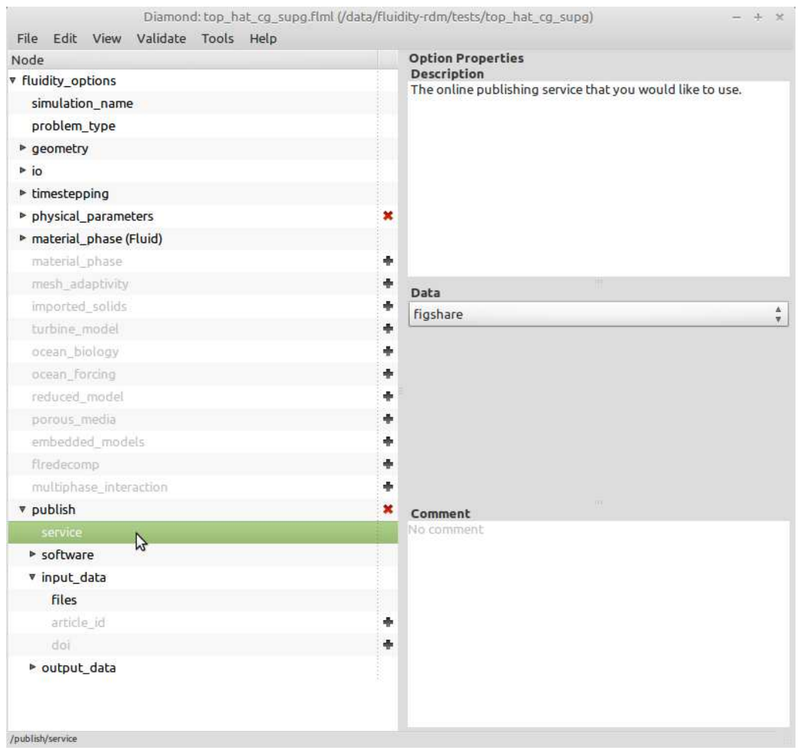


In the case of datasets, users only need to specify the files (e.g. `*.dat`) they would like published. MD5 checksums are used to selectively re-upload only those files (in existing repositories) that have been modified locally.

Since there are many possible workflows, it is not feasible to *fully* automate the publishing process with PyRDM alone. The library may therefore be viewed as more of a developer's toolkit for producing a publishing tool that is tailored towards the specific workflow being considered.

3. PyRDM application

PyRDM has been integrated into the workflow of Fluidity (fluidity-project.org), an open-source computational fluid dynamics code. Users can enable a 'publish' option in their simulation's setup file, as shown in the screenshot below:

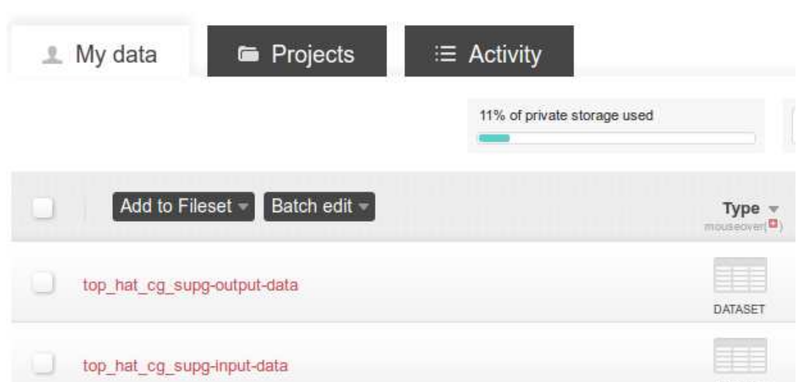


After the simulation has been performed, the user then runs a Fluidity-specific publishing tool which uses the PyRDM library. Only a small amount of information needs to be provided:

- Authentication details for Figshare, Zenodo and/or DSpace.
- A list of the data files they want to publish.
- Optionally: an existing publication ID and DOI.

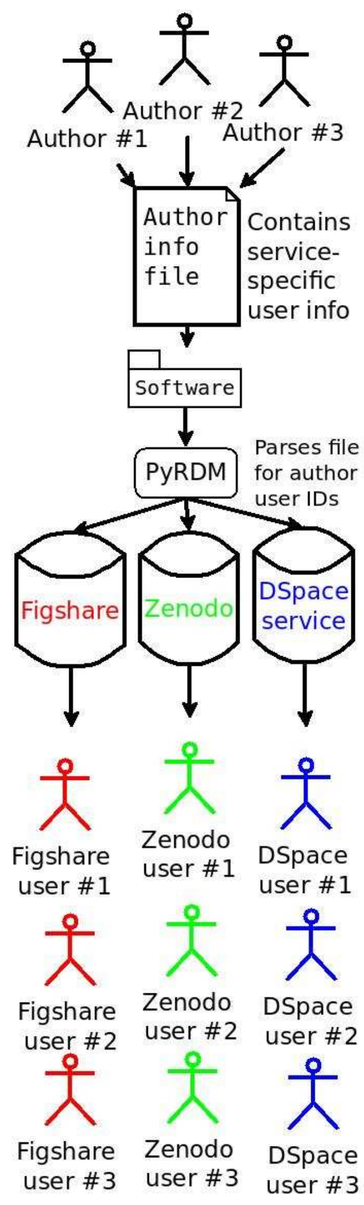
The source, input and output data are published to separate repositories. The DOIs minted for each one are stored in the simulation's setup file, so data can be updated at a later time, and also in the simulation's metadata:

```
<constant name="FluidityVersion"
type="string" value=
"1baf80aac1e7e735b1cf182bc20761a0c6df7767"/>
<constant name="SoftwareDOI"
type="string" value=
"http://dx.doi.org/10.6084/m9.figshare.1035081"/>
<constant name="InputDataDOI"
type="string" value=
"http://dx.doi.org/10.6084/m9.figshare.1035083"/>
<constant name="CompileTime" type="string"
value= "May 23 2014 15:22:23"/>
<constant name="StartTime" type="string" value=
"20140523 154857.775+0100"/>
```

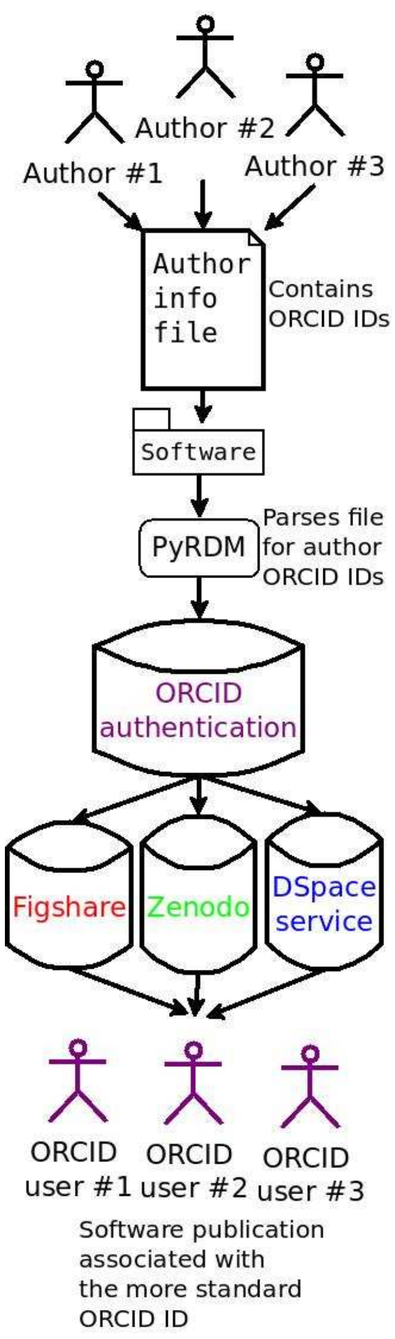


4. Current issues and limitations

Attempting to automatically affiliate software authors to an online repository can prove difficult due to lack of standardisation. In the near future, it is hoped that support for authenticating via ORCID (orcid.org) and using an ORCID ID when publishing via the Figshare/Zenodo/DSpace API will be added:



Software publication associated with service-specific user ID/account.



Software publication associated with the more standard ORCID ID

A lack of API support also exists at the 'publishing' level. For example, it is currently not possible to obtain server-side MD5 checksums with Figshare, or search for repositories with the Zenodo API. Further developments are necessary in this area to make the publication process smoother.

Some research involves proprietary and/or private data which cannot be shared, but at the same time digital curation is important for funders of the research. Figshare offers limited free private storage space for individual users, but this is not generally large enough to store complete modern day simulations. The number of collaborators who can view/modify the private data is also limited. Figshare for Institutions, which gives members of an institution private cloud storage, may be a more suitable platform for larger-scale research data management.

5. Acknowledgements and References

CTJ was funded by an internal grant entitled "Research data management: Where software meets data" from the Research Office at Imperial College London.

- Jacobs et al. (2014), DOI: 10.5334/jors.bj
- Piggott et al. (2008), DOI: 10.1002/fld.1663
- Figshare blog: figshare.com/blog
- Symplectic updates: symplectic.co.uk/elements-updates

AMCG

c.jacobs10@imperial.ac.uk
http://ctjacobs.github.io

Imperial College London