Open Journal Systems and Dataverse Integration-- Helping Journals to Upgrade Data Publication for Reusable Research

Micah Altman
Director of Research, MIT Libraries
<escience@mit.edu>
http://informatics.mit.edu

Eleni Castro
Research Coordinator, IQSS, Harvard U.
<ecastro@fas.harvard.edu>
http://www.iq.harvard.edu/people/eleni-castro

Mercè Crosas
Director of Data Science, IQSS, Harvard U.
<mcrosas@iq.harvard.edu>
http://scholar.harvard.edu/mercecrosas

Philip Durbin
Software Developer, IQSS, Harvard U.
<philip_durbin@harvard.edu>
http://www.iq.harvard.edu/people/philip-durbin

Alex Garnett
Data Curation & Digital Preservation Specialist, Simon Fraser University
<axfelix@gmail.com>
http://webambler.com/

Jen Whitney
Systems Librarian, Carleton College
<jen.j.whitney@gmail.com>
https://ca.linkedin.com/in/jenwhitney

Abstract

This article describes the novel open source tools for open data publication in open access journal workflows. This comprises a plugin for Open Journal Systems that supports a data submission, citation, review, and publication workflow; and an extension to the Dataverse system that provides a standard deposit API. We describe the function and design of these tools, provide examples of their use, and summarize their initial reception. We conclude by discussing future plans and potential impact.

Introduction

Academic libraries have been increasingly engaged in the open access movement and in stewarding content published as open access. Increasingly, funders have taken note of this movement -- and a slew of policies issued by federal agencies and by foundations have incentivised both open access publishing. Furthermore, such policies increasingly require access to research data.

Thus, libraries increasingly have the opportunity to engage with open access, publishing, and open data. These are all complex areas and, although overlapping, have not heretofore been integrated in a tangible and automated way. This project for the first time provides a framework and operational tool to integrate open access journal publishing, data publishing, and data preservation.

Over the last two years, the Dataverse Project (originally named, "Dataverse Network") (King 2007; Crosas, 2011, 2013) from Harvard University collaborated with the Public Knowledge Project (PKP) (MacGregor, Stranack & Willinsky, 2014) from Simon Fraser University and Stanford University on an open source project, funded by the Alfred P. Sloan Foundation, to enable open access journals to seamlessly manage the submission, review, and publication of data associated with published articles. The result has been the development of software that journal editors of journals using PKP's Open Journal Systems' (OJS) (Willinsky, 2005) can easily install and use to adopt appropriate data replication, citation, archiving and publication policies. This goal has been achieved -- through this project OJS now supports a complete data publishing workflow that is shipped as part of the standard OJS distribution and which automatically and seamlessly integrates with Dataverse, through a deposit API, for long-term access to research data associated with published articles.

The project resulted in two main outputs: a plugin for OJS that supports a data submission, citation, review, and publication workflow; and an extension to Dataverse that provides a standard deposit API. These extensions have been released in production and are integrated into development and maintenance. Furthermore, these extensions are being used to support interoperability with other systems such as the Center For Open Science's Open Science Framework (OSF) and Archivematica starting in the summer of 2015.

Extending Dataverse with SWORD

The Dataverse Project (King, 2007; Crosas, 2011, 2013) started in 2006 at Harvard's Institute for Quantitative Social Science (IQSS) as an open source software application to share, preserve, cite, explore and analyze research data. This effort was the continuation of a substantial sequence of development efforts at IQSS, Harvard and beyond (Altman, et al 1999; Altman 2002) From its beginnings, Dataverse (then referred as the 'Dataverse Network') has provided a robust infrastructure for data stewards to host and archive data, while offering

researchers an easy way to share and get credit for their data. Since then, there are now over ten Dataverse repositories that share metadata with each other, hosted in institutions around the world, which together serve more than 58,000 datasets with 750,000 data files (http://dataverse.org). These Dataverse repositories are using the Dataverse software in a variety of ways -- from supporting existing large data archives, to building institutional subject specific repositories, to use as institutional repositories. One of these Dataverse repositories is the Harvard Dataverse (https://dataverse.harvard.edu), that alone hosts more than 1,000 dataverses (containers of datasets) owned and managed by researchers, research groups, organizations, departments and journals. The Harvard Dataverse has so far served more than a million downloads of its datasets, allowing researchers around the world to reuse the data, discover new findings, and extend or verify previous work. While the Dataverse Project started from the social sciences for the social sciences, with the recent version 4.0 release it has now expanded to benefit a wide range of disciplines and scientific domains (astronomy, life sciences, etc), leveraging its progress in the social science domain to define and enhance data publishing across all research communities.

Through this project, we have enabled Dataverse to federate with other systems not only as a data provider but also as a data store. Prior to this project, Dataverse did not provide a standard API– all deposits were accomplished through GUI interfaces or through batch processes executed on the Dataverse server itself. Implementing a data deposit API enables the OJS system to federate with any instance of the Dataverse software in a standard, seamless way.

For interoperability, we selected the SWORD2 protocol (Stevenson, 2009). *SWORD* (Simple Web-service Offering Repository Deposit) is a protocol specification used by digital repositories to accept the deposit of content from multiple sources in different formats. Repositories such as ArXiv, DSPACE, Fedora, and Microsoft Zentity support SWORD. Through *SWORD*, data files, documentation, supporting materials and its corresponding descriptive metadata can be deposited into the Dataverse system.

We implemented version 2.0 of the SWORD protocol, which was originally developed by a collaboration of JISC funded efforts and other non-profits. SWORD2 supports a full set of *CRUD* (create, read, update, delete) operations – and thus provides a very general way of interacting with Dataverse. As a result, it has enabled other projects to more readily federate with Dataverse, including the (Sloan supported) Open Science Framework and the ROpenSci package for R-language integration with Dataverse.

Dataverse implements a full SWORD deposit interface. The core supported functions are listed below:
- Retrieve SWORD service document
- Create a dataset with an Atom entry
- Dublin Core Terms (DC Terms) Qualified Mapping - Dataverse DB Element Crosswalk
- List datasets in a dataverse
- Add files to a dataset with a zip file

- Display a dataset atom entry
- Display a dataset statement
- Delete a file by database id
- Replacing metadata for a dataset
- Delete a dataset
- Determine if a dataverse has been published
- Publish a dataverse
- Publish a dataset

The SWORD API is easily accessed through standard libraries or command line clients such as CURL. For example, the following command will list all datasets in a dataverse:

```
curl -u $API_TOKEN:
https://$HOSTNAME/dvn/api/data-deposit/v1.1/swordv2/collection/dataverse/$DATAVERSE_ALIAS
```

Or, to create a dataset, using an ATOM entry:

```
curl -u $API_TOKEN: --data-binary "@path/to/atom-entry-study.xml" -H "Content-Type:
application/atom+xml"
https://$HOSTNAME/dvn/api/data-deposit/v1.1/swordv2/collection/dataverse/$DATAVERSE_ALIAS
```

Example Atom entry (XML)

```
<?xml version="1.0"?>
<entry xmlns="http://www.w3.org/2005/Atom"
       xmlns:dcterms="http://purl.org/dc/terms/">
    <!-- some embedded metadata -->
    <dcterms:title>Roasting at Home</dcterms:title>
    <dcterms:creator>Peets, John</dcterms:creator>
    <dcterms:creator affiliation="Coffee Bean State University">Stumptown,
Jane</dcterms:creator>
    <!-- Dataverse controlled vocabulary subject term -->
    <dcterms:subject>Chemistry</dcterms:subject>
    <!-- keywords -->
    <dcterms:subject>coffee</dcterms:subject>
    <dcterms:subject>beverage</dcterms:subject>
    <dcterms:subject>caffeine</dcterms:subject>
    <dcterms:description>Considerations before you start roasting your own coffee at
home.</dcterms:description>
    <!-- Producer with financial or admin responsibility of the data -->
    <dcterms:publisher>Coffee Bean State University</dcterms:publisher>
```

```
   <dcterms:contributor type="Funder">CaffeineForAll</dcterms:contributor>
   <!-- production date -->
   <dcterms:date>2013-07-11</dcterms:date>
   <!-- kind of data -->
   <dcterms:type>aggregate data</dcterms:type>
   <!-- List of sources of the data collection-->
   <dcterms:source>Stumptown, Jane. 2011. Home Roasting. Coffeemill
Press.</dcterms:source>
   <!-- related materials -->
   <dcterms:relation>Peets, John. 2010. Roasting Coffee at the Coffee Shop. Coffeemill
Press</dcterms:relation>
   <!-- geographic coverage -->
   <dcterms:coverage>United States</dcterms:coverage>
   <dcterms:coverage>Canada</dcterms:coverage>
   <!-- license and restrictions -->
   <dcterms:license>NONE</dcterms:license>
   <dcterms:rights>Downloader will not use the Materials in any way prohibited by
applicable laws.</dcterms:rights>
   <!-- related publications -->
   <dcterms:isReferencedBy holdingsURI="http://dx.doi.org/10.1038/dvn333" agency="DOI"
IDNo="10.1038/dvn333">Peets, J., &amp; Stumptown, J. (2013). Roasting at Home. New
England Journal of Coffee, 3(1), 22-34.</dcterms:isReferencedBy>
</entry>
```

During the project, SWORD support has been prototyped, beta-tested, released, and updated. We also contributed minor extensions to the SWORD protocol and codebase, which has been accepted by the SWORD maintainers (http://swordapp.org/sword-v2/sword-v2-implementations/).

This API is now fully integrated into the current Dataverse release, and is a fully supported and maintained module in the Dataverse codebase. We have also deployed it in production, and it is being actively used to integrate OJS journals with the Harvard Dataverse instance.

SWORD does impose some practical limitations. The most significant has to do with metadata -- the standard implementation of the SWORD library uses simple dublin core terms for metadata, which are considerably less rich than the Data Documentation Initiative (DDI) schema used by

Dataverse.[1] A typical way of approaching this to include metadata in a packaging format, such as METS or MPEG DIDL. However the SimpleZip packaging format used by OJS and Dataverse does not have strong support for embedded metadata. To bridge this gap, project developers worked with the SWORD community to extend the API to support the addition of attributes to DC terms[2]: This enables richer metadata to be used directly in the client-server transaction, and uses this as the foundation for a more detailed translation to the to DDI.

Notwithstanding, Dataverse 4.0 supports additional API's that are often more convenient for direct access to and manipulation of individual data files. Dataverse also supports other API's such as OAI-PMH for harvesting of information; LOCKSS manifests and crawler API's for preservation; and a set of native JSON API's for fine search and fine-grained deposit and access. For detailed information on all Dataverse API's see: http://guides.dataverse.org/en/latest/api/index.html .

Designing an OJS Dataverse Plugin for Data Publication

Since 2001, PKP (MacGregor, Stranack & Willinsky, 2014) maintains and develops OJS, which includes developers, librarians and library staff at Simon Fraser University, Stanford University, and elsewhere. OJS is an open source journal management and publishing system which provides an online platform for journal editors to accept submissions, perform double blind peer review, edit, publish, and disseminate journal articles. PKP makes OJS freely available with the "purpose of making open access publishing a viable option for more journals" (OJS website: https://pkp.sfu.ca/ojs/). More recently, PKP has also developed other open source tools for managing conferences (Open Conference Systems (OCS)), managing book publications (Open Monograph Press (OMP)), and metadata harvesting (Open Harvester Systems (OHS)).

Although OJS allows for the publication of supplementary files alongside articles, it previously had no special provision for the review, publication, or presentation of common research data formats. This integration facilitates data sharing and archiving, allowing authors to be able to deposit their research data at the same time as their article. This streamlines the data submission process and provides an indexed, permanent identifier for published data. In keeping with OJS' focus on a robust scholarly workflow, Dataverse integration allows for consistent editorial policies to be applied to the submission and exposure of supplemental data; for example, data may or may not be subject to peer review, and in some cases it may be desirable to publish and provide an identifier to a dataset even when its corresponding article is rejected. It also allows embargoes of content as appropriate. Users are also able to configure whether they wish to make article data available immediately upon acceptance of an article by a

---

[1] The recently released version of Dataverse, 4.0 uses an internal schema, but still supports DDI for export and import.

[2] These extensions, along with bug fixes to the SWORD PHP library were accepted into the community supported SWORD codebase.

journal editor or to wait until the article itself is published, enabling faster-moving scholarly workflows.

The diagram below illustrates an information lifecycle workflow, which we envisioned for this project and demonstrates what a generalized ideal automated and integrated journal and data publishing workflow would look like (Castro, 2015, also see CODATA 2014 for more details on the data lifecycle). This type of workflow: enables journal editors and reviewers the ability to seamlessly manage the submission, review, and publication of data associated with published articles; and streamlines the author's article and corresponding data deposit process by automating deposit from OJS into Dataverse at the time of journal publication or journal article acceptance.
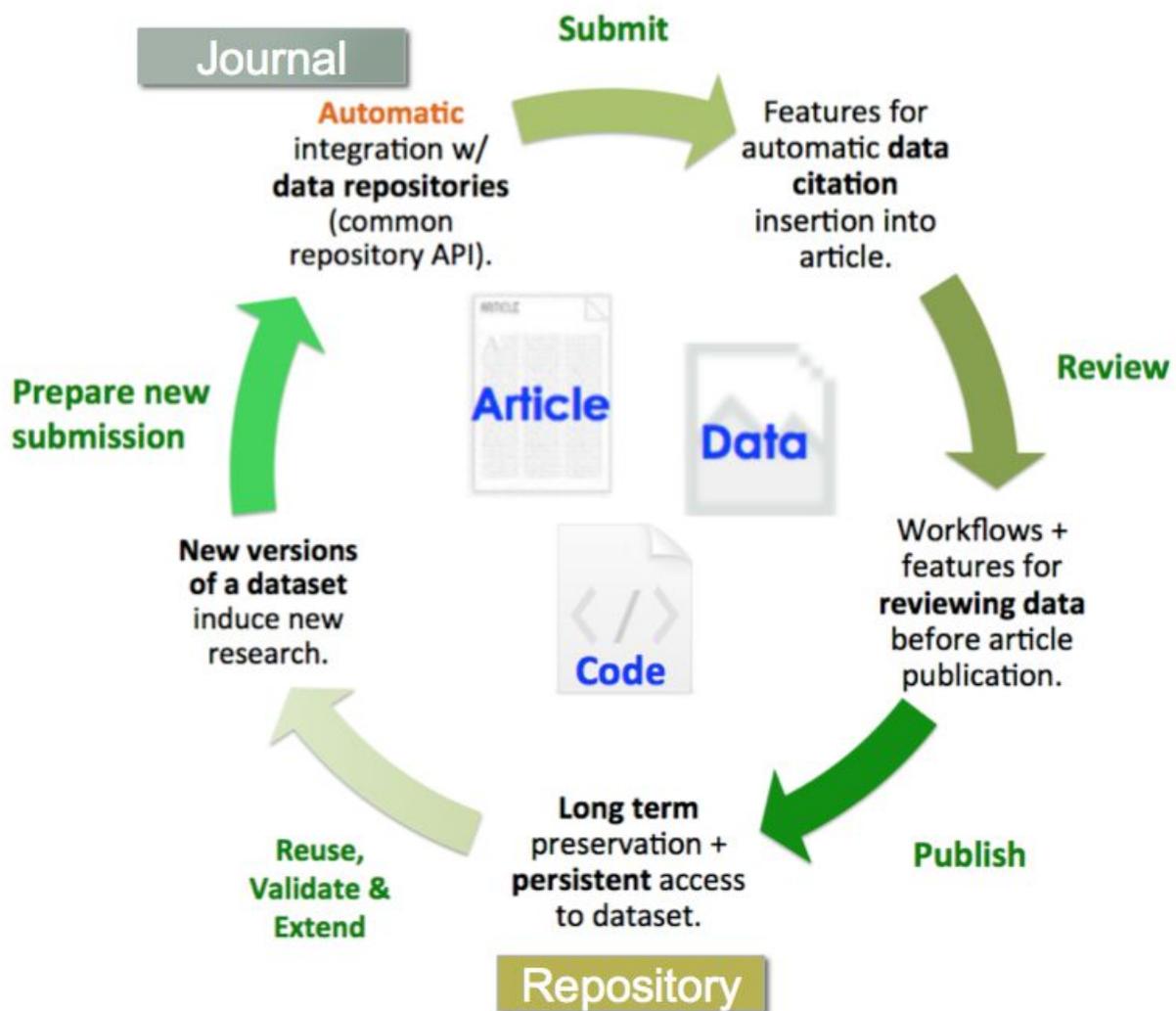
Figure 1. Lifecycle of an automated and integrated journal and data publishing workflow.

How does it work?

To make data sharing and archiving as easy as possible for authors, data files are deposited in conjunction with the journal's article submission process, resulting in a permanent 2-way linking between an article and its corresponding data. The full supported workflow is illustrated below:
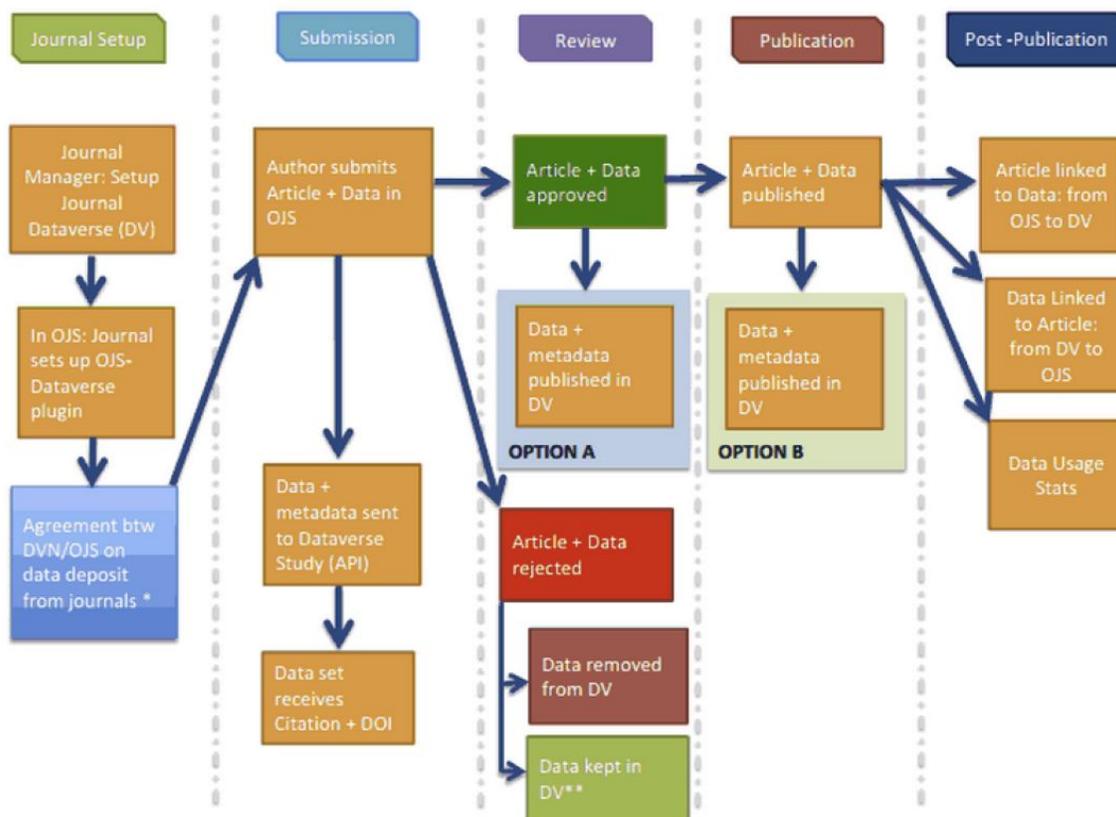
Figure 2. Data submission, review and publication workflows.

There are four major roles addressed by these workflows:

1. The **Author** submits their article and research data to the journal's OJS article submission system. (Note that the article and data do not have to be submitted at the same time. Authors can also submit data at a later time, or they can just provide a persistent link with a data citation pointing to the repository that their data is currently in.)
2. **Editors and/or Peer Reviewers** review the article and data.
3. If the article and corresponding research data are approved for publication, the **Authors'** research data and its corresponding metadata is automatically deposited from OJS into the Dataverse through the API. No redundant information need be entered. A permanent

identifier (DOI) will be automatically included that allows the data to be cited and tracked. There will be a data citation included in the journal article page in OJS (and ideally within the Reference section of the article) enabling readers of the article to quickly access the data.

4. The Dataverse stores the dataset metadata and files (including raw data, documentation, code, etc). There will also be a permanent publication citation link within the Dataverse for **researchers** to access the article in OJS that corresponds to this research data.

Editors, authors, and reviewers interact with the system through extensions to the OJS Graphical User Interface. For example, editors can set the data deposit and citation policies for journals, configure data review workflows, and designate a preferred repository using the following configuration dialogs:

Figure 3a. Editorial configuration dialog, part 1.

Edit Review Guidelines to include instructions for reviewing datasets.

▸ Example review guidelines   Show example guidelines for reviewing datasets

Copyedit Instructions

Edit Copyedit Instructions to include instructions for datasets.

▸ Example copyedit instructions   Show examples of copyediting instructions for datasets

Terms of Use

Provide the terms of use to which users must agree before depositing data files.

○ Fetch terms of use from the Dataverse set up for this journal.
◉ Enter custom terms of use in the field below.

Here are some terms

✂ 🗎 📋 | **B** *I* U̲ ⊟ ⊟ | ∞ ✂ 🌐 HTML ▦ ⬛

Metadata Settings

Citation format   [APA Citation Style ▾]
*

Choose format of publication citation displayed in study cataloguing information.

Public identifier   [▾]
type

Choose type of public identifier to link to article from publication citation. If no public identifier plugins are enabled, citations for published articles will use OJS URLs.

Workflow Settings

Require data   ☐   Require submissions to include data files deposited in Dataverse.
files

Release draft   [Editor accepts submission ▾]
studies

[Save] [Cancel]

* Denotes required field

Figure 3b. Editorial configuration dialog, part 2.

Authors submit data through OJS as part of the standard supported manuscript submission workflow:

Supplementary File Metadata

To index this supplementary material, provide the following metadata for the uploaded supplementary file.

Title *	Untitled
Creator (or owner) of file	
Keywords	
Type	Research Instrument
	Specify other
Brief description	

X   B   I   U   ≡   ≔   HTML

Publisher	
	Use only with formally published materials.
Contributor or sponsoring agency	
Date	2015-02-24 YYYY-MM-DD
	Date when data was collected or instrument created.
Source	
	Name of study or other point of origin.
Language	
	English=en; French=fr; Spanish=es. Additional codes.

Dataverse Plugin: Publish Data

Research data can be submitted for publication by depositing supplementary files in the journal's Dataverse or by supplying a citation to data already in a repository.

● Treat uploaded file as supplementary file. File **will not** be deposited in Dataverse.

○ Deposit file in Dataverse. Author accepts and agrees to Dataverse terms of use.

Study Description

Provide a summary of the purpose, nature, and scope of study data.

External Data Citation

If the data to be published with this submission is already in a different repository, please provide a data citation.

Supplementary File

File Name	26-109-1-SP.csv
Original file name	ca-500.csv
File Size	86KB
Date uploaded	2015-02-24 04:34 PM

☐ Present file to reviewers (without metadata), as it will not compromise blind review.

Replace file	Choose File No file chosen	Use Save to upload file.

Save and continue   Cancel

Figure 4. Author submission interface.

After publication, published data can be revised and updated using the standard Dataverse interfaces, as illustrated below:

Figure 5. Sample Data Management Interface.

Development work was jointly undertaken by IQSS and PKP. IQSS development resources were mostly dedicated to implementing, then expanding and documenting, the Dataverse API; OJS developers focused on the creation of the OJS Dataverse plugin itself. Both teams contributed project management expertise, consultation with their respective communities, and presentation / promotion opportunities. Both teams also benefited indirectly from this development work through the expansion of generic SWORDv2 API functionality in both OJS and Dataverse.

In designing the Dataverse plugin for OJS, care was taken to ensure that as many administrative functions as possible could be performed from a single interface of a user's choosing (e.g. electing to release or restrict data in Dataverse from the OJS side), and that any file or metadata updates made in the OJS interface after the time of publication would be properly reflected in Dataverse. Additionally, a schema was developed to maintain compatibility between Dataverse's native support for Data Documentation Initiative (DDI) metadata and OJS' Dublin Core metadata; this schema is also now openly available, and is sufficiently generic to be

useful outside of the OJS/Dataverse context. Through the use of the OJS Dataverse plugin, OJS is also capable of automatically "mirroring" data terms of use and other related access policies from a linked Dataverse in its existing policy text blocks. All of this was done in order to permit users to remain in a single coherent interface and to make the integration between the two platforms as natural as possible.

As of OJS 2.4.4, released in May 2014, the Dataverse integration functionality is distributed in the core set of OJS plugins, so that it can be enabled with a few clicks by an administrator, provided that there is a corresponding Dataverse to associate. There have been two full OJS point-release cycles (2.4.5 in September 2014 and 2.4.6 in March 2015) since then, and both have included minor workflow improvements (and other bug fixes) suggested by the Dataverse and OJS user communities following on from the initial release. The OJS Dataverse plugin is also compatible with OJS as far back as version 2.3.0, and with both the stable, widely used Dataverse 3.6.x branch and the recently released Dataverse 4.0.

Dissemination

The OJS Dataverse plugin that constitutes the main user interface component of this work is now integrated into the main OJS development branch and will be maintained by PKP throughout future releases, providing support for Dataverse API changes and other minor improvements. With respect to Dataverse, all of the functionality is now incorporated in the main Dataverse codebase, bundled (modularized) into the SWORD API support, and will be maintained in 4.0 and future releases

Access to the Dataverse code is provided through Github (https://github.com/IQSS/dataverse). The long term sustainability of software generally relies on adoption and use by organizations that contribute to development and maintenance. While there can be no guarantees, the broad adoption of Dataverse and OJS, and Harvard's own adoption of Dataverse for its library services, is promising. The Dataverse software is licensed (https://github.com/IQSS/dataverse/blob/master/LICENSE.md) under the Apache License, Version 2.0 (the "License"); and may not be used except in compliance with the License.

Furthermore, long-term access to the data accepted by journals using this plugin is provided by the Harvard Dataverse. Dataverse relies on multi-institutional replication and a permanent endowment to guarantee bit-level preservation.

Supporting Information

We developed documentation and policy templates in support of the software tools developed through the project. This includes documentation for developers, such as API implementation and testing; for copy editors and reviewers that provide policies, guidelines, and detailed workflow help for areas such as citing data; for authors submitting data associated with an

article; linking articles to existing data; preparation of data and documentation data archiving; data confidentiality; dealing with licensing and other intellectual property restrictions; and other data use restrictions.

The materials for end-users are integrated into the built-in help and template systems for the OJS plugin and Dataverse. Documentation for developers is integrated through github. Furthermore, the project web site (http://projects.iq.harvard.edu/ojs-dvn) provides links to all documentation, templates, and training material.

The project website (http://projects.iq.harvard.edu/ojs-dvn) is the single location to find all information and resources related to the project, which includes presentations, publications, documentation related to the open source code for the plugin and the API, documentation for users setting up and using the plugin in OJS (updated (whenever relevant) with each new release of OJS), a video tutorial of the plugin, and a blog highlighting project milestones and events.

3. Reception

Outreach was integrated into the project at its early stages. We originally targeted approximately 50 journals that expressed interest in adopting the plugin when we completed our first phase of development in Winter 2014. These journals tested a beta version of the software and provided feedback on the plugin's core functionality and on our proposed data publishing workflows.

We directly reached out to over 566 individual journals within OJS through one-on-one in person conversations at conferences, phone/web conference calls and email correspondences. On top of the above mentioned number we were also approached by an aggregator of OJS journals, which supports hundreds of journals all over Latin America. Throughout the project we have been dedicated to providing one-on-one training and support for all journals and publishers who were interested or were already trying out the plugin. This included email, phone, in-person, and virtual video conference calls depending on the type of assistance needed.

In the winter of 2014, we conducted online surveys of representatives from journals that had previously expressed interest in the tool. We received responses representing over 200 journals (each respondent represented a cluster of journals, typically those hosted by a particular association or collaborative). Results from this survey indicated that 28% were currently evaluating the newly released tools, 29% were in active planning stages, and the remainder were in early planning stages.

Respondents indicated almost universally that the tool was not difficult for editors and administrators to configure or for authors and readers to use. A small minority encountered difficulty with installation of the plugin which subsequently has been resolved through its inclusion in the main OJS distribution. However, the surveys did indicate that there was room to improve the interfaces as only a minority found the system "very easy" to use (on a Likert scale).

As a result of this feedback, we implemented a number of feature improvements and bug fixes to improve the interfaces. There remains substantial opportunity for user interface refinement -- especially for the reviewer workflow -- but most of this goes hand in hand with the UX improvements due for OJS 3.0.

Moreover, a substantial majority (72%-95%) of journals responding agreed (or strongly agreed) that data sharing (75%), citation (95%), and replicability (72%) were important to the journal (The responding journals likely reflected those with the most interest -- based on a nonrespondent follow-up of 191 journals, a third to a half of all journals regard these issues as important). Journals almost universally agreed (99%) that supporting authors of articles with submitting their data was important to their journals, even in the absence of a data sharing policy; we now provide easy-to-implement policy templates within the plugin.

Future Development

Preliminary work has also been done on porting the OJS Dataverse plugin to Open Monograph Press, which uses the same plugin architecture as the upcoming OJS 3.x branch. It is expected that PKP will be able to use this preliminary spec work to provide a version of the OJS Dataverse plugin for OJS 3.x (and Open Monograph Press) within a year of its planned release this autumn. More generally, the codebase that supports integration with Dataverse through the standard SWORD interface for data deposition offers the potential to integrate OJS with other repository and data management systems (such as DASH, Harvard's institutional repository which is built on DSPACE, and Archivematica) with substantially reduced effort.

With respect to Dataverse development, the SWORD functionality developed through this project also significantly facilitates integration with additional systems. For example, preliminary work has been completed in integrating Dataverse with the Open Science Framework (https://osf.io/w75g2/), using this standard. With the support of an additional award from Sloan, Dataverse is building upon this codebase to test integrations with other publishers such as Sage Publishing, PLOS, Elsevier, Nature, and F1000Research.

Expected Future Impact

A number of direct impacts of this project should become visible over the next three to five years. Based on preliminary estimates of a broader survey of open access journals we recently conducted (and are preparing for forthcoming submission), there are hundreds of OJS and DOAJ listed OA journals with some form of data replication policy. Currently, there are over 4200 journals that are actively using OJS (defined as publishing more than 10 articles in a year for more than one consecutive year, as of the end of 2014) and have access to the plugin or will automatically have access following their next upgrade cycle. We expect to continue to monitor measures of adoption, datasets deposited through the tool, and citations to data and articles

published through journals using the plugin. In the medium term we expect these and other measures to show significant impact.

The broadest long-term outcome of the project will be to increase the replicability and reuse of research. We expect many of these broader impacts to develop over the course of the next ten years. The automated support for data publication, sharing, and citation provided by these tools makes it substantially easier for journal editors to adopt effective data citation, sharing, and replication policies. These policies are becoming increasingly salient as major stakeholders have issued policies and mandates supporting open data. Within the last year alone, a number of federal agencies (most recently DOD), independent funders, and professional societies have issued policies requiring data sharing, with greater movement in this direction clearly on the horizon.

Author Statement

Authors are listed in alphabetical order. We describe contributions to the paper using a standard taxonomy (Allen et. al 2014). Micah Altman and Eleni Castro were the lead authors, taking responsibility for content and revisions. Micah Altman and Merce Crosas were responsible for the initial conceptualization of the work. Eleni Castro was primarily responsible for data collection. Micah Altman and Eleni Castro were responsible for data analysis. Phil Durbin, Alex Garnett, and Jen Whitney were primarily responsible for the software development described. Micah Altman and Eleni Castro authored the first draft of the manuscript. All authors contributed to the writing through critical review and commentary.

References

- Altman, Micah, Leonid Andreev, Mark Diggory, Gary King, Elizabeth Kolster, A. Sone, Sidney Verba, Daniel Kiskis, and Michael Krot. "Overview of the virtual data center project and software." In Proceedings of the 1st ACM/IEEE-CS joint conference on Digital libraries, pp. 203-204. ACM, 2001.
- Altman, Micah. "Open source software for Libraries: from *Greenstone* to the *Virtual Data Center* and beyond." Iassist Quarterly 25 (2002).
- Castro, E. Connecting Journal Articles With Their Underlying Data: Encouraging Sustainable Scholarship Through the PKP-Dataverse Integration Project. In: Jankowska, M. (Ed.), *Focus on Educating for Sustainability: Toolkit for Academic Libraries* (pp-141-149). Sacramento, CA: Library Juice Press.
- Castro, E, & Garnett, A. Building a Bridge Between Journal Articles and Research Data: The PKP-Dataverse Integration Project. *International Journal of Digital Curation* [Internet]. 2014;9(1):176-184.
- Castro, E. (2015). A bridge from publishing words to publishing data [poster]. *FORCE2015: Research Communications & e-Scholarship Conference,* Oxford

University.
http://datascience.iq.harvard.edu/presentations/bridge-publishing-words-publishing-data-poster

- CODATA (2014) Task Group on Data Citation Standards and Practices, CODATA-ICSTI. "Out of Cite, Out of Mind: The Current State of Practice, Policy, and Technology for the Citation of Data." Data Science Journal 12.0 (2013): CIDCR1-CIDCR75.
- Stevenson, A. (2009). *SWORD2 project final report.* http://www.webarchive.org.uk/wayback/archive/20140615083222/http://www.jisc.ac.uk/media/documents/programmes/reppres/sword2finalreport.pdf
- Crosas, M. (2011). The dataverse network®: An open-source application for sharing, discovering and preserving data. *D-Lib Magazine, 17*(1/2) doi:10.1045/january2011-crosas
- Crosas, M. (2013). A data sharing story. *Journal of eScience Librarianship, 1*(3) doi:10.7191/jeslib.2012.1020
- King, G. (2007). An introduction to the dataverse network as an infrastructure for data sharing. *Sociological Methods and Research, 36*(2), 173-199.
- MacGregor, J., Stranack, K., & Willinsky, J. (2014). The Public Knowledge Project: Open source tools for open access to scholarly communication. *Opening science* (pp. 165-175) Springer International Publishing.
- Willinsky, J. (2005). Open Journal Systems: An example of open source software for journal management and publishing. *Library Hi-Tech, 23*(4), 504-519.