ResearchCompendia.org:
Cyberinfrastructure for Reproducibility and Collaboration in Computational Science

Victoria Stodden | University of Illinois at Urbana-Champaign
Sheila Miguez and Jennifer Seiler | Columbia University

The authors outline three goals to consider in building cyberinfrastructure to support scientific research and dissemination. They also present ResearchCompendia, a project designed to facilitate reproducibility in computational science by persistently linking data and code that generated published findings to the article, and executing the code in the cloud to validate or certify those findings.

In 2004, Gentleman and Temple Lang[1] introduced the concept of the *research compendium*: a new way of disseminating computational science results that delivers not only the article, but also the software tools and data required to reproduce the published findings. We describe a prototypical software infrastructure based on this idea, called ResearchCompendia, designed to implement two aspects of the *compendium*: persistently linking data and code that generated published findings to the article; and executing the code in the cloud to validate or certify those findings. Truly reproducible computational research not only improves the reliability of scientific findings, but permits the reuse of tools and data that generated the findings, facilitating downstream collaboration.

We outline three goals for cyberinfrastructure (CI) in support of scientific investigation and dissemination. We posit that CI should reinforce scientific norms—such as transparency and reproducibility[2,3]—however, CI should embed and encourage best practices in scientific research, and consider the entire discovery pipeline, even if focusing only on supporting a subset of the scientific workflow. In this article, we develop these ideas in the context of the ResearchCompendia project, and then include a discussion of the future vision of CI in support of science.

Reproducibility and ResearchCompendia.org

Reproducible computational science has attracted attention since Stanford Professor Jon Claerbout developed the idea of *really reproducible* manuscripts in 1991. Since then, a number of researchers have adopted reproducible methods or introduced them in their role as journal editors, and a body of scholarly literature is beginning to emerge. In May 2013, a workshop report on Knowledge Infrastructures was released,[4] bringing attention to recent rapid changes in how "people create, share, and dispute" knowledge due to changes in communication and research technologies. They note that "new forms of collective discovery and knowledge production … are springing up within and across many academic disciplines" and call for a reexamination of scientific knowledge production, dissemination, and assessment. There have been numerous reports over the last few years from a variety of fields lamenting the irreproducibility of published scientific results and articles appearing in the popular press. A workshop was held in 2011 on computational tools for reproducible research called "Reproducible Research: Tools and Strategies for Scientific Computing" (see the *CiSE* special issue on this topic at http://scitation.aip.org/content/aip/journal/cise/14/4/10.1109/MCSE.2012.82). In December 2012, a workshop called "Reproducibility in Computational and Experimental

 1521-9615/15/$31.00 © 2015 IEEE Copublished by the IEEE CS and the AIP

Mathematics" was held at the Institute for Computational and Experimental Research in Mathematics (ICERM) at Brown University.[5]

Several demonstrated use cases and a rationale for reproducible research were given by David Donoho and his colleagues,[6] reproducibility was advocated for the social science community by Gary King among others,[7] and the topic was the subject of a 2011 special issue in the magazine *Science,* for example.[8] Recent news articles in *Nature* and *Science* call for the release of academic code and data.[9] Journals are beginning to require code and data to be made available to readers of their published articles, primarily on externally hosted sites.[10]

These issues are also being considered at the policy level. In February and then May 2013, the Obama administration issued an Executive Memorandum and an Executive Order, respectively. The Executive Memorandum requires that federal funding agencies submit plans for public access to publications and data—defined as "the digital recorded factual material commonly accepted in the scientific community as necessary to validate research findings including data sets used to support scholarly publications"—by August 2013. The Executive Order directs federal agencies to make government agency data publicly available. These two actions by the White House have had the effect of bringing data access to the fore in federal funding agency conversations and addressing new challenges about how to implement access to digital scholarly objects, including how to ensure reproducibility and how to persistently link together the data and code with the published article.

Computational researchers often make preprints and articles available in repositories such as the arXiv or SSRN, but no comparable convenient facility exists to disseminate the code and data associated with published scientific papers that links together the article, the data, and the code in a structured and persistent way. In this article, we describe a mechanism to knit these ideas into reproducible science, called ResearchCompendia.

We also present a roadmap for the incorporation of CI across the research landscape. This roadmap focuses on lodestars for technical development, and gives less consideration to the equally important issues of incentives, funding, and implementation strategies. We consider three principles to guide the development of computational infrastructure for science:

- *Supporting scientific norms*—not only should CI enable new discoveries, but it should also permit others to reproduce the computational findings, reuse and combine digital outputs such as datasets and code, and facilitate validation and comparisons to previous findings.
- *Supporting best practices in science*—CI in support of science should embed and encourage best practices in scientific research and discovery.
- *Taking a holistic approach to CI*—the complete end-to-end research pipeline should be considered to ensure interoperability and the effective implementation of 1 and 2.

As we'll describe below, ResearchCompendia prototypes CI solutions for computational publications. Of course, there are myriad tools emerging to support reproducibility and the dissemination of scientific results, of which this is only one effort.[11–15]

The "Ubiquity of Error" and the ResearchCompendia Architecture

As one of us noted in 2009, "[t]he scientific method's central motivation is the *ubiquity of error*—the awareness that mistakes and self-delusion can creep in absolutely anywhere and that the scientist's effort is primarily expended in recognizing and rooting out error." [6] In the context of traditional empirical research, the response to the ubiquity of error employs standardized approaches such as statistical hypothesis testing and the reporting of information in the publication that enables reproducibility, principally through the materials and methods section.

Research that utilizes computational resources is subject to a new additional source of error, not captured by traditional publication standards. We believe that considering the computational aspects of an experiment as part of the experimental design itself will improve our ability to root error out of the scientific discovery process. For example, coding errors, the implementation or execution of algorithms or methods, or data filtering and cleaning decisions, could all be better checked with access to code and data.

ResearchCompendia is a website housing a collection of *compendia pages*. Each compendia page is associated with an externally available article, either published in a journal or made available in a preprint repository such as arXiv or SSRN. Figure 1 gives an example of such a webpage for a paper published in 2013.

A compendia page links to the webpage where the publication is available, or if the publication is open access, ResearchCompendia will host a copy and users can download it directly. In addition, author-provided code and data are available

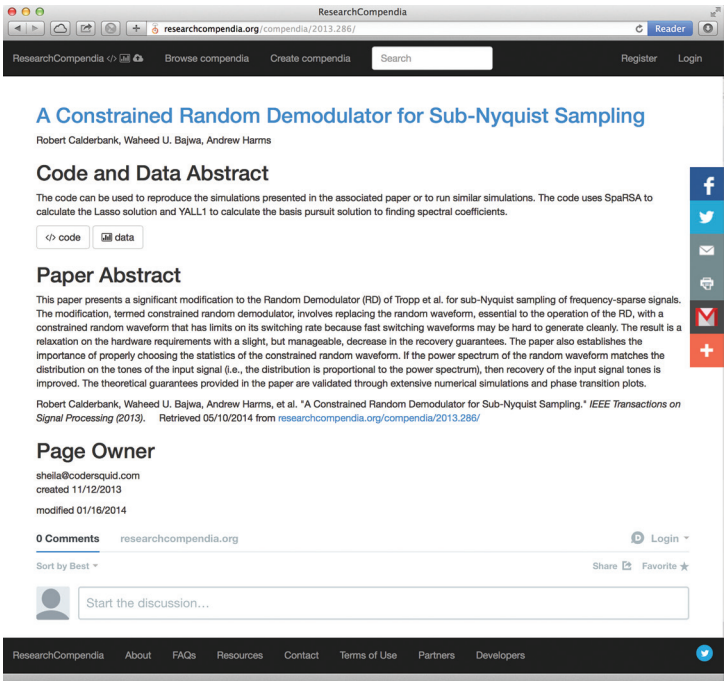

Figure 1. An example compendium page within the ResearchCompendia website. A compendia page links to the published paper, and gives the user access to the code and data associated with that publication. It also provides information and metadata about the code and data, and permits commenting.

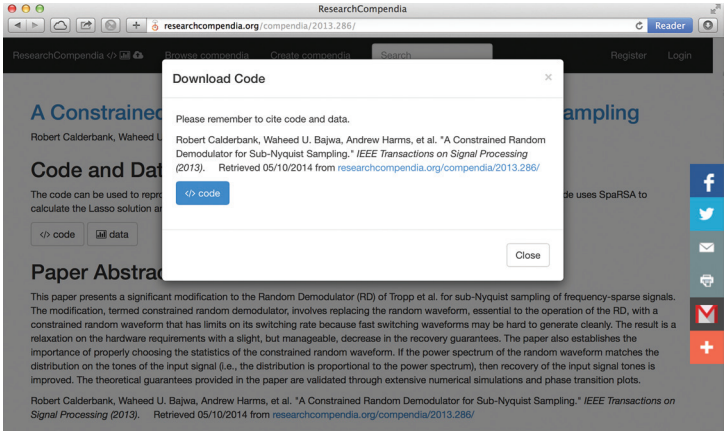

Figure 2. The suggested citation pop-up window that appears when a user clicks to download either code or data. The goal is to encourage good citation practices when reusing code and data.

for download by clicking the appropriate button, labeled "code" or "data." These two terms are left somewhat ambiguous deliberately, to permit the author leeway in deciding the appropriate software steps and data to include for the computational findings to be reproduced on a different system by an independent researcher. In some contexts, data are a starting point and analysis is carried out on these data. In other cases the data may be generated by the scripts themselves (and so data dissemination may not be necessary for replication purposes). If datasets or code are small enough for ResearchCompendia to host locally, it will store a copy. ResearchCompendia links to larger datasets or code hosted in external repositories. We also wish to improve the persistence of these digital scholarly objects by respecting the LOCKSS principle—Lots of Copies Keeps Stuff Safe (see www.lockss.org/about/what-is-lockss). For these reasons, we deposit data and code in external repositories whenever appropriate. To encourage proper citation, when a user clicks to download code or data, a suggested citation appears. See Figure 2 for an example.

Notice also that the compendium page provides descriptions of the code and data itself, not merely the abstract from the research article. Contributing programmers or other project members can be associated with the research objects as authors—there's no restriction to journal article authors alone. This permits flexibility in recognizing different types of contributions to research.

ResearchCompendia offers a larger infrastructure to support the creation and use of compendia pages. Prior to creating a compendia page, a user will create an account, and follow the steps to create a page. ResearchCompendia can fetch article DOI information to streamline the compendium page creation process, and DOIs are currently assigned by ResearchCompendia to all citable objects, such as code and data, in our labs pages at http://labs.researchcompendia.org/compendia/. Finally, it's also possible to leave comments on the compendium page, for example, notifying page owners of a bug in their code, or perhaps authors would like to notify users of an updated version of their code. Because of the motivating factor of reproducibility, ResearchCompendia will continue to provide the versions of the code and data that replicate results, even if errors are found within. Users will be alerted to these errors on the compendium page and the new versions will be provided in addition to the originals associated with the publication.

A high-level illustration of the information stored by ResearchCompendia appears in Figure 3. For each article, we keep the following:

- the user that created the compendium page;
- the contributors, such as article authors, programmers, or data curators;

- DOI information for citable objects such as the article, code, and data;
- the abstract describing the code and data;
- the abstract from the article;
- bibliographic information;
- licensing information for all digital scholarly objects;
- any related URLs, such as for the published paper;
- taxonomic information; and
- code, data, and other associated files such as the article.

Each compendia page is created by an account holder and has contributors, and some information about each contributor is stored in a separate table documenting their role in the project. Information is also gathered on account holders (if they choose to provide it), including their name, biography, affiliation, country, as well as information on associated URLs, permissions for access to various parts of ResearchCompendia, their email address, and their username.

We made a philosophical and practical decision to develop the ResearchCompendia code base as an open source and collaborative project. The code is hosted by GitHub and available at https://github.com/researchcompendia/researchcompendia/. We provide the source code in hope for collaboration, but also to permit others to stand up their own ResearchCompendia websites. We permissively license the code under the MIT license and data under a Creative Commons license (CC0), following the licensing guidelines from previous work.[16,17] Research code that's uploaded to ResearchCompendia has the MIT License as a default and with a different license upon request. Part of the rationale for default licensing is to simplify the upload process, and part is to maximize the ability of researchers to combine various codes into a new project.

Reliability, Executability, and Verification

As cyberinfrastructure becomes an increasingly important part of the scientific research pipeline, it can serve to encourage best practices in the scientific discovery process. As we described previously, ResearchCompendia seeks to encourage good practices such as the citation of code and data when reused. A second goal for ResearchCompendia is to certify or validate published findings, bringing our efforts more in line with the vision of Gentleman and Temple Lang described at the outset of this article. In our lab pages at http://labs.researchcompendia.org/compendia/, ResearchCompendia extends compendium page functionality to include execution of the research codes to verify the results in the publication, using the data provided by the author. On these "executable compendium pages," users have the additional option of running the code in the cloud directly through the compendium page. The creation of these pages requires ResearchCompendia to check the code submitted by the researchers (that is, compatibility, CPU requirements, computing time, verifying constraints on input parameters, and so on) and ensure that it replicates the figures from the original article. (See http://labs.researchcompendia.org/compendia for pilot executable compendium pages that permit the user to verify the computational findings.) Running the author's code can be very quick or take significant time. After successfully executing the code, we generally offer access to cached results, but users are free to run the code on their own independent platforms as well. In this sense, ResearchCompendia acts as a *certifier* of research results. See Figure 4 for an example executable compendium page. In addition to the code and data download buttons, notice the "verify" button and documentation of verification runs.

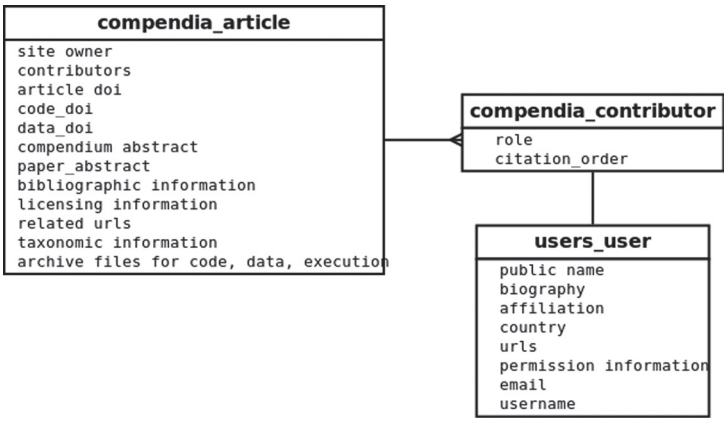

Figure 3. A graphical depiction of the data associated with a ResearchCompendia compendia page. Notice that DOIs are assigned to the code and to the data as primary citable objects.

For the creation of executable compendium pages, we've developed technology using lightweight virtual machines, commonly called containers, to create a local environment that executes the code. This is done for security reasons (to keep these code executions and any user input separate from the rest of ResearchCompendia) and to ensure that the necessary software and libraries are installed so that the codes will execute.

Figure 5 augments Figure 3 by showing the additional information collected for executable compendium pages, including the time of the verification run, whether or not it was successful, any standard output or error, parameter information, and archives output files and information.

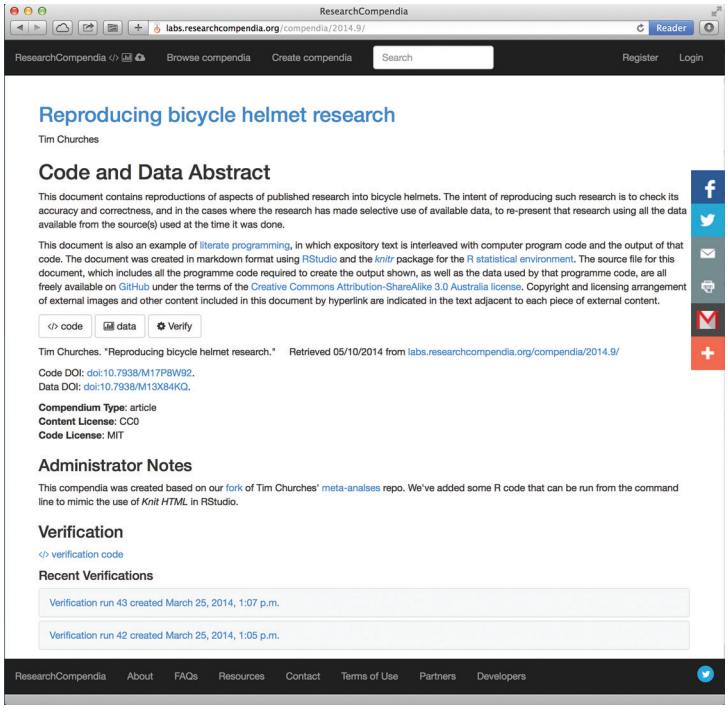

Figure 4. A prototypical executable compendia webpage in labs. ResearchCompendia.org. This webpage provides not only the code and data for download and reuse on independent computer systems, but users are also able to click the "verify" button on the compendia page and access results from our running of the code. In this way, ResearchCompendia can certify papers as reproducible.

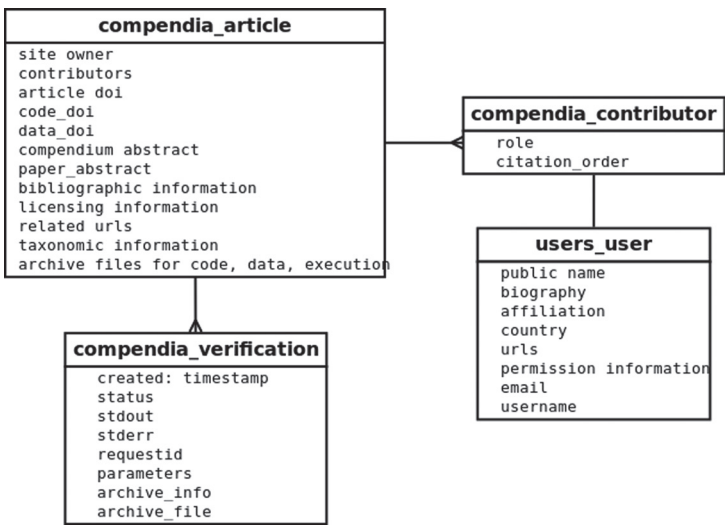

Figure 5. A graphical depiction of the data associated with an executable ResearchCompendia compendia webpage, augmented by the additional information associated with executing the code and verifying the research results contained in the paper.

Prioritizing the Complete Research Pipeline

Factors that are accelerating the implementation of reproducible research include the open source software movement (permitting sharing, reusing, and using good practices) and the adoption of cloud computing in scientific research (permitting the launch of complex jobs on thousands of cores with a single click, and providing common environments for code execution). The computations used in research today can be very complex, involving sometimes immense amounts of code to merge and clean datasets, and implement ambitious algorithms, but complexity also arises in research workflows combining various codes implementing these data processing or algorithmic steps. To illustrate ongoing trends, Figure 6 gives the staggering increase in lines of code submitted to the CALGO repository for the *ACM Transactions on Mathematical Software* (*TOMS*) from 1960 through 2013. The total number of lines of code submitted increases steadily on a log scale, from 875 lines in 1960 to nearly 5 million in 2012 (the proportion of total publications with associated code remained roughly constant at approximately 1/3, with standard error of about .12, and the journal consistently publishing around 35 articles each year).

The evolution of computing infrastructure will include the documentation of research pipelines with workflow tools, as researchers chain together complex software from difference sources, scripts, and codes in different languages. Researchers will "build on the shoulders of giants" through repurposing code and data from other authors. Even if good software practices are followed and each algorithm is well documented, computing infrastructure and communication standards will need to incorporate an additional notion of a research workflow that documents published computational findings so that they can be reproduced.

Discussion and Next Steps for ResearchCompendia

Future development of ResearchCompendia falls into two categories—short term and long term. In the short term, we'd like to extend the executability demonstrated in the labs to the entire website. We also plan to evolve the compendium webpage to *interact* with the user—to give users an opportunity to run the code with different inputs, such as alternative parameter settings or updated datasets (or other datasets uploaded to ResearchCompendia, perhaps those associated with different compendium pages).

We believe this will help research both for *validation* purposes and as a tool of *stability*. The

concept of validation is well known in the scientific computing and simulation-oriented disciplines. It refers to "[t]he process of determining the accuracy with which a model can predict observed physical events (or the important features of a physical reality)."[18] In other words, "[t]he process of determining the degree to which a model (and its associated data) is an accurate representation of the real world from the perspective of the intended uses of the model."[19] Validation is often carried out via comparison with independently generated results, or other sources of real-world data not used in the current model. ResearchCompendia offers an alternative avenue for validation. As it collects data from studies, these datasets themselves may be used to validate previous results. For example, we could imagine findings based on very small sample sizes (perhaps considered large at the time) being validated over much larger databases as they're submitted to ResearchCompendia.

ResearchCompendia can also facilitate the understanding of *stability* in scientific findings—the notion that the variability of estimators is bounded when the underlying data is perturbed in well-understood ways.[20] Code that runs in a system that's frequently augmented with new data sources could give an opportunity to test the stability of models using different datasets not included in the original study. ResearchCompendia develops the ability to validate findings on much larger datasets as it collects models and data from different articles investigating the same same or related scientific hypotheses. For example, code implementing a model on a small dataset could be executed on much larger datasets contributed to ResearchCompendia from other related studies.[21]

As we continue to execute code/data combinations, we'll gather the information about what makes code easy or difficult for us to run, and use that information to create a set of guidelines for code submission to ResearchCompendia. ResearchCompendia is a testbed for the implementation of the three ideas regarding scientific CI presented here, and we can monitor usage to discover successes and failures in this approach as researchers increasing share myriad types of data and code.

In the longer term, we expect scientific publishing to move away from the publication as a standalone object in .pdf format. The published findings are of course fixed at the time of publication, but so should be the versions of the code and data that generated those findings. Some barriers to overcome include: versioning of code and data to maintain reproducibility of published results; persistently connecting article, data, and code (including workflow information) in the Gentleman and Temple Lang spirit of a research compendium; and maximizing interoperability of code and data for reuse while minimizing the burden on the computational researcher, and safeguarding privacy and confidentiality concerns in the data.

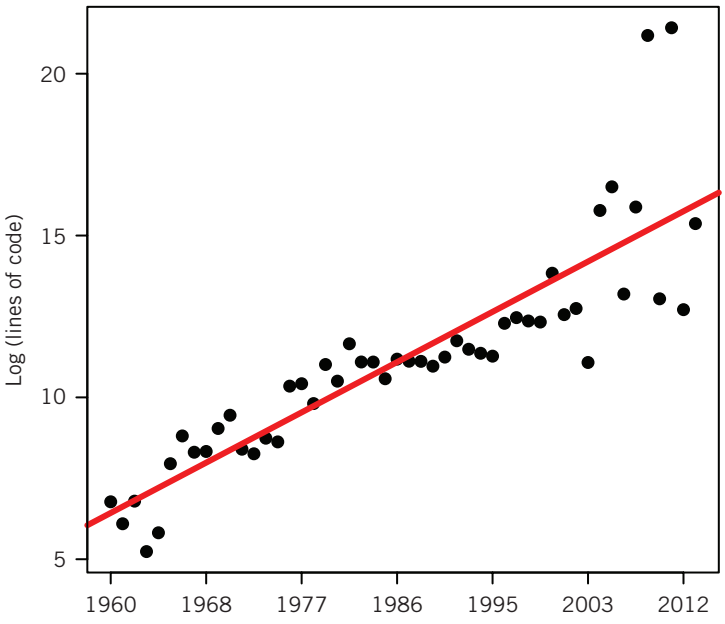

Figure 6. The increase in the number of lines of code submitted to the CALGO repository associated with the *ACM Transactions on Mathematical Software (TOMS)* journal, 1960–2013. The best-fit line shows the dramatic increase in code complexity and size, represented on a log scale. In recent years, the variability in the number of lines of code has increased. The journal published about 35 articles per year consistently throughout this time period, and about a third of the articles submitted code to the CALGO repository. These data also appear in Figure 1 of Victoria Stodden's article, "Reproducing Statistical Results," in the journal *Annual Review of Statistics and Its Application,* forthcoming in January 2015.

Interoperability includes reuse of code on different projects and on different systems. At this point, ResearchCompendia's initial execution of the code is carried out manually when creating the executable compendium page, but using the container approach described above will permit the service to scale. We anticipate that ResearchCompendia will be able to directly accept containers from researchers that house the fully functional software and data, thereby automating availability as an executable compendium page. Eventually, ResearchCompendia will supply the script or image for these containers to authors, who will then ensure that their results reproduce computationally in that sharable environment.

We plan to continue open source development of ResearchCompendia. Not only is this philosophically consistent with our larger goals of achieving greater transparency in scientific research and communication, but it permits a community to grow around these efforts and contribute back to the project. It also allows others to duplicate and extend the infrastructure in other contexts. We hope that such downstream use contributes to discoverability of code and data. As mentioned previously, ResearchCompendia hosts the data, code, and articles it makes available (barring size and/or legal barriers to doing so), and points to external copies when it doesn't. ResearchCompendia isn't meant to replace other forms of dissemination, such as supplemental material sections in journals or dedicated data repositories, for example, but to *augment* their efforts by providing a centralized, discoverable, and persistent way of linking the digital objects that comprise computational scholarship. To our knowledge there's no other service that focuses uniquely on reproducibility, and hence, the article, data, and code—the Research Compendium—as the appropriate unit of scholarly communication for computational science.

Acknowledgments

This research was supported by Alfred P. Sloan Foundation award number PG004545 "Facilitating Transparency in Scientific Publishing." The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

References

1. R. Gentleman and D. Temple Lang, "Statistical Analyses and Reproducible Research;" *Bioconductor Project Working Papers*, 2004; http://biostats.bepress.com/bioconductor/paper2.
2. R.K. Merton, "The Normative Structure of Science," *The Sociology of Science: Theoretical and Empirical Investigations*, Univ. of Chicago, 1973.
3. V. Stodden, "The Scientific Method in Practice: Reproducibility in the Computational Sciences," MIT Sloan Research Paper No. 4773-10, 2010; http://ssrn.com/abstract=1550193 or http://dx.doi.org/10.2139/ssrn.1550193.
4. P.N. Edwards et al., *Knowledge Infrastructures: Intellectual Frameworks and Research Challenges*, Deep Blue, 2013; http://hdl.handle.net/2027.42/97552.
5. D.H. Bailey, J. Borwein, and V. Stodden, "Set the Default to 'Open'," *Notices of the AMS*, June/July 2013, pp. 679–680.
6. D.L. Donoho et al., "Reproducible Research in Computational Harmonic Analysis," *Computing in Science & Eng.*, vol. 11, no. 1, 2009, pp. 8–18; doi:10.1109/MCSE.2009.15.
7. G. King, "Replication, Replication," *PS: Political Science and Politics*, vol. 28, 1995, pp. 443–499; http://j.mp/jCyfF1.
8. "Data Replication and Reproducibility," *Science* (special issue), vol. 334, no. 6060, 2011.
9. A. Morin et al., "Shining Light into Black Boxes," *Science*, vol. 336, no. 6078, 2012; www.sciencemag.org/content/336/6078/159.
10. V. Stodden, P. Guo, and H. Ma, "Toward Reproducible Computational Research: An Empirical Analysis of Data and Code Policy Adoption by Journals," *PLoS ONE*, vol. 8, no. 6, 2013, article no. e67111.
11. C. Stodden, C. Hurlin, and C. Perignon, "RunMyCode.org: A Novel Dissemination and Collaboration Platform for Executing Published Computational Results," *Proc. IEEE 8th Int'l Conf. E-Science*, 2012; http://dx.doi.org/10.2139/ssrn.2147710.
12. I.M. Mitchell, R.J. LeVeque, and V. Stodden, "Reproducible Research for Scientific Computing: Tools and Strategies for Changing the Culture," *Computing in Science & Eng.*, vol. 14, no. 4, 2012, pp. 13–17; doi:10.1109/MCSE.2012.38.
13. M. McLennan and R. Kennell, "HUBzero: A Platform for Dissemination and Collaboration in Computational Science and Engineering," *Computing in Science & Eng.*, vol. 12, no. 2, 2010, pp. 48–52.
14. D. De Roure, C. Goble, and R. Stevens, "The Design and Realisation of the myExperiment Virtual Research Environment for Social Sharing of Workflows," *Future Generation Computer Systems*, vol. 25, 2009, pp. 561–567; http://eprints.soton.ac.uk/id/eprint/265709.
15. V. Stodden, F. Leisch, and R.D. Peng, eds., *Implementing Reproducible Research*, CRC Press, 2014.
16. V. Stodden, "The Legal Framework for Reproducible Scientific Research: Licensing and Copyright," *Computing in Science & Eng.*, vol. 11, no. 1, 2009, pp. 35–40.
17. V. Stodden "Enabling Reproducible Research: Licensing for Scientific Innovation," *Int'l J. for Comm. Law and Policy*, vol. 13, no. 08–09, 2009; http://ijclp.net/old_website/article.php?doc=1&issue=13_2009.
18. J.T. Oden, "A Brief View of Verification, Validation, and Uncertainty Quantification," invited lecture presented at the *Foundations of Verification, Validation, and Uncertainty Quantification Conf.*, 10 June 2010; http://users.ices.utexas.edu/~serge/WebMMM/Talks/Oden-VVUQ-032610.pdf.
19. P. Roache, "Perspective: Validation—What Does It Mean?" *J. Fluids Eng.*, vol. 131, no. 3, 2009, article no. 034503.
20. B. Yu, "Stability," *Bernouilli*, vol. 19, no. 4, 2013, pp. 1484–1500.

21. M.D Stang et al., "A Systematic Statistical Approach to Evaluating Evidence from Observational Studies," *Ann. Rev. of Statistics and Its Applications*, vol. 1, no. 1, 2014, pp. 11–39.

Victoria Stodden is an associate professor in the Graduate School of Library and Information Science at the University of Illinois at Urbana-Champaign. Her current research interests include statistical methods for large data applications and reproducible computational research. Stodden has a PhD in statistics from Stanford University and an MLS from Stanford Law School. Contact her at victoria@stodden.net.

Sheila Miguez is a research scientist in the Department of Statistics at Columbia University. She enjoys helping people at Python workshops and Python office hours at her local hackerspace. Miguez has a BS in computer science and psychology from the University of Maryland College Park. Contact her at shekay@gmail.com.

Jennifer Seiler is a postdoctoral researcher in the Department of Statistics at Columbia University. Her current research interests include open science and astrophysics. Seiler has a PhD in physics from the Max Planck Institute for Gravitational Physics. Contact her at jenn.seiler@gmail.com.

cn *Selected articles and columns from IEEE Computer Society publications are also available for free at http://ComputingNow.computer.org.*