

Mozilla plan seeks to debug scientific code

Software experiment raises prospect of extra peer review.

Erika Check Hayden

24 September 2013

When ecologist Carl Boettiger wrote a blog post in June calling for greater stringency in the peer review of scientific software in research papers, he hardly expected to stir up controversy. But in 54 comments on the post, researchers have debated how detailed such reviews should be; one said that it was a “trifle arrogant” of Boettiger, of the University of California at Santa Cruz, to insist that computer code attain his stringent standards before publication.

Print

Now an offshoot of the Internet non-profit organization Mozilla has entered the debate, aiming to discover whether a review process could improve the quality of researcher-built software that is used in myriad fields today, ranging from ecology and biology to social science. In an experiment being run by the Mozilla Science Lab, software engineers have reviewed selected pieces of code from published papers in computational biology. “Scientific code does not have that comprehensive, off-the-shelf nature that we want to be associated with the way science is published and presented, and this is our attempt to poke at that issue,” says Mozilla Science Lab director Kaitlin Thaney.

Related stories

- Cancer institute tackles sloppy data
- Publish your computer code: it is good enough
- Computational science: ...Error

More related stories

Researchers increasingly rely on computation to perform tasks at every level of science, but most do not receive formal training in coding best practice. That has led to high-profile problems. Some scientists have argued, for example, that the fraudulent findings used as the basis for clinical trials in 2007 would have been exposed much earlier if cancer researcher Anil Potti of Duke University in Durham, North Carolina, had been compelled to publish his data and computer code along with his original papers.

More routinely, incorrect or slipshod code prevents other researchers from replicating work, and can even lead them astray. In 2006, Geoffrey Chang of the Scripps Research Institute in La Jolla, California, had to retract five research papers on crystal structure after finding a simple error in the code he was using, which had been provided by another lab. “That’s the kind of thing that should freak any scientist out,” says computational biologist Titus Brown at Michigan State University in East Lansing. “We don’t have good processes in place to detect that kind of thing in software.”

Mozilla is testing one potential process, deploying the type of code review that is routinely used on commercial software before it is released. Thaney says that the procedure is much like scientific peer review: “The reader looks for everything, from the equivalent of grammar and spelling to the correctness of the logic.” In this case, Mozilla opted to examine nine papers from *PLoS Computational Biology* that were selected by the journal’s editors in August. The reviewers looked at snippets of code up to 200 lines long that were included in the papers and written in widely used programming languages, such as R, Python and Perl.

“One worry is that scientists will be even more discouraged from publishing their code.”

The Mozilla engineers have discussed their findings with the papers’ authors, who can now choose what, if anything, to do with the markups — including whether to permit disclosure of the results. Those findings will not affect the status of their publications, says Marian Petre, a computer scientist at the Open University in Milton Keynes, UK, who will debrief the reviewers and authors. They expect to release a preliminary report on the project within the next few weeks.

Computational biologists are betting that the engineers will have found much to criticize in the scientific programming, but will also have learnt from the project. They may have been forced to brush up on their biology, lest they misunderstood the scientific objective of the code they were examining, Brown says.

Theo Bloom, editorial director for biology at non-profit publisher PLoS, shares that expectation, but says such reviews may still be useful, even if the Mozilla reviewers lack biological expertise. Yet that would prompt another question: how can journals conduct this type of review in a sustainable way?

The time and skill involved may justify paying reviewers, just as statistical reviewers of large clinical trials are paid. But researchers say that having software reviewers looking over their shoulder might backfire. “One worry I have is that, with reviews like this, scientists will be even more discouraged from publishing their code,” says biostatistician Roger Peng at the Johns Hopkins Bloomberg School of Public Health in Baltimore, Maryland. “We need to get more code out there, not improve how it looks.”

Nature **501**, 472 (26 September 2013) doi:10.1038/501472a

Related stories and links

From nature.com

- **Cancer institute tackles sloppy data**
12 October 2012
- **Publish your computer code: it is good enough**
13 October 2010
- **Computational science: ...Error**
13 October 2010

From elsewhere

- **Mozilla ScienceLab**
 - **Carl Boettiger’s blog**
-

For the best commenting experience, please login or register as a user and agree to our Community Guidelines. You will be re-directed back to this page where you will see comments updating in real-time and have the ability to recommend comments to other users.

1 comment

Subscribe to comments



Egon Willighagen • 2013-09-24 04:11 PM

It should be noted that the Open Research Computing journal never took off, with too few scientists willing to submit their paper for proper peer review of the source code, though it now features as a thematic series in another journal: <http://www.openresearchcomputation.com>

See other News & Comment articles from *Nature*

Nature ISSN 0028-0836 EISSN 1476-4687

© 2013 Nature Publishing Group, a division of Macmillan Publishers Limited. All Rights Reserved.

partner of AGORA, HINARI, OARE, INASP, CrossRef and COUNTER