TOOLBOX

DEMOCRATIC DATABASES: SCIENCE ON GITHUB

Scientists are turning to a software-development site to share data and code.

ILLUSTRATION BY THE PROJECT TWINS

BY JEFFREY PERKEL

When the Ebola outbreak in West Africa picked up pace in July 2014, Caitlin Rivers started to collect data on the people affected. Rivers, then a PhD student in computational epidemiology, wanted to model the outbreak's spread. So every day she downloaded PDF updates released by the ministries of health of the virus-stricken countries, and converted the numbers into computer-readable tables. Rather than keeping these files to herself, she posted them to GitHub.com, a hugely popular website for collaborative work on software code. Rivers thought the postings might attract those interested in up-to-date information from the Ebola outbreak. "I figured if I needed it, other people would, too," she says.

Rivers was right. Other researchers began to download the data and contribute to the project. On some days, third parties would download and convert the ministries' data before her, and load them into the GitHub repository. Others created programming scripts to do simple error-checks on the data, such as ensuring that the daily patient counts made sense. At the time, GitHub was "really the only place on the Internet that you could interact with these data as data, and not as a PDF", says Rivers, who was at Virginia Polytechnic Institute and State University in Blacksburg when she began the project, and is now an epidemiologist at the US Army Public Health Center in Edgewood, Maryland.

Launched in 2008 to assist software developers, GitHub now boasts some 15 million users and is an increasingly popular site for researchers to share, maintain and update scientific data sets and code (see 'Growing influence of GitHub'). GitHub is "the biggest revelation in my workflow … since I started writing code", says Daniel Falster, a postdoctoral researcher in ecology at Macquarie University in Sydney, Australia. "When we started using GitHub, it was just amazing. We now use it in everything that we do." Falster's Biomass and Allometry Database, which aggregates various measures of plant size from 176 studies, is stored on the site. So is the Open Tree of Life project, which aims to compile different published phylogenies to build one master 'tree of life'. It uses GitHub to store data files and publication records, and to accept new data sets from third parties.

Plenty of websites are dedicated to sharing data. But GitHub is specifically designed for transparent, open collaboration because it ▶

uses version-control software to track every change made to code or data. This means that large, distributed teams of programmers can work together on a project online, and users can scroll back in time through a file's version history, seeing each change, when it was made, by whom and for what purpose. Programmers can copy ('fork') a repository to experiment with new ideas; useful changes can be folded into the main project, while others can be ignored or rolled back later.

For instance, anyone can visit the GitHub-based Open Exoplanet Catalogue — a growing database of the thousands of known planets outside the Solar System — and submit new information through their browser. As with the Open Tree of Life, the project's main website doesn't have github.com in the URL address, so casual visitors wouldn't necessarily know that they are interacting with version-control software — but the files are openly available in a GitHub repository for more sophisticated users. Making an edit alerts the project's developers, including Hanno Rein, an astrophysicist at the University of Toronto in Canada, to review the suggested change. GitHub, says Rein, allows for a "way more democratic system" than would a static online catalogue of exoplanets, because any user can suggest changes and can even customize a version of the data set to their own specifications. Some 100 people have forked the project's repository, and Rein's smartphone app Exoplanet, which runs off the same database, has attracted around 10 million downloads.

FROM LINUX TO THE LAB

The software tool that GitHub relies on is called Git. It was created in 2005 by coder Linus Torvalds to manage development of the open-source operating system Linux — a huge project that involved thousands of independent programmers. "Git is a technology that's designed for very fine-grained, line-by-line monitoring of changes in source code," says Arfon Smith, a program manager for GitHub in Seattle, Washington. It is not the only version-control software available (another option is Mercurial), but it is one of the most popular.

Many programmers use Git on their own computers. For scientist coders, the tool works like a laboratory notebook for scientific computing, says Katy Huff, a nuclear engineer at the University of Illinois at Urbana–Champaign: just like a lab notebook, it keeps a lasting record of events. But its syntax and workflow are notoriously confusing. "I'm comfortable saying that the interface is unnecessarily non-intuitive," Huff says.

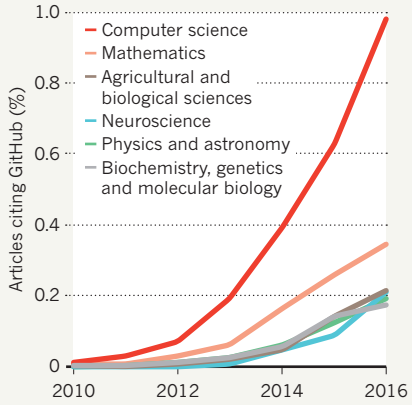
GitHub's prettier browser interface softens some of Git's hard edges, making it easier for novices to contribute. The site now hosts millions of projects, some personal, some massively collaborative, and is free for open-source projects. (Users and organizations that want to keep their files private pay US$7 per month and up. A related service called Bitbucket, which also runs on Git, offers unlimited free public and private repositories for up to five users; larger collaborations cost from $10 per month.)

Not every kind of data set works well with Git software. The tool records line by line how files have changed. It works well with text files such as source code, XML files, manuscripts written in Markdown or LaTeX, and CSV files (which can be exported from Excel, for instance). But it cannot effectively keep track of changes in non-human-readable 'binary' files, such as Microsoft Office documents and images, because the program's 'diff'ing' function, which identifies how files change from version to version, cannot interpret such data. "As soon as you introduce a binary format that isn't line-oriented, Git does a terrible, terrible job of versioning that content," Smith says.

GitHub also imposes file limitations; it has a hard limit of 100 megabytes per file, and a 'soft' cap of a gigabyte per repository. (A plug-in called Large File Storage allows Git and GitHub to more effectively handle larger files, although it still cannot report the differences between binary versions.)

GROWING INFLUENCE OF GITHUB

An increasing proportion of research articles cite GitHub in their references.



RICHARD VAN NOORDEN/SOURCE: ELSEVIER SCOPUS DATABASE

FAST AND FLEXIBLE

GitHub makes most sense for those researchers working with relatively small, text-based data sets that are being actively updated, curated and maintained by groups of scientists — such as Rivers' Ebola-virus project. Nick Loman, a microbial genomicist and bioinformatician at the University of Birmingham, UK, has also used the site to drive fast-paced studies of pathogens. Loman is a member of the ZiBRA (Zika in Brazil Real Time Analysis) project, an ongoing Brazilian surveillance effort that collects Zika-virus samples across the country and sequences and analyses them in real time. Traditionally, Loman says, DNA sequence data go to archives such as GenBank — and these data will too. But it can take time for those sites to release data to the public. GitHub, he says, provided a faster and more flexible way to disseminate draft data sets, rather like tweeting a research finding in advance of publication.

Because data sets on GitHub can be changed or deleted by their authors, the site doesn't guarantee a permanently citable archive, warns Smith. Those interested in creating a long-term, permanent record of their data set as it exists at a particular point in time — for example, when a paper is published — should consider storing the relevant version of their data on dedicated scientific sites, such as Zenodo and Figshare. Both of these sites allow GitHub users to archive snapshots of their repositories, and will provide a citable Digital Object Identifier (DOI) for the data set. According to Smith, some 8,000 GitHub users have done so.

Another data-sharing option is Dat, a general-purpose tool for sharing and syncing data between different computers. According to lead programmer Max Ogden in Portland, Oregon, Dat provides versioning in a similar way to Git for collaborative work, but includes a peer-to-peer file-sharing system for distributing data files. Ogden says that Dat is more adept at handling large binary files because it breaks them into chunks and transfers only those pieces that have changed.

Data sharing is a key requirement of open science, and researchers can share data sets anywhere they wish. But even if they don't use GitHub.com, scientists should consider using Git or a comparable tool to record changes to data sets and data-processing scripts, says Tracy Teal, executive director of Data Carpentry, a non-profit organization that trains researchers in working with data. Researchers interested in learning to use Git and GitHub have many online resources to turn to: Codecademy offers a free interactive tutorial, as does GitHub (try.github.io). Greg Wilson, founder of the research-computing skills site Software Carpentry, co-authored a how-to guide in January (J. D. Blischak *et al. PLoS Comput. Biol.* **12,** e1004668; 2016). And many programmers and bioinformaticians use Git — so they, too, can always be asked for help.

Despite their steep learning curves, Git and GitHub have a loyal fan base among scientists. Emily Jane McTavish, an evolutionary biologist at the University of California, Merced, and a member of the Open Tree of Life project, says it's an essential resource. "I don't know how I lived without it." ■

CORRECTION

The article 'Computers on the reef' (*Nature* **537,** 123–124; 2016) omitted to give the name of the system developed by Arjun Chennu and wrongly said that it is based on a neural-network algorithm. It is called HyperDiver, and it uses a machine-learning algorithm similar to the one used by CoralNet.

CORRECTION

The Toolbox article 'Democratic databases: science on GitHub' (*Nature* **538,** 127–128; 2016) misstated how the Git software records changes in files. It does in fact maintain multiple versions of the files.