

Method of the Year 2009 voting begins

We are now accepting nominations and collecting votes for the Method of the Year 2009.

It is difficult to dispute the idea that some methodological developments have the power to strongly influence the pace and direction of scientific inquiry. For each of the last two years, *Nature Methods* has celebrated this by selecting a methodology we felt had demonstrated the potential for such an impact.

In 2007, we picked next-generation sequencing, and this class of methods has already had a profound effect on the conduct of biological research. Only time will tell what the impact of super-resolution fluorescence microscopy—our choice in 2008—will be.

Last year, we invited you—our readers—to participate by nominating and voting for methodological developments. We are now halfway through 2009, and it is time once again to ask for your input. Just go to <http://www.nature.com/nmeth/votemoy2009>, log in

using your free nature.com account or take a minute to request an account, and you can vote on a selection of methodologies or nominate a method yourself.

You may nominate any recent method or class of methods, published anywhere in the scientific literature, that you believe is likely to have a profound impact on future biological research. You are welcome to nominate a method that you yourself developed, but please acknowledge your connection to it.

We will take the results of the popular vote into consideration when choosing the Method of the Year 2009, and the votes will also serve as inspiration for picking the Methods to Watch that accompany it. So start thinking about methods to nominate, and then go to <http://www.nature.com/nmeth/votemoy2009> and vote for your choice of Method of the Year 2009!

Software by any name

Computational biologists are often tempted to avoid providing a named software implementation of their new algorithm, but resisting this temptation helps avoid difficulties later on and benefits the wider community of biologists.

Say you've developed a novel research method or tool. You will likely have a strong desire to give it a clever name while cherishing a hope that the method and name will become widely used—think “polymerase chain reaction.” But if the method is a new algorithm implemented in software, you may decide to forego naming the software to avoid shifting readers' focus away from the algorithm itself.

Some readers may be asking themselves how a new algorithm is different from a new piece of software. In fact, although the two can be intimately related, there is a distinct difference. A new algorithm is a defined set of instructions for solving a problem in a new way—literally a new method. Software, on the other hand, although it is often required to implement an algorithm in a way that biologists can use, may instead be only a repackaging of existing algorithms to create a new tool.

New algorithms are communicated to the wider community using natural language supplemented with a series of mathematical operations, pseudocode

or computer language. This should be done in enough detail that users can incorporate the algorithm into software of their own design. Although a software implementation of the algorithm is often used to test the algorithm's performance before publication, this software isn't necessarily provided to readers. And if the software is provided, it is often not named.

There are a number of reasons why the author of a new algorithm may be reluctant to name the software implementation. For one thing, people often expect named software to be maintained and updated. This expectation may not seem unreasonable, but the reality is that after the graduate student or postdoc who wrote the software has left the lab, he or she will often be unable to continue supporting it by tracking down bugs or adding new requested functionality. And the principle investigator, regardless of desire, is unlikely to be familiar enough with the code to do so. Changes in operating systems and hardware make maintaining software even more complicated. An algorithm, on the

other hand, requires no support provided that all the necessary details were presented in the paper.

Release of software under an open-source license is often presented as a solution to this problem because this allows users to fix bugs or add functionality—and release under such a license is strongly encouraged by the community. But with the occasional exception, open-source software is almost never examined in detail by users and is modified even more rarely. The problem is particularly acute when most of the users are biologists, not bioinformaticians.

Although a lack of continuing support for software is regrettable, such support requires far more work than is necessary for typical lab methods or tools. Bench methods can almost always be reproduced from a detailed step-by-step protocol (easily distributed by e-mail), and users frequently introduce their own tweaks anyway. Clones of new genetically encoded tools just need to be expanded and mailed out. If a serious problem with a non-software-based tool does manifest itself, biologists will just decide not to use the tool rather than demand that the author fix it.

But there is another reason why software implementations often remain unnamed, one that biologists may not fully appreciate. A large number of software tools containing existing algorithms are published every year, often as two-page reports in bioinformatics journals, and the computational biology community views these software tools with less interest than novel algorithms. Some computational biologists may not even read past the tool name to determine whether a useful new algorithm is contained in the tool.

Anecdotal reports suggest that computational biologists often feel that the easiest way to prevent an algorithm from being seen by their peers as just

another tool—rather than an innovative new method—is to avoid providing a software implementation. Depending on the complexity of the algorithm and the application, the lack of software may or may not impede uptake by biologists. Alternatively, if software is provided, it may not be named.

Unfortunately, unless the new algorithm itself is named, supplying an unnamed software implementation of the algorithm can cause difficulties after the paper is published. Whereas the computational biology community is interested in novel algorithms, the larger biological research community wants user-friendly software that users can easily identify. A good name provides an easy and unique way of referring to the software and simplifies searching for papers describing work in which it was used. Although it is possible to use the author's name instead, this becomes problematic when other people need to repeatedly refer to different unnamed pieces of software, particularly if an author is associated with multiple pieces of software. Using one author's name may also be unfair if several people contributed to the software's development.

We believe the problems resulting from providing a named software implementation of a new algorithm, while real, do not outweigh the benefits to the larger biological community. If the main point of a paper is to describe a new algorithm and its use, authors should be careful to write the paper so to make it clear that the software tool is only an implementation of a novel algorithm. By doing so, authors can hope to avoid detracting from the interest of the work for other computational biologists while still providing an implementation that will be useful for the biologists who want to evaluate and use it.