25
The Open-Source Paradigm and the Production of Scientific Information: A Future Vision and Implications for Developing Countries

Charles M. Schweik
University of Massachusetts, United States
J. Morgan Grove
U.S. Forest Service
Tom P. Evans
Indiana University, United States

This presentation focuses on how open-source programming and the more recent movement in open content licensing might provide the building blocks for a new paradigm in collaborative scientific research.

There are some challenges in communication that we all are aware of, including the space limitations of paper media; the lack of library resources, particularly in developing countries; the challenges of scientific repeatability; and, often, distributed and uncoordinated research efforts.

Peter Drucker made an observation about innovation lags that occur when new technologies are introduced.[1] In his paper he provides the example of a lag in railroad innovation in the United States in the

nineteenth century. He states that railroads were first designed to move only people, and it took nearly 30 years before they were used to move freight. We would like to propose that in terms of the Internet, the Web, and the way scientific information is shared (e.g., current e-journals), we are currently experiencing such an innovation lag.

With the emergence of the Web the potential exists to innovate in terms of how scientific research is shared and Internet-based collaboration is accomplished. Instead of providing only papers with final results, the Web creates the opportunity to make available scientific content at various stages of the research process. Incentives for scientist participation are key, and therefore the idea of a peer-reviewed e-journal is vital. Consequently, we envision future Web-based e-journals as containing information at all stages of the research process. Systems of peer review could be instituted at each stage to ensure quality control and to provide an incentive for participation. This next-generation e-journal would also apply principles used in open-source programming collaboration, as well as open-content licensing, to promote distance learning between scientists and to speed up the innovation process. Such a collaborative paradigm could increase the ability to harness global collective action toward solving many difficult and complex problems that humanity faces and could have important implications for greater participation and communication between scientists and academics worldwide.

To describe this vision more fully this presentation will first discuss the general design principles of open-source projects based on research of literature on open-source programming, and some of our own preliminary empirical research. We will then discuss why we see the open-source model combined with the emerging phenomenon of open-content licensing as a potentially new paradigm for global collaborative scientific research. The

1 P. F. Drucker. 1999. "Beyond the Information Revolution," *The Atlantic Monthly*, October. Available at http://www.theatlantic.com/issues/99oct/9910drucker.htm.

presentation closes with an example of our vision that we are trying to initiate: an open-source/open-content approach applied to land use/land cover change modeling.

PRINCIPLES OF OPEN-SOURCE PROJECTS

The basic principles of open source are grounded upon an innovation in software licensing. While there are a variety of open-source license types, in general they require the free distribution of the software coupled with readable source code. These licenses often allow new derivative works to be developed from the digital content (program source code). If you are a programmer, you can interpret the programming logic and have the opportunity to contribute new functionality to that software, with one catch: many open-source licenses (e.g., the GNU General Public License) are of a "viral" nature, requiring improvements to follow the same licensing agreement. The licensing in open source is a critical component in how these projects work.

The Linux operating system and the Apache Web server are two high-profile open-source success stories. They provide examples of how the open-source collaborative paradigm can produce solutions to very complex problems. The question is whether these are anomalies or whether there is really something to this collaborative approach that could be applied in other complex problem contexts.

To explore this we should understand better how these projects work. Based upon a review of recent literature,[2] a key attribute of Internet-based open-source collaboration projects is an established team of volunteer programmers and testers. In some cases (Linux being one prime example) there are organizations (e.g., IBM) that actually pay employees to work on the project.

Modularity and parallel development are also important design principles for open-source projects. The idea is that by dividing the project into modules or small compartments, people around the world could select a particular module to improve, without interfering with anyone else who is working on the project. Improved modules can be resubmitted back to the project, subject to a peer review. If the improvement is deemed useful, it is added to a future release of the software.

Typically open-source projects are initiated by an individual or a small group who have a critical need for software functionality that is currently unavailable or is too expensive to purchase. They then develop a "kernel," a core piece of software with some promise for potential innovation and growth. For a high-quality project to emerge highly skilled or prominent

people in this particular area are required in order to give it promise for later participatory growth.

Internet-based collaborative infrastructure is also important for open-source projects to work. Here version control systems are crucial. These systems keep track of various changes to existing modules and allow the project administration to control and keep a record of different releases of the software. The Concurrent Versioning System is one of the most popular softwares of this type and is used in many open-source programming projects.

Project governance—including rules related to participation and conflict resolution and norms of behavior—is another potentially important aspect of open-source collaboration. To date there is very little literature addressing this subject,[3] but our hypothesis is that this is a very important factor that leads to the success or failure of these projects.

For many projects the goal is ultimately to achieve growth both in developer participation and in a user base. From a developer perspective this is really the law of numbers, where with more eyes problems are more easily solved.[4] The idea is to get a large participating group, hopefully globally, working on these problems. The high-profile success stories have achieved this; Linux and Apache enjoy a large community of developers with regional coordinators working in many languages. In the case of Linux there are approximately 18 different languages represented, and it is a potentially complex system of coordination and core staff.

2 See C. M. Schweik and A. Semenov. 2003. "The Institutional Design of 'Open Source' Programming: Implications for Addressing Complex Public Policy and Management Problems," *First Monday* 8(1), at http://www.firstmonday.org/issues/issue8_1/schweik/.

3 To examine some of the active open-source programming projects, visit http://sourceforge.net.

4 E. Raymond. 1998. "The Cathedral and the Bazaar," *First Monday*, Vol. 3, No. 3, available at http://www.firstmonday.dk/issues/issue3_3/raymond/.

One aspect of open source that has puzzled scholars (particularly economists) is the question of incentives to participate. Why would people

voluntarily and freely contribute their intellectual property to such an endeavor? Recent research has identified several reasons. The first is related to intrinsic motivations. Some people enjoy this work. They like programming and they find it interesting and creative to participate in these endeavors. Second, self-esteem sometimes plays a role. People often want to feel like they are part of a community and are contributing to some important endeavor. Third, there are altruistic and social and political motivations. The altruistic motivation for many open-source programmers is based on the idea that software should be free. Political motivations also can be important, as in Linux, where programmers participate in part to take on a perceived monopoly.

More recently analyses have shown that there are primary economic motivations, either to build skills for future economic gain or to self-promote. Regarding skill building, open source provides a valuable distance-learning function; over the Internet the programmer can look at the source code, learn about how other people approach a programming problem, try to do their own enhancements, and then participate in a system of peer review in which they receive critical feedback. This distance-learning attribute of open-source projects can be a significant motivator for participation. Self-promotion is another economic-related motivator. Becoming known in the community often leads to potential consulting or job opportunities.

Another motivation often reported in the open-source literature is personal need. There may be cases in which no software is readily available for a particular function or purpose. An individual programmer might try and work on the project but realizes it is a large or complex task. The programmer realizes he or she cannot do it alone and thereby embraces the open-source paradigm with the hope that someone else will share the work at hand.

This brings us to a very important point about the above summary of the characteristics of open-source projects: it is primarily based on available literature at the time of writing this presentation (February 2003), and this literature is simplistic and biased. One of the problems with the conceptualization of open-source projects as described above and its potential is that there is considerable hype surrounding it. This is largely because the literature to date focuses centrally on such high-growth

success stories as Linux and Apache and ignores what could be thousands of failed projects.

There are many open-source projects that have been initiated, but of the more than 50,000 currently listed,[5] many never will achieve the level of success (in terms of growth of a participant base) of a Linux or an Apache Web Server. In fact it is highly likely that a large percentage of the thousands of open-source software projects are written by individual college students who decided to license it as open source and place it on the Web. To look at the open-source paradigm as a collaboration paradigm there is a critical need to study the factors that lead to successful cases (however it might be defined) and also unsuccessful (failed) cases. Our earlier discussion about the lack of research on the structure of open-source governance mechanisms is a case in point. There should be a major theoretical and systematic research endeavor (which is now beginning)[6] that examines these projects more deeply and identifies the factors and institutional designs that lead to successful high-growth or failed open-source collaborations. The fact that collaborations like Linux, Apache, and other open-source software exist and thrive suggests that there is something very interesting about the concepts of open-source (and -content) licensing, and the collaborative systems that have achieved them. We should understand better what factors lead to successful collaborations, not only so that other software might be developed in this manner but also and more importantly because it might provide a new way for humans to collaborate on a global scale on difficult scientific problems facing humanity; this approach could have important implications related to increasing participation from scientists in countries worldwide.

OPEN SOURCE/OPEN CONTENT AS A NEW PARADIGM FOR COLLABORATIVE GLOBAL RESEARCH

One could consider software development as one type of scientific endeavor. And one could argue that the principles of open source provide an opportunity to greatly increase the speed at which new innovation is achieved.

5 For example, see http://sourceforge.net

6 See, for example, the latest workshop on open-source software engineering, available at: http://opensource.ucc.ie/icse2003/.

There are several reasons for this. First, the incremental publishing feature allows much faster communication of new findings. Second, because of its open-access nature and because it is Internet based, the open-source paradigm has the potential of reaching a larger, potentially global audience. Third, the distance-learning attribute of the open-source approach provides an incentive for many to participate.

Let us expand a bit on this third point. In his book, *The Future of Ideas*,[7] Lawrence Lessig explains that the tremendous growth on the Web from 1994 to 2000 was based on two factors. The first is the end-to-end design of the Internet. The Internet as a communication system was designed with a relatively simple transmission protocol. The complexity is largely introduced at the end points, namely the end-user computers and servers. Programmers could generally rely on the fact that the transmission protocol will continue to follow the simple structure and design, thereby allowing data transmission through the network to be very simple and standardized. This allows them to place more sophisticated programming logic—the innovations—at the end points of the Internet (browsers and servers).

The second factor leading to the exponential growth of the Web is that the designers of Web browsers such as Netscape and Internet Explorer provided a "view source" option under their menus. In early days of the Web development (e.g., 1994 to 2000) the way people learned how to program Web pages was by visiting other people's Web pages with these browsers, viewing their source code using the "view source" option, examining what they did, and then developing their own Web forms based on that new knowledge. This leads us to some key points: While not acknowledged as such, the Web is arguably the largest and most successful distance-learning program in history, and it has led to tremendous innovation over the past six to eight years. Even through Web pages were not licensed specifically as open source, the "view source" option in browsers meant that the Web pages at this point were indeed source that was open access. The exponential growth in Web pages during this six-

year period provides an important example of the kind of innovation that is possible by following an open-source approach.

Very recently the ideas of open-source licensing have moved into the broader domain of "open content." Essentially, open content extends the "copyleft" principles of open-source licensing into broader areas of any form of intellectual property. Any digital content, from music to academic papers, could be licensed in a similar fashion to that of open-source programs, where end users of the content are given permission to freely copy, distribute, and possibly derive new works based on the content. The Creative Commons project,[8] recently initiated by Lawrence Lessig and his colleagues, provides 11 different open-content licenses based on the answers of four different questions: (1) Is free copying and distribution permitted? (2) Is author attribution required? (3) Can derivative works be made from this content? (4) Can the content be used in commercial applications without permission? Different combinations of the answers to these questions create a spectrum of intellectual property rights, from full copyright on the one end to public domain on the other. By raising attention to this spectrum Lessig and colleagues show that authors of new content have more choice than just going with the traditional "copyright—all rights reserved" to "copyright—some rights reserved." For example, their "by attribution" license allows others to copy, distribute, display, and use copyrighted work, as well as produce new derivatives from this work, as long as they acknowledge the previous author. Adding their "no derivative works" license with "by attribution" means that people can copy, distribute, display, and use the work verbatim, but cannot define or develop new work using it.

Over the last year examples of what we refer to as open-content experiments have emerged. We refer to these as experiments because they have just started, and it is not clear which ones will actually succeed and which will fail. The oldest examples of this kind of idea include Project Gutenberg[9] (a public-domain project), which is placing e-books in the public domain on the Internet, the Free Software Foundation's GNU Free Documentation license[10] to support the open-content development of software documentation, and the Massachusetts Institute of Technology OpenCourseWare[11] project, in which academics share course content using open-content licensing.

7 L. Lessig. 2001. *The Future of Ideas: Fate of the Commons in a Connected World*, Random House Publishing, New York.

8 See http://creativecommons.org.

9 See http://promo.net/pg/.

10 See http://www.gnu.org/licenses/licenses.html#TOCFDL.

11 See http://ocw.mit.edu/OcwWeb/index.htm.

OPEN SOURCE/OPEN CONTENT AND THE SCIENTIFIC ENDEAVOR: AN EXAMPLE IN THE CONTEXT OF LAND COVER CHANGE MODELING

We have been working on an interdisciplinary project studying how land cover is changing and how computer-based models can be created to capture land cover change dynamics. This is an issue of great worldwide interest. Local governments, for example, are interested in tools that might help them project future scenarios of sprawl and how various public policies might change such projections. The global change community wishes to use these kinds of models to understand how forests may change and what that might mean for the global climate system. The challenge in developing these models is that the system that drives land cover change is often extremely complex and involves insights from a variety of scientific disciplines. It is a problem that could involve a large number of participants across the globe, including academics, scientists, policy analysts, and local and regional governments.

In 2002 we conducted a review of existing land cover change models and identified more than twenty developed by various organizations and funded by a variety of agencies.[12] These models can be extremely complex to use and understand because of the different technologies and academic disciplines that are represented within them. Our ability to build on advances made in these models is relatively low, mainly because of the significant costs involved in acquiring needed software technologies, and learning and applying these models once the technical infrastructure is available. We see the open-source/open-content approach as a way to move beyond the status quo and possibly to speed up our ability to build

on another's work much more rapidly than we have to date. Let us use this as an example of how we might apply open source to a collaborative, scientific endeavor.

To implement an open-source/open-content approach in this area the first step would be to identify a core group of willing participants. This would include modelers, who are similar to the programmers in open-source programming; data providers; scientists and academics; and other professionals who contribute theoretical arguments into these models, as well as the practitioners and other stakeholders who might be users and who also might be able to provide input into these models. The number and diversity of participants would be greater than in open-source programming.

The next step would be to identify existing models that could possibly be placed under an open-source/open-content license. In this context we would extend the idea of the open-source software "kernel" to land use models, making them available with an open-source license. Unlike traditional programming, in this scientific endeavor there would be multiple kernel types. The model itself (in whatever technology or approach) would be one type of kernel. These models are often informed by theoretical work and this content is often in the form of published or unpublished papers. These "theory kernels" could be placed under an open-content license. Data required to run the models could be considered a third type of kernel that could also fall under an open-content license. In the initiation phase all these land use modeling components or kernels would be modular to support parallel development.

An important consideration at the initiation of an open-source/open-content based scientific endeavor is the incentive structure to encourage participation by scientists worldwide. Interestingly the motivations of programmers in open-source projects and the motivations of scientific and academic researchers are very similar. They share the same intrinsic motivations. Researchers often participate in the creative process because they enjoy it or find the subject interesting or important to contribute intellectual property to it. There can be a self-esteem component that motivates their participation as well, a feeling that they are participating in a broader community of interested scholars. In terms of altruistic and social political motivations, researchers are often motivated in their work because they are trying to solve a problem facing humanity. There is also

the belief among many researchers that, like software, knowledge should be free, which might motivate them to participate in an open-content project. As such, there is a potential social and political movement very similar to the free software argument. In terms of the personal needs motivation that drives many open-source programmers, instead of a software gap there is often

12 C. Agarwal, G. M. Green, J. M. Grove, T. P. Evans, and C.M. Schweik. 2002. A *review and assessment of land-use change models: Dynamics of space, time, and human choice*, Gen. Tech. Rep. NE-297, U.S. Department of Agriculture, Forest Service, Northeastern Research Station, Newtown Square, PA., p. 61. Available at: http://www.srs.fs.usda.gov/pubs/viewpub.jsp?index=5027.

a scientific knowledge gap. The knowledge gap is complicated and often requires multiple disciplines, and researchers find themselves needing the assistance of others with other expertise.

The motivations described above in part could encourage scientists to participate in an open-source/open-content research effort. But as in open-source programming, we believe the most significant motivation is along the economic dimension—to gain new skills and to self-promote one's ideas. Researchers might participate in an open-source/open-content scientific collaboration to building new skills. For example, the global body of graduate students interested in land use change issues would be motivated to participate in such a project because of the benefits of what they could learn. Having research products as "open" provides the ability for one to distance learn the way scientists already distance learn by reading academic journals. Open-source/open-content collaboration would provide a setting for peer review as in open-source programming. Peer review is an important way that scientists learn from each other and it is a way for us to make advances in our own skills and knowledge. Self-promotion, or becoming known in the field that you are studying, is a motivation that is particularly important for academics, and especially for junior faculty who are trying to make a name for themselves. Graduate students could possibly gain some real self-promotion if they contributed an important new module (e.g., theory or computer code) to the effort. In short, the motivations driving potential participation in open-content

scientific collaboration are very similar to what motivates some open-source programmers to participate in those projects.

One of the great challenges to the open-source/open-content vision applied to the scientific endeavor is the traditional way scientists are promoted through peer-reviewed publishing. Junior (and higher-level) researchers in university settings are often evaluated on their publishing record and therefore have a strong motivation to protect their intellectual property accepted through traditional publishing media. The concept of a next-generation peer-reviewed e-journal is important, because it considers these important incentives for participation. What we are suggesting is that these next-generation e-journals provide mechanisms to "publish" various forms of intellectual property. In our example of land cover change modeling this might include papers on final results, but also papers on theoretical inputs that inform a model design, papers on how to use the model, as well as the other important products for the modeling endeavor, such as data sets and the model source or logic itself. If a researcher contributed a new module to the model (e.g., extended the model's functionality), this would be peer reviewed and considered a publication as well.

What this means is that there is an important need to figure out how to document intellectual property contributions for all three of those kernels —models, data, and the theoretical contributions. Can all these components of the research process be treated as a form of publishing, or possibly service contributions? This means that we will have to establish a system of governance and rules of operation for this next-generation e-journal (e.g., editorial boards for all kernel types), as well as effective conflict resolution mechanisms to govern a debate about which direction the modeling process should go.[13]

Creating the next-generation e-journal would also involve the selection of appropriate open-content licenses for these kernel types and the establishment of systems of peer review—not only for papers, which may be the theory or empirical work, but also for data and models. The final component would be to establish a project infrastructure, which would include the communication and version control systems.

Just as in the case of today's e-journals, theory, empirical research, and results would be critical submissions. In the next-generation e-journal we could see following the model of traditional volumes and issues, and also

the ability to produce incremental releases. In the electronic world there is no necessity to "print" a particular issue, rather you can just build it as new contributions are submitted and peer reviewed. Moreover, in land cover change modeling the opportunity exists to make hyperlink connections between related kernels. For example, links could be implemented that make it easy to see which data feed a certain model, or which theory (papers) informed a particular modeling logic.

We are associated with several major research groups interested in land cover change modeling, including the U.S. National Science Foundation's Long-term Ecological Research Network and Human Dimensions of Global Change research network. Along with the U.S. Department of Agriculture Forest Service we are trying to develop a core group of people interested in starting open-source/open-content collaborative research effort on land cover change modeling. At the time of this writing we have established a working group and have held a workshop to initiate the endeavor. Our short-term goal is to develop an institutional design around the collaboration, including the identification of available models, establishment of metadata standards for data and model modules, identification of the types of open-source/open-content licenses for various kernels, and the establishment of required communication systems and version control systems. Over the longer term we would like to move to a next-generation e-journal, perhaps in collaboration with an existing e-journal.

13 Conflict resolution mechanisms are not discussed in the open-source literature. For example, there must be conflicts between two modules that are competing with one another; how is it decided which goes into the next release of software?

CONCLUSION

Several conclusions can be made from this discussion. First, there is a great need for more in-depth research on open-source programming and how these projects work if we want to capitalize on them in the broader context of open content. The literature up until early 2003 is fairly naive on how open-source projects are structured, and a major point of our talk was that

there is a desperate need to understand what leads to success and failure of these projects, rather than basing all our knowledge on just the high-profile success stories, Linux and Apache Web Server, which may be anomalies. Fortunately more carefully crafted quantitative studies and deeper analyses on these projects are beginning to emerge.

Second, we should watch the open-content experiments and understand which ones are successful. There is, as far as we can see, little research on how these projects are managed. We are trying to initiate this kind of research.

Third, in scholarly (global) communication we should move beyond Drucker's innovation lag in current e-journals that follow the old volume-and-issue paper model and encourage the development of the next generation of e-journals that takes advantage of all the Web can offer. We are suggesting that convergence with the open-content licensing phenomenon might be a component of this, and that the idea of not only peer-reviewed papers but also data and other computer-based scientific research (e.g., computer models) will be important. This lag in scientific publication is similar to the current e-commerce and e-government movements, which have been gradually moving from simple Internet publishing of information to more sophisticated online transaction processing.

Our major point is that the continued growth of access to the Internet globally, coupled with the design of open-source licensing and collaborative principles and the emerging trend in open-content licensing point to a new, potentially significant way for humanity to tackle complex problems in science and other domains. There is still much to be learned about what works and what does not in open-source-like collaboration. But the promise is there; successful global, Internet-based collaboration of complex problems in open-source programming (e.g., Linux and Apache) suggest that there is a way to achieve global collective action in ways never before possible. It may lead to no less than a paradigm shift in the way new scientific knowledge can be generated and the way new learning is shared. Assuming Internet infrastructure continues to be built in countries that lack access, this open-content (e.g., open access), next-generation e-journal we envision and open-content collaborative projects could dramatically improve the sharing of knowledge and participation in all corners of the world.

The National Academies of Sciences, Engineering, and Medicine
500 Fifth St., NW | Washington, DC 20001