An Educational Open Source Development Model

From Cooperative Synchronicity to Intentional Collaboration

By Judi Harris and Kathleen Owings Swan

As educators, we have a long tradition of sharing materials and ideas. Another professional community has almost as long a history of idea and materials exchange and revision: an international network of open source software developers connected by the Internet. Working primarily as volunteers, this community has produced and fine-tuned many pieces of software, including the Internet's most popular Web server, the Apache Web server.

What if talented software developers could work with educators to develop and customize educational software? In addition, what if a mechanism were in place to help educators share the materials that they create in this virtually collaborative manner?

Communities and Roles

Development of educational open source software could—and probably should—be rooted in a new kind of cross-community collaboration. The nature of this collaboration is necessitated by the dispositions and work habits of the two communities whose motivations, values, and styles must be accommodated in this collaborative process: K–12 educators and open source programmers. Fortunately, members of both communities were represented in the Collaboration and Community task force of the NTLS.

It did not take long to understand that these two communities work, communicate, and are rewarded very differently. Almost immediately, we recognized the need for liaisons who comprehend both cultures to translate between them if members of both are to work together successfully to create educational open source materials. As our discussion continued, we realized that the liaison's work is quite complex; it expands to incorporate many roles.

In Table 1, we summarize the roles that educators, liaisons, and developers would play in the development of open source materials in our model. Please note that although we use the specific example of open source software development here, we see this model, with minor modifications, also describing the creation and refinement of non-computer-based educational materials.

As Table 1 indicates, the liaison's role in this collaboration is the most complex of the three. As such, it probably requires talents and prior experiences that are more rare than either of the other two roles. Liaisons need to be able to understand and function well in both groups. Technology-savvy teachers, school-based educational technology coordinators, university professors and graduate students interested in educational technology integration, and creatively/technologically gifted students are potential liaisons.

Please note that although we use the specific example of open source software development here, we see this model, with minor modifications, also describing the creation and refinement of non-computer-based educational materials.

Additional roles and responsibilities of these individuals include:

- Determine, in consultation with educators, which code revisions become elements of the core product and which do not (i.e., work to prevent "code forking"). Focus on the learning that can be supported by the software rather than the specifics of code.
- Build project "buy-in" and sustainability. Rewards are key (Table 2).
- Translate between educator and developer subcultures (e.g., explaining open source to educators).
- Serve as project champion, manager, facilitator, and "connector."

Development and refinement of each educational open source project will probably progress in identifiable stages, with the specific roles that each participant plays shifting over time as work on the project progresses. More importantly, roles played by team members collaborating on the same project are likely to shift according to individual interests, expertise, and motivations. This flexibility and fluidity of roles is crucial to the success of each project and to the educational open source movement itself. (We suspect that the educational open source movement, or metaproject, will develop in stages that are similar to project-related stages.) At all times, though, all roles must be filled in a particular project for the endeavor to succeed.

Rewards

The work done in each of the roles described in Table 1 is primarily voluntary. Though some open source projects are funded by grants or as works for hire, the impetus for the typical open source project at present is other than financial. Any feasible and sustainable educational open source development model must, therefore, take into account the varying—and hopefully complementary—motivations of the participants playing each of the roles with reference to a particular project. Table 2 outlines probable motivations for educational open source work.

The roles and corresponding rewards described in Table 2 will probably emerge as much from the nature of inspired work already extant in the educational and software development communities as from the new experiences and relationships to be found in future educational open source work.

Beginnings

At present, collaborations between educators and open source software developers are rare and rarely sustained. To shift this pattern, the first rounds of educational open source software will have to be planned and supported in strategic, somewhat preplanned ways. In a sense, we will have to create first artificially assembled collaborative teams to explore and document the as-yet-undiscovered intricacies of the bridged intercultural collaboration suggested earlier.

For the new working relationships between disparate subcultures to become organically initiated and sustained—and for the same to happen within the open source in education metaproject—a high-quality collection of well-received educational open source products must be conceived, developed, tested, and publicized. The instructional applicability and diffusibility of this first group of projects will, to a large extent, determine the success of the movement itself. Therefore, we strongly suggest that early educational open source workers not only seek gov-

This flexibility and fluidity of roles is crucial to the success of each project and to the educational open source movement itself.

Table 1. Open Source Development Roles

Educators/Students	*Liaisons*	*Software Developers*
Identify software needs.	Find educators and programmers to work together at the beginning of the project.	Listen and respond with code.
Provide feedback throughout development.	Locate resources to initiate projects.	Cooperate with other developers to refine code.
Review software content for accuracy and comprehensiveness.	Facilitate "development loops": recursive project development, testing, and tweaking.	Transfer core code to liaisons.
Generate further ideas and initiate revision cycles	Identify programmers to work on recursions of core project code and identify next project champion when he or she leaves the project.	Revise code in response to feedback.

Table 2. Open Source Development Role-Related Rewards

Educators	*Liaisons*	*Software Developers*
Acquire and use customized educational materials that help students learn content and process in more effective ways.	Solve a problem or address an educational need for themselves and/or others whom they care about (students and other educators).	Challenge of solving an authentic problem that nobody else has solved as well. Mental stimulation.
Access to software that is more applicable and adaptable to students' learning needs than other software.	Interpersonal networking for professional development.	List on résumé if others use the project widely (proof of concept).
Materials can't be taken away and can't expire.	Career advancement and publishing.	Active participation in "something bigger than themselves." Involvement in a community effort.
Positive publicity for being involved in an innovative endeavor.	Positive publicity for being involved in an innovative endeavor.	Positive (and permanent) publicity for being involved in an innovative and successful endeavor.
Professional acknowledgement by other educators, students, and community members.	May be part (or may become part) of the regular job description.	Altruism and egoism
	Reduce isolation of being an educator.	

At present, collaborations between educators and open source software developers are rare and rarely sustained. To shift this pattern, the first rounds of educational open source software will have to be planned and supported in strategic, somewhat preplanned ways.

ernment and private funding to make this first stage of software development possible, but also choose carefully the projects to be created. This first set of software will then have the best possible chance at widespread adoption in K–12 classrooms by virtue of the applications' inherent characteristics.

The success of the movement itself rests on nothing less than our perceptiveness about educational software needs and preferences, our knowledge of the nature of the two subcultures involved, and the adoptability of a relatively small number of particular products. Our ability to use these perceptions in creative, generative ways through the design, testing, and publicizing of educational software is the key to success or failure on both the individual project and metaproject levels. Will we be able to incorporate the open source community's cooperative synchronicity in software development into a cross-cultural model of intentional, sustainable collaboration? Only time will reveal the answer.

First Steps

The programmer-based open source movement is about a dozen years old. Surely in a decade or so, we will look back at assumptions undergirding the model suggested here and both nod at their verity and smile at our naïveté. Though we believe that the best approach to this endeavor is clear-sighted, strategic, and proactive, we also acknowledge that the best ways for the two communities to collaborate will necessarily emerge over time and through repeated, reflective trials. The next step, therefore, is to begin.

Judi Harris is the Pavey Chair in Educational Technology in the School of Education at the College of William & Mary. Her scholarship and service focus on curriculum-based telecomputing, telementoring, and authentic professional development. Judi directs The Electronic Emissary (http://emissary.ots.utexas.edu/emissary/), a K–12 curriculum-oriented telementoring service and research effort. Judi was the founding director for WINGS Online (http://emissary.ots.utexas.edu/wings/), a suite of online support services for novice teachers at the University of Texas at Austin.

Kathleen Owings Swan is a graduate instructor in the social studies program of the Curry School of Education at the University of Virginia. She is a graduate fellow in the Curry Center for Technology and Teacher Education.