Helping Faculty & Students to Participate in Humanitarian Free & Open Source Software: The OpenFE & OpenPath Projects

Mr. Clifton L Kussmaul, Muhlenberg College

Clif Kussmaul is Associate Professor of Computer Science at Muhlenberg College. Previously, he was Visiting Fulbright-Nehru Scholar at the University of Kerala, Chief Technology Officer for Elegance Technologies, Inc., Senior Member of Technical Staff with NeST Technologies, and Assistant Professor of CS at Moravian College. He has a PhD in Computer Science from the University of California, Davis, master's degrees in CS and Electro-acoustic Music from Dartmouth College, and bachelor's degrees in Engineering and Music from Swarthmore College. His professional interests and activities include active and guided inquiry learning, software engineering, entrepreneurship, digital signal processing, cognitive neuroscience, and music.

Heidi Ellis, Western New England University

Heidi Ellis is a Professor in the Computer Science and Information Technology department at Western New England University. Dr. Ellis has a long-time interest in software engineering education and has been interested in student participation in Humanitarian Free and Open Source Software (HFOSS) since 2006.

Dr. Gregory W Hislop, Drexel University (Eng. & Eng. Tech.)

Gregory Hislop is a Professor and Senior Associate Dean in the College of Computing and Informatics at Drexel University. His scholarly interests span computing education research, information technology for teaching and learning, and software engineering. Prior to coming to Drexel, Dr. Hislop spent 18 years working in government and industry, where his efforts included software development and support, technology planning and evaluation, and development and delivery of technical education.

Lori Postner, Nassau Community College
Darci Burdge, Nassau Community College

Darci Burdge is Assistant Chair and Professor of Computer Science at Nassau Community College. She has worked to increase students understanding of software development and the impact it can have on society. She is especially interested in broadening the perspective of the introductory Computer Science student beyond the programming concepts typically taught in these courses. She uses HFOSS projects as a means to providing real-world experience and finds that students are motivated, showing increased participation in classroom discussion especially among women. She is Co-PI on an NSF-funded project to assist faculty who are interested in involving students in HFOSS projects.

Helping Faculty & Students to Participate in Humanitarian Free & Open Source Software: The OpenFE & OpenPath Projects

Abstract

Students in computer science, software engineering, and related disciplines must master a broad range of technical knowledge, skills, tools, and processes. They must also learn to navigate, understand, and contribute to real-world code, documentation, and diverse communities of developers, users, and other stakeholders. One effective way for students to develop such knowledge, skills, and attitudes is to participate in Humanitarian Free & Open Source Software (HFOSS) projects. Research has shown that student participation in HFOSS projects has a positive impact on student motivation to study computing and a strong positive impact on perceived learning related to software engineering [1,2]. The OpenFE and OpenPath projects seek to help faculty and students participate in HFOSS projects and communities.

The 2012 NSF TUES OpenFE Project developed and expanded faculty expertise supporting student involvement in HFOSS projects, and created and evaluated learning materials to help students in a variety of settings, including community colleges. OpenFE significantly revised and enhanced the Professors Open Source Software Experience (POSSE), a faculty development program originally developed by Red Hat, Inc. OpenFE also helped faculty to develop, pilot, and disseminate a variety of learning materials [3].

The 2015 NSF IUSE OpenPath Project continues and expands this work with a focus on sequences (pathways) of topics and learning activities that can be integrated across the curriculum to provide faculty and students with a more gradual introduction to FOSS tools and practices. OpenPath also leverages Process Oriented Guided Inquiry Learning (POGIL) [4,5] to help students develop skills in communication, critical thinking, problem solving, and teamwork, which will make them more successful participants in HFOSS [6].

Together, the OpenFE and OpenPath projects have supported over 90 faculty from over 65 institutions to participate in POSSE and explore ways to help their students participate in HFOSS projects. A website (http://foss2serve.org) has more information about OpenFE and OpenPath, POSSE and other events, and learning materials.

Introduction

To prepare students for professional practice in computing careers, most undergraduate computing degree programs seek to provide a complex mix of knowledge and skills [7]. In order to contribute to large, ongoing software projects, students must master fundamental technical topics as well as skills in teamwork, communication, and problem solving. They must also develop personal attributes including collegiality, a sense of social responsibility, and appreciation for diversity [7,pg 15].

Introducing students to the scale and complexity of real computing environments is an excellent way to address these professional practice challenges, and builds on general learning research. Learning in an authentic context is important to develop expertise, which requires not only knowledge but also understanding the contexts in which it applies [8]. A sense of contributing to others and seeing the usefulness of the content provides a strong social motivation for learning [9].

Humanitarian Free & Open Source Software (HFOSS)

Students who participate in Free and Open Source Software (FOSS) projects interact with a distributed community while creating evidence of professional accomplishment [10,11]. Student participation can take many forms, including coding, testing, bug verification and correction, documentation, product packaging, and community development [12,13]. A growing community of educators is working on student participation in FOSS (see http://TeachingOpenSource.org).

Humanitarian Free and Open Source Software (HFOSS) improves the human condition in areas such as health care, disaster management, education, economic development, and accessibility. The HFOSS culture of openness is about community more than source code. Community openness exposes the work processes, plans, operational tasks, and software artifacts associated with large, evolving software systems that serve real clients. The culture also encourages participation by volunteers, including students. Student participation in HFOSS projects has shown potential to improve student learning, motivate the study of computing, attract more women to computing, and develop appreciation of the societal impact of computing [14,15]. Note: The OpenFE and OpenPath projects focus exclusively on HFOSS projects, so HFOSS is used below.

To help faculty learn about FOSS and support student participation, Red Hat, Inc. developed a multi-day workshop titled *Professor's Open Source Software Experience (POSSE)*, and ran nine POSSEs that reached 70 faculty. However, the workshops were stand-alone events and faculty had difficulty implementing the content afterwards.

OpenFE Goals & Activities

The 2012 NSF TUES OpenFE Project sought to develop faculty expertise, build community, and create learning materials for student participation in HFOSS. The goals and activities of OpenFE are summarized below.

Goal 1: Increase the number of instructors using HFOSS learning activities and build supportive communities for instructors. Many instructors have no experience with open source communities or their tools and few computing colleagues at their schools with whom to share development and piloting of new teaching approaches. Thus, OpenFE sought to: collaborate with Red Hat to expand and enhance POSSE to help instructors create and apply learning activities for HFOSS; establish online instructor learning groups to work together after POSSE; and help instructors and student become core participants in HFOSS projects, in order to improve sustainability and scalability.

Goal 2: Apply learning strategies of open source communities to undergraduate instruction. To engage students and foster learning, OpenFE will employ a range of learning activities built around HFOSS communities, and approach them as communities of practice and have students start from peripheral participation and progress to more central participation. Thus, OpenFE sought to: analyze FOSS learning strategies and adapt them for undergraduate education; use societal impact to motivate students and increase interest in studying computing; and evaluate how HFOSS participation affects student learning, using both indirect and direct measures.

Goal 3: Create HFOSS learning materials to improve computing education. A variety of HFOSS learning materials already exist. However, HFOSS learning appears to be applicable to much wider ranges of students, activities, and learning outcomes [13]. Thus, OpenFE sought to: develop and evaluate a wider range of HFOSS learning activities that recognize these ranges and common approaches to learning in HFOSS communities; map these activities to the ACM curriculum recommendations; and develop materials for a full course focused on HFOSS participation.

OpenFE Outcomes

OpenFE collaborated with Red Hat to evolve POSSE into three stages: Stage 1 is a series of online learning activities and virtual meetings; Stage 2 is a 2.5 day face-to-face meeting; and Stage 3 involves small learning groups to reinforce learning and support faculty members as they involve students in HFOSS.

OpenFE provided evidence that student participation in HFOSS has a positive impact on: student motivation to study computing; perceived learning related to software engineering; and major and career plans (weak evidence). OpenFE produced multiple publications [3,13,15,16-26] as well as course materials for getting both students and faculty members involved in HFOSS projects (see http://foss2serve.org). OpenFE supported a growing community of instructors who are involving students in HFOSS. OpenFE also improved understanding of:

- Possible roadblocks to student participation in HFOSS projects.
- The types for instructor support needed for student involvement in HFOSS.
- Guidelines for faculty members when involving students in an HFOSS project.
- Approaches to structure learning within HFOSS participation.
- Approaches to HFOSS project selection for use in a class.

Process Oriented Guide Inquiry Learning (POGIL)

In Process-Oriented Guided Inquiry Learning (POGIL), students work in small, self-managed teams using specifically designed materials to construct their own knowledge. Individual students have team roles to help keep all team members engaged. Instructors act as facilitators, observing student teams and providing support where needed. POGIL originated in college chemistry teaching [27,28] and has since spread across STEM disciplines with over 1,000 implementers. POGIL has shown potential to improve learning, develop teamwork and key process skills, and encourage individual responsibility and meta-cognition [4]. The guided

learning applies a learning cycle of exploration, concept invention, and application. Thus, POGIL is an active, constructivist, collaborative, student-centered learning approach.

The NSF TUES CS-POGIL Project applied POGIL to the CS curriculum, including topics in data structures & algorithms, software engineering, artificial intelligence, and theory of computation, as well as introductory courses at the college and high school levels [e.g. 29,30,31].

OpenPath Goals & Activities

The 2015 NSF IUSE OpenPath Project proposes that HFOSS and POGIL can work synergistically and create a better pathway to take students from initial studies to professional practice in computing. This begins with POGIL activities to learn about computing and software development and progresses to HFOSS participation. POGIL will scaffold the early learning, improve learning outcomes and help students develop process skills including critical thinking, problem solving, communication, and teamwork. This will better prepare students to participate in less scaffolded environments like HFOSS. HFOSS participation will connect factual knowledge with practice, continue to develop process skills, and provide context and motivation. The goals and key activities of OpenPath are summarized below.

Goal 1: Define clear learning progressions that enable students to participate in HFOSS and that can be integrated within existing curricula. Instructors often have limited preparation time, too much material to cover, limited ability to change curricula, and other institutional incentives. Thus, OpenPath seeks to: develop a model for HFOSS learning that defines and sequences learning activities that lead to HFOSS participation; and map this model to computing curricula recommendations [7,32] to help faculty integrate elements of the model into their courses.

Goal 2: Develop learning activities to enable HFOSS participation. This goal will include POGIL activities as appropriate, and consider both curricular and extracurricular learning. OpenPath will build on existing materials from http://cspogil.org, http://foss2serve.org, http://OpenHatch.org, http://TeachingOpenSource.org, and other sources, as well as an inventory of ideas (known as “50 ways to be a FOSSer” [13]). Thus, OpenPath seeks to: develop new learning activities (including POGIL activities) based on the model and mappings described above; and explore the use of extracurricular activities, including one day events and ongoing club style activities.

Goal 3: Expand the communities of computing educators who enable student participation in HFOSS and use POGIL. Thus, OpenPath seeks to: continue professional development to expand the community; integrate POGIL into POSSE and provide additional POGIL training for faculty who plan to develop POGIL-style learning activities; and expand support for faculty learning groups after the face-to-face POSSE workshops.

OpenPath Outcomes

OpenPath is in its second year, and is making progress toward each of its goals. For goal 1, a model for HFOSS learning is in development and will be mapped to curricular models. Multiple

"pathways" have been identified, each of which leads to a recognizable contribution to a HFOSS project [6]. An initial set of these pathways have been developed in more detail, including learning objectives, curricular mapping, and identification of relevant process skills and learning activities. For goal 2, existing materials are being reviewed and refined, and new materials are in development. For goal 3, POSSE materials have been revised, and POGIL practices have been incorporated into several POSSE sessions so that instructors experience a POGIL classroom environment and better understand the benefits and limitations of POGIL [33].

Ongoing Effort

The OpenFE project is currently winding down while the OpenPath project is still ongoing with an expected end date of August 2019. The community of HFOSS educators continues to expand. A longitudinal study of the impact of POSSE is underway with a combination of questionnaire and structured interview being used to elicit opinions of POSSE alumni. The goal is to identify factors that hinder faculty and factors that aid faculty when supporting student involvement in HFOSS. In addition, both Google and Mozilla have reached out to the team and expressed interest in supporting student engagement in HFOSS.

Conclusions

This executive summary paper has described the motivation, objectives, activities, and outcomes for two NSF projects focused on faculty and student participation in Humanitarian Free & Open Source Software (HFOSS) projects and communities.OpenFE revised and enhanced the Professors Open Source Software Experience (POSSE), to develop and expand faculty expertise supporting student involvement in HFOSS projects. OpenFE also created and evaluated learning materials to help students in a variety of settings, including community colleges. OpenPath continues and expands this work with a focus on sequences (pathways) of topics and learning activities that can be integrated across the curriculum. OpenPath also leverages Process Oriented Guided Inquiry Learning (POGIL) to help students develop relevant skills.

These projects are also exploring a variety of future directions, including:

- Identify and document more ways that students in introductory courses can participate in and contribute to HFOSS projects.
- Identify and develop relationships with HFOSS projects in a wider range of application areas and technologies.
- Develop additional scaffolding for consistent, repeatable learning activities, particularly in introductory courses. This might include wikis, discussion forums, task trackers, and source repositories that can be cloned to provide sandbox environments for students to practice and develop skills before work with actual HFOSS communities.

Acknowledgements

This material is based on work supported by US National Science Foundation grants DUE-1225688, 1225708, and 1225738 (OpenFE), and DUE-1524877, 1524898, and 1525039 (OpenPath).

References

1. H.J.C. Ellis, G.W. Hislop, S. Jackson, and L. Postner. Team project experiences in Humanitarian Free and Open Source Software (HFOSS). *ACM Trans. on Computing Education*, 15, 4, Article 18 (December 2015), 23 pages. DOI=http://dx.doi.org/10.1145/2684812
2. G.W. Hislop, H.J.C. Ellis, S.M. Pulimood, B. Morgan, S .Mello-Stark, B. Coleman, and C. MacDonell. A multi-institutional study of learning via student involvement in Humanitarian Free and Open Source Software projects. *International Computing Education Research Workshop*, Omaha, NE, Aug. 2015.
3. [H23] H.J.C. Ellis, G.W. Hislop, and M. Purcell. Project selection for student participation in humanitarian FOSS. In *Proc. of the ACM SIGITE Conf. on Information Technology Education (SIGITE)*, 155-156, 2013. http://doi.acm.org/10.1145/2512276.2512326
4. R.S. Moog, and J.N. Spencer, (Eds.). *Process-Oriented Guided Inquiry Learning* (POGIL). Oxford University Press, 2008.
5. H.H. Hu, C. Kussmaul, B. Knaeble, C. Mayfield, and A. Yadav. Results from a survey of faculty adoption of Process Oriented Guided Inquiry Learning (POGIL) in Computer Science. In *Proc. of the ACM Conf. on Innovation and Technology in Computer Science Education* (ITiCSE) (pp. 186–191). July 2016. https://doi.org/10.1145/2899415.2899471
6. H.J.C. Ellis and G.W. Hislop. Pathways to student learning within HFOSS. (poster). In *Proceedings of the ACM Conference on IT Education (SIGITE)*, 2016.
7. The Joint Task Force on Computing Curricula. *Computer Science Curricula 2013: Curriculum Guidelines for Undergraduate Degree Programs in Computer Science*. ACM and IEEE Computer Society, 2013.
8. H.A. Simon. Problem solving and education. In *Problem Solving and Education: Issues in Teaching and Research*, D.T. Tuma and R. Reif, eds. Erlbaum, 1980.
9. J.D. Bransford, A.L. Brown, and R.R. Cocking, eds. *How People Learn: Brain, Mind, Experience, and School*. Committee on Developments in the Science of Learning and Committee on Learning Research and Educational Practice, Commission on Behavioral and Social Sciences and Education, National Research Council, 2000.
10. E. Stroulia, K. Bauer, M. Craig, K. Reid, and G. Wilson. Teaching distributed software engineering with UCOSP: the undergraduate capstone open-source project. In *Proc. of the 2011 Community Building Workshop on Collaborative Teaching of Globally Distributed Software Development (CTGDSD '11)*. ACM, 2011. DOI=10.1145/1984665.1984670 http://doi.acm.org/10.1145/1984665.1984670
11. S. Chacon and J. Borenstein. Modernizing CS education with open source. *Open Source Convention (OSCON)*, Portland OR, 2014.
12. R. Morelli and T. de Lanerolle. FOSS 101: Engaging introductory students in the open source movement. In *Proc. of the ACM Tech. Symp.on Computer Science Education (SIGCSE)*. ACM, 2009. DOI=10.1145/1508865.1508977 http://doi.acm.org/10.1145/1508865.1508977
13. C. Kussmaul, H.J.C. Ellis, and G.W. Hislop. 50 ways to be a FOSSer: Simple ways to involve students & faculty (abstract). In *Proc. of the ACM Tech. Symp.on Computer Science Education (SIGCSE)*, 671, 2012. http://doi.acm.org/10.1145/2157136.2157393

14. H.J.C. Ellis, G.W. Hislop, J. Rodriguez, and R.A. Morelli. Student software engineering learning via participation in humanitarian FOSS projects. In *Proc. of the ASEE Conference and Exhibition*, San Antonio, TX, 2012.
15. H.J.C. Ellis, L. Postner, G.W. Hislop, and Stoney Jackson. Team project experiences in Humanitarian Free and Open Source Software (HFOSS). *ACM Trans. on Computing Education, Special Issue on Team Projects in Computing Education*, 15, 4, article 18, 2015.
16. H.J.C. Ellis, M. Purcell, and G.W. Hislop. An approach for evaluating FOSS projects for student participation. In *Proc. of the ACM Tech. Symp.on Computer Science Education (SIGCSE)*, 415-420, 2012. http://doi.acm.org/10.1145/2157136.2157260
17. H.J.C. Ellis, G.W. Hislop, M. Purcell, and L. Postner. Project selection for student participation in humanitarian FOSS. *J. Comput. Sci. Coll.* 28, 6, 16-18, 2013.
18. H.J.C. Ellis and G.W. Hislop. Community-based student learning via participation in humanitarian FOSS projects. *J. Comput. Sci. Coll.* 28, 6, 196-198, 2013.
19. M. Purcell, H.J.C. Ellis, and G.W. Hislop. An approach for evaluating open source projects for student participation. *J. Comput. Sci. Coll*. 28, 6, 199-200, 2013.
20. H.J.C. Ellis, G.W. Hislop, M. Purcell, M. Chua, and S. Dzaillas. Towards a model of faculty development for FOSS in education. *Conf. on Software Engineering Education and Training (CSEE&T)*, San Francisco, CA, May. 2013.
21. H.J.C. Ellis, G.W. Hislop, and J. Diggs. Developing HFOSS projects using integrated teams across levels and institutions. In *Proc. of the ACM Conf. on Innovation and Technology in Computer Science Education (ITiCSE)*, 349-349, 2013. http://doi.acm.org/10.1145/2462476.2465613
22. H.J.C. Ellis, S. Jackson, D. Burdge, G.W. Hislop, and J. Diggs. Developing HFOSS projects using integrated teams across levels and institutions. In *Proc. of the ACM SIGITE Conf. on Information Technology Education (SIGITE)*, 139-140, 2013. http://doi.acm.org/10.1145/2512276.2512317
23. D. Burdge, L. Postner, H.J.C. Ellis, and G.W. Hislop. Preparing for student participation in HFOSS projects: FOSS tools and techniques. *J. Comput. Sci. Coll.* 29, 3, 74-75, 2014.
24. H.J.C. Ellis and G.W. Hislop. Structuring software engineering learning within open source software participation. In *Proc. of the Conf. on Innovation & Technology in Computer Science Education (ITiCSE)*, 326-326, 2014. http://doi.acm.org/10.1145/2591708.2602681
25. H.J.C. Ellis, S. Jackson, D. Burdge, L. Postner, G.W. Hislop, and J. Diggs. Learning within a professional environment: Shared ownership of an HFOSS project. In *Proc. of the ACM SIGITE Conf. on Information Technology Education (SIGITE)*, 95-100, 2014. http://doi.acm.org/10.1145/2656450.2656468
26. H.J.C. Ellis, S. Jackson, G.W. Hislop, L. Postner, and D. Burdge. Getting started in open source a tour of a real project. *J. Comput. Sci. Coll.* 29, 6, 12-14, 2014.
27. J.J. Farrell, R.S. Moog, J.N. Spencer. A guided-inquiry general chemistry course. *Journal of Chemical Education*, 76, 4, 570, 1999. doi:10.1021/ed076p570
28. S.E. Lewis and J.E. Lewis. Seeking effectiveness and equity in a large college chemistry course: An HLM investigation of Peer-Led Guided Inquiry. *Journal of Research in Science Teaching*, 45, 794-811, 2008.

29. H.H. Hu, and T.D. Shepherd. Using POGIL to help students learn to program. *ACM Trans. on Computing Education*, 13, 3, 13:1-13:23, 2013.
30. H.H. Hu, and T.D. Shepherd. Teaching CS1 with POGIL activities and roles. In *Proc. of the ACM Tech. Symp.on Computer Science Education (SIGCSE)* Atlanta, GA, March 5-8, 2014.
31. H.H. Hu, M. Lang, C. Kussmaul, C. Mayfield, and T. Pirmann. Guided inquiry learning in context: perspectives on POGIL in CS. In *Proc. of the ACM Tech. Symp. on Computer Science Education (SIGCSE)*. ACM, 2014. DOI=10.1145/2538862.2538869 http://doi.acm.org/10.1145/2538862.2538869
32. The Joint Task Force on Computing Curricula. *Software Engineering 2004: Curriculum Guidelines for Undergraduate Degree Programs in Software Engineering*. ACM and IEEE Computer Society, 2004.
33. C. Kussmaul. Experience Report: Guiding faculty & students to participate in Humanitarian FOSS communities. *IEEE Int'l Conf. on Tech. for Education (T4E)*, Mumbai, India, 2016.