Panel - Teaching Students to Participate in Open Source Software Projects

Heidi J. C. Ellis, Gregory W. Hislop, Mel Chua, Clif Kussmaul, Matthew M. Burke
hellis@wnec.edu, hislop@drexel.edu, mel@redhat.com, kussmaul@muhlenberg.edu, mmburke@gwu.edu

Abstract – This panel will present several experiences in involving students in Open Source Software (OSS) projects from the perspectives of both the instructor and a member of the OSS community. OSS is growing rapidly and gaining market share in both industry (e.g., Linux and Mozilla) as well as academia (e.g. Moodle, Greenfoot, and Drupal). OSS projects have a culture built on volunteer participation to support software development. Computing degree programs desire to involve students in large-scale software projects to provide students with real-world experience and an understanding of the issues found in large, complex software projects. Involving computing students in OSS projects serves both the OSS community by providing development resources for the project while also serving the academic community by providing access to large software projects in which students can gain experience. However, the marriage of student and OSS project presents some challenges including identification of approachable OSS projects, creation of appropriate educational infrastructure, evaluation and grading, and more. Panelists will address the factors that contribute to student success in an OSS project.

Index Terms – Open source software, real-world education, student software project.

GOAL

Open Source Software (OSS) has become mainstream and the number of OSS projects is growing rapidly. For example, sourceforge.net has 230,000 registered OSS projects and over 2,000,000 registered users. Since software engineering artifacts are frequently accessible online and collaborators and contributors to the project are easily identifiable, OSS provides a practical and useful way to supply students with experience in many aspects of software development. Quite simply, the range and volume of software projects accessible to students via OSS is unmatched by anything publicly available prior to the emergence of open source. Students can view and learn from existing artifacts and can join an ongoing project. Students can interact with experienced developers, experience a variety of development environments, and participate in professional communications. Many members of both the open source community and academia have realized the benefits of student involvement in OSS and have formed the Teaching Open Source community [1]. However, from an instructional standpoint, involving students in OSS projects presents some challenges:

• How do instructors identify suitable OSS projects?
• What are ways that students can participate in and contribute to OSS projects?
• How do instructors involve students in OSS projects?
• How do instructors evaluate student contributions to OSS projects?
• What resources are available for instructors looking to involve students in OSS projects?

This panel will report on several diverse experiences involving students and instructors in OSS projects. The panel includes educators and members of OSS projects in order to provide perspectives from both academia and OSS environments.

PARTICIPANTS AND TOPICS

The panel participants will each discuss their experiences, focusing on the challenges encountered and how such challenges were met.

Heidi Ellis is one of the founding members of the HFOSS (Humanitarian FOSS) project [2,3,4] which focuses on involving students in OSS projects that improve the human condition. Heidi is also PI on the SoftHum project [5] which develops course materials that support student involvement in HFOSS projects. Heidi will moderate the discussion as well as share her experiences on how she started engaging students in Sahana, a disaster relief application. Heidi will explain how the HFOSS effort got started and the steps taken during the startup of the HFOSS effort to introduce students into HFOSS projects.

Greg Hislop is principal investigator or Co-PI on several projects that focus on developing ways to involve students in OSS as project contributors [4,5,6]. He is also working on ways to have students participate as providers of IT support services for users of open source products. Greg will discuss the various ways that OSS has been introduced into the Drexel curriculum for the computing degrees in computer science, software engineering, information systems, and information technology. Greg will include ideas for taking initial steps to add OSS activities to a computing curriculum. He will also address learning curve issues for instructors including sources of assignments and other instructional materials. Finally, he will discuss the range of assignments and contributions that students can make beyond contributions of source code.
Mel Chua is the former QA Engineering community lead for One Laptop Per Child (OLPC), the current head of the Fedora Marketing team, a board member of Sugar Labs, and the coordinator of the Professors' Open Source Summer Experience (POSSE) bootcamp for faculty interested in getting their students involved as open source contributors [7]. Mel will discuss the science behind building strong technical communities and describe how to find and create on-ramp opportunities for new contributors to any OSS project, including students.

Clif Kussmaul has contributed to and provided consulting using OSS such as Drupal and TWiki. He has also supervised a variety of student projects that use or extend OSS, including Drupal, Moodle, and SubjectsPlus. Clif will discuss ways to introduce students to OSS and OSS communities, based on the evolution of roles within such communities. He recommends a USABL model [8], where students use, study, add to, build, and leverage OSS. Clif will also describe various ways that OSS can be helped students who do not have strong backgrounds in computer science or software engineering.

Matt Burke has served as a mentor and project administrator for the Google Summer of Code program. In addition to frequently using open source software in his courses, he has developed assignments that require his students to read and analyze open source code. Matt will discuss ways GWU includes OSS in the curriculum, challenges in having your students read code, and issues that arise in mentoring students in OSS participation.

INTENDED AUDIENCE
The audience for this panel will be instructors across a variety of computing disciplines including software engineering, computer science, and information technology who desire a better understanding of ways to involve students in OSS projects. The panel will also appeal to instructors who are interested in the impact of OSS on education.

LIST OF EQUIPMENT
The panel will require overhead projection equipment with laptop connectivity as well as internet access.

ACKNOWLEDGEMENTS
This material is based on work supported by the National Science Foundation under Grant No. 0958204. Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation (NSF).

REFERENCES

AUTHOR INFORMATION
Heidi J. C. Ellis, Associate Professor and Chair of Computer Science and Information Technology, Western New England College, hellis@wnec.edu
Gregory W. Hislop, Associate Professor of Information Science and Technology and Computer Science, Drexel University, Hislop@drexel.edu
Mel Chua, Community Engineer in Red Hat's Community Architecture team, mel@redhat.com
Clif Kussmaul, Associate Professor of Computer Science, Muhlenberg College, kussmaul@muhlenberg.edu
Matthew M. Burke, Assistant Professor of Computer Science, The George Washington University, mmburke@gwu.edu