Envisioning National and International Research on the Multidisciplinary Empirical Science of Free/Open Source Software

Walt Scacchi, University of California, Irvine; Kevin Crowston, Syracuse University; Greg Madey, Notre Dame University; Megan Squire, Elon University

Overview

We seek to establish and sustain an agenda for a national program for research on free/open source software (FOSS, or sometimes FLOSS) by academic and industrial researchers in different disciplines. This proposal describes our vision for such a research agenda, along with the international workshop and supporting meetings we propose to conduct in order to develop the agenda to guide future research. The activities build from recent research meetings on FOSS support multi-disciplinary studies of FOSS development. We also identify our goals, assessment method, activities, outcomes, and results from recent meetings giving rise to this proposal.

Why we need a national research program in Free/Open Source Software

Even though Free/Open Source Software (FOSS) is widely used, we believe the much of the Computer Science research community has yet to fully recognize its potential to change the world of research and development of software-intensive systems across disciplines. Tens of thousands of FOSS projects are up and running world-wide, and millions of end-users of computing increasingly rely on FOSS-based systems. Growing numbers of research projects in physical, social, and human sciences, as well as the cultural arts are now routinely expecting to develop or use FOSS-based systems to best meet their needs. Similarly, growing numbers of businesses and government organizations are now looking to develop and use mission-critical software applications that are built with FOSS components. We believe reasons for such attention and investments can be attributed to the following observations about FOSS.

-- *FOSS development is participatory and user friendly* – Compared to prior software development methodologies and approaches that emphasize technical system functionality (e.g., service-oriented architectures, object-orientation, computer-aided software engineering, structured programming), FOSS development is both socially convivial and technically engaging. FOSS developers are also end-users of the software they build, so the division between developers and end-users is reduced or eliminated. This in turn streamlines and simplifies difficult software development activities like requirements specification/analysis and testing.

-- *FOSS projects enable large-scale, domain-specific learning* – The most commonly cited reason for joining a FOSS project is to learn—learn new

skills, learn new domains, learn from domain experts, learn from participant observation, etc. [Scacchi 2007]. Also, large decentralized FOSS development projects like Google's Summer of Code (and also South Korea's Winter of Code) demonstrate new regimes for annually enabling hands-on participatory learning by thousands of students worldwide, independent of geographical location, national origin, or prior education, that facilitate informal software engineering and computer science education.

-- *Many key FOSS projects are U.S. led* – FOSS projects enable people from around the world to participate in software development projects of their own choice, to meet their own interests, and to facilitate technical skill development. The majority of project contributors are international (70% of FOSS developers are based in EU countries [Reding 2007]). However, many key FOSS projects like the Linux Kernel, Apache Web server, Mozilla/Firefox Web browsers, OpenOffice productivity suite, and Eclipse interactive development environment are led by core developers working in the U.S.

-- *Transforming research practices across disciplines* – FOSS development processes, work practices, and project community dynamics are being adopted and put to work in R&D projects in the physical and biological sciences, and various fields of engineering, and have also become the subject of research in the economic, legal, and social sciences. Research is now underway in such diverse subjects such as Economics (motivations for FOSS developers; industry competitiveness), Law (FOSS license regimes), Public Policy (impact on balance of trade, FOSS adoption by local governments), Art (open source and open media artworks), Anthropology (FOSS practices in non-Western cultures), Organization Science (end-user innovation, public-private innovation approaches), Business/Management (corporate adoption of FOSS, maintainability of FOSS), Geography (FOSS-based GIS), Biology (open source bioinformatics), Physics and E-Science (astronomical software, open source grid software), Information Systems (understanding teamwork in FOSS development, success factors in FOSS development). However, some of these efforts have suffered from nominal or weak understanding of FOSS systems and technologies, while some early Computer Science-based studies of FOSS slight/ignore the social and community aspects that are essential to sustained FOSS projects. Overall, it is clear that FOSS is a domain of Computer Science that is gaining the research attention of scholars in many scientific and cultural disciplines, as well as shaping their research agendas.

-- *Transforming the global software and IT industries* – Every major IT and software company worldwide (including Microsoft, IBM, Oracle, and SAP) is investing in FOSS development projects. Every major science research government ministry supporting software development is funding FOSS projects. Growing numbers of national and regional governments, military/

defense agencies, and ministries of education worldwide are establishing policies that encourage the development and deployment of FOSS computing systems.

-- *Transforming society and culture*-- A small but growing number of scientific and cultural/arts disciplines as well as new government organizations in emerging arenas for collective action are embracing the move to "openness". One can now find references to "open science," "open source art/architecture," or "edge organizations" which point to new work and institutional practices where openness, transparency, and peer production - all hallmarks of the open source development paradigm - within decentralized organizational forms are the norm. In science, the Public Library of Science (PloS.org) has emerged as a leading source for publication of scientific research results that follow the practice of open science [cf. David 2004]. The U.S. Department of Defense has begun to refocus its research in the development of command and control systems towards those that assume and operate within an edge organization with open architectures [Alberts and Hayes 2003, Starrett 2007, Weathersby 2007].

-- *Innovation* – Successful FOSS systems and communities can grow at sustained exponential rates through ongoing contributions that realize continuous improvement and evolutionary adaptation [Deshpande 2008, Koch 2005, Scacchi 2006]. FOSS has become an engine of innovation within the global software community, and is seen as a basis for enabling new opportunities to enter global software markets and challenge incumbent firms [Reding 2007].
A small but growing community of FOSS researchers in Computer Science and related disciplines are now actively engaged in a variety of empirical studies of FOSS development processes, work practices, and project community dynamics to help understand what works, when, where, why and how in FOSS projects of different kinds. We seek to develop a new vision and research agenda for the FOSS research community. Community members have individually addressed a number of interesting issues about the creation and use of FOSS, but there is not an articulated overall vision for the research, nor does the research systematically connect to national priorities. To support the development of a coherent agenda, we are requesting support to organize and conduct an international research workshop and related meetings (before and after the workshop) whose goal is to articulate and produce an agenda for funding, operating, contributing to, and sustaining a national FOSS research program based at the NSF.

The FOSS research community is growing across and within multiple disciplines including Computer Science, Software Engineering, Information Systems, Information Studies/Informatics, and others, as well as connecting to researchers in industrial research labs or large non-academic FOSS

projects. The community is of a manageable size, making it feasible to bring together leading researchers and to disseminate the vision across research groups.

What is our vision for the future of FOSS?

The development of FOSS is global socio-technical movement leading the way towards open science, open content, and open culture. But it is one of the few such movements, or perhaps the only one at present, that has Computer Science at its core. We believe that FOSS is a game-changing social innovation of historic proportions that is transforming how people work together to develop complex systems (and systems of systems). Many of the grand challenge topics for engineering research (cf. www.engineeringchallenges.org) increasingly rely on the development of FOSS systems (e.g., the International Thermonuclear Energy Research (ITER) project for fusion research), and in some topics, the development and experimentation with FOSS-based systems are central to research activities (e.g., advanced health informatics, secure cyberspace, enhanced virtual reality, and advanced personalized learning systems). Elsewhere, the Debian Gnu/Linux software distribution may be the largest software system ever created, constituting more than 400M source lines of code. The development of the core infrastructure to the World-Wide Web and Internet primarily rests on FOSS systems and concepts (e.g., TCP/IP stack, network application protocols (HTTP, FTP, SMTP, etc.), Web browsers, and Web servers). No corporation or government enterprise appears capable of now building and sustaining software systems of such size and complexity that can overcome what is achieved with FOSS.

Successful FOSS systems and their associated communities of developers and end-users demonstrate sustained exponential growth in more than 40% of the cases [Deshpande 2008, Koch 2005, Scacchi 2006]. Sustained exponential growth of a computing technology, e.g., computer processors and disk storage devices (e.g., following Moore's Law), eventually change our social worlds and our worlds of scientific inquiry and education, across all disciplines. They also transform the roles and goals of Computer Science as the core research discipline that drives this kind of socio-technical transformation.

However, despite these wide-ranging impacts, FOSS is not an explicit part of the research agenda for the National Science Foundation, nor any other U.S. research funding agency. On the other hand, the EU has a dedicated research program, as well as overarching policy direction encouraging the development of FOSS in all areas of research and development. We believe there is a critical need for a strategic investment in FOSS research that builds on, complements, and leverages other programmatic investments in the CISE disciplines within the U.S.

What is the role of computer science in this future?

As we have discussed above, FOSS is radically transforming how software is being developed by different communities in different disciplines. However, FOSS remains a computing technology at its core. As such, it is amenable to both technological advances and socio-technical innovations that can emerge from research in the CS community. But so far, most of the advances in the development and practice of FOSS do not directly emerge from academic Computer Science programs, or do so but in ways where the legacy of originating concepts/advances is lost or obscured.

What are key research questions for FOSS research

- How does FOSS as a diverse socio-technical movement accomplish global software development, without a traditional central authority or source of funding/resources?
- How do distributed groups make decisions? What sort of conflicts are common, and how are conflicts settled?
- What are the differences and similarities between FOSS projects and proprietary (non-FOSS) projects? Is there a taxonomy of characteristics of these two types of projects? Are there hybrid projects, and how are these described?
- How do we measure "success" of a FOSS project? What are the various attributes of a project that might help us measure success? Do we have all the data we need, or are there additional measures that we need to collect?
- What are the different ways that software developers (makers of the technology) are incentivized ("paid") within the various types of FOSS projects? How does this incentive structure compare to proprietary projects? What do the developers themselves report are the best and worst incentives?
- How can the benefits of FOSS be translated into a language technology decision-makers can understand? Are there "best practices" for FOSS technology adoption or for rollovers from proprietary to FOSS models within businesses or governments?
- What are the various techniques and technologies that help self-organized groups to work effectively? How can these self-organizing techniques and technologies be applied to other domains?
- What are the different roles in a FOSS project (e.g., core developer, active user)? What levels of contribution is needed from members in various roles are needed to sustain a project (e.g., how important are active users)?
- How long can such a movement be sustained?
- Are there conditions or events that constitute an inflection point that will mark the decline of FOSS as a socio-technical movement?

We need to articulate both multi-disciplinary and inter-disciplinary perspectives on how and why FOSS has become such a source of technology-centered global transformation, and what the future may hold. We need to identify key research problems and experimental studies. We also need to identify future roles that Computer Science can play in fostering, sustaining, and expanding the ongoing development of FOSS as a realm of technology development and use, as an engine of innovation in other scientific and cultural disciplines, and as a socio-technical movement that has Computer Science at its core. Such transformational capabilities arising from advances in computer science, like personal computers and the world-wide web.

As such, our objective is to bring together a diverse audience of computer scientists whose research activities focus on the development, use, and evolution of FOSS systems, tools, techniques, and concepts in a way that can most effectively articulate a new research agenda that can become the basis for a new cross-disciplinary program on FOSS at NSF.

Assessment

We employ multiple criteria for assessing our effort. First, by engaging in recurring workshops and meetings of the researchers actively engaged in studies of FOSS, we further build and sustain this research community and the agenda of topics of interest to the community. Without successful meetings and community development, the pace of U.S. based FOSS research will lag. Second, much like the FOSS projects we study, our proposed effort must lead to the emergence of a shared, global research infrastructure that itself is open to access, study, modification, and redistribution by research community members. Without such openness, scientific knowledge of value to both academic and industrial audiences will be limited. Last, the community of research participants in the proposed effort is international in scope, spanning multiple disciplines, and both industry and academic institutions, such that any results, reports, papers, presentations, or online resources (community Web sites) are publicly available and accessible. Without engagement of the U.S., European, and Asian FOSS research communities and research data collections, then U.S. based FOSS researchers will be at a disadvantage in advancing scientific knowledge of FOSS development practices and consequences, compared to the sizable funding advantages into FOSS-based research now in place within the European Community.

Activities and outcomes

We adopt the CCC visioning strategy as the basis for specifying the activities and outcomes we seek to perform. First, through recent meetings of the FOSS research community focused on FOSS research data and data repositories, we have found there is widespread interest in moving toward a

common, open, and federated environment for sharing FOSS research data, analyses (including data provenance), models, simulations, and publications. In this regard, we have established a basis for *nucleating* a diffuse set of interests into an emerging common vision for future research. Second, through this proposal, we seek to conduct a set of meetings and workshop that will *crystalize* the community vision as well as *broaden* the research agenda and audience we seek to engage. Third, resulting from these two activities will be a collection of deliverable outcomes including documents specifying the research program and recommended agenda for action, meeting and Workshop reports, and Workshop Web site where participant contributions (participant research biography and interests, Workshop presentations, group wikis, community blog, and related online research publications) will be hosted. These deliverables embody our *formulation* of our research program going forward. Last, we seek to actively engage program managers from research funding agencies like the National Science Foundation in order to initiate discussions that can precipitate actions (e.g., invited presentation or working group meetings at NSF) that can move our proposed research program and agenda into new/existing research programs and solicitations, and thus represent the beginning of the *execution* of our research program.

Recent meetings on FOSS Research

There have been four workshops focused on addressing topics and issues of FOSS, mostly focusing on data repositories or infrastructures starting in Spring 2006, but two have been held outside of the U.S. This has limited the participation of FOSS researchers working in the U.S. The first such workshop, *Workshop on Public Data about Software Development*, was held in Como, Italy in conjunction with the 2nd International Conference on Open Source Systems (10 June 2006). Similarly, the *2nd Workshop on Public Data about Software Development*, was held in Limerick, Ireland in conjunction with the 3rd International Conference on Open Source Systems (14 June 2007). Finally, there was also the *First International Workshop on Emerging Trends in FLOSS Research and Development*, held in conjunction with the 29th International Conference on Software Engineering, which was held in Minneapolis, MN (21 May 2007). It was at this meeting that some of the FOSS researchers began in-depth discussion of the problems and challenges of developing FOSS data repositories and what advantages might be realized if these emerging FOSS research infrastructures could become more transparent and interoperable. This discussion continued in earnest at the *2nd Workshop on Public Data about Software Development,* which resulted in both the U.S. FOSS researchers and their counterparts in Europe to agree to begin the effort for moving to common national and international information infrastructures for FOSS research. Finally, in February 2008, a small, two day NSF funded workshop was held at UC Irvine with 20 FOSS researchers based in the U.S. (no funds were available to support

international participants) focused on the subject of FOSS repositories and research infrastructures. This workshop helped to identify major research accomplishments as well as to begin to establish the community of researchers whose future research into FOSS critically depends on access to various FOSS development data sets and multi-project repositories. The record and results from this Workshop in the form of participant wiki and blog contribution, hyperlinked presentations, and online research publications can be found at the Workshop's Web site, fossrri.rotterdam.ics.uci.edu.

At this 2008 Workshop, participants engaged in a close review and critical discussion of three large FOSS multi-project data repositories (FLOSSmole, SourceForge Database at Notre Dame University, and Google Code Project Hosting). Both FLOSSmole and the Notre Dame SF Database already have dozens of external users who repeatedly access, query, and download FOSS data from these repositories, and who often engage in quantitative, statistical, or social network analysis of the data selected. Participants from the Workshop also reviewed the practices and needs of researchers who focus primarily on qualitative and other forms of data analysis (including data mining and knowledge discovery) to better understand or explain FOSS development processes, work practices, and project community dynamics. Participants also discussed the advantages and disadvantages of whether it would be desirable to the research community (including the people at the Workshop) to have a commercial service like Google to setup and operate a "research data service" that would be focused to needs of the FOSS research community. Aspects of the review and discussion of these topics were captured by the Workshop participants through a content management system that was setup and structured to support this capability. This in turn enabled the participants to articulate their views, issues, and concerns in ways that were open to review by others, as well as providing support for reflection and subsequent revision of earlier contributions to the scholarly debates at hand.

Workshop findings and observations informing our proposed visioning project

Empirical studies of FOSS development (FOSSD) are expanding the scope of what we can observe, discover, analyze, or learn about how large software systems can be or have been developed. In addition to traditional methods used to investigate FOSS like reflective practice, industry polls, survey research [Hertel 2003], and ethnographic studies, comparatively new techniques for mining software repositories [Howison 2007, Garg 2004, Gasser 2004, Robles 2004] and multi-modal modeling and analysis of the socio-technical processes and networks found in sustained FOSSD projects [Scacchi, *et al.* 2006, Scacchi 2007] show that the empirical study of FOSSD is growing and expanding. This in turn will contribute to and help advance

the empirical computer science in fields like Software Engineering, which previously were limited by restricted access to data characterizing large, proprietary software development projects. Additionally, such studies will help inform FOSSD projects in other scientific and cultural disciplines, and thus highlight the contribution of computer science research and education to those disciplines. Subsequently, empirical studies of software products, processes, projects/organizations will increasingly rely on data collected from FOSS development projects. Thus, the future of empirical studies of software development practices, processes, and projects will increasingly be cast as studies of FOSSD efforts.

The diversity and population of FOSS projects and multi-project repositories is unclear and unknown. There is great interest in the research community for a baseline and ongoing census of FOSS multi-project repositories. As FOSS projects choose to collect, organize, and share the raw data of software development as an activity in their self-interest, then it behooves us within the research community to offer some guidance or incentives for these independent FOSS projects to contribute to such a census. Similarly, we need to articulate what benefits (e.g., socio-economic impacts or intellectual contributions) the research community might offer in return to the FOSS projects that contribute to such a census.

-- Data vary in *content*, with types such as communications (threaded discussions, chats, digests, Web pages, Wikis/Blogs), documentation (user and developer documentation, HOWTO tutorials, FAQs), development data (source code, bug reports, design documents, attributed file directory structures, CVS check-in logs) [Scacchi 2002, 2007], and programming languages [Delorey 2007].

-- Data originates from different *types of repository* sources [Deshpande 2008, Hahsler 2005, Howison 2006, Gao 2007, Mockus 2002] . These include shared file systems, communication systems, version control systems, issue tracking systems, content management systems, multi-project FOSS portals (SourceForge.net, Freshmeat.net, Savannah.org, Advogato.org, Tigris.org, etc.), collaborative development or project management environments, FOSS code indexes or link servers (free-soft.org, LinuxLinks.com), search engines (Google.com/code, krugle.org), and others. Each type and instance of such a data repository may differ in the storage data model (relational, object-oriented, hierarchical, network), application data model (data definition schemas), data formats, data type semantics, and conflicts in data model namespaces (due to synonyms and homonyms), modeled, or derived data dependencies. Consequently, data from FOSS repositories is typically heterogeneous and difficult to integrate beyond semantic hypertext linking [Noll 1991], rather than homogeneous and comparatively easy to integrate.

-- Data can be found from various spatial and temporal *locations*, such as community Web sites, software repositories and indexes, and individual FOSS project Web sites. Data may also be located within secondary sources appearing in research papers or paper collections (e.g., MIT FOSS research paper repository at opensource.mit.edu), where researchers have published some form of their data set within a publication [Mockus 2002, Scacchi, *et al.* 2006, Wasserman 2007].

-- Different *types of data extraction tools and interfaces* (query languages, application program interfaces, Open Data Base Connectors, command shells, embedded scripting languages, or object request brokers) are needed to select, extract, categorize, and other activities that mine, gather, and prepare data from one or more sources for further analysis [Garg 2004, German 2003, Jensen 2006, Kawaguchi 2003, Ripoche 2003, Robles 2004], as well as providing new kinds of tools and techniques for visualizing evolving software systems and the social networks that develop them [De Souza 2007, Ogawa 2007a,b].

-- Most FOSS project data is available as *artifacts or byproducts* of development, usage, or maintenance activities in FOSS communities. These artifacts/byproducts are a critical part of the FOSS innovation process [West 2006]. However, very little data is directly available in forms specifically intended for research use. This artifact/byproduct origin has several implications for the needs expressed above [Gasser 2004, Robles 2006, Scacchi 2002, 2007].

The open and public accessibility of data from FOSS project repositories and multi-project repositories (e.g., SourceForge.net, FLOSSmole, Google Code [cf. Howison 2007, Gao 2007, Garg 2004, Gasser 2004, Robles 2004]) is providing a new, empirically grounded view of software technology and software development practice—a view that enables comparative, cross-sectional, and ecosystem level studies. This in turn means news kinds of research questions can be posed and new knowledge can be discovered, derived, or created. For example, repository-based studies of successful FOSS projects (of which there are now at least a few thousand such projects) indicate that their software code base, functionality, development artifacts, and developer contributions, and user base can undergo sustained exponential growth, apparently in contradiction to long-standing "laws of software evolution" which traditionally predict sub-linear, inverse square growth rates [cf. Capiluppi 2004, Desphande 2008, Koch 2005, Scacchi 2006]. As such, the kind of research questions that follow may ask what model or theory accounts for the super-linear evolution of FOSS systems? Another example: are there software patterns that constitute a kind of "software genome" that characterize the evolutionary mechanisms of

different families of independently developed FOSS systems? Similarly, are the critical software components or functions that lie at the heart of different software families, and does such software represent a critical evolutionary or security vulnerability (e.g., the `glibc` library is commonly bound with FOSS coded in the C programming language)? Also, what development processes best characterize FOSS projects that demonstrate sustained exponential growth of their code and functionality base, as well as the growth of the number of contributors, but with comparable growth/decline of software quality? Last, what can we learn about the evolution of FOSS systems across multiple releases, across multiple platforms, and across different independently developed variants? Exploring any questions like these all require data drawn from multiple FOSS projects or project repositories, and this data is now available. As such, we are on the verge of possible discontinuous advances in our knowledge about software, based on empirical studies of FOSS.

Articulating new knowledge of software products, processes, practices, and organizational forms depends in part on careful and systematic empirical study of FOSS project data. However, this data is not trivial to collect, use, or analyze. As such, there is need to articulate practices for the curation of FOSS project data in forms that increase the likelihood for the data use, reuse, and (re)analysis by people in different disciplines and settings. There is also need to help capture data provenance as well as data annotation and data analysis workflow tools & techniques. Other science disciplines have recognized similar needs, so there is an opportunity for current investments in such areas to be structured to both discipline-specific and cross-discipline research efforts. At present, the FOSS research community has little practice and access to these tools and techniques, and as a result, has little incentive to take on their application or reinvention.

The commercial software products and service industry in the U.S. is in an awkward strategic position regarding whether or how to take advantage of FOSS, or the results arising from studies of FOSS development data. Software product companies like Microsoft seem hesitant about what to do about FOSS, while software service companies like Google seem to embrace FOSS (as do computer vendors like IBM and SUN). But it appears that all software companies can benefit from empirical studies of FOSS products, processes, practices, and organizational forms that are comparative or cross-sectional, for different competitive reasons. Last, companies like Google, SUN, IBM, and Microsoft Research have established a community of FOSS development projects under their corporate sponsorship. These projects are sponsored as a way for these companies to help increase the pool of future software developers who might then transition into the commercial software workforce. These projects also serve to provide a situated, real-world experiment in informal software engineering education, that often takes place outside of the traditional higher education

environment. However, "data" from these informal educational experiences is generally not open, nor publicly available, as it is sometimes said to be sensitive, confidential, and proprietary. Thus it is unclear how well these informal experiments work, or whether/how they can be improved both from a corporate perspective as well as from an academic perspective. Perhaps there is an opportunity to bring together the academic software research and software engineering education community together with the commercial software industry through a government sponsored or coordinated forum so as to articulate how to best advance U.S. socio-economic and scholarly interests for mutual benefit and growth of the software community.

Workshop organizing committee biographies

Members of the Workshop organizing committee are Walt Scacchi, Institute for Software Research, University of California, Irvine; Megan Squire, Computer Science Department, Elon University; Kevin Crowston, School of Information, Syracuse University; and Greg Madey, Computer Science and Engineering Department, Notre Dame University.

-- *Walt Scacchi* (www.ics.uci.edu/~wscacchi/) is senior research scientist and research faculty member at the Institute for Software Research, UC Irvine. He received a Ph.D. in Information and Computer Science from UC Irvine in 1981. From 1981-1998, he was on the faculty at the University of Southern California. In 1999, he joined the Institute for Software Research at UC Irvine. He has published more than 150 research papers, and has directed 50 externally funded research projects. In 2007, he served as General Chair of the 3rd IFIP International Conference on Open Source Systems (OSS2007), Limerick, IE, in 2009, served as Program Chair for the OSS 2009 Doctoral Consortium, Skovde, Sweden, and in 2010, serves as Co-Chair for the OSS 2010 Doctoral Consortium, South Bend, Indiana. He is supported in part on this effort through NSF grant #0808783. No review, approval, nor endorsement implied.

-- *Kevin Crowston* (crowston.syr.edu/) is a Professor of Information Studies at the Syracuse University School of Information Studies. Prior to moving to Syracuse, he taught for five years at the University of Michigan Business School. He received his A.B. (1984) in Applied Mathematics (Computer Science) from Harvard University and a Ph.D. (1991) in Information Technologies from the Sloan School of Management, MIT.

-- *Greg Madey* (www.nd.edu/~gmadey/) is Associate Professor of Computer Science and Engineering at Notre Dame University. He has a Ph.D. in Operations Research from Case Western Reserve University. He has worked for several aerospace firms (today part of Lockheed-Martin and Northrup-

Grumman) in R&D advanced projects and strategy. He has also served as faculty member in Information Systems within a college of business.

-- *Megan Squire* (facstaff.elon.edu/msquire/) is an Associate Professor in the Department of Computing Sciences at Elon University. Her primary research focus is on data mining and large database systems, particularly for software engineering data. She was co-organizer of the 2006-2008 Workshops on Public Data about Software Development (along with Gregorio Robles and Jesus Gonzalez-Barahona). She has published a number of papers on tools for analyzing open source projects, and has spoken about open source data collection at such diverse events as the Mining Software Repositories workshop at ICSE and the O'Reilly Open Source Convention. She has a PhD in computer science from Nova Southeastern University.

In addition to their respective research and teaching accomplishments, Scacchi has served as PI on four NSF funded and three DoD funded projects focused on free/open source software development; Squire is PI on the FLOSSmole repository project [Howison, *et al.* 2007] (funded in part by NSF); Crowston is PI on FLOSSmole project and two other NSF-funded projects focused on open source software; and Madey is PI on the SourceForge Metadata Database project [Gao, *et al*. 2007] (funded in part by NSF). Others FOSS researchers identified above will be invited to serve on the Workshop Organizing Committee as part of this effort.

References

- Alberts, D.S. and Hayes, R.E., *Power to the Edge: Command and Control in the Information Age*, CCRP Publications, 2003. http://www.dodccrp.org
- Capiluppi, A., Morisio, M., and Lago, P., Evolution of Understandability in OSS Projects, *Proc. Eighth European Conf. Software Maintenance and Reengineering* (CSMR'04), 2004.
- David, P.A. Understanding the emergence of 'open science' institutions: Functionalist economics in historical context, *Industrial and Corporate Change,* 13(4), 571–589, 2004.
- Delorey, D., Knutson, C., Chun, S., Do Programming Languages Affect Productivity? A Case Study Using Data from Open Source Projects, *Proc. First Intern. Workshop on Emerging Trends in FLOSS Research and Development*, Minneapolis, MN, May 2007.
- Deshpande, A. and Riehle, D., The Total Growth of Open Source, *Proc. Fourth IFIP International Conference on Open Source Systems (OSS2008)*, Milan, IT (to appear, September 2008).
- De Souza, C.R.B., Quirk, S., Trainer, E., and Redmiles, D.F., Supporting Collaborative Software Development through the Visualization of Socio-Technical Dependencies. In *Proc. 2007 Intern.*

ACM Conference on Supporting Group Work. ACM Press, Sanibel Is, FL, 147-156, 2007.

- Dietz, T., Ostrom, E., and Stern, P.C., The Struggle to Govern the Commons, *Science*, 302, 1907-1912, Dec. 2003.
- Edwards, P., Jackson, S., Bowker, G., and Knobel, C., *Understanding Infrastructure: Dynamics, Tensions, and Design*, Report on NSF Workshop on "History & Theory of Infrastructure: Lessons for New Scientific Cyberinfrastructures", January 2007.
- English, R. and Schweik, C., Identifying Success and Tragedy of FLOSS Commons: A Preliminary Classification of SourceForge.net Projects, *Proc. First Intern. Workshop on Emerging Trends in FLOSS Research and Development*, Minneapolis, MN, May 2007.
- Gao, Y., Van Antwerp, M., Christley, S., and Madey, G., A Research Collaboratory for Open Source Software Research, *Proc. First Intern. Workshop on Emerging Trends in FLOSS Research and Development*, Minneapolis, MN, May 2007.
- Garg, P.J., Gschwind, T., and Inoue, K., Multi-Project Software Engineering: An Example *Proc. Intern. Workshop on Mining Software Repositories*, Edinburgh, Scotland, May 2004.
- Gasser, L., Ripoche, G. and Sandusky, R., Research Infrastructure for Empirical Science of F/OSS, *Proc. Intern. Workshop on Mining Software Repositories*, Edinburgh, Scotland, May 2004.
- Gasser, L. and Scacchi, W., Towards a Global Research Infrastructure for Multidisciplinary Study of Free/Open Source Software Development, *Proc. Fourth IFIP International Conference on Open Source Systems (OSS2008)*, Milan, IT (to appear, September 2008).
- German, D. and Mockus, A., Automating the Measurement of Open Source Projects. In *Proc. 3rd. Workshop Open Source Software Engineering*, Portland, OR, 63-68, May 2003.
- Hahsler, M. and Koch, S., Discussion of a Large-Scale Open Source Data Collection Methodology, *Proc. 38th Hawaii Intern. Conf. Systems Sciences*, Kailua-Kona, HI, Jan 2005.
- Hertel, G., Neidner, S., and Hermann, S., Motivation of software developers in Open Source projects: an Internet-based survey of contributors to the Linux kernel, *Research Policy*, 32(7), 1159-1177, July 2003.
- Howison, J., Conklin, M., & Crowston, K. (2006). FLOSSmole: A collaborative repository for FLOSS research data and analyses. *Intern. J. Information Technology and Web Engineering*, 1(3), 17–26.
- Jensen, C. and Scacchi, W., Experiences in Discovering, Modeling, and Reenacting Open Source Software Development Processes, in M. Li, B.E. Boehm, and L. Osterweil (Eds.), *Unifying the Software Process Spectrum: Proc. Software Process Workshop*, Beijing, China, 442-469, Springer-Verlag, 2006.

- Kawaguchi, S., Garg, P.K.,Matsushita, M., and Inoue, K., On Automatic Categorization of Open Source Software, in *Proc. 3rd. Workshop Open Source Software Engineering*, Portland, OR, 63-68, May 2003.
- Koch, S. Evolution of Open Source Software Systems—A Large-Scale Investigation, in *Proc. 1st Intern, Conf. Open Source Systems (OSS2005)*, Genoa, Italy, 2005.
- Mockus, A., Fielding, R.T., and Herbsleb, J., Two case studies of open source software development: Apache and Mozilla. *ACM Transactions on Software Engineering and Methodology*, 11(3):1-38, July 2002.
- Noll, J. and Scacchi, W., Integrating Diverse Information Repositories: A Distributed Hypertext Approach, *Computer*, 24(12), 38-45, December 1991.
- Ogawa, M. and Ma, K.-L., StarGate: A Unified, Interactive Visualization of Software Projects, *Proc. IEEE PacificVis 2008,* March, 2008, 191-198.
- Ogawa, M., Ma, K.-L., Devanbu, P., Bird, C., and Gourley, A., Visualizing social interaction in open source software projects, in *APVIS'07: Proc. 2007 Asia-Pacific Symp. on Visualization*, 25-32, IEEE Computer Society, 2007.
- O'Mahony, S., Guarding the Commons: How community managed software projects protect their work, *Research Policy*, 32(7), 1179-1198, July 2003.
- Reding, V., *Truffle 100: Towards a European Software Strategy*, European Commission on Information Society and Media, Brussels, http://ec.europa.eu/commission_barroso/reding/docs/speeches/brussels_20071119.pdf , 19 November 2007.
- Ripoche, G. and Gasser, L., Scalable automatic extraction of process models for understanding F/OSS bug repair. In *Proc.Intern.Conf. Software & Systems Engineering and their Applications (CSSEA'03)*, Paris, France, Dec. 2003.
- Robles, G., Gonzalez-Barahona, J.M., Ghosh, R., GluTheos: Automating the Retrieval and Analysis of Data from Publicly Available Software Repositories, *Proc. Intern. Workshop on Mining Software Repositories*, Edinburgh, Scotland, May 2004.
- Robles, G., Gonzalez-Barahona, J.M., Merelo, J., Beyond source code: the importance of other artifacts in software development, *J. Systems and Software*, 79(9), 1233-1248, 2006.
- Scacchi, W., Understanding the Requirements for Developing Open Source Software Systems, *IEE Proceedings--Software*, 149(1), 24-39, February 2002.
- Scacchi, W., Understanding Free/Open Source Software Evolution, in N.H. Madhavji, J.F. Ramil and D. Perry (Eds.), *Software Evolution and*

Feedback: Theory and Practice, 181-206, Wiley, New York, 2006. Original manuscript available on Web, April 2004.

- Scacchi, W., Free/Open Source Software Development: Recent Research Results and Emerging Opportunities, Invited Address, *Proc. European Software Engineering Conf. and ACM SIGSOFT Symp. Foundations of Software Engineering*, Dubrovnik, Croatia, 459-468, September 2007.
- Scacchi, W. and Alspaugh, T., Emerging Issues in the Acquisition of Open Source Software within the U.S. Department of Defense, *Proc. 5th Annual Acquisition Research Symposium*, Vol. 1, 230-244, NPS-AM-08-036, Naval Postgraduate School, Monterey, CA.
- Scacchi, W., Jensen, C., Noll, J. and Elliott, M., Multi-Modal Modeling, Analysis and Validation of Open Source Software Development Processes, *Intern. J. Information Technology and Web Engineering*, 1(3), 49-63, 2006.
- Star, S.L. and Ruhleder, K., Steps Toward an Ecology of Infrastructure: Design and access for large information spaces. *Information Systems Research*, 7(1), 111-134, 1996.
- Starrett, E., Software Acquisition in the Army, *Crosstalk: The Journal of Defense Software Engineering*, 4-8, May 2007, http://stsc.hill.af.mil/crosstalk.
- Wasserman, A. and Capra, E., Evaluating Software Engineering Processes in Commercial and Community Open Source Projects, *Proc. First Intern. Workshop on Emerging Trends in FLOSS Research and Development*, Minneapolis, MN, May 2007.
- Weathersby, J.M., Open Source Software and the Long Road to Sustainability within the U.S. DoD IT System, *The DoD Software Tech News*, 10(2), 20-23, June 2007.
- West, J. and Gallagher, S., Patterns of Open Innovation in Open Source Software, Chapter 5 in H. Chesbrough, W. Vanhaverbeke and J. West, (Eds.), *Open Innovation: Researching a New Paradigm*, Oxford University Press, 2006.