Collaborative Research:CRI:CRD: Data and Analysis Archive for Research on Free and Open Source Software and Its Development

Project Summary

This project will develop a CISE research community resource in the form of a broadly-shared data and analysis archive to further research on Free/Libre Open Source Software (FLOSS) and its development. The goal of the infrastructure is to improve the reproducibility and consistency of this research and to expand access to the data and thus the community, with a secondary goal of providing an educational opportunity for undergraduate computer science students. Specifically, we propose a distributed, collaborative community resource called FLOSSmole, to collect, organize and share comparable data and analyses of FLOSS development. FLOSSmole is designed to be a piece of research infrastructure: it is a framework for organizing and a system for facilitating access to the massive amounts of data collected by many simultaneous and currently unconnected FLOSS research efforts.

Expected intellectual merits
This project will contribute significantly to advancing knowledge and understanding within the FLOSS research community by enabling cooperation in data collection, aggregation and sharing, thus providing synergies to on-going and newly developed projects. The FLOSS research community is dedicated to understanding how FLOSS projects are developed and managed, how the software develops, and how it is used. As FLOSS projects become more ubiquitous, quality data to describe and explain their successes becomes more important. Furthermore, research on FLOSS and its development processes can teach us about other areas of interest across the computing fields (Harrison, 2001), such as software evolution. As well, FLOSS development teams are potential training grounds for future software developers, making it important to understand how developers join and work in these teams. Finally, because FLOSS development provides numerous examples of successful computer-supported collaborative work, our project is relevant to researchers in CISE areas such as Human-Centered Computing and NSF-wide initiatives such as Cyberinfrastructure.

Expected broader impact
This project will benefit society by promoting international collaboration and data sharing among research teams dedicated to understanding how FLOSS projects are developed, managed and sustained. FLOSS is integral to today's internet and is a foundation of tomorrow's innovation, thus this project will support the improvement of an important piece of collaborative research infrastructure used by academics, by practitioners in the software industry, and by society in general. The project will also have beneficial impacts for the undergraduate and graduate students who work on it, positively impacting the quality of courses and projects. Finally, as an open source project itself, our community resource continues an important trend in scientific research toward opening and sharing data in order to promote collaboration, to reduce duplicative efforts, and to promote compatibility between research teams. Sharing code, data, schemas, queries, and experience promotes teaching and learning within the community.

Collaborative Research:CRI:CRD: Data and Analysis Archive for Research on Free and Open Source Software and Its Development

1. Project Overview

This project will develop a CISE research community resource in the form of a broadly-shared data and analysis archive to further research on Free/Libre Open Source Software (FLOSS) development. The goal of the infrastructure is to improve the reproducibility and consistency of this research and to expand access to the data and thus the community, with a secondary goal of providing an educational opportunity for undergraduate computer science students. Specifically, we propose a distributed, collaborative community resource called FLOSSmole, designed to collect, share, and store comparable data and analyses of open source software and its development. FLOSSmole is designed to be a piece of research infrastructure; it is a framework for organizing and sharing the massive amounts of data collected by many simultaneous and currently unconnected FLOSS research efforts.

1.1 Research and education activities to be enabled

The proposed community infrastructure development project will enable a new wave of research on the development and use of Free/Libre Open Source Software (FLOSS). FLOSS is a broad term used to embrace software developed and released under an "open source" license. Such licenses allow inspection, modification and redistribution of the software source code without charge ("free as in beer"). Much (though not all) of this software is also "free software", meaning that derivative works must be made available under the same unrestrictive license terms ("free as in speech", thus "libre"). (We have chosen to use the acronym FLOSS rather than the more common OSS to emphasize this dual meaning.)

FLOSS is an important topic of research for CISE and thus deserving of infrastructure support for several reasons. First, FLOSS development is an important topic in and of itself because FLOSS underlies much of today's computing infrastructure. There are thousands of FLOSS projects, spanning a wide range of applications. Due to their size, success and influence, the Linux operating system and the Apache Web Server (and related projects) are the most well known, but hundreds of others are in widespread use, including projects on Internet infrastructure (e.g., sendmail, bind), user applications (e.g., Mozilla, OpenOffice) and programming languages (e.g., Perl, Python, gcc and most recently, Java). Many are popular (indeed, some dominate their market segment) and the code has been found to be generally of good quality (Stamelos et al., 2002). FLOSS is an increasingly important commercial phenomenon involving all kinds of software development firms, large, small and startup. Millions of users depend on systems such as Linux and the Internet (heavily dependent on FLOSS tools), but as Scacchi (2002, p. 1) notes, "little is known about how people in these communities coordinate software development across different settings, or about what software processes, work practices, and organizational contexts are necessary to their success". The practical importance of this phenomenon cries out for more in-depth research and better infrastructure to support this research.

Furthermore, research on FLOSS and its development processes can teach us about other areas of interest across the computing fields (Harrison, 2001). For example, the accessibility of FLOSS source code enables a new range of empirical studies of software engineering, such as examples of large-scale software evolution. The practice of software development in general has the potential to be instrumented in ways that were not previously possible. The history of science shows that new instrumentation often drives new discoveries, but we first need decent and appropriate instrumentation. This project will contribute to this development.

Understanding FLOSS development teams is also important because they are potentially training grounds for future software developers. As Arent and Nørbjerg (2000, p. 8) note, in these teams, "developers collectively acquire and develop new skills and experiences", thus providing students an opportunity to participate in more realistic software development projects. Again, the potential value of these experiences demands research on how developers join and work in FLOSS teams to illuminate what and how students can learn in these settings.

Finally, research on FLOSS development is relevant to CISE areas such as Human-Centered Computing and NSF-wide initiatives such as Cyberinfrastructure because FLOSS development provides numerous examples of successful computer-supported collaborative work across organizational boundaries. A 2002 EU/NSF workshop on priorities for FLOSS research identified the need both for learning "from open source modes of organization and production that could perhaps be applied to other areas" and for "a concerted effort on open source in itself, for itself" (Ghosh, 2002). The bulk of FLOSS development is carried out at a distance over information technologies, highly successful FLOSS teams provide us with evidence of how best to design and use cyberinfrastructure environments to support distributed engineering and, in combination with research on scientific collaboratories, distributed science (Crowston et al., 2006a). As research increasingly turns to the collaboratory model of science (Finholt & Olson, 1997), our FLOSS project will provide valuable lessons for creation of a distributed scientific collaboratory and data archive.

In addition to the intellectual merits of this proposal, the design of the project will also enable unique experience and education for undergraduate computing students, particularly in the areas of cyber-infrastructure and collaborative computing, thus contributing to NSF's educational goals.

1.2 How FLOSS research can be improved by a data and analysis archive infrastructure

In the preceding section, we have argued that research on FLOSS development practices has the potential to benefit CISE research in several ways. However, to gain these benefits requires better support for research on FLOSS development. To motivate the proposed infrastructure to be developed under this grant, we first briefly describe recent research on FLOSS and discuss where infrastructure support will be helpful.

Recent research on FLOSS. The apparent success of a few highly-visible FLOSS projects has lead to myriad claims of the superiority of this development approach over traditional methods in software engineering. The popularity of FLOSS has been attributed to the speed of development and the reliability, portability, and scalability of the resulting software as well as the low cost (Crowston & Scozzi, 2002; Hallen et al., 1999; Leibovitch, 1999; Pfaff, 1998; Prasad, n.d.; Valloppillil, 1998; Valloppillil & Cohen, 1998). In turn, the quality of the software and speed of development have been attributed to two factors: that developers are also users of the software and the availability of source code. First, FLOSS projects often originate from a personal need (Moody, 2001; Vixie, 1999), which attracts the attention of other users and inspire them to contribute to the project. Since developers are also users of the software, they understand the system requirements in a deep way, eliminating the ambiguity that often characterizes the traditional software development process: programmers know their own needs (Kraut & Streeter, 1995). (Of course, over-reliance on this mode of requirements gathering may also limit the applicability of the FLOSS model.) Second, in FLOSS projects, the source code is open to modification, enabling users to become co-developers by developing fixes or enhancements. As a result, FLOSS bugs can be fixed and features evolved quickly.

The implications of these claims are far-reaching—exciting if true, costly and foolish if false—yet the evidence remains frustratingly anecdotal and circumstantial. The NSF has recognized, through its research funding described below in section 3.2, that such claims deserve to be verified, tested and their application and limits explored. Such research is bearing fruit, but is hampered by the

difficulties and inconsistency of data collection and availability as well as lack of clarity in academic workflows. This proposal will enhance the synergy of existing and future NSF-funded projects and provide a boost for the quality and pace of research on FLOSS and its development.

The nascent research literature on FLOSS has addressed a variety of questions. First, researchers have examined the implications of FLOSS from economic and policy perspectives. For example, some authors have examined the implications of free software for commercial software companies or the implications of intellectual property laws for FLOSS (e.g., Di Bona et al., 1999; Kogut & Metiu, 2001; Lerner & Tirole, 2001).

Second, various explanations have been proposed for the decision by individuals to contribute to projects without pay (e.g., Bessen, 2002; Franck & Jungwirth, 2002; Hann et al., 2002; Hertel et al., 2003; Markus et al., 2000). These authors have mentioned factors such as personal interest, ideological commitment, development of skills (Ljungberg, 2000) or enhancement of reputation (Markus et al., 2000). Initial research has shown that learning and expression drives the contributions of the most productive participants, paid or unpaid (Lakhani & Wolf, 2003). Models also demonstrate that engineering can benefit from a reward system based on reputation, not unlike that of scientific research (Lerner & Tirole, 2002).

Third, a few authors have investigated the processes of FLOSS development (e.g., Raymond, 1998; Stewart & Ammeter, 2002). Much of this research examines factors for the success of (though there have been few systematic comparison across multiple projects, e.g., Stewart & Gosain, 2001). One problem is that success is defined differently for FLOSS projects than proprietary projects (Crowston et al, 2003; Crowston et al 2006a). Research on the strengths and limitations of the FLOSS development approach desperately need agreed and widely shared measures of team effectiveness to identify which teams to study in more depth.

Fourth, empirical work has begun to illuminate the structure and function of FLOSS development teams. Gallivan (2001) analyzes descriptions of the FLOSS process and suggests that teams rely on a variety of social control mechanisms rather than on trust. Several authors have described teams as having a hierarchical or onion-like structure (Cox, 1998; Gacek & Arief, 2004; Moon & Sproull, 2000). Active users also play an important role (O'Reilly, 1999). Research suggests that more than 50 percent of the time and cost of non-FLOSS software projects is consumed by mundane work such as testing (Shepard et al., 2001, p. 103). The FLOSS process enables hundreds of people to work on these parts of the process (Lee & Cole, 2003). Bug and issue tracking repositories and user mailing lists provide evidence about a development team's interaction with its users, the collection and implementation of requirements and the incidence of bugs and other defects (Stamelos et al., 2002). In efforts to improve security and quality of software, the debate rages between openness and secrecy, between expert and crowd approaches. The "many eyes" model of quality testing makes intriguing claims (Raymond, 1998), and research is seeking to first validate and then explain the operation and limits of this model. Intriguingly, it has been argued that the distributed nature of FLOSS development may actually lead to more robust and maintainable code. Because developers cannot consult each other easily, it may be that they make fewer assumptions about how their code will be used and thus write more robust code that is highly modularized (Lee & Cole, 2003).

Finally, some researchers have focused on the nature of the code developed, examining versioned histories of software as it develops, a team output, allows researchers to ask questions about the evolution of that software, "In what way does software evolve?" (German & Mockus, 2003; Smith et al., 2005), "Does adding more people length or shorten the software's development?" (Robles et al., 2005), "Does it always get more complex, or does re-factoring produce cycles of increasing and decreasing complexity?" "How do language features, such as automated garbage collection, effect

the software design over time?" There are a full range of software metrics that can be applied to measure features such as complexity, coupling, quality in terms of bugs and design clarity (Schach & Offutt, 2002), and comparisons with proprietary software can be undertaken (Paulson et al., 2004).

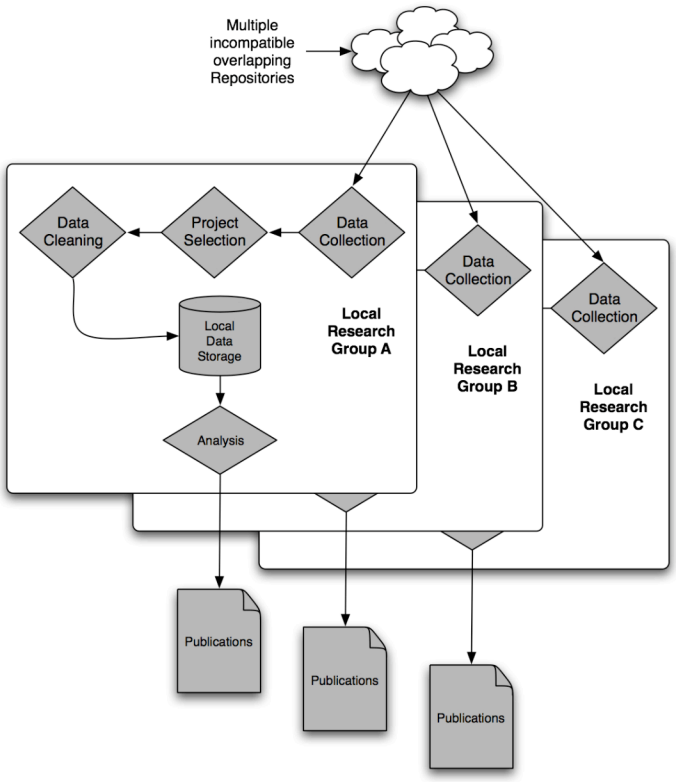

Figure 1: A typical workflow for FLOSS research in multiple groups. This process is non-collaborative, repeated and doesn't build on similar work of others, leading to papers that are difficult to relate to one another.

The problem to be addressed by the proposed infrastructure. The proposed project will develop an infrastructure to facilitate sharing of data and analyses in the FLOSS research community. The research briefly summarized above has relied on several different kinds of scientific evidence, such as the archives created by the FLOSS developers, versioned code repositories, mailing list messages and bug and issue tracking repositories (German, 2003). FLOSS teams retain and make public archives of many of their activities as by-products of their open technology-supported collaboration. However, the easy availability of primary data provides a misleading picture of ease of conducting research on FLOSS. Precisely because these data are by-products, they are generally not in a form that is useful for researchers. Instead potentially useful data is locked up in HTML pages, CVS log files, or text-only mailing list archives. Even those databases that are available (e.g., http://www.nd.edu/~oss/ Data/data.html) are dumps of databases designed and optimized for the delivery of websites, not for scientific inquiry. Furthermore because FLOSS projects are hosted in a variety of "forges" (of which Sourceforge is the largest) these problems are multiplied. FLOSS research projects, therefore, expend significant energy collecting and re-structuring these archives for their research. This process, depicted in Figure 1 adapted from (Howison et al., 2006a), is repetitive, inconsistent and wasteful.

For example, a typical approach for obtaining data is to spider the websites of the various project repositories. However, spidering data is fraught with practical complexities (Howison & Crowston, 2004). Spidering is a time-intensive and resource-consuming process, and one that is being

unnecessarily replicated throughout the world of FLOSS research. Since the data presented by the various repositories is usually stored in a database, one way around the problem of spidering is to access the databases directly. However, this approach still has difficulties. For one thing, the data may have to be anonymized (e.g., to remove developer emails or passwords) and must be refactored from the schema that supports a website into one more appropriate for scientific analysis. Furthermore, the data requires some level of cleaning. For example, one project studied had imported bug tracker data from another system by simply cutting and pasting. As a result, all of the imported bugs had opening and closing dates on the same day a minute or so apart. Had that data been analyzed as is, it would have seriously biased any results. Another problematic area is calculated fields, such as activity or downloads, for which there is incomplete publicly available information on their formula or correctness.

Even pristine and labeled data from repositories is not sufficient because of the need to integrate data from across multiple sources. Different repositories store different data. Different forges can have projects with the same names, different developers can have the same name across multiple forges, and the same developer can go by multiple names. Forges have different terminology for things like developer roles, project topics, and even programming languages. They often have fields which are named the same in multiple forges but which represent different data. Furthermore, because projects may pick and choose which piece of a forge they use, data for a single project may be distributed across various data sources, requiring additional work to integrate.

Each stage in the process is not only repeated by each research group, but is done differently in ways that are not always apparent from research publications (Hahsler & Koch, 2005; Howison et al., 2006a). This inconsistency hampers the reproducibility of the research and thus limits scientific progress. As a specific example, different researchers have examined different samples of FLOSS projects. Because of the cost involved in preparing data for analysis, researchers are rarely able or willing to rerun their analyses on different samples of projects. As well, it is not always clear exactly what projects are included in a sample, due to publishing space restrictions. As a result, research findings are difficult to cumulate, because of the likelihood that researchers are literally not talking about the same thing. Similar problems can be identified at each stage of the process above: different researchers will extract different data at different points in time, take different approaches to processing and cleaning data and make different decisions about analyses, but without all of these decisions being visible, auditable or reproducible.

In principle, these problems can be addressed by individual researchers better documenting what they have done. However, research publications typically have restrictions on publication lengths that make complete discussion impossible. Furthermore, published papers are just the tip of the iceberg, and knowing what others have done does not necessarily make it any easier to replicate the results. Therefore, our project proposed here, in addition to providing data in a consistent, research accessible form, will make it easier to share workflows, allowing researchers to build on each others' cleaning, sampling and analyses. The effects of decisions on results can also be investigated. Our system will not enforce or require the use of particular tool-chains--researchers are in the best position to make such choices--but will provide a framework for structuring and sharing tool-chains. As an initial proof of concept that lays fertile ground for collaboration, this proposal seeks funding for students to work with researchers to reproduce seminal findings in key publications using the data and toolchains from our system.

These problems are not limited to research on FLOSS or indeed to computing research in general. Increasingly every scientific field, such as biology and climate change research, is dealing with questions about the archiving, availability and usability of primary data sets (Anderson, 2004). The international organization CODATA has highlighted these issues in their call for datasets to be

treated as 'first-class scientific publications'. This proposal draws on their recommendations for data repositories, formalized in an ISO standard, in our implementation plans below. In a wider sense FLOSSmole aspires to the success of data and analysis repositories such as TREC (the Text REtrival Conference), funded by the NIST (http://trec.nist.gov/). TREC, running since 1992, has provided "gold standard" collections of text documents for evaluations of retrieval systems. The "gold standard" allows research groups to compare their success in identifying the documents that ought to be retrieved by particular queries. The TREC website claims that "Retrieval system effectiveness approximately doubled in the first six years of TREC" (http://trec.nist.gov/overview.html). TREC has provided a central object of collaboration, facilitated precise and comparable measurement of competing systems and facilitated transfer of technologies of retrieval to practice. FLOSSmole will provide a central object of collaboration and improve the precision and compatibility of project sampling and analyses. Over time we hope that the work built on FLOSSmole will become a gold standard for FLOSS analyses, but the first step is to provide the object of collaboration, the data and analyses archive.

1.3 Related projects

The problems described above have already impelled the creation of a nascent infrastructure for research. There are three closely related projects that provide a good basis for the development described in this proposal, and others that provide an inspirational model. However, the limitations of these projects shows the problems inherent with assuming that the necessary sharing of data and analyses will happen in the context of a single grant and the importance of an infrastructure for promoting synergies across projects.

The proposed work will continue and expand the efforts of the first of the projects, FLOSSmole (http://ossmole.sourceforge.net). FLOSSmole is a current joint project of the two PIs (Conklin et al., 2005; Howison et al., 2006a; Howison et al., 2005). Even in its infancy, the FLOSSmole project data has already been used by many international researchers (Crowston et al., 2005a; Crowston & Howison, 2005; Howison et al., 2006a; Weiss). As the attached letters of support indicate (see Supplementary Documentation), academic researchers and practitioners in industry are convinced that it is important to continue the work begun with this project. FLOSSmole provides high-quality and widely-used datasets of data about FLOSS projects from a central repository of data that have been collected and prepared in a decentralized manner. The initial data were the products of spider runs that each researcher had done separately and the project continues to spider several forges on a regular basis to keep the data current. As well, the project has received donations of data from other researchers and even corporate partners on a limited basis. However, FLOSSmole has no separate support; instead, it has been funded through the private efforts of the PIs along with spin-offs of their funded research. This lack of funding limits how much support can be provided to the community on using the data. As well, FLOSSmole currently addresses only the initial stage of the chain described above, making raw data more widely available. It does not currently have the resources to support more intensive data cleaning or integration or to provide ways to share analyses and results. Adding these capabilities is the goal of the present proposal.

A second closely related infrastructure project are the regular dumps of the Sourceforge database made available to academics through cooperation between Greg Madey, at the University of Notre Dame, and Sourceforge. This archive was funded in part by National Science Foundation, CISE/IIS-Digital Society & Technology, under Grant No. 02-22829. The data has been used very productively by researchers at Notre Dame and collaborators for modeling the activities of open source teams (e.g., Christley & Madey, 2005; Xu et al., 2005). This archive provides a number of lessons for the proposed infrastructure development. First, as with FLOSSmole, the project addresses only the first step in the research process by providing access to raw data, but without supporting the other stages

of the process. Second, it is limited to one source of data, SourceForge, and so does not address data integration, something that seems unlikely to happen without a project devoted to infrastructure. Third, the databases provided are direct replications of the database underlying the SourceForge website, and so are not easily understood in research terms (a wiki has recently been created to gather notes on deciphering the structure and its research meaning, but again further efforts in this direction will require direct support). Fourth, the community can currently access the data only via a web-based query form, which makes accessing the data in research-ready form a difficult manual process (we hypothesize that the Notre Dame research group has more direct access to the database, explaining their greater ability to use the data productively). Finally, the scope of the community is restricted because data is available only to academic researchers and then only under a contract that forbids further sharing of the data and publications that could be construed as a criticism of Sourceforge or the University of Notre Dame. Despite these limitations, the SourceForge/Notre Dame data is a valuable resource. While FLOSSmole will not be able to include that data in our archive, licensed users of that data will be able to independently include it in their analyses. For example, FLOSSmole could develop modules that will assist licensed users to access the Notre Dame repository to run queries and retrieve the data into local databases for analysis.

A final related project is CVSanalY (Robles et al., 2004), based in Spain and funded under the EU's 6th framework. This repository provides tools to extract and analyse the log files from project's source code revision control systems (CVS and SVN), thus complementing the data provided by the previous two projects. CVSanalY has made available the results of using their scripts on the entire Sourceforge project set. CVSanalY has provided data for publications at both the ICSE Mining Software Repositories and the PROMISE workshop described below (González-Barahona & Robles, 2004; Massey, 2005; Robles et al., 2005; Robles-Martínez et al., 2003). The organizers of CVSanalY and FLOSSmole have cooperated in the past. The CVSanalY principals were co-conveners, with this project's PI, of the Workshop on Public Data about Software Development (WoPDaSD) at the IFIP Open Source Working Group conference in July 2006. CVSanalY and FLOSSmole have also standardized their format for naming projects so that analyses can more easily integrate data from both datasets. The CVSanalY principals are on the advisory board for this project, described below, and we intend to integrate the CVSanalY data and the FLOSSmole repository and to continue working together in building this research community.

Beyond FLOSS research, the PROMISE archive in Software Engineering (Cukic, 2005; Sayyad Shirabad & Menzies, 2005) collects data sets designed to help create predictor models, such as understanding which modules are most likely to develop defects. They model themselves on the successful UCI Machine Learning data repository, which functions in a similar manner to TREC (Newman et al., 1998). The PROMISE workshop in February 2005 saw the publication of seven papers, and the donation of the datasets for each of these. The best papers were republished in the November/December edition of IEEE Software. A similar workshop is planned for 2006, with a special edition of the Journal of Empirical Software Engineering. The PROMISE data includes a well-respected set from NASA and a number of open source software projects. They do not collect data themselves, but re-publish the data collected by research teams, together with metadata about the data. The emphasis of the repository is on learning models for the software itself, and, although they are likely open to it, it does not extend to archives of project communications. The request for a standardized format for the data and metadata about its collection is a step in the right direction, and we anticipate that our version-controlled approach to a common database and analysis workflow ought to improve the reproducibility and ease of collaboration for such repositories. The FLOSSmole advisory board includes participants in the PROMISE workshops and repository and we hope to work with and learn from them.

The existing work of the FLOSSmole project and other FLOSS data repositories demonstrates that research can be boosted by building a community around universally available, research-ready data archives. When research is based on the same data sets, researchers avoid talking at cross purposes, allowing more precise conversations that advance the field. The proposed additions to FLOSSmole that will be funded by this grant goes further than existing repositories, assisting with automated data collection including sources outside Sourceforge, and designing a collaborative workflow to improve the ease of sharing sampling and analysis techniques. These developments are not likely to happen without specific infrastructure support, but would greatly enhance the synergies of different research projects.

1.4 Community input

In order to better understand the kind of infrastructure that would be useful in supporting the research community, the PIs have both organized several workshops discussing FLOSS data availability and sharing over the past two years. These meetings have also laid the groundwork for the collaboration that would be extended by the work described in this proposal. Megan Conklin co-convened a workshop, held in July 2006, at the recently formed International Federation for Information Processing (IFIP) Open-Source Software Working Group 2.13 Conference, which brought together software engineering researchers making active use of large scale FLOSS data archives. Kevin Crowston co-convened a professional development workshop, filled to capacity, on FLOSS for the Organizations, Communications and Information Systems division of the Academy of Management, held in August 2006 (http://floss.syr.edu/ Presentations/FlossDataTutAoM2006/). In January 2005-2007, Kevin Crowston co-chaired the mini-track on FLOSS research at the Hawai'i International Conference on System Sciences (HICSS), and specifically included data archive use on that agenda.

Another important venue for soliciting community involvement has been the annual Mining Software Repositories (MSR) workshops at the International Conference on Software Engineering (ICSE) (http://msr.uwaterloo.ca/msr2006/). The 2006 workshop attracted over 30 papers. Both PIs have attended and published in these workshops over the past three years (Conklin et al., 2005; Howison & Crowston, 2004) and built close working relationships with the participants, who are increasingly turning to research on FLOSS, while maintaining valuable comparisons with proprietary development. This year the workshop instituted a "Challenge", which asked the community to base their research on two FLOSS sources (ArgoUML and CVSanalY, mentioned in section 1.3 above), demonstrating the desire in the community for highly comparable and reproducible results. The organizers of the Challenge attended the IFIP workshop described above.

To ensure that the infrastructure developed continues to be responsive to the broad research and education community around FLOSS, we have formed a board of advisors. The board includes both experts in scientific data sharing as well as leading FLOSS researchers, including several associated with the related projects mentioned above. Members of our Advisory Board include leading researchers on FLOSS (Karim Lakhani, Harvard Business School; Martin Michlmayr, University of Cambridge, U.K.; Gregorio Robles, Universidad Rey Juan Carlos, Spain) and experts in scholarly communication, digital libraries, scientific information (Christine Borgman, Professor & Presidential Chair in Information Studies, UCLA; R. David Lankes, Associate Professor, Syracuse University; Jian Qin, Associate Professor, Syracuse University).

1.5 Conclusion: An infrastructure to support FLOSS research is needed

The preceding sections have made the argument that FLOSS research is important but that the current approach to the research leads to needless duplication of effort, resulting in slower progress and less insight. This proposal will enable a step-up in the quantity and quality of FLOSSmole's archives and facilitate community growth through dissemination activities. A funded FLOSSMole

will produce a data and analysis archive for an entire sub-field of researchers and will extend the set of individuals and departments that are able to conduct research on free and open source software and its development. The end result will be more reproducible, traceable research that proceeds at a faster pace with increased quality as researchers share data but also their techniques for cleaning, sampling and analysis and link to these from their papers.

2. Infrastructure to be developed

In this section we describe the infrastructure to be developed in more detail, providing a description of the desired functionality and some technical details, though of course the detailed technical design and development is what would be funded by the grant. We also note several issues that will have to be addressed in the project development and our initial thoughts on those topics.

2.1 An archive for data and analysis workflows

The project will implement an archive to make available FLOSS data and analyses in a manner that enables research teams in the community to maintain their own academic workflow, but also to build on other's work and to make it as easy as possible to share their data and analysis with the FLOSS research community. FLOSSmole will support a framework workflow with the steps identified in Figure 2:

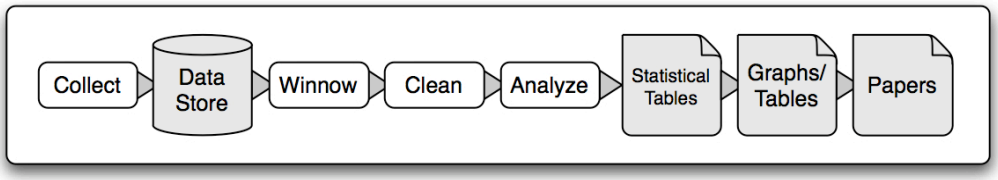

Figure 2: FLOSSmole workflow

Collection scripts bring data, either from direct dumps, or from web spidering, into the project's data store. The next three steps are steps in specific research projects, where researchers need to winnow projects under consideration, e.g., sampling on some basis such as software language or team growth. Cleaning the data may involve ensuring consistent identifiers and weeding out the many dead-on-arrival code dumps that may skew analyses. Analyses are the highly varied methods of examining the data, such as creating software metrics, examining defect cycles and lifetimes, examining regularity and pace of development or conducting social network analysis. Often these are the heart of the research and represent significant contributions. Analyses produce artifacts such as statistical tables for correlation analysis and graphs or tables for the results sections of papers. Storing the completed paper, typically in PDF, allows future researchers to observe the entire workflow and its eventual representation in a published paper.

Currently each team has developed its data store and workflow in locally efficient but globally idiosyncratic ways, based, perhaps, on the style of individual graduate students. These incompatibilities mean that sharing is difficult and time-consuming. FLOSSmole will not enforce the use of identical tools, but will provide a shared reference point for structuring research involving these large data sets. As an example of the kind of detail that might be addressed, it is important that the character encoding used across repositories is preserved throughout the toolchain, particularly for qualitative and mutli-language work. Unicode provides the tools to do so, but unfortunately the differing agendas of different groups has, to date, meant that collected data sets do not have full provenance nor consistent character encodings. The funding requested in this proposal would allow us to develop tools that make it easy for collaborating researchers to 'do the right thing'.

A key aspect of the development of a useful data archive is the development of appropriate metadata to describe the data collected. Best practices for data archives have been addressed by the data

archiving and digital library research communities (e.g., Anderson, 2002; 2004; Borgman, 2007). Experience from the establishment of data archives in the areas of astronomy and climate data have been generalized and formalized in an ISO standard, "Reference Model for an Open Archival Information System" (http://nost.gsfc.nasa.gov/isoas/). They identify Metadata, including Nomenclature, as primary considerations (along with Incentives for participation, Funding, Selection and Appraisal, and Planning for long-term access, which we address below). Accurate and useful metadata is vital and three main categories have been identified: 1) technical (bits to data), 2) provenance and context discovery (data to information) and 3) documentation of use (information to knowledge) (Woodyard, 2002).

For example, one problem we are currently facing is that available data comes from many difference repositories, and many projects use multiple parts of several different repositories. Software developers and team members use multiple email addresses, and most have different usernames on different repositories. It is important that these identifiers be standardized and mappings established. It is exactly this type of painstaking manual work that is hardest and most wasteful to reproduce. Multiple research teams should not waste time on these duplicative efforts but instead enjoy the synergies of building on prior work. FLOSSmole will collect such work and devote development resources to matching across repositories. Such work will benefit from the research on nomenclature standardization in the digital archives literature (Woodyard, 2002). We will also work with projects like the Galactic Project Registry, the aim of which is to begin to standardize naming conventions and project descriptors across projects.

This project has three advisory board members (Borgman, Qin and Lankes) from the digital library community to assist us in meeting relevant standards and learning from the research in this area. Storage of full workflows, from collection scripts to full database history, to separate steps of winnowing and cleaning the data will allow our repository to 'self-document' and go a substantial way to forming the metadata for provenance highlighted in the digital archives literature.

2.2 Technical plan

In this section we describe the technical infrastructure to be developed in more detail, though the detailed design will be an activity to be supported by the grant. As noted above, we are aware that comparable infrastructures have been developed in other scientific disciplines. To the extent possible, we plan to build on the software created for these projects, including that created by the NSF cyberinfrastructure project grants, and thus avoid reinventing the wheel. However, we anticipate that the specific nature of the data and analyses in FLOSS research will require some tailoring to be applicable. An early step in the design process for each stage of development will include a careful assessment of existing packages that could be used as a basis for development.

The system will support a modular data flow, where each step in an analysis will be stored as a directory under version control, e.g., in a source code control system such as SVN. The system interface will allow flexible combination and recombination of scripts, e.g., a script to select a sample of projects connected to scripts for performing various analyses or producing graphs and tables. Existing collection scripts will be improved and new scripts written and maintained by our project, allowing researchers to start with cleaned up data. Each data collection run will result in a database file in the data store. Providing the data as a series of database dump files will enable individual project teams to maintain local database servers, improving speed and eliminating the substantial overhead of the project team managing a shared database server, query tool and authentication system. Using a revision control system will enable new data collection files to be obtained with a simple download, increasing the ease and speed of data distribution. Effort will be required to update

past collections to the current database schema, but revision control will assist in tracking such changes.

The project needs to store the full history of a database so that analyses and workflows can specify that they start with the database as it was at a particular point in time, both data and schema. The project will explore options for such database capabilities, but the initial candidate technique is to simply record the history of additions/changes to the database using the binary logging capabilities of the open source MySQL database. These full log files can be 'feed' to a local database with a 'stop date', allowing scripts to access the database state as it was when their work was completed, including the schema current at that time.

The further steps of the workflow will operate on the local database, using SQL queries to winnow (creating smaller samples for analysis) and clean as locally required. Analysis scripts will query the database, likely creating statistical tables to be represented as graphs or results tables and finally included into a paper.

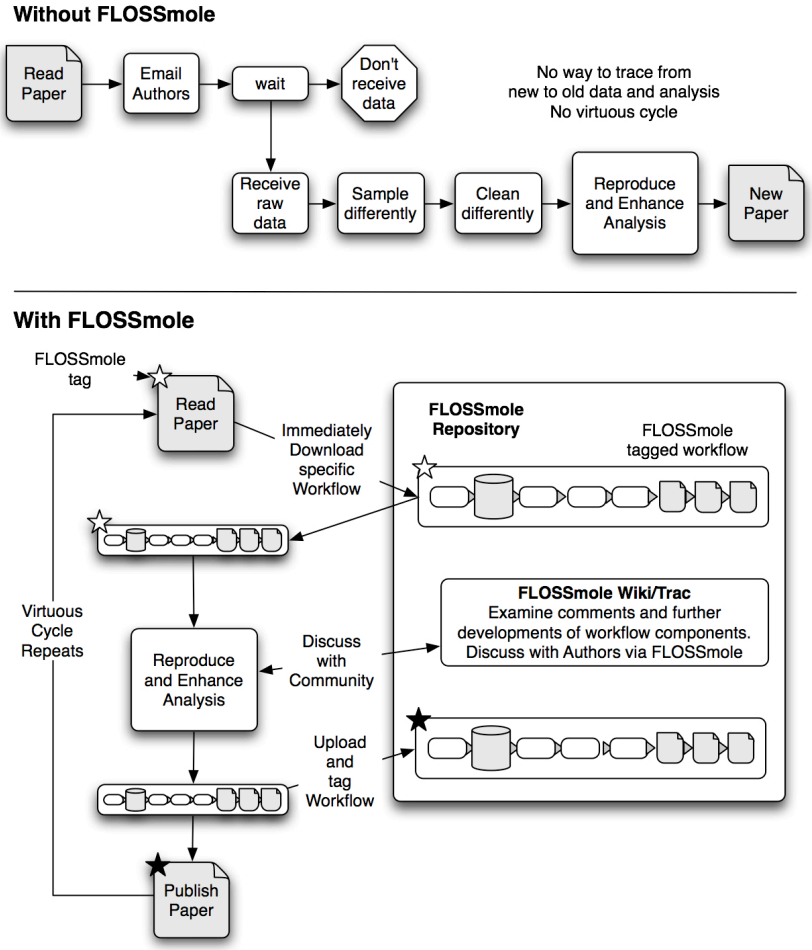

Figure 3: A comparison of a researcher seeking to build on existing research with and without the FLOSSmole repository. With FLOSSmole the process is quicker, cumulative and brings community knowledge to bear.

Once a piece of research is complete the research team may upload their changes and tag the repository with a tag uniquely identifying the workflow for that paper. Including a reference to FLOSSmole with that tag in the paper, would enable interested readers to check-out the repository, data and analyses, exactly as they were when the paper was written. The project will explore the

recently created Digital Object Indentifier (DOI) namespace for datasets by the TIB DOI Registration Agency (DOI Foundation, 2005).

Future researchers could then selectively update the datastore directory, or try a different sample or cleaning technique, to see the impact on the published analysis. By viewing changes forward from that tag and by referring to the wiki pages discussing them, researchers can quickly see what further has been done with these data or analyses. Furthermore, future papers can "pick and choose" individual elements, such as cleaning decisions, from stored workflows simply by providing a the relevant tags in their papers. This set-up for the repository facilitates "view source" on FLOSS research—direct, reproducible communication of research decisions.

The project will, over time, develop a library of concepts and their measures that will assist researchers in conceptualizing their research and in implementing their ideas in a comparable manner (this corresponds to the nomenclature aspect of the metadata mentioned above). For example, a researcher interested in investigating the relationship between effort and success would be able to query the concept library, discovering similar measures already implemented in earlier workflows. They would be able to pull up workflows employing these measures, discuss their operation, and adapt elements of earlier work to their proposed new measures. Once their new measure was available to the system, they could re-run the earlier analyses and observe the results of their new conceptualization. Clearly there are significant challenges in this vision of 'pluggable research', but closely documented data and storage of workflows in the accessible archive that we propose are the first step. In Year 3, our goal is to pursue the development of the ontology and explore the design and implementation of interfaces, including graphical interfaces, that improve the 'pluggablility' of different steps in the workflows. The re-constructed seminal papers will be an excellent source for such analyses.

Clearly not all research efforts are alike. Some may use each step of the workflow while others may begin or end anywhere in the chain, as local requirements dictate. Further not all projects will use similar tools at different steps, some may prefer perl or ruby, while others may make use of interactive activities, such as outlier elimination or graphical statistics programs, and as a result they may not be fully "compilable" (in which case detailed descriptive documentation would be requested). However those teams that wish to make use of elements of other teams' analyses will be able to do this much more easily. Workspaces on FLOSSmole's central server could be made available for those without adequate systems of their own.

FLOSSmole intends to impose strict compatibility standards on the data collection and storage formats, but not on the succeeding steps in the workflow. Compatibility between the past workflows of research teams would likely be a matter of merging SVN branches manually, but the point is that the system makes it possible for those who wish to coordinate to do so.

2.3 Issues to be addressed

In addition to the technical issues briefly described above, our project will address two social issues that are critical for the project's success: incentives for contribution and privacy.

Incentives. Our repository design addresses the question of incentives for contribution, noted above as one of the key issues in data archives. Prior efforts in this area make it clear that "build it and they will come" does not work. Promoting contribution will be an on-going effort in the project, so here we provide our initial thoughts. First, it is critical that the cost of contributing be as low as possible. Part of the funding for the project in the initial years will support students to help with this process. Second, contributing should provide specific and visible benefits to researchers so that archiving will not only be a "post-research project activity ... a necessary overhead" (Anderson, 2002, p 193).

Rather than merely storing primary data, our workflow approach is intended to actually assist groups during their research activities. The historical workflows, the community wiki and, eventually, the conceptual ontology, will all assist in the process of research, as depicted in Figure 3 above. Data and workflow sharing does not only occur across different research teams, but perhaps more often within such teams, especially as graduate students come and go over time. The repository therefore becomes a convenient memory for local research teams, who may also retain local repositories to incorporate data they are not yet able to share (such as data on proprietary organizations), while pushing public data and analyses into the shared server. This is an additional incentive for research groups to adopt compatible practices and to share their work. Finally, it is important that sharing have benefits beyond the immediate research. The lack of academic recognition for work in contributing to data archives is a current disincentive for participation (Anderson, 2004). However, our workflow approach binds the contribution of data to papers published with that data, and contributors will be able to count uses of the data and workflow as citations to the original paper. In this way, the repository's ability to "track forward" will be additional evidence of author's contributions to their fields of research. As well, our dissemination work will include work with journal editors and conference organizers to promote repository use, thus creating further participation incentives.

Privacy. FLOSSmole will only collect data, communications, and activity artifacts that were intended to be made public, such as code contributions or public email messages. We believe, and our advisory board confirms, that the repository will be of great benefit to the community if we continue to operate in the spirit of open collaboration exemplified by FLOSS development. However there may be significant ethical considerations in the aggregation and reporting of this data for purposes other than the production of the FLOSS software. There is an ongoing vigorous debate in our research community about breaching developer privacy in any large system of aggregated development data (Robles et al., 2005). For example, if we aggregate several code repositories and are now able to show in a colorful graph that Suzy Developer is 10 times more productive than Bob Coder, does this violate Bob's privacy? If we can show that Suzy's code changes are 5 times more likely to cause errors than Bob's, does that violate Suzy's privacy? The next generation of FLOSSmole should have the ability to hash the unique keys indicating a developer's identity. It may also be advisable to provide a simple "opt-out" system for projects or individuals who do not wish to be in the repository or its reported results. This effort will have to be researched, implemented, and documented for our community. Fortunately for our team, Gregorio Robles, the lead author of the study cited above, will be a member of our advisory board.

3. Research Projects that will benefit from this proposal

In this section we briefly describe the research and education activities the infrastructure will enable, by describing some research and education projects that will use the infrastructure or be enabled by it, as well as some thought about how the project can increase participation in CISE research. The community the project will serve includes existing FLOSSMole partners, participants in the co-PIs FLOSS data workshops, as well as current and future NSF funded open source software and virtual teams research groups. We have invited key members of this community to sit on our advisory board and we attach nine letters of support from research groups to this proposal. This is a substantial community that will be able to draw on our data and analysis archive to improve the quality and pace of research in a manner that would not be possible without funding this project. In the remainder of this section, we discuss these groups in more detail.

3.1 FLOSSmole's current participants

One of the most important goals of the initial version of the FLOSSmole project is that the data, code, and database schemas should be easily accessible to researchers, without sign-up or contracts.

This stance reflects the principles behind open source software itself--if a user wants to look at the code, she is free to do so. This commitment to accessibility has served FLOSSmole well already. The SWIK project (http://swik.sourcelabs.com), an independent effort by programmers at Sourcelabs, is a wiki-based database of open source projects. This entire project was created in one month, using data made public by FLOSSmole. These industry practitioners obtained our data from the repository at their convenience, created their project, and sent us a thank-you note telling us about the project when they were done. Another corporation, O'Reilly & Associates (http://www.ora.com/), has a research group that has used our data to predict book sales on technical topics and to predict trends in FLOSS development. They continue to be impressed by the easy access and high quality of our data.

In another illustrative case, Dawid Weiss, a researcher in Poland, had written a paper in which he described various conclusions about FLOSS development after collecting data from Sourceforge (Weiss, 2005). After discovering the FLOSSmole data, Weiss then compared his data and collection methodology to the FLOSSmole data collection techniques and results and revised his initial paper to reflect this comparison. He found our FLOSSmole dataset online, conducted numerous analyses over the span of a few days, then contacted our team to share his results after the fact. Similarly, Joseph Davis, an Australian researcher, was able to donate data from his research project, and that data filled a gap in our early data collection efforts.

These experiences illustrate the convenience and necessity of having a publicly-available dataset of this information. No advance notice or coordination was necessary for these researchers to use our data. Because our project was designed with collaboration in mind, these sorts of comparative results can now be easily integrated into the FLOSSmole database, and then used in tandem with other FLOSSmole data or alone. As such, we have now fully integrated the Weiss and Davis data into the FLOSSmole database, and Weiss is now an active developer and contributor to our mailing list. The Swik founder, Alex Bosworth, is still regular contributor and user, as is Roger Magoulas from O'Reilly research group. Both have attached letters of support.

American Universities		
• Carnegie Mellon University	• Brigham Young U.	• Elon University
• U. of Massachussets, Amherst	• U. of North Carolina, Wilmington	• Syracuse University
	• U. of California, Davis	• University of Southern California
International Universities		
• Aristotle U. of Thessaloniki (Greece)	• Swiss Federal Institute of Technology (Switzerland)	• Poznan U. of Technology (Poland)
• U. Federal de Para (Brazil)	• U. Hildesheim (Germany)	• Wirtschafts U., Wein (Austria)
• U. Rey Juan Carlos (Spain)	• Simon Fraser U. (Canada)	• Taiwan U. (Taiwan)
Corporate participants		
• Microsoft	• O'Reilly and Associates	• Sourcelabs
• Spikesource	• Charles River Ventures	• TransPac/Krugle

Table 1: Current FLOSSmole Participants

FLOSSmole's current mailing list has 30 subscribers, quite a few of whom joined in order to represent a larger research group, and this list is growing at a rate of 1 or 2 new subscriptions per month. The number of downloads of our data sets has grown from 8 in October 2004 (our first release), to 119 in October 2005, to 435 in October of 2006 (our most recent release as of this writing). This growth is one of the motivators for this proposal. Table 1 shows some of our most active participants.

3.2 Providing Leverage to existing NSF funding

The NSF has recognized the growing importance of research on FLOSS and its development by funding a number of research teams.

We searched the NSF database for grants that would benefit from this research infrastructure. We identified nine currently funded projects, all from the CISE IIS and CCF programs, with total funding of $4.1m. In the past five years another twelve projects, 10 from CISE and 2 from SBE have been funded for a total of $2.7m. (Included in these figures are two research grants and one planning grant for Kevin Crowston, our co-PI.) These projects include efforts such as "Longitudinal effects of Design in Open Source Projects" (Premkumar Devanbu, #0613949, $750,000), "Coordination, communication, and collaboration in open source software development" (James Herbsleb, #414698, $400,000), "Discovering the Processes, Practices, Community Dynamics and Principles for Developing Open Source Software Systems" (Walt Scacchi, #534771, $115,999) and "Organizational Dynamics of Software Problems, Bugs, Failures and Repairs" (Leslie Gasser, #205346, $545,991). A full list of these projects has been added to the Supplementary Documents portion of this proposal.

Each of these projects will have devoted valuable time to collecting and cleaning data on FLOSS and its development. This proposal provides a way for the NSF to leverage its existing support; FLOSSmole encourages researchers to focus their efforts on the high value-added steps in the research chain, freeing these groups from using funding for repeated episodes of data gathering. Furthermore, the collaborative design of FLOSSmole ensures that researchers will be able to easily share their analyses as well as the data they have collected to date. The PIs have begun to contact these projects for collaboration, and two of the PIs are already on our advisory board and have written letters of support (Dr Devanbu and Dr Scacchi). Three others are personally known to the PIs, through the data workshops co-organized by the PIs, described above, and other research venues. Funding this proposal would leverage the money that NSF has already invested and will invest going forward in research on FLOSS and its development.

4. Management plan

In this section we describe how the infrastructure described above will be created. The proposed project will be carried out as a collaboration between the Department of Computing Sciences at Elon University and the School of Information Studies at Syracuse University. Elon, an undergraduate institution, will be responsible for the bulk of the implementation and development work. Syracuse University will be responsible for requirements determination, based on already existing research projects, community development, consultation with the digital archives research and the development of the concept library. In the remainder of this section we describe in turn the specific work tasks, the coordination plan, and plans for dissemination, sustainability and evaluation.

4.1 Work tasks

The proposed project includes six main work tasks (including dissemination). Each of the first five tasks is described below, and the overall work plan is summarized in the timeline shown on page 3 of the budget justification. Dissemination is discussed in section 4.3, "Dissemination Plan". The technical basis for each of these tasks was discussed earlier in section 2.2, "Technical Plan".

Planning and coordination. The project team and the research community will draw on the existing infrastructure of the FLOSSmole project and make significant improvements. Some of the collaborative tools for this project are currently housed at Sourceforge.net, whereas the hardware and database are hosted at Syracuse. The ossmole-discuss mailing list and the #ossmole IRC channel on freenet have proven to be lively and fruitful venues for collaboration. We will be able to take advantage of the issue tracking and possibly the SVN hosting provided by Sourceforge. These are tools that are familiar to both researchers and practitioners in the FLOSS community. The budget also contains provision for travel, which will be used to travel to community relevant conferences for dissemination (see section 4.3 below and additional information in budget justification document).

Collection script development and maintenance. The PI, Megan Conklin, will oversee undergraduate computer science students who will prepare, document and maintain collection scripts for the various forges of interest. Currently we have scripts that collect project metadata from Sourceforge, Freshmeat, Rubyforge, and ObjectWeb. We will add support for other sizable code forges, such as the new repositories introduced by Google, Apple and Sun. This will be the primary responsibility of the Elon team; its PI will be assisted by undergraduate computer science students to accomplish this.

Database Storage. The project team must investigate the best way to hold the full historical history of a database, including schema changes, in a manner so that changes to the database can be quickly distributed to collaborator's local databases. The current candidate is to maintain and to "re-play on demand" the binary logs from a MySQL server, allowing incremental distribution of newly collected data. However, further investigation may reveal a need for development work on this topic. The implementation of the database plans is the primary responsibility of the team at Elon.

Developing access methods (support, web, and version control). The team at Elon will take primary responsibility for developing the web interface for the project, the visualization and query applications, the workflow located in the SVN repository, and its management practices. The primary data and application servers will be hosted at Elon. The team at Syracuse will also maintain a mirror of the primary servers.

Reconstruction of seminal papers. This work effort includes the reconstruction of workflow, data, and analyses of seminal published papers in the field of empirical FLOSS software engineering. The project intends to proactively contact authors of seminal papers in FLOSS research and include their data and analyses in the archive. Ideally the authors themselves will work with the project team to adapt their data and work for compatibility with the FLOSSmole data archive and workflow. However we have budgeted sufficient undergraduate and graduate assistance to reproduce the important analyses in collaboration with the authors. This service to the community will ensure that FLOSSmole will reflect the best work in the field and allow further work to start with the "best practices". This will be primarily the responsibility of the Syracuse graduate assistant who will draw on the digital archiving recommendations with regard to selection and appraisal.

Development of concept ontology and metadata. As the archive grows, the data and analyses within it will begin to form a picture of the ontology of concepts in the field. In Year 3, the Syracuse Information School GA will develop this ontology and, with assistance from the Elon team, an interface to access and learn from it.

Based on preliminary assessment of the development effort required, we are requesting funding for one graduate student at Syracuse University and for two undergraduate students at Elon University (only one during the first year), as well as a small amount of funding to support the PIs. A timeline is included in the budget justification document that shows the planned schedule for the development activities.

4.2 Coordination plan

Principal Investigators. Both PIs will devote effort during the academic year to project management and oversight. All PIs will share in overall project design and report writing. Each PI will be responsible for overseeing work on those aspects:

- Dr. Conklin will direct the project and be responsible for general project oversight and reporting to NSF, as well as technical design of the collaboratory and overseeing development.
- Dr. Crowston will oversee work on the requirements determination.

Project Management. We will use two project management techniques to coordinate the work on this project. First, we will have regular all-hands meetings of the project members to share findings and to plan the work. Initially, these meetings will be every other week, but the frequency of meetings will be adjusted depending on our experience and the pace of the work being carried out at the time. These formal meetings of all project participants will augment the regular interaction of the teams of PIs and students working on the data analysis and expected frequent interactions of the students. The undergraduate students will meet at least weekly with each other and the PI in a formal setting to discuss progress. The PI anticipates working with the undergraduates informally on a near-daily basis during the summers. Second, an initial project activity will be the development of a more detailed timeline (based on the initial one found in the budget justification document) for measuring progress.

4.3 Dissemination plan

Technical Support to Research Community. Time is allocated for students and project team members to assist our end-users in planning their FLOSSmole research. A project wiki will be established and will be a primary source of documentation for the project, but we will also prepare screencasts (screen capture videos) of development and use of the FLOSSmole repository. Our end users also have been interested in seeing "best practices" documents or case studies showing how other researchers have used the data. Excellent support and documentation are fundamental to ensuring that the use of the data and analyses in the repository results in high-quality research. This responsibility will be shared between the participants at Elon and Syracuse.

Encouraging Contributions. FLOSSmole is a collaborative project and thus its success depends, in part, on the quality of the collaborators and community that forms around it. In addition to an excellent web presence and lively electronic discussion groups, the project will go "on the road" to the venues in which active FLOSS researchers gather. The PIs are actively publishing in these venues, and have conducted data-focused workshops in recent years, which are discussed in detail in section 1.4, "Community Input" and in the next line item below. Encouraging contributions will be a shared responsibility of the project PIs.

Workshops. Our budget includes funds for annual workshops (one per year of the project). The purpose of the workshops is to introduce the collaboratory project, discuss the available data, work through a number of workflows, demonstrating how the data is accessed and how one can use existing samples or build on other's analyses. We will encourage the addition of data collected by those in the audience and offer our services in replicating seminal analyses. Workshop planning and attendance will be a shared responsibility of both project PIs.

Encouraging Research Use of Data. The project will also work with journal editors and conference organizers, including those on our advisory board, to promote calls for papers based on data and workflows in the FLOSSmole repository. As discussed in section 1.3 "Related Projects", this strategy has worked well for the creators of the PROMISE repository. We expect that such calls will be a source of additional contributions to the archive, as contributors will be expected to contribute their data and analyses to that the resulting papers can be "checked-out" of the repository by the research audience, demonstrating the collaboration that would not be possible without the developments funded under the FLOSSmole project. Encouraging the use of data for research will be a shared responsibility of both project PIs.

Broadening participation. The activity timeline includes a substantial amount of time spent in year two to develop appropriate educational materials for showing how to use the valuable FLOSSmole data in academic settings, especially with undergraduates. Since Elon University is an undergraduate institution, these educational materials will be designed to achieve specific goals and objectives in Elon's related computer science and information systems courses such as database systems, net-

centric computing, high-performance and collaborative computing, and senior seminar courses. We expect that these materials can easily be tested at Elon and provided in an open source context to other universities as well, perhaps in collaboration with the NSF funded 'Recourse' project to bring open source methods to software engineering research (Dionisio, 2006) There is more description of the undergraduate component of this project in the RUI Impact Statement. Thus, broadening participation through dissemination of pedagogical materials for undergraduates is primarily a responsibility of the project PI at Elon.

4.4 Sustainability

The FLOSSmole project will create a long-term community resource, and we are conscious of the need to ensure it can function long after the currently requested support. Firstly, the data and workflows collected during the project period will be maintained in the repository. The hosting requirements for the repository are minimal, and we expect hardware and storage costs to drop faster than our data expands, meaning that the central repository can be maintained on departmental computing resources. The repository set-up lends itself to simple migration and mirroring.

Despite all efforts to link directly to forge databases, spidering scripts will likely remain important for future data gathering. The scripts developed by the project will require maintenance, as forges alter their sites for future functionality. We expect that community members will take on maintenance responsibilities and keep those scripts up-to-date using the infrastructure created during the project. Community members will be motivated by their on-going research needs. Research on online collaboration and scientific collaborations, including ours, argues that the best start for an open project is a clear center with a committed vision, a "cathedral" before the "bazaar," that creates the grounds for collaboration (Senyard & Michlmayr, 2004).

The main activity that will not be possible without continued funding is providing active support by students for new projects and for the archiving of seminal research. However, we anticipate that the documentation and community built up during the funded period will create a vibrant community capable of self-supporting. We will also encourage projects seeking NSF support to designate a portion of their budget to costs associated with producing a high-quality workflow for sharing with the FLOSSmole project. It is our hope that the project will be beneficial enough to researchers that being archived will be a mark of quality and an activity looked on favorably by reviewers in the grant award process. Syracuse or Elon may be able to provide experienced students for sub-contract to assist projects in using the repository and archiving their data and workflows.

4.5 Evaluation plan

We plan to evaluate the project at least annually to gauge the degree of impact it is having and to provide feedback for future development. One of the tasks for the initial months of the grant is to develop a more complete evaluation plan, so in this section we briefly describe our initial plans. We have identified three main stakeholders for the project: 1) the researchers working with FLOSS data, who want better access to data and analyses, 2) the students who will be working on the project, who want an enriched educational opportunity, and 3) NSF, who will be providing the funding to "enable discovery, learning, and innovation in all computing fields". Additional stakeholders may be identified in the future, e.g., those who might use the services to support educational activities.

To assess how well the project does in satisfying the first set of stakeholders, the researchers, we will track the number of inquiries about the project, the number of active users of the data and analyses, and the number who contribute data and analyses and who cite the archive in their publications. During the first year, we will set specific targets to be meet for each of these, but 25-50 active users seems achievable based on the number of researchers in the area, the number of current users of

FLOSSmole, and the number that could be supported with the requested funding. To gauge satisfaction with the project, we will periodically survey users and non-users to understand what additional services are needed.

To assess the project's success with the student assistants, we will develop measurable outcomes for what students will learn from their participation. As described in the RUI Impact Statement included with the Elon proposal, undergraduate students benefit enormously from experience building software for real-world projects and working with the faculty mentor in an undergraduate research and development setting. Additionally, through our development of educational materials ("hands-on" labs and experiential activities that use our data) for courses such as data mining and high-performance computing, we hope to show increases in positive student attitudes toward the relevance of their own coursework. These materials can be used with students who are not working as assistants on this project, but who can benefit from real-world application of skills.

Finally, to assess how well the project does in satisfying NSF's goals, we will track the number and quality of papers published based on FLOSSmole data and analyses, funded projects being supported or developed, and other measures of innovation in the community. Of particular interest will be the development of measures of broadening participation in the research area.

5. Conclusions

In this proposal, we develop a development and infrastructure plan for a collaborative data and analysis archive for the FLOSS research community. We propose methods for storing and maintaining this important data, as well as new techniques for unifying disparate data sources and specifying and defining reusable data-driven research workflows.

5.1 Expected intellectual merits

This project will contribute significantly to advancing knowledge and understanding within the FLOSS research community by enabling cooperation in data collection, aggregation and sharing, thus providing synergies to on-going and newly developed projects. The FLOSS research community is dedicated to understanding how FLOSS projects are developed and managed, how the software develops and how it is used. As FLOSS projects become more ubiquitous, quality data to describe and explain their successes becomes more important. Furthermore, research on FLOSS and its development processes can teach us about other areas of interest across the computing fields (Harrison, 2001), such as software evolution. As well, FLOSS development teams are potential training grounds for future software developers, making it important to understand how developers join and work in these teams. Finally, research on FLOSS development is relevant to CISE areas such as Human-Centered Computing and NSF-wide initiatives such as Cyberinfrastructure: FLOSS development provides numerous examples of successful computer-supported collaborative work.

The ideas presented in this proposal for improving and extending the already-useful FLOSSmole project are significant, practical, and achievable. The open source research community has strongly signaled its support for further development of a cooperative infrastructure such as the one we propose here. The community understands that our team is experienced and trusted (via the creation and maintenance of the precursor FLOSSmole project). For this project, we have assembled a team that has the vision, leadership, and talent to carry out the next generation of development work on this important community resource.

5.2 Expected broader impact

This project will benefit society by promoting collaboration and data sharing among research teams dedicated to understanding how FLOSS projects are developed, managed and sustained. As FLOSS

projects become more ubiquitous, describing them with quality data becomes more important. FLOSS is integral to today's Internet and is a foundation of tomorrow's innovation, thus this project will support research that should improve an important piece of collaborative research infrastructure used by academics in many disciplines, practitioners in the software industry and society in general.

To ensure that our project has a significant impact on the FLOSS research community, we have assembled an advisory board to guide us in building the most useful infrastructure possible. We have assembled an advisory board has a significant international composition, so we are hopeful that this collaboration will be a vehicle for expending the perspectives, knowledge, and skills of our team as a whole. Interactions with practitioners in industry as important as well; for example, we have already presented our work at the industry-oriented conferences and a good number of participants in our research community are from industry. We plan to continue this sort of interaction in the future.

Because one of the institutions collaborating on the proposal is an undergraduate institution, we have included with that proposal an RUI Impact Statement (see Supplementary Documentation), which covers in detail the broader impacts the proposal will have on teaching and learning. In brief, working on this project will provide a vehicle for efforts to improve our pedagogy, will positively impact the quality of courses for our undergraduate students, and will have beneficial results for our department and our undergraduate institution as a whole.

Finally, as an open source project itself, our community resource continues an important trend in scientific research toward opening and sharing data in order to promote collaboration, to reduce duplicative efforts, and to promote compatibility between research teams. Sharing code, data, schemas, queries, and experience promotes teaching and learning within the community.

5.3 Results from prior NSF funding

The co-PI for this grant, Kevin Crowston, has been funded by four NSF grants within the past 48 months. Three of these are related to the current proposal. The first is HSD 05-27457 ($684,882, 2005-2008, plus a $29,487 supplement for international fieldwork), "Investigating the Dynamics of Free/Libre Open Source Software Development Teams". This project was funded at the end of 2005 and work on it is now underway. The other two are IIS 04-14468 ($327,026, 2004-2006) and SGER IIS 03-41475 ($12,052, 2003-2004), both entitled "Effective work practices for Open Source Software development". These grants have provided support for travel to conferences (e.g., ApacheCon and OSCon) to observe, interview and seek support from developers and to present preliminary results, and for the purchase of data analysis software and equipment. This work has resulted in six journal papers (Crowston & Howison, 2005, 2006; Crowston et al., 2006a; Crowston & Scozzi, 2002; Crowston et al., In press; Howison et al., 2006a), multiple conference papers (e.g., Crowston et al., 2003; Crowston et al., 2005c; Crowston et al., 2005d; Crowston et al., 2006b; Howison et al., 2005; Howison et al., 2006b; Li et al., 2006) and workshop presentations (e.g., Conklin et al., 2005; Crowston et al., 2004a, 2004b; Crowston et al., 2005b; Crowston & Howison, 2003; Crowston & Scozzi, 2004; Howison & Crowston, 2004), with additional papers under review. These grants support a total of four PhD students; several others have been involved. The work supported by these three grants uses data from FLOSSMole and these grants have partially supported the initial pilot development on which the current grant will build. Crowston's fourth grant is IIS 04-14482 ($302,685, 2005-2006, with Barbara Kwasnik), for "How can document-genre metadata improve information-access for large digital collections?" The grant partially supported work on conference papers, a conference mini-track and journal special issue (Kwasnik & Crowston, 2005). Earlier work by the PIs on genre has appeared in journals (e.g., Crowston & Kwasnik, 2003) and conference papers (e.g., Kwasnik & Crowston, 2004). The grant funds two PhD students.

Collaborative Research:CRI:CRD: Data and analysis archive for research on Free and Open Source Software and its development

References Cited

Anderson, W. L. (2002). CODATA work in archiving scientific data. Information Services & Use, 22(2/3):63–68.

Anderson, W. L. (2004). Some challenges and issues in managing, and preserving access to, long-lived collections of digital scientific and technical data. Data Science Journal, 3:191–202.

Arent, J., & Nørbjerg, J. (2000). Software Process Improvement as Organizational Knowledge Creation: A Multiple Case Analysis. Paper presented at the Proceedings of the 33rd Hawaii International Conference on System Sciences (HICSS-33), Wailea, Maui, HI

Bessen, J. (2002). Open Source Software: Free Provision of Complex Public Goods: Research on Innovation.

Borgman, C. (2007). Scholarship in the Digital Age: Information, Infrastructure, and the Internet. MIT Press.

Christley, S., & Madey, G. (2005). An Algorithm for Temporal Analysis of Social Positions, NAACSOS2005. Notre Dame, IN.

Conklin, M., Howison, J., & Crowston, K. (2005). Collaboration Using OSSmole: A repository of FLOSS data and analyses, Symposium on Mining Software Repositories. St. Louis.

Cox, A. (1998). Cathedrals, Bazaars and the Town Council. Retrieved 22 March 2004, from http://slashdot.org/features/98/10/13/1423253.shtml

Crowston, K., Annabi, H., & Howison, J. (2003). Defining Open Source Software project success. Paper presented at the Proceedings of the 24th International Conference on Information Systems (ICIS 2003), Seattle, WA.

Crowston, K., Annabi, H., Howison, J., & Masango, C. (2004a). Effective work practices for Software Engineering: Free/Libre Open Source Software Development. Paper presented at the WISER Workshop on Interdisciplinary Software Engineering Research, SIGSOFT 2004/FSE-12 Conference,, Newport Beach, CA.

Crowston, K., Annabi, H., Howison, J., & Masango, C. (2004b). Towards a portfolio of FLOSS project success measures. Paper presented at the Workshop on Open Source Software Engineering, 26th International Conference on Software Engineering, Edinburgh.

Crowston, K., Annabi, H., Howison, J., & Masango, C. (2005a). Effective work practices for FLOSS development: A model and propositions. In Proceedings of the Hawai'i International Conference on System Science (HICSS). Big Island, Hawai'i.

Crowston, K., Heckman, R., Annabi, H., & Masango, C. (2005b). A structurational perspective on leadership in Free/Libre Open Source Software teams. Paper presented at the OSSCon, Genova, Italy.

Crowston, K., & Howison, J. (2003, 14 December). The social structure of Open Source Software development teams. Paper presented at the The IFIP 8.2 Working Group on Information Systems in Organizations Organizations and Society in Information Systems (OASIS) 2003 Workshop, Seattle, WA.

Crowston, K., & Howison, J. (2005). The social structure of free and open source software development. First Monday, 10(2).

Crowston, K., & Howison, J. (2006). Hierarchy and Centralization in Free and Open Source Software team communications. Knowledge, Technology & Policy, 18(4), 65–85.

Crowston, K., Howison, J., & Annabi, H. (2006a). Information systems success in Free and Open Source Software development: Theory and measures. Software Process—Improvement and Practice, 11(2), 123–148.

Crowston, K., Howison, J., Masango, C., & Eseryel, U. Y. (2005c). Face-to-face interactions in self-organizing distributed teams. Paper presented at the Academy of Management Conference, Honolulu, HI.

Crowston, K., & Kwasnik, B. H. (2003). Can document-genre metadata improve information access to large digital collections? Library Trends, 52(2), 345–361.

Crowston, K., & Scozzi, B. (2002). Open source software projects as virtual organizations: Competency rallying for software development. IEE Proceedings Software, 149(1), 3–17.

Crowston, K., & Scozzi, B. (2004). Coordination practices for bug fixing within FLOSS development teams Paper presented at the Presentation at 1st International Workshop on Computer Supported Activity Coordination, 6th International Conference on Enterprise Information Systems, Porto, Portugal.

Crowston, K., Wei, K., Li, Q., Eseryel, U. Y., & Howison, J. (2005d). Coordination of Free/Libre Open Source Software development. Paper presented at the International Conference on Information Systems (ICIS 2005), Las Vegas, NV, USA.

Crowston, K., Wei, K., Li, Q., Eseryel, U. Y., & Howison, J. (In press). Self-organization of teams in free/libre open source software development. Information and Software Technology Journal, Special issue on Understanding the Social Side of Software Engineering, Qualitative Software Engineering Research, Accepted with major revisions.

Crowston, K., Wei, K., Li, Q., & Howison, J. (2006b). Core and periphery in Free/Libre and Open Source software team communications. Paper presented at the Hawai'i International Conference on System System (HICSS-39), Kaua'i, Hawai'i.

Cukic, B. (2005). The Promise of Public Software Engineering Data Repositories. IEEE Software, 22(6), 20–22.

Di Bona, C., Ockman, S., & Stone, M. (Eds.). (1999). Open Sources: Voices from the Open Source Revolution. Sebastopol, CA: O'Reilly & Associates.

Dionisio, John David N (2006) Recourse. Funded by NSF Grant #511732 "Cultivating an Open Source Software Culture Among Computer Science Undergraduate Students". Available at http://recourse.cs.lmu.edu/ Assessed Nov 14 2006.

DOI Foundation (2005) New DOI Registration Agency for scientific data appointed: 1,500,000 datasets to be registered by end of 2005, Press Release, http://www.doi.org/news/TIBNews-050405.html Retrieved 14 November 2006.

Finholt, T. A., & Olson, G. M. (1997). From laboratories to collaboratories: A new organizational form for scientific collaboration. Psychological Science, 8(1), 28-35.

Franck, E., & Jungwirth, C. (2002). Reconciling investors and donators: The governance structure of open source (Working Paper No. No. 8): Lehrstuhl für Unternehmensführung und -politik, Universität Zürich.

Gacek, C., & Arief, B. (2004). The many meanings of Open Source. IEEE Software, 21(1), 34–40.

Gallivan, M. J. (2001). Striking a balance between trust and control in a virtual organization: A content analysis of open source software case studies. Information Systems Journal, 11(4), 277–304.

German, D., & Mockus, A. (2003). Automating the Measurement of Open Source Projects, Proceedings of the ICSE 3rd Workshop on Open Source.

German, D. M. (2003). The GNOME project: A case study of open source, global software development. Software Process: Improvement and Practice, 8(4), 201–215.

Ghosh, R. A. (2002, 14 October). Free/Libre and Open Source Software: Survey and Study. Report of the FLOSS Workshop on Advancing the Research Agenda on Free / Open Source Software. Retrieved 16 March, 2006, from http://www.infonomics.nl/FLOSS/report/workshopreport.htm

González-Barahona, J. M., & Robles, G. (2004). Community structure of modules in the Apache project, Proceedings of the ICSE 4th Workshop on Open Source (Vol. 4, pp. 49–54).

Hahsler, M., & Koch, S. (2005). Discussion of a Large-Scale Open Source Data Collection Methodology, Proceedings of the 38th Annual Hawaii International Conference on System Sciences (HICSS 2005). Big Island, HI.

Hallen, J., Hammarqvist, A., Juhlin, F., & Chrigstrom, A. (1999). Linux in the workplace. IEEE Software, 16(1), 52–57.

Hann, I.-H., Roberts, J., Slaughter, S., & Fielding, R. (2002). Economic incentives for participating in open source software projects. In Proceedings of the Twenty-Third International Conference on Information Systems (pp. 365–372).

Harrison, W. (2001). Editorial: Open Source and Empirical Software Engineering. Empirical Software Engineering, 6(3), 193-194.

Hertel, G., Niedner, S., & Herrmann, S. (2003). Motivation of Software Developers in Open Source Projects: An Internet-based Survey of Contributors to the Linux Kernel. Research Policy, 32(7), 1159–1177.

Howison, J., Conklin, M., & Crowston, K. (2006a). FLOSSmole: A collaborative repository for FLOSS research data and analyses. International Journal of Information Technology and Web Engineering, 1(3), 17–26.

Howison, J., Conklin, M. S., & Crowston, K. (2005, 11–14 July). OSSmole: A collaborative repository for FLOSS research data and analyses. Paper presented at the 1st International Conference on Open Source Software, Genova, Italy.

Howison, J., & Crowston, K. (2004). The perils and pitfalls of mining SourceForge. Paper presented at the Presentation at the Workshop on Mining Software Repositories, 26th International Conference on Software Engineering, Edinburgh, Scotland.

Howison, J., Inoue, K., & Crowston, K. (2006b). Social dynamics of FLOSS team communications. Paper presented at the The Second International Conference on Open Source Systems, Como, Italy.

Kogut, B., & Metiu, A. (2001). Open-source software development and distributed innovation. Oxford Review of Economic Policy, 17(2), 248–264.

Kraut, R. E., & Streeter, L. A. (1995). Coordination in software development. Communications of the ACM, 38(3), 69–81.

Kwasnik, B. H., & Crowston, K. (2004). A framework for creating a facetted classification for genres: Addressing issues of multidimensionality. In Proceedings of the Hawai'i International Conference on System Science (HICSS). Big Island, Hawai'i.

Kwasnik, B. H., & Crowston, K. (2005). Genres of digital documents: Introduction to the special issue. Information, Technology & People, 18(2), 76–88.

Lakhani, K. R., & Wolf, B. (2003). Why Hackers Do What They Do: Understanding Motivation and Effort in Free/Open Source Software Projects. Retrieved 1 March, 2005, from http://opensource.mit.edu/papers/lakhaniwolf.pdf

Lee, G. K., & Cole, R. E. (2003). From a firm-based to a community-based model of knowledge creation: The case of Linux kernel development. Organization Science, 14(6), 633–649.

Leibovitch, E. (1999). The business case for Linux. IEEE Software, 16(1), 40–44.

Lerner, J., & Tirole, J. (2001). The open source movement: Key research questions. European Economic Review, 45, 819–826.

Lerner, J., & Tirole, J. (2002). Some simple economics of Open Source. The Journal of Industrial Economics, 2(1), 197–234.

Li, Q., Crowston, K., Heckman, R., & Howison, J. (2006). Language and power in self-organizing distributed teams. Paper presented at the OCIS Division, Academy of Management Conference, Atlanta, GA.

Ljungberg, J. (2000). Open Source Movements as a Model for Organizing. European Journal of Information Systems, 9(4).

Markus, M. L., Manville, B., & Agres, E. C. (2000). What makes a virtual organization work? Sloan Management Review, 42(1), 13–26.

Massey, B. (2005). Longitudinal Analysis of Long-Timescale Open Source Repository Data, Proceedings of the 2005 PROMISE Workshop at ICSE.

Moody, G. (2001). Rebel code—Inside Linux and the open source movement. Cambridge, MA: Perseus Publishing.

Moon, J. Y., & Sproull, L. (2000). Essence of distributed work: The case of Linux kernel. First Monday, 5(11).

Newman, D. J., Hettich, S., Blake, C. L., & Merz, C. J. (1998). UCI repository of machine learning databases. from http://www.ics.uci.edu/~mlearn/MLRepository.html

O'Reilly, T. (1999). Lessons from open source software development. Communications of the ACM, 42(4), 33–37.

Paulson, J. W., Succi, G., & Eberlein, A. (2004). An empirical study of open-source and closed-source software products. IEEE Transactions on Software Engineering, 30(4), 246–256.

Pfaff, B. (1998). Society and open source: Why open source software is better for society than proprietary closed source software. from http://www.msu.edu/user/pfaffben/writings/anp/oss-is-better.html

Prasad, G. C. (n.d.). A hard look at Linux's claimed strengths…. From http://www.osopinion.com/Opinions/GaneshCPrasad/GaneshCPrasad2-2.html

Raymond, E. S. (1998). The cathedral and the bazaar. First Monday, 3(3).

Robles, G., Amor, J. J., & González-Barahona, J. M. (2005). Evolution and growth in large libre software projects, The 8th International Workshop on Principles of Software Evolution. Lisbon, Portugal.

Robles, G., Koch, S., & González-Barahona, J. M. (2004). Remote analysis and measurement of libre software systems by means of the CVSanaly tool, Proceedings of the 2nd ICSE Workshop on Remote Analysis and Measurement of Software Systems (RAMSS), 26th International Conference on Software Engineering. Edinburgh, Scotland.

Robles-Martínez, G., González-Barahona, J. M., Centeno-González, J., Matellán-Olivera, V., & Rodero-Merino, L. (2003). Studying the evolution of libre software projects using publicly available data, Proceedings of the ICSE 3rd Workshop on Open Source.

Sayyad Shirabad, J., & Menzies, T. J. (2005). The PROMISE Repository of Software Engineering Databases. from http://promise.site.uottawa.ca/SERepository

Scacchi, W. (2002). Understanding the requirements for developing Open Source Software systems. IEE Proceedings Software, 149(1), 24–39.

Schach, S. R., & Offutt, J. (2002). On the Nonmaintainability of Open-Source Software. Paper presented at the Proceedings of the ICSE 2nd Workshop on Open Source.

Senyard, A., & Michlmayr, M. (2004). How to Have a Successful Free Software Project. In Proceedings of the 11th Asia-Pacific Software Engineering Conference (pp. 84–91). Busan, Korea: IEEE Computer Society.

Shepard, T., Lamb, M., & Kelly, D. (2001). More testing should be taught. Communication of the ACM, 44(6), 103–108.

Smith, N., Capiluppi, A., & Ramil, J. F. (2005). A Study of Open Source Software Evolution Data using Qualitative Simulation. Software Process: Improvement and Practice, 10, 287–300.

Stamelos, I., Angelis, L., Oikonomou, A., & Bleris, G. L. (2002). Code quality analysis in open source software development. Information Systems Journal, 12(1), 43–60.

Stewart, K. J., & Ammeter, T. (2002). An exploratory study of factors influencing the level of vitality and popularity of open source projects. In Proceedings of the Twenty-Third International Conference on Information Systems (pp. 853–857).

Stewart, K. J., & Gosain, S. (2001). Impacts of ideology, trust, and communication on effectivness in open source software development teams. Paper presented at the Twenty-Second International Conference on Information Systems, New Orleans, LA.

Valloppillil, V. (1998). Halloween I: Open Source Software. from http://www.opensource.org/halloween/halloween1.html

Valloppillil, V., & Cohen, J. (1998). Halloween II: Linux OS Competitive Analysis. From http://www.opensource.org/halloween/halloween2.html

Vixie, P. (1999). Software engineering. In C. Di Bona, S. Ockman & M. Stone (Eds.), Open sources: Voices from the open source revolution. San Francisco: O'Reilly.

Weiss, D. A Large Crawl and Quantitative Analysis of Open Source Projects Hosted on SourceForge. Poznan, Poland: Institute of Computing Science, Poznan University of Technology.

Woodyard, D. (2002). Metadata and preservation. Information Services & Use, 22(2/3):121–126.

Xu, J., Gao, Y., Christley, S., & Madey, G. (2005). A Topological Analysis of the Open Souce Software Development Community, Proceedings of the 38th Annual Hawaii International Conference on System Sciences (HICSS 2005). Big Island, HI.