On the Shoulders of Giants

Earl Barr
Dept. of Computer Science
University of California, Davis
etbarr@ucdavis.edu

Christian Bird
Dept. of Computer Science
University of California, Davis
cabird@ucdavis.edu

Eric Hyatt
Hyatt Howell, Ltd.
Berkeley, CA
hyatt@ncal.net

Tim Menzies
Lane Dept. of CS & EE
West Virginia University
tim@menzies.us

Gregorio Robles
GSyC/LibreSoft
Universidad Rey Juan Carlos
grex@gsyc.urjc.es

ABSTRACT

Science rests on peer review and the wide-spread dissemination of knowledge. Software engineering research will advance further and faster if the sharing of data and tools were easier and more wide-spread. Pragmatic concerns hinder the realization of this ideal: the time and effort required and the risk of being scooped. We examine the costs and benefits of facilitating sharing in our field in an effort to help the community understand what problems exist and find a solution. We examine how other fields, such as medicine and physics, handle sharing, describe the value of sharing for replication and innovation, and address practical concerns such as standards and warehousing. To launch what we hope will become an ongoing discussion of solutions in our community, we present some ways forward that mitigate the risk of sharing — partial sharing, registry, escrow, and market.

Categories and Subject Descriptors

I.0 [**Computing Methodologies**]: General; K.7 [**The Computing Profession**]: General

General Terms

Experimentation, Standardization

Keywords

Replication, Data Sharing

1. INTRODUCTION

Sharing is fundamental to science; in-between paradigm shifts, science builds on the work of others. Bernard Chartres, a twelfth century scholar, wrote "We are like dwarfs standing upon the shoulders of giants, and so able to see more and see farther than the ancients." Inspiring our title, this quote captures the benefits of sharing, which include tool/data reuse, replication, and the improved quality that accompanies the possibility of increased scrutiny.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
FoSER 2010, November 7–8, 2010, Santa Fe, New Mexico, USA.
Copyright 2010 ACM 978-1-4503-0427-6/10/11 ...$10.00.

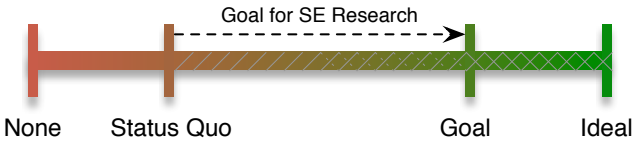

Figure 1: Data Sharing Continuum

Norms in the software engineering community already require a high-level description of tools, methodology and processed (summary) data[1]. From these requirements, differentiated replication and clean-room tool re-implementation are already possible. Thus, the question is not whether or not to share, since we already share. The question we address is whether, and to what extent, to facilitate sharing, as shown in Figure 1. The problem with the status quo is that it hinders progress by necessitating redundant work.

One of us, Gregorio Robles, recently quantified the extent of this redundant, and, in some cases, potentially unreproducible work [18]; he attempted to replicate research published in the working conference on Mining Software Repositories (MSR) over its lifetime and found that only 2, of the 154 experimental studies published, provided the data and the tools required for replication and further research. Robles proposed the adoption of a new convention, the inclusion of a "Barriers to Replication" section, which contains links to raw and processed data and tools, including source, and is analogous to the now ubiquitous "Threats to Validity".

Robles's proposal met some resistance, following his presentation at MSR 2010. A number of concerns were voiced in the ensuing discussion — the low-yield work to clean and package data, the distraction of tool support, and the danger of providing another research team with data or tools that they can use in the race to the next publication. These concerns boil down to cost and risk aversion. In the face of a deadline, inessential work is often left undone; it costs time and money to clean up data or hammer tools into shape. However, we are all better off if this price is paid. The PROMISE [5] and SIRS [9] artifact and data repositories are existence proofs that it *is* possible to pay the price and attest its benefits. The danger of racing another team to the next publication is a harder problem. Considerable time and effort is spent building tools and collecting data. Researchers need to recoup that investment: For every year of data collection and cleaning or tool design and fashioning, we believe that a researcher needs roughly two publications. Although the risk of being scooped may be small, some researchers are not willing to take it.

[1]We focus on data and tools unencumbered by intellectual property constraints until Section 5.

Weighed against these objections, facilitating sharing carries a number of benefits. The first is increased confidence in our results. Exact replications give us confidence in results while differentiated replications help us understand how well the results generalize to other contexts. Increased sharing of both data and tools make such replications easier and less prone to error. Replication is at the heart of the scientific method; replicated experiments are the gold standard for empirical truth and the source of our confidence in an empirical claim. Software engineering is an increasingly important arena of economic activity; momentous decisions are made based on our results [15]. It behooves us to make our results as solid as we can.

As in the fields of education and sociology, data gathered by a researcher to answer one question can later be analyzed by another researcher to answer different questions. This encourages newcomers, as researchers entering a field can leverage existing tools and begin working on their own innovations rather than re-implementing existing work. In other fields, not only is data shared but practitioners must use that data, *i.e.* the current findings in their field, to justify their methods and techniques. David Budgen documents this "evidence-based" reasoning in medicine [7]. He wishes our own field would adopt "evidence-based software engineering", but acknowledged that this requires changes to how we conduct our research [8]. These changes include deciding what to share and how. Specific standards and data formats aid the user of that data and those tools, but burden the sharing researcher. Finding the sweet spot between these concerns is a new research direction that we elaborate in Section 4.

What incents an author to shoulder the burden of increased sharing? In the long run, the criteria for paper acceptance could include meeting the new sharing standards. Then sharing data and tools would be a requirement for publishing papers: something for which researchers already have ample incentive. The transition from current practice to this state of affairs would be painful. To ease this transition, more venues could reward researchers who share, as MSR did last year when allocating an additional page to data sharers.

We claim that improving the sharing of data and tools in our field will accelerate innovation and dramatically impact the future of software engineering research; it will

1. Reduce redundant effort;
2. Attract researchers by lowering barriers to entry; and
3. Facilitate the replication of important findings.

The question of increased sharing is important and, therefore, fraught. In the *Editor's Letter* of February's *Communications of the ACM* [20], Moshe Vardi called for more debate, stating "Vigorous debate, I believe, exposes all sides of an issue — their strengths and weaknesses." In this spirit, we attempt to dispassionately describe both sides of the debate and propose some ways forward: partial sharing options and three sharing mechanisms — registry, escrow and market. These proposals address the concerns of those reluctant to make their data and tools available, while, at the same time, improving sharing, replication, and reuse in our field.

2. THE STATUS QUO

Sharing already occurs in our field. When the review process works, papers are rejected if they do not describe their methodology, present processed data and its sources, and explain the algorithms their tools realize. When these descriptions are scant, scientific norms dictate replying to requests for elaborations of methodology used, raw data, and algorithmic details. Thus, new uses of the data or tool and even replication are already possible; they entail independently collecting the requisite data and re-implementing the tools. This redundant work could be avoided if the original researchers were more open with their work. Currently, research teams appear to believe that the costs of increased sharing outweigh its benefits.

Pragmatically, just because one has gathered data and developed tools does not mean that it is easy to share them. Data may be stored in custom formats or in databases with opaque schemas. Similarly, tools are usually research prototypes that often suffer from poor code quality, can require specialized environments to build and run, may need special input formats for data, and may generate output that is difficult to understand. Their implementation may be sound and rest on solid principles, but may be far from fit for public consumption. If asked to share data or tools, a researcher's pride in his craftsmanship impels him to rework their tools into a form that is understandable, usable and *elegant*, at considerable cost.

The time and effort put into gathering data and implementing techniques into tools represent a considerable investment on the part of a researcher. It seems unreasonable to spend a year mining and massaging data or developing and fixing a tool and then immediately release them to the public after a single publication. Publications are the currency of our field. When data is hard won, one wants to reap the benefits of its collection rather than provide the fuel for others' publications. Some data sets are rich enough to generate multiple distinct papers, and, in fact, some Ph.D.'s have rested on a single tool or set of data. With graduate careers and tenure on the line, one can reasonably ask, "Why should I share my work with others when I haven't yet fully benefited from it myself?" It is much easier to move on to the next research project or add a new feature to the current prototype than spend time and effort that appears to only serve others' interests. Ours is not the only field to grapple with such issues. Although data sharing is the norm in their field, astrophysicists are currently debating how long to wait before making recently obtained, valuable data about potential planets publicly available [16].

In contrast to other disciplines, our field lacks well-accepted, well-defined response variables. For instance, humans have evolved little since the invention of ECG machines, so their output can be interpreted similarly for any human. Software, on the other hand, is evolving at a breakneck pace (witness the recent rise of the mobile phone "apps"); it is not clear how to repeatably measure a phenomenon in different software systems. Combining this measurement problem and the fact that controlled experiment is rare in our field, some contend that exact replication is impossible and, even when differentiated replication is possible, it is expensive [13]. For these proponents of the status quo, increased sharing is more likely to increase racing to results than lead to better, verified results.

3. FACILITATING SHARING

While it is true that sharing already occurs, it imposes wasteful, redundant work on researchers. Facilitating sharing will encourage reuse, lower barriers to entry, improve the quality of tools and data, enhance technology transfer, and give our community, and industry, more confidence in our results. In this section, we address each point made in the previous section, except one: the problem of recovering one's investment in data or tools. Section 4 describes our proposals for mitigating this risk.

Scientific norms require researchers to respond to requests for clarification about their published work. Unfortunately, norms are not laws and prescribe rather than enforce behavior. In a 2006 study of data witholding [22], approximately half of 1,077 doctoral students and postdoctoral fellows indicated that data withholding

had had a negative effect on the progress of their research. Of the 679 in chemical engineering and computer science, 139 (23.0%) reported that they had asked for and been denied access to data, materials, or programming associated with published research.

Although some projects have shared their data or tools in an easily digestible and distributable form, this situation is rare. Often requests are met with some form of "the software is a prototype and not ready yet"[2]. In one case, we have repeatedly requested the tool and, each time, have been rebuffed with this response. This response is ironic in a community whose very topic of research is the creation of software; *we* should know that software rarely ever reaches a level of quality with which its developers are completely satisfied. When a prototype has been shared with us, we have had to adapt it for our own purposes, but with much less effort than re-implementing the software on our own.

Some research areas require considerable investment in tools or data collection to enter; lack of sharing exacerbates the problem. A new player would need to re-implement the tools or redundantly collect the data, neither of which has any real research value, and, in doing so, waste time and fall behind. This barrier stymies even a researcher with good ideas that no one else is pursuing. In fact, it may be easier to get a post-doc with a lab in the area than to re-implement their tools. This slows down research and innovation [22].

While preparing data and tools for sharing may appear to benefit only others, it can actually directly benefit a research team. A study in computational science by the WAVE lab at Stanford found they were unable to replicate their own results due to a lack of coding standards for reproducibility [6]. If code is made public, it is much less likely to be lost and the correct version is easier to find.

Sharing is indicative of the maturity and impact of a research field. Piwowar and Chapman [17] recently found that journals with stronger data sharing policies have higher median impact factors and that more academic journals have data sharing policies (82%) than commercial journals (46%). The NIH requires all grant proposals exceeding $500,000 per year address data sharing in their applications [2]. Nature requires the disclosure of any restrictions on the sharing of data and materials; its editorial principle is "An inherent principle of publication is that others should be able to replicate and build upon the authors' published claims" [3]. The United States National Science Foundation (NSF) announced that, by October 2010, all proposals must include a data management plan [1]. The NSF's Ed Seidel stated that "openly sharing data will pave the way for researchers to communicate and collaborate more effectively... It will address the need for data from publicly-funded research to be made public."

Replication, the ability to reach the same scientific conclusion, separates the sciences, including computer science, from stage magic. When distinct experiments disagree, scientists employ replication. Facilitating sharing aids replication, which, in turn, increases confidence in our results. Conflicts do arise. In astronomy, there was a longstanding disagreement about the value of H_0, the Hubble constant. In 1975, vandenBurgh reported $H_0 = 95 + 15 - 12$ km/s/Mpc and Sandage reported $H_0 = 55 \pm 5$ km/s/Mpc [19]. Careful replication that deconstructed the data and reasoning behind these two results tackled this discrepancy and motivated the design of new experiments and new instruments to collect more accurate data. In 2001, the Hubble telescope made a measurement that seems to have resolved the conflict, reporting $H_0 = 72 \pm 8$ km/s/Mpc. The ugly extreme of conflict is scientific fraud. In physics, data sharing led to the discovery of scientific fraud, notably in the Ninov and Schön affairs.

[2]Indeed, the next step in Robles' study would be to contact the authors and report the results.

The Stanford WAVE lab sums up our belief in importance of sharing [6]: *"An article about computational science in a scientific publication is* not *the scholarship itself, it is merely* advertising *of the scholarship. The actual scholarship is the complete software development environment and the complete set of instructions which generated the figures."*

4. WAYS FORWARD

A classic incentive compatibility problem underlies the status quo. The cost of sharing to society is low and the value is potentially high, in terms of verified results and increased innovation. To a research team, in contrast, the cost is high — additional work and the potential of lost publications — and the value, mainly citations, is low. In this section, we outline some ways forward toward our goal of increased sharing. First, we discuss what research material to share, then three different mechanisms for sharing it, before closing with where to host it.

The sharing of research material can be complete or partial. For a tool, partial sharing could be publishing its binary, not its source. Access to the binary could ease reviewing, since reviewers could selectively verify some of a researcher's claims, and facilitate comparative studies, such as demonstrating that a proposed new technique is superior to its predecessors. Some researchers might wish to commercialize their tools (e.g. Coverity's prevent [4]). Sharing only the binary addresses this concern, as well as the reluctance to share when the source does not meet its creator's pride in his craftsmanship. Hybrid solutions include providing selected portions of a tool's source along with its binary. For data, partial sharing could allow others to analyze the data without directly accessing it. A researcher could be allowed to upload analysis code, which the data owner could inspect to see if it is a data farmer. For example, an R script containing a logistic predictive model could analyze data and return only summary results (*e.g.* model coefficients, precision, recall, or correlation values) used to validate a hypothesis. The inference problem, inferring sensitive data from insensitive data, rears its head here [10], but its solution or mitigation in our application itself presents interesting directions for research.

Whatever our community requires researchers to share, we also need sharing mechanisms. We introduce three below.

Registry

One way to improve sharing is to record requests and responses in a public registry. A potential consumer would register a request for a tool or data by identifying a paper and its authors, who would be notified via email. The grantee updates the server when he fulfills the request. This creates a public record of which requests are deferred, granted, and denied. The registry could also contain information such as a researcher's reasons for not sharing or the requestor's feedback. This registry would put the originating researchers' reputation at stake. The community would notice a researcher who ignores many requests as well as one who consistently responds in a timely manner. We note that privacy may be an issue here.

Escrow

Alternately, our community could escrow research materials. Upon acceptance of a paper for publication in a journal or conference, the data and tools gathered, used, or developed during the research must, as a condition for acceptance, be submitted along with the final manuscript. These materials would be held in escrow for a period of time, and then made available to the public for use in replication and as a foundation for further research. While this proposal does not directly benefit the original researcher, it protects her from races, reducing her risk. More importantly, it captures

the external benefit of improving the quality of data and tools and ensuring their eventual release. Of course, this proposal does not relieve a researcher from the obligation to respond to the queries of other researchers, before or after the public release of the data.

Using escrow is not without precedent. In astrophysics, it is standard practice for an astronomer to have exclusive rights to their new observations for six months, after which time they revert to the public domain. While it requires *'the timely release'* of data, the *"NIH continues to expect that the initial investigators may benefit from first and continuing use but not from prolonged exclusive use."* [2]. In medicine, initial paper submission to final publication in a research journal can take years. Thus, this data sharing policy gives researchers exclusive use of their data for a substantial period of time. Publication time frames in software engineering are markedly shorter. Determining exactly what these time frames should be is a question for the larger community, and may require some experimentation and fine-tuning; we believe that three years is a reasonable starting point.

Market

The risk of the premature release of data or tools can be expressed in terms of money, since jobs and promotions are ultimately at stake. Thus, another way to defray this risk is to create a market and charge for the release of data or tools. To protect their investment, a researcher could initially charge a high price, then drop that price as their interest in the research materials wanes. The obvious objection is price discovery. How do we assign a price to data or a tool? What is the value of a first tier publication? We do not pretend to know the answers to these questions, but we do think that our community can answer them. These questions seem easier to answer than pricing a human life, which economists have already done [21]. To bootstrap the market, granting institutions could require the release of data or tools and set the initial price. A market-based approach generalizes our escrow proposal. Under escrow the research materials are initially private to all then public to all; it is a market where the price is initially infinite, then drops to zero.

Standards and Hosting

At MSR 2010, exporting tools in a self-contained virtual machine was proposed to eliminate dependency problems and to ease building and running a tool. Existing research into the definition of formats that combine data, analysis algorithms and write-ups include Compendiums [11] and Sweave [12]. The adoption of such a bundling convention would have saved a project that we abandoned after failing to overcome dependency and environment issues in building a tool written in Modula-3. While such standards are important, they are not *required* for sharing. Witness the PROMISE data repository [5], which accepts data in any format, thus lightening the burden on the original researcher at the cost of more work on the part of the data consumer. Finding the sweet spot in apportioning this burden combines technical issues, such as defining schema and metadata, with sociological concerns and is an avenue for future research.

If researchers are willing to share their research material, we must decide where. In his study of replication, Robles [18] found that many URLs for tools and data mentioned in papers were dead. Setting up and maintaining a warehouse for data and tools can be cumbersome and time-consuming. Although specialized, the PROMISE repository demonstrates that hosting *can* be provided at reasonable cost (only an hour or so maintenance per month). Nonetheless, hosting is not free. Parties who might be able and willing to provide this service to the community include 1) publishers, such as the ACM and IEEE, 2) funding agencies, like the NSF, and 3) industrial players, possibly Google[3] or a consortium. As research is a living thing, regardless of who hosts it, researchers should be able to update their data as it is cleaned or flaws are found and tools as bugs are fixed or features added.

5. INTELLECTUAL PROPERTY

In this paper, we focus on data and tools that researchers *could* share if they so desired. A portion of software engineering research is conducted on proprietary data that commercial entities are unwilling to share due to competitive concerns. If sharing data were made a requirement for research publication, such entities would probably publish less empirical research. Faced with a choice between published research under the sharing rules of the status quo and no research at all, we prefer published research. We acknowledge that, if our escrow proposal were accepted, we are essentially advocating an exception for research that depends on intellectual property, which effectively raises the cost for academic publication without a commensurate raise in the cost of industrial publication, where the two compete.

Data sanitization may partially solve this conundrum. Recently, in an effort to share data with the research community and help drive research on collaborative development, IBM released an archive of software artifacts [24] from the development of JAZZ, a software lifecycle tool. IBM indicated to us that one employee spent approximately one week sanitizing the data. A number of top tier research publications have resulted from the sharing of this data, *e.g.* [23].

6. FUTURE WORK AND CONCLUSION

Software engineering is an increasingly important economic arena. The results of our work already weigh heavily on how businesses and thus society allocates resources. We have argued that the self-discipline encouraged by increased sharing of tools and data will facilitate and improve research. If research is easier to replicate, it will be easier for techniques discovered in academia to be adopted outside of the research community. Sharing in academia can thus increase the value of research to society at large.

Research material could be shared free-form or constrained by standards. The former aids the sharer; the latter the consumer. Research into balancing their competing interests would benefit our community and industry. As sharing becomes more commonplace, we will be in the position of having a multitude of possible techniques for independently validating and verifying (V&V) results with shared data, but no clear guidance on when to use one over another. One of us, Tim Menzies *et al.* [14] searched the literature for any study comparatively evaluating V&V techniques in the current IEEE standard V&V without success. Research into this area will help our field converge towards a standard practice for replication.

Acknowledgements. This work was partially funded by NSF projects CCF-0810879 and SOD-IIS-0613949 This material is based in part upon work supported by the U.S. Department of Homeland Security under Grant Award Number 2006-CS-001-000001, under the auspices of the Institute for Information Infrastructure Protection (I3P) research program. The I3P is managed by Dartmouth College. The views and conclusions contained in this document are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of the U.S. Department of Homeland Security, the I3P, or Dartmouth College.

[3]Google already stores, indexes, and republishes academic data; Google Scholar's slogan is, in fact, "Stand on the shoulders of giants," so hosting data is in line with their stated purpose.

7. REFERENCES

[1] Editorial Policies: Availability of data and materials, 2003. http://www.nature.com/authors/editorial_policies/availability.html.

[2] Final NIH Statement on Sharing Research Data, 2003. http://grants.nih.gov/grants/guide/notice-files/NOT-OD-03-032.html.

[3] Guide to Publication Policies of the Nature Journals, 2009. http://www.nature.com/authors/gta.pdf.

[4] A. Bessey, K. Block, B. Chelf, A. Chou, B. Fulton, S. Hallem, C. Henri-Gros, A. Kamsky, S. McPeak, and D. Engler. A few billion lines of code later: using static analysis to find bugs in the real world. *Communnications of the ACM*, 53(2):66–75, 2010.

[5] G. Boetticher, T. Menzies, and T. Ostrand. PROMISE Repository of empirical software engineering data. West Virginia University, Department of Computer Science, 2007. http://promisedata.org/.

[6] J. Buckheit and D. Donoho. Wavelab and reproducible research. *Lecture Notes on Statistics*, 1995.

[7] D. Budgen, 2006. Keynote address, Conference on Software Engineering Education and Training.

[8] D. Budgen, B. Kitchenham, and P. Brereton. Is evidence-based software engineering mature enough for practice & policy? In *Proceedings of the Software Engineering Workshop*, 2009.

[9] H. Do, S. Elbaum, and G. Rothermel. Supporting controlled experimentation with testing techniques: An infrastructure and its potential impact. *Empirical Software Engineering*, 10(4):405–435, 2005.

[10] C. Farkas and S. Jajodia. The inference problem: a survey. *SIGKDD Explorations Newsletter*, 4(2):6–11, 2002.

[11] R. Gentleman and D. Temple Lang. Statistical analyses and reproducible research. *Journal of Computational and Graphical Statistics*, 16(1):1–23, March 2007.

[12] F. Leisch. Sweave: Dynamic generation of statistical reports using literate data analysis. In W. Härdle and B. Rönz, editors, *Proceedings in Computational Statistics (2002)*, pages 575–580. Physica Verlag, Heidelberg, 2002.

[13] J. Lung, J. Aranda, S. M. Easterbrook, and G. V. Wilson. On the difficulty of replicating human subjects studies in software engineering. In *Proceedings of the International Conference on Software Engineering*, 2008.

[14] T. Menzies, M. Benson, K. Costello, C. Moats, M. Northey, and J. Richardson. Learning better IV&V practices. *Innovations in Systems and Software Engineering*, 4(2):169–183, 2008.

[15] L. Osterweil, L. Clarke, M. Evangelist, J. Kramer, D. Rombach, and A. Wolf. The impact project: determining the impact of software engineering research upon practice. *ACM SIGSOFT Software Engineering Notes*, 25(6):109, 2000.

[16] D. Overbye. In the hunt for planets, who owns the data? New York Times, http://www.nytimes.com/2010/06/15/science/space/15kepler.html, June 14, 2010.

[17] H. A. Piwowar and W. W. Chapman. A review of journal policies for sharing research data. In *Proceedings of the 12th International Conference on Electronic Publishing*, 2008.

[18] G. Robles. Replicating MSR: A Study of the Potential Replicability of Papers Published in the Mining Software Repositories Proceedings. In *Proceedings of the Working Conference on Mining Software Repositories*, 2010.

[19] A. Sandage, M. Sandage, and J. Kristian, editors. *Galaxies and the Universe*, volume IX. University of Chicago Press, 1975.

[20] M. Y. Vardi. More debate, please! *Communications of the ACM*, 53(1):5–5, 2010.

[21] W. Viscusi and J. Aldy. The value of a statistical life: a critical review of market estimates throughout the world. *Journal of Risk and Uncertainty*, 27(1):5–76, 2003.

[22] C. Vogeli, R. Yucel, E. Bendavid, L. Jones, M. Anderson, K. Louis, and E. Campbell. Data withholding and the next generation of scientists: results of a national survey. *Academic Medicine*, 81(2):128–36, 2006.

[23] T. Wolf, A. Schroter, D. Damian, and T. Nguyen. Predicting build failures using social network analysis on developer communication. In *Proceedings of the International Conference on Software Engineering*, 2009.

[24] A. Ying, K. Ehrlich, H. Li-Te Cheng, T. Frauenhofer, and F. van Ham. Jazz development data: a community perspective. In *Proceedings of International Workshop on Infrastructure for Research in Collaborative Software Engineering*, 2008.