
The Trouble With Social Computing Systems Research

Michael S. Bernstein

MIT CSAIL
Cambridge, MA
msbernst@mit.edu

Mark S. Ackerman °

University of Michigan
Ann Arbor, MI
ackerm@umich.edu

Ed H. Chi °

Palo Alto Research Center
Palo Alto, CA
echi@parc.com

Robert C. Miller °

MIT CSAIL
Cambridge, MA
rcm@mit.edu

° These authors contributed equally to the paper and are listed alphabetically.

Copyright is held by the author/owner(s).
CHI 2011, May 7–12, 2011, Vancouver, BC, Canada.
ACM 978-1-4503-0268-5/11/05.

Abstract

Social computing has led to an explosion of research in understanding users, and has the potential to similarly revolutionize systems research. However, the number of papers designing and building new sociotechnical systems has not kept pace. In this paper we analyze the reasons for this disparity, ranging from misaligned methodological incentives, evaluation expectations and research relevance compared to industry. We suggest improvements for the community to consider and evolve so that we can chart the future of our field.

General Terms

Social computing, systems research, evaluation

Introduction

The rise of social computing is impacting SIGCHI research immensely. Wikipedia, Twitter, Delicious, Facebook and Mechanical Turk have all led to exciting work understanding people and their interactions by using large, naturally occurring datasets. These results are just the tip of the iceberg.

Those invested in the *systems* research community in social computing hope for a similar trajectory of novel, impactful sociotechnical systems. By systems research, we mean *research whose main contribution is the presentation of a new sociotechnical artifact, design, or*

platform. Traditional CSCW research had no shortage of systems research, especially focusing on distributed teams and collaboration [1][17][30]. In some ways, systems research has already moved forward: we have dropped our assumptions of single-display, knowledge-work-focused, isolated users [10]. This broader focus, married with a massive growth in platforms, APIs, and interest in social computing, would suggest that we should see lots of new interesting research systems.

Unfortunately, the evidence suggests otherwise. Consider submissions to the Interaction Beyond the Individual track at CHI 2011. Papers submitted to this track that chose “Understanding Users” as a primary contribution outnumbered those that selected “Systems, Tools, Architectures and Infrastructure” by a ratio of four to one this year [26]. It is possible that this number reflects overall submission ratios to CHI. But if not, either the systems papers are being diverted to other tracks and venues, or there is simply much less systems work being produced in social computing. Both outcomes are worrying.

In this paper we chart the future of social computing systems research by assessing three challenges it faces today. First, social computing systems are caught between social science and computer science, with each discipline de-valuing work at the intersection. Second, social computing systems face a unique set of challenges in evaluation: expectations of exponential growth and criticisms of snowball sampling. Finally: is academia even the right place for social computing research? Do researchers either need to join industry, or turn their academic groups into product groups, in order to execute their ideas?

Where possible, we will offer proposed solutions to these problems. They are not perfect – we hope that the community will take up our suggestions, improve them and encourage further debate. Our goal is to raise awareness of the situation and to open a conversation about how to fix it.

Related Work

Ackerman characterized the fundamental design challenge for CSCW as the understanding and elimination of the *socio-technical gap*: the distance between the social support we know we must provide, and the technology that we know how to build [2]. CSCW systems can suffer from critical mass problems [25] and misaligned incentives [15]. The socio-technical gap, critical mass problems, and misaligned incentives are still very real challenges in social computing systems.

We are not the first to raise the plight of systems papers in SIGCHI conferences. All systems research faces challenges, particularly with evaluation. Prior researchers argue that reviewers should moderate their expectations for evaluations in systems work:

- Evaluation is just one component of a paper, and issues with it should not doom a paper [23][27].
- Longitudinal studies should not be required [22].
- Controlled comparisons should not be required, if the system is sufficiently innovative or aimed at wicked problems [14][22][29].

Not all researchers share these opinions. In particular, Zhai argues that existing evaluation requirements are still the best way to evaluate we know for now [35].

Others have also discussed methodological challenges in HCI research. Kaye and Sengers related how

psychologists and designers clashed about study methodology in the conversation of discount usability analysis methods [18]. Barkhuus traced the history of evaluation at CHI and found trends of fewer users in studies and more papers with studies as methodological approaches settled into a truce [3].

Novelty: Between A Rock and A Hard Science

Social computing systems research bridges the technical research familiar to CHI and UIST with the intellectual explorations of social computing, social science and CSCW. Ideally, these two camps should be combining methodological strengths. Unfortunately, they can actively undermine each other.

Following Brooks [8] and Lampe [19], we split the world of sociotechnical research into those following a computer science engineering tradition (“Builders”) and those following a social science tradition (“Studios”). Of course, most SIGCHI researchers work as both – including the authors of this paper. But, these abstractions are useful to describe what is happening.

Studios: Strength in Numbers

Studios’ goal is to see proof that an interesting social interaction has occurred, and an explanation of why it occurred. Social science has developed a rich set of methods for seeking this proof, but the reality of social computing systems deployments is that they are messy: more engineering and design than science. This science vs. engineering situation creates understandable tension [8]. However, the prevalence of Studios in social computing means that Studios are often the most available reviewers for a systems paper on a social computing topic.

Social computing systems are often evaluated with field studies and field experiments (*living laboratory* studies [10]), which capture ecologically valid situations. These studies will trade off many aspects of validity, producing a biased sample or large manipulations that make it difficult to identify which factors led to observed behavior. When Studios review this work, even well-intentioned ones may then fall into the Fatal Flaw Fallacy [27]: rejecting a systems research paper because of a problem with the evaluation’s internal validity that, on balance, really should not be damning. Solutions like online A/B testing and multi-week studies are often out of scope for systems papers, especially if they have small, transient volunteer populations.

Social computing systems are particularly vulnerable to Studios critique because of reviewer sampling bias. A large percentage of social computing research is focused on understanding people – so it is very likely that the reviewers on a social computing article will be Studios rather than Builders. (There are relatively few people who perform studies on tangible interaction, for example, but a large number of those working on Facebook research are social scientists.)

Builders: Keep It Simple, Stupid – or Not?

Given the methodological mismatch with Studios, we might consider having Builders reviewing systems papers. Unfortunately, systems papers in social computing are not able to articulate their value in a way that Builders might appreciate either.

Builders want to see a contribution with technical novelty: this often translates into *elegant complexity*. Memorable technical contributions are simple ideas that enable interesting, complex scenarios. Systems demos

will thus target flashy tasks, aim years ahead of the technology adoption curve, or assume technically literate (often expert) users. For example, end user programming, novel interaction techniques, and augmented reality research all make assumptions about Moore's Law, adoption, or user training.

Social computing systems, however, are not often in a position to display elegant complexity. Truly transformative social changes like microblogging are often successful *because* they are simple. So, interfaces aimed ahead of the adoption curve may not attract much use on social networks or crowd computing platforms. A complex new commenting interface might be a powerful design, but it may be equally difficult to convince large numbers of commenters to try it [19].

Caught In the Middle

Researchers are thus stuck between making a system *technically* interesting, in which case a crowd will rarely use it because it is too complex, and simplifying it to produce *socially* interesting outcomes, in which case Builder colleagues may dismiss it as less novel and Studier colleagues may balk at an uncontrolled field study. Here, a CHI metareviewer claims that a paper has fallen victim to this problem (paraphrased)¹:

The contribution needs to take one strong stance or another. Either it describes a novel system or a novel social interaction. If it's a system, then I question the novelty. If it's an interaction, then the ideas need more development.

¹ Issues pointed out by metareviewers are paraphrased to protect identity. These reviews do come from real papers, but the point is not any particular review: it is that we think they may constitute a trend. We have cited metareviewers because they must decide which concerns are most valid to raise.

For example, Twitter would likely never have been accepted as a CHI paper: there were no complex design or technical challenges, and a first study would have come from a peculiar subpopulation. It is possible to avoid this problem by veering hard to one side of the disciplinary chasm. For example, recommender systems and single-user tools Eddi [6] or Statler [32] can showcase complexity. But to accept polarization as our only solution rules out a broad class of interesting research.

A Proposal for Judging Novelty

The combination of strong Studiers and strong Builders in the subfield of social computing has immense potential if we can harness it. The challenge as we see it is that social computing systems cannot articulate contributions in a language that either Builders or Studiers speak currently. Our goal, then, should be to create a shared language for research contributions. Here we propose the Social/Technical yardstick for consideration. We can start with two contribution types.

Social contributions change how people interact. They enable new social affordances, and are foreign to most Builders. For example:

- New forms of social interaction: e.g., shared organizational memory [1] or friendsourcing [7].
- Design tweaks that impact social interactions: for example, increasing online participation [4].
- Socially translucent systems [13]: interactive systems that allow users to rely on social intuitions.

Technical contributions are novel designs, algorithms, and infrastructures. They are the mechanisms supporting social affordances, but are more foreign to Studiers. For example:

- Highly original designs, applications, and visualizations designed to collect and manage social data, or powered by social data (e.g., [6], [33])
- New algorithms that coordinate crowd work or derive signal from social data: e.g., Find-Fix-Verify [5] or collaborative filtering.
- Platforms and infrastructures for developing social computing applications (e.g., [24]).

The last critical element is an interaction effect: paired Social and Technical contributions can increase each other's value. ManyEyes is a good example [34]: neither visualization authoring nor community discussion are hugely novel alone. The combination, however, produced an extremely influential system.

Evaluation: Challenges in Living Labs

Evaluation is evolving in human-computer interaction, and in many ways social computing is leading the way. Living laboratory studies [10] of social computing systems have broken out of the university basement, focusing on ecologically valid situations and enabling many more users to experience our research ideas.

Innovating in evaluation strategies means that we are the first to experience challenges with them. There is already a lively ongoing discourse about how to evaluate systems research in HCI [14][22][23][27]. Social computing systems are facing a new set of challenges in the form of evaluation expectations and biases. Not all reviewers fall into these new biases, but they are a perceptible force and we argue they may be poorly founded. Simply put, our expectations for evaluation have exceeded our ability to perform them.

Expecting Exponential Growth

Reviewers often expect that research systems have exponential (or large) growth in voluntary participation, and will question a system's value without it. Here is a CHI metareviewer, paraphrased:

As most of the other reviewers mentioned, your usage data is not really compelling because only a small fraction of Facebook is using the application. Worse, your numbers aren't growing in anything like an exponential fashion.

There are a number of reasons why reviewers might expect exponential growth. First, large numbers of users legitimize the idea: growth is strong evidence that the idea is a good one and that the system may generalize. Second, public research systems are on a level playing field with non-research applications. Usage numbers are the *lingua franca* for evaluating non-research social systems, so why not research systems as well? Last, unlike other systems, social computing systems can do large-scale rollouts, so the burden may be on them to try. We do agree that if exponential growth does not occur, authors should acknowledge this and explore why.

However, it misses the mark to *require* exponential growth for a research system. One major reason this is a mistake is that it puts social computing systems on unequal footing with other system evaluations. Papers in CHI 2006 had a median of 16 participants: program committees considered this number acceptable for testing the research's claims [3]. Just because a system is more openly available does not mean that we need orders of magnitude more users to understand its effects. Sixteen friends communicating together on a Facebook application may still give an accurate picture.

Another double standard is a conflation of *usefulness* and *usability* [21]. Usefulness asks whether a system solves an important problem; usability asks how users interact with the system. Typically in systems papers, authors prove usefulness through argumentation in the paper's introduction, then prove usability through evaluation. Evaluations will shy away from usefulness because it is hard to prove scientifically. Instead, we pay participants to come and use our technology temporarily (assuming away the motivation problem), because we are trying to understand the effects of the system once somebody has picked it up. This should be sufficient for social computing systems. However, reviewers in social computing systems papers will look at an evaluation and decide that a lack of spread empirically disproves any claim of usefulness. ("Otherwise, wouldn't people flock to the system?") But why require usefulness – voluntary usage – in social computing systems, when we assume it away – via money – for other systems research?

A final double standard is whether we expect risky hypothesis testing or conservative existence proofs of our systems' worthiness [14]. A public deployment is the riskiest hypothesis testing possible: the system will only succeed if it has gotten almost everything right, including marketing, social spread, graphic design, and usability. Systems evaluations elsewhere in CSCW, CHI and UIST seek only existence proofs of specific conditions where they work. We will not argue whether existence proofs are always the correct way to evaluate a systems paper, but it is problematic to hold a double standard for social computing systems papers.

The second major reason it is a mistake to require exponential growth is that a system may fail to grow

for reasons entirely unrelated to its research goals. Even small problems in the social formula could doom a deployment: minor *channel factors* like logins, slow or buggy software can change a user's actions [31]. If we want novel social interactions, rather than immediate success we should expect a series of work getting us continually closer. Trying to achieve Last.fm on the first try is absurd; we need precursors like Firefly first. In fact, we may learn more from failed deployments of systems with well-positioned design rationale.

Get Out of the Snow! No Snowball Sampling

Live deployments on the web have raised the question of snowball sampling: starting with a local group in the social graph and letting a system spread organically. CHI generally regards snowball sampling as bad practice. There is good reason for this concern: the first participants will have a strong impact on the sample, introducing systematic and unpredictable bias into the results. Here, a paper metareviewer calls out the sampling technique (paraphrased):

The authors' choice of study method – snowball sampling their system by advertising within their own social network – potentially leads to serious problems with validity.

Authors must be careful not to overclaim their conclusions based on a biased sample. However, given well-scoped claims, some reviewers will still argue that systems should recruit a random sample of users, or make a case that a new online community is broadly representative of the population it is targeted at.

What we must recognize is that *snowball sampling is inevitable in social systems*. Social systems spread through social channels – this is fundamental to how they operate. We need to embrace this process.

Second, random sampling can be an impossible standard for systems research. All it takes is a single influential user to tweet about a system and the sample will be biased. Further, many social computing platforms like Twitter and Wikipedia are beyond the researcher's ability to recruit randomly. Likewise, an online community might only be able to recruit citizens of one or two regions – the sample may be biased, but it is certainly enough to learn the highest-order bit lessons about the software. Random sampling is often impossible in social science too: snowball sampling is welcomed there to reach difficult populations.

Finally, convenience sampling is (frankly) quite common in systems research outside of social computing. Again, it is problematic to levy different requirements against social computing papers than other papers. Half of CHI papers through 2006 used a primarily student population [3]. Other common tactics often also constitute a convenience (and effectively snowballed) sample, like recruiting locals or university employees. In fact, a social network-based advertisement is likely to reach a more geographically diverse population than most CHI studies do. We may debate whether convenience sampling in CHI is reasonable on the whole (e.g., Barkhuus [3]), but we should not be applying the criteria unevenly.

A Proposal for Judging Evaluations

Because our methodological approaches have evolved, it is time to develop meaningful and consistent norms about these concerns. An exhortation to take the long view and review systems papers holistically (e.g., [14][22][29]) can be difficult to apply consistently. So, here we propose more specific suggestions, aiming to retain methodological validity when possible.

Separate Evaluation of Spread from Steady-State

We argued that exponential growth is a faulty signal to use in evaluation. But, there are times when we should expect to see viral spread in an evaluation: when the research contribution *makes claims about spread*.

We can separate out two types of usage evaluations: *spread* and *steady-state*. All social systems go through these two phases: (1) a user base is recruited, then (2) users interact with the application. An evaluation can focus on how a system spreads, or it can focus on how the system is used once it is adopted.

Paper authors should evaluate their system with respect to the claims they are making. If the claims focus on attracting contributions or increasing adoption, for instance, then a spread evaluation is appropriate – the authors need to show that the system is increasing contributions or adoption. If, instead, the paper makes claims about introducing a new style of social interaction, then we can ignore questions of adoption for the moment and focus on what happens when people have started using the system. This logic is again parallel to that of laboratory evaluations: we solve the questions of motivation and adoption (spread) by paying participants, and focus on the effects of the software once people are using it (called *compelled tasks* in Jared Spool's terminology). Authors will need to address the other aspects of their system evaluation in writing and identify limitations, but evaluations should not be required to accomplish both goals.

Treat Any Amount of Voluntary Use As A Success

We need to stop treating a small amount of voluntary use as a failure, and instead recognize it for what it is: *a success*. Most systems studies in CHI have to pay

participants to come in and use buggy, incomplete research software. Any voluntary use is better than most CHI research systems will ever see. The field studies that social computing systems perform are much harder to execute than the laboratory studies that other papers perform. These papers should get *extra* leeway for taking this approach, not less.

Make A Few Snowballs

As we argued, it is almost impossible to get a random sample of users in a field evaluation. To begin, authors should be careful not to overclaim that their observations generalize to an entire population. Beyond this, we propose a compromise: a few different snowballs can help mitigate bias. Authors should make an effort to seed their system at a few different points in the social network, characterize those populations and any limitations they introduce, and note any differences in usage. But we should not reject a paper because its sample falls near the authors' social network. There may still be sufficient data to evaluate the authors' claims relatively even-handedly. Yes, perhaps the evaluation group is more technically apt than the average Facebook user; but most student participants in SIGCHI user studies are too [3].

Research At A Disadvantage with Industry

The next challenge that social computing systems research faces as it forges new ground is, oddly, its continued relevance. Systems research is often ahead of the curve, using new technology to push the limits of what is possible. But in the age of the social computing, academic research now trails industry platforms. A researcher must function as an entire start-up – marketing, engineering, design, QA – and compete with companies for attention and users. It is not surprising

that researchers worry whether start-ups are actually a better path for them (see Landay [22] and associated comments). If they stay in academia, researchers must satisfy themselves with limited access to platforms they didn't create, or chance attracting a user population from scratch and then work to maintain it.

It is difficult to build systems on closed platforms. In social computing, researchers often go where the users are, and the users are largely on closed platforms like Wikipedia, Facebook or Twitter. These platforms are typically averse to letting researchers make changes to their interface. If a researcher wanted to try changing Twitter to embed non-text media in tweets, they should not expect cooperation. Instead, we must re-implement these sites and then find a complicated means of attracting naturalistic use. For example, Hoffman et al. mirrored and altered Wikipedia, then used advertisements cued on Wikipedia titles to attract users [16]. Wikidashboard also proxied Wikipedia [33].

Many would go farther and argue that social computing systems research is a misguided enterprise entirely. Brooks argues that, with the exception of source control and Microsoft Word's Track Changes, CSCW has had no impact on collaboration tools [8]. Welsh claims that researchers cannot really understand large-scale phenomena using small, toy research environments.² Researchers at companies like Facebook, IBM, and Microsoft advertise exclusive access to their systems as a benefit of working there. Some hold that if your research competes with industry, you should go to an industry lab (see Ko's comment in Landay [22]). Does

² <http://matt-welsh.blogspot.com/2010/10/computing-at-scale-or-how-google-has.html>

this mean that social computing PhDs should be carried out under the co-supervision of industry? It seems unwise to tie research progress to industrial resources.

Some academics choose to create their own communities and succeed, for example Scratch [28], MovieLens³, and IBM Beehive [12]. These researchers have the benefit of continually experimenting with their own platforms and modifying them to pursue new research. Colleagues and our own experience indicate that this approach carries risks as well. Researchers devote a large amount of time to product needs, support, and engineering that pays back little. In short, the research *becomes a product*. Such researchers express frustration that their later contributions are then written off as “just scaling up” an old idea.

We have no ready answers. (If we did, we would already be applying them.) However, we believe that academic research can innovate where industry may never go due to missing market incentives. Clearly, industry will continue to refine existing platforms [11]; but systems research should strike out to create alternate visions and fashion new futures.

This discussion also provides a framework for authors to articulate why they are taking a particular approach:

- Co-opting an existing community: the researchers are able to collaborate with the existing community, or are satisfied with playing inside a developer sandbox. It is easier to integrate with a user base, but is often harder to make big changes.
- Creating a new community: existing communities do not support the needs that this community fills,

or it is difficult to co-opt existing sites. This choice gives complete freedom, but recruitment is difficult. Authors should report this decision and any limitations that come with it to help scope readers’ expectations.

Conclusion

Social computing systems research is struggling in SIGCHI. Some challenges relate to our research questions: interesting social innovations may not be interesting technically, and field studies are rarely controlled enough to satisfy social scientists. In response, we laid out the Social/Technical yardstick for valuing research claims. Other challenges are in evaluation: a lack of exponential growth and snowball sampling are incorrectly dooming papers. We argued that different styles of system evaluation are possible, and that not all share these requirements. Finally, we considered the place of social computing systems research with respect to industry.

As much as we would like to have answers, we know that this is just the beginning of a conversation. We invite the community to participate and contribute their suggestions. We will use the opportunity of alt.chi’s open reviewing process to learn more about the community’s concerns and suggestions, then use them to shape the final version of this paper. Beyond the scope of alt.chi, we hope that this work will help catalyze the social computing community to discuss the role of systems research more openly and directly.

Acknowledgments

The authors would like to thank: Travis Kriplean, Mihir Kedia, Eric Gilbert, Cliff Lampe, David Karger, Bjoern Hartmann, Adam Marcus, Andrés Monroy-Hernandez, Drew Harry, Paul André, and David Ayman Shamma.

³ <http://www.movielens.org/>

References

- [1] Ackerman, M. 1994. Augmenting the Organizational Memory: A Field Study of Answer Garden. In *Proc. CSCW '94*.
- [2] Ackerman, M.S. 2000. The Intellectual Challenge of CSCW: The Gap Between Social Requirements and Technical Feasibility. *Human-Computer Interaction* 15(2).
- [3] Barkhuus, L., and Rode, J.A. 2007. From Mice to Men – 24 years of evaluation in CHI. In *Proc. alt.chi '07*.
- [4] Beenen, G., Ling, K., Wang, X., et al. 2004. Using social psychology to motivate contributions to online communities. In *Proc. CSCW '04*.
- [5] Bernstein, M.S., Little, G., Miller, R.C., et al. Soylent: A Word Processor with a Crowd Inside. In *Proc. UIST '10*.
- [6] Bernstein, M.S., Suh, B., Hong, L., Chen, J., Kairam, S., and Chi, E.H. 2010. Eddi: Interactive topic-based browsing of social streams. In *Proc. UIST 2010*.
- [7] Bernstein, M.S., Tan, D., Smith, G., et al. 2010. Personalization via Friendsourcing. *TOCHI* 17(2).
- [8] Brooks, F. 1996. The computer scientist as toolsmith II. *Communications of the ACM* 39(3).
- [9] Brooks, F. 2010. *The Design of Design*. Addison Wesley: Upper Saddle River, NJ.
- [10] Chi, E.H. 2009. A position paper on 'Living Laboratories': rethinking ecological designs and experimentation in human-computer interaction. HCI International 2009.
- [11] Christensen, C.M. 1997. *The Innovator's Dilemma*. Harvard Business Press.
- [12] DiMicco, J., Millen, D.R., Geyer, W., et al. 2008. Motivations for social networking at work. In *Proc. CSCW 2008*.
- [13] Erickson, T., and Kellogg, W. 2000. Social Translucence: An approach to designing systems that support social processes. *TOCHI* 7(1).
- [14] Greenberg, S., and Buxton, B. 2008. Evaluation considered harmful (some of the time). In *Proc. CHI '08*.
- [15] Grudin, J. 1989. Why groupware applications fail: problems in design and evaluation. *Office: Technology and People* 4(3): 187-211.
- [16] Hoffmann, R., Amershi, S., Patel, K., et al. 2009. Amplifying community content creation with mixed initiative information extraction. In *Proc. CHI '09*.
- [17] Ishii, H., and Kobayashi, M. 1992. ClearBoard: a seamless medium for shared drawing and conversation with eye contact. In *Proc. CHI '92*.
- [18] Kaye, J., and Sengers, P. 2007. The evolution of evaluation. In *Proc. alt.chi 2007*.
- [19] Kriplean, T., Toomim, M., Morgan, J., et al. 2011. Toward REFLECT-ive Communication: Closing the speaker-listener feedback loop on the web. In *Proc. CSCW '11 Horizon*.
- [20] Lampe, C. 2010. The machine in the ghost: a socio-technical approach to user-generated content research. WikiSym keynote. <http://www.slideshare.net/cliff Lampe/the-machine-in-the-ghost-a-sociotechnical-perspective>
- [21] Landauer, T. 1995. *The Trouble with Computers: usefulness, usability and productivity*. MIT Press, Cambridge.
- [22] Landay, J.A. 2009. I give up on CHI/UIST. Blog. <http://dubfuture.blogspot.com/2009/11/i-give-up-on-chiui.html>.
- [23] Lieberman, H. 2003. Tyranny of Evaluation. *CHI Fringe*.
- [24] Little, G., Chilton, L., Goldman, M., and Miller, R.C. TurkKit: Human Computation Algorithms on Mechanical Turk. *UIST '10*.
- [25] Markus, L., 1987. Towards a "critical mass" theory of interactive media: Universal access, interdependence, and diffusion. *Communication Research* 14: 491-511.
- [26] Morris, M. Personal communication. December 22, 2010.
- [27] Olsen Jr., D. 2007. Evaluating User Interface Systems Research. In *Proc. UIST '07*.
- [28] Resnick, M., Maloney, J., Monroy-Hernández, A., et al. 2009. Scratch: programming for all. *Commun. ACM* 52(11).
- [29] Rittel, H.W.J. and Webber, M.M. 1973. Dilemmas in a general theory of planning. *Policy Sciences* 4:2, pp. 155-169.
- [30] Roseman, M., and Greenberg, S. 1996. Building real-time groupware with GroupKit, a groupware toolkit. *TOCHI* 3(1).
- [31] Ross, L., and Nisbett, R. 1991. *The person and the situation*. Temple University Press.
- [32] Shamma, D., Kennedy, L., and Churchill, E. Tweetgeist: Can the twitter timeline reveal the structure of broadcast events? In *CSCW 2010 Horizon*.
- [33] Suh, B., Chi, E.H., Kittur, A., and Pendleton, B.A. 2008. Lifting the veil: improving accountability and social transparency in Wikipedia with WikiDashboard. *Proc. CHI '08*.
- [34] Viegas, F., Wattenberg, M., van Ham, F., et al. 2007. ManyEyes: a Site for Visualization at Internet Scale. *IEEE Trans. Viz. and Comp. Graph.* Nov/Dec 2007: 1121-1128.
- [35] Zhai, S. 2003. Evaluation is the worst form of HCI research except all those other forms that have been tried. Essay published at *CHI Place*.