Blogging Developer Knowledge: Motivations, Challenges, and Future Directions

Chris Parnin
College of Computing
Georgia Institute of Technology
Atlanta, Georgia USA
chris.parnin@gatech.edu

Christoph Treude
School of Computer Science
McGill University
Montreal, Canada
ctreude@cs.mcgill.ca

Margaret-Anne Storey
Department of Computer Science
University of Victoria
Victoria, Canada
mstorey@uvic.ca

***Abstract*—Why do software developers place so much effort into writing public blog posts about their knowledge, experiences, and opinions on software development? What are the benefits, problems, and tools needed—what can the research community do to help? In this paper, we describe a research agenda aimed at understanding the motivations and issues of software development blogging. We interviewed developers as well as mined and analyzed their blog posts. For this initial study, we selected developers from various backgrounds: IDE plugin development, mobile development, and web development. We found that developers used blogging for a variety of functions such as documentation, technology discussion, and announcing progress. They were motivated by a variety of reasons such as personal branding, knowledge retention, and feedback. Among the challenges for blog authors identified in our initial study, we found primitive tool support, difficulty recreating and recalling recent development experiences, and management of blog comments. Finally, many developers expressed that the motivations and benefits they received for blogging in public did not directly translate to corporate settings.**

I. Introduction

Widely accessible infrastructure and social media technology such as blogs, wikis, and Q&A sites (*e.g.*, Stack Overflow) have changed how developers contribute and share knowledge [1]. There is growing evidence that these forms of social media are supplanting and improving upon many traditional forms of knowledge sharing such as official documentation, books, or asking colleagues for help.

Previously, we found when analyzing a sample of blog posts from developers using jQuery [2], that the posts cover 88% of the jQuery API methods, feature tutorials and personal experiences about using the methods, and are created by multiple developers contributing a few blog posts each. We also analyzed discussions on Stack Overflow about the Android API and found they contained more explanations and code examples than the official documentation [3]. We also have preliminary evidence that developers will visit a social media site more often than its traditional counterpart[1].

Despite being such a widely practiced and important phenomenon, scant research has been performed on the role of blogging in software development. Our goal is to understand the motivations, processes, and problems associated with blogging about software development, and to identify research directions and new tools that would assist developers wishing to blog. In this initial study, we asked four questions:

1) What do software developers blog about?
2) Why do software developers blog?
3) How interactive are software development blogs?
4) What are the challenges for developers who blog?

II. Background

Besides our previous study [2], we only know of one additional study of software development blogs. Pagano and Maalej examined the blogging behaviors of developers who commit to open source projects: Eclipse, GNOME, PostgreSQL, and Python [4]. They used a blog aggregator site to find bloggers associated with the development projects, and then matched the blogger's identity to source code committers in the project. In the study, only 1.8% of blog posts contained source code, and the most frequent topic was the discussion of functional requirements and domain concepts about the project (as determined by an automated LDA topic modeling algorithm). In contrast to their work, our research focuses on the motivations and challenges of developers who blog.

Before describing our study of software developer bloggers in the next section, we highlight research results on blogging in other contexts, focusing on corporate blogging and motivation.

In many companies outside of software development, blogging has found its way into corporate culture. In a study from 2007, Efimova and Grudin [5] describe emergent blogging practices in a corporate setting. They found blogging in the enterprise to be an experimental, rapidly evolving terrain in which balancing personal and corporate incentives and issues is one of the challenges that bloggers face. In a similar study, Huh *et al.* [6] found that the corporate blogging community provides access to tacit knowledge and that it contributes to new forms of collaboration within the enterprise.

Several studies have looked at the motivation and challenges of bloggers, however there exists no study that looks specifically at the motivation of blogging software developers. Yardi *et al.* [7] found that corporate bloggers did not receive as much attention as they expected when they contributed. In some cases, that led to frustration. Gurzick and Lutters [8] investigated the life cycle of blogs and identified several additional reasons for abandonment such as the discovery of other

[1] blog.ninlabs.com/2013/03/api-documentation/

tools to fulfill similar functions or changes in employment. Nardi *et al.* [9] found that the interactivity in blogs is limited, and they speculate that many bloggers contribute to blogs in order to reflect, share opinions and advice.

III. Methodology

We performed a preliminary study that involved manually inspecting blog posts and surveying the authors of the blog posts. As there exists no central directory of all software related blogs, an early research challenge is identifying the best methodology for sampling blogs. For our preliminary work, we first selected three technology areas: IDE plugin development, mobile development, and web development. To widen our coverage of each technology area, we chose two complementary technology platforms for each area: Eclipse and Visual Studio plugins, Android and iPhone development, and Django and jQuery development.

For each technology platform, we generated a list of three keyword searches. We studied up on the technology platforms to identify several common concepts, tasks, or questions developers might have. To select our keywords, we used a mixture of tasks and specific technical terms such as the name of an API call. For example, for jQuery, we selected the following keyword searches: "jquery div id", "jquery ajax", and "jquery json". We put each of our queries into Google Blog Search and selected the first 3-4 results listed, yielding a total of 55 blogs (52 having comment data), 6,726 posts and 32,188 comments. Exported data in JSON format are available for download for other researchers[2]. We inspected 300 randomly selected blog posts to understand what types of blog posts developers wrote, focusing on three features: post topic, presence of code, and exposition style.

To understand the community interactions and social structures in software blogs, we selected blog posts with both low and high frequency of comments from each blog author in order to evenly sample potentially different feedback types that might occur. When inspecting a post's comments, we noted the post topic, comment type, and conversation structure. In total, we analyzed 435 comments from 93 posts.

To get a deeper understanding of the motivations and challenges involved with software development blogs, we then sent a survey to the authors of all 55 blogs and received 30 responses. The survey is available online[3].

IV. Preliminary Findings

A. Blogs Posts Focus on Code but with Multiple Facets

Figure 1 shows the different kinds of blog posts we identified in our study. We observed **documentation**, *tutorials and experience reports about technology*, to be the most frequent type of post (40%). There were nearly three times as many posts about documentation in comparison to other types of posts. Our survey confirms this result with 71% of developers saying their most common type of blog post is about a recent coding experience. Overall, we found 35% of the blog posts we inspected to contain code, but many posts discussed APIs and technology without explicit mentions of code.

Documentation posts can further be broken down by the type of documentation they contain (see Figure 1). Surprisingly, **workarounds**, *a type of post describing how to circumvent a problem* and **simple code snippets**, *a type of post with only code*, were less common then we had expected. For workarounds, our survey revealed that developers may have trouble replicating the circumstances and different attempts of their original problem or abstracting to a form understandable by a wider audience.

Many authors announced **progress**—*new releases and enhancements to their personal or company projects* (12%). This allowed the authors to spread the word or solicit early feedback or engage in community building. Other authors, such as leaders of open source projects, used their blog to provide an authoritative voice for their project. In some posts, they would outline a vision for future directions of the project, or attempt to settle an open debate within the community.

Many authors who were not necessarily in any authoritative role often expressed viewpoints and engaged in **technology discussion**—*long discussions over best practices or principles of software engineering* (12%). Often these posts capture the inner struggle of a developer or a community learning how to solve a difficult problem. For example, one developer described his struggle with developing asynchronous web technologies in a post titled: "How do we kick our synchronous addiction?". These types of posts also provide high-level commentary for certain technologies describing benefits and dislikes. For companies looking for feedback or developer sentiment on their technologies, there are many posts that can be collected and analyzed from the community.

B. Motivations: Personal Branding, Sharing, and Retention

The authors described the benefits of blogging as follows:

Personal branding: Developers felt maintaining a blog was an effective tool for future employment or attracting clients. A blog provides a place for showcasing past projects and demonstrating problem solving and communication skills. One developer states this point clearly in our survey: *"I've been told by my boss and a few older colleagues that owning a blog and keeping it up to date with relevant information is quite a good selling point for myself. Also, as I live in Australia, it's nice to see so much overseas traffic. It makes me feel as though there is definitely the potential for a job offer overseas."*

Evangelism and recruitment: Some technology platforms had higher corporate sponsorship than others. When comparing Visual Studio and Eclipse plugins, 60% of Visual Studio bloggers could be considered corporate bloggers in comparison to only one author affiliated with IBM for Eclipse plugins. One developer describes how blogging helps educate a community around a corporation's technology: *"Partly because I want to talk about this stuff, but also because it is a good career builder. Now, as a Sun employee, I blog to educate other programmers about our new technologies."*

[2] https://sites.google.com/site/developerblogs/data/blogging-data

[3] https://sites.google.com/site/developerblogs/home/main-survey

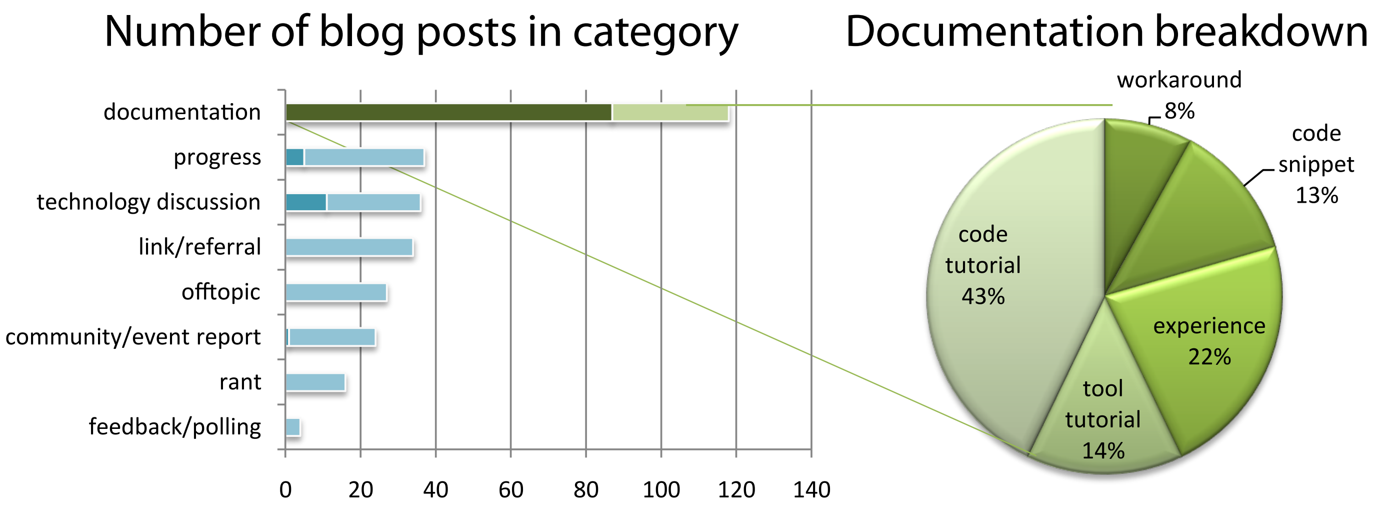

Fig. 1. Breakdown of inspected blog posts. On the left, we show the number of blog posts in each category (dark regions indicate blog posts containing source code). On the right, posts about documentation are further broken down by their type of documentation.

Personal knowledge repository: The most obvious and frequent use of a blog was to catalog experiences. In our survey, the most cited benefit from blogging about coding was that the process of writing helped the author learn and remember the information better (93% participants). Motivation to blog an experience included lack of documentation or frustration level involved in problem solving an undocumented issue. For niche or new and emerging technologies, such as iPhone development, this was especially true and the contributed blog posts established initial documentation for that community.

Solicit feedback: Developers do not directly ask for feedback, but often indirectly receive feedback through comments or emails. Often a community member may contribute improved code solutions or identify errors, as one developer mentioned in our survey: *"I find it to be useful way to a) archive my interesting ideas and bits of code, and b) get feedback from the community on what I'm doing."*

C. *Blog Comments Facilitate Community Interactions*

From our data, we found that many posts receive feedback: 52% of posts have comments. The number of comments does vary, responded-to posts typically received 1-10 comments.

From the 93 manually inspected posts, we found the most frequent type of feedback authors received were **questions** and **expressions of gratitude**. Questions were found in the comments of 32 posts. The questions asked for clarification or help in running code samples. The questions often prompted the author to update the blog post with clarifications (or attach better code samples). But not all these questions were answered; 10 out of 32 posts with questions had unresolved questions. Expressing gratitude is primarily beneficial to the author and motivated them to continue contributing future blog posts, but may offer little value to the community. However, these expressions may benefit other readers in gauging how many other developers faced similar circumstances ("I'm not the only one!").

In addition to simple responses, deeper communication structure emerged. Especially in blog posts on technology discussions, many readers weighed in with their own opinions on the matter. The discussion often resulted in many comments and participants and gave shape to a community voice expressing many well-thought out aspects of the topic.

V. Challenges for Bloggers

A. *Managing Blogging Resources and Enthusiasm*

When examining the first posts of several blogs, we noted four cases where the blog authors reported losing blogs due to hosting failures or malicious attacks. Other times developers noted in our survey that they had difficulty porting between systems or felt the blog had decayed and decided to restart. A common theme that emerged was that blogging would undergo **rhythmic cycles** of enthusiasm and disinterest and often require rebirth. One developer describes this situation: *"14,000+ comments, all spam. So much so that the database query was timing out. It was easier to just nuke the original blog and database and create a new one. Oh well, the last post on the original blog was over 2 years ago anyway."*

B. *Shepherding Readership and Community Contributions*

Some blog posts received hundreds of comments, largely composed of questions to the author or other readers. Authors lose interest in responding to questions after a while, being overwhelmed by too many questions, or not being able to give a reasonable reply. In the last case, a few questions were from readers soliciting help on related problems, but without expressing sufficient context within the comment or giving a proper incentive for answering the question. Researchers may want to explore support for migrating or incorporating different social media techniques such as reputation into social interactions. In general, several improvements can be made to distinguish and improve how questions are asked and answered.

Community members often contribute suggestions to improve code, but do not have rich authoring tool support. Recently, some blogs have started to host code segments remotely using services called "gists", which are stored in git repositories. Improving this technology to allow authors easily to approve other community members' access and update gists would be a step in the right direction. Blogging platforms such as medium.com have recently offered support for embedding notes in text[4].

C. Blogging is Time-Consuming

Authors invest a considerable amount of time to create content for their blogs. From our survey, developers estimated that they typically spend from 30 minutes to several hours to prepare a single blog post. Some even claim spending weeks editing longer entries.

The authors identified several time-consuming aspects:

Preparing code samples and attachments: Authors frequently complained about preparing and managing code samples. The process of finding, extracting, and formatting code for presentation was done manually. Further, some sites made it difficult to host assets such as code attachments or images.

Many developers were also inconsistent in formatting code. Only 25% of the blogs used code syntax highlighting for code samples, which can make reading the posts more difficult for readers. Although blogging platforms continue to improve, this illustrates that, for many bloggers, tool support is still emerging out of a primitive state.

Making examples concise and reproducible: Making code simple and understandable outside of the original problem context is difficult: *"Often ideas for blog posts arise during day-to-day coding at my job, and usually the projects where I have found a solution contain extraneous dependencies and customized business logic... simple blog entries take time."*

Bloggers delay writing posts: Developers are unlikely to write during the experience that inspired the entry (only 20% of authors reported writing a post the same day), or to write up multiple posts at once. Most will write several days after completing their effort, which negatively affects their ability to recall and recreate the experience. Developers need tools that will help them recall and reconstruct how-to perform a programming task from the past (beyond source code diffs).

D. Blogging in the Workplace

Several bloggers cited reasons why they might blog in public but not in the workplace. Some reported that corporate blogging might limit exposure and thus usefulness, and also impose censorship. Still, some authors were enthusiastic about blogging at the workplace. One remedy that bloggers identified was to use a more simplified note-taking style of blogging with less formal structure. Research on new incentives and contribution models is needed to encourage and improve workplace blogging.

VI. Conclusions and Future Work

Developers use blogs to support a variety of development activities. Benefits such as personal branding and maintaining a personal knowledge repository motivate them. Still, developers face many challenges that often result in less frequent blogging: Several bloggers only wrote a blog post every few months despite wanting to contribute more often. Bloggers need better tool support for preparing posts and allowing for better community interactions—the research community is ideally situated to provide these new tool ideas. Researchers also need to find better ways to sample and discover the wide range of topics that developers address using social media.

There are many research directions to explore:

1) Publishing tools: How to support or automate blogging of coding tasks?
2) Corporate blogging: What incentives will encourage developers to blog in the workplace?
3) Darknet: Who is *not* participating and why not?
4) Knowledge tools: Aggregating, curating, and mining blog post content.
5) Content staleness: Do posts stand the test of time? What to do with obsolete blog posts?
6) Web provenance: What happens with code copied from online sources?

References

[1] C. Treude, F. F. Filho, B. Cleary, and M.-A. D. Storey, "Programming in a socially networked world: the evolution of the social programmer," The Future of Collaborative Software Development (FutureCSD), 2012.

[2] C. Parnin and C. Treude, "Measuring api documentation on the web," in *Proceedings of the 2nd International Workshop on Web 2.0 for Software Engineering*, ser. Web2SE '11. New York, NY, USA: ACM, 2011, pp. 25–30.

[3] C. Parnin, C. Treude, L. Grammel, and M.-A. Storey., "Crowd documentation: Exploring the coverage and the dynamics of api discussions on stack overflow," Georgia Institute of Technology, Tech. Rep. GIT-CS-12-05, 2012.

[4] D. Pagano and W. Maalej, "How do developers blog? an explorative study," in *Proceedings of the Eigth International Working Conference on Mining Software Repositories*, 2011.

[5] L. Efimova and J. Grudin, "Crossing boundaries: A case study of employee blogging," in *HICSS '07: Proceedings of the 40th Annual Hawaii International Conference on System Sciences*. Washington, DC, USA: IEEE Computer Society, 2007, p. 86.

[6] J. Huh, L. Jones, T. Erickson, W. A. Kellogg, R. K. E. Bellamy, and J. C. Thomas, "Blogcentral: the role of internal blogs at work," in *CHI '07: CHI '07 extended abstracts on Human factors in computing systems*. New York, NY, USA: ACM, 2007, pp. 2447–2452.

[7] S. Yardi, S. A. Golder, and M. J. Brzozowski, "Blogging at work and the corporate attention economy," in *CHI '09: Proceedings of the 27th international conference on Human factors in computing systems*. New York, NY, USA: ACM, 2009, pp. 2071–2080.

[8] D. Gurzick and W. G. Lutters, "From the personal to the profound: understanding the blog life cycle," in *CHI '06: CHI '06 extended abstracts on Human factors in computing systems*. New York, NY, USA: ACM, 2006, pp. 827–832.

[9] B. A. Nardi, D. J. Schiano, and M. Gumbrecht, "Blogging as social activity, or, would you let 900 million people read your diary?" in *CSCW '04: Proceedings of the 2004 ACM conference on Computer supported cooperative work*. New York, NY, USA: ACM, 2004, pp. 222–231.

[4] https://medium.com/about/5972c72b18f2