IBM conducted research on a new type of web-based conversation tool that is different from the conventional forum. The researchers termed this new type a “flash forum” due to its bursty nature by analogy to a flash mob. A main problem of the flash forum is that conventional news readers are not appropriate anymore given the distinct differences: lack of importance of authorship information, large size, tight focus overall with overlapping topics between threads, and a short timeframe for the conversation. In order to deal with these challenges, the ForumReader was developed and the process of the design had some interesting implications.

One major theme of the design was to focus on content rather than people. Rather than using communication to meet people and to hold conversation, the emphasis is instead on knowledge and finding interesting posts. A potential problem of lessened communication is that repeated posts may become frequent and collaboration on ideas about similar topics may lessen, providing no direction. For the first issue, ForumReader cleverly implemented a search function that could perform realtime visual text searches. This provides a way to quickly sense what is present in unwieldy flash forums, helping a user see if the idea or topic has been posted previously so that they can avoid repeating it. However, this requires a user to hold that responsibility and in a time-pressed environment like a flash forum, it’s probably not a realistic expectation. Since there may be many repeats, collaboration decreases because the same subjects may be emphasized over and over, really creating no growth. A possible fix to this is to add “related posts,” taking other posts with shared common words, in the text view so that users can jump from one topic to another, and try not to repeat on what has been previously said.

Considering the above constraints, it’s interesting that IBM chooses a flash forum format for their company-wide discussion about values. The event was only held for 3 days and brought over 8,000 posts from almost 5,000 participants. If the intent of the event was to generate discussion about values and collaborate on ideas on what the company’s values should look like, this format would have to be the worst. Because of low reply count and low percentage of authors posting more than once, there’s really very little feedback going on. The discussion is limited by the influx of ideas. Even with the ForumReader version they used, called ValuesJam, didn’t seem to increase posts per participant much. On the other hand, maybe the point is to encourage lurking and information gain over actual discussion. If that’s the case, IBM chose the right format... and their true intent of informing the employees rather than using the event to discuss values.

Afterwards, the reader was made public to Slashdot and new features were implemented to help users find the interesting posts. Color was added, which seemed to be one of the most important features according to the ratings. Coloring by moderation type was just as important as other highly ranked features such as the search and the simple text view. This suggests that topic atmosphere (funny, insightful) is more important than a numerical rating. People’s goals are reflected in these atmospheres
and perhaps they want to filter messages by what they deem “interesting discussions.” But this may have a different connotation for each individual. Again, the goal goes back to helping the user find what is interesting to them. Perhaps further study should emphasize and categorize specific interests, not just the need to condense of information, but on enhancing the entertainment value as well.

Critique – CodeSaw: A Social Visualization of Distributed Software Development

CodeSaw is a visualization that helps paint a picture of open-source collaboration. The archives of open-source projects give significant information about the actual conversations of a project but were a largely untapped resource. By using CodeSaw, the flow of thousands of conversations can be understood at once. This in turn gives developers a chance to reflect on a long project, connect to other developers, have incentive to make their development strategy more efficient, and continue to contribute to the project.

The major focus of the project was to help developers connect to the community. CodeSaw accomplished this by allowing people to see how other people have contributed comparatively to themselves. Also, by comparing users, this also helped people see the cycles in development, discover roles that other people played, and confirm support they receive. Trends in a person’s development were also spotted. For example, one person may contribute code mainly during the summer while another contributes with another developer. By seeing these trends, people can actually analyze social data and make speculations about others that they wouldn’t have known before. This is probably the most powerful insight of CodeSaw and really targets the key problem of developers feeling disconnected from the rest of the community.

After the field study, another issue arose which led to another great feature. People commented that in communities with only a few developers, CodeSaw actually made them realize how small the community actually was. This made developers feel lonely because there were no other people on the visualization except for them. To address this issue, the research team came up with spatial messaging, which allows anyone viewing the visualization to leave public messages on the visualization itself. This allows sharing of thoughts between people and inclusion of others into the development history even though they didn’t partake in the actual coding. Another way to include the broader community is to also keep track of people who submit bug reports. Also, there are bound to be people who are waiting for a release or visit the site of the open-source project to lurk. If those people could be captured as a side option, developers may get a larger sense of the people that there are people communicating with him outside of just code development.

Personally, I really like the idea of CodeSaw a lot more than some of the other visualizations that we’ve read about. It first and foremost targets a large need and addresses the need in an iterative way. I also enjoyed the snippets of personal comments from the people who have used CodeSaw and each of them had different perspectives of it. If anything, reading all of these papers has given me more insight and appreciation for the actual research process itself.
Critique – Bridging The Gap: A Genre Analysis of Weblogs

First of all, this article was particularly interesting to me because I was planning to actually create a new personal site/blog by the end of graduation. However, I was really struggling with determining the purpose of the site and how to enhance the usage of the site with the blog. This paper helps me analyze the types of blogs out there currently by author characteristics, purpose, time and structure. In fact, with all the data presented, I can strategically make my blog somewhat unique and also leverage the most popular types.

Some interesting statistics were given about blogs that I didn’t know of before. The fact that over 90 percent of blogs are by one author signifies to me that blogs still leverage the original concept of a personal website, with the personal touch being very critical. Most blogs are updated daily, which is a surprise to me because I always thought that was a bit overboard. Also, almost 70 percent of bloggers are located in the USA. To me at least, that can’t be indicative of the whole population. Since I’m pretty active in some Asian-interest communities, I know that blogging has become a sensation, but how many non-English blogs are captured?

Another huge downside to this study is that it ignores blogs integrated to other websites. Now, many websites like Facebook and forum communities have blog-like features internally. To address my original concern about Asia, in Korea, a hugely popular site called Cyworld also has blogs internal to the social network. Cyworld could have added a couple million available blogs easily, perhaps changing the percentage of USA located bloggers. With the influx of blogs everywhere and in different context, can these statistics really be representative of the general population? Using the blog tracking website may be a start, but a more diverse sampling is needed. I think the authors try to address this issue at the end when drawing the weblog continuum.

Although the study is interesting in itself by providing raw data and on the different genres of blogs, it still lacks critical application (for me at least). The characteristics of blogs make them attractive for people to create their own, but what makes a blog attractive to others? Given that a certain percentage of blogs are by males, does this significantly change the popularity of the blog or increase discussion/interaction? Sadly, the implications for the percentages and the different types were largely missed in this paper.

At the conclusion, the authors strongly believe that blogs will eventually make differences between it and CMC obsolete. On one hand, I would agree because I’m trying to integrate my personal site and the blog together because of the blog’s flexibility. However, I am also trying to maintain it separate because the chronology isn’t as important and displaying other archival content like art and web design isn’t always the best with a blog post. Blog software communities like WordPress are trying to bridge that gap with modules that can implement specific content features only if an author needs it. With this trend growing, it does blur the definition of blog and CMC a bit more now for me.