
The paper discusses the CodeSaw, social visualization software development for distributed software community. The software visualizes the distributed software development from two different perspectives: code repositories and project communication. As stated by the paper, open source software development differs from traditional development by lacking the face-to-face interactions, not having any plan, and having minimum schedule.

Although there exists some project visualization software, CodeSaw differ from them. SeeSys uses tree-map visualization of the source code file system but is limited to very large projects with relatively stable contents. Auger uses line-oriented style presenting one user. CodeSaw is different from both by abstracting the information of one year of a project, showing all people involved, and having a spatial messaging. The software is based on the result of the initial interviews that usually the developers of the open-source community feel disconnected from the rest, and they use project mailing list to maintain awareness. Since most of the open-source communities are limited to 8 active developers, CodeSaw shows only up to 8 developers. However, as partially mentioned by the paper, it doesn’t consider other people who help the development by sending feedbacks and patches, reporting bugs, or solving special problems with specific part of the code with limited one-time writing a vital code. I like the fact that the research focuses on the share of the developers in the development, than the project itself, but since the factor for measuring the participation is the raw number of lines of written code, I think this doesn’t reflect the skill of the developer and the real share of development. As an example, more skillful programmers use smaller but more complex algorithms to solve the same problems. This way, their overall share of their contribution would be not reported correctly. Besides, the amount of contribution strongly depends on the part they are developing. One might solve an unsolved problem in development by spending much time but the result could be developed in only few lines of code. I think more advanced research could be done on finding the best measurement factor of the share of each user in overall development than just using the number of code lines.

It is very interesting to see how the system is capable of extracting social information from the visualization results, e.g. the fact that one contributor only writes code in the summer can be used to guess that he is a student with more free time in the summer. This information could be used in finding who are the main developers of the open-source software projects and how can the project coordinators help them by knowing these facts, e.g. in case of scheduling, knowing the fact that one is a student could guide the coordinator to assign him less tasks during the examination weeks.

It is also very thoughtful of the authors to use the frequency of appearance of words in the emails to visualize the communications. I think this is the best way to minimize the large pile of communication data into specific key words.

I think one of the highlight of the research is using the fact that using the small multiples are the best solution in visualizing. However, this would limit the application of the system into limited time range and groups with small users. Further studies should be done in expanding the system into more crowded groups and over larger time period. One solution is using the same technique but relying on grouping of users rather than focusing on each. As an example in a large project with multiple participants, instead of focusing on each person, people could be grouped by the area they are working, e.g. User Interface developers, Data Miners ... then in case user wants to get more details, he should be able to focus on each group and then see the result for each person.
Social Visualization Course Assignment: “Reading 03”

Also the research was successful in achieving good balance between revealing private information to the world and leaving enough of it ambiguous so as to not invade privacy.

It was also thoughtful of researchers to add spatial messaging, allowing users to leave comments on the visualization itself. Also generating automatic mail to the mailing lists regarding the spatial data was also a very strong solution for not introducing a parallel communication tool.

In summary, CodeSaw is an innovative tool to encourage developers to participate more in open-source software development by showing their participation share in overall project.

**Review of “Bridging the Gap: A Genre Analysis of Weblogs”**

The paper reviews Weblog as the frequently modified web pages in which dated entries are listed in reverse chronological sequence. Blogging is a simple tool for everyone without any specific computer skill to express himself and could have the social affect of making people more thoughtful and articulate observers of the world around them. The paper tries to provide an empirical snapshot of current stage of blogging for future purposes. It introduces weblogging as a bridge between multimedia HTML documents and text-based computer-mediated communication. It reviews different common types of weblogs, such as: Filter blogs, where author pre-surf the web and directs readers to selected content, Personal Journal, where author express his thoughts and internal workings, and notebooks, where author write longer focused essays. Previous researches predict that although earlier blogs were filter blogs, journal type would become more popular.

To study the issue, authors selected randomly sample weblogs and analyze them from different perspectives. Sample weblogs were filtered in a way not to include blogs with languages other than English, less amount of text, use of blogging for other purposes such as community center events announcements. Also not active blogs or the one with very few posts were excluded. I think the authors should also consider the fact that part of popularity of weblogs is based on the number of pages being created. So there exist lots of people who create a weblog and then rarely update it. It was better if a sample of random weblogs was selected without any filtering and then the results were compared for both filtered and not filtered cases. This way, the reader gains more insight about what percentage of weblogs of the sample have been filtered.

Also during the study, the intervals of update between the last 2 posts were calculated. I think in most of the weblogs were authors update the pages less frequently and daily, whenever he updates, he posts more than one post to his page. So using only the last two posts to calculate the frequency of updates might result in frequency of several times a day. However, in reality the author might update the page every 3-4 days but each time posts more than one post. So I think calculating the average of time intervals between all posts in a weblog is the best way to study the frequency of update.

The interesting fact, that 40% of bloggers are teenagers, show that new generation gets involved with the new technology more easily. This could be even increased considering the fact that many of the remaining 60% are also around 20 years old. One reason for this could be the fact that older generation has already found their own way of expressing themselves, as an example through homepages, or commenting about general topics, writing in the local papers or using part of their homepages for this. Getting used to a method would make it harder for them to switch to something new. On the other hand, younger generation who are seeking the best way to introduce and express themselves would find
weblogging the newer and easier tool for this. This is supported by the fact that in most blogs the identity of the authors are apparent since they want to express themselves.

Paper also discusses the fact that only about half of the blogs link to other blogs. This in my opinion could partially be because of the lack of skill among the young authors about how to insert a link into a post. So a good way to test this could be checking the percentage of blogs with at least only one link. This way the existence of one link shows that the author knows how to put a link inside but instead decided not to put any rather than not knowing how to insert one.

Also it is indicated in the paper that in contrast to our belief, weblogs are not as interactive as we thought. The average of .3 comments per post shows that there are only few weblogs supporting commenting and even among those, few people post comments. I also think it should be considered that most related weblogs are having link to each others. This means that author have some kind of interest or similarity with the author of the linked website, so instead of posting a comment to ones weblog, he might post a new post in his own weblog as a response to the post in the other weblog. This way not only he has expressed himself but also have used his own developed tool for replying, which is logical because many of the posts are expressing ones beliefs about facts or stuffs he read on others’ weblogs.

**Review of “Flash Forums and ForumReader: Navigating a New Kind of Large-scale Online Discussion”**

The paper reviews flash forum, conversations that are distinguished by their lack of authorship information, large size, tight overall focus with overlapping between threads’ topics, and constrained short time-frame for conversations. Two examples of that are Slashdot.org and Jams by IBM. In both examples readers are able to post comments during a limited time. The authors develop a tool called ForumReader in order to help orient users and guide them to their areas of interests within the flash forums. This developed tool combines data visualization with automatic topic extraction.

Paper studies the goals of users in interacting with a large discussion as to find ideas or information, receive support, take the pulse of a community, and meet people. The paper added that time limits have the powerful effect of spurring people to post sooner, knowing that posts late in the discussions are unlikely to elicit responses or be read as the forum approaches closure. I would like to add that although at the beginning of a discussion the limited timeframe of discussion could result in more and faster initial comments, while reaching the end of timeframe, it could also be a preventive tool for readers to post. I believe that after certain period of time, readers would hesitate to post, since they know that the time of discussion is going to end soon (e.g. is in its final half part) and probably no one would response to their post. I think further studies should be conducted on studying the overall effect of applying time frame to discussion on number of comments being posted.

The proposed ForumReader offers different types of visualization techniques through having thumbnail view, highlight view, test view, search view and tree view. Thumbnail view uses rectangles to represent each message in the discussion, drawing them in depth-first order with indentation indicating thread depth and rectangle height corresponding to message length. The first revision of the ForumReader was called ValuesJam used by IBM. Based on feedbacks on ValuesJam, slashdog was developed by adding two features: read wear and color coding for the text in the tree widget. The result shows that 16% of users were able to use ValuesJam to find their areas of interest in the Jam. User’s goal to use the system
was mainly to find interesting discussion threads or find well thought out individual posts. Also the result from the users of Slashdog tool shows that adding visualization to a no-text analytics condition improved performance, whereas adding visualization to a text-analytics condition reduced performance.

Reports of people also show that people use the color-by-author feature to primarily look for their own posts. One of the lessons drawn from Jam deployment and experimental evaluation is that most useful kind of data is about how much others find a message valuable. The Slashdog moderation values were considered important by almost all of the subjects.