1 CodeSaw: A Social Visualization of Distributed Software Development

CodeSaw is an experimental visualization of social activity in open-source projects. For its corpus it uses two main sources of information: the code repository for the project, and the related mailing list(s). The visualization indicates the activity of the core contributors both in terms of code contributions and in terms of amount of communication. The authors released the software publicly and used a set of developers as a test set. Most of the users indicated that the visualization was useful and that they would use it again, especially or only near release time.

I though the most insightful and useful part of this visualization was putting the two indicators on the same (time) axis. This does two things: it helps indicate when the project was active (and for whom), and possibly at what stage; when a feature is being introduced, for instance, one would expect spikes in email followed by spikes in code commits. The other useful part that results from using both statistics is that by using small multiples, it is easy to tell what the roles of the core contributors are. In some projects, it is just as important to have active thinkers, managers, and problem-solvers as it is to have code monkeys. Knowing who is who for a project could help both new and long-time contributors in communicating effectively. It also provides a rich aesthetic history of the project’s progress.

One of the shortcomings of the study was its inclusion of only the core developers. I understand this may have been due to restraints on the size of the data set or inaccessability of data, but many projects have significant contributions from developers who are not core members. Relating the contributions of interlopers with releases, for instance, could be interesting. Another useful addition would be the ability to group contributions by a set of files (indicating a part or component of the code), while simultaneously organizing the email content by keywords or a filter, to further distinguish the developers’ roles in parts of large projects. Linux kernel developers, for instance, may just touch one part of the filesystem code, or may just work on one driver. Also, some code that is committed is later taken out.

Lastly, the social messaging was a kind of interesting addition, but I’m not sure that it would be of practical use for communication if the test users felt they would only use the software occasionally or at release time. The feature could be useful as a tagging mechanism, to go back and look at the project’s visual history and tag a past release with further information.
Bridging the Gap: A Genre Analysis of Weblogs

This paper (from 2003, though it doesn’t mention what months) was about an attempt to analyse a sample of weblogs to determine their genre makeup and other characteristics. The paper challenges assumptions about the origins and content of blogs from the point of view of other authors. They separated blogs into four categories, on the two axes of personal/topical and individual/community. They found a small occurrence of community blogs, and the majority of blogs were personal journals. What the paper calls “filter” blogs we would probably call “linkblogs.” The conclusion of the paper was that based simply on this genre analysis, the history of blogs was more likely from the personal diary than from linkrolls.

First, I thought the sampling method was interesting in what it did and did not take into account as bias. The authors chose blogs from a service which aggregated blogs from a variety of other services, which probably meant that the blogs they collected had been updated lately. Also, the study rejected blogs that had not posted within two weeks prior to the study. However, I didn’t feel that these conditions were really sufficient to rule out bias introduced by low-frequency blogs. It would have been more interesting to weigh the final results proportional to total blog volume or to post frequency. My guess is that personal journals would not be as well represented if this were the case.

The statistical connections drawn were interesting, especially the statistic about personal information on blogs. One thing that was confusing was that their conclusion claimed that blogs were both more commonly personal journals and probably descended from diaries rather than “filters,” but for much of the analysis the assumption about the early prevalence of filter-style blogs is assumed. In any case, I felt that for many blogs it would be oversimplifying to give them a single category. Many blogs alternate between personal entries, links, and topical articles.

The authors briefly mention that personal home pages probably were part of the history of blogs, but I felt this connection was understated. A home page used to be a place where you kept both personal information, a summary of current activity, and often lots of links to sites you found interesting. This is still basically what a lot of blogs do, they just do it on a rolling basis that is able to respond to other bloggers and current events. The way blogging sites advertise for bloggers is very similar to the way Geocities and the like used to advertise for webpage “owners.”