Clay Shirky's Writings About the Internet

Economics & Culture, Media & Community

clay@shirky.com

Group as User: Flaming and the Design of Social Software

First published November 5, 2004 on the "Networks, Economics, and Culture" mailing list.
Subscribe to the mailing list.

When we hear the word "software," most of us think of things like Word, Powerpoint, or Photoshop, tools for individual users. These tools treat the computer as a box, a self-contained environment in which the user does things. Much of the current literature and practice of software design -- feature requirements, UI design, usability testing -- targets the individual user, functioning in isolation.

And yet, when we poll users about what they actually do with their computers, some form of social interaction always tops the list -- conversation, collaboration, playing games, and so on. The practice of software design is shot through with computer-as-box assumptions, while our actual behavior is closer to computer-as-door, treating the device as an entrance to a social space.

We have grown quite adept at designing interfaces and interactions between computers and machines, but our social tools -- the software the users actually use most often -- remain badly misfit to their task. Social interactions are far more complex and unpredictable than human/computer interaction, and that unpredictability defeats classic user-centric design. As a result, tools used daily by tens of millions are either ignored as design challenges, or treated as if the only possible site of improvement is the user-to-tool interface.

The design gap between computer-as-box and computer-as-door persists because of a diminished conception of the user. The user of a piece of social software is not just a collection of individuals, but a group. Individual users take on roles that only make sense in groups: leader, follower, peacemaker, process nazi, and so on. There are also behaviors that can only occur in groups, from consensus building to social climbing. And yet, despite these obvious differences between personal and social behaviors, we have very little design practice that treats the group as an entity to be designed for.

There is enormous value to be gotten in closing that gap, and it doesn't require complicated new tools. It just requires new ways of looking at old problems. Indeed, much of the most important work in social software has been technically simple but socially complex.

Learning From Flame Wars

Mailing lists were the first widely available piece of social software. (PLATO beat mailing lists by a decade, but had a limited user base.) Mailing lists were also the first widely analyzed virtual communities. And for roughly thirty years, almost any description of mailing lists of any length has mentioned flaming, the tendency of list members to forgo standards of public decorum when attempting to communicate with some

ignorant moron whose to stupid to know how too spell and deserves to DIE, die a PAINFUL DEATH, you PINKO SCUMBAG!!!

Yet despite three decades of descriptions of flaming, it is often treated by designers as a mere side-effect, as if each eruption of a caps-lock-on argument was surprising or inexplicable.

Flame wars are not surprising; they are one of the most reliable features of mailing list practice. If you assume a piece of software is for what it does, rather than what its designer's stated goals were, then mailing list software is, among other things, a tool for creating and sustaining heated argument. (This is true of other conversational software as well -- the WELL, usenet, Web BBSes, and so on.)

This tension in outlook, between 'flame war as unexpected side-effect' and 'flame war as historical inevitability,' has two main causes. The first is that although the environment in which a mailing list runs is computers, the environment in which a flame war runs is people. You couldn't go through the code of the Mailman mailing list tool, say, and find the comment that reads "The next subroutine ensures that misunderstandings between users will be amplified, leading to name-calling and vitriol." Yet the software, when adopted, will frequently produce just that outcome.

The user's mental model of a word processor is of limited importance -- if a word processor supports multiple columns, users can create multiple columns; if not, then not. The users' mental model of social software, on the other hand, matters enormously. For example, 'personal home pages' and weblogs are very similar technically -- both involve local editing and global hosting. The difference between them was mainly in the user's conception of the activity. The pattern of weblogging appeared before the name weblog was invented, and the name appeared before any of the current weblogging tools were designed. Here the shift was in the user's mental model of publishing, and the tools followed the change in social practice.

In addition, when software designers do regard the users of social software, it is usually in isolation. There are many sources of this habit: ubiquitous network access is relatively recent, it is conceptually simpler to treat users as isolated individuals than as social actors, and so on. The cumulative effect is to make maximizing individual flexibility a priority, even when that may produce conflict with the group goals.

Flaming, an un-designed-for but reliable product of mailing list software, was our first clue to the conflict between the individual and the group in mediated spaces, and the initial responses to it were likewise an early clue about the weakness of the single-user design center.

Netiquette and Kill Files

The first general response to flaming was netiquette. Netiquette was a proposed set of behaviors that assumed that flaming was caused by (who else?) individual users. If you could explain to *each* user what was wrong with flaming, *all* users would stop.

This mostly didn't work. The problem was simple -- the people who didn't know netiquette needed it most. They were also the people least likely to care about the opinion of others, and thus couldn't be easily convinced to adhere to its tenets.

Interestingly, netiquette came tantalizingly close to addressing group phenomena. Most versions advised, among other techniques, contacting flamers directly, rather than replying to them on the list. Anyone who has tried this technique knows it can be surprisingly effective. Even here, though, the collective drafters of

netiquette misinterpreted this technique. Addressing the flamer directly works not because he realizes the error of his ways, but because it deprives him of an audience. Flaming is not just personal expression, it is a kind of performance, brought on in a social context.

This is where the 'direct contact' strategy falls down. Netiquette docs typically regarded direct contact as a way to engage the flamer's rational self, and convince him to forgo further flaming. In practice, though, the recidivism rate for flamers is high. People behave differently in groups, and while momentarily engaging them one-on-one can have a calming effect, that is a change in social context, rather than some kind of personal conversion. Once the conversation returns to a group setting, the temptation to return to performative outbursts also returns.

Another standard answer to flaming has been the kill file, sometimes called a bozo filter, which is a list of posters whose comments you want filtered by the software before you see them. (In the lore of usenet, there is even a sound effect -- *plonk* -- that the kill-file-ee is said to make when dropped in the kill file.)

Kill files are also generally ineffective, because merely removing one voice from a flame war doesn't do much to improve the signal to noise ratio -- if the flamer in question succeeds in exciting a response, removing his posts alone won't stem the tide of pointless replies. And although people have continually observed (for thirty years now) that "if everyone just ignores user X, he will go away," the logic of collective action makes that outcome almost impossible to orchestrate -- it only takes a couple of people rising to bait to trigger a flame war, and the larger the group, the more difficult it is to enforce the discipline required of all members.

The Tragedy of the Conversational Commons

Flaming is one of a class of economic problems known as The Tragedy of the Commons. Briefly stated, the tragedy of the commons occurs when a group holds a resource, but each of the individual members has an incentive to overuse it. (The original essay used the illustration of shepherds with common pasture. The group as a whole has an incentive to maintain the long-term viability of the commons, but with each individual having an incentive to overgraze, to maximize the value they can extract from the communal resource.)

In the case of mailing lists (and, again, other shared conversational spaces), the commonly held resource is communal attention. The group as a whole has an incentive to keep the signal-to-noise ratio high and the conversation informative, even when contentious. Individual users, though, have an incentive to maximize expression of their point of view, as well as maximizing the amount of communal attention they receive. It is a deep curiosity of the human condition that people often find negative attention more satisfying than inattention, and the larger the group, the likelier someone is to act out to get that sort of attention.

However, proposed responses to flaming have consistently steered away from group-oriented solutions and towards personal ones. The logic of collective action, alluded to above, rendered these personal solutions largely ineffective. Meanwhile attempts at encoding social bargains weren't attempted because of the twin forces of door culture (a resistance to regarding social features as first-order effects) and a horror of censorship (maximizing individual freedom, even when it conflicts with group goals.)

Weblog and Wiki Responses

When considering social engineering for flame-proofed-ness, it's useful to contemplate both weblogs and

wikis, neither of which suffer from flaming in anything like the degree mailing lists and other conversational spaces do. Weblogs are relatively flame-free because they provide little communal space. In economic parlance, weblogs solve the tragedy of the commons through enclosure, the subdividing and privatizing of common space.

Every bit of the weblog world is operated by a particular blogger or group of bloggers, who can set their own policy for accepting comments, including having no comments at all, deleting comments from anonymous or unfriendly visitors, and so on. Furthermore, comments are almost universally displayed away from the main page, greatly limiting their readership. Weblog readers are also spared the need for a bozo filter. Because the mailing list pattern of 'everyone sees everything' has never been in effect in the weblog world, there is no way for anyone to hijack existing audiences to gain attention.

Like weblogs, wikis also avoid the tragedy of the commons, but they do so by going to the other extreme. Instead of everything being owned, nothing is. Whereas a mailing list has individual and inviolable posts but communal conversational space, in wikis, even the writing is communal. If someone acts out on a wiki, the offending material can be subsequently edited or removed. Indeed, the history of the Wikipedia , host to communal entries on a variety of contentious topics ranging from Islam to Microsoft, has seen numerous and largely failed attempts to pervert or delete entire entries. And because older versions of wiki pages are always archived, it is actually easier to restore damage than cause it. (As an analogy, imagine what cities would look like if it were easier to clean graffiti than to create it.)

Weblogs and wikis are proof that you can have broadly open discourse without suffering from hijacking by flamers, by creating a social structure that encourages or deflects certain behaviors. Indeed, the basic operation of both weblogs and wiki -- write something locally, then share it -- is the pattern of mailing lists and BBSes as well. Seen in this light, the assumptions made by mailing list software looks less like The One True Way to design a social contract between users, and more like one strategy among many.

Reviving Old Tools

This possibility of adding novel social components to old tools presents an enormous opportunity. To take the most famous example, the Slashdot moderation system puts the ability to rate comments into the hands of the users themselves. The designers took the traditional bulletin board format -- threaded posts, sorted by time -- and added a quality filter. And instead of assuming that all users are alike, the Slashdot designers created a karma system, to allow them to discriminate in favor of users likely to rate comments in ways that would benefit the community. And, to police *that* system, they created a meta-moderation system, to solve the 'Who will guard the guardians' problem. (All this is documented in the Slashdot FAQ, our version of Federalist Papers #10.)

Rating, karma, meta-moderation -- each of these systems is relatively simple in technological terms. The effect of the whole, though, has been to allow Slashdot to support an enormous user base, while rewarding posters who produce broadly valuable material and quarantining offensive or off-topic posts.

Likewise, Craigslist took the mailing list, and added a handful of simple features with profound social effects. First, all of Craigslist is an enclosure, owned by Craig (whose title is not Founder, Chairman, and Customer Service Representative for nothing.) Because he has a business incentive to make his list work, he and his staff remove posts if enough readers flag them as inappropriate. Like Slashdot, he violates the assumption that social software should come with no group limits on individual involvement, and Craigslist works better because of it.

And, on the positive side, the addition of a "Nominate for 'Best of Craigslist'" button in every email creates a social incentive for users to post amusing or engaging material. The 'Best of' button is a perfect example of the weakness of a focus on the individual user. In software optimized for the individual, such a button would be incoherent -- if you like a particular post, you can just save it to your hard drive. But users don't merely save those posts to their hard drives; they click that button. Like flaming, the 'Best of' button also assumes the user is reacting in relation to an audience, but here the pattern is harnessed to good effect. The only reason you would nominate a post for 'Best of' is if you wanted other users to see it -- if you were acting in a group context, in other words.

Novel Operations on Social Facts

Jonah Brucker-Cohen's [Bumplist](#) stands out as an experiment in experimenting the social aspect of mailing lists. Bumplist, whose motto is "an email community for the determined", is a mailing list for 6 people, which anyone can join. When the 7th user joins, the first is bumped and, if they want to be back on, must re-join, bumping the second user, ad infinitum. (As of this writing, Bumplist is at 87,414 subscribes and 81,796 re-subscribes.) Bumplist's goal is more polemic than practical; Brucker-Cohen describes it as a re-examination of the culture and rules of mailing lists. However, it is a vivid illustration of the ways simple changes to well-understood software can produce radically different social effects.

You could easily imagine many such experiments. What would it take, for example, to design a mailing list that was flame-retardant? Once you stop regarding all users as isolated actors, a number of possibilities appear. You could institute induced lag, where, once a user contributed 5 posts in the space of an hour, a cumulative 10 minute delay would be added to each subsequent post. Every post would be delivered eventually, but it would retard the rapid-reply nature of flame wars, introducing a cooling off period for the most vociferous participants.

You could institute a kind of thread jail, where every post would include a 'Worst of' button, in the manner of Craigslist. Interminable, pointless threads (e.g. Which Operating System Is Objectively Best?) could be sent to thread jail if enough users voted them down. (Though users could obviously change subject headers and evade this restriction, the surprise, first noted by Julian Dibbell, is how often users respect negative *communal* judgment, even when they don't respect the negative judgment of individuals. [See [Rape in Cyberspace](#) -- search for "aggressively antisocial vibes."])

You could institute a 'Get a room!' feature, where any conversation that involved two users ping-ponging six or more posts (substitute other numbers to taste) would be automatically re-directed to a sub-list, limited to that pair. The material could still be archived, and so accessible to interested lurkers, but the conversation would continue without the attraction of an audience.

You could imagine a similar exercise, working on signal/noise ratios generally, and keying off the fact that there is always a most active poster on mailing lists, who posts much more often than even the second most active, and much *much* more often than the median poster. Oddly, the most active poster is often not even aware that they occupy this position (seeing ourselves as others see us is difficult in mediated spaces as well,) but making them aware of it often causes them to self-moderate. You can imagine flagging all posts by the most active poster, whoever that happened to be, or throttling the maximum number of posts by any user to some multiple of average posting tempo.

And so on. The number of possible targets for experimentation is large and combinatorial, and those targets exist in any social context, not just in conversational spaces.

Rapid, Iterative Experimentation

Though most of these sorts of experiments won't be of much value, rapid, iterative experiment is the best way to find those changes that are positive. The Slashdot FAQ makes it clear that the now-stable ratings+karma+meta-moderation system could only have evolved with continued adjustment over time. This was possible because the engineering challenges were relatively straightforward, and the user feedback swift.

That sort of experimentation, however, has been the exception rather than the rule. In thirty years, the principal engineering work on mailing lists has been on the administrative experience -- the Mailman tool now offers a mailing list administrator nearly a hundred configurable options, many with multiple choices. However, the *social* experience of a mailing list over those three decades has hardly changed at all.

This is not because experimenting with social experience is technologically hard, but because it is conceptually foreign. The assumption that the computer is a box, used by an individual in isolation, is so pervasive that it is adhered to even when it leads to investment of programmer time in improving every aspect of mailing lists except the interaction that makes them worthwhile in the first place.

Once you regard the group mind as part of the environment in which the software runs, though, a universe of un-tried experimentation opens up. A social inventory of even relatively ancient tools like mailing lists reveals a wealth of untested models. There is no guarantee that any given experiment will prove effective, of course. The feedback loops of social life always produce unpredictable effects. Anyone seduced by the idea of social perfectibility or total control will be sorely disappointed, because users regularly reject attempts to affect or alter their behavior, whether by gaming the system or abandoning it.

But given the breadth and simplicity of potential experiments, the ease of collecting user feedback, and most importantly the importance users place on social software, even a few successful improvements, simple and iterative though they may be, can create disproportionate value, as they have done with Craigslist and Slashdot, and as they doubtless will with other such experiments.

First published November 5, 2004 on the "Networks, Economics, and Culture" mailing list.
Subscribe to the mailing list.

Clay Shirky's Writings About the Internet

Economics & Culture, Media & Community, Open Source

clay@shirky.com