P A U L   G R A H A M

WHAT I'VE LEARNED FROM HACKER NEWS

February 2009

Hacker News was two years old last week. Initially it was supposed to be a side project—an application to sharpen Arc on, and a place for current and future Y Combinator founders to exchange news. It's grown bigger and taken up more time than I expected, but I don't regret that because I've learned so much from working on it.

Growth

When we launched in February 2007, weekday traffic was around 1600 daily uniques. It's since grown to around 22,000. This growth rate is a bit higher than I'd like. I'd like the site to grow, since a site that isn't growing at least slowly is probably dead. But I wouldn't want it to grow as large as Digg or Reddit—mainly because that would dilute the character of the site, but also because I don't want to spend all my time dealing with scaling.

I already have problems enough with that. Remember, the original motivation for HN was to test a new programming language, and moreover one that's focused on experimenting with language design, not performance. Every time the site gets slow, I fortify myself by recalling McIlroy and Bentley's famous quote

> The key to performance is elegance, not battalions of special cases.

and look for the bottleneck I can remove with least code. So far I've been able to keep up, in the sense that performance has remained consistently mediocre despite 14x growth. I don't know what I'll do next, but I'll probably think of something.

This is my attitude to the site generally. Hacker News is an experiment, and an experiment in a very young field. Sites of this type are only a few years old. Internet conversation generally is only a few decades old. So we've probably only discovered a fraction of what we eventually will.

That's why I'm so optimistic about HN. When a technology is this young, the existing solutions are usually terrible; which means it must be possible to do much better; which means many problems that seem insoluble aren't. Including, I hope, the problem that has afflicted so many previous communities: being ruined by growth.

Dilution

Users have worried about that since the site was a few months old. So far these alarms have been false, but they may not

always be. Dilution is a hard problem. But probably soluble; it doesn't mean much that open conversations have "always" been destroyed by growth when "always" equals 20 instances.

But it's important to remember we're trying to solve a new problem, because that means we're going to have to try new things, most of which probably won't work. A couple weeks ago I tried displaying the names of users with the highest average comment scores in orange. [1] That was a mistake. Suddenly a culture that had been more or less united was divided into haves and have-nots. I didn't realize how united the culture had been till I saw it divided. It was painful to watch. [2]

So orange usernames won't be back. (Sorry about that.) But there will be other equally broken-seeming ideas in the future, and the ones that turn out to work will probably seem just as broken as those that don't.

Probably the most important thing I've learned about dilution is that it's measured more in behavior than users. It's bad behavior you want to keep out more than bad people. User behavior turns out to be surprisingly malleable. If people are expected to behave well, they tend to; and vice versa.

Though of course forbidding bad behavior does tend to keep away bad people, because they feel uncomfortably constrained in a place where they have to behave well. But this way of keeping them out is gentler and probably also more effective than overt barriers.

It's pretty clear now that the broken windows theory applies to community sites as well. The theory is that minor forms of bad behavior encourage worse ones: that a neighborhood with lots of graffiti and broken windows becomes one where robberies occur. I was living in New York when Giuliani introduced the reforms that made the broken windows theory famous, and the transformation was miraculous. And I was a Reddit user when the opposite happened there, and the transformation was equally dramatic.

I'm not criticizing Steve and Alexis. What happened to Reddit didn't happen out of neglect. From the start they had a policy of censoring nothing except spam. Plus Reddit had different goals from Hacker News. Reddit was a startup, not a side project; its goal was to grow as fast as possible. Combine rapid growth and zero censorship, and the result is a free for all. But I don't think they'd do much differently if they were doing it again. Measured by traffic, Reddit is much more successful than Hacker News.

But what happened to Reddit won't inevitably happen to HN. There are several local maxima. There can be places that are free for alls and places that are more thoughtful, just as there are in the real world; and people will behave differently depending on which they're in, just as they do in the real world.

I've observed this in the wild. I've seen people cross-posting on Reddit and Hacker News who actually took the trouble to write two versions, a flame for Reddit and a more subdued version for

HN.

Submissions

There are two major types of problems a site like Hacker News needs to avoid: bad stories and bad comments. So far the danger of bad stories seems smaller. The stories on the frontpage now are still roughly the ones that would have been there when HN started.

I once thought I'd have to weight votes to keep crap off the frontpage, but I haven't had to yet. I wouldn't have predicted the frontpage would hold up so well, and I'm not sure why it has. Perhaps only the more thoughtful users care enough to submit and upvote links, so the marginal cost of one random new user approaches zero. Or perhaps the frontpage protects itself, by advertising what type of submission is expected.

The most dangerous thing for the frontpage is stuff that's too easy to upvote. If someone proves a new theorem, it takes some work by the reader to decide whether or not to upvote it. An amusing cartoon takes less. A rant with a rallying cry as the title takes zero, because people vote it up without even reading it.

Hence what I call the Fluff Principle: on a user-voted news site, the links that are easiest to judge will take over unless you take specific measures to prevent it.

Hacker News has two kinds of protections against fluff. The most common types of fluff links are banned as off-topic. Pictures of kittens, political diatribes, and so on are explicitly banned. This keeps out most fluff, but not all of it. Some links are both fluff, in the sense of being very short, and also on topic.

There's no single solution to that. If a link is just an empty rant, editors will sometimes kill it even if it's on topic in the sense of being about hacking, because it's not on topic by the real standard, which is to engage one's intellectual curiosity. If the posts on a site are characteristically of this type I sometimes ban it, which means new stuff at that url is auto-killed. If a post has a linkbait title, editors sometimes rephrase it to be more matter-of-fact. This is especially necessary with links whose titles are rallying cries, because otherwise they become implicit "vote up if you believe such-and-such" posts, which are the most extreme form of fluff.

The techniques for dealing with links have to evolve, because the links do. The existence of aggregators has already affected what they aggregate. Writers now deliberately write things to draw traffic from aggregators—sometimes even specific ones. (No, the irony of this statement is not lost on me.) Then there are the more sinister mutations, like linkjacking—posting a paraphrase of someone else's article and submitting that instead of the original. These can get a lot of upvotes, because a lot of what's good in an article often survives; indeed, the closer the paraphrase is to plagiarism, the more survives. [3]

I think it's important that a site that kills submissions provide a way for users to see what got killed if they want to. That keeps editors honest, and just as importantly, makes users confident they'd know if the editors stopped being honest. HN users can do this by flipping a switch called showdead in their profile. [4]

Comments

Bad comments seem to be a harder problem than bad submissions. While the quality of links on the frontpage of HN hasn't changed much, the quality of the median comment may have decreased somewhat.

There are two main kinds of badness in comments: meanness and stupidity. There is a lot of overlap between the two—mean comments are disproportionately likely also to be dumb—but the strategies for dealing with them are different. Meanness is easier to control. You can have rules saying one shouldn't be mean, and if you enforce them it seems possible to keep a lid on meanness.

Keeping a lid on stupidity is harder, perhaps because stupidity is not so easily distinguishable. Mean people are more likely to know they're being mean than stupid people are to know they're being stupid.

The most dangerous form of stupid comment is not the long but mistaken argument, but the dumb joke. Long but mistaken arguments are actually quite rare. There is a strong correlation between comment quality and length; if you wanted to compare the quality of comments on community sites, average length would be a good predictor. Probably the cause is human nature rather than anything specific to comment threads. Probably it's simply that stupidity more often takes the form of having few ideas than wrong ones.

Whatever the cause, stupid comments tend to be short. And since it's hard to write a short comment that's distinguished for the amount of information it conveys, people try to distinguish them instead by being funny. The most tempting format for stupid comments is the supposedly witty put-down, probably because put-downs are the easiest form of humor. [5] So one advantage of forbidding meanness is that it also cuts down on these.

Bad comments are like kudzu: they take over rapidly. Comments have much more effect on new comments than submissions have on new submissions. If someone submits a lame article, the other submissions don't all become lame. But if someone posts a stupid comment on a thread, that sets the tone for the region around it. People reply to dumb jokes with dumb jokes.

Maybe the solution is to add a delay before people can respond to a comment, and make the length of the delay inversely proportional to some prediction of its quality. Then dumb threads would grow slower. [6]

People

I notice most of the techniques I've described are conservative: they're aimed at preserving the character of the site rather than enhancing it. I don't think that's a bias of mine. It's due to the shape of the problem. Hacker News had the good fortune to start out good, so in this case it's literally a matter of preservation. But I think this principle would also apply to sites with different origins.

The good things in a community site come from people more than technology; it's mainly in the prevention of bad things that technology comes into play. Technology certainly can enhance discussion. Nested comments do, for example. But I'd rather use a site with primitive features and smart, nice users than a more advanced one whose users were idiots or trolls.

So the most important thing a community site can do is attract the kind of people it wants. A site trying to be as big as possible wants to attract everyone. But a site aiming at a particular subset of users has to attract just those—and just as importantly, repel everyone else. I've made a conscious effort to do this on HN. The graphic design is as plain as possible, and the site rules discourage dramatic link titles. The goal is that the only thing to interest someone arriving at HN for the first time should be the ideas expressed there.

The downside of tuning a site to attract certain people is that, to those people, it can be too attractive. I'm all too aware how addictive Hacker News can be. For me, as for many users, it's a kind of virtual town square. When I want to take a break from working, I walk into the square, just as I might into Harvard Square or University Ave in the physical world. [7] But an online square is more dangerous than a physical one. If I spent half the day loitering on University Ave, I'd notice. I have to walk a mile to get there, and sitting in a cafe feels different from working. But visiting an online forum takes just a click, and feels superficially very much like working. You may be wasting your time, but you're not idle. Someone is wrong on the Internet, and you're fixing the problem.

Hacker News is definitely useful. I've learned a lot from things I've read on HN. I've written several essays that began as comments there. So I wouldn't want the site to go away. But I would like to be sure it's not a net drag on productivity. What a disaster that would be, to attract thousands of smart people to a site that caused them to waste lots of time. I wish I could be 100% sure that's not a description of HN.

I feel like the addictiveness of games and social applications is still a mostly unsolved problem. The situation now is like it was with crack in the 1980s: we've invented terribly addictive new things, and we haven't yet evolved ways to protect ourselves from them. We will eventually, and that's one of the problems I hope to focus on next.

Notes

[1] I tried ranking users by both average and median comment score, and average (with the high score thrown out) seemed the more accurate predictor of high quality. Median may be the more accurate predictor of low quality though.

[2] Another thing I learned from this experiment is that if you're going to distinguish between people, you better be sure you do it right. This is one problem where rapid prototyping doesn't work.

Indeed, that's the intellectually honest argument for not discriminating between various types of people. The reason not to do it is not that everyone's the same, but that it's bad to do wrong and hard to do right.

[3] When I catch egregiously linkjacked posts I replace the url with that of whatever they copied. Sites that habitually linkjack get banned.

[4] Digg is notorious for its lack of transparency. The root of the problem is not that the guys running Digg are especially sneaky, but that they use the wrong algorithm for generating their frontpage. Instead of bubbling up from the bottom as they get more votes, as on Reddit, stories start at the top and get pushed down by new arrivals.

The reason for the difference is that Digg is derived from Slashdot, while Reddit is derived from Delicious/popular. Digg is Slashdot with voting instead of editors, and Reddit is Delicious/popular with voting instead of bookmarking. (You can still see fossils of their origins in their graphic design.)

Digg's algorithm is very vulnerable to gaming, because any story that makes it onto the frontpage is the new top story. Which in turn forces Digg to respond with extreme countermeasures. A lot of startups have some kind of secret about the subterfuges they had to resort to in the early days, and I suspect Digg's is the extent to which the top stories were de facto chosen by human editors.

[5] The dialog on Beavis and Butthead was composed largely of these, and when I read comments on really bad sites I can hear them in their voices.

[6] I suspect most of the techniques for discouraging stupid comments have yet to be discovered. Xkcd implemented a particularly clever one in its IRC channel: don't allow the same thing twice. Once someone has said "fail," no one can ever say it again. This would penalize short comments especially, because they have less room to avoid collisions in.

Another promising idea is the stupid filter, which is just like a probabilistic spam filter, but trained on corpora of stupid and non-stupid comments instead.

You may not have to kill bad comments to solve the problem. Comments at the bottom of a long thread are rarely seen, so it may be enough to incorporate a prediction of quality in the comment sorting algorithm.

[7] What makes most suburbs so demoralizing is that there's no center to walk to.

Thanks to Justin Kan, Jessica Livingston, Robert Morris, Alexis Ohanian, Emmet Shear, and Fred Wilson for reading drafts of this.

[?] [Comment](#) on this essay.