

# Inside GitHub's Super-Lean Management Strategy--And How It Drives Innovation

Using a DIY management strategy pulled from the open-source world, GitHub has built a product that coders everywhere love. But can it actually work anywhere else?

By [Chris Dannen \(/user/chris-dannen\)](#)

LISTEN TO THIS STORY

▶ 00:00

48:05 ◀

Imagine you could run a company on autopilot: no tier of “managers,” just people creating value by doing what they love and letting the rest fall into place. How much money would you save by eliminating all that bureaucracy? How much faster could you move? How much conflict could you erase? How much bigger could you grow? How much more creative would the culture be?

There is at least one company that has been experimenting to get closer to the ideal: GitHub.

If you're not familiar with [GitHub](#) (<http://github.com>), ask a programmer. GitHub is a collaboration platform for software developers, used by 3 million people and quickly getting adopted by businesses. *BusinessWeek* [called](#) (<http://www.businessweek.com/articles/2013-06-20/github-got-silly-rich-dot-next-step-make-more-awesome>) GitHub CEO Tom Preston-Warner “Grand Poobah of the

software universe” back in June, not long after the company raised \$100M to go after enterprise software development, mostly from Andreessen-Horowitz.

GitHub is a privately owned, for-profit company, but it’s built (like a lot of software companies) on an open-source project ([https://en.wikipedia.org/wiki/Open\\_source](https://en.wikipedia.org/wiki/Open_source)). Theirs is called Git, originally authored by Linus Torvalds, the dude who made Linux, and released under GNU General Public License in 2008 ([https://en.wikipedia.org/wiki/Git\\_\(software\)](https://en.wikipedia.org/wiki/Git_(software))).

What GitHub has done is taken the practice of open-source collaboration--which is done on something akin to a volunteer basis--and applied them to the organization of their entire company. And the outcome has been a product that is universally beloved and relied upon among technology-industry types and university computer science groups, which are bunches of people who can be (ahem) somewhat picky. The question I was interested in when I started this story was: What is the connection between the structure of the startup and the quality of its product?

This is a story about how to run a company, but more specifically, how to run a company that embraces open-source culture--and how the seemingly counter-intuitive principles behind open-source culture seem to be good for GitHub, which is growing quickly, and also apparently just good for its product. And its marketing. And its hiring strategy. And its culture.



## The Formula Is Open Allocation

At GitHub, people work on an open allocation (<http://www.forbes.com/sites/quora/2012/11/26/what-is-open-allocation/>) basis. Unlike traditional companies where projects are assigned top-down, GitHubbers tackle whatever projects they want, without any formal requests or managerial interference. Sure, GitHub is only 175 employees, so there are limitations to this experiment, but Valve Software (400 employees) has grown to be a \$2.5B company with a very similar open allocation structure (<http://michaelochurch.wordpress.com/2012/08/09/dont-look-now-but-valve-just-humiliated-your-corporate-culture/>). Is this too good to be true?

Sure, GitHub is a special case--the company makes collaboration software for developers, so of course they're going to be especially good at collaborating, aren't they? The question this story seeks to answer is: What makes this strategy effective for GitHub, and can it actually work anywhere else?

## Why Company Structure Matters

Since humans first figured out agriculture, we have been pondering what to do with surpluses. But the kind of surplus that most evades us is the surplus of creativity that exists in all our brains. Pundits have mythologized the topic, but only abstractly: There's Google's now-defunct concept of 20% time

([http://www.huffingtonpost.com/2013/08/16/google-20-percent-time\\_n\\_3768586.html](http://www.huffingtonpost.com/2013/08/16/google-20-percent-time_n_3768586.html))

and Clay Shirky's idea that free time leads people to create valuable things

(<https://en.wikipedia.org>

[/wiki/Cognitive\\_Surplus](https://en.wikipedia.org/wiki/Cognitive_Surplus)). But these concepts conflict directly with common sense. You can't just let people do whatever they want all the time and expect to get anything valuable done. Right?

Well, as it turns out, these working conditions can attract some creative people. When I visited GitHub's office at the end of the summer, I met a design and UX researcher named Chrissie Brodigan, who had recently been hired away from Mozilla (<https://en.wikipedia.org/wiki/Mozilla>), the free-software community that makes the Firefox web browser, to do a new, clever form of UX evaluation called deprivation testing (<http://www.fastcolabs.com/3010972/open-company/how-github-uses-deprivation-testing-to-hone-product-design>).

When I spoke to employees, they told me the most attractive thing about open allocation was its accessibility--they liked the option to work with whomever they liked on something they thought was cool. (And that includes the specifications of the company's uber-programmable new office, which they moved into this summer--but we already wrote that story here (<http://www.fastcolabs.com/3014552/open-company/lessons-in-software-architecture-from-githubs-new-office>).)

## The Four Big Problems With Innovation

Why mess with company organization anyhow?

Open allocation is one theory that answers one of the most crucial questions for any company: how to innovate and make that innovation repeatable.

There are four essential problems with systematized innovation, and in most businesses, structure can be a solution.

McKinsey says that 62% of corporate executives report manipulating company structure to drive innovation efforts ([http://www.mckinsey.com/insights/innovation/making\\_innovation\\_structures\\_work\\_mckinsey\\_global\\_survey\\_results](http://www.mckinsey.com/insights/innovation/making_innovation_structures_work_mckinsey_global_survey_results)).

But structure is particularly fun to talk about for startups; the same McKinsey paper says that people experiment more with structure when they're trying to make something new.

Traditional companies often have small units that are responsible for "innovating" while the rest of the company toils away under a more hierarchical system. What's fascinating about GitHub's approach is that it applies a single innovation-centric model across the entire organization. In most extant companies, the people who move up the chain are selected by a competitive grading process--Google's internal 4.0 grading system and Microsoft's stack ranking are two examples.

But surely those companies would never have implemented such rigid structures if they didn't work, would they? And what's so hard about innovating, anyway? Why aren't there more Apples and less RIMs? Why do most corporations peak and collapse inside 50 years (<http://www.businessweek.com/chapter/degeus.htm>), not (say) 250 years?

Let's say you're a startup founder. You had a great idea and a pool of interested users, and through vigorous iteration, you managed to achieve product market fit (<http://www.fastcolabs.com/3014841/why-you-should-find-product-market-fit-before-sniffing-around-for-venture-money>). With money in the

bank, you start hiring. With all your new resources, you want to keep reproducing the process that got you success in the first place. This is called “innovating,” a word that has begun to lose meaning, we hear it so often. I like this definition I found in an academic paper from 1986 (<http://www.d.umn.edu/%7Erdtaylor/p1.pdf>), back in the golden age of management theory:

The process of innovation is defined as the development and implementation of new ideas by people who over time engage in transactions with others within an institutional context.

In other words: How do you get the same old people in the same old company to keep cranking our novel, valuable things?

As soon as you try to reproduce your successes, you run into four problems, which are succinctly outlined in that same paper I linked above. I'll paraphrase them here.

**Managing Attention:** People and their organizations are largely designed to focus on, harvest, and protect existing practices rather than pay attention to developing new ideas. The more successful an organization is, the harder it is for them to pay attention to new ideas, needs, and opportunities.

**Creating Currency:** While the invention or conception of innovative ideas may be an individual activity, it takes a whole company to execute--and it's hard to get a big group behind a new idea because of social and political dynamics.

**Getting Units To Work Together:** As a new product or service comes to fruition, ideas, people, and transactions proliferate quickly, which means you need more people to pitch in



and help. The more individuals get involved, the easier it is for them to individually lose sight of the whole innovation effort.

**Institutionalized Leadership:** Some innovations are so vital that they rightfully require the structure and practices of management to change radically. Most companies' infrastructure isn't flexible this way, making it hard for structure to keep up with innovation.



## What The Next Generation Of Companies Needs To Know

There are other forces at work that are making it imperative to test out this whole "company structure" hypothesis.

- Young employees--Millennials--don't respect hierarchy for the sake of itself (<http://www.fastcompany.com/1720052/googles-greatest-innovation-may-be-its-management-practice>). They want fast-moving, lean platforms on which to build their work. They'd rather leave than fight an organization into tolerating them, because they always wanted an excuse to go freelance.
- Barriers to entry keep getting lower. Big software companies throw away

billions on R&D on floundering products, while small, capable teams are creating enormously valuable businesses with just a few million bucks in seed capital in software and (increasingly) in hardware. Diverting your human resources to explore new experiments has never been easier.

So while this story is about GitHub, what the team there has learned could apply to almost any creative organization--design, fashion, publishing, advertising, software, consulting--that is trying to get its best work out of a bunch of unruly twenty- and thirty-somethings who want to move fast and try out new, promising ideas.

## The Key Ingredient To Open Allocation Is Communication

There are a few logical leaps between company structure and innovative products. After poking around GitHub, I realized that once you have an "open allocation" management structure, you still need something else: *open, easy-to-use internal platforms* and *stellar communication*.

If anyone can join any project, then workers need readily accessible training materials and documentation--otherwise, switching projects comes with too much friction as the new person struggles to get oriented.

In some ways, the structure of a company like GitHub is just a distribution network for information about *the purpose of this company and how it works*--and subsequently *what it should be making*.

As you'll see in the interviews below, the "mission" comes from the people at the top, and it's up to the rest of the self-organizing masses to figure out how to execute their part.

Keeping everyone on the same page about the



incremental adjustments in direction is really important--otherwise, word about these small corrections doesn't make it to certain units of the business, which don't adjust, eventually leading to huge gulfs in purpose between teams. That's how you end up like Apple in the 1980s ([https://en.wikipedia.org/wiki/History\\_of\\_Apple\\_Inc.](https://en.wikipedia.org/wiki/History_of_Apple_Inc.)): one team building a revolutionary product like the Macintosh and another team slogging away at an overpriced failure like the Apple III.

Good communication also makes it easier to start new, adjacent businesses. When word of a cool new project reaches the top of the company, the leaders can use it to make a course correction, which is then redistributed throughout the network so that everyone is made aware that "New Initiative X" is now part of the business.

As one GitHubber will talk about later in this story, this process was exemplified when the company bought a 3-D printer and he slowly began spending all his time with it. Now his main project involves making GitHub repositories better handle 3-D file collaboration. He's even spent some time connecting the 3-D printer to Hubot, the company's internal chat bot (also open source [here \(http://hubot.github.com/\)](http://hubot.github.com/)), so that GitHubbers can ask Hubot to 3-D print things or update them on the status of a print job.



## Three Ways Good Communication Prevents Problems

Communication isn't just key to self-organization--it also solves or simplifies a bunch of other hurdles that growing companies face.

1. **Training is easier.** When communication--not hierarchy--is the glue that holds businesses units together, it's easy for new employees to be themselves and feel like their interests, projects, and initiatives belong at the company, and it allows them to start creating valuable work right away.
2. **Marketing collateral is a natural by-product.** When your company communicates internally with polished, clear, and well-produced content, it is easy to repurpose that material for external communications. The kind of communication that is required for self-organization will end up producing all the events, schwag, and content you need to build and publicize an authentic brand. (More on how

that works at GitHub below.)

### 3. **Hiring becomes more focused.**

Instead of trying to find a candidate with exact skills to fill a specific role, open allocation management allows people who are hiring to select for only one essential attribute--the ability to be a self-starter.

## Lessons From Open Allocation: Interviews With Six GitHubbers

The rest of this monstrous, 8,000-word piece consists of interviews with GitHub employees and founders about how open allocation takes care of innovation, training, marketing, and hiring--with far less overhead than might be required in a traditionally organized company. This results in what is known as an "emergent organization ([https://en.wikipedia.org/wiki/Emergent\\_organization](https://en.wikipedia.org/wiki/Emergent_organization))."

## Speakers

**Tom Preston-Werner**

([//www.fastcompany.com/person](https://www.fastcompany.com/person/tom-preston-werner)

[/tom-preston-werner](https://www.fastcompany.com/person/tom-preston-werner)): CEO and co-founder

**Scott Chacon** ([//www.fastcompany.com](https://www.fastcompany.com/person/scott-chacon)

[/person/scott-chacon](https://www.fastcompany.com/person/scott-chacon)): office 3.0 lead, CIO and co-founder

**Peter Furia** ([//www.fastcompany.com/person](https://www.fastcompany.com/person/peter-furia)

[/peter-furia](https://www.fastcompany.com/person/peter-furia)): Streaming Eagle member and video lead

**Mike Skalnik**: 3D printing lead developer

**Kami Richey**: community manager and events & sponsorships lead

**Liz Clinkenbeard**: communications

## Be Able To Summarize Your

# Company's Purpose In Just A Few Words

**Tom Preston-Warner:** We focus a ton on communication in this company. And all of the tools that we build are really about enhancing communication; that's what GitHub itself is all about. It's about enabling communication between a broader set of people than would otherwise be efficient.

It's a fan-out structure of strategic communication, so that each individual team leader knows, well, where is the company trying to go? It's gonna be pretty much impossible to convince someone in this company to do something that they don't think is legitimate, that they don't agree with, that they don't think is a good idea. So, so much of this [strategy] is about convincing people that the direction you want to go in--the strategic vision--makes sense.

There's a lot of internal pitching and selling on what you're trying to accomplish, and I spend a lot of my time doing that. I do that every Friday at our all-hands, that we call "beer-thirty." It happens at 4:30 p.m. every Friday, and that's a chance for me and some other people to communicate to the entire company what we're thinking about, from a strategic perspective, how we think about culture, how we think about hiring, how we think about approaching business, all of these things.

## Empower People Close To The Problem

**Scott Chacon:** We try to move decision-making capabilities to people closest to the problem. [Projects have] a "primary responsible person," or PRP, in open allocation lingo. The question is, how do you get there?

Instead of having more of top-down hierarchy, where there's this visionary [who dictates] where

we want the product or the company to go, and we delegate all of the work--we want to do the opposite, which is assume the people closest to the problem know the problem best and probably make the best decisions on it. We enable them to do so.

## Allow People To Work On What They Like

**Tom:** There's a parallel to it that is the style of organization that we have, and that's something called "open allocation." This is a fundamental part of how we work, and why being highly networked makes this style of organization better.

You first identify what's important to you as a company, through strategy, and then each team does that as well. They each say, "Here's the strategy, here's our product. What is the most important thing for us?" And that's gonna also be in the support group, the sales group, the system operations group. They all look at what's important to them, in order to achieve the strategic goals. So that's step one, identify what's important.

So then step two is within those teams, allocate themselves to whatever's important. So people are identifying things that interest them, and aligning to one of the things that's important. So that's why it's called open allocation. It's that, what you work on is up to you, within the bounds of what's important to the company. And that's the general case. When it's working well, people are aligning to what's most important.

## Only Hire People Like This...

**Peter Furia:** There are a number of companies that have this 20% rule where employees are encouraged to work on stuff they are personally excited about outside of their main responsibility 20% of the time. I think in some ways you can say GitHub has a 100% rule, which is basically

always be working on stuff you are personally excited about or that you find fulfilling. Find ways to make it fulfilling and exciting or allocate your time on stuff that you feel excited about.

Some people hear that and they think, "Oh God, that's chaos. Like how is anyone actually doing anything good for a company?" I think that we select people to bring on to the team, bring into the company, who are naturally self-motivated, naturally passionate about what they do and about technology. Everything will always have some aspect of it that will benefit the company or that someone believes will be of benefit to the company. I think that's just kind of a given. It's not like everyone says, "Oh great, well I'm going to go do cooking today," or something unrelated to collaboration. Everyone is working on something that they believe will benefit the company in some way.



## Make New Projects Easy

**Mike Skalnik:** It mostly started out just as a definitely a toy project; something who knows where this will go, and it's turned into, "Okay, it would be nice if we could collaborate on these files themselves." We started looking at that, and that fits in with the general idea of GitHub should be easy enough for everyone to use, not just

developers.

I think this is a realization that we've realized from collaborating on code, and that is starting to apply to other things: Collaborating on those files is important, and collaborating on anything all the time is always better than just doing something yourself. Making it easier for that to happen is always something that we care about.

I started splitting my time more between what I was originally working on and that, and as that progress has grown, I've definitely phased myself off from what I initially would have done.

That's changed what I work on at GitHub. Now I work on displaying different file formats: The three model formats that 3-D printers normally use to figure out how to print an object.

## Let People Self-Educate And Self-Train

**Tom:** The Internet is the best communication channel that has ever been made, and if you think about the Internet as really just a substrate for communication, then it's finally possible to build a company this way--an emergent organization. Everything that we do communication-wise is designed for mass consumption of people within the company.

## Everyone Should Be Able Make High-Quality Internal Media

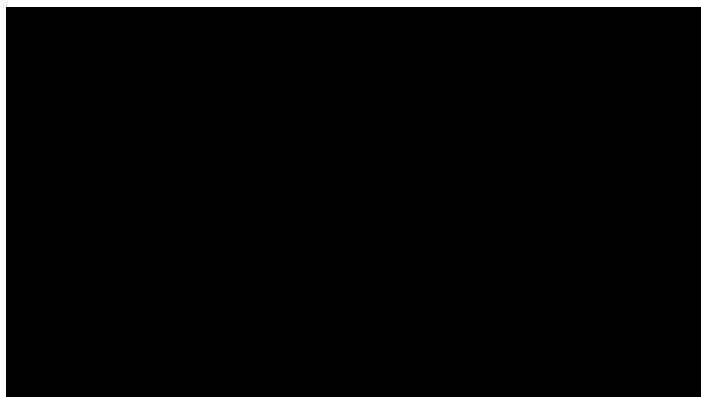
**Peter:** I was hired in February of this year. I've been here about four months. I think that I was brought on here to help lead the video charge in kind of beyond where it started, which was around a need to live-stream company events to its vast remote workforce for the most part. It's about 70% remote employees here. It started out, I think as a need to share all-hands meetings via live video streams, and then I came on with a background in video production, digital media, and advertising to see if I could kind of



bring some more of a filmmaking background as well as digital video and social video.

[Before GitHub] I started a company with my two lifelong best friends that started with a viral video and kind of serendipitously we'd end up building a business out of doing viral video for brands. Well, actually there is a lot of internal video that I'm working on, but there is also a lot of external video that is in its infancy but that I think a lot of people are really excited about.

The mission is constantly changing. The general philosophy here is one of open allocation, and I think it is a term a lot of people use, and so initially there was a need for streaming. Now there is clear a big appetite here for videos that will strengthen our culture as well as kind of share our culture and our companies philosophy with the rest of the world.



It's a pretty awesome assignment for me to come from a world of needing to really work with whatever brand came to us, regardless of how we may have felt personally about their company or brand. We had to try to make something that would get buzz or would be well received on the Internet. Now I have one client and that's GitHub.

It's kind of the way I like to think about it, I have one brand that I'm working for and it's a pretty awesome brand to be working for because a lot of the products people are working on they're generally excited about, and the focus is quality and making things great--making users' lives better as opposed to making money or trying to

target a demographic or some other things that you encounter when you are working in advertising.

I think that a huge part of [all this] is making sure people are always working on stuff they're personally excited about and invested in and I think that that just leads naturally into innovation. I think showing what people are working on is not only awesome because they're doing impressive stuff, but also because showing why they're excited about it, why they think it's an awesome feature or an awesome product.

GitHub also does something called "lightning profiles" of each of its new employees, because it allows them to get to know one another in a standardized, repeatable, scalable way. The first step of communication is an introduction; we've all been at parties where people mill about each other awkwardly, not feeling open to conversation because they weren't introduced.

## Allow People To Self-Organize In Chat

**Scott:** We have system for saying who's on what logical team, and we do have a number of teams.

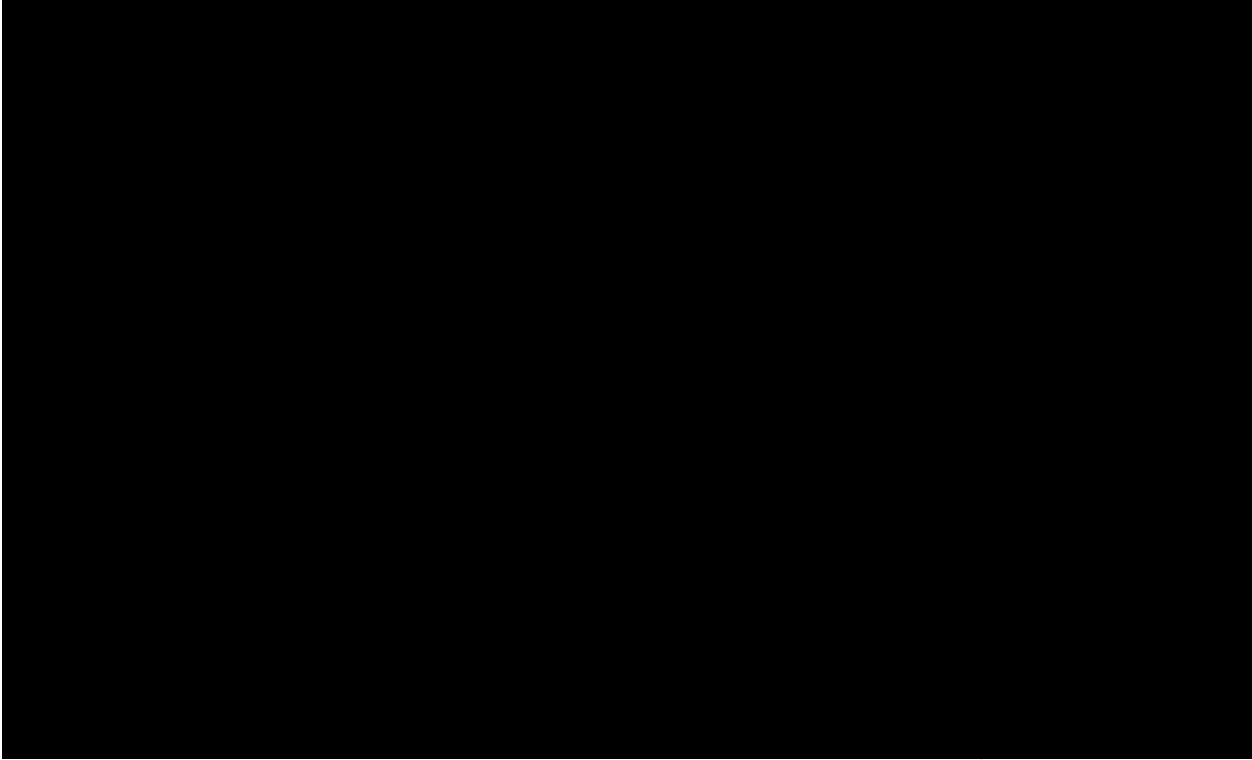
But really, [what project you're on] is based on what chat room you're in; everybody in the company uses chat primarily. Seventy percent of the company works remotely so our real office is in the chat rooms. We try to make everybody feel as much a part of the company as people that are in San Francisco or people that aren't. All of the real work, the real office is the chat room.

When you choose, there's a whole bunch, there's probably 70 or 80 rooms in our chat world, you align with the team when you choose which room to be involved in. That's how we figure out who's on what team, who is working on what problem--it's what chat rooms they are in every day.

## Value Storytelling

**Peter:** We've definitely got ambitious goals. I think we want to grow the team and we want to produce more and more content. I think one thing that we are excited about doing when we have our new office, we'll have a dedicated production studio and the ability to kind of run it like a miniature TV station, and it might be a lot more internal video, but I think we'd love to be doing weekly mini TV shows just for employees.

Comedy is central to a lot of our video content. I think that just to strip it down to the fundamentals, I think that comedy is fun and people like having fun while they're working if it's possible. Adding comedy to our video content is a way to take ideas or products or initiatives that otherwise might only be super interesting to people who really understand them well and making them understandable or interesting or compelling through stories and comedy.



Training videos allow anyone to teach themselves how to do anything inside the company. Some of these things are technical, but most are workflow-related. Training is a time-suck, and great content can serve as a platform for people to teach themselves. That

leaves you free to hire as many people as you want and know that the noobs won't wander around the office asking people how to turn on the bathroom light or work the coffee maker. They can slip right in and belong right away. One of the internal projects that I have been working on are what we call the Caffeine Training Series, and these are videos designed to educate employees about how to use the various caffeine-oriented things in the kitchen of GitHub, coffee and tea mainly.

There is another documentary series called *Passion Projects*. I don't know if you've heard about this, but a [GitHubber] named Julie Ann Horvath spearheaded a series of talks hosted by GitHub that basically profile awesome women in the Valley and in technology who are entrepreneurs or developers or engineers or coders or who are doing awesome stuff in tech.

In addition to this talk series we are producing some mini documentaries about these women where we kind of look at what's their Passion Project, what are they working on, how did they get into it. I think one of the goals is to inspire other women or girls to look at stuff in tech and engineering.

I'm shipping probably three to five lightning profile videos per week. We're doing our all-hands meeting every week. We have the Passion Projects once a month. I'd say we're, in terms of different types of content, there's probably like a 400% increase from last year to this year.

## Let People Learn When They Can

**Scott:** We can record things for people when they come on later, new hires and things like that, that they can go back and watch important talks or important all-hands meetings and that people can watch offline asynchronously. We do all-hands meetings every week, but people are all over the world, so there is no time that's good

for everybody.

Our video room in the new [office] is much larger and so it's more sound absorbent or soundproof. There's a floating floor and there's the pike walls which are like the curved walls so you can shoot stuff that doesn't look like it had an ending in it, and green screen capabilities.

We have like lighting rack; we can move lights around on a viewing optic rack. It's a built for production, actual long-term production stuff. We'll probably have desks and chairs and things like we have in the current room, but it will be a lot more configurable, so we'll be able to do different types of setups rather than just that one.

There will also a small room within that room that has all of the post-production capabilities including audio post-production and things like that. The room will be sound proof enough that we can do high-quality audio recording; there's also three rooms that are set aside for podcast or Skype stuff, so they'll have better mic filtering systems, things like that--hardware compressors--because we do a lot of Skyping for interviewing and hiring sort of stuff.

## Don't Bother Trying To Be Flat

**Tom:** I don't think about [GitHub's structure] as being "flat." I think about it as being highly networked. And that means that instead of everyone being the same, you embrace that everyone is different, and you look at the strength of connections between people, the communication channels, and how information travels amongst them, and then you can draw a diagram. Instead of saying "Okay, everybody's the same, here's the org-chart, it's just a flat line," you say the org-chart really looks like a neural network, where you have nodes in a graph and then connections between them, and they go everywhere. Everyone talks to anyone.

# Understand Open-Source Philosophy

**Tom:** I think it helps if you understand how open-source works and that, in an open-source community, you don't tell people what to do, you present something that's really great to them, and they say "Wow, this is very cool, I want to help out, inform me, and help me see what's important to this project so that I can put my work to the best use." That's open-source, and that really is also open allocation.

Everyone that works here opted in by wanting to be hired here. In us hiring them, they're saying, "I want to work on this project called GitHub." And so that's the most natural and easy way to do [other projects].

## Plan Real-Life Events With Open Allocation, Too

**Kami Richey:** I'm here in San Francisco, but my team is actually distributed too. We have two different apps for planning meetups: one is an internal app that allows GitHubbers to request their drink-ups saying, "Hey, I will be at location X on this day and I want to do a drink-up." My team and I can jump in there and help set the logistics of the actual [event].

---

By allowing self-organization, Kami Richey and her team of three people have supervised the execution of hundreds and hundreds of GitHub events all over the country--far more than they would have been able to if they planned everything themselves. Every part of these gatherings is produced internally; GitHub employees even perform the musical acts (<https://soundcloud.com/brntbeer/sets/code-base-2>).

- Drink-up attendees in 2012:  
~5,500

- Drink-up attendees in 2013 to date: ~3,000
- Hackathon attendees in 2013 to date: ~14,000
- Conference attendees where GitHub is speaking or sponsoring in 2013 to date: ~100,000 (in person - even more watch talks online)

---

As we've scaled we've found a lot of people do the drink-up on their own. Just giving them the tools to do it on their own, we still have a way to track where we're going and where we've been.

After you have a drink-up or another event, we encourage people to talk about it internally. We have an app that we call Team--one of the features of it is a "status update" to share with the whole company, and we encourage people to do that internally. [Hubbers] can say: I met these people, or I [heard] about some new project, some new app, or I met somebody that maybe will be a potential GitHubber in the future. We encourage this inquisitiveness internally.

Because we are so distributed, we need a way to stay in communication about what people are doing. [The Team app] has a couple of different features and one of them is a status update feed. There's also a location [map] so you can see where every GitHubber is in the world, physically. The World Domination app, that's what they call the map that shows where every Hubber is, and all of their contact info: their name, email, cell phone number, and other data.

## Use The Right Communication Channel At The Right Time

**Tom:** Something that we're actively thinking about right now is how to have discussions most efficiently, and cover the most people. Something that we've discovered is that



strategic discussions should happen synchronously, meaning in real time, either face to face or on the phone or on Google Hangout, or something like that. Because it's how you can cover a broad space of possibilities very quickly, as opposed to an email or on GitHub; those discussions are more difficult, because you have to take the time to write out a complete thought, and it just takes too long. The bandwidth of those conversations is not high enough unless they're real-time.

So that's something that we've discovered is that being conscious of what type of conversation you're having, whether it's strategic or tactical and using the right kind of venue for that, whether it be online or it be in person, and it can still be distributed; you can still have a distributed team talk strategically by using video chat, for instance. Geography still isn't a barrier to that--it's just the type of conversation that you're having that matters.



## Self-Organization Can Run Your Publicity Engine, Too

**Kami:** We use our blog and our Twitter feed for [publicizing] most of those events. If we're working with a conference or a user group, we have them broadcast it out to their community

too.

We crowdsource that so that almost all gets covered. Everyone in the company has access to the company blog and the Twitter feed. It's just not me, it's actually everyone. But you can't just post on the blog post and ship it. Just like sending anything else [at GitHub] you send a pull request. If I do a blog post, I would do the draft and I would call on someone else and say, "Hey, I have a blog post, I need you to check it out." That way you get a second set of eyes, sometimes a third set of eyes, just to go over what you're posting about, just to make sure that there are no grammar mistakes, or that you're not announcing something before it's ready to go out.

**Liz Clinkenbeard:** People say, "I want to tell you guys about this tweet, is it cool?" Then usually about five or six people will respond.

## Divide Strategic And Tactical Responsibilities Like This

**Tom:** If you do things in that way, you have a style of management that is much more emergent than a traditional company that has strict hierarchy and departments and control structures. Some nodes in this graph will be strategic, some of them will be tactical, and some of them will be both.

So some people in the company think strategically at a broad level about the company, people like me. Other people are leading teams, and really helping those teams succeed, and thinking about a product and how it fits within the broader strategy of the company. And then you have tactical people who are on the ground, implementing the work toward those strategic visions. And all of those people are equally important to make things happen. You can't have just one or the other, you have to have the right balance.

Honestly, you can be both strategic and tactical; there's nothing that says that you can't, and in fact it's really great if you are both, because it means you stay close to the things that you're working on. You don't lose touch with the technology or with the customers or with whatever it is that you're working on and leading up.

And so in that sense, you have people talking to each other and talking to people on other teams directly, instead of going up through a hierarchical chain and then down another chain in order to get information. We just go directly, and that's totally appropriate.

## How To Decide What's Important

**Tom:** The system also allows for a lot of flexibility, because the teams are deciding what's most important amongst themselves, and you also have a chance for people to work on things that are interesting to them, but not as "important," depending on how that team thinks of innovation. And so the definition of "important" can be broad. Important can also be, "We need to experiment with some things." And so while we say people work on whatever they want here, it's within that kind of framework, to say, determine what's important and then work on the most interesting one of those to you.

## Allow People Some Privacy

**Tom:** The point is to not avoid secrets. Sometimes a group will be working and there's no reason for people outside of it to really know what's going on there yet. If they're spec'ing out some sort of experimental effort, or they're in talks with some group for some reason--then other people in the company might not hear about it right away.

But as soon as it gets to a level where input from the company could make that decision better, then it's opened up and communicated to the

company, so people can be involved, especially if it affects them and if they think it's important to them. A lot of companies don't embrace openness.

And part of open source and open allocation is the concept of being broadly open to the whole company about where you're going, what you're doing, what the groups are doing, what research and development is being done--everything.

## **Even The Building Should Be Openly Hackable**

**Scott:** I went through a lot of trouble trying to make sure that all of our building systems have API, or as many as we possibly can. We're working on getting sub-metering--like per-panel metering so we can see the energy usage as API, so we can query it.

All of our lights are LED and they're powered over Econet and they have a rough API on them so we can change them, we can dim them, we can see the current wattage draw on all of them. We can program them to do different things; we can see the occupancy under every single light. Every single light has a sensor on it that does ambient light and temperature and occupancy status. All of that's available to all of our developers; everybody in the company.

**"The number of times that I've had to describe API to [building contractors] I almost ripped my hair out."**

We are encouraging people that hack the building to save energy usage by modifying the programs that control the lights. We can do stuff like daylight harvesting where, since we can see what the ambient light is under every single light, we can use that to modify the dim on the light to try to save light, if , let's say it becomes brighter near a window or something like that.

That's one of the other things that we're in. We have API controller of the HVAC system, we'd like to do a net like intelligent building heating system to try to save energy and only heat rooms when we know that there is going to be a meeting there and then turn off the heat again after that, and predict when certain parts of the building are going to be heated and try to save as much energy as possible that way as well.

We have basically full control over all of the building systems in the new building and I'm really fascinated to see what we end up doing with that. The number of times that I've had to describe API to [building contractors] I almost ripped my hair out.



## Grow Like A Tree, Not A Weed

**Tom:** But having that kind of structure replicated throughout the whole organization, it kind of doesn't matter how big it gets, so long as this process happens throughout each of these things. As teams get bigger, then they start to have sub-teams, and then there's team leaders of teams within larger teams. And the leaders of those bigger teams start to become more and more strategic, and if they're developers, maybe they start coding less, and thinking more strategically about the product, and how it fits in,

and they're talking to the people who are thinking at a broader strategic level above them.

And so that can scale out really well, but it can still preserve all of the structure that allows for innovation. That is, people identifying what's important to achieve the strategic goals that their team leader is helping them understand, and then working on things that are most interesting to them within that subset of things that have been identified as important. It remains to be seen, but so far that's kind of how it boils down.

## How You Know It's Time To Experiment With Open Allocation

**Tom:** In the beginning [of GitHub] everyone could weigh in on decisions sort of equally and you had to have really broad consensus to do so, and that works for a while, but at some point it breaks down, and you start to be unable to make decisions very well, and you get into, what people call "design by committee."

That's not good; it can cause a lot of stagnation in decision making. And so one thing we did that was really effective was to step back and say, "Well, everyone shouldn't be involved in every decision. It's not efficient." And it ends up producing really poor-quality experiences and products actually. And so what makes sense? Who should be making the decisions?

Well, the people that work on something every day should be making the decisions. But, you want to preserve a culture where anyone can contribute to anything they think is interesting. And so the way we work now is, the people who work on something every day make the decisions on those things, but everyone in the company is encouraged to contribute and act as an advisor to decisions that are being made anywhere else in the company.

But they know at the end of the day that they're not gonna be responsible for that decision. But the information they provide can lead to that team making a better decision. And so you start to scale that way, by preserving the things that make you what you are--everybody can contribute, you embrace cooperation, but then you scale by dividing out who is most responsible for what areas of the company. And so that's how we're kind of scaling, by thinking very carefully about, when do you need to narrow the scope of who's involved in something.

## Keep Experimenting With Structure

**Tom:** Open allocation is a system we're still experimenting with and trying to see if it fits well. Something you have to understand about GitHub is that we're not dogmatic about solutions to problems. We like to be flexible in thinking through problems in our context and then choosing and experimenting with ways to solve problems. And that is what drives us--not any specific kind of organizational structure or something like open allocation.

A year from now we might be using something that's not open allocation, but is the evolution of open allocation, as we've determined that something else works better. And that's really fine. We just want to run this company in a way that maintains our core values, things that we find most important, and that allows people to work and have a great happiness in working.

## How Can You Convert A Traditional Company To Open Allocation?

**Tom:** How I think it would happen within an existing hierarchical company is that the manager of some team would say, "All right people, the way that we are gonna work within our small section of this company is using these concepts of open allocation."



**“You know, these huge companies complain all the time about not being able to innovate, and yet they put these very strict, controlling hierarchies in place, and they’re mystified by why they can’t innovate--well, if you tell people what to do every single day, and they get in trouble if they do something that’s not on their list, then how could you possibly innovate?”**

So they carve out a chunk of the company and run it differently. Maybe it’s a decent-sized group, maybe it’s like 30 people or something. And the manager of that whole group talks to the managers underneath them and says, “All right, so you guys are now not managers in the traditional sense, but you are the leaders of these teams. And if you want these teams to be successful, then you need them to align with you by their own option, and if people stop aligning with you, then maybe you’re not working here that long.”

It becomes turned upside down--your main job as the manager of that section is to communicate strategy, and then let other people determine what’s important in order to accomplish that strategy. And just the possibilities that it opens are amazing.

You know, these huge companies complain all the time about not being able to innovate, and yet they put these very strict, controlling hierarchies in place, and they’re mystified by why they can’t innovate--well, if you tell people what to do every single day, and they get in trouble if they do something that’s not on their list, then how could you possibly innovate?

## Can This Work In Non-Tech Companies

**Scott:** It depends on I guess, what that question means like it could work in any type of company, I think. A lot of companies are experimenting with that type of thing, not just pure software companies, or not just companies that are creating tools for software developers.

We depart from [the open-source model] in ways that we think are important, around things like design and documentation, which open source has been relatively bad at. We're trying to make up for that and have a balance. At the end of the day, we want all of the important decisions on implementation and on tactics to be the people closest to the problem, the people on the ground. Open [allocation] has not been great at attracting other types of professionals: technical documentation people, people that like to write, people that like to teach.



## Don't Forget The Goal

**Peter:** In my mind there's kind of a common line that keeps popping up for me which is: We think there's a better way, we think that maybe there's a better way to do management at a company. We think that there might be a better way to do

project allocation or maybe there's a better way to communicate. We have a number of different tools that we use for communication and for management of our time. I think it is just wanting to share the GitHub philosophy and kind of the GitHub approach with the rest of the world, at the highest level.

## Conclusion: Know Your Message Cold, Then Let People Roam

Communication is about appealing to people's interests. You can't force people to do their best work under threat of termination--at least not often--because it's personally degrading and emotionally taxing. Mandates are toxic. As we wrote in Aaron Schildkrout's must-read [seven-part product development series](http://www.fastcolabs.com/3015920/open-company/unblocked-a-guide-to-making-things-people-love-part-1) (<http://www.fastcolabs.com/3015920/open-company/unblocked-a-guide-to-making-things-people-love-part-1>), you need to *unblock* your employees to harness their real creativity.

Just because you're at the top of a company doesn't mean people will listen. Getting people to think of you as a "thought leader" inside the company means creating some proof in the form of content. (I'm defining "content" loosely here, as in company blog posts, company-wide email memos, training videos, speeches, documentation, clever Tshirt designs, and other manifestations of your company's mission.) For obvious reasons, it behooves startup founders to do this inside their company first--both to figure out what to say, and to enfranchise their employees first and foremost.

That great communication is vital to an efficient company is almost tautological, but the simplicity of the concept belies the difficulty of execution. There are lots of brilliant engineers and designers out there--why don't they all coalesce into incredible startups? *Because just getting along can be hard.*

For a company to grow and scale, you have to convince an increasingly big group to hang around you all day--and pin their careers, perhaps years of their lives, on this venture. And that means being extremely likeable. Being likeable means communicating thoroughly, respectfully, frequently, and effectively, and not being boring.

If you only absorb one thing from this 8,000-word piece, it's that the people at the top of the company need to clearly present the message behind its product--the mission, the problem you are all solving for the people who are going to use your product. Your message is your hypothesis about how the world should be different, and how you plan to make it that way.

Once employees have bought into the message, let them go about spreading it proudly in their own way, through their own work, and you might end up with a product that is as beloved by its users as GitHub.

[Image: Flickr user [Asparukh Akanayev \(https://secure.flickr.com/photos/aakanayev/10279970094/in/photostream/\)](https://secure.flickr.com/photos/aakanayev/10279970094/in/photostream/)]



**CHRIS DANNEN**

I'm Senior Editor at FastCompany Digital in New York and editor of FastCo.Labs. I've written for FastCompany, CBS MoneyWatch, MIT Technology Review, The Atlantic and others. I co-created the enterprise collaboration Writebot. Career interludes include stints as a management consultant, an outdoor sports journalist, and a brand rep for Red Bull Energy Drink.

(<http://www.fastcolabs.com/user/chris-dannen>)

[Continue \(http://www.fastcolabs.com/user/chris-dannen\)](http://www.fastcolabs.com/user/chris-dannen)

October 18, 2013 | 11:30 AM

**YOU MIGHT ALSO LIKE**  
(<http://www.fastcolabs.com/user/chris-dannen>)



Watch  
Photoshop Turn  
A Woman Into A  
Completely  
Different  
Woman  
Co.Create

(<http://www.fastcocreate.com/3020948/watch-photoshop-turn-a-woman-into-a-completely-different-woman>) (<http://blog.gigya.com/white-paper-marketers-wake-up-4-social-consumer-signals-you-cant-afford-to-ignore/>)



This Google

(<http://www.fastcolabs.com/3014713/this-google-glass-porno-flick-foreshadows-your-future-sex-life>)

[?]

ADD NEW COMMENT

SIGN IN

Type your comment here.

23 COMMENTS

STEVE BURKETT ([HTTPS://TWITTER.COM/INTENT/USER?SCREEN\\_NAME=STVBRKETT](https://twitter.com/intent/user?screen_name=stevburkett)) > CHRIS

DANNEN 3 MONTHS AGO

"So while this story is about GitHub...that is trying to get its best work out of a bunch of unruly twenty-and thirty-somethings who want to move fast and try out new, promising ideas." just 20-30-ish? older need not apply? :)



[Share](#) [Link](#) [Reply](#)

Comment removed.

JOSEPH ROBISON > CHRIS DANNEN 4 MONTHS AGO

Awesome in-depth masterpiece, saved.



[Delete](#) [Share](#) [Link](#) [Reply](#)

HAMEED > CHRIS DANNEN [5 MONTHS AGO](#)

Superb article....read it twice. Business Yoga....



[Delete](#) [Share](#) [Link](#) [Reply](#)

NICHOLAS M. CUMMINGS > CHRIS DANNEN [5 MONTHS AGO](#)

Fascinating article, but I'm curious what kinds of formal infrastructure and tracking they have in place for keeping track of people



[Delete](#) [Share](#) [Link](#) [Reply](#)

JOSH ANDERSON > NICHOLAS M. CUMMINGS [5 MONTHS AGO](#)

Isn't that the point of this approach? Trust your team and stop trying to "keep track" of them...



[Delete](#) [Share](#) [Link](#) [Reply](#)

NICHOLAS M. CUMMINGS > JOSH ANDERSON [5 MONTHS AGO](#)

Sure but even a company as great as Google struggles to keep track of employees that Xooglers jokingly refer to as VIPs (Vesting in Peace). It's a real concern once you hit scale. Furthermore, how do you review performance and provide upward

mobility and professional development if nobody really knows what anyone has done?

To be clear, I love it. I was ridiculed during my brief stint part-time at Boston University for something resembling socialism and feudalism, a flat structure no greater than three or four layers deep where you had entry level people, "knights" who were proven to stay, "lord" who could run projects, and the executive team, where everyone at a certain rank could vote on major decisions and owned a stake... You could flexibly move people around as needed. But where I got torn to shreds by the organizational behavior future-corporate-politickers and thinkers-inside-the-box was tracking, and I think to some extent their points are valid.

   [Delete](#) [Share](#) [Link](#) [Reply](#)

JOSH ANDERSON > NICHOLAS M.

**CUMMINGS** [5 MONTHS AGO](#)

I think the need to track is part of the overall problem. The need to track implies distrust. Distrust ruins team.

That admittedly sounds all rosey and wonderland-ish, but I believe it can be done. Sure, it is hard to build a company like that, and it will get even harder as you grow. That's not going to stop folks from making it happen. Rigor around hiring to your values and managing/firing to your values will be the keys to success in this approach.



   [Delete](#) [Share](#) [Link](#) [Reply](#)

PALLADIO > **CHRIS DANNEN** [5 MONTHS AGO](#)

Great research! I've been following Valve for some time, and "old world" companies such as Gore or Morning Star. It would be great to compare with those examples that have failed (e.g. Ascaron), in order to find out what is crucial to success and how to be strong in difficult situations, too.

   [Delete](#) [Share](#) [Link](#) [Reply](#)

GENIE LAUREN > **CHRIS DANNEN** [5 MONTHS AGO](#)

On one hand, I'd love to work in an environment like this. On the other, I'm an introvert. So, I'm wondering how they balance constant collaboration with allowing people who work better alone, to work alone.

   [Delete](#) [Share](#) [Link](#) [Reply](#)

SEE NEW COMMENTS

LOAD MORE COMMENTS