You Weren't Meant to Have a Boss

March 2008, rev. June 2008Technology tends to separate normal from natural. Our bodies  
weren't designed to eat the foods that people in rich countries eat, or  
to get so little exercise.   
There may be a similar problem with the way we work:   
a normal job may be as bad for us intellectually as white flour  
or sugar is for us physically.I began to suspect this after spending several years working   
with startup founders. I've now worked with over 200 of them, and I've  
noticed a definite difference between programmers working on their  
own startups and those working for large organizations.  
I wouldn't say founders seem happier, necessarily;  
starting a startup can be very stressful. Maybe the best way to put  
it is to say that they're happier in the sense that your body is  
happier during a long run than sitting on a sofa eating  
doughnuts.Though they're statistically abnormal, startup founders seem to be  
working in a way that's more natural for humans.I was in Africa last year and saw a lot of animals in the wild that  
I'd only seen in zoos before. It was remarkable how different they  
seemed. Particularly lions. Lions in the wild seem about ten times  
more alive. They're like different animals. I suspect that working  
for oneself feels better to humans in much the same way that living  
in the wild must feel better to a wide-ranging predator like a lion.  
Life in a zoo is easier, but it isn't the life they were designed  
for.  
TreesWhat's so unnatural about working for a big company? The root of  
the problem is that humans weren't meant to work in such large  
groups.Another thing you notice when you see animals in the wild is that  
each species thrives in groups of a certain size. A herd of impalas  
might have 100 adults; baboons maybe 20; lions rarely 10. Humans  
also seem designed to work in groups, and what I've read about  
hunter-gatherers accords with research on organizations and my own  
experience to suggest roughly what the ideal size is: groups of 8  
work well; by 20 they're getting hard to manage; and a group of 50  
is really unwieldy.  
[1]  
Whatever the upper limit is, we are clearly not meant to work in  
groups of several hundred. And yet—for reasons having more  
to do with technology than human nature—a great many people  
work for companies with hundreds or thousands of employees.Companies know groups that large wouldn't work, so they divide  
themselves into units small enough to work together. But to  
coordinate these they have to introduce something new: bosses.These smaller groups are always arranged in a tree structure. Your  
boss is the point where your group attaches to the tree. But when  
you use this trick for dividing a large group into smaller ones,  
something strange happens that I've never heard anyone mention  
explicitly. In the group one level up from yours, your boss  
represents your entire group. A group of 10 managers is not merely  
a group of 10 people working together in the usual way. It's really  
a group of groups. Which means for a group of 10 managers to work  
together as if they were simply a group of 10 individuals, the group  
working for each manager would have to work as if they were a single  
person—the workers and manager would each share only one  
person's worth of freedom between them.In practice a group of people are never able to act as if they were  
one person. But in a large organization divided into groups in  
this way, the pressure is always in that direction. Each group  
tries its best to work as if it were the small group of individuals  
that humans were designed to work in. That was the point of creating  
it. And when you propagate that constraint, the result is that  
each person gets freedom of action in inverse proportion to the  
size of the entire tree.  
[2]Anyone who's worked for a large organization has felt this. You  
can feel the difference between working for a company with 100  
employees and one with 10,000, even if your group has only 10 people.  
Corn SyrupA group of 10 people within a large organization is a kind of fake  
tribe. The number of people you interact with is about right. But  
something is missing: individual initiative. Tribes of hunter-gatherers  
have much more freedom. The leaders have a little more power than other  
members of the tribe, but they don't generally tell them what to  
do and when the way a boss can.It's not your boss's fault. The real problem is that in the group  
above you in the hierarchy, your entire group is one virtual person.  
Your boss is just the way that constraint is imparted to you.So working in a group of 10 people within a large organization feels  
both right and wrong at the same time. On the surface it feels  
like the kind of group you're meant to work in, but something major  
is missing. A job at a big company is like high fructose corn  
syrup: it has some of the qualities of things you're meant to like,  
but is disastrously lacking in others.Indeed, food is an excellent metaphor to explain what's wrong with  
the usual sort of job.For example, working for a big company is the default thing to do,  
at least for programmers. How bad could it be? Well, food shows  
that pretty clearly. If you were dropped at a random point in  
America today, nearly all the food around you would be bad for you.  
Humans were not designed to eat white flour, refined sugar, high  
fructose corn syrup, and hydrogenated vegetable oil. And yet if  
you analyzed the contents of the average grocery store you'd probably  
find these four ingredients accounted for most of the calories.  
"Normal" food is terribly bad for you. The only people who eat  
what humans were actually designed to eat are a few Birkenstock-wearing  
weirdos in Berkeley.If "normal" food is so bad for us, why is it so common? There are  
two main reasons. One is that it has more immediate appeal. You  
may feel lousy an hour after eating that pizza, but eating the first  
couple bites feels great. The other is economies of scale.  
Producing junk food scales; producing fresh vegetables doesn't.  
Which means (a) junk food can be very cheap, and (b) it's worth  
spending a lot to market it.If people have to choose between something that's cheap, heavily  
marketed, and appealing in the short term, and something that's  
expensive, obscure, and appealing in the long term, which do you  
think most will choose?It's the same with work. The average MIT graduate wants to work  
at Google or Microsoft, because it's a recognized brand, it's safe,  
and they'll get paid a good salary right away. It's the job  
equivalent of the pizza they had for lunch. The drawbacks will  
only become apparent later, and then only in a vague sense of  
malaise.And founders and early employees of startups, meanwhile, are like  
the Birkenstock-wearing weirdos of Berkeley: though a tiny minority  
of the population, they're the ones living as humans are meant to.  
In an artificial world, only extremists live naturally.  
ProgrammersThe restrictiveness of big company jobs is particularly hard on  
programmers, because the essence of programming is to build new  
things. Sales people make much the same pitches every day; support  
people answer much the same questions; but once you've written a  
piece of code you don't need to write it again. So a programmer  
working as programmers are meant to is always making new things.  
And when you're part of an organization whose structure gives each  
person freedom in inverse proportion to the size of the tree, you're  
going to face resistance when you do something new.This seems an inevitable consequence of bigness. It's true even  
in the smartest companies. I was talking recently to a founder who  
considered starting a startup right out of college, but went to  
work for Google instead because he thought he'd learn more there.  
He didn't learn as much as he expected. Programmers learn by doing,  
and most of the things he wanted to do, he couldn't—sometimes  
because the company wouldn't let him, but often because the company's  
code wouldn't let him. Between the drag of legacy code, the overhead  
of doing development in such a large organization, and the restrictions  
imposed by interfaces owned by other groups, he could only try a  
fraction of the things he would have liked to. He said he has  
learned much more in his own startup, despite the fact that he has  
to do all the company's errands as well as programming, because at  
least when he's programming he can do whatever he wants.An obstacle downstream propagates upstream. If you're not allowed  
to implement new ideas, you stop having them. And vice versa: when  
you can do whatever you want, you have more ideas about what to do.  
So working for yourself makes your brain more powerful in the same  
way a low-restriction exhaust system makes an engine more powerful.Working for yourself doesn't have to mean starting a startup, of  
course. But a programmer deciding between a regular job at a big  
company and their own startup is probably going to learn more doing  
the startup.You can adjust the amount of freedom you get by scaling the size  
of company you work for. If you start the company, you'll have the  
most freedom. If you become one of the first 10 employees you'll  
have almost as much freedom as the founders. Even a company with  
100 people will feel different from one with 1000.Working for a small company doesn't ensure freedom. The tree  
structure of large organizations sets an upper bound on freedom,  
not a lower bound. The head of a small company may still choose  
to be a tyrant. The point is that a large organization is compelled  
by its structure to be one.  
ConsequencesThat has real consequences for both organizations and individuals.  
One is that companies will inevitably slow down as they grow larger,  
no matter how hard they try to keep their startup mojo. It's a  
consequence of the tree structure that every large organization is  
forced to adopt.Or rather, a large organization could only avoid slowing down if  
they avoided tree structure. And since human nature limits the  
size of group that can work together, the only way I can imagine  
for larger groups to avoid tree structure would be to have no  
structure: to have each group actually be independent, and to work  
together the way components of a market economy do.That might be worth exploring. I suspect there are already some  
highly partitionable businesses that lean this way. But I don't  
know any technology companies that have done it.There is one thing companies can do short of structuring themselves  
as sponges: they can stay small. If I'm right, then it really  
pays to keep a company as small as it can be at every stage.  
Particularly a technology company. Which means it's doubly important  
to hire the best people. Mediocre hires hurt you twice: they get  
less done, but they also make you big, because you need more of  
them to solve a given problem.For individuals the upshot is the same: aim small. It will always  
suck to work for large organizations, and the larger the organization,  
the more it will suck.In an essay I wrote a couple years ago   
I advised graduating seniors  
to work for a couple years for another company before starting their  
own. I'd modify that now. Work for another company if you want  
to, but only for a small one, and if you want to start your own  
startup, go ahead.The reason I suggested college graduates not start startups immediately  
was that I felt most would fail. And they will. But ambitious  
programmers are better off doing their own thing and failing than  
going to work at a big company. Certainly they'll learn more. They  
might even be better off financially. A lot of people in their  
early twenties get into debt, because their expenses grow even  
faster than the salary that seemed so high when they left school.  
At least if you start a startup and fail your net worth will be  
zero rather than negative.   
[3]We've now funded so many different types of founders that we have  
enough data to see patterns, and there seems to be no benefit from  
working for a big company. The people who've worked for a few years  
do seem better than the ones straight out of college, but only  
because they're that much older.The people who come to us from big companies often seem kind of  
conservative. It's hard to say how much is because big companies  
made them that way, and how much is the natural conservatism that  
made them work for the big companies in the first place. But  
certainly a large part of it is learned. I know because I've seen  
it burn off.Having seen that happen so many times is one of the things that  
convinces me that working for oneself, or at least for a small  
group, is the natural way for programmers to live. Founders arriving  
at Y Combinator often have the downtrodden air of refugees. Three  
months later they're transformed: they have so much more   
confidence  
that they seem as if they've grown several inches taller.   
[4]  
Strange as this sounds, they seem both more worried and happier at the same  
time. Which is exactly how I'd describe the way lions seem in the  
wild.Watching employees get transformed into founders makes it clear  
that the difference between the two is due mostly to environment—and  
in particular that the environment in big companies is toxic to  
programmers. In the first couple weeks of working on their own  
startup they seem to come to life, because finally they're working  
the way people are meant to.Notes[1]  
When I talk about humans being meant or designed to live a  
certain way, I mean by evolution.[2]  
It's not only the leaves who suffer. The constraint propagates  
up as well as down. So managers are constrained too; instead of  
just doing things, they have to act through subordinates.[3]  
Do not finance your startup with credit cards. Financing a  
startup with debt is usually a stupid move, and credit card debt  
stupidest of all. Credit card debt is a bad idea, period. It is  
a trap set by evil companies for the desperate and the foolish.[4]  
The founders we fund used to be younger (initially we encouraged  
undergrads to apply), and the first couple times I saw this I used  
to wonder if they were actually getting physically taller.Thanks to Trevor Blackwell, Ross Boucher, Aaron Iba, Abby  
Kirigin, Ivan Kirigin, Jessica Livingston, and Robert Morris for  
reading drafts of this.