

# Table of Contents

- [How to Convince Investors](#)
- [Investor Herd Dynamics](#)
- [How to Raise Money](#)
- [Before the Startup](#)
- [Mean People Fail](#)
- [The Fatal Pinch](#)
- [How You Know](#)
- [How to Be an Expert in a Changing World](#)
- [Let the Other 95% of Great Programmers In](#)
- [Don't Talk to Corp Dev](#)
- [What Doesn't Seem Like Work?](#)
- [The Ronco Principle](#)
- [What Microsoft Is this the Altair Basic of?](#)
- [Change Your Name](#)
- [Why It's Safe for Founders to Be Nice](#)
- [Default Alive or Default Dead?](#)
- [Write Like You Talk](#)
- [A Way to Detect Bias](#)
- [Jessica Livingston](#)
- [The Refragmentation](#)
- [Economic Inequality](#)
- [Life is Short](#)
- [How to Make Pittsburgh a Startup Hub](#)
- [The Risk of Discovery](#)
- [Charisma / Power](#)
- [General and Surprising](#)
- [The Bus Ticket Theory of Genius](#)
- [Novelty and Heresy](#)
- [The Lesson to Unlearn](#)
- [Having Kids](#)
- [Fashionable Problems](#)
- [The Two Kinds of Moderate](#)
- [Haters](#)
- [Being a Noob](#)
- [How to Write Usefully](#)
- [Coronavirus and Credibility](#)
- [Orthodox Privilege](#)
- [The Four Quadrants of Conformism](#)
- [Modeling a Wealth Tax](#)
- [Early Work](#)
- [How to Think for Yourself](#)
- [The Airbnbs](#)
- [Billionaires Build](#)
- [Earnestness](#)
- [What I Worked On](#)
- [Donate Unrestricted](#)
- [Write Simply](#)

- [How People Get Rich Now](#)
- [The Real Reason to End the Death Penalty](#)
- [An NFT That Saves Lives](#)
- [Crazy New Ideas](#)
- [Fierce Nerds](#)
- [A Project of One's Own](#)
- [How to Work Hard](#)
- [Weird Languages](#)
- [Beyond Smart](#)
- [Is There Such a Thing as Good Taste?](#)
- [Putting Ideas into Words](#)
- [Heresy](#)
- [What I've Learned from Users](#)
- [Alien Truth](#)
- [What You \(Want to\)\\* Want](#)
- [The Need to Read](#)
- [How to Get New Ideas](#)
- [How to Do Great Work](#)
- [Superlinear Returns](#)
- [The Best Essay](#)
- [How to Start Google](#)
- [The Reddits](#)
- [The Right Kind of Stubborn](#)
- [Founder Mode](#)
- [When To Do What You Love](#)
- [Writes and Write-Notes](#)
- [The Origins of Wokeness](#)
- [What to Do](#)

# How to Convince Investors

August 2013

When people hurt themselves lifting heavy things, it's usually because they try to lift with their back. The right way to lift heavy things is to let your legs do the work. Inexperienced founders make the same mistake when trying to convince investors. They try to convince with their pitch. Most would be better off if they let their startup do the work — if they started by understanding why their startup is worth investing in, then simply explained this well to investors.

Investors are looking for startups that will be very successful. But that test is not as simple as it sounds. In startups, as in a lot of other domains, the distribution of outcomes follows a power law, but in startups the curve is startlingly steep. The big successes are so big they [dwarf](#) the rest. And since there are only a handful each year (the conventional wisdom is 15), investors treat "big success" as if it were binary. Most are interested in you if you seem like you have a chance, however small, of being one of the 15 big successes, and otherwise not. [\[1\]](#)

(There are a handful of angels who'd be interested in a company with a high probability of being moderately successful. But angel investors like big successes too.)

How do you seem like you'll be one of the big successes? You need three things: formidable founders, a promising market, and (usually) some evidence of success so far.

## Formidable

The most important ingredient is formidable founders. Most investors decide in the first few minutes whether you seem like a winner or a loser, and once their opinion is set it's hard to change. [\[2\]](#) Every startup has reasons both to invest and not to invest. If investors think you're a winner they focus on the former, and if not they focus on the latter. For example, it might be a rich market, but with a slow sales cycle. If investors are impressed with you as founders, they say they want to invest because it's a rich market, and if not, they say they can't invest because of the slow sales cycle.

They're not necessarily trying to mislead you. Most investors are genuinely unclear in their own minds why they like or dislike startups. If you seem like a winner, they'll like your idea more. But don't be too smug about this weakness of theirs, because you have it too; almost everyone does.

There is a role for ideas of course. They're fuel for the fire that starts with liking

the founders. Once investors like you, you'll see them reaching for ideas: they'll be saying "yes, and you could also do x." (Whereas when they don't like you, they'll be saying "but what about y?")

But the foundation of convincing investors is to seem formidable, and since this isn't a word most people use in conversation much, I should explain what it means. A formidable person is one who seems like they'll get what they want, regardless of whatever obstacles are in the way. Formidable is close to confident, except that someone could be confident and mistaken. Formidable is roughly justifiably confident.

There are a handful of people who are really good at seeming formidable — some because they actually are very formidable and just let it show, and others because they are more or less con artists. [3] But most founders, including many who will go on to start very successful companies, are not that good at seeming formidable the first time they try fundraising. What should they do? [4]

What they should not do is try to imitate the swagger of more experienced founders. Investors are not always that good at judging technology, but they're good at judging confidence. If you try to act like something you're not, you'll just end up in an uncanny valley. You'll depart from sincere, but never arrive at convincing.

## Truth

The way to seem most formidable as an inexperienced founder is to stick to the truth. How formidable you seem isn't a constant. It varies depending on what you're saying. Most people can seem confident when they're saying "one plus one is two," because they know it's true. The most diffident person would be puzzled and even slightly contemptuous if they told a VC "one plus one is two" and the VC reacted with skepticism. The magic ability of people who are good at seeming formidable is that they can do this with the sentence "we're going to make a billion dollars a year." But you can do the same, if not with that sentence with some fairly impressive ones, so long as you convince yourself first.

That's the secret. Convince yourself that your startup is worth investing in, and then when you explain this to investors they'll believe you. And by convince yourself, I don't mean play mind games with yourself to boost your confidence. I mean truly evaluate whether your startup is worth investing in. If it isn't, don't try to raise money. [5] But if it is, you'll be telling the truth when you tell investors it's worth investing in, and they'll sense that. You don't have to be a smooth presenter if you understand something well and tell the truth about it.

To evaluate whether your startup is worth investing in, you have to be a domain expert. If you're not a domain expert, you can be as convinced as you like about your idea, and it will seem to investors no more than an instance of the Dunning-Kruger effect. Which in fact it will usually be. And investors can tell fairly quickly whether you're a domain expert by how well you answer their questions. Know everything about your market. [6]

Why do founders persist in trying to convince investors of things they're not convinced of themselves? Partly because we've all been trained to.

When my friends Robert Morris and Trevor Blackwell were in grad school, one of

their fellow students was on the receiving end of a question from their faculty advisor that we still quote today. When the unfortunate fellow got to his last slide, the professor burst out:

Which one of these conclusions do you actually believe?

One of the artifacts of the way schools are organized is that we all get trained to talk even when we have nothing to say. If you have a ten page paper due, then ten pages you must write, even if you only have one page of ideas. Even if you have no ideas. You have to produce something. And all too many startups go into fundraising in the same spirit. When they think it's time to raise money, they try gamely to make the best case they can for their startup. Most never think of pausing beforehand to ask whether what they're saying is actually convincing, because they've all been trained to treat the need to present as a given — as an area of fixed size, over which however much truth they have must needs be spread, however thinly.

The time to raise money is not when you need it, or when you reach some artificial deadline like a Demo Day. It's when you can convince investors, and not before.

[7]

And unless you're a good con artist, you'll never convince investors if you're not convinced yourself. They're far better at detecting bullshit than you are at producing it, even if you're producing it unknowingly. If you try to convince investors before you've convinced yourself, you'll be wasting both your time.

But pausing first to convince yourself will do more than save you from wasting your time. It will force you to organize your thoughts. To convince yourself that your startup is worth investing in, you'll have to figure out why it's worth investing in. And if you can do that you'll end up with more than added confidence. You'll also have a provisional roadmap of how to succeed.

## **Market**

Notice I've been careful to talk about whether a startup is worth investing in, rather than whether it's going to succeed. No one knows whether a startup is going to succeed. And it's a good thing for investors that this is so, because if you could know in advance whether a startup would succeed, the stock price would already be the future price, and there would be no room for investors to make money. Startup investors know that every investment is a bet, and against pretty long odds.

So to prove you're worth investing in, you don't have to prove you're going to succeed, just that you're a sufficiently good bet. What makes a startup a sufficiently good bet? In addition to formidable founders, you need a plausible path to owning a big piece of a big market. Founders think of startups as ideas, but investors think of them as markets. If there are  $x$  number of customers who'd pay an average of  $\$y$  per year for what you're making, then the total addressable market, or TAM, of your company is  $\$xy$ . Investors don't expect you to collect all

that money, but it's an upper bound on how big you can get.

Your target market has to be big, and it also has to be capturable by you. But the market doesn't have to be big yet, nor do you necessarily have to be in it yet. Indeed, it's often better to start in a [small](#) market that will either turn into a big one or from which you can move into a big one. There just has to be some plausible sequence of hops that leads to dominating a big market a few years down the line.

The standard of plausibility varies dramatically depending on the age of the startup. A three month old company at Demo Day only needs to be a promising experiment that's worth funding to see how it turns out. Whereas a two year old company raising a series A round needs to be able to show the experiment worked.

[8]

But every company that gets really big is "lucky" in the sense that their growth is due mostly to some external wave they're riding, so to make a convincing case for becoming huge, you have to identify some specific trend you'll benefit from. Usually you can find this by asking "why now?" If this is such a great idea, why hasn't someone else already done it? Ideally the answer is that it only recently became a good idea, because something changed, and no one else has noticed yet.

Microsoft for example was not going to grow huge selling Basic interpreters. But by starting there they were perfectly poised to expand up the stack of microcomputer software as microcomputers grew powerful enough to support one. And microcomputers turned out to be a really huge wave, bigger than even the most optimistic observers would have predicted in 1975.

But while Microsoft did really well and there is thus a temptation to think they would have seemed a great bet a few months in, they probably didn't. Good, but not great. No company, however successful, ever looks more than a pretty good bet a few months in. Microcomputers turned out to be a big deal, and Microsoft both executed well and got lucky. But it was by no means obvious that this was how things would play out. Plenty of companies seem as good a bet a few months in. I don't know about startups in general, but at least half the startups we fund could make as good a case as Microsoft could have for being on a path to dominating a large market. And who can reasonably expect more of a startup than that?

## Rejection

If you can make as good a case as Microsoft could have, will you convince investors? Not always. A lot of VCs would have rejected Microsoft. [9] Certainly some rejected Google. And getting rejected will put you in a slightly awkward position, because as you'll see when you start fundraising, the most common question you'll get from investors will be "who else is investing?" What do you say if you've been fundraising for a while and no one has committed yet? [10]

The people who are really good at acting formidable often solve this problem by giving investors the impression that while no investors have committed yet, several are about to. This is arguably a permissible tactic. It's slightly dickish of investors to care more about who else is investing than any other aspect of your startup, and misleading them about how far along you are with other investors seems the complementary countermove. It's arguably an instance of scamming a scammer. But I don't recommend this approach to most founders, because most founders wouldn't be able to carry it off. This is the single most common lie told to investors, and you have to be really good at lying to tell members of some profession the most common lie they're told.

If you're not a master of negotiation (and perhaps even if you are) the best solution is to tackle the problem head-on, and to explain why investors have turned you down and why they're mistaken. If you know you're on the right track, then you also know why investors were wrong to reject you. Experienced investors are well aware that the best ideas are also the scariest. They all know about the VCs who rejected Google. If instead of seeming evasive and ashamed about having been turned down (and thereby implicitly agreeing with the verdict) you talk candidly about what scared investors about you, you'll seem more confident, which they like, and you'll probably also do a better job of presenting that aspect of your startup. At the very least, that worry will now be out in the open instead of being a gotcha left to be discovered by the investors you're currently talking to, who will be proud of and thus attached to their discovery. [\[11\]](#)

This strategy will work best with the best investors, who are both hard to bluff and who already believe most other investors are conventional-minded drones doomed always to miss the big outliers. Raising money is not like applying to college, where you can assume that if you can get into MIT, you can also get into Foobar State. Because the best investors are much smarter than the rest, and the best startup ideas look initially like [bad ideas](#), it's not uncommon for a startup to be rejected by all the VCs except the best ones. That's what happened to Dropbox. Y Combinator started in Boston, and for the first 3 years we ran alternating batches in Boston and Silicon Valley. Because Boston investors were so few and so timid, we used to ship Boston batches out for a second Demo Day in Silicon Valley. Dropbox was part of a Boston batch, which means all those Boston investors got the first look at Dropbox, and none of them closed the deal. Yet another backup and syncing thing, they all thought. A couple weeks later, Dropbox raised a series A round from Sequoia. [\[12\]](#)

## Different

Not understanding that investors view investments as bets combines with the ten page paper mentality to prevent founders from even considering the possibility of being certain of what they're saying. They think they're trying to convince investors of something very uncertain — that their startup will be huge — and convincing anyone of something like that must obviously entail some wild feat of salesmanship. But in fact when you raise money you're trying to convince investors



of something so much less speculative — whether the company has all the elements of a good bet — that you can approach the problem in a qualitatively different way. You can convince yourself, then convince them.

And when you convince them, use the same matter-of-fact language you used to convince yourself. You wouldn't use vague, grandiose marketing-speak among yourselves. Don't use it with investors either. It not only doesn't work on them, but seems a mark of incompetence. Just be concise. Many investors explicitly use that as a test, reasoning (correctly) that if you can't explain your plans concisely, you don't really understand them. But even investors who don't have a rule about this will be bored and frustrated by unclear explanations. [\[13\]](#)

So here's the recipe for impressing investors when you're not already good at seeming formidable:

1. Make something worth investing in.
2. Understand why it's worth investing in.
3. Explain that clearly to investors.

If you're saying something you know is true, you'll seem confident when you're saying it. Conversely, never let pitching draw you into bullshitting. As long as you stay on the territory of truth, you're strong. Make the truth good, then just tell it.

## Notes

[1] There's no reason to believe this number is a constant. In fact it's our explicit goal at Y Combinator to increase it, by encouraging people to start startups who otherwise wouldn't have.

[2] Or more precisely, investors decide whether you're a loser or possibly a winner. If you seem like a winner, they may then, depending on how much you're raising, have several more meetings with you to test whether that initial impression holds up.

But if you seem like a loser they're done, at least for the next year or so. And when they decide you're a loser they usually decide in way less than the 50 minutes they may have allotted for the first meeting. Which explains the astonished stories one always hears about VC inattentiveness. How could these people make investment decisions well when they're checking their messages

during startups' presentations? The solution to that mystery is that they've already made the decision.

[3] The two are not mutually exclusive. There are people who are both genuinely formidable, and also really good at acting that way.

[4] How can people who will go on to create giant companies not seem formidable early on? I think the main reason is that their experience so far has trained them to keep their wings folded, as it were. Family, school, and jobs encourage cooperation, not conquest. And it's just as well they do, because even being Genghis Khan is probably 99% cooperation. But the result is that most people emerge from the tube of their upbringing in their early twenties compressed into the shape of the tube. Some find they have wings and start to spread them. But this takes a few years. In the beginning even they don't know yet what they're capable of.

[5] In fact, change what you're doing. You're investing your own time in your startup. If you're not convinced that what you're working on is a sufficiently good bet, why are you even working on that?

[6] When investors ask you a question you don't know the answer to, the best response is neither to bluff nor give up, but instead to explain how you'd figure out the answer. If you can work out a preliminary answer on the spot, so much the better, but explain that's what you're doing.

[7] At YC we try to ensure startups are ready to raise money on Demo Day by encouraging them to ignore investors and instead focus on their companies till about a week before. That way most reach the stage where they're sufficiently convincing well before Demo Day. But not all do, so we also give any startup that wants to the option of deferring to a later Demo Day.

[8] Founders are often surprised by how much harder it is to raise the next round. There is a qualitative difference in investors' attitudes. It's like the difference between being judged as a kid and as an adult. The next time you raise money, it's not enough to be promising. You have to be delivering results.

So although it works well to show growth graphs at either stage, investors treat them differently. At three months, a growth graph is mostly evidence that the founders are effective. At two years, it has to be evidence of a promising market and a company tuned to exploit it.

[9] By this I mean that if the present day equivalent of the 3 month old Microsoft presented at a Demo Day, there would be investors who turned them down. Microsoft itself didn't raise outside money, and indeed the venture business barely existed when they got started in 1975.

[10] The best investors rarely care who else is investing, but mediocre investors almost all do. So you can use this question as a test of investor quality.

[11] To use this technique, you'll have to find out why investors who rejected you did so, or at least what they claim was the reason. That may require asking, because investors don't always volunteer a lot of detail. Make it clear when you ask that you're not trying to dispute their decision — just that if there is some weakness in your plans, you need to know about it. You won't always get a real reason out of them, but you should at least try.

[12] Dropbox wasn't rejected by all the East Coast VCs. There was one firm that wanted to invest but tried to lowball them.

[13] Alfred Lin points out that it's doubly important for the explanation of a startup to be clear and concise, because it has to convince at one remove: it has to work not just on the partner you talk to, but when that partner re-tells it to colleagues.

We consciously optimize for this at YC. When we work with founders create a Demo Day pitch, the last step is to imagine how an investor would sell it to colleagues.

**Thanks** to Marc Andreessen, Sam Altman, Patrick Collison, Ron Conway, Chris Dixon, Alfred Lin, Ben Horowitz, Steve Huffman, Jessica Livingston, Greg Mcadoo, Andrew Mason, Geoff Ralston, Yuri Sagalov, Emmett Shear, Rajat Suri, Garry Tan, Albert Wenger, Fred Wilson, and Qasar Younis for reading drafts of this.

# Investor Herd Dynamics

August 2013

The biggest component in most investors' opinion of you is the opinion of other investors. Which is of course a recipe for exponential growth. When one investor wants to invest in you, that makes other investors want to, which makes others want to, and so on.

Sometimes inexperienced founders mistakenly conclude that manipulating these forces is the essence of fundraising. They hear stories about stampedes to invest in successful startups, and think it's therefore the mark of a successful startup to have this happen. But actually the two are not that highly correlated. Lots of startups that cause stampedes end up flaming out (in extreme cases, partly as a result of the stampede), and lots of very successful startups were only moderately popular with investors the first time they raised money.

So the point of this essay is not to explain how to create a stampede, but merely to explain the forces that generate them. These forces are always at work to some degree in fundraising, and they can cause surprising situations. If you understand them, you can at least avoid being surprised.

One reason investors like you more when other investors like you is that you actually become a better investment. Raising money decreases the risk of failure. Indeed, although investors hate it, you are for this reason justified in raising your valuation for later investors. The investors who invested when you had no money were taking more risk, and are entitled to higher returns. Plus a company that has raised money is literally more valuable. After you raise the first million dollars, the company is at least a million dollars more valuable, because it's the same company as before, plus it has a million dollars in the bank. [\[1\]](#)

Beware, though, because later investors so hate to have the price raised on them that they resist even this self-evident reasoning. Only raise the price on an investor you're comfortable with losing, because some will angrily refuse. [\[2\]](#)

The second reason investors like you more when you've had some success at fundraising is that it makes you more confident, and an investors' opinion of [you](#) is the foundation of their opinion of your company. Founders are often surprised how quickly investors seem to know when they start to succeed at raising money. And

while there are in fact lots of ways for such information to spread among investors, the main vector is probably the founders themselves. Though they're often clueless about technology, most investors are pretty good at reading people. When fundraising is going well, investors are quick to sense it in your increased confidence. (This is one case where the average founder's inability to remain poker-faced works to your advantage.)

But frankly the most important reason investors like you more when you've started to raise money is that they're bad at judging startups. Judging startups is hard even for the best investors. The mediocre ones might as well be flipping coins. So when mediocre investors see that lots of other people want to invest in you, they assume there must be a reason. This leads to the phenomenon known in the Valley as the "hot deal," where you have more interest from investors than you can handle.

The best investors aren't influenced much by the opinion of other investors. It would only dilute their own judgment to average it together with other people's. But they are indirectly influenced in the practical sense that interest from other investors imposes a deadline. This is the fourth way in which offers beget offers. If you start to get far along the track toward an offer with one firm, it will sometimes provoke other firms, even good ones, to make up their minds, lest they lose the deal.

Unless you're a wizard at negotiation (and if you're not sure, you're not) be very careful about exaggerating this to push a good investor to decide. Founders try this sort of thing all the time, and investors are very sensitive to it. If anything oversensitive. But you're safe so long as you're telling the truth. If you're getting far along with investor B, but you'd rather raise money from investor A, you can tell investor A that this is happening. There's no manipulation in that. You're genuinely in a bind, because you really would rather raise money from A, but you can't safely reject an offer from B when it's still uncertain what A will decide.

Do not, however, tell A who B is. VCs will sometimes ask which other VCs you're talking to, but you should never tell them. Angels you can sometimes tell about other angels, because angels cooperate more with one another. But if VCs ask, just point out that they wouldn't want you telling other firms about your conversations, and you feel obliged to do the same for any firm you talk to. If they push you, point out that you're inexperienced at fundraising — which is always a safe card to play — and you feel you have to be extra cautious. [3]

While few startups will experience a stampede of interest, almost all will at least initially experience the other side of this phenomenon, where the herd remains clumped together at a distance. The fact that investors are so much influenced by other investors' opinions means you always start out in something of a hole. So don't be demoralized by how hard it is to get the first commitment, because much of the difficulty comes from this external force. The second will be easier.

## Notes

[1] An accountant might say that a company that has raised a million dollars is no richer if it's convertible debt, but in practice money raised as convertible debt is little different from money raised in an equity round.

[2] Founders are often surprised by this, but investors can get very emotional. Or rather indignant; that's the main emotion I've observed; but it is very common, to the point where it sometimes causes investors to act against their own interests. I know of one investor who invested in a startup at a \$15 million valuation cap. Earlier he'd had an opportunity to invest at a \$5 million cap, but he refused because a friend who invested earlier had been able to invest at a \$3 million cap.

[3] If an investor pushes you hard to tell them about your conversations with other investors, is this someone you want as an investor?

**Thanks** to Paul Buchheit, Jessica Livingston, Geoff Ralston, and Garry Tan for reading drafts of this.

[Russian Translation](#)

# How to Raise Money

September 2013

Most startups that raise money do it more than once. A typical trajectory might be (1) to get started with a few tens of thousands from something like Y Combinator or individual angels, then (2) raise a few hundred thousand to a few million to build the company, and then (3) once the company is clearly succeeding, raise one or more later rounds to accelerate growth.

Reality can be messier. Some companies raise money twice in phase 2. Others skip phase 1 and go straight to phase 2. And at Y Combinator we get an increasing number of companies that have already raised amounts in the hundreds of thousands. But the three phase path is at least the one about which individual startups' paths oscillate.

This essay focuses on phase 2 fundraising. That's the type the startups we fund are doing on Demo Day, and this essay is the advice we give them.

## Forces

Fundraising is hard in both senses: hard like lifting a heavy weight, and hard like solving a puzzle. It's hard like lifting a weight because it's intrinsically hard to convince people to part with large sums of money. That problem is irreducible; it should be hard. But much of the other kind of difficulty can be eliminated. Fundraising only seems a puzzle because it's an alien world to most founders, and I hope to fix that by supplying a map through it.

To founders, the behavior of investors is often opaque — partly because their motivations are obscure, but partly because they deliberately mislead you. And the misleading ways of investors combine horribly with the wishful thinking of inexperienced founders. At YC we're always warning founders about this danger, and investors are probably more circumspect with YC startups than with other companies they talk to, and even so we witness a constant series of explosions as these two volatile components combine. [\[1\]](#)

If you're an inexperienced founder, the only way to survive is by imposing external constraints on yourself. You can't trust your intuitions. I'm going to give you a set of rules here that will get you through this process if anything will. At certain moments you'll be tempted to ignore them. So rule number zero is: these rules exist for a reason. You wouldn't need a rule to keep you going in one direction if there weren't powerful forces pushing you in another.

The ultimate source of the forces acting on you are the forces acting on investors.

Investors are pinched between two kinds of fear: fear of investing in startups that fizzle, and fear of missing out on startups that take off. The cause of all this fear is the very thing that makes startups such attractive investments: the successful ones grow very fast. But that fast growth means investors can't wait around. If you wait till a startup is obviously a success, it's too late. To get the really high returns, you have to invest in startups when it's still unclear how they'll do. But that in turn makes investors nervous they're about to invest in a flop. As indeed they often are.

What investors would like to do, if they could, is wait. When a startup is only a few months old, every week that passes gives you significantly more information about them. But if you wait too long, other investors might take the deal away from you. And of course the other investors are all subject to the same forces. So what tends to happen is that they all wait as long as they can, then when some act the rest have to.

### **Don't raise money unless you want it and it wants you.**

Such a high proportion of successful startups raise money that it might seem fundraising is one of the defining qualities of a startup. Actually it isn't. [Rapid growth](#) is what makes a company a startup. Most companies in a position to grow rapidly find that (a) taking outside money helps them grow faster, and (b) their growth potential makes it easy to attract such money. It's so common for both (a) and (b) to be true of a successful startup that practically all do raise outside money. But there may be cases where a startup either wouldn't want to grow faster, or outside money wouldn't help them to, and if you're one of them, don't raise money.

The other time not to raise money is when you won't be able to. If you try to raise money before you can [convince](#) investors, you'll not only waste your time, but also burn your reputation with those investors.

### **Be in fundraising mode or not.**

One of the things that surprises founders most about fundraising is how distracting it is. When you start fundraising, everything else grinds to a halt. The problem is not the time fundraising consumes but that it becomes the [top idea in your mind](#). A startup can't endure that level of distraction for long. An early stage startup grows mostly because the founders [make](#) it grow, and if the founders look away, growth usually drops sharply.

Because fundraising is so distracting, a startup should either be in fundraising mode or not. And when you do decide to raise money, you should focus your whole attention on it so you can get it done quickly and get back to work. [2]

You can take money from investors when you're not in fundraising mode. You just can't expend any attention on it. There are two things that take attention: convincing investors, and negotiating with them. So when you're not in fundraising mode, you should take money from investors only if they require no convincing, and are willing to invest on terms you'll take without negotiation. For example, if a reputable investor is willing to invest on a convertible note, using standard paperwork, that is either uncapped or capped at a good valuation, you can take that without having to think. [3] The terms will be whatever they turn out to be in your next equity round. And "no convincing" means just that: zero time spent meeting with investors or preparing materials for them. If an investor says they're



ready to invest, but they need you to come in for one meeting to meet some of the partners, tell them no, if you're not in fundraising mode, because that's fundraising. [4] Tell them politely; tell them you're focusing on the company right now, and that you'll get back to them when you're fundraising; but do not get sucked down the slippery slope.

Investors will try to lure you into fundraising when you're not. It's great for them if they can, because they can thereby get a shot at you before everyone else. They'll send you emails saying they want to meet to learn more about you. If you get cold-emailed by an associate at a VC firm, you shouldn't meet even if you are in fundraising mode. Deals don't happen that way. [5] But even if you get an email from a partner you should try to delay meeting till you're in fundraising mode. They may say they just want to meet and chat, but investors never just want to meet and chat. What if they like you? What if they start to talk about giving you money? Will you be able to resist having that conversation? Unless you're experienced enough at fundraising to have a casual conversation with investors that stays casual, it's safer to tell them that you'd be happy to later, when you're fundraising, but that right now you need to focus on the company. [6]

Companies that are successful at raising money in phase 2 sometimes tack on a few investors after leaving fundraising mode. This is fine; if fundraising went well, you'll be able to do it without spending time convincing them or negotiating about terms.

## **Get introductions to investors.**

Before you can talk to investors, you have to be introduced to them. If you're presenting at a Demo Day, you'll be introduced to a whole bunch simultaneously. But even if you are, you should supplement these with intros you collect yourself.

Do you have to be introduced? In phase 2, yes. Some investors will let you email them a business plan, but you can tell from the way their sites are organized that they don't really want startups to approach them directly.

Intros vary greatly in effectiveness. The best type of intro is from a well-known investor who has just invested in you. So when you get an investor to commit, ask them to introduce you to other investors they respect. [7] The next best type of intro is from a founder of a company they've funded. You can also get intros from other people in the startup community, like lawyers and reporters.

There are now sites like AngelList, FundersClub, and WeFunder that can introduce you to investors. We recommend startups treat them as auxiliary sources of money. Raise money first from leads you get yourself. Those will on average be better investors. Plus you'll have an easier time raising money on these sites once you can say you've already raised some from well-known investors.

## **Hear no till you hear yes.**

Treat investors as saying no till they unequivocally say yes, in the form of a definite offer with no contingencies.

I mentioned earlier that investors prefer to wait if they can. What's particularly dangerous for founders is the way they wait. Essentially, they lead you on. They seem like they're about to invest right up till the moment they say no. If they even

say no. Some of the worse ones never actually do say no; they just stop replying to your emails. They hope that way to get a free option on investing. If they decide later that they want to invest — usually because they've heard you're a hot deal — they can pretend they just got distracted and then restart the conversation as if they'd been about to. [8]

That's not the worst thing investors will do. Some will use language that makes it sound as if they're committing, but which doesn't actually commit them. And wishful thinking founders are happy to meet them half way. [9]

Fortunately, the next rule is a tactic for neutralizing this behavior. But to work it depends on you not being tricked by the no that sounds like yes. It's so common for founders to be misled/mistaken about this that we designed a [protocol](#) to fix the problem. If you believe an investor has committed, get them to confirm it. If you and they have different views of reality, whether the source of the discrepancy is their sketchiness or your wishful thinking, the prospect of confirming a commitment in writing will flush it out. And till they confirm, regard them as saying no.

### **Do breadth-first search weighted by expected value.**

When you talk to investors your m.o. should be breadth-first search, weighted by expected value. You should always talk to investors in parallel rather than serially. You can't afford the time it takes to talk to investors serially, plus if you only talk to one investor at a time, they don't have the pressure of other investors to make them act. But you shouldn't pay the same attention to every investor, because some are more promising prospects than others. The optimal solution is to talk to all potential investors in parallel, but give higher priority to the more promising ones. [10]

Expected value = how likely an investor is to say yes, multiplied by how good it would be if they did. So for example, an eminent investor who would invest a lot, but will be hard to convince, might have the same expected value as an obscure angel who won't invest much, but will be easy to convince. Whereas an obscure angel who will only invest a small amount, and yet needs to meet multiple times before making up his mind, has very low expected value. Meet such investors last, if at all. [11]

Doing breadth-first search weighted by expected value will save you from investors who never explicitly say no but merely drift away, because you'll drift away from them at the same rate. It protects you from investors who flake in much the same way that a distributed algorithm protects you from processors that fail. If some investor isn't returning your emails, or wants to have lots of meetings but isn't progressing toward making you an offer, you automatically focus less on them. But you have to be disciplined about assigning probabilities. You can't let how much you want an investor influence your estimate of how much they want you.

### **Know where you stand.**

How do you judge how well you're doing with an investor, when investors habitually seem more positive than they are? By looking at their actions rather than their words. Every investor has some track they need to move along from the first conversation to wiring the money, and you should always know what that track consists of, where you are on it, and how fast you're moving forward.

Never leave a meeting with an investor without asking what happens next. What more do they need in order to decide? Do they need another meeting with you? To talk about what? And how soon? Do they need to do something internally, like talk to their partners, or investigate some issue? How long do they expect it to take? Don't be too pushy, but know where you stand. If investors are vague or resist answering such questions, assume the worst; investors who are seriously interested in you will usually be happy to talk about what has to happen between now and wiring the money, because they're already running through that in their heads. [\[12\]](#)

If you're experienced at negotiations, you already know how to ask such questions. [\[13\]](#) If you're not, there's a trick you can use in this situation. Investors know you're inexperienced at raising money. Inexperience there doesn't make you unattractive. Being a noob at technology would, if you're starting a technology startup, but not being a noob at fundraising. Larry and Sergey were noobs at fundraising. So you can just confess that you're inexperienced at this and ask how their process works and where you are in it. [\[14\]](#)

### **Get the first commitment.**

The biggest factor in most investors' opinions of you is the opinion of [other investors](#). Once you start getting investors to commit, it becomes increasingly easy to get more to. But the other side of this coin is that it's often hard to get the first commitment.

Getting the first substantial offer can be half the total difficulty of fundraising. What counts as a substantial offer depends on who it's from and how much it is. Money from friends and family doesn't usually count, no matter how much. But if you get \$50k from a well known VC firm or angel investor, that will usually be enough to set things rolling. [\[15\]](#)

### **Close committed money.**

It's not a deal till the money's in the bank. I often hear inexperienced founders say things like "We've raised \$800,000," only to discover that zero of it is in the bank so far. Remember the twin fears that torment investors? The fear of missing out that makes them jump early, and the fear of jumping onto a turd that results? This is a market where people are exceptionally prone to buyer's remorse. And it's also one that furnishes them plenty of excuses to gratify it. The public markets snap startup investing around like a whip. If the Chinese economy blows up tomorrow, all bets are off. But there are lots of surprises for individual startups too, and they tend to be concentrated around fundraising. Tomorrow a big competitor could appear, or you could get C&Ded, or your cofounder could quit. [\[16\]](#)

Even a day's delay can bring news that causes an investor to change their mind. So when someone commits, get the money. Knowing where you stand doesn't end when they say they'll invest. After they say yes, know what the timetable is for getting the money, and then babysit that process till it happens. Institutional investors have people in charge of wiring money, but you may have to hunt angels down in person to collect a check.

Inexperienced investors are the ones most likely to get buyer's remorse. Established ones have learned to treat saying yes as like diving off a diving board,

and they also have more brand to preserve. But I've heard of cases of even top-tier VC firms welching on deals.

## **Avoid investors who don't "lead."**

Since getting the first offer is most of the difficulty of fundraising, that should be part of your calculation of expected value when you start. You have to estimate not just the probability that an investor will say yes, but the probability that they'd be the *first* to say yes, and the latter is not simply a constant fraction of the former. Some investors are known for deciding quickly, and those are extra valuable early on.

Conversely, an investor who will only invest once other investors have is worthless initially. And while most investors are influenced by how interested other investors are in you, there are some who have an explicit policy of only investing after other investors have. You can recognize this contemptible subspecies of investor because they often talk about "leads." They say that they don't lead, or that they'll invest once you have a lead. Sometimes they even claim to be willing to lead themselves, by which they mean they won't invest till you get \$x from other investors. (It's great if by "lead" they mean they'll invest unilaterally, and in addition will help you raise more. What's lame is when they use the term to mean they won't invest unless you can raise more elsewhere.) [\[17\]](#)

Where does this term "lead" come from? Up till a few years ago, startups raising money in phase 2 would usually raise equity rounds in which several investors invested at the same time using the same paperwork. You'd negotiate the terms with one "lead" investor, and then all the others would sign the same documents and all the money change hands at the closing.

Series A rounds still work that way, but things now work differently for most fundraising prior to the series A. Now there are rarely actual rounds before the A round, or leads for them. Now startups simply raise money from investors one at a time till they feel they have enough.

Since there are no longer leads, why do investors use that term? Because it's a more legitimate-sounding way of saying what they really mean. All they really mean is that their interest in you is a function of other investors' interest in you. I.e. the spectral signature of all mediocre investors. But when phrased in terms of leads, it sounds like there is something structural and therefore legitimate about their behavior.

When an investor tells you "I want to invest in you, but I don't lead," translate that in your mind to "No, except yes if you turn out to be a hot deal." And since that's the default opinion of any investor about any startup, they've essentially just told you nothing.

When you first start fundraising, the expected value of an investor who won't "lead" is zero, so talk to such investors last if at all.

## **Have multiple plans.**

Many investors will ask how much you're planning to raise. This question makes founders feel they should be planning to raise a specific amount. But in fact you shouldn't. It's a mistake to have fixed plans in an undertaking as unpredictable as

fundraising.

So why do investors ask how much you plan to raise? For much the same reasons a salesperson in a store will ask "How much were you planning to spend?" if you walk in looking for a gift for a friend. You probably didn't have a precise amount in mind; you just want to find something good, and if it's inexpensive, so much the better. The salesperson asks you this not because you're supposed to have a plan to spend a specific amount, but so they can show you only things that cost the most you'll pay.

Similarly, when investors ask how much you plan to raise, it's not because you're supposed to have a plan. It's to see whether you'd be a suitable recipient for the size of investment they like to make, and also to judge your ambition, reasonableness, and how far you are along with fundraising.

If you're a wizard at fundraising, you can say "We plan to raise a \$7 million series A round, and we'll be accepting termsheets next tuesday." I've known a handful of founders who could pull that off without having VCs laugh in their faces. But if you're in the inexperienced but earnest majority, the solution is analogous to the solution I recommend for [pitching](#) your startup: do the right thing and then just tell investors what you're doing.

And the right strategy, in fundraising, is to have multiple plans depending on how much you can raise. Ideally you should be able to tell investors something like: we can make it to profitability without raising any more money, but if we raise a few hundred thousand we can hire one or two smart friends, and if we raise a couple million, we can hire a whole engineering team, etc.

Different plans match different investors. If you're talking to a VC firm that only does series A rounds (though there are few of those left), it would be a waste of time talking about any but your most expensive plan. Whereas if you're talking to an angel who invests \$20k at a time and you haven't raised any money yet, you probably want to focus on your least expensive plan.

If you're so fortunate as to have to think about the upper limit on what you should raise, a good rule of thumb is to multiply the number of people you want to hire times \$15k times 18 months. In most startups, nearly all the costs are a function of the number of people, and \$15k per month is the conventional total cost (including benefits and even office space) per person. \$15k per month is high, so don't actually spend that much. But it's ok to use a high estimate when fundraising to add a margin for error. If you have additional expenses, like manufacturing, add in those at the end. Assuming you have none and you think you might hire 20 people, the most you'd want to raise is  $20 \times \$15k \times 18 = \$5.4 \text{ million}$ . [\[18\]](#)

### **Underestimate how much you want.**

Though you can focus on different plans when talking to different types of investors, you should on the whole err on the side of underestimating the amount you hope to raise.

For example, if you'd like to raise \$500k, it's better to say initially that you're trying to raise \$250k. Then when you reach \$150k you're more than half done. That sends two useful signals to investors: that you're doing well, and that they have to decide quickly because you're running out of room. Whereas if you'd said

you were raising \$500k, you'd be less than a third done at \$150k. If fundraising stalled there for an appreciable time, you'd start to read as a failure.

Saying initially that you're raising \$250k doesn't limit you to raising that much. When you reach your initial target and you still have investor interest, you can just decide to raise more. Startups do that all the time. In fact, most startups that are very successful at fundraising end up raising more than they originally intended.

I'm not saying you should lie, but that you should lower your expectations initially. There is almost no downside in starting with a low number. It not only won't cap the amount you raise, but will on the whole tend to increase it.

A good metaphor here is angle of attack. If you try to fly at too steep an angle of attack, you just stall. If you say right out of the gate that you want to raise a \$5 million series A round, unless you're in a very strong position, you not only won't get that but won't get anything. Better to start at a low angle of attack, build up speed, and then gradually increase the angle if you want.

### **Be profitable if you can.**

You will be in a much stronger position if your collection of plans includes one for raising zero dollars — i.e. if you can make it to profitability without raising any additional money. Ideally you want to be able to say to investors "We'll succeed no matter what, but raising money will help us do it faster."

There are many analogies between fundraising and dating, and this is one of the strongest. No one wants you if you seem desperate. And the best way not to seem desperate is not to *be* desperate. That's one reason we urge startups during YC to keep expenses low and to try to make it to [ramen profitability](#) before Demo Day. Though it sounds slightly paradoxical, if you want to raise money, the best thing you can do is get yourself to the point where you don't need to.

There are almost two distinct modes of fundraising: one in which founders who need money knock on doors seeking it, knowing that otherwise the company will die or at the very least people will have to be fired, and one in which founders who don't need money take some to grow faster than they could merely on their own revenues. To emphasize the distinction I'm going to name them: type A fundraising is when you don't need money, and type B fundraising is when you do.

Inexperienced founders read about famous startups doing what was type A fundraising, and decide they should raise money too, since that seems to be how startups work. Except when they raise money they don't have a clear path to profitability and are thus doing type B fundraising. And they are then surprised how difficult and unpleasant it is.

Of course not all startups can make it to ramen profitability in a few months. And some that don't still manage to have the upper hand over investors, if they have some other advantage like extraordinary growth numbers or exceptionally formidable founders. But as time passes it gets increasingly difficult to fundraise from a position of strength without being profitable. [\[19\]](#)

### **Don't optimize for valuation.**

When you raise money, what should your valuation be? The most important thing

to understand about valuation is that it's not that important.

Founders who raise money at high valuations tend to be unduly proud of it. Founders are often competitive people, and since valuation is usually the only visible number attached to a startup, they end up competing to raise money at the highest valuation. This is stupid, because fundraising is not the test that matters. The real test is revenue. Fundraising is just a means to that end. Being proud of how well you did at fundraising is like being proud of your college grades.

Not only is fundraising not the test that matters, valuation is not even the thing to optimize about fundraising. The number one thing you want from phase 2 fundraising is to get the money you need, so you can get back to focusing on the real test, the success of your company. Number two is good investors. Valuation is at best third.

The empirical evidence shows just how unimportant it is. Dropbox and Airbnb are the most successful companies we've funded so far, and they raised money after Y Combinator at premoney valuations of \$4 million and \$2.6 million respectively. Prices are so much higher now that if you can raise money at all you'll probably raise it at higher valuations than Dropbox and Airbnb. So let that satisfy your competitiveness. You're doing better than Dropbox and Airbnb! At a test that doesn't matter.

When you start fundraising, your initial valuation (or valuation cap) will be set by the deal you make with the first investor who commits. You can increase the price for later investors, if you get a lot of interest, but by default the valuation you got from the first investor becomes your asking price.

So if you're raising money from multiple investors, as most companies do in phase 2, you have to be careful to avoid raising the first from an over-eager investor at a price you won't be able to sustain. You can of course lower your price if you need to (in which case you should give the same terms to investors who invested earlier at a higher price), but you may lose a bunch of leads in the process of realizing you need to do this.

What you can do if you have eager first investors is raise money from them on an uncapped convertible note with an MFN clause. This is essentially a way of saying that the valuation cap of the note will be determined by the next investors you raise money from.

It will be easier to raise money at a lower valuation. It shouldn't be, but it is. Since phase 2 prices vary at most 10x and the big successes generate returns of at least 100x, investors should pick startups entirely based on their estimate of the probability that the company will be a big success and hardly at all on price. But although it's a mistake for investors to care about price, a significant number do. A startup that investors seem to like but won't invest in at a cap of \$x will have an easier time at \$x/2. [\[20\]](#)

### **Yes/no before valuation.**

Some investors want to know what your valuation is before they even talk to you about investing. If your valuation has already been set by a prior investment at a specific valuation or cap, you can tell them that number. But if it isn't set because you haven't closed anyone yet, and they try to push you to name a price, resist



doing so. If this would be the first investor you've closed, then this could be the tipping point of fundraising. That means closing this investor is the first priority, and you need to get the conversation onto that instead of being dragged sideways into a discussion of price.

Fortunately there is a way to avoid naming a price in this situation. And it is not just a negotiating trick; it's how you (both) should be operating. Tell them that valuation is not the most important thing to you and that you haven't thought much about it, that you are looking for investors you want to partner with and who want to partner with you, and that you should talk first about whether they want to invest at all. Then if they decide they do want to invest, you can figure out a price. But first things first.

Since valuation isn't that important and getting fundraising rolling is, we usually tell founders to give the first investor who commits as low a price as they need to. This is a safe technique so long as you combine it with the next one. [\[21\]](#)

### **Beware "valuation sensitive" investors.**

Occasionally you'll encounter investors who describe themselves as "valuation sensitive." What this means in practice is that they are compulsive negotiators who will suck up a lot of your time trying to push your price down. You should therefore never approach such investors first. While you shouldn't chase high valuations, you also don't want your valuation to be set artificially low because the first investor who committed happened to be a compulsive negotiator. Some such investors have value, but the time to approach them is near the end of fundraising, when you're in a position to say "this is the price everyone else has paid; take it or leave it" and not mind if they leave it. This way, you'll not only get market price, but it will also take less time.

Ideally you know which investors have a reputation for being "valuation sensitive" and can postpone dealing with them till last, but occasionally one you didn't know about will pop up early on. The rule of doing breadth first search weighted by expected value already tells you what to do in this case: slow down your interactions with them.

There are a handful of investors who will try to invest at a lower valuation even when your price has already been set. Lowering your price is a backup plan you resort to when you discover you've let the price get set too high to close all the money you need. So you'd only want to talk to this sort of investor if you were about to do that anyway. But since investor meetings have to be arranged at least a few days in advance and you can't predict when you'll need to resort to lowering your price, this means in practice that you should approach this type of investor last if at all.

If you're surprised by a lowball offer, treat it as a backup offer and delay responding to it. When someone makes an offer in good faith, you have a moral obligation to respond in a reasonable time. But lowballing you is a dick move that should be met with the corresponding countermove.

### **Accept offers greedily.**

I'm a little leery of using the term "greedily" when writing about fundraising lest non-programmers misunderstand me, but a greedy algorithm is simply one that



doesn't try to look into the future. A greedy algorithm takes the best of the options in front of it right now. And that is how startups should approach fundraising in phases 2 and later. Don't try to look into the future because (a) the future is unpredictable, and indeed in this business you're often being deliberately misled about it and (b) your first priority in fundraising should be to get it finished and get back to work anyway.

If someone makes you an acceptable offer, take it. If you have multiple incompatible offers, take the best. Don't reject an acceptable offer in the hope of getting a better one in the future.

These simple rules cover a wide variety of cases. If you're raising money from many investors, roll them up as they say yes. As you start to feel you've raised enough, the threshold for acceptable will start to get higher.

In practice offers exist for stretches of time, not points. So when you get an acceptable offer that would be incompatible with others (e.g. an offer to invest most of the money you need), you can tell the other investors you're talking to that you have an offer good enough to accept, and give them a few days to make their own. This could lose you some that might have made an offer if they had more time. But by definition you don't care; the initial offer was acceptable.

Some investors will try to prevent others from having time to decide by giving you an "exploding" offer, meaning one that's only valid for a few days. Offers from the very best investors explode less frequently and less rapidly — Fred Wilson never gives exploding offers, for example — because they're confident you'll pick them. But lower-tier investors sometimes give offers with very short fuses, because they believe no one who had other options would choose them. A deadline of three working days is acceptable. You shouldn't need more than that if you've been talking to investors in parallel. But a deadline any shorter is a sign you're dealing with a sketchy investor. You can usually call their bluff, and you may need to. [\[22\]](#)

It might seem that instead of accepting offers greedily, your goal should be to get the best investors as partners. That is certainly a good goal, but in phase 2 "get the best investors" only rarely conflicts with "accept offers greedily," because the best investors don't usually take any longer to decide than the others. The only case where the two strategies give conflicting advice is when you have to forgo an offer from an acceptable investor to see if you'll get an offer from a better one. If you talk to investors in parallel and push back on exploding offers with excessively short deadlines, that will almost never happen. But if it does, "get the best investors" is in the average case bad advice. The best investors are also the most selective, because they get their pick of all the startups. They reject nearly everyone they talk to, which means in the average case it's a bad trade to exchange a definite offer from an acceptable investor for a potential offer from a better one.

(The situation is different in phase 1. You can't apply to all the incubators in parallel, because some offset their schedules to prevent this. In phase 1, "accept offers greedily" and "get the best investors" do conflict, so if you want to apply to multiple incubators, you should do it in such a way that the ones you want most decide first.)

Sometimes when you're raising money from multiple investors, a series A will emerge out of those conversations, and these rules even cover what to do in that

case. When an investor starts to talk to you about a series A, keep taking smaller investments till they actually give you a termsheet. There's no practical difficulty. If the smaller investments are on convertible notes, they'll just convert into the series A round. The series A investor won't like having all these other random investors as bedfellows, but if it bothers them so much they should get on with giving you a termsheet. Till they do, you don't know for sure they will, and the greedy algorithm tells you what to do. [23]

## **Don't sell more than 25% in phase 2.**

If you do well, you will probably raise a series A round eventually. I say probably because things are changing with series A rounds. Startups may start to skip them. But only one company we've funded has so far, so tentatively assume the path to huge passes through an A round. [24]

Which means you should avoid doing things in earlier rounds that will mess up raising an A round. For example, if you've sold more than about 40% of your company total, it starts to get harder to raise an A round, because VCs worry there will not be enough stock left to keep the founders motivated.

Our rule of thumb is not to sell more than 25% in phase 2, on top of whatever you sold in phase 1, which should be less than 15%. If you're raising money on uncapped notes, you'll have to guess what the eventual equity round valuation might be. Guess conservatively.

(Since the goal of this rule is to avoid messing up the series A, there's obviously an exception if you end up raising a series A in phase 2, as a handful of startups do.)

## **Have one person handle fundraising.**

If you have multiple founders, pick one to handle fundraising so the other(s) can keep working on the company. And since the danger of fundraising is not the time taken up by the actual meetings but that it becomes the top idea in your mind, the founder who handles fundraising should make a conscious effort to insulate the other founder(s) from the details of the process. [25]

(If the founders mistrust one another, this could cause some friction. But if the founders mistrust one another, you have worse problems to worry about than how to organize fundraising.)

The founder who handles fundraising should be the CEO, who should in turn be the most formidable of the founders. Even if the CEO is a programmer and another founder is a salesperson? Yes. If you happen to be that type of founding team, you're effectively a single founder when it comes to fundraising.

It's ok to bring all the founders to meet an investor who will invest a lot, and who needs this meeting as the final step before deciding. But wait till that point. Introducing an investor to your cofounder(s) should be like introducing a girl/boyfriend to your parents — something you do only when things reach a certain stage of seriousness.

Even if there are still one or more founders focusing on the company during fundraising, growth will slow. But try to get as much growth as you can, because fundraising is a segment of time, not a point, and what happens to the company

during that time affects the outcome. If your numbers grow significantly between two investor meetings, investors will be hot to close, and if your numbers are flat or down they'll start to get cold feet.

## **You'll need an executive summary and (maybe) a deck.**

Traditionally phase 2 fundraising consists of presenting a slide deck in person to investors. Sequoia describes what such a deck should [contain](#), and since they're the customer you can take their word for it.

I say "traditionally" because I'm ambivalent about decks, and (though perhaps this is wishful thinking) they seem to be on the way out. A lot of the most successful startups we fund never make decks in phase 2. They just talk to investors and explain what they plan to do. Fundraising usually takes off fast for the startups that are most successful at it, and they're thus able to excuse themselves by saying that they haven't had time to make a deck.

You'll also want an executive summary, which should be no more than a page long and describe in the most matter of fact language what you plan to do, why it's a good idea, and what progress you've made so far. The point of the summary is to remind the investor (who may have met many startups that day) what you talked about.

Assume that if you give someone a copy of your deck or executive summary, it will be passed on to whoever you'd least like to have it. But don't refuse on that account to give copies to investors you meet. You just have to treat such leaks as a cost of doing business. In practice it's not that high a cost. Though founders are rightly indignant when their plans get leaked to competitors, I can't think of a startup whose outcome has been affected by it.

Sometimes an investor will ask you to send them your deck and/or executive summary before they decide whether to meet with you. I wouldn't do that. It's a sign they're not really interested.

## **Stop fundraising when it stops working.**

When do you stop fundraising? Ideally when you've raised enough. But what if you haven't raised as much as you'd like? When do you give up?

It's hard to give general advice about this, because there have been cases of startups that kept trying to raise money even when it seemed hopeless, and miraculously succeeded. But what I usually tell founders is to stop fundraising when you start to get a lot of air in the straw. When you're drinking through a straw, you can tell when you get to the end of the liquid because you start to get a lot of air in the straw. When your fundraising options run out, they usually run out in the same way. Don't keep sucking on the straw if you're just getting air. It's not going to get better.

## **Don't get addicted to fundraising.**

Fundraising is a chore for most founders, but some find it more interesting than working on their startup. The work at an early stage startup often consists of unglamorous [schleps](#). Whereas fundraising, when it's going well, can be quite the opposite. Instead of sitting in your grubby apartment listening to users complain

about bugs in your software, you're being offered millions of dollars by famous investors over lunch at a nice restaurant. [26]

The danger of fundraising is particularly acute for people who are good at it. It's always fun to work on something you're good at. If you're one of these people, beware. Fundraising is not what will make your company successful. Listening to users complain about bugs in your software is what will make you successful. And the big danger of getting addicted to fundraising is not merely that you'll spend too long on it or raise too much money. It's that you'll start to think of yourself as being already successful, and lose your taste for the schlep you need to undertake to actually be successful. Startups can be destroyed by this.

When I see a startup with young founders that is fabulously successful at fundraising, I mentally decrease my estimate of the probability that they'll succeed. The press may be writing about them as if they'd been anointed as the next Google, but I'm thinking "this is going to end badly."

### **Don't raise too much.**

Though only a handful of startups have to worry about this, it is possible to raise too much. The dangers of raising too much are subtle but insidious. One is that it will set impossibly high expectations. If you raise an excessive amount of money, it will be at a high valuation, and the danger of raising money at too high a valuation is that you won't be able to increase it sufficiently the next time you raise money.

A company's valuation is expected to rise each time it raises money. If not it's a sign of a company in trouble, which makes you unattractive to investors. So if you raise money in phase 2 at a post-money valuation of \$30 million, the pre-money valuation of your next round, if you want to raise one, is going to have to be at least \$50 million. And you have to be doing really, really well to raise money at \$50 million.

It's very dangerous to let the competitiveness of your current round set the performance threshold you have to meet to raise your next one, because the two are only loosely coupled.

But the money itself may be more dangerous than the valuation. The more you raise, the more you spend, and spending a lot of money can be disastrous for an early stage startup. Spending a lot makes it harder to become profitable, and perhaps even worse, it makes you more rigid, because the main way to spend money is people, and the more people you have, the harder it is to change directions. So if you do raise a huge amount of money, don't spend it. (You will find that advice almost impossible to follow, so hot will be the money burning a hole in your pocket, but I feel obliged at least to try.)

### **Be nice.**

Startups raising money occasionally alienate investors by seeming arrogant. Sometimes because they are arrogant, and sometimes because they're noobs clumsily attempting to mimic the toughness they've observed in experienced founders.

It's a mistake to behave arrogantly to investors. While there are certain situations in which certain investors like certain kinds of arrogance, investors vary greatly in

this respect, and a flick of the whip that will bring one to heel will make another roar with indignation. The only safe strategy is never to seem arrogant at all.

That will require some diplomacy if you follow the advice I've given here, because the advice I've given is essentially how to play hardball back. When you refuse to meet an investor because you're not in fundraising mode, or slow down your interactions with an investor who moves too slow, or treat a contingent offer as the no it actually is and then, by accepting offers greedily, end up leaving that investor out, you're going to be doing things investors don't like. So you must cushion the blow with soft words. At YC we tell startups they can blame us. And now that I've written this, everyone else can blame me if they want. That plus the inexperience card should work in most situations: sorry, we think you're great, but PG said startups shouldn't \_\_\_\_, and since we're new to fundraising, we feel like we have to play it safe.

The danger of behaving arrogantly is greatest when you're doing well. When everyone wants you, it's hard not to let it go to your head. Especially if till recently no one wanted you. But restrain yourself. The startup world is a small place, and startups have lots of ups and downs. This is a domain where it's more true than usual that pride goeth before a fall. [27]

Be nice when investors reject you as well. The best investors are not wedded to their initial opinion of you. If they reject you in phase 2 and you end up doing well, they'll often invest in phase 3. In fact investors who reject you are some of your warmest leads for future fundraising. Any investor who spent significant time deciding probably came close to saying yes. Often you have some internal champion who only needs a little more evidence to convince the skeptics. So it's wise not merely to be nice to investors who reject you, but (unless they behaved badly) to treat it as the beginning of a relationship.

### **The bar will be higher next time.**

Assume the money you raise in phase 2 will be the last you ever raise. You must make it to profitability on this money if you can.

Over the past several years, the investment community has evolved from a strategy of anointing a small number of winners early and then supporting them for years to a strategy of spraying money at early stage startups and then ruthlessly culling them at the next stage. This is probably the optimal strategy for investors. It's too hard to pick winners early on. Better to let the market do it for you. But it often comes as a surprise to startups how much harder it is to raise money in phase 3.

When your company is only a couple months old, all it has to be is a promising experiment that's worth funding to see how it turns out. The next time you raise money, the experiment has to have worked. You have to be on a trajectory that leads to going public. And while there are some ideas where the proof that the experiment worked might consist of e.g. query response times, usually the proof is profitability. Usually phase 3 fundraising has to be type A fundraising.

In practice there are two ways startups hose themselves between phases 2 and 3. Some are just too slow to become profitable. They raise enough money to last for two years. There doesn't seem any particular urgency to be profitable. So they don't make any effort to make money for a year. But by that time, not making

money has become habitual. When they finally decide to try, they find they can't.

The other way companies hose themselves is by letting their expenses grow too fast. Which almost always means hiring too many people. You usually shouldn't go out and hire 8 people as soon as you raise money at phase 2. Usually you want to wait till you have growth (and thus usually revenues) to justify them. A lot of VCs will encourage you to hire aggressively. VCs generally tell you to spend too much, partly because as money people they err on the side of solving problems by spending money, and partly because they want you to sell them more of your company in subsequent rounds. Don't listen to them.

### **Don't make things complicated.**

I realize it may seem odd to sum up this huge treatise by saying that my overall advice is not to make fundraising too complicated, but if you go back and look at this list you'll see it's basically a simple recipe with a lot of implications and edge cases. Avoid investors till you decide to raise money, and then when you do, talk to them all in parallel, prioritized by expected value, and accept offers greedily. That's fundraising in one sentence. Don't introduce complicated optimizations, and don't let investors introduce complications either.

Fundraising is not what will make you successful. It's just a means to an end. Your primary goal should be to get it over with and get back to what will make you successful — making things and talking to users — and the path I've described will for most startups be the surest way to that destination.

Be good, take care of yourselves, and *don't leave the path*.

### **Notes**

[1] The worst explosions happen when unpromising-seeming startups encounter mediocre investors. Good investors don't lead startups on; their reputations are too valuable. And startups that seem promising can usually get enough money from good investors that they don't have to talk to mediocre ones. It is the unpromising-seeming startups that have to resort to raising money from mediocre investors. And it's particularly damaging when these investors flake, because unpromising-seeming startups are usually more desperate for money.

(Not all unpromising-seeming startups do badly. Some are merely ugly ducklings in the sense that they violate current startup fashions.)

[2] One YC founder told me:

I think in general we've done ok at fundraising, but I managed to screw up twice at the exact same thing — trying to focus on building the company and fundraising at the same time.

[3] There is one subtle danger you have to watch out for here, which I warn about later: beware of getting too high a valuation from an eager investor, lest that set an impossibly high target when raising additional money.

[4] If they really need a meeting, then they're not ready to invest, regardless of what they say. They're still deciding, which means you're being asked to come in and convince them. Which is fundraising.

[5] Associates at VC firms regularly cold email startups. Naive founders think "Wow, a VC is interested in us!" But an associate is not a VC. They have no decision-making power. And while they may introduce startups they like to partners at their firm, the partners discriminate against deals that come to them this way. I don't know of a single VC investment that began with an associate cold-emailing a startup. If you want to approach a specific firm, get an intro to a partner from someone they respect.

It's ok to talk to an associate if you get an intro to a VC firm or they see you at a Demo Day and they begin by having an associate vet you. That's not a promising lead and should therefore get low priority, but it's not as completely worthless as a cold email.

Because the title "associate" has gotten a bad reputation, a few VC firms have started to give their associates the title "partner," which can make things very confusing. If you're a YC startup you can ask us who's who; otherwise you may have to do some research online. There may be a special title for actual partners. If someone speaks for the firm in the press or a blog on the firm's site, they're probably a real partner. If they're on boards of directors they're probably a real partner.

There are titles between "associate" and "partner," including "principal" and "venture partner." The meanings of these titles vary too much to generalize.

[6] For similar reasons, avoid casual conversations with potential acquirers. They can lead to distractions even more dangerous than fundraising. Don't even take a meeting with a potential acquirer unless you want to sell your company right now.

[7] Joshua Reeves specifically suggests asking each investor to intro you to two more investors.

Don't ask investors who say no for introductions to other investors. That will in many cases be an anti-recommendation.

[8] This is not always as deliberate as it sounds. A lot of the delays and disconnects between founders and investors are induced by the customs of the venture business, which have evolved the way they have because they suit investors' interests.

[9] One YC founder who read a draft of this essay wrote:

This is the most important section. I think it might bear stating even more clearly. "Investors will deliberately affect more interest than they

have to preserve optionality. If an investor seems very interested in you, they still probably won't invest. The solution for this is to assume the worst — that an investor is just feigning interest — until you get a definite commitment."

[10] Though you should probably pack investor meetings as closely as you can, Jeff Byun mentions one reason not to: if you pack investor meetings too closely, you'll have less time for your pitch to evolve.

Some founders deliberately schedule a handful of lame investors first, to get the bugs out of their pitch.

[11] There is not an efficient market in this respect. Some of the most useless investors are also the highest maintenance.

[12] Incidentally, this paragraph is sales 101. If you want to see it in action, go talk to a car dealer.

[13] I know one very smooth founder who used to end investor meetings with "So, can I count you in?" delivered as if it were "Can you pass the salt?" Unless you're very smooth (if you're not sure...), do not do this yourself. There is nothing more unconvincing, for an investor, than a nerdy founder trying to deliver the lines meant for a smooth one.

Investors are fine with funding nerds. So if you're a nerd, just try to be a good nerd, rather than doing a bad imitation of a smooth salesman.

[14] Ian Hogarth suggests a good way to tell how serious potential investors are: the resources they expend on you after the first meeting. An investor who's seriously interested will already be working to help you even before they've committed.

[15] In principle you might have to think about so-called "signalling risk." If a prestigious VC makes a small seed investment in you, what if they don't want to invest the next time you raise money? Other investors might assume that the VC knows you well, since they're an existing investor, and if they don't want to invest in your next round, that must mean you suck. The reason I say "in principle" is that in practice signalling hasn't been much of a problem so far. It rarely arises, and in the few cases where it does, the startup in question usually is doing badly and is doomed anyway.

If you have the luxury of choosing among seed investors, you can play it safe by excluding VC firms. But it isn't critical to.

[16] Sometimes a competitor will deliberately threaten you with a lawsuit just as you start fundraising, because they know you'll have to disclose the threat to potential investors and they hope this will make it harder for you to raise money. If this happens it will probably frighten you more than investors. Experienced investors know about this trick, and know the actual lawsuits rarely happen. So if



you're attacked in this way, be forthright with investors. They'll be more alarmed if you seem evasive than if you tell them everything.

[17] A related trick is to claim that they'll only invest contingently on other investors doing so because otherwise you'd be "undercapitalized." This is almost always bullshit. They can't estimate your minimum capital needs that precisely.

[18] You won't hire all those 20 people at once, and you'll probably have some revenues before 18 months are out. But those too are acceptable or at least accepted additions to the margin for error.

[19] Type A fundraising is so much better that it might even be worth doing something different if it gets you there sooner. One YC founder told me that if he were a first-time founder again he'd "leave ideas that are up-front capital intensive to founders with established reputations."

[20] I don't know whether this happens because they're innumerate, or because they believe they have zero ability to predict startup outcomes (in which case this behavior at least wouldn't be irrational). In either case the implications are similar.

[21] If you're a YC startup and you have an investor who for some reason insists that you decide the price, any YC partner can estimate a market price for you.

[22] You should respond in kind when investors behave upstandingly too. When an investor makes you a clean offer with no deadline, you have a moral obligation to respond promptly.

[23] Tell the investors talking to you about an A round about the smaller investments you raise as you raise them. You owe them such updates on your cap table, and this is also a good way to pressure them to act. They won't like you raising other money and may pressure you to stop, but they can't legitimately ask you to commit to them till they also commit to you. If they want you to stop raising money, the way to do it is to give you a series A termsheet with a no-shop clause.

You can relent a little if the potential series A investor has a great reputation and they're clearly working fast to get you a termsheet, particularly if a third party like YC is involved to ensure there are no misunderstandings. But be careful.

[24] The company is Weebly, which made it to profitability on a seed investment of \$650k. They did try to raise a series A in the fall of 2008 but (no doubt partly because it was the fall of 2008) the terms they were offered were so bad that they decided to skip raising an A round.

[25] Another advantage of having one founder take fundraising meetings is that you never have to negotiate in real time, which is something inexperienced

founders should avoid. One YC founder told me:

Investors are professional negotiators and can negotiate on the spot very easily. If only one founder is in the room, you can say "I need to circle back with my co-founder" before making any commitments. I used to do this all the time.

[26] You'll be lucky if fundraising feels pleasant enough to become addictive. More often you have to worry about the other extreme — becoming demoralized when investors reject you. As one (very successful) YC founder wrote after reading a draft of this:

It's hard to mentally deal with the sheer scale of rejection in fundraising and if you are not in the right mindset you will fail. Users may love you but these supposedly smart investors may not understand you at all. At this point for me, rejection still rankles but I've come to accept that investors are just not super thoughtful for the most part and you need to play the game according to certain somewhat depressing rules (many of which you are listing) in order to win.

[27] The actual sentence in the King James Bible is "Pride goeth before destruction, and an haughty spirit before a fall."

**Thanks** to Slava Akhmechet, Sam Altman, Nate Blecharczyk, Adora Cheung, Bill Clerico, John Collison, Patrick Collison, Parker Conrad, Ron Conway, Travis Deyle, Jason Freedman, Joe Gebbia, Mattan Griffl, Kevin Hale, Jacob Heller, Ian Hogarth, Justin Kan, Professor Moriarty, Nikhil Nirmel, David Petersen, Geoff Ralston, Joshua Reeves, Yuri Sagalov, Emmett Shear, Rajat Suri, Garry Tan, and Nick Tomarelli for reading drafts of this.

[Russian Translation](#)

# Before the Startup

October 2014

*(This essay is derived from a guest lecture in Sam Altman's [startup class](#) at Stanford. It's intended for college students, but much of it is applicable to potential founders at other ages.)*

One of the advantages of having kids is that when you have to give advice, you can ask yourself "what would I tell my own kids?" My kids are little, but I can imagine what I'd tell them about startups if they were in college, and that's what I'm going to tell you.

Startups are very counterintuitive. I'm not sure why. Maybe it's just because knowledge about them hasn't permeated our culture yet. But whatever the reason, starting a startup is a task where you can't always trust your instincts.

It's like skiing in that way. When you first try skiing and you want to slow down, your instinct is to lean back. But if you lean back on skis you fly down the hill out of control. So part of learning to ski is learning to suppress that impulse. Eventually you get new habits, but at first it takes a conscious effort. At first there's a list of things you're trying to remember as you start down the hill.

Startups are as unnatural as skiing, so there's a similar list for startups. Here I'm going to give you the first part of it — the things to remember if you want to prepare yourself to start a startup.

## Counterintuitive

The first item on it is the fact I already mentioned: that startups are so weird that if you trust your instincts, you'll make a lot of mistakes. If you know nothing more than this, you may at least pause before making them.

When I was running Y Combinator I used to joke that our function was to tell founders things they would ignore. It's really true. Batch after batch, the YC partners warn founders about mistakes they're about to make, and the founders ignore them, and then come back a year later and say "I wish we'd listened."

Why do the founders ignore the partners' advice? Well, that's the thing about

counterintuitive ideas: they contradict your intuitions. They seem wrong. So of course your first impulse is to disregard them. And in fact my joking description is not merely the curse of Y Combinator but part of its *raison d'être*. If founders' instincts already gave them the right answers, they wouldn't need us. You only need other people to give you advice that surprises you. That's why there are a lot of ski instructors and not many running instructors. [1]

You can, however, trust your instincts about people. And in fact one of the most common mistakes young founders make is not to do that enough. They get involved with people who seem impressive, but about whom they feel some misgivings personally. Later when things blow up they say "I knew there was something off about him, but I ignored it because he seemed so impressive."

If you're thinking about getting involved with someone — as a cofounder, an employee, an investor, or an acquirer — and you have misgivings about them, trust your gut. If someone seems slippery, or bogus, or a jerk, don't ignore it.

This is one case where it pays to be self-indulgent. Work with people you genuinely like, and you've known long enough to be sure.

## **Expertise**

The second counterintuitive point is that it's not that important to know a lot about startups. The way to succeed in a startup is not to be an expert on startups, but to be an expert on your users and the problem you're solving for them. Mark Zuckerberg didn't succeed because he was an expert on startups. He succeeded despite being a complete noob at startups, because he understood his users really well.

If you don't know anything about, say, how to raise an angel round, don't feel bad on that account. That sort of thing you can learn when you need to, and forget after you've done it.

In fact, I worry it's not merely unnecessary to learn in great detail about the mechanics of startups, but possibly somewhat dangerous. If I met an undergrad who knew all about convertible notes and employee agreements and (God forbid) class FF stock, I wouldn't think "here is someone who is way ahead of their peers." It would set off alarms. Because another of the characteristic mistakes of young founders is to go through the motions of starting a startup. They make up some plausible-sounding idea, raise money at a good valuation, rent a cool office, hire a bunch of people. From the outside that seems like what startups do. But the next step after rent a cool office and hire a bunch of people is: gradually realize how completely fucked they are, because while imitating all the outward forms of a startup they have neglected the one thing that's actually essential: making something people want.

## **Game**

We saw this happen so often that we made up a name for it: playing house. Eventually I realized why it was happening. The reason young founders go through the motions of starting a startup is because that's what they've been trained to do for their whole lives up to that point. Think about what you have to do to get into college, for example. Extracurricular activities, check. Even in college classes most of the work is as artificial as running laps.

I'm not attacking the educational system for being this way. There will always be a certain amount of fakeness in the work you do when you're being taught something, and if you measure their performance it's inevitable that people will exploit the difference to the point where much of what you're measuring is artifacts of the fakeness.

I confess I did it myself in college. I found that in a lot of classes there might only be 20 or 30 ideas that were the right shape to make good exam questions. The way I studied for exams in these classes was not (except incidentally) to master the material taught in the class, but to make a list of potential exam questions and work out the answers in advance. When I walked into the final, the main thing I'd be feeling was curiosity about which of my questions would turn up on the exam. It was like a game.

It's not surprising that after being trained for their whole lives to play such games, young founders' first impulse on starting a startup is to try to figure out the tricks for winning at this new game. Since fundraising appears to be the measure of success for startups (another classic noob mistake), they always want to know what the tricks are for convincing investors. We tell them the best way to [convince investors](#) is to make a startup that's actually doing well, meaning [growing fast](#), and then simply tell investors so. Then they want to know what the tricks are for growing fast. And we have to tell them the best way to do that is simply to make something people want.

So many of the conversations YC partners have with young founders begin with the founder asking "How do we..." and the partner replying "Just..."

Why do the founders always make things so complicated? The reason, I realized, is that they're looking for the trick.

So this is the third counterintuitive thing to remember about startups: starting a startup is where gaming the system stops working. Gaming the system may continue to work if you go to work for a big company. Depending on how broken the company is, you can succeed by sucking up to the right people, giving the impression of productivity, and so on. [2] But that doesn't work with startups. There is no boss to trick, only users, and all users care about is whether your product does what they want. Startups are as impersonal as physics. You have to make something people want, and you prosper only to the extent you do.

The dangerous thing is, faking does work to some degree on investors. If you're super good at sounding like you know what you're talking about, you can fool

investors for at least one and perhaps even two rounds of funding. But it's not in your interest to. The company is ultimately doomed. All you're doing is wasting your own time riding it down.

So stop looking for the trick. There are tricks in startups, as there are in any domain, but they are an order of magnitude less important than solving the real problem. A founder who knows nothing about fundraising but has made something users love will have an easier time raising money than one who knows every trick in the book but has a flat usage graph. And more importantly, the founder who has made something users love is the one who will go on to succeed after raising the money.

Though in a sense it's bad news in that you're deprived of one of your most powerful weapons, I think it's exciting that gaming the system stops working when you start a startup. It's exciting that there even exist parts of the world where you win by doing good work. Imagine how depressing the world would be if it were all like school and big companies, where you either have to spend a lot of time on bullshit things or lose to people who do. [3] I would have been delighted if I'd realized in college that there were parts of the real world where gaming the system mattered less than others, and a few where it hardly mattered at all. But there are, and this variation is one of the most important things to consider when you're thinking about your future. How do you win in each type of work, and what would you like to win by doing? [4]

## **All-Consuming**

That brings us to our fourth counterintuitive point: startups are all-consuming. If you start a startup, it will take over your life to a degree you cannot imagine. And if your startup succeeds, it will take over your life for a long time: for several years at the very least, maybe for a decade, maybe for the rest of your working life. So there is a real opportunity cost here.

Larry Page may seem to have an enviable life, but there are aspects of it that are unenviable. Basically at 25 he started running as fast as he could and it must seem to him that he hasn't stopped to catch his breath since. Every day new shit happens in the Google empire that only the CEO can deal with, and he, as CEO, has to deal with it. If he goes on vacation for even a week, a whole week's backlog of shit accumulates. And he has to bear this uncomplainingly, partly because as the company's daddy he can never show fear or weakness, and partly because billionaires get less than zero sympathy if they talk about having difficult lives. Which has the strange side effect that the difficulty of being a successful startup founder is concealed from almost everyone except those who've done it.

Y Combinator has now funded several companies that can be called big successes, and in every single case the founders say the same thing. It never gets any easier. The nature of the problems change. You're worrying about construction delays at your London office instead of the broken air conditioner in your studio apartment. But the total volume of worry never decreases; if anything it increases.

Starting a successful startup is similar to having kids in that it's like a button you push that changes your life irrevocably. And while it's truly wonderful having kids, there are a lot of things that are easier to do before you have them than after. Many of which will make you a better parent when you do have kids. And since you can delay pushing the button for a while, most people in rich countries do.

Yet when it comes to startups, a lot of people seem to think they're supposed to start them while they're still in college. Are you crazy? And what are the universities thinking? They go out of their way to ensure their students are well supplied with contraceptives, and yet they're setting up entrepreneurship programs and startup incubators left and right.

To be fair, the universities have their hand forced here. A lot of incoming students are interested in startups. Universities are, at least de facto, expected to prepare them for their careers. So students who want to start startups hope universities can teach them about startups. And whether universities can do this or not, there's some pressure to claim they can, lest they lose applicants to other universities that do.

Can universities teach students about startups? Yes and no. They can teach students about startups, but as I explained before, this is not what you need to know. What you need to learn about are the needs of your own users, and you can't do that until you actually start the company. [5] So starting a startup is intrinsically something you can only really learn by doing it. And it's impossible to do that in college, for the reason I just explained: startups take over your life. You can't start a startup for real as a student, because if you start a startup for real you're not a student anymore. You may be nominally a student for a bit, but you won't even be that for long. [6]

Given this dichotomy, which of the two paths should you take? Be a real student and not start a startup, or start a real startup and not be a student? I can answer that one for you. Do not start a startup in college. How to start a startup is just a subset of a bigger problem you're trying to solve: how to have a good life. And though starting a startup can be part of a good life for a lot of ambitious people, age 20 is not the optimal time to do it. Starting a startup is like a brutally fast depth-first search. Most people should still be searching breadth-first at 20.

You can do things in your early 20s that you can't do as well before or after, like plunge deeply into projects on a whim and travel super cheaply with no sense of a deadline. For unambitious people, this sort of thing is the dreaded "failure to launch," but for the ambitious ones it can be an incomparably valuable sort of exploration. If you start a startup at 20 and you're sufficiently successful, you'll never get to do it. [7]

Mark Zuckerberg will never get to bum around a foreign country. He can do other things most people can't, like charter jets to fly him to foreign countries. But success has taken a lot of the serendipity out of his life. Facebook is running him

as much as he's running Facebook. And while it can be very cool to be in the grip of a project you consider your life's work, there are advantages to serendipity too, especially early in life. Among other things it gives you more options to choose your life's work from.

There's not even a tradeoff here. You're not sacrificing anything if you forgo starting a startup at 20, because you're more likely to succeed if you wait. In the unlikely case that you're 20 and one of your side projects takes off like Facebook did, you'll face a choice of running with it or not, and it may be reasonable to run with it. But the usual way startups take off is for the founders to [make them](#) take off, and it's gratuitously stupid to do that at 20.

## Try

Should you do it at any age? I realize I've made startups sound pretty hard. If I haven't, let me try again: starting a startup is really hard. What if it's too hard? How can you tell if you're up to this challenge?

The answer is the fifth counterintuitive point: you can't tell. Your life so far may have given you some idea what your prospects might be if you tried to become a mathematician, or a professional football player. But unless you've had a very strange life you haven't done much that was [like](#) being a startup founder. Starting a startup will change you a lot. So what you're trying to estimate is not just what you are, but what you could grow into, and who can do that?

For the past 9 years it was my job to predict whether people would have what it took to start successful startups. It was easy to tell how smart they were, and most people reading this will be over that threshold. The hard part was predicting how tough and ambitious they would become. There may be no one who has more experience at trying to predict that, so I can tell you how much an expert can know about it, and the answer is: not much. I learned to keep a completely open mind about which of the startups in each batch would turn out to be the stars.

The founders sometimes think they know. Some arrive feeling sure they will ace Y Combinator just as they've aced every one of the (few, artificial, easy) tests they've faced in life so far. Others arrive wondering how they got in, and hoping YC doesn't discover whatever mistake caused it to accept them. But there is little correlation between founders' initial attitudes and how well their companies do.

I've read that the same is true in the military — that the swaggering recruits are no more likely to turn out to be really tough than the quiet ones. And probably for the same reason: that the tests involved are so different from the ones in their previous lives.

If you're absolutely terrified of starting a startup, you probably shouldn't do it. But if you're merely unsure whether you're up to it, the only way to find out is to try. Just not now.



## Ideas

So if you want to start a startup one day, what should you do in college? There are only two things you need initially: an idea and cofounders. And the m.o. for getting both is the same. Which leads to our sixth and last counterintuitive point: that the way to get startup ideas is not to try to think of startup ideas.

I've written a whole [essay](#) on this, so I won't repeat it all here. But the short version is that if you make a conscious effort to think of startup ideas, the ideas you come up with will not merely be bad, but bad and plausible-sounding, meaning you'll waste a lot of time on them before realizing they're bad.

The way to come up with good startup ideas is to take a step back. Instead of making a conscious effort to think of startup ideas, turn your mind into the type that startup ideas form in without any conscious effort. In fact, so unconsciously that you don't even realize at first that they're startup ideas.

This is not only possible, it's how Apple, Yahoo, Google, and Facebook all got started. None of these companies were even meant to be companies at first. They were all just side projects. The best startups almost have to start as side projects, because great ideas tend to be such outliers that your conscious mind would reject them as ideas for companies.

Ok, so how do you turn your mind into the type that startup ideas form in unconsciously? (1) Learn a lot about things that matter, then (2) work on problems that interest you (3) with people you like and respect. The third part, incidentally, is how you get cofounders at the same time as the idea.

The first time I wrote that paragraph, instead of "learn a lot about things that matter," I wrote "become good at some technology." But that prescription, though sufficient, is too narrow. What was special about Brian Chesky and Joe Gebbia was not that they were experts in technology. They were good at design, and perhaps even more importantly, they were good at organizing groups and making projects happen. So you don't have to work on technology per se, so long as you work on problems demanding enough to stretch you.

What kind of problems are those? That is very hard to answer in the general case. History is full of examples of young people who were working on important problems that [no one else](#) at the time thought were important, and in particular that their parents didn't think were important. On the other hand, history is even fuller of examples of parents who thought their kids were wasting their time and who were right. So how do you know when you're working on real stuff? [\[8\]](#)

I know how *I* know. Real problems are interesting, and I am self-indulgent in the sense that I always want to work on interesting things, even if no one else cares about them (in fact, especially if no one else cares about them), and find it very hard to make myself work on boring things, even if they're supposed to be important.

My life is full of case after case where I worked on something just because it seemed interesting, and it turned out later to be useful in some worldly way. [Y Combinator itself](#) was something I only did because it seemed interesting. So I seem to have some sort of internal compass that helps me out. But I don't know what other people have in their heads. Maybe if I think more about this I can come up with heuristics for recognizing genuinely interesting problems, but for the moment the best I can offer is the hopelessly question-begging advice that if you have a taste for genuinely interesting problems, indulging it energetically is the best way to prepare yourself for a startup. And indeed, probably also the best way to live. [\[9\]](#)

But although I can't explain in the general case what counts as an interesting problem, I can tell you about a large subset of them. If you think of technology as something that's spreading like a sort of fractal stain, every moving point on the edge represents an interesting problem. So one guaranteed way to turn your mind into the type that has good startup ideas is to get yourself to the leading edge of some technology — to cause yourself, as Paul Buchheit put it, to "live in the future." When you reach that point, ideas that will seem to other people uncannily prescient will seem obvious to you. You may not realize they're startup ideas, but you'll know they're something that ought to exist.

For example, back at Harvard in the mid 90s a fellow grad student of my friends Robert and Trevor wrote his own voice over IP software. He didn't mean it to be a startup, and he never tried to turn it into one. He just wanted to talk to his girlfriend in Taiwan without paying for long distance calls, and since he was an expert on networks it seemed obvious to him that the way to do it was turn the sound into packets and ship it over the Internet. He never did any more with his software than talk to his girlfriend, but this is exactly the way the best startups get started.

So strangely enough the optimal thing to do in college if you want to be a successful startup founder is not some sort of new, vocational version of college focused on "entrepreneurship." It's the classic version of college as education for its own sake. If you want to start a startup after college, what you should do in college is learn powerful things. And if you have genuine intellectual curiosity, that's what you'll naturally tend to do if you just follow your own inclinations. [\[10\]](#)

The component of entrepreneurship that really matters is domain expertise. The way to become Larry Page was to become an expert on search. And the way to become an expert on search was to be driven by genuine curiosity, not some ulterior motive.

At its best, starting a startup is merely an ulterior motive for curiosity. And you'll do it best if you introduce the ulterior motive toward the end of the process.

So here is the ultimate advice for young would-be startup founders, boiled down to two words: just learn.

## Notes

[1] Some founders listen more than others, and this tends to be a [predictor of success](#). One of the things I remember about the Airbnbs during YC is how intently they listened.

[2] In fact, this is one of the reasons startups are possible. If big companies weren't plagued by internal inefficiencies, they'd be proportionately more effective, leaving less room for startups.

[3] In a startup you have to spend a lot of time on [schleps](#), but this sort of work is merely unglamorous, not bogus.

[4] What should you do if your true calling is gaming the system? Management consulting.

[5] The company may not be incorporated, but if you start to get significant numbers of users, you've started it, whether you realize it yet or not.

[6] It shouldn't be that surprising that colleges can't teach students how to be good startup founders, because they can't teach them how to be good employees either.

The way universities "teach" students how to be employees is to hand off the task to companies via internship programs. But you couldn't do the equivalent thing for startups, because by definition if the students did well they would never come back.

[7] Charles Darwin was 22 when he received an invitation to travel aboard the HMS Beagle as a naturalist. It was only because he was otherwise unoccupied, to a degree that alarmed his family, that he could accept it. And yet if he hadn't we probably would not know his name.

[8] Parents can sometimes be especially conservative in this department. There are some whose definition of important problems includes only those on the critical path to med school.

[9] I did manage to think of a heuristic for detecting whether you have a taste for interesting ideas: whether you find known boring ideas intolerable. Could you endure studying literary theory, or working in middle management at a large company?

[10] In fact, if your goal is to start a startup, you can stick even more closely to the ideal of a liberal education than past generations have. Back when students focused mainly on getting a job after college, they thought at least a little about how the courses they took might look to an employer. And perhaps even worse, they might shy away from taking a difficult class lest they get a low grade, which would harm their all-important GPA. Good news: users [don't care](#) what your GPA was. And I've never heard of investors caring either. Y Combinator certainly never asks what classes you took in college or what grades you got in them.

**Thanks** to Sam Altman, Paul Buchheit, John Collison, Patrick Collison, Jessica Livingston, Robert Morris, Geoff Ralston, and Fred Wilson for reading drafts of this.

[Arabic Translation](#)

# Mean People Fail

November 2014

It struck me recently how few of the most successful people I know are mean. There are exceptions, but remarkably few.

Meanness isn't rare. In fact, one of the things the internet has shown us is how mean people can be. A few decades ago, only famous people and professional writers got to publish their opinions. Now everyone can, and we can all see the long tail of meanness that had previously been hidden.

And yet while there are clearly a lot of mean people out there, there are next to none among the most successful people I know. What's going on here? Are meanness and success inversely correlated?

Part of what's going on, of course, is selection bias. I only know people who work in certain fields: startup founders, programmers, professors. I'm willing to believe that successful people in other fields are mean. Maybe successful hedge fund managers are mean; I don't know enough to say. It seems quite likely that most successful drug lords are mean. But there are at least big chunks of the world that mean people don't rule, and that territory seems to be growing.

My wife and Y Combinator cofounder Jessica is one of those rare people who have x-ray vision for character. Being married to her is like standing next to an airport baggage scanner. She came to the startup world from investment banking, and she has always been struck both by how consistently successful startup founders turn out to be good people, and how consistently bad people fail as startup founders.

Why? I think there are several reasons. One is that being mean makes you stupid. That's why I hate fights. You never do your best work in a fight, because fights are not sufficiently general. Winning is always a function of the situation and the people involved. You don't win fights by thinking of big ideas but by thinking of tricks that work in one particular case. And yet fighting is just as much work as thinking about real problems. Which is particularly painful to someone who cares how their brain is used: your brain goes fast but you get nowhere, like a car spinning its wheels.

Startups don't win by attacking. They win by transcending. There are exceptions of course, but usually the way to win is to race ahead, not to stop and fight.

Another reason mean founders lose is that they can't get the best people to work for them. They can hire people who will put up with them because they need a job. But the best people have other options. A mean person can't convince the best people to work for him unless he is super convincing. And while having the best people helps any organization, it's critical for startups.

There is also a complementary force at work: if you want to build great things, it helps to be driven by a spirit of benevolence. The startup founders who end up richest are not the ones driven by money. The ones driven by money take the big acquisition offer that nearly every successful startup gets en route. [1] The ones who keep going are driven by something else. They may not say so explicitly, but they're usually trying to improve the world. Which means people with a desire to improve the world have a natural advantage. [2]

The exciting thing is that startups are not just one random type of work in which meanness and success are inversely correlated. This kind of work is the future.

For most of history success meant control of scarce resources. One got that by fighting, whether literally in the case of pastoral nomads driving hunter-gatherers into marginal lands, or metaphorically in the case of Gilded Age financiers contending with one another to assemble railroad monopolies. For most of history, success meant success at zero-sum games. And in most of them meanness was not a handicap but probably an advantage.

That is changing. Increasingly the games that matter are not zero-sum. Increasingly you win not by fighting to get control of a scarce resource, but by having new ideas and building new things. [3]

There have long been games where you won by having new ideas. In the third century BC, Archimedes won by doing that. At least until an invading Roman army killed him. Which illustrates why this change is happening: for new ideas to matter, you need a certain degree of civil order. And not just not being at war. You also need to prevent the sort of economic violence that nineteenth century magnates practiced against one another and communist countries practiced against their citizens. People need to feel that what they create can't be stolen. [4]

That has always been the case for thinkers, which is why this trend began with them. When you think of successful people from history who weren't ruthless, you get mathematicians and writers and artists. The exciting thing is that their m.o. seems to be spreading. The games played by intellectuals are leaking into the real world, and this is reversing the historical polarity of the relationship between meanness and success.

So I'm really glad I stopped to think about this. Jessica and I have always worked hard to teach our kids not to be mean. We tolerate noise and mess and junk food, but not meanness. And now I have both an additional reason to crack down on it, and an additional argument to use when I do: that being mean makes you fail.

## Notes

[1] I'm not saying all founders who take big acquisition offers are driven only by money, but rather that those who don't aren't. Plus one can have benevolent motives for being driven by money — for example, to take care of one's family, or to be free to work on projects that improve the world.

[2] It's unlikely that every successful startup improves the world. But their founders, like parents, truly believe they do. Successful founders are in love with their companies. And while this sort of love is as blind as the love people have for one another, it is genuine.

[3] [Peter Thiel](#) would point out that successful founders still get rich from controlling monopolies, just monopolies they create rather than ones they capture. And while this is largely true, it means a big change in the sort of person who wins.

[4] To be fair, the Romans didn't mean to kill Archimedes. The Roman commander specifically ordered that he be spared. But he got killed in the chaos anyway.

In sufficiently disordered times, even thinking requires control of scarce resources, because living at all is a scarce resource.

**Thanks** to Sam Altman, Ron Conway, Daniel Gackle, Jessica Livingston, Robert Morris, Geoff Ralston, and Fred Wilson for reading drafts of this.

[Portuguese Translation](#)

[Japanese Translation](#)

[Arabic Translation](#)

# The Fatal Pinch

December 2014

Many startups go through a point a few months before they die where although they have a significant amount of money in the bank, they're also losing a lot each month, and revenue growth is either nonexistent or mediocre. The company has, say, 6 months of runway. Or to put it more brutally, 6 months before they're out of business. They expect to avoid that by raising more from investors. [\[1\]](#)

That last sentence is the fatal one.

There may be nothing founders are so prone to delude themselves about as how interested investors will be in giving them additional funding. It's hard to convince investors the first time too, but founders expect that. What bites them the second time is a confluence of three forces:

1. The company is spending more now than it did the first time it raised money.
2. Investors have much higher standards for companies that have already raised money.
3. The company is now starting to read as a failure. The first time it raised money, it was neither a success nor a failure; it was too early to ask. Now it's possible to ask that question, and the default answer is failure, because at this point that is the default outcome.

I'm going to call the situation I described in the first paragraph "the fatal pinch." I try to resist coining phrases, but making up a name for this situation may snap founders into realizing when they're in it.

One of the things that makes the fatal pinch so dangerous is that it's self-reinforcing. Founders overestimate their chances of raising more money, and so are slack about reaching profitability, which further decreases their chances of raising money.

Now that you know about the fatal pinch, how do you avoid it? Y Combinator tells founders who raise money to act as if it's the last they'll ever get. Because the self-reinforcing nature of this situation works the other way too: the less you need further investment, the easier it is to get.

What do you do if you're already in the fatal pinch? The first step is to re-evaluate



the probability of raising more money. I will now, by an amazing feat of clairvoyance, do this for you: the probability is zero. [2]

Three options remain: you can shut down the company, you can increase how much you make, and you can decrease how much you spend.

You should shut down the company if you're certain it will fail no matter what you do. Then at least you can give back the money you have left, and save yourself however many months you would have spent riding it down.

Companies rarely *have* to fail though. What I'm really doing here is giving you the option of admitting you've already given up.

If you don't want to shut down the company, that leaves increasing revenues and decreasing expenses. In most startups, expenses = people, and decreasing expenses = firing people. [3] Deciding to fire people is usually hard, but there's one case in which it shouldn't be: when there are people you already know you should fire but you're in denial about it. If so, now's the time.

If that makes you profitable, or will enable you to make it to profitability on the money you have left, you've avoided the immediate danger.

Otherwise you have three options: you either have to fire good people, get some or all of the employees to take less salary for a while, or increase revenues.

Getting people to take less salary is a weak solution that will only work when the problem isn't too bad. If your current trajectory won't quite get you to profitability but you can get over the threshold by cutting salaries a little, you might be able to make the case to everyone for doing it. Otherwise you're probably just postponing the problem, and that will be obvious to the people whose salaries you're proposing to cut. [4]

Which leaves two options, firing good people and making more money. While trying to balance them, keep in mind the eventual goal: to be a successful product company in the sense of having a single thing lots of people use.

You should lean more toward firing people if the source of your trouble is overhiring. If you went out and hired 15 people before you even knew what you were building, you've created a broken company. You need to figure out what you're building, and it will probably be easier to do that with a handful of people than 15. Plus those 15 people might not even be the ones you need for whatever you end up building. So the solution may be to shrink and then figure out what direction to grow in. After all, you're not doing those 15 people any favors if you fly the company into ground with them aboard. They'll all lose their jobs eventually, along with all the time they expended on this doomed company.

Whereas if you only have a handful of people, it may be better to focus on trying to make more money. It may seem facile to suggest a startup make more money,

as if that could be done for the asking. Usually a startup is already trying as hard as it can to sell whatever it sells. What I'm suggesting here is not so much to try harder to make money but to try to make money in a different way. For example, if you have only one person selling while the rest are writing code, consider having everyone work on selling. What good will more code do you when you're out of business? If you have to write code to close a certain deal, go ahead; that follows from everyone working on selling. But only work on whatever will get you the most revenue the soonest.

Another way to make money differently is to sell different things, and in particular to do more consultingish work. I say consultingish because there is a long slippery slope from making products to pure consulting, and you don't have to go far down it before you start to offer something really attractive to customers. Although your product may not be very appealing yet, if you're a startup your programmers will often be way better than the ones your customers have. Or you may have expertise in some new field they don't understand. So if you change your sales conversations just a little from "do you want to buy our product?" to "what do you need that you'd pay a lot for?" you may find it's suddenly a lot easier to extract money from customers.

Be ruthlessly mercenary when you start doing this, though. You're trying to save your company from death here, so make customers pay a lot, quickly. And to the extent you can, try to avoid the worst pitfalls of consulting. The ideal thing might be if you built a precisely defined derivative version of your product for the customer, and it was otherwise a straight product sale. You keep the IP and no billing by the hour.

In the best case, this consultingish work may not be just something you do to survive, but may turn out to be the [thing-that-doesn't-scale](#) that defines your company. Don't expect it to be, but as you dive into individual users' needs, keep your eyes open for narrow openings that have wide vistas beyond.

There is usually so much demand for custom work that unless you're really incompetent there has to be some point down the slope of consulting at which you can survive. But I didn't use the term slippery slope by accident; customers' insatiable demand for custom work will always be pushing you toward the bottom. So while you'll probably survive, the problem now becomes to survive with the least damage and distraction.

The good news is, plenty of successful startups have passed through near-death experiences and gone on to flourish. You just have to realize in time that you're near death. And if you're in the fatal pinch, you are.

[1] There are a handful of companies that can't reasonably expect to make money for the first year or two, because what they're building takes so long. For these companies substitute "progress" for "revenue growth." You're not one of these companies unless your initial investors agreed in advance that you were. And frankly even these companies wish they weren't, because the illiquidity of "progress" puts them at the mercy of investors.

[2] There's a variant of the fatal pinch where your existing investors help you along by promising to invest more. Or rather, where you read them as promising to invest more, while they think they're just mentioning the possibility. The way to solve this problem, if you have 8 months of runway or less, is to try to get the money right now. Then you'll either get the money, in which case (immediate) problem solved, or at least prevent your investors from helping you to remain in denial about your fundraising prospects.

[3] Obviously, if you have significant expenses other than salaries that you can eliminate, do it now.

[4] Unless of course the source of the problem is that you're paying yourselves high salaries. If by cutting the founders' salaries to the minimum you need, you can make it to profitability, you should. But it's a bad sign if you needed to read this to realize that.

**Thanks** to Sam Altman, Paul Buchheit, Jessica Livingston, and Geoff Ralston for reading drafts of this.

[Arabic Translation](#)

# How You Know

December 2014

I've read Villehardouin's chronicle of the Fourth Crusade at least two times, maybe three. And yet if I had to write down everything I remember from it, I doubt it would amount to much more than a page. Multiply this times several hundred, and I get an uneasy feeling when I look at my bookshelves. What use is it to read all these books if I remember so little from them?

A few months ago, as I was reading Constance Reid's excellent biography of Hilbert, I figured out if not the answer to this question, at least something that made me feel better about it. She writes:

Hilbert had no patience with mathematical lectures which filled the students with facts but did not teach them how to frame a problem and solve it. He often used to tell them that "a perfect formulation of a problem is already half its solution."

That has always seemed to me an important point, and I was even more convinced of it after hearing it confirmed by Hilbert.

But how had I come to believe in this idea in the first place? A combination of my own experience and other things I'd read. None of which I could at that moment remember! And eventually I'd forget that Hilbert had confirmed it too. But my increased belief in the importance of this idea would remain something I'd learned from this book, even after I'd forgotten I'd learned it.

Reading and experience train your model of the world. And even if you forget the experience or what you read, its effect on your model of the world persists. Your mind is like a compiled program you've lost the source of. It works, but you don't know why.

The place to look for what I learned from Villehardouin's chronicle is not what I remember from it, but my mental models of the crusades, Venice, medieval culture, siege warfare, and so on. Which doesn't mean I couldn't have read more attentively, but at least the harvest of reading is not so miserably small as it might seem.

This is one of those things that seem obvious in retrospect. But it was a surprise to me and presumably would be to anyone else who felt uneasy about (apparently)

forgetting so much they'd read.

Realizing it does more than make you feel a little better about forgetting, though. There are specific implications.

For example, reading and experience are usually "compiled" at the time they happen, using the state of your brain at that time. The same book would get compiled differently at different points in your life. Which means it is very much worth reading important books multiple times. I always used to feel some misgivings about rereading books. I unconsciously lumped reading together with work like carpentry, where having to do something again is a sign you did it wrong the first time. Whereas now the phrase "already read" seems almost ill-formed.

Intriguingly, this implication isn't limited to books. Technology will increasingly make it possible to relive our experiences. When people do that today it's usually to enjoy them again (e.g. when looking at pictures of a trip) or to find the origin of some bug in their compiled code (e.g. when Stephen Fry succeeded in remembering the childhood trauma that prevented him from singing). But as technologies for recording and playing back your life improve, it may become common for people to relive experiences without any goal in mind, simply to learn from them again as one might when rereading a book.

Eventually we may be able not just to play back experiences but also to index and even edit them. So although not knowing how you know things may seem part of being human, it may not be.

**Thanks** to Sam Altman, Jessica Livingston, and Robert Morris for reading drafts of this.

[Japanese Translation](#)

# How to Be an Expert in a Changing World

December 2014

If the world were static, we could have monotonically increasing confidence in our beliefs. The more (and more varied) experience a belief survived, the less likely it would be false. Most people implicitly believe something like this about their opinions. And they're justified in doing so with opinions about things that don't change much, like human nature. But you can't trust your opinions in the same way about things that change, which could include practically everything else.

When experts are wrong, it's often because they're experts on an earlier version of the world.

Is it possible to avoid that? Can you protect yourself against obsolete beliefs? To some extent, yes. I spent almost a decade investing in early stage startups, and curiously enough protecting yourself against obsolete beliefs is exactly what you have to do to succeed as a startup investor. Most really good startup ideas look like bad ideas at first, and many of those look bad specifically because some change in the world just switched them from bad to good. I spent a lot of time learning to recognize such ideas, and the techniques I used may be applicable to ideas in general.

The first step is to have an explicit belief in change. People who fall victim to a monotonically increasing confidence in their opinions are implicitly concluding the world is static. If you consciously remind yourself it isn't, you start to look for change.

Where should one look for it? Beyond the moderately useful generalization that human nature doesn't change much, the unfortunate fact is that change is hard to predict. This is largely a tautology but worth remembering all the same: change that matters usually comes from an unforeseen quarter.

So I don't even try to predict it. When I get asked in interviews to predict the future, I always have to struggle to come up with something plausible-sounding on the fly, like a student who hasn't prepared for an exam. [\[1\]](#) But it's not out of laziness that I haven't prepared. It seems to me that beliefs about the future are so rarely correct that they usually aren't worth the extra rigidity they impose, and

that the best strategy is simply to be aggressively open-minded. Instead of trying to point yourself in the right direction, admit you have no idea what the right direction is, and try instead to be super sensitive to the winds of change.

It's ok to have working hypotheses, even though they may constrain you a bit, because they also motivate you. It's exciting to chase things and exciting to try to guess answers. But you have to be disciplined about not letting your hypotheses harden into anything more. [2]

I believe this passive m.o. works not just for evaluating new ideas but also for having them. The way to come up with new ideas is not to try explicitly to, but to try to solve problems and simply not discount weird hunches you have in the process.

The winds of change originate in the unconscious minds of domain experts. If you're sufficiently expert in a field, any weird idea or apparently irrelevant question that occurs to you is ipso facto worth exploring. [3] Within Y Combinator, when an idea is described as crazy, it's a compliment—in fact, on average probably a higher compliment than when an idea is described as good.

Startup investors have extraordinary incentives for correcting obsolete beliefs. If they can realize before other investors that some apparently unpromising startup isn't, they can make a huge amount of money. But the incentives are more than just financial. Investors' opinions are explicitly tested: startups come to them and they have to say yes or no, and then, fairly quickly, they learn whether they guessed right. The investors who say no to a Google (and there were several) will remember it for the rest of their lives.

Anyone who must in some sense bet on ideas rather than merely commenting on them has similar incentives. Which means anyone who wants such incentives can have them, by turning their comments into bets: if you write about a topic in some fairly durable and public form, you'll find you worry much more about getting things right than most people would in a casual conversation. [4]

Another trick I've found to protect myself against obsolete beliefs is to focus initially on people rather than ideas. Though the nature of future discoveries is hard to predict, I've found I can predict quite well what sort of people will make them. Good new ideas come from earnest, energetic, independent-minded people.

Betting on people over ideas saved me countless times as an investor. We thought Airbnb was a bad idea, for example. But we could tell the founders were earnest, energetic, and independent-minded. (Indeed, almost pathologically so.) So we suspended disbelief and funded them.

This too seems a technique that should be generally applicable. Surround yourself with the sort of people new ideas come from. If you want to notice quickly when your beliefs become obsolete, you can't do better than to be friends with the people whose discoveries will make them so.

It's hard enough already not to become the prisoner of your own expertise, but it will only get harder, because change is accelerating. That's not a recent trend; change has been accelerating since the paleolithic era. Ideas beget ideas. I don't expect that to change. But I could be wrong.

## Notes

[1] My usual trick is to talk about aspects of the present that most people haven't noticed yet.

[2] Especially if they become well enough known that people start to identify them with you. You have to be extra skeptical about things you want to believe, and once a hypothesis starts to be identified with you, it will almost certainly start to be in that category.

[3] In practice "sufficiently expert" doesn't require one to be recognized as an expert—which is a trailing indicator in any case. In many fields a year of focused work plus caring a lot would be enough.

[4] Though they are public and persist indefinitely, comments on e.g. forums and places like Twitter seem empirically to work like casual conversation. The threshold may be whether what you write has a title.

**Thanks** to Sam Altman, Patrick Collison, and Robert Morris for reading drafts of this.

[Spanish Translation](#)

[Arabic Translation](#)



# Let the Other 95% of Great Programmers In

December 2014

American technology companies want the government to make immigration easier because they say they can't find enough programmers in the US. Anti-immigration people say that instead of letting foreigners take these jobs, we should train more Americans to be programmers. Who's right?

The technology companies are right. What the anti-immigration people don't understand is that there is a huge variation in ability between competent programmers and exceptional ones, and while you can train people to be competent, you can't train them to be exceptional. Exceptional programmers have an aptitude for and [interest in](#) programming that is not merely the product of training. [[1](#)]

The US has less than 5% of the world's population. Which means if the qualities that make someone a great programmer are evenly distributed, 95% of great programmers are born outside the US.

The anti-immigration people have to invent some explanation to account for all the effort technology companies have expended trying to make immigration easier. So they claim it's because they want to drive down salaries. But if you talk to startups, you find practically every one over a certain size has gone through legal contortions to get programmers into the US, where they then paid them the same as they'd have paid an American. Why would they go to extra trouble to get programmers for the same price? The only explanation is that they're telling the truth: there are just not enough great programmers to go around. [[2](#)]

I asked the CEO of a startup with about 70 programmers how many more he'd hire if he could get all the great programmers he wanted. He said "We'd hire 30 tomorrow morning." And this is one of the hot startups that always win recruiting battles. It's the same all over Silicon Valley. Startups are that constrained for talent.

It would be great if more Americans were trained as programmers, but no amount of training can flip a ratio as overwhelming as 95 to 5. Especially since programmers are being trained in other countries too. Barring some cataclysm, it

will always be true that most great programmers are born outside the US. It will always be true that most people who are great at anything are born outside the US. [3]

Exceptional performance implies immigration. A country with only a few percent of the world's population will be exceptional in some field only if there are a lot of immigrants working in it.

But this whole discussion has taken something for granted: that if we let more great programmers into the US, they'll want to come. That's true now, and we don't realize how lucky we are that it is. If we want to keep this option open, the best way to do it is to take advantage of it: the more of the world's great programmers are here, the more the rest will want to come here.

And if we don't, the US could be seriously fucked. I realize that's strong language, but the people dithering about this don't seem to realize the power of the forces at work here. Technology gives the best programmers huge leverage. The world market in programmers seems to be becoming dramatically more liquid. And since good people like good colleagues, that means the best programmers could collect in just a few hubs. Maybe mostly in one hub.

What if most of the great programmers collected in one hub, and it wasn't here? That scenario may seem unlikely now, but it won't be if things change as much in the next 50 years as they did in the last 50.

We have the potential to ensure that the US remains a technology superpower just by letting in a few thousand great programmers a year. What a colossal mistake it would be to let that opportunity slip. It could easily be the defining mistake this generation of American politicians later become famous for. And unlike other potential mistakes on that scale, it costs nothing to fix.

So please, get on with it.

## Notes

[1] How much better is a great programmer than an ordinary one? So much better that you can't even measure the difference directly. A great programmer doesn't merely do the same work faster. A great programmer will invent things an ordinary programmer would never even think of. This doesn't mean a great programmer is infinitely more valuable, because any invention has a finite market value. But it's easy to imagine cases where a great programmer might invent things worth 100x or even 1000x an average programmer's salary.

[2] There are a handful of consulting firms that rent out big pools of foreign programmers they bring in on H1-B visas. By all means crack down on these. It should be easy to write legislation that distinguishes them, because they are so different from technology companies. But it is dishonest of the anti-immigration people to claim that companies like Google and Facebook are driven by the same motives. An influx of inexpensive but mediocre programmers is the last thing they'd want; it would destroy them.

[3] Though this essay talks about programmers, the group of people we need to import is broader, ranging from designers to programmers to electrical engineers. The best one could do as a general term might be "digital talent." It seemed better to make the argument a little too narrow than to confuse everyone with a neologism.

**Thanks** to Sam Altman, John Collison, Patrick Collison, Jessica Livingston, Geoff Ralston, Fred Wilson, and Qasar Younis for reading drafts of this.

[Spanish Translation](#)

# Don't Talk to Corp Dev

January 2015

Corporate Development, aka corp dev, is the group within companies that buys other companies. If you're talking to someone from corp dev, that's why, whether you realize it yet or not.

It's usually a mistake to talk to corp dev unless (a) you want to sell your company right now and (b) you're sufficiently likely to get an offer at an acceptable price. In practice that means startups should only talk to corp dev when they're either doing really well or really badly. If you're doing really badly, meaning the company is about to die, you may as well talk to them, because you have nothing to lose. And if you're doing really well, you can safely talk to them, because you both know the price will have to be high, and if they show the slightest sign of wasting your time, you'll be confident enough to tell them to get lost.

The danger is to companies in the middle. Particularly to young companies that are growing fast, but haven't been doing it for long enough to have grown big yet. It's usually a mistake for a promising company less than a year old even to talk to corp dev.

But it's a mistake founders constantly make. When someone from corp dev wants to meet, the founders tell themselves they should at least find out what they want. Besides, they don't want to offend Big Company by refusing to meet.

Well, I'll tell you what they want. They want to talk about buying you. That's what the title "corp dev" means. So before agreeing to meet with someone from corp dev, ask yourselves, "Do we want to sell the company right now?" And if the answer is no, tell them "Sorry, but we're focusing on growing the company." They won't be offended. And certainly the founders of Big Company won't be offended. If anything they'll think more highly of you. You'll remind them of themselves. They didn't sell either; that's why they're in a position now to buy other companies. [\[1\]](#)

Most founders who get contacted by corp dev already know what it means. And yet even when they know what corp dev does and know they don't want to sell, they take the meeting. Why do they do it? The same mix of denial and wishful thinking that underlies most mistakes founders make. It's flattering to talk to someone who wants to buy you. And who knows, maybe their offer will be surprisingly high. You

should at least see what it is, right?

No. If they were going to send you an offer immediately by email, sure, you might as well open it. But that is not how conversations with corp dev work. If you get an offer at all, it will be at the end of a long and unbelievably distracting process. And if the offer is surprising, it will be surprisingly low.

Distractions are the thing you can least afford in a startup. And conversations with corp dev are the worst sort of distraction, because as well as consuming your [attention](#) they undermine your morale. One of the tricks to surviving a grueling process is not to stop and think how tired you are. Instead you get into a sort of flow. [2] Imagine what it would do to you if at mile 20 of a marathon, someone ran up beside you and said "You must feel really tired. Would you like to stop and take a rest?" Conversations with corp dev are like that but worse, because the suggestion of stopping gets combined in your mind with the imaginary high price you think they'll offer.

And then you're really in trouble. If they can, corp dev people like to turn the tables on you. They like to get you to the point where you're trying to convince them to buy instead of them trying to convince you to sell. And surprisingly often they succeed.

This is a very slippery slope, greased with some of the most powerful forces that can work on founders' minds, and attended by an experienced professional whose full time job is to push you down it.

Their tactics in pushing you down that slope are usually fairly brutal. Corp dev people's whole job is to buy companies, and they don't even get to choose which. The only way their performance is measured is by how cheaply they can buy you, and the more ambitious ones will stop at nothing to achieve that. For example, they'll almost always start with a lowball offer, just to see if you'll take it. Even if you don't, a low initial offer will demoralize you and make you easier to manipulate.

And that is the most innocent of their tactics. Just wait till you've agreed on a price and think you have a done deal, and then they come back and say their boss has vetoed the deal and won't do it for more than half the agreed upon price. Happens all the time. If you think investors can behave badly, it's nothing compared to what corp dev people can do. Even corp dev people at companies that are otherwise benevolent.

I remember once complaining to a friend at Google about some nasty trick their corp dev people had pulled on a YC startup.

"What happened to Don't be Evil?" I asked.

"I don't think corp dev got the memo," he replied.

The tactics you encounter in M&A conversations can be like nothing you've experienced in the otherwise comparatively [upstanding](#) world of Silicon Valley. It's as if a chunk of genetic material from the old-fashioned robber baron business world got incorporated into the startup world. [3]

The simplest way to protect yourself is to use the trick that John D. Rockefeller, whose grandfather was an alcoholic, used to protect himself from becoming one. He once told a Sunday school class

Boys, do you know why I never became a drunkard? Because I never took the first drink.

Do you want to sell your company right now? Not eventually, right now. If not, just don't take the first meeting. They won't be offended. And you in turn will be guaranteed to be spared one of the worst experiences that can happen to a startup.

If you do want to sell, there's another set of [techniques](#) for doing that. But the biggest mistake founders make in dealing with corp dev is not doing a bad job of talking to them when they're ready to, but talking to them before they are. So if you remember only the title of this essay, you already know most of what you need to know about M&A in the first year.

## Notes

[1] I'm not saying you should never sell. I'm saying you should be clear in your own mind about whether you want to sell or not, and not be led by manipulation or wishful thinking into trying to sell earlier than you otherwise would have.

[2] In a startup, as in most competitive sports, the task at hand almost does this for you; you're too busy to feel tired. But when you lose that protection, e.g. at the final whistle, the fatigue hits you like a wave. To talk to corp dev is to let yourself feel it mid-game.

[3] To be fair, the apparent misdeeds of corp dev people are magnified by the fact that they function as the face of a large organization that often doesn't know its own mind. Acquirers can be surprisingly indecisive about acquisitions, and their flakiness is indistinguishable from dishonesty by the time it filters down to you.

**Thanks** to Marc Andreessen, Jessica Livingston, Geoff Ralston, and Qasar Younis

for reading drafts of this.

# What Doesn't Seem Like Work?

January 2015

My father is a mathematician. For most of my childhood he worked for Westinghouse, modelling nuclear reactors.

He was one of those lucky people who know early on what they want to do. When you talk to him about his childhood, there's a clear watershed at about age 12, when he "got interested in maths."

He grew up in the small Welsh seacoast town of [Pwllheli](#). As we retraced his walk to school on Google Street View, he said that it had been nice growing up in the country.

"Didn't it get boring when you got to be about 15?" I asked.

"No," he said, "by then I was interested in maths."

In another conversation he told me that what he really liked was solving problems. To me the exercises at the end of each chapter in a math textbook represent work, or at best a way to reinforce what you learned in that chapter. To him the problems were the reward. The text of each chapter was just some advice about solving them. He said that as soon as he got a new textbook he'd immediately work out all the problems — to the slight annoyance of his teacher, since the class was supposed to work through the book gradually.

Few people know so early or so certainly what they want to work on. But talking to my father reminded me of a heuristic the rest of us can use. If something that seems like work to other people doesn't seem like work to you, that's something you're well suited for. For example, a lot of programmers I know, including me, actually like debugging. It's not something people tend to volunteer; one likes it the way one likes popping zits. But you may have to like debugging to like programming, considering the degree to which programming consists of it.

The stranger your tastes seem to other people, the stronger evidence they probably are of what you should do. When I was in college I used to write papers for my friends. It was quite interesting to write a paper for a class I wasn't taking. Plus they were always so relieved.



It seemed curious that the same task could be painful to one person and pleasant to another, but I didn't realize at the time what this imbalance implied, because I wasn't looking for it. I didn't realize how hard it can be to decide what you should work on, and that you sometimes have to [figure it out](#) from subtle clues, like a detective solving a case in a mystery novel. So I bet it would help a lot of people to ask themselves about this explicitly. What seems like work to other people that doesn't seem like work to you?

**Thanks** to Sam Altman, Trevor Blackwell, Jessica Livingston, Robert Morris, and my father for reading drafts of this.

[Robert Morris: All About Programming.](#)

[French Translation](#)

# The Ronco Principle

January 2015

No one, VC or angel, has invested in more of the top startups than Ron Conway. He knows what happened in every deal in the Valley, half the time because he arranged it.

And yet he's a super nice guy. In fact, nice is not the word. Ronco is good. I know of zero instances in which he has behaved badly. It's hard even to imagine.

When I first came to Silicon Valley I thought "How lucky that someone so powerful is so benevolent." But gradually I realized it wasn't luck. It was by being benevolent that Ronco became so powerful. All the deals he gets to invest in come to him through referrals. Google did. Facebook did. Twitter was a referral from Evan Williams himself. And the reason so many people refer deals to him is that he's proven himself to be a good guy.

Good does not mean being a pushover. I would not want to face an angry Ronco. But if Ron's angry at you, it's because you did something wrong. Ron is so old school he's Old Testament. He will smite you in his just wrath, but there's no malice in it.

In almost every domain there are advantages to seeming good. It makes people trust you. But actually being good is an expensive way to seem good. To an amoral person it might seem to be overkill.

In some fields it might be, but apparently not in the startup world. Though plenty of investors are jerks, there is a clear trend among them: the most successful investors are also the most upstanding. [\[1\]](#)

It was not always this way. I would not feel confident saying that about investors twenty years ago.

What changed? The startup world became more transparent and more unpredictable. Both make it harder to seem good without actually being good.

It's obvious why transparency has that effect. When an investor maltreats a founder now, it gets out. Maybe not all the way to the press, but other founders hear about it, and that investor starts to lose deals. [\[2\]](#)

The effect of unpredictability is more subtle. It increases the work of being inconsistent. If you're going to be two-faced, you have to know who you should be nice to and who you can get away with being nasty to. In the startup world, things change so rapidly that you can't tell. The random college kid you talk to today might in a couple years be the CEO of the hottest startup in the Valley. If you can't tell who to be nice to, you have to be nice to everyone. And probably the only people who can manage that are the people who are genuinely good.

In a sufficiently connected and unpredictable world, you can't seem good without being good.

As often happens, Ron discovered how to be the investor of the future by accident. He didn't foresee the future of startup investing, realize it would pay to be upstanding, and force himself to behave that way. It would feel unnatural to him to behave any other way. He was already [living in the future](#).

Fortunately that future is not limited to the startup world. The startup world is more transparent and unpredictable than most, but almost everywhere the trend is in that direction.

## Notes

[1] I'm not saying that if you sort investors by benevolence you've also sorted them by returns, but rather that if you do a scatterplot with benevolence on the x axis and returns on the y, you'd see a clear upward trend.

[2] Y Combinator in particular, because it aggregates data from so many startups, has a pretty comprehensive view of investor behavior.

**Thanks** to Sam Altman and Jessica Livingston for reading drafts of this.

[Japanese Translation](#)

# What Microsoft Is this the Altair Basic of?

February 2015

One of the most valuable exercises you can try if you want to understand startups is to look at the most successful companies and explain why they were not as lame as they seemed when they first launched. Because they practically all seemed lame at first. Not just small, lame. Not just the first step up a big mountain. More like the first step into a swamp.

A Basic interpreter for the Altair? How could that ever grow into a giant company? People sleeping on airbeds in strangers' apartments? A web site for college students to stalk one another? A wimpy little single-board computer for hobbyists that used a TV as a monitor? A new search engine, when there were already about 10, and they were all trying to de-emphasize search? These ideas didn't just seem small. They seemed wrong. They were the kind of ideas you could not merely ignore, but ridicule.

Often the founders themselves didn't know why their ideas were promising. They were attracted to these ideas by instinct, because they were [living in the future](#) and they sensed that something was missing. But they could not have put into words exactly how their ugly ducklings were going to grow into big, beautiful swans.

Most people's first impulse when they hear about a lame-sounding new startup idea is to make fun of it. Even a lot of people who should know better.

When I encounter a startup with a lame-sounding idea, I ask "What Microsoft is this the Altair Basic of?" Now it's a puzzle, and the burden is on me to solve it. Sometimes I can't think of an answer, especially when the idea is a made-up one. But it's remarkable how often there does turn out to be an answer. Often it's one the founders themselves hadn't seen yet.

Intriguingly, there are sometimes multiple answers. I talked to a startup a few days ago that could grow into 3 distinct Microsofts. They'd probably vary in size by orders of magnitude. But you can never predict how big a Microsoft is going to be, so in cases like that I encourage founders to follow whichever path is most

immediately exciting to them. Their instincts got them this far. Why stop now?

# Change Your Name

August 2015

If you have a US startup called X and you don't have x.com, you should probably change your name.

The reason is not just that people can't find you. For companies with mobile apps, especially, having the right domain name is not as critical as it used to be for getting users. The problem with not having the .com of your name is that it signals weakness. Unless you're so big that your reputation precedes you, a marginal domain suggests you're a marginal company. Whereas (as Stripe shows) having x.com signals strength even if it has no relation to what you do.

Even good founders can be in denial about this. Their denial derives from two very powerful forces: identity, and lack of imagination.

X is what we *are*, founders think. There's no other name as good. Both of which are false.

You can fix the first by stepping back from the problem. Imagine you'd called your company something else. If you had, surely you'd be just as attached to that name as you are to your current one. The idea of switching to your current name would seem repellent. [\[1\]](#)

There's nothing intrinsically great about your current name. Nearly all your attachment to it comes from it being attached to you. [\[2\]](#)

The way to neutralize the second source of denial, your inability to think of other potential names, is to acknowledge that you're bad at naming. Naming is a completely separate skill from those you need to be a good founder. You can be a great startup founder but hopeless at thinking of names for your company.

Once you acknowledge that, you stop believing there is nothing else you could be called. There are lots of other potential names that are as good or better; you just can't think of them.

How do you find them? One answer is the default way to solve problems you're bad at: find someone else who can think of names. But with company names there is another possible approach. It turns out almost any word or word pair that is not

an obviously bad name is a sufficiently good one, and the number of such domains is so large that you can find plenty that are cheap or even untaken. So make a list and try to buy some. That's what [Stripe](#) did. (Their search also turned up [parse.com](#), which their friends at Parse took.)

The reason I know that naming companies is a distinct skill orthogonal to the others you need in a startup is that I happen to have it. Back when I was running YC and did more office hours with startups, I would often help them find new names. 80% of the time we could find at least one good name in a 20 minute office hour slot.

Now when I do office hours I have to focus on more important questions, like what the company is doing. I tell them when they need to change their name. But I know the power of the forces that have them in their grip, so I know most won't listen. [3]

There are of course examples of startups that have succeeded without having the .com of their name. There are startups that have succeeded despite any number of different mistakes. But this mistake is less excusable than most. It's something that can be fixed in a couple days if you have sufficient discipline to acknowledge the problem.

100% of the top 20 YC companies by valuation have the .com of their name. 94% of the top 50 do. But only 66% of companies in the current batch have the .com of their name. Which suggests there are lessons ahead for most of the rest, one way or another.

## Notes

[1] Incidentally, this thought experiment works for [nationality and religion](#) too.

[2] The liking you have for a name that has become part of your identity manifests itself not directly, which would be easy to discount, but as a collection of specious beliefs about its intrinsic qualities. (This too is true of nationality and religion as well.)

[3] Sometimes founders know it's a problem that they don't have the .com of their name, but delusion strikes a step later in the belief that they'll be able to buy it despite having no evidence it's for sale. Don't believe a domain is for sale unless the owner has already told you an asking price.

**Thanks** to Sam Altman, Jessica Livingston, and Geoff Ralston for reading drafts of this.





# Why It's Safe for Founders to Be Nice

August 2015

I recently got an email from a founder that helped me understand something important: why it's safe for startup founders to be nice people.

I grew up with a cartoon idea of a very successful businessman (in the cartoon it was always a man): a rapacious, cigar-smoking, table-thumping guy in his fifties who wins by exercising power, and isn't too fussy about how. As I've written before, one of the things that has surprised me most about startups is [how few](#) of the most successful founders are like that. Maybe successful people in other industries are; I don't know; but not startup founders. [\[1\]](#)

I knew this empirically, but I never saw the math of why till I got this founder's email. In it he said he worried that he was fundamentally soft-hearted and tended to give away too much for free. He thought perhaps he needed "a little dose of sociopath-ness."

I told him not to worry about it, because so long as he built something good enough to spread by word of mouth, he'd have a superlinear growth curve. If he was bad at extracting money from people, at worst this curve would be some constant multiple less than 1 of what it might have been. But a constant multiple of any curve is exactly the same shape. The numbers on the Y axis are smaller, but the curve is just as steep, and when anything grows at the rate of a successful startup, the Y axis will take care of itself.

Some examples will make this clear. Suppose your company is making \$1000 a month now, and you've made something so great that it's growing at 5% a week. Two years from now, you'll be making about \$160k a month.

Now suppose you're so un-rapacious that you only extract half as much from your users as you could. That means two years later you'll be making \$80k a month instead of \$160k. How far behind are you? How long will it take to catch up with where you'd have been if you were extracting every penny? A mere 15 weeks. After two years, the un-rapacious founder is only 3.5 months behind the rapacious one. [\[2\]](#)

If you're going to optimize a number, the one to choose is your [growth rate](#). Suppose as before that you only extract half as much from users as you could, but

that you're able to grow 6% a week instead of 5%. Now how are you doing compared to the rapacious founder after two years? You're already ahead—\$214k a month versus \$160k—and pulling away fast. In another year you'll be making \$4.4 million a month to the rapacious founder's \$2 million.

Obviously one case where it would help to be rapacious is when growth depends on that. What makes startups different is that usually it doesn't. Startups usually win by making something so great that people recommend it to their friends. And being rapacious not only doesn't help you do that, but probably hurts. [3]

The reason startup founders can safely be nice is that making great things is compounded, and rapacity isn't.

So if you're a founder, here's a deal you can make with yourself that will both make you happy and make your company successful. Tell yourself you can be as nice as you want, so long as you work hard on your growth rate to compensate. Most successful startups make that tradeoff unconsciously. Maybe if you do it consciously you'll do it even better.

## Notes

[1] Many think successful startup founders are driven by money. In fact the secret weapon of the most successful founders is that they aren't. If they were, they'd have taken one of the acquisition offers that every fast-growing startup gets on the way up. What drives the most successful founders is the same thing that drives most people who make things: the company is their project.

[2] In fact since  $2 \approx 1.05^{15}$ , the un-rapacious founder is always 15 weeks behind the rapacious one.

[3] The other reason it might help to be good at squeezing money out of customers is that startups usually lose money at first, and making more per customer makes it easier to get to profitability before your initial funding runs out. But while it is very common for startups to [die](#) from running through their initial funding and then being unable to raise more, the underlying cause is usually slow growth or excessive spending rather than insufficient effort to extract money from existing customers.

**Thanks** to Sam Altman, Harj Taggar, Jessica Livingston, and Geoff Ralston for

reading drafts of this, and to Randall Bennett for being such a nice guy.

# Default Alive or Default Dead?

October 2015

When I talk to a startup that's been operating for more than 8 or 9 months, the first thing I want to know is almost always the same. Assuming their expenses remain constant and their revenue growth is what it has been over the last several months, do they make it to profitability on the money they have left? Or to put it more dramatically, by default do they live or die?

The startling thing is how often the founders themselves don't know. Half the founders I talk to don't know whether they're default alive or default dead.

If you're among that number, Trevor Blackwell has made a handy [calculator](#) you can use to find out.

The reason I want to know first whether a startup is default alive or default dead is that the rest of the conversation depends on the answer. If the company is default alive, we can talk about ambitious new things they could do. If it's default dead, we probably need to talk about how to save it. We know the current trajectory ends badly. How can they get off that trajectory?

Why do so few founders know whether they're default alive or default dead? Mainly, I think, because they're not used to asking that. It's not a question that makes sense to ask early on, any more than it makes sense to ask a 3 year old how he plans to support himself. But as the company grows older, the question switches from meaningless to critical. That kind of switch often takes people by surprise.

I propose the following solution: instead of starting to ask too late whether you're default alive or default dead, start asking too early. It's hard to say precisely when the question switches polarity. But it's probably not that dangerous to start worrying too early that you're default dead, whereas it's very dangerous to start worrying too late.

The reason is a phenomenon I wrote about earlier: the [fatal pinch](#). The fatal pinch is default dead + slow growth + not enough time to fix it. And the way founders end up in it is by not realizing that's where they're headed.

There is another reason founders don't ask themselves whether they're default

alive or default dead: they assume it will be easy to raise more money. But that assumption is often false, and worse still, the more you depend on it, the falser it becomes.

Maybe it will help to separate facts from hopes. Instead of thinking of the future with vague optimism, explicitly separate the components. Say "We're default dead, but we're counting on investors to save us." Maybe as you say that, it will set off the same alarms in your head that it does in mine. And if you set off the alarms sufficiently early, you may be able to avoid the fatal pinch.

It would be safe to be default dead if you could count on investors saving you. As a rule their interest is a function of growth. If you have steep revenue growth, say over 5x a year, you can start to count on investors being interested even if you're not profitable. [1] But investors are so fickle that you can never do more than start to count on them. Sometimes something about your business will spook investors even if your growth is great. So no matter how good your growth is, you can never safely treat fundraising as more than a plan A. You should always have a plan B as well: you should know (as in write down) precisely what you'll need to do to survive if you can't raise more money, and precisely when you'll have to switch to plan B if plan A isn't working.

In any case, growing fast versus operating cheaply is far from the sharp dichotomy many founders assume it to be. In practice there is surprisingly little connection between how much a startup spends and how fast it grows. When a startup grows fast, it's usually because the product hits a nerve, in the sense of hitting some big need straight on. When a startup spends a lot, it's usually because the product is expensive to develop or sell, or simply because they're wasteful.

If you're paying attention, you'll be asking at this point not just how to avoid the fatal pinch, but how to avoid being default dead. That one is easy: don't hire too fast. Hiring too fast is by far the biggest killer of startups that raise money. [2]

Founders tell themselves they need to hire in order to grow. But most err on the side of overestimating this need rather than underestimating it. Why? Partly because there's so much work to do. Naive founders think that if they can just hire enough people, it will all get done. Partly because successful startups have lots of employees, so it seems like that's what one does in order to be successful. In fact the large staffs of successful startups are probably more the effect of growth than the cause. And partly because when founders have slow growth they don't want to face what is usually the real reason: the product is not appealing enough.

Plus founders who've just raised money are often encouraged to overhire by the VCs who funded them. Kill-or-cure strategies are optimal for VCs because they're protected by the portfolio effect. VCs want to blow you up, in one sense of the phrase or the other. But as a founder your incentives are different. You want above all to survive. [3]

Here's a common way startups die. They make something moderately appealing

and have decent initial growth. They raise their first round fairly easily, because the founders seem smart and the idea sounds plausible. But because the product is only moderately appealing, growth is ok but not great. The founders convince themselves that hiring a bunch of people is the way to boost growth. Their investors agree. But (because the product is only moderately appealing) the growth never comes. Now they're rapidly running out of runway. They hope further investment will save them. But because they have high expenses and slow growth, they're now unappealing to investors. They're unable to raise more, and the company dies.

What the company should have done is address the fundamental problem: that the product is only moderately appealing. Hiring people is rarely the way to fix that. More often than not it makes it harder. At this early stage, the product needs to evolve more than to be "built out," and that's usually easier with fewer people. [4]

Asking whether you're default alive or default dead may save you from this. Maybe the alarm bells it sets off will counteract the forces that push you to overhire. Instead you'll be compelled to seek growth in other ways. For example, by [doing things that don't scale](#), or by redesigning the product in the way only founders can. And for many if not most startups, these paths to growth will be the ones that actually work.

Airbnb waited 4 months after raising money at the end of Y Combinator before they hired their first employee. In the meantime the founders were terribly overworked. But they were overworked evolving Airbnb into the astonishingly successful organism it is now.

## Notes

[1] Steep usage growth will also interest investors. Revenue will ultimately be a constant multiple of usage, so  $x\%$  usage growth predicts  $x\%$  revenue growth. But in practice investors discount merely predicted revenue, so if you're measuring usage you need a higher growth rate to impress investors.

[2] Startups that don't raise money are saved from hiring too fast because they can't afford to. But that doesn't mean you should avoid raising money in order to avoid this problem, any more than that total abstinence is the only way to avoid becoming an alcoholic.

[3] I would not be surprised if VCs' tendency to push founders to overhire is not even in their own interest. They don't know how many of the companies that get killed by overspending might have done well if they'd survived. My guess is a significant number.

[4] After reading a draft, Sam Altman wrote:

"I think you should make the hiring point more strongly. I think it's roughly correct to say that YC's most successful companies have never been the fastest to hire, and one of the marks of a great founder is being able to resist this urge."

Paul Buchheit adds:

"A related problem that I see a lot is premature scaling—founders take a small business that isn't really working (bad unit economics, typically) and then scale it up because they want impressive growth numbers. This is similar to over-hiring in that it makes the business much harder to fix once it's big, plus they are bleeding cash really fast."

**Thanks** to Sam Altman, Paul Buchheit, Joe Gebbia, Jessica Livingston, and Geoff Ralston for reading drafts of this.

# Write Like You Talk

October 2015

Here's a simple trick for getting more people to read what you write: write in spoken language.

Something comes over most people when they start writing. They write in a different language than they'd use if they were talking to a friend. The sentence structure and even the words are different. No one uses "pen" as a verb in spoken English. You'd feel like an idiot using "pen" instead of "write" in a conversation with a friend.

The last straw for me was a sentence I read a couple days ago:

The mercurial Spaniard himself declared: "After Altamira, all is decadence."

It's from Neil Oliver's *A History of Ancient Britain*. I feel bad making an example of this book, because it's no worse than lots of others. But just imagine calling Picasso "the mercurial Spaniard" when talking to a friend. Even one sentence of this would raise eyebrows in conversation. And yet people write whole books of it.

Ok, so written and spoken language are different. Does that make written language worse?

If you want people to read and understand what you write, yes. Written language is more complex, which makes it more work to read. It's also more formal and distant, which gives the reader's attention permission to drift. But perhaps worst of all, the complex sentences and fancy words give you, the writer, the false impression that you're saying more than you actually are.

You don't need complex sentences to express complex ideas. When specialists in some abstruse topic talk to one another about ideas in their field, they don't use sentences any more complex than they do when talking about what to have for lunch. They use different words, certainly. But even those they use no more than necessary. And in my experience, the harder the subject, the more informally experts speak. Partly, I think, because they have less to prove, and partly because the harder the ideas you're talking about, the less you can afford to let language get in the way.



Informal language is the athletic clothing of ideas.

I'm not saying spoken language always works best. Poetry is as much music as text, so you can say things you wouldn't say in conversation. And there are a handful of writers who can get away with using fancy language in prose. And then of course there are cases where writers don't want to make it easy to understand what they're saying—in corporate announcements of bad news, for example, or at the more [bogus](#) end of the humanities. But for nearly everyone else, spoken language is better.

It seems to be hard for most people to write in spoken language. So perhaps the best solution is to write your first draft the way you usually would, then afterward look at each sentence and ask "Is this the way I'd say this if I were talking to a friend?" If it isn't, imagine what you would say, and use that instead. After a while this filter will start to operate as you write. When you write something you wouldn't say, you'll hear the clank as it hits the page.

Before I publish a new essay, I read it out loud and fix everything that doesn't sound like conversation. I even fix bits that are phonetically awkward; I don't know if that's necessary, but it doesn't cost much.

This trick may not always be enough. I've seen writing so far removed from spoken language that it couldn't be fixed sentence by sentence. For cases like that there's a more drastic solution. After writing the first draft, try explaining to a friend what you just wrote. Then replace the draft with what you said to your friend.

People often tell me how much my essays sound like me talking. The fact that this seems worthy of comment shows how rarely people manage to write in spoken language. Otherwise everyone's writing would sound like them talking.

If you simply manage to write in spoken language, you'll be ahead of 95% of writers. And it's so easy to do: just don't let a sentence through unless it's the way you'd say it to a friend.

**Thanks** to Patrick Collison and Jessica Livingston for reading drafts of this.

[Japanese Translation](#)

[Arabic Translation](#)



# A Way to Detect Bias

October 2015

This will come as a surprise to a lot of people, but in some cases it's possible to detect bias in a selection process without knowing anything about the applicant pool. Which is exciting because among other things it means third parties can use this technique to detect bias whether those doing the selecting want them to or not.

You can use this technique whenever (a) you have at least a random sample of the applicants that were selected, (b) their subsequent performance is measured, and (c) the groups of applicants you're comparing have roughly equal distribution of ability.

How does it work? Think about what it means to be biased. What it means for a selection process to be biased against applicants of type x is that it's harder for them to make it through. Which means applicants of type x have to be better to get selected than applicants not of type x. [1] Which means applicants of type x who do make it through the selection process will outperform other successful applicants. And if the performance of all the successful applicants is measured, you'll know if they do.

Of course, the test you use to measure performance must be a valid one. And in particular it must not be invalidated by the bias you're trying to measure. But there are some domains where performance can be measured, and in those detecting bias is straightforward. Want to know if the selection process was biased against some type of applicant? Check whether they outperform the others. This is not just a heuristic for detecting bias. It's what bias means.

For example, many suspect that venture capital firms are biased against female founders. This would be easy to detect: among their portfolio companies, do startups with female founders outperform those without? A couple months ago, one VC firm (almost certainly unintentionally) published a study showing bias of this type. First Round Capital found that among its portfolio companies, startups with female founders [outperformed](#) those without by 63%. [2]

The reason I began by saying that this technique would come as a surprise to many people is that we so rarely see analyses of this type. I'm sure it will come as a surprise to First Round that they performed one. I doubt anyone there realized

that by limiting their sample to their own portfolio, they were producing a study not of startup trends but of their own biases when selecting companies.

I predict we'll see this technique used more in the future. The information needed to conduct such studies is increasingly available. Data about who applies for things is usually closely guarded by the organizations selecting them, but nowadays data about who gets selected is often publicly available to anyone who takes the trouble to aggregate it.

## Notes

[1] This technique wouldn't work if the selection process looked for different things from different types of applicants—for example, if an employer hired men based on their ability but women based on their appearance.

[2] As Paul Buchheit points out, First Round excluded their most successful investment, Uber, from the study. And while it makes sense to exclude outliers from some types of studies, studies of returns from startup investing, which is all about hitting outliers, are not one of them.

**Thanks** to Sam Altman, Jessica Livingston, and Geoff Ralston for reading drafts of this.

[Arabic Translation](#)

[Swedish Translation](#)

# Jessica Livingston

November 2015

A few months ago an article about Y Combinator said that early on it had been a "one-man show." It's sadly common to read that sort of thing. But the problem with that description is not just that it's unfair. It's also misleading. Much of what's most novel about YC is due to Jessica Livingston. If you don't understand her, you don't understand YC. So let me tell you a little about Jessica.

YC had 4 founders. Jessica and I decided one night to start it, and the next day we recruited my friends Robert Morris and Trevor Blackwell. Jessica and I ran YC day to day, and Robert and Trevor read applications and did interviews with us.

Jessica and I were already dating when we started YC. At first we tried to act "professional" about this, meaning we tried to conceal it. In retrospect that seems ridiculous, and we soon dropped the pretense. And the fact that Jessica and I were a couple is a big part of what made YC what it was. YC felt like a family. The founders early on were mostly young. We all had dinner together once a week, cooked for the first couple years by me. Our first building had been a private home. The overall atmosphere was shockingly different from a VC's office on Sand Hill Road, in a way that was entirely for the better. There was an authenticity that everyone who walked in could sense. And that didn't just mean that people trusted us. It was the perfect quality to instill in startups. Authenticity is one of the most important things YC looks for in founders, not just because fakers and opportunists are annoying, but because authenticity is one of the main things that separates the most successful startups from the rest.

Early YC was a family, and Jessica was its mom. And the culture she defined was one of YC's most important innovations. Culture is important in any organization, but at YC culture wasn't just how we behaved when we built the product. At YC, the culture was the product.

Jessica was also the mom in another sense: she had the last word. Everything we did as an organization went through her first — who to fund, what to say to the public, how to deal with other companies, who to hire, everything.

Before we had kids, YC was more or less our life. There was no real distinction between working hours and not. We talked about YC all the time. And while there might be some businesses that it would be tedious to let infect your private life, we

liked it. We'd started YC because it was something we were interested in. And some of the problems we were trying to solve were endlessly difficult. How do you recognize good founders? You could talk about that for years, and we did; we still do.

I'm better at some things than Jessica, and she's better at some things than me. One of the things she's best at is judging people. She's one of those rare individuals with x-ray vision for character. She can see through any kind of faker almost immediately. Her nickname within YC was the Social Radar, and this special power of hers was critical in making YC what it is. The earlier you pick startups, the more you're picking the founders. Later stage investors get to try products and look at growth numbers. At the stage where YC invests, there is often neither a product nor any numbers.

Others thought YC had some special insight about the future of technology. Mostly we had the same sort of insight Socrates claimed: we at least knew we knew nothing. What made YC successful was being able to pick good founders. We thought Airbnb was a bad idea. We funded it because we liked the founders.

During interviews, Robert and Trevor and I would pepper the applicants with technical questions. Jessica would mostly watch. A lot of the applicants probably read her as some kind of secretary, especially early on, because she was the one who'd go out and get each new group and she didn't ask many questions. She was ok with that. It was easier for her to watch people if they didn't notice her. But after the interview, the three of us would turn to Jessica and ask "What does the Social Radar say?" [\[1\]](#)

Having the Social Radar at interviews wasn't just how we picked founders who'd be successful. It was also how we picked founders who were good people. At first we did this because we couldn't help it. Imagine what it would feel like to have x-ray vision for character. Being around bad people would be intolerable. So we'd refuse to fund founders whose characters we had doubts about even if we thought they'd be successful.

Though we initially did this out of self-indulgence, it turned out to be very valuable to YC. We didn't realize it in the beginning, but the people we were picking would become the YC alumni network. And once we picked them, unless they did something really egregious, they were going to be part of it for life. Some now think YC's alumni network is its most valuable feature. I personally think YC's advice is pretty good too, but the alumni network is certainly among the most valuable features. The level of trust and helpfulness is remarkable for a group of such size. And Jessica is the main reason why.

(As we later learned, it probably cost us little to reject people whose characters we had doubts about, because how good founders are and how well they do are [not orthogonal](#). If bad founders succeed at all, they tend to sell early. The most successful founders are almost all good.)

If Jessica was so important to YC, why don't more people realize it? Partly because I'm a writer, and writers always get disproportionate attention. YC's brand was initially my brand, and our applicants were people who'd read my essays. But there is another reason: Jessica hates attention. Talking to reporters makes her nervous. The thought of giving a talk paralyzes her. She was even uncomfortable at our wedding, because the bride is always the center of attention. [2]

It's not just because she's shy that she hates attention, but because it throws off the Social Radar. She can't be herself. You can't watch people when everyone is watching you.

Another reason attention worries her is that she hates bragging. In anything she does that's publicly visible, her biggest fear (after the obvious fear that it will be bad) is that it will seem ostentatious. She says being too modest is a common problem for women. But in her case it goes beyond that. She has a horror of ostentation so visceral it's almost a phobia.

She also hates fighting. She can't do it; she just shuts down. And unfortunately there is a good deal of fighting in being the public face of an organization.

So although Jessica more than anyone made YC unique, the very qualities that enabled her to do it mean she tends to get written out of YC's history. Everyone buys this story that PG started YC and his wife just kind of helped. Even YC's haters buy it. A couple years ago when people were attacking us for not funding more female founders (than exist), they all treated YC as identical with PG. It would have spoiled the narrative to acknowledge Jessica's central role at YC.

Jessica was boiling mad that people were accusing *her* company of sexism. I've never seen her angrier about anything. But she did not contradict them. Not publicly. In private there was a great deal of profanity. And she wrote three separate essays about the question of female founders. But she could never bring herself to publish any of them. She'd seen the level of vitriol in this debate, and she shrank from engaging. [3]

It wasn't just because she disliked fighting. She's so sensitive to character that it repels her even to fight with dishonest people. The idea of mixing it up with linkbait journalists or Twitter trolls would seem to her not merely frightening, but disgusting.

But Jessica knew her example as a successful female founder would encourage more women to start companies, so last year she did something YC had never done before and hired a PR firm to get her some interviews. At one of the first she did, the reporter brushed aside her insights about startups and turned it into a sensationalistic story about how some guy had tried to chat her up as she was waiting outside the bar where they had arranged to meet. Jessica was mortified, partly because the guy had done nothing wrong, but more because the story treated her as a victim significant only for being a woman, rather than one of the most knowledgeable investors in the Valley.

After that she told the PR firm to stop.

You're not going to be hearing in the press about what Jessica has achieved. So let me tell you what Jessica has achieved. Y Combinator is fundamentally a nexus of people, like a university. It doesn't make a product. What defines it is the people. Jessica more than anyone curated and nurtured that collection of people. In that sense she literally made YC.

Jessica knows more about the qualities of startup founders than anyone else ever has. Her immense data set and x-ray vision are the perfect storm in that respect. The qualities of the founders are the best predictor of how a startup will do. And startups are in turn the most important source of growth in mature economies.

The person who knows the most about the most important factor in the growth of mature economies — that is who Jessica Livingston is. Doesn't that sound like someone who should be better known?

## Notes

[1] Harj Taggar reminded me that while Jessica didn't ask many questions, they tended to be important ones:

"She was always good at sniffing out any red flags about the team or their determination and disarmingly asking the right question, which usually revealed more than the founders realized."

[2] Or more precisely, while she likes getting attention in the sense of getting credit for what she has done, she doesn't like getting attention in the sense of being watched in real time. Unfortunately, not just for her but for a lot of people, how much you get of the former depends a lot on how much you get of the latter.

Incidentally, if you saw Jessica at a public event, you would never guess she hates attention, because (a) she is very polite and (b) when she's nervous, she expresses it by smiling more.

[3] The existence of people like Jessica is not just something the mainstream media needs to learn to acknowledge, but something feminists need to learn to acknowledge as well. There are successful women who don't like to fight. Which means if the public conversation about women consists of fighting, their voices will be silenced.



There's a sort of Gresham's Law of conversations. If a conversation reaches a certain level of incivility, the more thoughtful people start to leave. No one understands female founders better than Jessica. But it's unlikely anyone will ever hear her speak candidly about the topic. She ventured a toe in that water a while ago, and the reaction was so violent that she decided "never again."

**Thanks** to Sam Altman, Paul Buchheit, Patrick Collison, Daniel Gackle, Carolynn Levy, Jon Levy, Kirsty Nathoo, Robert Morris, Geoff Ralston, and Harj Taggar for reading drafts of this. And yes, Jessica Livingston, who made me cut surprisingly little.

# The Refragmentation

January 2016

One advantage of being old is that you can see change happen in your lifetime. A lot of the change I've seen is fragmentation. US politics is much more polarized than it used to be. Culturally we have ever less common ground. The creative class flocks to a handful of happy cities, abandoning the rest. And increasing economic inequality means the spread between rich and poor is growing too. I'd like to propose a hypothesis: that all these trends are instances of the same phenomenon. And moreover, that the cause is not some force that's pulling us apart, but rather the erosion of forces that had been pushing us together.

Worse still, for those who worry about these trends, the forces that were pushing us together were an anomaly, a one-time combination of circumstances that's unlikely to be repeated — and indeed, that we would not want to repeat.

The two forces were war (above all World War II), and the rise of large corporations.

The effects of World War II were both economic and social. Economically, it decreased variation in income. Like all modern armed forces, America's were socialist economically. From each according to his ability, to each according to his need. More or less. Higher ranking members of the military got more (as higher ranking members of socialist societies always do), but what they got was fixed according to their rank. And the flattening effect wasn't limited to those under arms, because the US economy was conscripted too. Between 1942 and 1945 all wages were set by the National War Labor Board. Like the military, they defaulted to flatness. And this national standardization of wages was so pervasive that its effects could still be seen years after the war ended. [\[1\]](#)

Business owners weren't supposed to be making money either. FDR said "not a single war millionaire" would be permitted. To ensure that, any increase in a company's profits over prewar levels was taxed at 85%. And when what was left after corporate taxes reached individuals, it was taxed again at a marginal rate of 93%. [\[2\]](#)

Socially too the war tended to decrease variation. Over 16 million men and women from all sorts of different backgrounds were brought together in a way of life that was literally uniform. Service rates for men born in the early 1920s approached

80%. And working toward a common goal, often under stress, brought them still closer together.

Though strictly speaking World War II lasted less than 4 years for the US, its effects lasted longer. Wars make central governments more powerful, and World War II was an extreme case of this. In the US, as in all the other Allied countries, the federal government was slow to give up the new powers it had acquired. Indeed, in some respects the war didn't end in 1945; the enemy just switched to the Soviet Union. In tax rates, federal power, defense spending, conscription, and nationalism, the decades after the war looked more like wartime than prewar peacetime. [3] And the social effects lasted too. The kid pulled into the army from behind a mule team in West Virginia didn't simply go back to the farm afterward. Something else was waiting for him, something that looked a lot like the army.

If total war was the big political story of the 20th century, the big economic story was the rise of a new kind of company. And this too tended to produce both social and economic cohesion. [4]

The 20th century was the century of the big, national corporation. General Electric, General Foods, General Motors. Developments in finance, communications, transportation, and manufacturing enabled a new type of company whose goal was above all scale. Version 1 of this world was low-res: a Duplo world of a few giant companies dominating each big market. [5]

The late 19th and early 20th centuries had been a time of consolidation, led especially by J. P. Morgan. Thousands of companies run by their founders were merged into a couple hundred giant ones run by professional managers. Economies of scale ruled the day. It seemed to people at the time that this was the final state of things. John D. Rockefeller said in 1880

The day of combination is here to stay. Individualism has gone, never to return.

He turned out to be mistaken, but he seemed right for the next hundred years.

The consolidation that began in the late 19th century continued for most of the 20th. By the end of World War II, as Michael Lind writes, "the major sectors of the economy were either organized as government-backed cartels or dominated by a few oligopolistic corporations."

For consumers this new world meant the same choices everywhere, but only a few of them. When I grew up there were only 2 or 3 of most things, and since they were all aiming at the middle of the market there wasn't much to differentiate them.

One of the most important instances of this phenomenon was in TV. Here there were 3 choices: NBC, CBS, and ABC. Plus public TV for eggheads and communists. The programs that the 3 networks offered were indistinguishable. In fact, here

there was a triple pressure toward the center. If one show did try something daring, local affiliates in conservative markets would make them stop. Plus since TVs were expensive, whole families watched the same shows together, so they had to be suitable for everyone.

And not only did everyone get the same thing, they got it at the same time. It's difficult to imagine now, but every night tens of millions of families would sit down together in front of their TV set watching the same show, at the same time, as their next door neighbors. What happens now with the Super Bowl used to happen every night. We were literally in sync. [6]

In a way mid-century TV culture was good. The view it gave of the world was like you'd find in a children's book, and it probably had something of the effect that (parents hope) children's books have in making people behave better. But, like children's books, TV was also misleading. Dangerously misleading, for adults. In his autobiography, Robert MacNeil talks of seeing gruesome images that had just come in from Vietnam and thinking, we can't show these to families while they're having dinner.

I know how pervasive the common culture was, because I tried to opt out of it, and it was practically impossible to find alternatives. When I was 13 I realized, more from internal evidence than any outside source, that the ideas we were being fed on TV were crap, and I stopped watching it. [7] But it wasn't just TV. It seemed like everything around me was crap. The politicians all saying the same things, the consumer brands making almost identical products with different labels stuck on to indicate how prestigious they were meant to be, the balloon-frame houses with fake "colonial" skins, the cars with several feet of gratuitous metal on each end that started to fall apart after a couple years, the "red delicious" apples that were red but only nominally apples. And in retrospect, it was crap. [8]

But when I went looking for alternatives to fill this void, I found practically nothing. There was no Internet then. The only place to look was in the chain bookstore in our local shopping mall. [9] There I found a copy of *The Atlantic*. I wish I could say it became a gateway into a wider world, but in fact I found it boring and incomprehensible. Like a kid tasting whisky for the first time and pretending to like it, I preserved that magazine as carefully as if it had been a book. I'm sure I still have it somewhere. But though it was evidence that there was, somewhere, a world that wasn't red delicious, I didn't find it till college.

It wasn't just as consumers that the big companies made us similar. They did as employers too. Within companies there were powerful forces pushing people toward a single model of how to look and act. IBM was particularly notorious for this, but they were only a little more extreme than other big companies. And the models of how to look and act varied little between companies. Meaning everyone within this world was expected to seem more or less the same. And not just those in the corporate world, but also everyone who aspired to it — which in the middle of the 20th century meant most people who weren't already in it. For most of the 20th century, working-class people tried hard to look middle class. You can see it in

old photos. Few adults aspired to look dangerous in 1950.

But the rise of national corporations didn't just compress us culturally. It compressed us economically too, and on both ends.

Along with giant national corporations, we got giant national labor unions. And in the mid 20th century the corporations cut deals with the unions where they paid over market price for labor. Partly because the unions were monopolies. [\[10\]](#) Partly because, as components of oligopolies themselves, the corporations knew they could safely pass the cost on to their customers, because their competitors would have to as well. And partly because in mid-century most of the giant companies were still focused on finding new ways to milk economies of scale. Just as startups rightly pay AWS a premium over the cost of running their own servers so they can focus on growth, many of the big national corporations were willing to pay a premium for labor. [\[11\]](#)

As well as pushing incomes up from the bottom, by overpaying unions, the big companies of the 20th century also pushed incomes down at the top, by underpaying their top management. Economist J. K. Galbraith wrote in 1967 that "There are few corporations in which it would be suggested that executive salaries are at a maximum." [\[12\]](#)

To some extent this was an illusion. Much of the de facto pay of executives never showed up on their income tax returns, because it took the form of perks. The higher the rate of income tax, the more pressure there was to pay employees upstream of it. (In the UK, where taxes were even higher than in the US, companies would even pay their kids' private school tuitions.) One of the most valuable things the big companies of the mid 20th century gave their employees was job security, and this too didn't show up in tax returns or income statistics. So the nature of employment in these organizations tended to yield falsely low numbers about economic inequality. But even accounting for that, the big companies paid their best people less than market price. There was no market; the expectation was that you'd work for the same company for decades if not your whole career. [\[13\]](#)

Your work was so illiquid there was little chance of getting market price. But that same illiquidity also encouraged you not to seek it. If the company promised to employ you till you retired and give you a pension afterward, you didn't want to extract as much from it this year as you could. You needed to take care of the company so it could take care of you. Especially when you'd been working with the same group of people for decades. If you tried to squeeze the company for more money, you were squeezing the organization that was going to take care of *them*. Plus if you didn't put the company first you wouldn't be promoted, and if you couldn't switch ladders, promotion on this one was the only way up. [\[14\]](#)

To someone who'd spent several formative years in the armed forces, this situation didn't seem as strange as it does to us now. From their point of view, as big company executives, they were high-ranking officers. They got paid a lot more

than privates. They got to have expense account lunches at the best restaurants and fly around on the company's Gulfstreams. It probably didn't occur to most of them to ask if they were being paid market price.

The ultimate way to get market price is to work for yourself, by starting your own company. That seems obvious to any ambitious person now. But in the mid 20th century it was an alien concept. Not because starting one's own company seemed too ambitious, but because it didn't seem ambitious enough. Even as late as the 1970s, when I grew up, the ambitious plan was to get lots of education at prestigious institutions, and then join some other prestigious institution and work one's way up the hierarchy. Your prestige was the prestige of the institution you belonged to. People did start their own businesses of course, but educated people rarely did, because in those days there was practically zero concept of starting what we now call a [startup](#): a business that starts small and grows big. That was much harder to do in the mid 20th century. Starting one's own business meant starting a business that would start small and stay small. Which in those days of big companies often meant scurrying around trying to avoid being trampled by elephants. It was more prestigious to be one of the executive class riding the elephant.

By the 1970s, no one stopped to wonder where the big prestigious companies had come from in the first place. It seemed like they'd always been there, like the chemical elements. And indeed, there was a double wall between ambitious kids in the 20th century and the origins of the big companies. Many of the big companies were roll-ups that didn't have clear founders. And when they did, the founders didn't seem like us. Nearly all of them had been uneducated, in the sense of not having been to college. They were what Shakespeare called rude mechanicals. College trained one to be a member of the professional classes. Its graduates didn't expect to do the sort of grubby menial work that Andrew Carnegie or Henry Ford started out doing. [\[15\]](#)

And in the 20th century there were more and more college graduates. They increased from about 2% of the population in 1900 to about 25% in 2000. In the middle of the century our two big forces intersect, in the form of the GI Bill, which sent 2.2 million World War II veterans to college. Few thought of it in these terms, but the result of making college the canonical path for the ambitious was a world in which it was socially acceptable to work for Henry Ford, but not to be Henry Ford. [\[16\]](#)

I remember this world well. I came of age just as it was starting to break up. In my childhood it was still dominant. Not quite so dominant as it had been. We could see from old TV shows and yearbooks and the way adults acted that people in the 1950s and 60s had been even more conformist than us. The mid-century model was already starting to get old. But that was not how we saw it at the time. We would at most have said that one could be a bit more daring in 1975 than 1965. And indeed, things hadn't changed much yet.

But change was coming soon. And when the Duplo economy started to

disintegrate, it disintegrated in several different ways at once. Vertically integrated companies literally dis-integrated because it was more efficient to. Incumbents faced new competitors as (a) markets went global and (b) technical innovation started to trump economies of scale, turning size from an asset into a liability. Smaller companies were increasingly able to survive as formerly narrow channels to consumers broadened. Markets themselves started to change faster, as whole new categories of products appeared. And last but not least, the federal government, which had previously smiled upon J. P. Morgan's world as the natural state of things, began to realize it wasn't the last word after all.

What J. P. Morgan was to the horizontal axis, Henry Ford was to the vertical. He wanted to do everything himself. The giant plant he built at River Rouge between 1917 and 1928 literally took in iron ore at one end and sent cars out the other. 100,000 people worked there. At the time it seemed the future. But that is not how car companies operate today. Now much of the design and manufacturing happens in a long supply chain, whose products the car companies ultimately assemble and sell. The reason car companies operate this way is that it works better. Each company in the supply chain focuses on what they know best. And they each have to do it well or they can be swapped out for another supplier.

Why didn't Henry Ford realize that networks of cooperating companies work better than a single big company? One reason is that supplier networks take a while to evolve. In 1917, doing everything himself seemed to Ford the only way to get the scale he needed. And the second reason is that if you want to solve a problem using a network of cooperating companies, you have to be able to coordinate their efforts, and you can do that much better with computers. Computers reduce the transaction costs that Coase argued are the *raison d'être* of corporations. That is a fundamental change.

In the early 20th century, big companies were synonymous with efficiency. In the late 20th century they were synonymous with inefficiency. To some extent this was because the companies themselves had become sclerotic. But it was also because our standards were higher.

It wasn't just within existing industries that change occurred. The industries themselves changed. It became possible to make lots of new things, and sometimes the existing companies weren't the ones who did it best.

Microcomputers are a classic example. The market was pioneered by upstarts like Apple. When it got big enough, IBM decided it was worth paying attention to. At the time IBM completely dominated the computer industry. They assumed that all they had to do, now that this market was ripe, was to reach out and pick it. Most people at the time would have agreed with them. But what happened next illustrated how much more complicated the world had become. IBM did launch a microcomputer. Though quite successful, it did not crush Apple. But even more importantly, IBM itself ended up being supplanted by a supplier coming in from the side — from software, which didn't even seem to be the same business. IBM's big mistake was to accept a non-exclusive license for DOS. It must have seemed a

safe move at the time. No other computer manufacturer had ever been able to outsell them. What difference did it make if other manufacturers could offer DOS too? The result of that miscalculation was an explosion of inexpensive PC clones. Microsoft now owned the PC standard, and the customer. And the microcomputer business ended up being Apple vs Microsoft.

Basically, Apple bumped IBM and then Microsoft stole its wallet. That sort of thing did not happen to big companies in mid-century. But it was going to happen increasingly often in the future.

Change happened mostly by itself in the computer business. In other industries, legal obstacles had to be removed first. Many of the mid-century oligopolies had been anointed by the federal government with policies (and in wartime, large orders) that kept out competitors. This didn't seem as dubious to government officials at the time as it sounds to us. They felt a two-party system ensured sufficient competition in politics. It ought to work for business too.

Gradually the government realized that anti-competitive policies were doing more harm than good, and during the Carter administration it started to remove them. The word used for this process was misleadingly narrow: deregulation. What was really happening was de-oligopolization. It happened to one industry after another. Two of the most visible to consumers were air travel and long-distance phone service, which both became dramatically cheaper after deregulation.

Deregulation also contributed to the wave of hostile takeovers in the 1980s. In the old days the only limit on the inefficiency of companies, short of actual bankruptcy, was the inefficiency of their competitors. Now companies had to face absolute rather than relative standards. Any public company that didn't generate sufficient returns on its assets risked having its management replaced with one that would. Often the new managers did this by breaking companies up into components that were more valuable separately. [\[17\]](#)

Version 1 of the national economy consisted of a few big blocks whose relationships were negotiated in back rooms by a handful of executives, politicians, regulators, and labor leaders. Version 2 was higher resolution: there were more companies, of more different sizes, making more different things, and their relationships changed faster. In this world there were still plenty of back room negotiations, but more was left to market forces. Which further accelerated the fragmentation.

It's a little misleading to talk of versions when describing a gradual process, but not as misleading as it might seem. There was a lot of change in a few decades, and what we ended up with was qualitatively different. The companies in the S&P 500 in 1958 had been there an average of 61 years. By 2012 that number was 18 years. [\[18\]](#)

The breakup of the Duplo economy happened simultaneously with the spread of computing power. To what extent were computers a precondition? It would take a



book to answer that. Obviously the spread of computing power was a precondition for the rise of startups. I suspect it was for most of what happened in finance too. But was it a precondition for globalization or the LBO wave? I don't know, but I wouldn't discount the possibility. It may be that the refragmentation was driven by computers in the way the industrial revolution was driven by steam engines. Whether or not computers were a precondition, they have certainly accelerated it.

The new fluidity of companies changed people's relationships with their employers. Why climb a corporate ladder that might be yanked out from under you? Ambitious people started to think of a career less as climbing a single ladder than as a series of jobs that might be at different companies. More movement (or even potential movement) between companies introduced more competition in salaries. Plus as companies became smaller it became easier to estimate how much an employee contributed to the company's revenue. Both changes drove salaries toward market price. And since people vary dramatically in productivity, paying market price meant salaries started to diverge.

By no coincidence it was in the early 1980s that the term "yuppie" was coined. That word is not much used now, because the phenomenon it describes is so taken for granted, but at the time it was a label for something novel. Yuppies were young professionals who made lots of money. To someone in their twenties today, this wouldn't seem worth naming. Why wouldn't young professionals make lots of money? But until the 1980s, being underpaid early in your career was part of what it meant to be a professional. Young professionals were paying their dues, working their way up the ladder. The rewards would come later. What was novel about yuppies was that they wanted market price for the work they were doing now.

The first yuppies did not work for startups. That was still in the future. Nor did they work for big companies. They were professionals working in fields like law, finance, and consulting. But their example rapidly inspired their peers. Once they saw that new BMW 325i, they wanted one too.

Underpaying people at the beginning of their career only works if everyone does it. Once some employer breaks ranks, everyone else has to, or they can't get good people. And once started this process spreads through the whole economy, because at the beginnings of people's careers they can easily switch not merely employers but industries.

But not all young professionals benefitted. You had to produce to get paid a lot. It was no coincidence that the first yuppies worked in fields where it was easy to measure that.

More generally, an idea was returning whose name sounds old-fashioned precisely because it was so rare for so long: that you could make your fortune. As in the past there were multiple ways to do it. Some made their fortunes by creating wealth, and others by playing zero-sum games. But once it became possible to make one's fortune, the ambitious had to decide whether or not to. A physicist who chose physics over Wall Street in 1990 was making a sacrifice that a physicist in

1960 didn't have to think about.

The idea even flowed back into big companies. CEOs of big companies make more now than they used to, and I think much of the reason is prestige. In 1960, corporate CEOs had immense prestige. They were the winners of the only economic game in town. But if they made as little now as they did then, in real dollar terms, they'd seem like small fry compared to professional athletes and whiz kids making millions from startups and hedge funds. They don't like that idea, so now they try to get as much as they can, which is more than they had been getting. [\[19\]](#)

Meanwhile a similar fragmentation was happening at the other end of the economic scale. As big companies' oligopolies became less secure, they were less able to pass costs on to customers and thus less willing to overpay for labor. And as the Duplo world of a few big blocks fragmented into many companies of different sizes — some of them overseas — it became harder for unions to enforce their monopolies. As a result workers' wages also tended toward market price. Which (inevitably, if unions had been doing their job) tended to be lower. Perhaps dramatically so, if automation had decreased the need for some kind of work.

And just as the mid-century model induced social as well as economic cohesion, its breakup brought social as well as economic fragmentation. People started to dress and act differently. Those who would later be called the "creative class" became more mobile. People who didn't care much for religion felt less pressure to go to church for appearances' sake, while those who liked it a lot opted for increasingly colorful forms. Some switched from meat loaf to tofu, and others to Hot Pockets. Some switched from driving Ford sedans to driving small imported cars, and others to driving SUVs. Kids who went to private schools or wished they did started to dress "preppy," and kids who wanted to seem rebellious made a conscious effort to look disreputable. In a hundred ways people spread apart. [\[20\]](#)

Almost four decades later, fragmentation is still increasing. Has it been net good or bad? I don't know; the question may be unanswerable. Not entirely bad though. We take for granted the forms of fragmentation we like, and worry only about the ones we don't. But as someone who caught the tail end of mid-century [conformism](#), I can tell you it was no utopia. [\[21\]](#)

My goal here is not to say whether fragmentation has been good or bad, just to explain why it's happening. With the centripetal forces of total war and 20th century oligopoly mostly gone, what will happen next? And more specifically, is it possible to reverse some of the fragmentation we've seen?

If it is, it will have to happen piecemeal. You can't reproduce mid-century cohesion the way it was originally produced. It would be insane to go to war just to induce more national unity. And once you understand the degree to which the economic history of the 20th century was a low-res version 1, it's clear you can't reproduce that either.

20th century cohesion was something that happened at least in a sense naturally. The war was due mostly to external forces, and the Duplo economy was an evolutionary phase. If you want cohesion now, you'd have to induce it deliberately. And it's not obvious how. I suspect the best we'll be able to do is address the symptoms of fragmentation. But that may be enough.

The form of fragmentation people worry most about lately is [economic inequality](#), and if you want to eliminate that you're up against a truly formidable headwind that has been in operation since the stone age. Technology.

Technology is a lever. It magnifies work. And the lever not only grows increasingly long, but the rate at which it grows is itself increasing.

Which in turn means the variation in the amount of wealth people can create has not only been increasing, but accelerating. The unusual conditions that prevailed in the mid 20th century masked this underlying trend. The ambitious had little choice but to join large organizations that made them march in step with lots of other people — literally in the case of the armed forces, figuratively in the case of big corporations. Even if the big corporations had wanted to pay people proportionate to their value, they couldn't have figured out how. But that constraint has gone now. Ever since it started to erode in the 1970s, we've seen the underlying forces at work again. [\[22\]](#)

Not everyone who gets rich now does it by creating wealth, certainly. But a significant number do, and the Baumol Effect means all their peers get dragged along too. [\[23\]](#) And as long as it's possible to get rich by creating wealth, the default tendency will be for economic inequality to increase. Even if you eliminate all the other ways to get rich. You can mitigate this with subsidies at the bottom and taxes at the top, but unless taxes are high enough to discourage people from creating wealth, you're always going to be fighting a losing battle against increasing variation in productivity. [\[24\]](#)

That form of fragmentation, like the others, is here to stay. Or rather, back to stay. Nothing is forever, but the tendency toward fragmentation should be more forever than most things, precisely because it's not due to any particular cause. It's simply a reversion to the mean. When Rockefeller said individualism was gone, he was right for a hundred years. It's back now, and that's likely to be true for longer.

I worry that if we don't acknowledge this, we're headed for trouble. If we think 20th century cohesion disappeared because of few policy tweaks, we'll be deluded into thinking we can get it back (minus the bad parts, somehow) with a few countertweaks. And then we'll waste our time trying to eliminate fragmentation, when we'd be better off thinking about how to mitigate its consequences.

## Notes

[1] Lester Thurow, writing in 1975, said the wage differentials prevailing at the end of World War II had become so embedded that they "were regarded as 'just' even after the egalitarian pressures of World War II had disappeared. Basically, the same differentials exist to this day, thirty years later." But Goldin and Margo think market forces in the postwar period also helped preserve the wartime compression of wages — specifically increased demand for unskilled workers, and oversupply of educated ones.

(Oddly enough, the American custom of having employers pay for health insurance derives from efforts by businesses to circumvent NWLB wage controls in order to attract workers.)

[2] As always, tax rates don't tell the whole story. There were lots of exemptions, especially for individuals. And in World War II the tax codes were so new that the government had little acquired immunity to tax avoidance. If the rich paid high taxes during the war it was more because they wanted to than because they had to.

After the war, federal tax receipts as a percentage of GDP were about the same as they are now. In fact, for the entire period since the war, tax receipts have stayed close to 18% of GDP, despite dramatic changes in tax rates. The lowest point occurred when marginal income tax rates were highest: 14.1% in 1950. Looking at the data, it's hard to avoid the conclusion that tax rates have had little effect on what people actually paid.

[3] Though in fact the decade preceding the war had been a time of unprecedented federal power, in response to the Depression. Which is not entirely a coincidence, because the Depression was one of the causes of the war. In many ways the New Deal was a sort of dress rehearsal for the measures the federal government took during wartime. The wartime versions were much more drastic and more pervasive though. As Anthony Badger wrote, "for many Americans the decisive change in their experiences came not with the New Deal but with World War II."

[4] I don't know enough about the origins of the world wars to say, but it's not inconceivable they were connected to the rise of big corporations. If that were the case, 20th century cohesion would have a single cause.

[5] More precisely, there was a bimodal economy consisting, in Galbraith's words, of "the world of the technically dynamic, massively capitalized and highly organized corporations on the one hand and the hundreds of thousands of small and traditional proprietors on the other." Money, prestige, and power were concentrated in the former, and there was near zero crossover.

[6] I wonder how much of the decline in families eating together was due to the

decline in families watching TV together afterward.

[7] I know when this happened because it was the season *Dallas* premiered. Everyone else was talking about what was happening on *Dallas*, and I had no idea what they meant.

[8] I didn't realize it till I started doing research for this essay, but the meretriciousness of the products I grew up with is a well-known byproduct of oligopoly. When companies can't compete on price, they compete on tailfins.

[9] Monroeville Mall was at the time of its completion in 1969 the largest in the country. In the late 1970s the movie *Dawn of the Dead* was shot there. Apparently the mall was not just the location of the movie, but its inspiration; the crowds of shoppers drifting through this huge mall reminded George Romero of zombies. My first job was scooping ice cream in the Baskin-Robbins.

[10] Labor unions were exempted from antitrust laws by the Clayton Antitrust Act in 1914 on the grounds that a person's work is not "a commodity or article of commerce." I wonder if that means service companies are also exempt.

[11] The relationships between unions and unionized companies can even be symbiotic, because unions will exert political pressure to protect their hosts. According to Michael Lind, when politicians tried to attack the A&P supermarket chain because it was putting local grocery stores out of business, "A&P successfully defended itself by allowing the unionization of its workforce in 1938, thereby gaining organized labor as a constituency." I've seen this phenomenon myself: hotel unions are responsible for more of the political pressure against Airbnb than hotel companies.

[12] Galbraith was clearly puzzled that corporate executives would work so hard to make money for other people (the shareholders) instead of themselves. He devoted much of *The New Industrial State* to trying to figure this out.

His theory was that professionalism had replaced money as a motive, and that modern corporate executives were, like (good) scientists, motivated less by financial rewards than by the desire to do good work and thereby earn the respect of their peers. There is something in this, though I think lack of movement between companies combined with self-interest explains much of observed behavior.

[13] Galbraith (p. 94) says a 1952 study of the 800 highest paid executives at 300 big corporations found that three quarters of them had been with their company for more than 20 years.

[14] It seems likely that in the first third of the 20th century executive salaries were low partly because companies then were more dependent on banks, who would have disapproved if executives got too much. This was certainly true in the beginning. The first big company CEOs were J. P. Morgan's hired hands.

Companies didn't start to finance themselves with retained earnings till the 1920s. Till then they had to pay out their earnings in dividends, and so depended on banks for capital for expansion. Bankers continued to sit on corporate boards till the Glass-Steagall act in 1933.

By mid-century big companies funded 3/4 of their growth from earnings. But the early years of bank dependence, reinforced by the financial controls of World War II, must have had a big effect on social conventions about executive salaries. So it may be that the lack of movement between companies was as much the effect of low salaries as the cause.

Incidentally, the switch in the 1920s to financing growth with retained earnings was one cause of the 1929 crash. The banks now had to find someone else to lend to, so they made more margin loans.

[15] Even now it's hard to get them to. One of the things I find hardest to get into the heads of would-be startup founders is how important it is to do certain kinds of menial work early in the life of a company. Doing [things that don't scale](#) is to how Henry Ford got started as a high-fiber diet is to the traditional peasant's diet: they had no choice but to do the right thing, while we have to make a conscious effort.

[16] Founders weren't celebrated in the press when I was a kid. "Our founder" meant a photograph of a severe-looking man with a walrus mustache and a wing collar who had died decades ago. The thing to be when I was a kid was an *executive*. If you weren't around then it's hard to grasp the cachet that term had. The fancy version of everything was called the "executive" model.

[17] The wave of hostile takeovers in the 1980s was enabled by a combination of circumstances: court decisions striking down state anti-takeover laws, starting with the Supreme Court's 1982 decision in *Edgar v. MITE Corp.*; the Reagan administration's comparatively sympathetic attitude toward takeovers; the Depository Institutions Act of 1982, which allowed banks and savings and loans to buy corporate bonds; a new SEC rule issued in 1982 (rule 415) that made it possible to bring corporate bonds to market faster; the creation of the junk bond business by Michael Milken; a vogue for conglomerates in the preceding period that caused many companies to be combined that never should have been; a decade of inflation that left many public companies trading below the value of their assets; and not least, the increasing complacency of managements.

[18] Foster, Richard. "Creative Destruction Whips through Corporate America." *Innosight*, February 2012.

[19] CEOs of big companies may be overpaid. I don't know enough about big companies to say. But it is certainly not impossible for a CEO to make 200x as much difference to a company's revenues as the average employee. Look at what Steve Jobs did for Apple when he came back as CEO. It would have been a good deal for the board to give him 95% of the company. Apple's market cap the day

Steve came back in July 1997 was 1.73 billion. 5% of Apple now (January 2016) would be worth about 30 billion. And it would not be if Steve hadn't come back; Apple probably wouldn't even exist anymore.

Merely including Steve in the sample might be enough to answer the question of whether public company CEOs in the aggregate are overpaid. And that is not as facile a trick as it might seem, because the broader your holdings, the more the aggregate is what you care about.

[20] The late 1960s were famous for social upheaval. But that was more rebellion (which can happen in any era if people are provoked sufficiently) than fragmentation. You're not seeing fragmentation unless you see people breaking off to both left and right.

[21] Globally the trend has been in the other direction. While the US is becoming more fragmented, the world as a whole is becoming less fragmented, and mostly in good ways.

[22] There were a handful of ways to make a fortune in the mid 20th century. The main one was drilling for oil, which was open to newcomers because it was not something big companies could dominate through economies of scale. How did individuals accumulate large fortunes in an era of such high taxes? Giant tax loopholes defended by two of the most powerful men in Congress, Sam Rayburn and Lyndon Johnson.

But becoming a Texas oilman was not in 1950 something one could aspire to the way starting a startup or going to work on Wall Street were in 2000, because (a) there was a strong local component and (b) success depended so much on luck.

[23] The Baumol Effect induced by startups is very visible in Silicon Valley. Google will pay people millions of dollars a year to keep them from leaving to start or join startups.

[24] I'm not claiming variation in productivity is the only cause of economic inequality in the US. But it's a significant cause, and it will become as big a cause as it needs to, in the sense that if you ban other ways to get rich, people who want to get rich will use this route instead.

**Thanks** to Sam Altman, Trevor Blackwell, Paul Buchheit, Patrick Collison, Ron Conway, Chris Dixon, Benedict Evans, Richard Florida, Ben Horowitz, Jessica Livingston, Robert Morris, Tim O'Reilly, Geoff Ralston, Max Roser, Alexia Tsotsis, and Qasar Younis for reading drafts of this. Max also told me about several valuable sources.

## Bibliography

Allen, Frederick Lewis. *The Big Change*. Harper, 1952.

Averitt, Robert. *The Dual Economy*. Norton, 1968.

Badger, Anthony. *The New Deal*. Hill and Wang, 1989.

Bainbridge, John. *The Super-Americans*. Doubleday, 1961.

Beatty, Jack. *Colossus*. Broadway, 2001.

Brinkley, Douglas. *Wheels for the World*. Viking, 2003.

Brownlee, W. Elliot. *Federal Taxation in America*. Cambridge, 1996.

Chandler, Alfred. *The Visible Hand*. Harvard, 1977.

Chernow, Ron. *The House of Morgan*. Simon & Schuster, 1990.

Chernow, Ron. *Titan: The Life of John D. Rockefeller*. Random House, 1998.

Galbraith, John. *The New Industrial State*. Houghton Mifflin, 1967.

Goldin, Claudia and Robert A. Margo. "The Great Compression: The Wage Structure in the United States at Mid-Century." NBER Working Paper 3817, 1991.

Gordon, John. *An Empire of Wealth*. HarperCollins, 2004.

Klein, Maury. *The Genesis of Industrial America, 1870-1920*. Cambridge, 2007.

Lind, Michael. *Land of Promise*. HarperCollins, 2012.

Mickelthwaite, John, and Adrian Wooldridge. *The Company*. Modern Library, 2003.

Nasaw, David. *Andrew Carnegie*. Penguin, 2006.

Sobel, Robert. *The Age of Giant Corporations*. Praeger, 1993.

Thurow, Lester. *Generating Inequality: Mechanisms of Distribution*. Basic Books, 1975.

Witte, John. *The Politics and Development of the Federal Income Tax*. Wisconsin, 1985.



**Related:**

[Too Many Elite American Men Are Obsessed With Work and Wealth](#)

# Economic Inequality

January 2016

Since the 1970s, economic inequality in the US has increased dramatically. And in particular, the rich have gotten a lot richer. Nearly everyone who writes about the topic says that economic inequality should be decreased.

I'm interested in this question because I was one of the founders of a company called Y Combinator that helps people start startups. Almost by definition, if a startup succeeds, its founders become rich. Which means by helping startup founders I've been helping to increase economic inequality. If economic inequality should be decreased, I shouldn't be helping founders. No one should be.

But that doesn't sound right. What's going on here? What's going on is that while economic inequality is a single measure (or more precisely, two: variation in income, and variation in wealth), it has multiple causes. Many of these causes are bad, like tax loopholes and drug addiction. But some are good, like Larry Page and Sergey Brin starting the company you use to find things online.

If you want to understand economic inequality — and more importantly, if you actually want to fix the bad aspects of it — you have to tease apart the components. And yet the trend in nearly everything written about the subject is to do the opposite: to squash together all the aspects of economic inequality as if it were a single phenomenon.

Sometimes this is done for ideological reasons. Sometimes it's because the writer only has very high-level data and so draws conclusions from that, like the proverbial drunk who looks for his keys under the lamppost, instead of where he dropped them, because the light is better there. Sometimes it's because the writer doesn't understand critical aspects of inequality, like the role of technology in wealth creation. Much of the time, perhaps most of the time, writing about economic inequality combines all three.

---

The most common mistake people make about economic inequality is to treat it as a single phenomenon. The most naive version of which is the one based on the pie fallacy: that the rich get rich by taking money from the poor.

Usually this is an assumption people start from rather than a conclusion they arrive at by examining the evidence. Sometimes the pie fallacy is stated explicitly:

...those at the top are grabbing an increasing fraction of the nation's income — so much of a larger share that what's left over for the rest is diminished.... [\[1\]](#)

Other times it's more unconscious. But the unconscious form is very widespread. I think because we grow up in a world where the pie fallacy is actually true. To kids, wealth *is* a fixed pie that's shared out, and if one person gets more, it's at the expense of another. It takes a conscious effort to remind oneself that the real world doesn't work that way.

In the real world you can create wealth as well as taking it from others. A woodworker creates wealth. He makes a chair, and you willingly give him money in return for it. A high-frequency trader does not. He makes a dollar only when someone on the other end of a trade loses a dollar.

If the rich people in a society got that way by taking wealth from the poor, then you have the degenerate case of economic inequality, where the cause of poverty is the same as the cause of wealth. But instances of inequality don't have to be instances of the degenerate case. If one woodworker makes 5 chairs and another makes none, the second woodworker will have less money, but not because anyone took anything from him.

Even people sophisticated enough to know about the pie fallacy are led toward it by the custom of describing economic inequality as a ratio of one quantile's income or wealth to another's. It's so easy to slip from talking about income shifting from one quantile to another, as a figure of speech, into believing that is literally what's happening.

Except in the degenerate case, economic inequality can't be described by a ratio or even a curve. In the general case it consists of multiple ways people become poor, and multiple ways people become rich. Which means to understand economic inequality in a country, you have to go find individual people who are poor or rich and figure out why. [\[2\]](#)

If you want to understand *change* in economic inequality, you should ask what those people would have done when it was different. This is one way I know the rich aren't all getting richer simply from some new system for transferring wealth to them from everyone else. When you use the would-have method with startup founders, you find what most would have done [back in 1960](#), when economic inequality was lower, was to join big companies or become professors. Before Mark Zuckerberg started Facebook, his default expectation was that he'd end up working at Microsoft. The reason he and most other startup founders are richer than they would have been in the mid 20th century is not because of some right turn the country took during the Reagan administration, but because progress in technology

has made it much easier to start a new company that [grows fast](#).

Traditional economists seem strangely averse to studying individual humans. It seems to be a rule with them that everything has to start with statistics. So they give you very precise numbers about variation in wealth and income, then follow it with the most naive speculation about the underlying causes.

But while there are a lot of people who get rich through rent-seeking of various forms, and a lot who get rich by playing zero-sum games, there are also a significant number who get rich by creating wealth. And creating wealth, as a source of economic inequality, is different from taking it — not just morally, but also practically, in the sense that it is harder to eradicate. One reason is that variation in productivity is accelerating. The rate at which individuals can create wealth depends on the technology available to them, and that grows exponentially. The other reason creating wealth is such a tenacious source of inequality is that it can expand to accommodate a lot of people.

---

I'm all for shutting down the crooked ways to get rich. But that won't eliminate great variations in wealth, because as long as you leave open the option of getting rich by creating wealth, people who want to get rich will do that instead.

Most people who get rich tend to be fairly driven. Whatever their other flaws, laziness is usually not one of them. Suppose new policies make it hard to make a fortune in finance. Does it seem plausible that the people who currently go into finance to make their fortunes will continue to do so, but be content to work for ordinary salaries? The reason they go into finance is not because they love finance but because they want to get rich. If the only way left to get rich is to start startups, they'll start startups. They'll do well at it too, because determination is the main factor in the success of a startup. [3] And while it would probably be a good thing for the world if people who wanted to get rich switched from playing zero-sum games to creating wealth, that would not only not eliminate great variations in wealth, but might even exacerbate them. In a zero-sum game there is at least a limit to the upside. Plus a lot of the new startups would create new technology that further accelerated variation in productivity.

Variation in productivity is far from the only source of economic inequality, but it is the irreducible core of it, in the sense that you'll have that left when you eliminate all other sources. And if you do, that core will be big, because it will have expanded to include the efforts of all the refugees. Plus it will have a large Baumol penumbra around it: anyone who could get rich by creating wealth on their own account will have to be paid enough to prevent them from doing it.

You can't prevent great variations in wealth without preventing people from getting rich, and you can't do that without preventing them from starting startups.

So let's be clear about that. Eliminating great variations in wealth would mean eliminating startups. And that doesn't seem a wise move. Especially since it would only mean you eliminated startups in your own country. Ambitious people already move halfway around the world to further their careers, and startups can operate from anywhere nowadays. So if you made it impossible to get rich by creating wealth in your country, people who wanted to do that would just leave and do it somewhere else. Which would certainly get you a lower Gini coefficient, along with a lesson in being careful what you ask for. [4]

I think rising economic inequality is the inevitable fate of countries that don't choose something worse. We had a 40 year stretch in the middle of the 20th century that convinced some people otherwise. But as I explained in [The Refragmentation](#), that was an anomaly — a unique combination of circumstances that compressed American society not just economically but culturally too. [5]

And while some of the growth in economic inequality we've seen since then has been due to bad behavior of various kinds, there has simultaneously been a huge increase in individuals' ability to create wealth. Startups are almost entirely a product of this period. And even within the startup world, there has been a qualitative change in the last 10 years. Technology has decreased the cost of starting a startup so much that founders now have the upper hand over investors. Founders get less diluted, and it is now common for them to retain [board control](#) as well. Both further increase economic inequality, the former because founders own more stock, and the latter because, as investors have learned, founders tend to be better at running their companies than investors.

While the surface manifestations change, the underlying forces are very, very old. The acceleration of productivity we see in Silicon Valley has been happening for thousands of years. If you look at the history of stone tools, technology was already accelerating in the Mesolithic. The acceleration would have been too slow to perceive in one lifetime. Such is the nature of the leftmost part of an exponential curve. But it was the same curve.

You do not want to design your society in a way that's incompatible with this curve. The evolution of technology is one of the most powerful forces in history.

Louis Brandeis said "We may have democracy, or we may have wealth concentrated in the hands of a few, but we can't have both." That sounds plausible. But if I have to choose between ignoring him and ignoring an exponential curve that has been operating for thousands of years, I'll bet on the curve. Ignoring any trend that has been operating for thousands of years is dangerous. But exponential growth, especially, tends to bite you.

---

If accelerating variation in productivity is always going to produce some baseline growth in economic inequality, it would be a good idea to spend some time

thinking about that future. Can you have a healthy society with great variation in wealth? What would it look like?

Notice how novel it feels to think about that. The public conversation so far has been exclusively about the need to decrease economic inequality. We've barely given a thought to how to live with it.

I'm hopeful we'll be able to. Brandeis was a product of the Gilded Age, and things have changed since then. It's harder to hide wrongdoing now. And to get rich now you don't have to buy politicians the way railroad or oil magnates did. [6] The great concentrations of wealth I see around me in Silicon Valley don't seem to be destroying democracy.

There are lots of things wrong with the US that have economic inequality as a symptom. We should fix those things. In the process we may decrease economic inequality. But we can't start from the symptom and hope to fix the underlying causes. [7]

The most obvious is poverty. I'm sure most of those who want to decrease economic inequality want to do it mainly to help the poor, not to hurt the rich. [8] Indeed, a good number are merely being sloppy by speaking of decreasing economic inequality when what they mean is decreasing poverty. But this is a situation where it would be good to be precise about what we want. Poverty and economic inequality are not identical. When the city is turning off your [water](#) because you can't pay the bill, it doesn't make any difference what Larry Page's net worth is compared to yours. He might only be a few times richer than you, and it would still be just as much of a problem that your water was getting turned off.

Closely related to poverty is lack of social mobility. I've seen this myself: you don't have to grow up rich or even upper middle class to get rich as a startup founder, but few successful founders grew up desperately poor. But again, the problem here is not simply economic inequality. There is an enormous difference in wealth between the household Larry Page grew up in and that of a successful startup founder, but that didn't prevent him from joining their ranks. It's not economic inequality per se that's blocking social mobility, but some specific combination of things that go wrong when kids grow up sufficiently poor.

One of the most important principles in Silicon Valley is that "you make what you measure." It means that if you pick some number to focus on, it will tend to improve, but that you have to choose the right number, because only the one you choose will improve; another that seems conceptually adjacent might not. For example, if you're a university president and you decide to focus on graduation rates, then you'll improve graduation rates. But only graduation rates, not how much students learn. Students could learn less, if to improve graduation rates you made classes easier.

Economic inequality is sufficiently far from identical with the various problems that have it as a symptom that we'll probably only hit whichever of the two we aim at.

If we aim at economic inequality, we won't fix these problems. So I say let's aim at the problems.

For example, let's attack poverty, and if necessary damage wealth in the process. That's much more likely to work than attacking wealth in the hope that you will thereby fix poverty. [9] And if there are people getting rich by tricking consumers or lobbying the government for anti-competitive regulations or tax loopholes, then let's stop them. Not because it's causing economic inequality, but because it's stealing. [10]

If all you have is statistics, it seems like that's what you need to fix. But behind a broad statistical measure like economic inequality there are some things that are good and some that are bad, some that are historical trends with immense momentum and others that are random accidents. If we want to fix the world behind the statistics, we have to understand it, and focus our efforts where they'll do the most good.

## Notes

[1] Stiglitz, Joseph. *The Price of Inequality*. Norton, 2012. p. 32.

[2] Particularly since economic inequality is a matter of outliers, and outliers are disproportionately likely to have gotten where they are by ways that have little to do with the sort of things economists usually think about, like wages and productivity, but rather by, say, ending up on the wrong side of the "War on Drugs."

[3] Determination is the most important factor in deciding between success and failure, which in startups tend to be sharply differentiated. But it takes more than determination to create one of the hugely successful startups. Though most founders start out excited about the idea of getting rich, purely mercenary founders will usually take one of the big acquisition offers most successful startups get on the way up. The founders who go on to the next stage tend to be driven by a sense of mission. They have the same attachment to their companies that an artist or writer has to their work. But it is very hard to predict at the outset which founders will do that. It's not simply a function of their initial attitude. Starting a company changes people.

[4] After reading a draft of this essay, Richard Florida told me how he had once talked to a group of Europeans "who said they wanted to make Europe more entrepreneurial and more like Silicon Valley. I said by definition this will give you

more inequality. They thought I was insane — they could not process it."

[5] Economic inequality has been decreasing globally. But this is mainly due to the erosion of the kleptocracies that formerly dominated all the poorer countries. Once the playing field is leveler politically, we'll see economic inequality start to rise again. The US is the bellwether. The situation we face here, the rest of the world will sooner or later.

[6] Some people still get rich by buying politicians. My point is that it's no longer a precondition.

[7] As well as problems that have economic inequality as a symptom, there are those that have it as a cause. But in most if not all, economic inequality is not the primary cause. There is usually some injustice that is allowing economic inequality to turn into other forms of inequality, and that injustice is what we need to fix. For example, the police in the US treat the poor worse than the rich. But the solution is not to make people richer. It's to make the police treat people more equitably. Otherwise they'll continue to maltreat people who are weak in other ways.

[8] Some who read this essay will say that I'm clueless or even being deliberately misleading by focusing so much on the richer end of economic inequality — that economic inequality is really about poverty. But that is exactly the point I'm making, though sloppier language than I'd use to make it. The real problem is poverty, not economic inequality. And if you conflate them you're aiming at the wrong target.

Others will say I'm clueless or being misleading by focusing on people who get rich by creating wealth — that startups aren't the problem, but corrupt practices in finance, healthcare, and so on. Once again, that is exactly my point. The problem is not economic inequality, but those specific abuses.

It's a strange task to write an essay about why something isn't the problem, but that's the situation you find yourself in when so many people mistakenly think it is.

[9] Particularly since many causes of poverty are only partially driven by people trying to make money from them. For example, America's abnormally high incarceration rate is a major cause of poverty. But although [for-profit prison companies](#) and [prison guard unions](#) both spend a lot lobbying for harsh sentencing laws, they are not the original source of them.

[10] Incidentally, tax loopholes are definitely not a product of some power shift due to recent increases in economic inequality. The golden age of economic equality in the mid 20th century was also the golden age of tax avoidance. Indeed, it was so widespread and so effective that I'm skeptical whether economic inequality was really so low then as we think. In a period when people are trying to hide wealth from the government, it will tend to be hidden from statistics too. One sign of the potential magnitude of the problem is the discrepancy between government receipts as a percentage of GDP, which have remained more or less



constant during the entire period from the end of World War II to the present, and tax rates, which have varied dramatically.

**Thanks** to Sam Altman, Tiffani Ashley Bell, Patrick Collison, Ron Conway, Richard Florida, Ben Horowitz, Jessica Livingston, Robert Morris, Tim O'Reilly, Max Roser, and Alexia Tsotsis for reading drafts of this.

**Note:** This is a new version from which I removed a pair of metaphors that made a lot of people mad, essentially by macroexpanding them. If anyone wants to see the old version, I put it [here](#).

## **Related:**

[The Short Version](#)

[A Reply to Ezra Klein](#)

[A Reply to Russell Okung](#)

[French Translation](#)

# Life is Short

January 2016

Life is short, as everyone knows. When I was a kid I used to wonder about this. Is life actually short, or are we really complaining about its finiteness? Would we be just as likely to feel life was short if we lived 10 times as long?

Since there didn't seem any way to answer this question, I stopped wondering about it. Then I had kids. That gave me a way to answer the question, and the answer is that life actually is short.

Having kids showed me how to convert a continuous quantity, time, into discrete quantities. You only get 52 weekends with your 2 year old. If Christmas-as-magic lasts from say ages 3 to 10, you only get to watch your child experience it 8 times. And while it's impossible to say what is a lot or a little of a continuous quantity like time, 8 is not a lot of something. If you had a handful of 8 peanuts, or a shelf of 8 books to choose from, the quantity would definitely seem limited, no matter what your lifespan was.

Ok, so life actually is short. Does it make any difference to know that?

It has for me. It means arguments of the form "Life is too short for x" have great force. It's not just a figure of speech to say that life is too short for something. It's not just a synonym for annoying. If you find yourself thinking that life is too short for something, you should try to eliminate it if you can.

When I ask myself what I've found life is too short for, the word that pops into my head is "bullshit." I realize that answer is somewhat tautological. It's almost the definition of bullshit that it's the stuff that life is too short for. And yet bullshit does have a distinctive character. There's something fake about it. It's the junk food of experience. [\[1\]](#)

If you ask yourself what you spend your time on that's bullshit, you probably already know the answer. Unnecessary meetings, pointless disputes, bureaucracy, posturing, dealing with other people's mistakes, traffic jams, addictive but unrewarding pastimes.

There are two ways this kind of thing gets into your life: it's either forced on you, or it tricks you. To some extent you have to put up with the bullshit forced on you

by circumstances. You need to make money, and making money consists mostly of errands. Indeed, the law of supply and demand ensures that: the more rewarding some kind of work is, the cheaper people will do it. It may be that less bullshit is forced on you than you think, though. There has always been a stream of people who opt out of the default grind and go live somewhere where opportunities are fewer in the conventional sense, but life feels more authentic. This could become more common.

You can do it on a smaller scale without moving. The amount of time you have to spend on bullshit varies between employers. Most large organizations (and many small ones) are steeped in it. But if you consciously prioritize bullshit avoidance over other factors like money and prestige, you can probably find employers that will waste less of your time.

If you're a freelancer or a small company, you can do this at the level of individual customers. If you fire or avoid toxic customers, you can decrease the amount of bullshit in your life by more than you decrease your income.

But while some amount of bullshit is inevitably forced on you, the bullshit that sneaks into your life by tricking you is no one's fault but your own. And yet the bullshit you choose may be harder to eliminate than the bullshit that's forced on you. Things that lure you into wasting your time have to be really good at tricking you. An example that will be familiar to a lot of people is arguing online. When someone contradicts you, they're in a sense attacking you. Sometimes pretty overtly. Your instinct when attacked is to defend yourself. But like a lot of instincts, this one wasn't designed for the world we now live in. Counterintuitive as it feels, it's better most of the time not to defend yourself. Otherwise these people are literally taking your life. [2]

Arguing online is only incidentally addictive. There are more dangerous things than that. As I've written before, one byproduct of technical progress is that things we like tend to become [more addictive](#). Which means we will increasingly have to make a conscious effort to avoid addictions 💎 to stand outside ourselves and ask "is this how I want to be spending my time?"

As well as avoiding bullshit, one should actively seek out things that matter. But different things matter to different people, and most have to learn what matters to them. A few are lucky and realize early on that they love math or taking care of animals or writing, and then figure out a way to spend a lot of time doing it. But most people start out with a life that's a mix of things that matter and things that don't, and only gradually learn to distinguish between them.

For the young especially, much of this confusion is induced by the artificial situations they find themselves in. In middle school and high school, what the other kids think of you seems the most important thing in the world. But when you ask adults what they got wrong at that age, nearly all say they cared too much what other kids thought of them.

One heuristic for distinguishing stuff that matters is to ask yourself whether you'll care about it in the future. Fake stuff that matters usually has a sharp peak of seeming to matter. That's how it tricks you. The area under the curve is small, but its shape jabs into your consciousness like a pin.

The things that matter aren't necessarily the ones people would call "important." Having coffee with a friend matters. You won't feel later like that was a waste of time.

One great thing about having small children is that they make you spend time on things that matter: them. They grab your sleeve as you're staring at your phone and say "will you play with me?" And odds are that is in fact the bullshit-minimizing option.

If life is short, we should expect its shortness to take us by surprise. And that is just what tends to happen. You take things for granted, and then they're gone. You think you can always write that book, or climb that mountain, or whatever, and then you realize the window has closed. The saddest windows close when other people die. Their lives are short too. After my mother died, I wished I'd spent more time with her. I lived as if she'd always be there. And in her typical quiet way she encouraged that illusion. But an illusion it was. I think a lot of people make the same mistake I did.

The usual way to avoid being taken by surprise by something is to be consciously aware of it. Back when life was more precarious, people used to be aware of death to a degree that would now seem a bit morbid. I'm not sure why, but it doesn't seem the right answer to be constantly reminding oneself of the grim reaper hovering at everyone's shoulder. Perhaps a better solution is to look at the problem from the other end. Cultivate a habit of impatience about the things you most want to do. Don't wait before climbing that mountain or writing that book or visiting your mother. You don't need to be constantly reminding yourself why you shouldn't wait. Just don't wait.

I can think of two more things one does when one doesn't have much of something: try to get more of it, and savor what one has. Both make sense here.

How you live affects how long you live. Most people could do better. Me among them.

But you can probably get even more effect by paying closer attention to the time you have. It's easy to let the days rush by. The "flow" that imaginative people love so much has a darker cousin that prevents you from pausing to savor life amid the daily slurry of errands and alarms. One of the most striking things I've read was not in a book, but the title of one: James Salter's *Burning the Days*.

It is possible to slow time somewhat. I've gotten better at it. Kids help. When you have small children, there are a lot of moments so perfect that you can't help noticing.

It does help too to feel that you've squeezed everything out of some experience. The reason I'm sad about my mother is not just that I miss her but that I think of all the things we could have done that we didn't. My oldest son will be 7 soon. And while I miss the 3 year old version of him, I at least don't have any regrets over what might have been. We had the best time a daddy and a 3 year old ever had.

Relentlessly prune bullshit, don't wait to do things that matter, and savor the time you have. That's what you do when life is short.

## Notes

[1] At first I didn't like it that the word that came to mind was one that had other meanings. But then I realized the other meanings are fairly closely related. Bullshit in the sense of things you waste your time on is a lot like intellectual bullshit.

[2] I chose this example deliberately as a note to self. I get attacked a lot online. People tell the craziest lies about me. And I have so far done a pretty mediocre job of suppressing the natural human inclination to say "Hey, that's not true!"

**Thanks** to Jessica Livingston and Geoff Ralston for reading drafts of this.

[Korean Translation](#)

[Japanese Translation](#)

[Chinese Translation](#)

# How to Make Pittsburgh a Startup Hub

April 2016

*(This is a talk I gave at an event called Opt412 in Pittsburgh. Much of it will apply to other towns. But not all, because as I say in the talk, Pittsburgh has some important advantages over most would-be startup hubs.)*

What would it take to make Pittsburgh into a startup hub, like Silicon Valley? I understand Pittsburgh pretty well, because I grew up here, in Monroeville. And I understand Silicon Valley pretty well because that's where I live now. Could you get that kind of startup ecosystem going here?

When I agreed to speak here, I didn't think I'd be able to give a very optimistic talk. I thought I'd be talking about what Pittsburgh could do to become a startup hub, very much in the subjunctive. Instead I'm going to talk about what Pittsburgh can do.

What changed my mind was an article I read in, of all places, the *New York Times* food section. The title was "[Pittsburgh's Youth-Driven Food Boom](#)." To most people that might not even sound interesting, let alone something related to startups. But it was electrifying to me to read that title. I don't think I could pick a more promising one if I tried. And when I read the article I got even more excited. It said "people ages 25 to 29 now make up 7.6 percent of all residents, up from 7 percent about a decade ago." Wow, I thought, Pittsburgh could be the next Portland. It could become the cool place all the people in their twenties want to go live.

When I got here a couple days ago, I could feel the difference. I lived here from 1968 to 1984. I didn't realize it at the time, but during that whole period the city was in free fall. On top of the flight to the suburbs that happened everywhere, the steel and nuclear businesses were both dying. Boy are things different now. It's not just that downtown seems a lot more prosperous. There is an energy here that was not here when I was a kid.

When I was a kid, this was a place young people left. Now it's a place that attracts them.

What does that have to do with startups? Startups are made of people, and the average age of the people in a typical startup is right in that 25 to 29 bracket.

I've seen how powerful it is for a city to have those people. Five years ago they shifted the center of gravity of Silicon Valley from the peninsula to San Francisco. Google and Facebook are on the peninsula, but the next generation of big winners are all in SF. The reason the center of gravity shifted was the talent war, for programmers especially. Most 25 to 29 year olds want to live in the city, not down in the boring suburbs. So whether they like it or not, founders know they have to be in the city. I know multiple founders who would have preferred to live down in the Valley proper, but who made themselves move to SF because they knew otherwise they'd lose the talent war.

So being a magnet for people in their twenties is a very promising thing to be. It's hard to imagine a place becoming a startup hub without also being that. When I read that statistic about the increasing percentage of 25 to 29 year olds, I had exactly the same feeling of excitement I get when I see a startup's graphs start to creep upward off the x axis.

Nationally the percentage of 25 to 29 year olds is 6.8%. That means you're .8% ahead. The population is 306,000, so we're talking about a surplus of about 2500 people. That's the population of a small town, and that's just the surplus. So you have a toehold. Now you just have to expand it.

And though "youth-driven food boom" may sound frivolous, it is anything but. Restaurants and cafes are a big part of the personality of a city. Imagine walking down a street in Paris. What are you walking past? Little restaurants and cafes. Imagine driving through some depressing random exurb. What are you driving past? Starbucks and McDonalds and Pizza Hut. As Gertrude Stein said, there is no there there. You could be anywhere.

These independent restaurants and cafes are not just feeding people. They're making there be a there here.

So here is my first concrete recommendation for turning Pittsburgh into the next Silicon Valley: do everything you can to encourage this youth-driven food boom. What could the city do? Treat the people starting these little restaurants and cafes as your users, and go ask them what they want. I can guess at least one thing they might want: a fast permit process. San Francisco has left you a huge amount of room to beat them in that department.

I know restaurants aren't the prime mover though. The prime mover, as the Times article said, is cheap housing. That's a big advantage. But that phrase "cheap housing" is a bit misleading. There are plenty of places that are cheaper. What's special about Pittsburgh is not that it's cheap, but that it's a cheap place you'd actually want to live.

Part of that is the buildings themselves. I realized a long time ago, back when I was a poor twenty-something myself, that the best deals were places that had once been rich, and then became poor. If a place has always been rich, it's nice but

too expensive. If a place has always been poor, it's cheap but grim. But if a place was once rich and then got poor, you can find palaces for cheap. And that's what's bringing people here. When Pittsburgh was rich, a hundred years ago, the people who lived here built big solid buildings. Not always in the best taste, but definitely solid. So here is another piece of advice for becoming a startup hub: don't destroy the buildings that are bringing people here. When cities are on the way back up, like Pittsburgh is now, developers race to tear down the old buildings. Don't let that happen. Focus on historic preservation. Big real estate development projects are not what's bringing the twenty-somethings here. They're the opposite of the new restaurants and cafes; they subtract personality from the city.

The empirical evidence suggests you cannot be too strict about historic preservation. The tougher cities are about it, the better they seem to do.

But the appeal of Pittsburgh is not just the buildings themselves. It's the neighborhoods they're in. Like San Francisco and New York, Pittsburgh is fortunate in being a pre-car city. It's not too spread out. Because those 25 to 29 year olds do not like driving. They prefer walking, or bicycling, or taking public transport. If you've been to San Francisco recently you can't help noticing the huge number of bicyclists. And this is not just a fad that the twenty-somethings have adopted. In this respect they have discovered a better way to live. The beards will go, but not the bikes. Cities where you can get around without driving are just better period. So I would suggest you do everything you can to capitalize on this. As with historic preservation, it seems impossible to go too far.

Why not make Pittsburgh the most bicycle and pedestrian friendly city in the country? See if you can go so far that you make San Francisco seem backward by comparison. If you do, it's very unlikely you'll regret it. The city will seem like a paradise to the young people you want to attract. If they do leave to get jobs elsewhere, it will be with regret at leaving behind such a place. And what's the downside? Can you imagine a headline "City ruined by becoming too bicycle-friendly?" It just doesn't happen.

So suppose cool old neighborhoods and cool little restaurants make this the next Portland. Will that be enough? It will put you in a way better position than Portland itself, because Pittsburgh has something Portland lacks: a first-rate research university. CMU plus little cafes means you have more than hipsters drinking lattes. It means you have hipsters drinking lattes while talking about distributed systems. Now you're getting really close to San Francisco.

In fact you're better off than San Francisco in one way, because CMU is downtown, but Stanford and Berkeley are out in the suburbs.

What can CMU do to help Pittsburgh become a startup hub? Be an even better research university. CMU is one of the best universities in the world, but imagine what things would be like if it were the very best, and everyone knew it. There are a lot of ambitious people who must go to the best place, wherever it is. If CMU were it, they would all come here. There would be kids in Kazakhstan dreaming of



one day living in Pittsburgh.

Being that kind of talent magnet is the most important contribution universities can make toward making their city a startup hub. In fact it is practically the only contribution they can make.

But wait, shouldn't universities be setting up programs with words like "innovation" and "entrepreneurship" in their names? No, they should not. These kind of things almost always turn out to be disappointments. They're pursuing the wrong targets. The way to get innovation is not to aim for innovation but to aim for something more specific, like better batteries or better 3D printing. And the way to learn about entrepreneurship is to do it, which you [can't in school](#).

I know it may disappoint some administrators to hear that the best thing a university can do to encourage startups is to be a great university. It's like telling people who want to lose weight that the way to do it is to eat less.

But if you want to know where startups come from, look at the empirical evidence. Look at the histories of the most successful startups, and you'll find they grow organically out of a couple of founders building something that starts as an interesting side project. Universities are great at bringing together founders, but beyond that the best thing they can do is get out of the way. For example, by not claiming ownership of "intellectual property" that students and faculty develop, and by having liberal rules about deferred admission and leaves of absence.

In fact, one of the most effective things a university could do to encourage startups is an elaborate form of getting out of the way invented by Harvard. Harvard used to have exams for the fall semester after Christmas. At the beginning of January they had something called "Reading Period" when you were supposed to be studying for exams. And Microsoft and Facebook have something in common that few people realize: they were both started during Reading Period. It's the perfect situation for producing the sort of side projects that turn into startups. The students are all on campus, but they don't have to do anything because they're supposed to be studying for exams.

Harvard may have closed this window, because a few years ago they moved exams before Christmas and shortened reading period from 11 days to 7. But if a university really wanted to help its students start startups, the empirical evidence, weighted by market cap, suggests the best thing they can do is literally nothing.

The culture of Pittsburgh is another of its strengths. It seems like a city has to be socially liberal to be a startup hub, and it's pretty clear why. A city has to tolerate strangeness to be a home for startups, because startups are so strange. And you can't choose to allow just the forms of strangeness that will turn into big startups, because they're all intermingled. You have to tolerate all strangeness.

That immediately rules out [big chunks of the US](#). I'm optimistic it doesn't rule out Pittsburgh. One of the things I remember from growing up here, though I didn't

realize at the time that there was anything unusual about it, is how well people got along. I'm still not sure why. Maybe one reason was that everyone felt like an immigrant. When I was a kid in Monroeville, people didn't call themselves American. They called themselves Italian or Serbian or Ukranian. Just imagine what it must have been like here a hundred years ago, when people were pouring in from twenty different countries. Tolerance was the only option.

What I remember about the culture of Pittsburgh is that it was both tolerant and pragmatic. That's how I'd describe the culture of Silicon Valley too. And it's not a coincidence, because Pittsburgh was the Silicon Valley of its time. This was a city where people built new things. And while the things people build have changed, the spirit you need to do that kind of work is the same.

So although an influx of latte-swilling hipsters may be annoying in some ways, I would go out of my way to encourage them. And more generally to tolerate strangeness, even unto the degree wacko Californians do. For Pittsburgh that is a conservative choice: it's a return to the city's roots.

Unfortunately I saved the toughest part for last. There is one more thing you need to be a startup hub, and Pittsburgh hasn't got it: investors. Silicon Valley has a big investor community because it's had 50 years to grow one. New York has a big investor community because it's full of people who like money a lot and are quick to notice new ways to get it. But Pittsburgh has neither of these. And the cheap housing that draws other people here has no effect on investors.

If an investor community grows up here, it will happen the same way it did in Silicon Valley: slowly and organically. So I would not bet on having a big investor community in the short term. But fortunately there are three trends that make that less necessary than it used to be. One is that startups are increasingly cheap to start, so you just don't need as much outside money as you used to. The second is that thanks to things like Kickstarter, a startup can get to revenue faster. You can put something on Kickstarter from anywhere. The third is programs like Y Combinator. A startup from anywhere in the world can go to YC for 3 months, pick up funding, and then return home if they want.

My advice is to make Pittsburgh a great place for startups, and gradually more of them will stick. Some of those will succeed; some of their founders will become investors; and still more startups will stick.

This is not a fast path to becoming a startup hub. But it is at least a path, which is something few other cities have. And it's not as if you have to make painful sacrifices in the meantime. Think about what I've suggested you should do. Encourage local restaurants, save old buildings, take advantage of density, make CMU the best, promote tolerance. These are the things that make Pittsburgh good to live in now. All I'm saying is that you should do even more of them.

And that's an encouraging thought. If Pittsburgh's path to becoming a startup hub is to be even more itself, then it has a good chance of succeeding. In fact it

probably has the best chance of any city its size. It will take some effort, and a lot of time, but if any city can do it, Pittsburgh can.

**Thanks** to Charlie Cheever and Jessica Livingston for reading drafts of this, and to Meg Cheever for organizing Opt412 and inviting me to speak.

# The Risk of Discovery

January 2017

Because biographies of famous scientists tend to edit out their mistakes, we underestimate the degree of risk they were willing to take. And because anything a famous scientist did that wasn't a mistake has probably now become the conventional wisdom, those choices don't seem risky either.

Biographies of Newton, for example, understandably focus more on physics than alchemy or theology. The impression we get is that his unerring judgment led him straight to truths no one else had noticed. How to explain all the time he spent on alchemy and theology? Well, smart people are often kind of crazy.

But maybe there is a simpler explanation. Maybe the smartness and the craziness were not as separate as we think. Physics seems to us a promising thing to work on, and alchemy and theology obvious wastes of time. But that's because we know how things turned out. In Newton's day the three problems seemed roughly equally promising. No one knew yet what the payoff would be for inventing what we now call physics; if they had, more people would have been working on it. And alchemy and theology were still then in the category Marc Andreessen would describe as "huge, if true."

Newton made three bets. One of them worked. But they were all risky.

[Japanese Translation](#)

# Charisma / Power

January 2017

People who are powerful but uncharismatic will tend to be disliked. Their power makes them a target for criticism that they don't have the charisma to disarm. That was Hillary Clinton's problem. It also tends to be a problem for any CEO who is more of a builder than a schmoozer. And yet the builder-type CEO is (like Hillary) probably the best person for the job.

I don't think there is any solution to this problem. It's human nature. The best we can do is to recognize that it's happening, and to understand that being a magnet for criticism is sometimes a sign not that someone is the wrong person for a job, but that they're the right one.

# General and Surprising

September 2017

The most valuable insights are both general and surprising.  $F = ma$  for example. But general and surprising is a hard combination to achieve. That territory tends to be picked clean, precisely because those insights are so valuable.

Ordinarily, the best that people can do is one without the other: either surprising without being general (e.g. gossip), or general without being surprising (e.g. platitudes).

Where things get interesting is the moderately valuable insights. You get those from small additions of whichever quality was missing. The more common case is a small addition of generality: a piece of gossip that's more than just gossip, because it teaches something interesting about the world. But another less common approach is to focus on the most general ideas and see if you can find something new to say about them. Because these start out so general, you only need a small delta of novelty to produce a useful insight.

A small delta of novelty is all you'll be able to get most of the time. Which means if you take this route, your ideas will seem a lot like ones that already exist. Sometimes you'll find you've merely rediscovered an idea that did already exist. But don't be discouraged. Remember the huge multiplier that kicks in when you do manage to think of something even a little new.

Corollary: the more general the ideas you're talking about, the less you should worry about repeating yourself. If you write enough, it's inevitable you will. Your brain is much the same from year to year and so are the stimuli that hit it. I feel slightly bad when I find I've said something close to what I've said before, as if I were plagiarizing myself. But rationally one shouldn't. You won't say something exactly the same way the second time, and that variation increases the chance you'll get that tiny but critical delta of novelty.

And of course, ideas beget ideas. (That sounds [familiar](#).) An idea with a small amount of novelty could lead to one with more. But only if you keep going. So it's doubly important not to let yourself be discouraged by people who say there's not much new about something you've discovered. "Not much new" is a real achievement when you're talking about the most general ideas.

It's not true that there's nothing new under the sun. There are some domains where there's almost nothing new. But there's a big difference between nothing and almost nothing, when it's multiplied by the area under the sun.

**Thanks** to Sam Altman, Patrick Collison, and Jessica Livingston for reading drafts of this.

[Japanese Translation](#)

# The Bus Ticket Theory of Genius

November 2019

Everyone knows that to do great work you need both natural ability and determination. But there's a third ingredient that's not as well understood: an obsessive interest in a particular topic.

To explain this point I need to burn my reputation with some group of people, and I'm going to choose bus ticket collectors. There are people who collect old bus tickets. Like many collectors, they have an obsessive interest in the minutiae of what they collect. They can keep track of distinctions between different types of bus tickets that would be hard for the rest of us to remember. Because we don't care enough. What's the point of spending so much time thinking about old bus tickets?

Which leads us to the second feature of this kind of obsession: there is no point. A bus ticket collector's love is disinterested. They're not doing it to impress us or to make themselves rich, but for its own sake.

When you look at the lives of people who've done great work, you see a consistent pattern. They often begin with a bus ticket collector's obsessive interest in something that would have seemed pointless to most of their contemporaries. One of the most striking features of Darwin's book about his voyage on the Beagle is the sheer depth of his interest in natural history. His curiosity seems infinite. Ditto for Ramanujan, sitting by the hour working out on his slate what happens to series.

It's a mistake to think they were "laying the groundwork" for the discoveries they made later. There's too much intention in that metaphor. Like bus ticket collectors, they were doing it because they liked it.

But there is a difference between Ramanujan and a bus ticket collector. Series matter, and bus tickets don't.

If I had to put the recipe for genius into one sentence, that might be it: to have a disinterested obsession with something that matters.

Aren't I forgetting about the other two ingredients? Less than you might think. An obsessive interest in a topic is both a proxy for ability and a substitute for



determination. Unless you have sufficient mathematical aptitude, you won't find series interesting. And when you're obsessively interested in something, you don't need as much determination: you don't need to push yourself as hard when curiosity is pulling you.

An obsessive interest will even bring you luck, to the extent anything can. Chance, as Pasteur said, favors the prepared mind, and if there's one thing an obsessed mind is, it's prepared.

The disinterestedness of this kind of obsession is its most important feature. Not just because it's a filter for earnestness, but because it helps you discover new ideas.

The paths that lead to new ideas tend to look unpromising. If they looked promising, other people would already have explored them. How do the people who do great work discover these paths that others overlook? The popular story is that they simply have better vision: because they're so talented, they see paths that others miss. But if you look at the way great discoveries are made, that's not what happens. Darwin didn't pay closer attention to individual species than other people because he saw that this would lead to great discoveries, and they didn't. He was just really, really interested in such things.

Darwin couldn't turn it off. Neither could Ramanujan. They didn't discover the hidden paths that they did because they seemed promising, but because they couldn't help it. That's what allowed them to follow paths that someone who was merely ambitious would have ignored.

What rational person would decide that the way to write great novels was to begin by spending several years creating an imaginary elvish language, like Tolkien, or visiting every household in southwestern Britain, like Trollope? No one, including Tolkien and Trollope.

The bus ticket theory is similar to Carlyle's famous definition of genius as an infinite capacity for taking pains. But there are two differences. The bus ticket theory makes it clear that the source of this infinite capacity for taking pains is not infinite diligence, as Carlyle seems to have meant, but the sort of infinite interest that collectors have. It also adds an important qualification: an infinite capacity for taking pains about something that matters.

So what matters? You can never be sure. It's precisely because no one can tell in advance which paths are promising that you can discover new ideas by working on what you're interested in.

But there are some heuristics you can use to guess whether an obsession might be one that matters. For example, it's more promising if you're creating something, rather than just consuming something someone else creates. It's more promising if something you're interested in is difficult, especially if it's [more difficult for other people](#) than it is for you. And the obsessions of talented people are more likely to

be promising. When talented people become interested in random things, they're not truly random.

But you can never be sure. In fact, here's an interesting idea that's also rather alarming if it's true: it may be that to do great work, you also have to waste a lot of time.

In many different areas, reward is proportionate to risk. If that rule holds here, then the way to find paths that lead to truly great work is to be willing to expend a lot of effort on things that turn out to be every bit as unpromising as they seem.

I'm not sure if this is true. On one hand, it seems surprisingly difficult to waste your time so long as you're working hard on something interesting. So much of what you do ends up being useful. But on the other hand, the rule about the relationship between risk and reward is so powerful that it seems to hold wherever risk occurs. [Newton's](#) case, at least, suggests that the risk/reward rule holds here. He's famous for one particular obsession of his that turned out to be unprecedentedly fruitful: using math to describe the world. But he had two other obsessions, alchemy and theology, that seem to have been complete wastes of time. He ended up net ahead. His bet on what we now call physics paid off so well that it more than compensated for the other two. But were the other two necessary, in the sense that he had to take big risks to make such big discoveries? I don't know.

Here's an even more alarming idea: might one make all bad bets? It probably happens quite often. But we don't know how often, because these people don't become famous.

It's not merely that the returns from following a path are hard to predict. They change dramatically over time. 1830 was a really good time to be obsessively interested in natural history. If Darwin had been born in 1709 instead of 1809, we might never have heard of him.

What can one do in the face of such uncertainty? One solution is to hedge your bets, which in this case means to follow the obviously promising paths instead of your own private obsessions. But as with any hedge, you're decreasing reward when you decrease risk. If you forgo working on what you like in order to follow some more conventionally ambitious path, you might miss something wonderful that you'd otherwise have discovered. That too must happen all the time, perhaps even more often than the genius whose bets all fail.

The other solution is to let yourself be interested in lots of different things. You don't decrease your upside if you switch between equally genuine interests based on which seems to be working so far. But there is a danger here too: if you work on too many different projects, you might not get deeply enough into any of them.

One interesting thing about the bus ticket theory is that it may help explain why different types of people excel at different kinds of work. Interest is much more

unevenly distributed than ability. If natural ability is all you need to do great work, and natural ability is evenly distributed, you have to invent elaborate theories to explain the skewed distributions we see among those who actually do great work in various fields. But it may be that much of the skew has a simpler explanation: different people are interested in different things.

The bus ticket theory also explains why people are less likely to do great work after they have children. Here interest has to compete not just with external obstacles, but with another interest, and one that for most people is extremely powerful. It's harder to find time for work after you have kids, but that's the easy part. The real change is that you don't want to.

But the most exciting implication of the bus ticket theory is that it suggests ways to encourage great work. If the recipe for genius is simply natural ability plus hard work, all we can do is hope we have a lot of ability, and work as hard as we can. But if interest is a critical ingredient in genius, we may be able, by cultivating interest, to cultivate genius.

For example, for the very ambitious, the bus ticket theory suggests that the way to do great work is to relax a little. Instead of gritting your teeth and diligently pursuing what all your peers agree is the most promising line of research, maybe you should try doing something just for fun. And if you're stuck, that may be the vector along which to break out.

I've always liked [Hamming's](#) famous double-barrelled question: what are the most important problems in your field, and why aren't you working on one of them? It's a great way to shake yourself up. But it may be overfitting a bit. It might be at least as useful to ask yourself: if you could take a year off to work on something that probably wouldn't be important but would be really interesting, what would it be?

The bus ticket theory also suggests a way to avoid slowing down as you get older. Perhaps the reason people have fewer new ideas as they get older is not simply that they're losing their edge. It may also be because once you become established, you can no longer mess about with irresponsible side projects the way you could when you were young and no one cared what you did.

The solution to that is obvious: remain irresponsible. It will be hard, though, because the apparently random projects you take up to stave off decline will read to outsiders as evidence of it. And you yourself won't know for sure that they're wrong. But it will at least be more fun to work on what you want.

It may even be that we can cultivate a habit of intellectual bus ticket collecting in kids. The usual plan in education is to start with a broad, shallow focus, then gradually become more specialized. But I've done the opposite with my kids. I know I can count on their school to handle the broad, shallow part, so I take them deep.

When they get interested in something, however random, I encourage them to go preposterously, bus ticket collectorly, deep. I don't do this because of the bus ticket theory. I do it because I want them to feel the joy of learning, and they're never going to feel that about something I'm making them learn. It has to be something they're interested in. I'm just following the path of least resistance; depth is a byproduct. But if in trying to show them the joy of learning I also end up training them to go deep, so much the better.

Will it have any effect? I have no idea. But that uncertainty may be the most interesting point of all. There is so much more to learn about how to do great work. As old as human civilization feels, it's really still very young if we haven't nailed something so basic. It's exciting to think there are still discoveries to make about discovery. If that's the sort of thing you're interested in.

## Notes

[1] There are other types of collecting that illustrate this point better than bus tickets, but they're also more popular. It seemed just as well to use an inferior example rather than offend more people by telling them their hobby doesn't matter.

[2] I worried a little about using the word "disinterested," since some people mistakenly believe it means not interested. But anyone who expects to be a genius will have to know the meaning of such a basic word, so I figure they may as well start now.

[3] Think how often genius must have been nipped in the bud by people being told, or telling themselves, to stop messing about and be responsible. Ramanujan's mother was a huge enabler. Imagine if she hadn't been. Imagine if his parents had made him go out and get a job instead of sitting around at home doing math.

On the other hand, anyone quoting the preceding paragraph to justify not getting a job is probably mistaken.

[4] 1709 Darwin is to time what the [Milanese Leonardo](#) is to space.

[5] "An infinite capacity for taking pains" is a paraphrase of what Carlyle wrote. What he wrote, in his *History of Frederick the Great*, was "... it is the fruit of

'genius' (which means transcendent capacity of taking trouble, first of all)...." Since the paraphrase seems the name of the idea at this point, I kept it.

Carlyle's *History* was published in 1858. In 1785 Hérault de Séchelles quoted Buffon as saying "Le génie n'est qu'une plus grande aptitude à la patience." (Genius is only a greater aptitude for patience.)

[6] Trollope was establishing the system of postal routes. He himself sensed the obsessiveness with which he pursued this goal.

It is amusing to watch how a passion will grow upon a man. During those two years it was the ambition of my life to cover the country with rural letter-carriers.

Even Newton occasionally sensed the degree of his obsessiveness. After computing pi to 15 digits, he wrote in a letter to a friend:

I am ashamed to tell you to how many figures I carried these computations, having no other business at the time.

Incidentally, Ramanujan was also a compulsive calculator. As Kanigel writes in his excellent biography:

One Ramanujan scholar, B. M. Wilson, later told how Ramanujan's research into number theory was often "preceded by a table of numerical results, carried usually to a length from which most of us would shrink."

[7] Working to understand the natural world counts as creating rather than consuming.

Newton tripped over this distinction when he chose to work on theology. His beliefs did not allow him to see it, but chasing down paradoxes in nature is fruitful in a way that chasing down paradoxes in sacred texts is not.

[8] How much of people's propensity to become interested in a topic is inborn? My experience so far suggests the answer is: most of it. Different kids get interested in different things, and it's hard to make a child interested in something they wouldn't otherwise be. Not in a way that sticks. The most you can do on behalf of a topic is to make sure it gets a fair showing to make it clear to them, for example, that there's more to math than the dull drills they do in school. After that it's up to the child.

**Thanks** to Marc Andreessen, Trevor Blackwell, Patrick Collison, Kevin Lacker, Jessica Livingston, Jackie McDonough, Robert Morris, Lisa Randall, Zak Stone, and [my 7 year old](#) for reading drafts of this.

[Spanish Translation](#)

[Russian Translation](#)

[Korean Translation](#)

[Armenian Translation](#)

# Novelty and Heresy

November 2019

If you discover something new, there's a significant chance you'll be accused of some form of heresy.

To discover new things, you have to work on ideas that are good but non-obvious; if an idea is obviously good, other people are probably already working on it. One common way for a good idea to be non-obvious is for it to be hidden in the shadow of some mistaken assumption that people are very attached to. But anything you discover from working on such an idea will tend to contradict the mistaken assumption that was concealing it. And you will thus get a lot of heat from people attached to the mistaken assumption. Galileo and Darwin are famous examples of this phenomenon, but it's probably always an ingredient in the resistance to new ideas.

So it's particularly dangerous for an organization or society to have a culture of pouncing on heresy. When you suppress heresies, you don't just prevent people from contradicting the mistaken assumption you're trying to protect. You also suppress any idea that implies indirectly that it's false.

Every cherished mistaken assumption has a dead zone of unexplored ideas around it. And the more preposterous the assumption, the bigger the dead zone it creates.

There is a positive side to this phenomenon though. If you're looking for new ideas, one way to find them is by [looking for heresies](#). When you look at the question this way, the depressingly large dead zones around mistaken assumptions become excitingly large mines of new ideas.

[Japanese Translation](#)

[Russian Translation](#)





# The Lesson to Unlearn

December 2019

The most damaging thing you learned in school wasn't something you learned in any specific class. It was learning to get good grades.

When I was in college, a particularly earnest philosophy grad student once told me that he never cared what grade he got in a class, only what he learned in it. This stuck in my mind because it was the only time I ever heard anyone say such a thing.

For me, as for most students, the measurement of what I was learning completely dominated actual learning in college. I was fairly earnest; I was genuinely interested in most of the classes I took, and I worked hard. And yet I worked by far the hardest when I was studying for a test.

In theory, tests are merely what their name implies: tests of what you've learned in the class. In theory you shouldn't have to prepare for a test in a class any more than you have to prepare for a blood test. In theory you learn from taking the class, from going to the lectures and doing the reading and/or assignments, and the test that comes afterward merely measures how well you learned.

In practice, as almost everyone reading this will know, things are so different that hearing this explanation of how classes and tests are meant to work is like hearing the etymology of a word whose meaning has changed completely. In practice, the phrase "studying for a test" was almost redundant, because that was when one really studied. The difference between diligent and slack students was that the former studied hard for tests and the latter didn't. No one was pulling all-nighters two weeks into the semester.

Even though I was a diligent student, almost all the work I did in school was aimed at getting a good grade on something.

To many people, it would seem strange that the preceding sentence has a "though" in it. Aren't I merely stating a tautology? Isn't that what a diligent student is, a straight-A student? That's how deeply the conflation of learning with grades has infused our culture.

Is it so bad if learning is conflated with grades? Yes, it is bad. And it wasn't till

decades after college, when I was running Y Combinator, that I realized how bad it is.

I knew of course when I was a student that studying for a test is far from identical with actual learning. At the very least, you don't retain knowledge you cram into your head the night before an exam. But the problem is worse than that. The real problem is that most tests don't come close to measuring what they're supposed to.

If tests truly were tests of learning, things wouldn't be so bad. Getting good grades and learning would converge, just a little late. The problem is that nearly all tests given to students are terribly hackable. Most people who've gotten good grades know this, and know it so well they've ceased even to question it. You'll see when you realize how naive it sounds to act otherwise.

Suppose you're taking a class on medieval history and the final exam is coming up. The final exam is supposed to be a test of your knowledge of medieval history, right? So if you have a couple days between now and the exam, surely the best way to spend the time, if you want to do well on the exam, is to read the best books you can find about medieval history. Then you'll know a lot about it, and do well on the exam.

No, no, no, experienced students are saying to themselves. If you merely read good books on medieval history, most of the stuff you learned wouldn't be on the test. It's not good books you want to read, but the lecture notes and assigned reading in this class. And even most of that you can ignore, because you only have to worry about the sort of thing that could turn up as a test question. You're looking for sharply-defined chunks of information. If one of the assigned readings has an interesting digression on some subtle point, you can safely ignore that, because it's not the sort of thing that could be turned into a test question. But if the professor tells you that there were three underlying causes of the Schism of 1378, or three main consequences of the Black Death, you'd better know them. And whether they were in fact the causes or consequences is beside the point. For the purposes of this class they are.

At a university there are often copies of old exams floating around, and these narrow still further what you have to learn. As well as learning what kind of questions this professor asks, you'll often get actual exam questions. Many professors re-use them. After teaching a class for 10 years, it would be hard not to, at least inadvertently.

In some classes, your professor will have had some sort of political axe to grind, and if so you'll have to grind it too. The need for this varies. In classes in math or the hard sciences or engineering it's rarely necessary, but at the other end of the spectrum there are classes where you couldn't get a good grade without it.

Getting a good grade in a class on  $x$  is so different from learning a lot about  $x$  that you have to choose one or the other, and you can't blame students if they choose

grades. Everyone judges them by their grades ♦ graduate programs, employers, scholarships, even their own parents.

I liked learning, and I really enjoyed some of the papers and programs I wrote in college. But did I ever, after turning in a paper in some class, sit down and write another just for fun? Of course not. I had things due in other classes. If it ever came to a choice of learning or grades, I chose grades. I hadn't come to college to do badly.

Anyone who cares about getting good grades has to play this game, or they'll be surpassed by those who do. And at elite universities, that means nearly everyone, since someone who didn't care about getting good grades probably wouldn't be there in the first place. The result is that students compete to maximize the difference between learning and getting good grades.

Why are tests so bad? More precisely, why are they so hackable? Any experienced programmer could answer that. How hackable is software whose author hasn't paid any attention to preventing it from being hacked? Usually it's as porous as a colander.

Hackable is the default for any test imposed by an authority. The reason the tests you're given are so consistently bad ♦ so consistently far from measuring what they're supposed to measure ♦ is simply that the people creating them haven't made much effort to prevent them from being hacked.

But you can't blame teachers if their tests are hackable. Their job is to teach, not to create unhackable tests. The real problem is grades, or more precisely, that grades have been overloaded. If grades were merely a way for teachers to tell students what they were doing right and wrong, like a coach giving advice to an athlete, students wouldn't be tempted to hack tests. But unfortunately after a certain age grades become more than advice. After a certain age, whenever you're being taught, you're usually also being judged.

I've used college tests as an example, but those are actually the least hackable. All the tests most students take their whole lives are at least as bad, including, most spectacularly of all, the test that gets them into college. If getting into college were merely a matter of having the quality of one's mind measured by admissions officers the way scientists measure the mass of an object, we could tell teenage kids "learn a lot" and leave it at that. You can tell how bad college admissions are, as a test, from how unlike high school that sounds. In practice, the freakishly specific nature of the stuff ambitious kids have to do in high school is directly proportionate to the hackability of college admissions. The classes you don't care about that are mostly memorization, the random "extracurricular activities" you have to participate in to show you're "well-rounded," the standardized tests as artificial as chess, the "essay" you have to write that's presumably meant to hit some very specific target, but you're not told what.

As well as being bad in what it does to kids, this test is also bad in the sense of

being very hackable. So hackable that whole industries have grown up to hack it. This is the explicit purpose of test-prep companies and admissions counsellors, but it's also a significant part of the function of private schools.

Why is this particular test so hackable? I think because of what it's measuring. Although the popular story is that the way to get into a good college is to be really smart, admissions officers at elite colleges neither are, nor claim to be, looking only for that. What are they looking for? They're looking for people who are not simply smart, but admirable in some more general sense. And how is this more general admirableness measured? The admissions officers feel it. In other words, they accept who they like.

So what college admissions is a test of is whether you suit the taste of some group of people. Well, of course a test like that is going to be hackable. And because it's both very hackable and there's (thought to be) a lot at stake, it's hacked like nothing else. That's why it distorts your life so much for so long.

It's no wonder high school students often feel alienated. The shape of their lives is completely artificial.

But wasting your time is not the worst thing the educational system does to you. The worst thing it does is to train you that the way to win is by hacking bad tests. This is a much subtler problem that I didn't recognize until I saw it happening to other people.

When I started advising startup founders at Y Combinator, especially young ones, I was puzzled by the way they always seemed to make things overcomplicated. How, they would ask, do you raise money? What's the trick for making venture capitalists want to invest in you? The best way to make VCs want to invest in you, I would explain, is to actually be a good investment. Even if you could trick VCs into investing in a bad startup, you'd be tricking yourselves too. You're investing time in the same company you're asking them to invest money in. If it's not a good investment, why are you even doing it?

Oh, they'd say, and then after a pause to digest this revelation, they'd ask: What makes a startup a good investment?

So I would explain that what makes a startup promising, not just in the eyes of investors but in fact, is [growth](#). Ideally in revenue, but failing that in usage. What they needed to do was get lots of users.

How does one get lots of users? They had all kinds of ideas about that. They needed to do a big launch that would get them "exposure." They needed influential people to talk about them. They even knew they needed to launch on a tuesday, because that's when one gets the most attention.

No, I would explain, that is not how to get lots of users. The way you get lots of users is to make the product really great. Then people will not only use it but

recommend it to their friends, so your growth will be exponential once you [get it started](#).

At this point I've told the founders something you'd think would be completely obvious: that they should make a good company by making a good product. And yet their reaction would be something like the reaction many physicists must have had when they first heard about the theory of relativity: a mixture of astonishment at its apparent genius, combined with a suspicion that anything so weird couldn't possibly be right. Ok, they would say, dutifully. And could you introduce us to such-and-such influential person? And remember, we want to launch on Tuesday.

It would sometimes take founders years to grasp these simple lessons. And not because they were lazy or stupid. They just seemed blind to what was right in front of them.

Why, I would ask myself, do they always make things so complicated? And then one day I realized this was not a rhetorical question.

Why did founders tie themselves in knots doing the wrong things when the answer was right in front of them? Because that was what they'd been trained to do. Their education had taught them that the way to win was to hack the test. And without even telling them they were being trained to do this. The younger ones, the recent graduates, had never faced a non-artificial test. They thought this was just how the world worked: that the first thing you did, when facing any kind of challenge, was to figure out what the trick was for hacking the test. That's why the conversation would always start with how to raise money, because that read as the test. It came at the end of YC. It had numbers attached to it, and higher numbers seemed to be better. It must be the test.

There are certainly big chunks of the world where the way to win is to hack the test. This phenomenon isn't limited to schools. And some people, either due to ideology or ignorance, claim that this is true of startups too. But it isn't. In fact, one of the most striking things about startups is the degree to which you win by simply doing good work. There are edge cases, as there are in anything, but in general you win by getting users, and what users care about is whether the product does what they want.

Why did it take me so long to understand why founders made startups overcomplicated? Because I hadn't realized explicitly that schools train us to win by hacking bad tests. And not just them, but me! I'd been trained to hack bad tests too, and hadn't realized it till decades later.

I had lived as if I realized it, but without knowing why. For example, I had avoided working for big companies. But if you'd asked why, I'd have said it was because they were bogus, or bureaucratic. Or just yuck. I never understood how much of my dislike of big companies was due to the fact that you win by hacking bad tests.

Similarly, the fact that the tests were unhackable was a lot of what attracted me to

startups. But again, I hadn't realized that explicitly.

I had in effect achieved by successive approximations something that may have a closed-form solution. I had gradually undone my training in hacking bad tests without knowing I was doing it. Could someone coming out of school banish this demon just by knowing its name, and saying begone? It seems worth trying.

Merely talking explicitly about this phenomenon is likely to make things better, because much of its power comes from the fact that we take it for granted. After you've noticed it, it seems the elephant in the room, but it's a pretty well camouflaged elephant. The phenomenon is so old, and so pervasive. And it's simply the result of neglect. No one meant things to be this way. This is just what happens when you combine learning with grades, competition, and the naive assumption of unhackability.

It was mind-blowing to realize that two of the things I'd puzzled about the most ♦ the bogusness of high school, and the difficulty of getting founders to see the obvious ♦ both had the same cause. It's rare for such a big block to slide into place so late.

Usually when that happens it has implications in a lot of different areas, and this case seems no exception. For example, it suggests both that education could be done better, and how you might fix it. But it also suggests a potential answer to the question all big companies seem to have: how can we be more like a startup? I'm not going to chase down all the implications now. What I want to focus on here is what it means for individuals.

To start with, it means that most ambitious kids graduating from college have something they may want to unlearn. But it also changes how you look at the world. Instead of looking at all the different kinds of work people do and thinking of them vaguely as more or less appealing, you can now ask a very specific question that will sort them in an interesting way: to what extent do you win at this kind of work by hacking bad tests?

It would help if there was a way to recognize bad tests quickly. Is there a pattern here? It turns out there is.

Tests can be divided into two kinds: those that are imposed by authorities, and those that aren't. Tests that aren't imposed by authorities are inherently unhackable, in the sense that no one is claiming they're tests of anything more than they actually test. A football match, for example, is simply a test of who wins, not which team is better. You can tell that from the fact that commentators sometimes say afterward that the better team won. Whereas tests imposed by authorities are usually proxies for something else. A test in a class is supposed to measure not just how well you did on that particular test, but how much you learned in the class. While tests that aren't imposed by authorities are inherently unhackable, those imposed by authorities have to be made unhackable. Usually they aren't. So as a first approximation, bad tests are roughly equivalent to tests

imposed by authorities.

You might actually like to win by hacking bad tests. Presumably some people do. But I bet most people who find themselves doing this kind of work don't like it. They just take it for granted that this is how the world works, unless you want to drop out and be some kind of hippie artisan.

I suspect many people implicitly assume that working in a field with bad tests is the price of making lots of money. But that, I can tell you, is false. It used to be true. In the mid-twentieth century, when the economy was [composed of oligopolies](#), the only way to the top was by playing their game. But it's not true now. There are now ways to get rich by doing good work, and that's part of the reason people are so much more excited about getting rich than they used to be. When I was a kid, you could either become an engineer and make cool things, or make lots of money by becoming an "executive." Now you can make lots of money by making cool things.

Hacking bad tests is becoming less important as the link between work and authority erodes. The erosion of that link is one of the most important trends happening now, and we see its effects in almost every kind of work people do. Startups are one of the most visible examples, but we see much the same thing in writing. Writers no longer have to submit to publishers and editors to reach readers; now they can go direct.

The more I think about this question, the more optimistic I get. This seems one of those situations where we don't realize how much something was holding us back until it's eliminated. And I can foresee the whole bogus edifice crumbling. Imagine what happens as more and more people start to ask themselves if they want to win by hacking bad tests, and decide that they don't. The kinds of work where you win by hacking bad tests will be starved of talent, and the kinds where you win by doing good work will see an influx of the most ambitious people. And as hacking bad tests shrinks in importance, education will evolve to stop training us to do it. Imagine what the world could look like if that happened.

This is not just a lesson for individuals to unlearn, but one for society to unlearn, and we'll be amazed at the energy that's liberated when we do.

[1] If using tests only to measure learning sounds impossibly utopian, that is already the way things work at Lambda School. Lambda School doesn't have grades. You either graduate or you don't. The only purpose of tests is to decide at each stage of the curriculum whether you can continue to the next. So in effect the whole school is pass/fail.

[2] If the final exam consisted of a long conversation with the professor, you could prepare for it by reading good books on medieval history. A lot of the hackability of tests in schools is due to the fact that the same test has to be given to large numbers of students.

[3] Learning is the naive algorithm for getting good grades.

[4] [Hacking](#) has multiple senses. There's a narrow sense in which it means to compromise something. That's the sense in which one hacks a bad test. But there's another, more general sense, meaning to find a surprising solution to a problem, often by thinking differently about it. Hacking in this sense is a wonderful thing. And indeed, some of the hacks people use on bad tests are impressively ingenious; the problem is not so much the hacking as that, because the tests are hackable, they don't test what they're meant to.

[5] The people who pick startups at Y Combinator are similar to admissions officers, except that instead of being arbitrary, their acceptance criteria are trained by a very tight feedback loop. If you accept a bad startup or reject a good one, you will usually know it within a year or two at the latest, and often within a month.

[6] I'm sure admissions officers are tired of reading applications from kids who seem to have no personality beyond being willing to seem however they're supposed to seem to get accepted. What they don't realize is that they are, in a sense, looking in a mirror. The lack of authenticity in the applicants is a reflection of the arbitrariness of the application process. A dictator might just as well complain about the lack of authenticity in the people around him.

[7] By good work, I don't mean morally good, but good in the sense in which a good craftsman does good work.

[8] There are borderline cases where it's hard to say which category a test falls in. For example, is raising venture capital like college admissions, or is it like selling to a customer?

[9] Note that a good test is merely one that's unhackable. Good here doesn't mean morally good, but good in the sense of working well. The difference between fields with bad tests and good ones is not that the former are bad and the latter are good, but that the former are bogus and the latter aren't. But those two measures are not unrelated. As Tara Ploughman said, the path from good to evil goes through bogus.



[10] People who think the recent increase in [economic inequality](#) is due to changes in tax policy seem very naive to anyone with experience in startups. Different people are getting rich now than used to, and they're getting much richer than mere tax savings could make them.

[11] Note to tiger parents: you may think you're training your kids to win, but if you're training them to win by hacking bad tests, you are, as parents so often do, training them to fight the last war.

**Thanks** to Austen Allred, Trevor Blackwell, Patrick Collison, Jessica Livingston, Robert Morris, and Harj Taggar for reading drafts of this.

[Russian Translation](#)

[Arabic Translation](#)

[Swedish Translation](#)

# Having Kids

December 2019

Before I had kids, I was afraid of having kids. Up to that point I felt about kids the way the young Augustine felt about living virtuously. I'd have been sad to think I'd never have children. But did I want them now? No.

If I had kids, I'd become a parent, and parents, as I'd known since I was a kid, were uncool. They were dull and responsible and had no fun. And while it's not surprising that kids would believe that, to be honest I hadn't seen much as an adult to change my mind. Whenever I'd noticed parents with kids, the kids seemed to be terrors, and the parents pathetic harried creatures, even when they prevailed.

When people had babies, I congratulated them enthusiastically, because that seemed to be what one did. But I didn't feel it at all. "Better you than me," I was thinking.

Now when people have babies I congratulate them enthusiastically and I mean it. Especially the first one. I feel like they just got the best gift in the world.

What changed, of course, is that I had kids. Something I dreaded turned out to be wonderful.

Partly, and I won't deny it, this is because of serious chemical changes that happened almost instantly when our first child was born. It was like someone flipped a switch. I suddenly felt protective not just toward our child, but toward all children. As I was driving my wife and new son home from the hospital, I approached a crosswalk full of pedestrians, and I found myself thinking "I have to be really careful of all these people. Every one of them is someone's child!"

So to some extent you can't trust me when I say having kids is great. To some extent I'm like a religious cultist telling you that you'll be happy if you join the cult too 💎 but only because joining the cult will alter your mind in a way that will make you happy to be a cult member.

But not entirely. There were some things about having kids that I clearly got wrong before I had them.

For example, there was a huge amount of selection bias in my observations of parents and children. Some parents may have noticed that I wrote "Whenever I'd noticed parents with kids." Of course the times I noticed kids were when things were going wrong. I only noticed them when they made noise. And where was I when I noticed them? Ordinarily I never went to places with kids, so the only times I encountered them were in shared bottlenecks like airplanes. Which is not exactly a representative sample. Flying with a toddler is something very few parents enjoy.

What I didn't notice, because they tend to be much quieter, were all the great moments parents had with kids. People don't talk about these much ♦ the magic is hard to put into words, and all other parents know about them anyway ♦ but one of the great things about having kids is that there are so many times when you feel there is nowhere else you'd rather be, and nothing else you'd rather be doing. You don't have to be doing anything special. You could just be going somewhere together, or putting them to bed, or pushing them on the swings at the park. But you wouldn't trade these moments for anything. One doesn't tend to associate kids with peace, but that's what you feel. You don't need to look any further than where you are right now.

Before I had kids, I had moments of this kind of peace, but they were rarer. With kids it can happen several times a day.

My other source of data about kids was my own childhood, and that was similarly misleading. I was pretty bad, and was always in trouble for something or other. So it seemed to me that parenthood was essentially law enforcement. I didn't realize there were good times too.

I remember my mother telling me once when I was about 30 that she'd really enjoyed having me and my sister. My god, I thought, this woman is a saint. She not only endured all the pain we subjected her to, but actually enjoyed it? Now I realize she was simply telling the truth.

She said that one reason she liked having us was that we'd been interesting to talk to. That took me by surprise when I had kids. You don't just love them. They become your friends too. They're really interesting. And while I admit small children are disastrously fond of repetition (anything worth doing once is worth doing fifty times) it's often genuinely fun to play with them. That surprised me too. Playing with a 2 year old was fun when I was 2 and definitely not fun when I was 6. Why would it become fun again later? But it does.

There are of course times that are pure drudgery. Or worse still, terror. Having kids is one of those intense types of experience that are hard to imagine unless you've had them. But it is not, as I implicitly believed before having kids, simply your DNA heading for the lifeboats.

Some of my worries about having kids were right, though. They definitely make you less productive. I know having kids makes some people get their act together, but if your act was already together, you're going to have less time to do it in. In

particular, you're going to have to work to a schedule. Kids have schedules. I'm not sure if it's because that's how kids are, or because it's the only way to integrate their lives with adults', but once you have kids, you tend to have to work on their schedule.

You will have chunks of time to work. But you can't let work spill promiscuously through your whole life, like I used to before I had kids. You're going to have to work at the same time every day, whether inspiration is flowing or not, and there are going to be times when you have to stop, even if it is.

I've been able to adapt to working this way. Work, like love, finds a way. If there are only certain times it can happen, it happens at those times. So while I don't get as much done as before I had kids, I get enough done.

I hate to say this, because being ambitious has always been a part of my identity, but having kids may make one less ambitious. It hurts to see that sentence written down. I squirm to avoid it. But if there weren't something real there, why would I squirm? The fact is, once you have kids, you're probably going to care more about them than you do about yourself. And attention is a zero-sum game. Only one idea at a time can be the [top idea in your mind](#). Once you have kids, it will often be your kids, and that means it will less often be some project you're working on.

I have some hacks for sailing close to this wind. For example, when I write essays, I think about what I'd want my kids to know. That drives me to get things right. And when I was writing [Bel](#), I told my kids that once I finished it I'd take them to Africa. When you say that sort of thing to a little kid, they treat it as a promise. Which meant I had to finish or I'd be taking away their trip to Africa. Maybe if I'm really lucky such tricks could put me net ahead. But the wind is there, no question.

On the other hand, what kind of wimpy ambition do you have if it won't survive having kids? Do you have so little to spare?

And while having kids may be warping my present judgement, it hasn't overwritten my memory. I remember perfectly well what life was like before. Well enough to miss some things a lot, like the ability to take off for some other country at a moment's notice. That was so great. Why did I never do that?

See what I did there? The fact is, most of the freedom I had before kids, I never used. I paid for it in loneliness, but I never used it.

I had plenty of happy times before I had kids. But if I count up happy moments, not just potential happiness but actual happy moments, there are more after kids than before. Now I practically have it on tap, almost any bedtime.

People's experiences as parents vary a lot, and I know I've been lucky. But I think the worries I had before having kids must be pretty common, and judging by other parents' faces when they see their kids, so must the happiness that kids bring.

## **Note**

[1] Adults are sophisticated enough to see 2 year olds for the fascinatingly complex characters they are, whereas to most 6 year olds, 2 year olds are just defective 6 year olds.

**Thanks** to Trevor Blackwell, Jessica Livingston, and Robert Morris for reading drafts of this.

[Arabic Translation](#)

[Slovak Translation](#)

# Fashionable Problems

December 2019

I've seen the same pattern in many different fields: even though lots of people have worked hard in the field, only a small fraction of the space of possibilities has been explored, because they've all worked on similar things.

Even the smartest, most imaginative people are surprisingly conservative when deciding what to work on. People who would never dream of being fashionable in any other way get sucked into working on fashionable problems.

If you want to try working on unfashionable problems, one of the best places to look is in fields that people think have already been fully explored: essays, Lisp, venture funding ♦ you may notice a pattern here. If you can find a new approach into a big but apparently played out field, the value of whatever you discover will be [multiplied](#) by its enormous surface area.

The best protection against getting drawn into working on the same things as everyone else may be to [genuinely love](#) what you're doing. Then you'll continue to work on it even if you make the same mistake as other people and think that it's too marginal to matter.

[Japanese Translation](#)

[Arabic Translation](#)

[French Translation](#)

# The Two Kinds of Moderate

December 2019

There are two distinct ways to be politically moderate: on purpose and by accident. Intentional moderates are trimmers, deliberately choosing a position mid-way between the extremes of right and left. Accidental moderates end up in the middle, on average, because they make up their own minds about each question, and the far right and far left are roughly equally wrong.

You can distinguish intentional from accidental moderates by the distribution of their opinions. If the far left opinion on some matter is 0 and the far right opinion 100, an intentional moderate's opinion on every question will be near 50. Whereas an accidental moderate's opinions will be scattered over a broad range, but will, like those of the intentional moderate, average to about 50.

Intentional moderates are similar to those on the far left and the far right in that their opinions are, in a sense, not their own. The defining quality of an ideologue, whether on the left or the right, is to acquire one's opinions in bulk. You don't get to pick and choose. Your opinions about taxation can be predicted from your opinions about sex. And although intentional moderates might seem to be the opposite of ideologues, their beliefs (though in their case the word "positions" might be more accurate) are also acquired in bulk. If the median opinion shifts to the right or left, the intentional moderate must shift with it. Otherwise they stop being moderate.


Accidental moderates, on the other hand, not only choose their own answers, but choose their own questions. They may not care at all about questions that the left and right both think are terribly important. So you can only even measure the politics of an accidental moderate from the intersection of the questions they care about and those the left and right care about, and this can sometimes be vanishingly small.

It is not merely a manipulative rhetorical trick to say "if you're not with us, you're against us," but often simply false.

Moderates are sometimes derided as cowards, particularly by the extreme left. But while it may be accurate to call intentional moderates cowards, openly being an accidental moderate requires the most courage of all, because you get attacked from both right and left, and you don't have the comfort of being an orthodox

member of a large group to sustain you.

Nearly all the most impressive people I know are accidental moderates. If I knew a lot of professional athletes, or people in the entertainment business, that might be different. Being on the far left or far right doesn't affect how fast you run or how well you sing. But someone who works with ideas has to be independent-minded to do it well.

Or more precisely, you have to be independent-minded about the ideas you work with. You could be mindlessly doctrinaire in your politics and still be a good mathematician. In the 20th century, a lot of very smart people were Marxists  just no one who was smart about the subjects Marxism involves. But if the ideas you use in your work intersect with the politics of your time, you have two choices: be an accidental moderate, or be mediocre.

## Notes

[1] It's possible in theory for one side to be entirely right and the other to be entirely wrong. Indeed, ideologues must always believe this is the case. But historically it rarely has been.

[2] For some reason the far right tend to ignore moderates rather than despise them as backsliders. I'm not sure why. Perhaps it means that the far right is less ideological than the far left. Or perhaps that they are more confident, or more resigned, or simply more disorganized. I just don't know.

[3] Having heretical opinions doesn't mean you have to express them openly. It may be [easier to have them](#) if you don't.

**Thanks** to Austen Allred, Trevor Blackwell, Patrick Collison, Jessica Livingston, Amjad Masad, Ryan Petersen, and Harj Taggar for reading drafts of this.

[Japanese Translation](#)





# Haters

January 2020

*(I originally intended this for startup founders, who are often surprised by the attention they get as their companies grow, but it applies equally to anyone who becomes famous.)*

If you become sufficiently famous, you'll acquire some fans who like you too much. These people are sometimes called "fanboys," and though I dislike that term, I'm going to have to use it here. We need some word for them, because this is a distinct phenomenon from someone simply liking your work.

A fanboy is obsessive and uncritical. Liking you becomes part of their identity, and they create an image of you in their own head that is much better than reality. Everything you do is good, because you do it. If you do something bad, they find a way to see it as good. And their love for you is not, usually, a quiet, private one. They want everyone to know how great you are.

Well, you may be thinking, I could do without this kind of obsessive fan, but I know there are all kinds of people in the world, and if this is the worst consequence of fame, that's not so bad.

Unfortunately this is not the worst consequence of fame. As well as fanboys, you'll have haters.

A hater is obsessive and uncritical. Disliking you becomes part of their identity, and they create an image of you in their own head that is much worse than reality. Everything you do is bad, because you do it. If you do something good, they find a way to see it as bad. And their dislike for you is not, usually, a quiet, private one. They want everyone to know how awful you are.

If you're thinking of checking, I'll save you the trouble. The second and fifth paragraphs are identical except for "good" being switched to "bad" and so on.

I spent years puzzling about haters. What are they, and where do they come from? Then one day it dawned on me. Haters are just fanboys with the sign switched.

Note that by haters, I don't simply mean trolls. I'm not talking about people who say bad things about you and then move on. I'm talking about the much smaller

group of people for whom this becomes a kind of obsession and who do it repeatedly over a long period.

Like fans, haters seem to be an automatic consequence of fame. Anyone sufficiently famous will have them. And like fans, haters are energized by the fame of whoever they hate. They hear a song by some pop singer. They don't like it much. If the singer were an obscure one, they'd just forget about it. But instead they keep hearing her name, and this seems to drive some people crazy. Everyone's always going on about this singer, but she's no good! She's a fraud!

That word "fraud" is an important one. It's the spectral signature of a hater to regard the object of their hatred as a [fraud](#). They can't deny their fame. Indeed, their fame is if anything exaggerated in the hater's mind. They notice every mention of the singer's name, because every mention makes them angrier. In their own minds they exaggerate both the singer's fame and her lack of talent, and the only way to reconcile those two ideas is to conclude that she has tricked everyone.

What sort of people become haters? Can anyone become one? I'm not sure about this, but I've noticed some patterns. Haters are generally losers in a very specific sense: although they are occasionally talented, they have never achieved much. And indeed, anyone successful enough to have achieved significant fame would be unlikely to regard another famous person as a fraud on that account, because anyone famous knows how random fame is.

But haters are not always complete losers. They are not always the proverbial guy living in his mom's basement. Many are, but some have some amount of talent. In fact I suspect that a sense of frustrated talent is what drives some people to become haters. They're not just saying "It's unfair that so-and-so is famous," but "It's unfair that so-and-so is famous, and not me."

Could a hater be cured if they achieved something impressive? My guess is that's a moot point, because they [never will](#). I've been able to observe for long enough that I'm fairly confident the pattern works both ways: not only do people who do great work never become haters, haters never do great work. Although I dislike the word "fanboy," it's evocative of something important about both haters and fanboys. It implies that the fanboy is so slavishly predictable in his admiration that he's diminished as a result, that he's less than a man.

Haters seem even more diminished. I can imagine being a fanboy. I can think of people whose work I admire so much that I could abase myself before them out of sheer gratitude. If P. G. Wodehouse were still alive, I could see myself being a Wodehouse fanboy. But I could not imagine being a hater.

Knowing that haters are just fanboys with the sign bit flipped makes it much easier to deal with them. We don't need a separate theory of haters. We can just use existing techniques for dealing with obsessive fans.

The most important of which is simply not to think much about them. If you're like

most people who become famous enough to acquire haters, your initial reaction will be one of mystification. Why does this guy seem to have it in for me? Where does his obsessive energy come from, and what makes him so appallingly nasty? What did I do to set him off? Is it something I can fix?

The mistake here is to think of the hater as someone you have a dispute with. When you have a dispute with someone, it's usually a good idea to try to understand why they're upset and then fix things if you can. Disputes are distracting. But it's a false analogy to think of a hater as someone you have a dispute with. It's an understandable mistake, if you've never encountered haters before. But when you realize that you're dealing with a hater, and what a hater is, it's clear that it's a waste of time even to think about them. If you have obsessive fans, do you spend any time wondering what makes them love you so much? No, you just think "some people are kind of crazy," and that's the end of it.

Since haters are equivalent to fanboys, that's the way to deal with them too. There may have been something that set them off. But it's not something that would have set off a normal person, so there's no reason to spend any time thinking about it. It's not you, it's them.

## Notes

[1] There are of course some people who are genuine frauds. How can you distinguish between x calling y a fraud because x is a hater, and because y is a fraud? Look at neutral opinion. Actual frauds are usually pretty conspicuous. Thoughtful people are rarely taken in by them. So if there are some thoughtful people who like y, you can usually assume y is not a fraud.

[2] I would make an exception for teenagers, who sometimes act in such extreme ways that they are literally not themselves. I can imagine a teenage kid being a hater and then growing out of it. But not anyone over 25.

[3] I have a much worse memory for misdeeds than my wife Jessica, who is a connoisseur of character, but I don't wish it were better. Most disputes are a waste of time even if you're in the right, and it's easy to bury the hatchet with someone if you can't remember why you were mad at them.

[4] A competent hater will not merely attack you individually but will try to get mobs after you. In some cases you may want to refute whatever bogus claim they made in order to do so. But err on the side of not, because ultimately it probably

won't matter.

**Thanks** to Austen Allred, Trevor Blackwell, Patrick Collison, Christine Ford, Daniel Gackle, Jessica Livingston, Robert Morris, Elon Musk, Harj Taggar, and Peter Thiel for reading drafts of this.

[Japanese Translation](#)

[Arabic Translation](#)

[Polish Translation](#)

# Being a Noob

January 2020

When I was young, I thought old people had everything figured out. Now that I'm old, I know this isn't true.

I constantly feel like a noob. It seems like I'm always talking to some startup working in a new field I know nothing about, or reading a book about a topic I don't understand well enough, or visiting some new country where I don't know how things work.

It's not pleasant to feel like a noob. And the word "noob" is certainly not a compliment. And yet today I realized something encouraging about being a noob: the more of a noob you are locally, the less of a noob you are globally.

For example, if you stay in your home country, you'll feel less of a noob than if you move to Farawavia, where everything works differently. And yet you'll know more if you move. So the feeling of being a noob is inversely correlated with actual ignorance.

But if the feeling of being a noob is good for us, why do we dislike it? What evolutionary purpose could such an aversion serve?

I think the answer is that there are two sources of feeling like a noob: being stupid, and doing something novel. Our dislike of feeling like a noob is our brain telling us "Come on, come on, figure this out." Which was the right thing to be thinking for most of human history. The life of hunter-gatherers was complex, but it didn't change as much as life does now. They didn't suddenly have to figure out what to do about cryptocurrency. So it made sense to be biased toward competence at existing problems over the discovery of new ones. It made sense for humans to dislike the feeling of being a noob, just as, in a world where food was scarce, it made sense for them to dislike the feeling of being hungry.

Now that too much food is more of a problem than too little, our dislike of feeling hungry leads us astray. And I think our dislike of feeling like a noob does too.

Though it feels unpleasant, and people will sometimes ridicule you for it, the more you feel like a noob, the better.

[Japanese Translation](#)

[Arabic Translation](#)

[French Translation](#)

[Korean Translation](#)

[Polish Translation](#)

[Chinese Translation](#)

[Serbian Translation](#)

[French Translation](#)

# How to Write Usefully

February 2020

What should an essay be? Many people would say persuasive. That's what a lot of us were taught essays should be. But I think we can aim for something more ambitious: that an essay should be useful.

To start with, that means it should be correct. But it's not enough merely to be correct. It's easy to make a statement correct by making it vague. That's a common flaw in academic writing, for example. If you know nothing at all about an issue, you can't go wrong by saying that the issue is a complex one, that there are many factors to be considered, that it's a mistake to take too simplistic a view of it, and so on.

Though no doubt correct, such statements tell the reader nothing. Useful writing makes claims that are as strong as they can be made without becoming false.

For example, it's more useful to say that Pike's Peak is near the middle of Colorado than merely somewhere in Colorado. But if I say it's in the exact middle of Colorado, I've now gone too far, because it's a bit east of the middle.

Precision and correctness are like opposing forces. It's easy to satisfy one if you ignore the other. The converse of vaporous academic writing is the bold, but false, rhetoric of demagogues. Useful writing is bold, but true.

It's also two other things: it tells people something important, and that at least some of them didn't already know.

Telling people something they didn't know doesn't always mean surprising them. Sometimes it means telling them something they knew unconsciously but had never put into words. In fact those may be the more valuable insights, because they tend to be more fundamental.

Let's put them all together. Useful writing tells people something true and important that they didn't already know, and tells them as unequivocally as possible.

Notice these are all a matter of degree. For example, you can't expect an idea to be novel to everyone. Any insight that you have will probably have already been



had by at least one of the world's 7 billion people. But it's sufficient if an idea is novel to a lot of readers.

Ditto for correctness, importance, and strength. In effect the four components are like numbers you can multiply together to get a score for usefulness. Which I realize is almost awkwardly reductive, but nonetheless true.

---

How can you ensure that the things you say are true and novel and important? Believe it or not, there is a trick for doing this. I learned it from my friend Robert Morris, who has a horror of saying anything dumb. His trick is not to say anything unless he's sure it's worth hearing. This makes it hard to get opinions out of him, but when you do, they're usually right.

Translated into essay writing, what this means is that if you write a bad sentence, you don't publish it. You delete it and try again. Often you abandon whole branches of four or five paragraphs. Sometimes a whole essay.

You can't ensure that every idea you have is good, but you can ensure that every one you publish is, by simply not publishing the ones that aren't.

In the sciences, this is called publication bias, and is considered bad. When some hypothesis you're exploring gets inconclusive results, you're supposed to tell people about that too. But with essay writing, publication bias is the way to go.

My strategy is loose, then tight. I write the first draft of an essay fast, trying out all kinds of ideas. Then I spend days rewriting it very carefully.

I've never tried to count how many times I proofread essays, but I'm sure there are sentences I've read 100 times before publishing them. When I proofread an essay, there are usually passages that stick out in an annoying way, sometimes because they're clumsily written, and sometimes because I'm not sure they're true. The annoyance starts out unconscious, but after the tenth reading or so I'm saying "Ugh, that part" each time I hit it. They become like briars that catch your sleeve as you walk past. Usually I won't publish an essay till they're all gone ♦ till I can read through the whole thing without the feeling of anything catching.

I'll sometimes let through a sentence that seems clumsy, if I can't think of a way to rephrase it, but I will never knowingly let through one that doesn't seem correct. You never have to. If a sentence doesn't seem right, all you have to do is ask why it doesn't, and you've usually got the replacement right there in your head.

This is where essayists have an advantage over journalists. You don't have a deadline. You can work for as long on an essay as you need to get it right. You don't have to publish the essay at all, if you can't get it right. Mistakes seem to

lose courage in the face of an enemy with unlimited resources. Or that's what it feels like. What's really going on is that you have different expectations for yourself. You're like a parent saying to a child "we can sit here all night till you eat your vegetables." Except you're the child too.

I'm not saying no mistake gets through. For example, I added condition (c) in ["A Way to Detect Bias"](#) after readers pointed out that I'd omitted it. But in practice you can catch nearly all of them.

There's a trick for getting importance too. It's like the trick I suggest to young founders for getting startup ideas: to make something you yourself want. You can use yourself as a proxy for the reader. The reader is not completely unlike you, so if you write about topics that seem important to you, they'll probably seem important to a significant number of readers as well.

Importance has two factors. It's the number of people something matters to, times how much it matters to them. Which means of course that it's not a rectangle, but a sort of ragged comb, like a Riemann sum.

The way to get novelty is to write about topics you've thought about a lot. Then you can use yourself as a proxy for the reader in this department too. Anything you notice that surprises you, who've thought about the topic a lot, will probably also surprise a significant number of readers. And here, as with correctness and importance, you can use the Morris technique to ensure that you will. If you don't learn anything from writing an essay, don't publish it.

You need humility to measure novelty, because acknowledging the novelty of an idea means acknowledging your previous ignorance of it. Confidence and humility are often seen as opposites, but in this case, as in many others, confidence helps you to be humble. If you know you're an expert on some topic, you can freely admit when you learn something you didn't know, because you can be confident that most other people wouldn't know it either.

The fourth component of useful writing, strength, comes from two things: thinking well, and the skillful use of qualification. These two counterbalance each other, like the accelerator and clutch in a car with a manual transmission. As you try to refine the expression of an idea, you adjust the qualification accordingly. Something you're sure of, you can state baldly with no qualification at all, as I did the four components of useful writing. Whereas points that seem dubious have to be held at arm's length with perhapses.

As you refine an idea, you're pushing in the direction of less qualification. But you can rarely get it down to zero. Sometimes you don't even want to, if it's a side point and a fully refined version would be too long.

Some say that qualifications weaken writing. For example, that you should never begin a sentence in an essay with "I think," because if you're saying it, then of course you think it. And it's true that "I think x" is a weaker statement than simply

"x." Which is exactly why you need "I think." You need it to express your degree of certainty.

But qualifications are not scalars. They're not just experimental error. There must be 50 things they can express: how broadly something applies, how you know it, how happy you are it's so, even how it could be falsified. I'm not going to try to explore the structure of qualification here. It's probably more complex than the whole topic of writing usefully. Instead I'll just give you a practical tip: Don't underestimate qualification. It's an important skill in its own right, not just a sort of tax you have to pay in order to avoid saying things that are false. So learn and use its full range. It may not be fully half of having good ideas, but it's part of having them.

There's one other quality I aim for in essays: to say things as simply as possible. But I don't think this is a component of usefulness. It's more a matter of consideration for the reader. And it's a practical aid in getting things right; a mistake is more obvious when expressed in simple language. But I'll admit that the main reason I write simply is not for the reader's sake or because it helps get things right, but because it bothers me to use more or fancier words than I need to. It seems inelegant, like a program that's too long.

I realize florid writing works for some people. But unless you're sure you're one of them, the best advice is to write as simply as you can.

---

I believe the formula I've given you, importance + novelty + correctness + strength, is the recipe for a good essay. But I should warn you that it's also a recipe for making people mad.

The root of the problem is novelty. When you tell people something they didn't know, they don't always thank you for it. Sometimes the reason people don't know something is because they don't want to know it. Usually because it contradicts some cherished belief. And indeed, if you're looking for novel ideas, popular but mistaken beliefs are a good place to find them. Every popular mistaken belief creates a [dead zone](#) of ideas around it that are relatively unexplored because they contradict it.

The strength component just makes things worse. If there's anything that annoys people more than having their cherished assumptions contradicted, it's having them flatly contradicted.

Plus if you've used the Morris technique, your writing will seem quite confident. Perhaps offensively confident, to people who disagree with you. The reason you'll seem confident is that you are confident: you've cheated, by only publishing the things you're sure of. It will seem to people who try to disagree with you that you never admit you're wrong. In fact you constantly admit you're wrong. You just do it

before publishing instead of after.

And if your writing is as simple as possible, that just makes things worse. Brevity is the diction of command. If you watch someone delivering unwelcome news from a position of inferiority, you'll notice they tend to use lots of words, to soften the blow. Whereas to be short with someone is more or less to be rude to them.

It can sometimes work to deliberately phrase statements more weakly than you mean. To put "perhaps" in front of something you're actually quite sure of. But you'll notice that when writers do this, they usually do it with a wink.

I don't like to do this too much. It's cheesy to adopt an ironic tone for a whole essay. I think we just have to face the fact that elegance and curtness are two names for the same thing.

You might think that if you work sufficiently hard to ensure that an essay is correct, it will be invulnerable to attack. That's sort of true. It will be invulnerable to valid attacks. But in practice that's little consolation.

In fact, the strength component of useful writing will make you particularly vulnerable to misrepresentation. If you've stated an idea as strongly as you could without making it false, all anyone has to do is to exaggerate slightly what you said, and now it is false.

Much of the time they're not even doing it deliberately. One of the most surprising things you'll discover, if you start writing essays, is that people who disagree with you rarely disagree with what you've actually written. Instead they make up something you said and disagree with that.

For what it's worth, the countermove is to ask someone who does this to quote a specific sentence or passage you wrote that they believe is false, and explain why. I say "for what it's worth" because they never do. So although it might seem that this could get a broken discussion back on track, the truth is that it was never on track in the first place.

Should you explicitly forestall likely misinterpretations? Yes, if they're misinterpretations a reasonably smart and well-intentioned person might make. In fact it's sometimes better to say something slightly misleading and then add the correction than to try to get an idea right in one shot. That can be more efficient, and can also model the way such an idea would be discovered.

But I don't think you should explicitly forestall intentional misinterpretations in the body of an essay. An essay is a place to meet honest readers. You don't want to spoil your house by putting bars on the windows to protect against dishonest ones. The place to protect against intentional misinterpretations is in end-notes. But don't think you can predict them all. People are as ingenious at misrepresenting you when you say something they don't want to hear as they are at coming up with rationalizations for things they want to do but know they shouldn't. I suspect

it's the same skill.

---

As with most other things, the way to get better at writing essays is to practice. But how do you start? Now that we've examined the structure of useful writing, we can rephrase that question more precisely. Which constraint do you relax initially? The answer is, the first component of importance: the number of people who care about what you write.

If you narrow the topic sufficiently, you can probably find something you're an expert on. Write about that to start with. If you only have ten readers who care, that's fine. You're helping them, and you're writing. Later you can expand the breadth of topics you write about.

The other constraint you can relax is a little surprising: publication. Writing essays doesn't have to mean publishing them. That may seem strange now that the trend is to publish every random thought, but it worked for me. I wrote what amounted to essays in notebooks for about 15 years. I never published any of them and never expected to. I wrote them as a way of figuring things out. But when the web came along I'd had a lot of practice.

Incidentally, [Steve Wozniak](#) did the same thing. In high school he designed computers on paper for fun. He couldn't build them because he couldn't afford the components. But when Intel launched 4K DRAMs in 1975, he was ready.

---

How many essays are there left to write though? The answer to that question is probably the most exciting thing I've learned about essay writing. Nearly all of them are left to write.

Although [the essay](#) is an old form, it hasn't been assiduously cultivated. In the print era, publication was expensive, and there wasn't enough demand for essays to publish that many. You could publish essays if you were already well known for writing something else, like novels. Or you could write book reviews that you took over to express your own ideas. But there was not really a direct path to becoming an essayist. Which meant few essays got written, and those that did tended to be about a narrow range of subjects.

Now, thanks to the internet, there's a path. Anyone can publish essays online. You start in obscurity, perhaps, but at least you can start. You don't need anyone's permission.

It sometimes happens that an area of knowledge sits quietly for years, till some change makes it explode. Cryptography did this to number theory. The internet is

doing it to the essay.

The exciting thing is not that there's a lot left to write, but that there's a lot left to discover. There's a certain kind of idea that's best discovered by writing essays. If most essays are still unwritten, most such ideas are still undiscovered.

## Notes

[1] Put railings on the balconies, but don't put bars on the windows.

[2] Even now I sometimes write essays that are not meant for publication. I wrote several to figure out what Y Combinator should do, and they were really helpful.

**Thanks** to Trevor Blackwell, Daniel Gackle, Jessica Livingston, and Robert Morris for reading drafts of this.

[Spanish Translation](#)

[Japanese Translation](#)

# Coronavirus and Credibility

April 2020

I recently saw a [video](#) of TV journalists and politicians confidently saying that the coronavirus would be no worse than the flu. What struck me about it was not just how mistaken they seemed, but how daring. How could they feel safe saying such things?

The answer, I realized, is that they didn't think they could get caught. They didn't realize there was any danger in making false predictions. These people constantly make false predictions, and get away with it, because the things they make predictions about either have mushy enough outcomes that they can bluster their way out of trouble, or happen so far in the future that few remember what they said.

An epidemic is different. It falsifies your predictions rapidly and unequivocally.

But epidemics are rare enough that these people clearly didn't realize this was even a possibility. Instead they just continued to use their ordinary m.o., which, as the epidemic has made clear, is to talk confidently about things they don't understand.

An event like this is thus a uniquely powerful way of taking people's measure. As Warren Buffett said, "It's only when the tide goes out that you learn who's been swimming naked." And the tide has just gone out like never before.

Now that we've seen the results, let's remember what we saw, because this is the most accurate test of credibility we're ever likely to have. I hope.

[Finnish Translation](#)

[German Translation](#)

[French Translation](#)





# Orthodox Privilege

July 2020

"Few people are capable of expressing with equanimity opinions which differ from the prejudices of their social environment. Most people are even incapable of forming such opinions."

◆ Einstein

There has been a lot of talk about privilege lately. Although the concept is overused, there is something to it, and in particular to the idea that privilege makes you blind ◆ that you can't see things that are visible to someone whose life is very different from yours.

But one of the most pervasive examples of this kind of blindness is one that I haven't seen mentioned explicitly. I'm going to call it *orthodox privilege*: The more conventional-minded someone is, the more it seems to them that it's safe for everyone to express their opinions.

It's safe for *them* to express their opinions, because the source of their opinions is whatever it's currently acceptable to believe. So it seems to them that it must be safe for everyone. They literally can't imagine a true statement that would get you in trouble.

And yet at every point in history, there [were](#) true things that would get you in trouble to say. Is ours the first where this isn't so? What an amazing coincidence that would be.

Surely it should at least be the default assumption that our time is not unique, and that there are true things you can't say now, just as there have always been. You would think. But even in the face of such overwhelming historical evidence, most people will go with their gut on this one.

In the most extreme cases, people suffering from orthodox privilege will not only deny that there's anything true that you can't say, but will accuse you of heresy merely for saying there is. Though if there's more than one heresy current in your time, these accusations will be weirdly non-deterministic: you must either be an xist or a yist.

Frustrating as it is to deal with these people, it's important to realize that they're in earnest. They're not pretending they think it's impossible for an idea to be both unorthodox and true. The world really looks that way to them.

Indeed, this is a uniquely tenacious form of privilege. People can overcome the blindness induced by most forms of privilege by learning more about whatever they're not. But they can't overcome orthodox privilege just by learning more. They'd have to become more independent-minded. If that happens at all, it doesn't happen on the time scale of one conversation.

It may be possible to convince some people that orthodox privilege must exist even though they can't sense it, just as one can with, say, dark matter. There may be some who could be convinced, for example, that it's very unlikely that this is the first point in history at which there's nothing true you can't say, even if they can't imagine specific examples.

But in general I don't think it will work to say "check your privilege" about this type of privilege, because those in its demographic don't realize they're in it. It doesn't seem to conventional-minded people that they're conventional-minded. It just seems to them that they're right. Indeed, they tend to be particularly sure of it.

Perhaps the solution is to appeal to politeness. If someone says they can hear a high-pitched noise that you can't, it's only polite to take them at their word, instead of demanding evidence that's impossible to produce, or simply denying that they hear anything. Imagine how rude that would seem. Similarly, if someone says they can think of things that are true but that cannot be said, it's only polite to take them at their word, even if you can't think of any yourself.

**Thanks** to Sam Altman, Trevor Blackwell, Patrick Collison, Antonio Garcia-Martinez, Jessica Livingston, Robert Morris, Michael Nielsen, Geoff Ralston, Max Roser, and Harj Taggar for reading drafts of this.

# The Four Quadrants of Conformism

July 2020

One of the most revealing ways to classify people is by the degree and aggressiveness of their conformism. Imagine a Cartesian coordinate system whose horizontal axis runs from conventional-minded on the left to independent-minded on the right, and whose vertical axis runs from passive at the bottom to aggressive at the top. The resulting four quadrants define four types of people. Starting in the upper left and going counter-clockwise: aggressively conventional-minded, passively conventional-minded, passively independent-minded, and aggressively independent-minded.

I think that you'll find all four types in most societies, and that which quadrant people fall into depends more on their own personality than the beliefs prevalent in their society. [\[1\]](#)

Young children offer some of the best evidence for both points. Anyone who's been to primary school has seen the four types, and the fact that school rules are so arbitrary is strong evidence that which quadrant people fall into depends more on them than the rules.

The kids in the upper left quadrant, the aggressively conventional-minded ones, are the tattletales. They believe not only that rules must be obeyed, but that those who disobey them must be punished.

The kids in the lower left quadrant, the passively conventional-minded, are the sheep. They're careful to obey the rules, but when other kids break them, their impulse is to worry that those kids will be punished, not to ensure that they will.

The kids in the lower right quadrant, the passively independent-minded, are the dreamy ones. They don't care much about rules and probably aren't 100% sure what the rules even are.

And the kids in the upper right quadrant, the aggressively independent-minded, are the naughty ones. When they see a rule, their first impulse is to question it. Merely being told what to do makes them inclined to do the opposite.

When measuring conformism, of course, you have to say with respect to what, and this changes as kids get older. For younger kids it's the rules set by adults. But as

kids get older, the source of rules becomes their peers. So a pack of teenagers who all flout school rules in the same way are not independent-minded; rather the opposite.

In adulthood we can recognize the four types by their distinctive calls, much as you could recognize four species of birds. The call of the aggressively conventional-minded is "Crush <outgroup>!" (It's rather alarming to see an exclamation point after a variable, but that's the whole problem with the aggressively conventional-minded.) The call of the passively conventional-minded is "What will the neighbors think?" The call of the passively independent-minded is "To each his own." And the call of the aggressively independent-minded is "Eppur si muove."

The four types are not equally common. There are more passive people than aggressive ones, and far more conventional-minded people than independent-minded ones. So the passively conventional-minded are the largest group, and the aggressively independent-minded the smallest.

Since one's quadrant depends more on one's personality than the nature of the rules, most people would occupy the same quadrant even if they'd grown up in a quite different society.

Princeton professor Robert George recently wrote:

I sometimes ask students what their position on slavery would have been had they been white and living in the South before abolition. Guess what? They all would have been abolitionists! They all would have bravely spoken out against slavery, and worked tirelessly against it.

He's too polite to say so, but of course they wouldn't. And indeed, our default assumption should not merely be that his students would, on average, have behaved the same way people did at the time, but that the ones who are aggressively conventional-minded today would have been aggressively conventional-minded then too. In other words, that they'd not only not have fought against slavery, but that they'd have been among its staunchest defenders.

I'm biased, I admit, but it seems to me that aggressively conventional-minded people are responsible for a disproportionate amount of the trouble in the world, and that a lot of the customs we've evolved since the Enlightenment have been designed to protect the rest of us from them. In particular, the retirement of the concept of heresy and its replacement by the principle of freely debating all sorts of different ideas, even ones that are currently considered unacceptable, without any punishment for those who try them out to see if they work. [2]

Why do the independent-minded need to be protected, though? Because they have all the new ideas. To be a successful scientist, for example, it's not enough just to be right. You have to be right when everyone else is wrong. Conventional-minded people can't do that. For similar reasons, all successful startup CEOs are not merely independent-minded, but aggressively so. So it's no coincidence that

societies prosper only to the extent that they have customs for keeping the conventional-minded at bay. [3]

In the last few years, many of us have noticed that the customs protecting free inquiry have been weakened. Some say we're overreacting ♦ that they haven't been weakened very much, or that they've been weakened in the service of a greater good. The latter I'll dispose of immediately. When the conventional-minded get the upper hand, they always say it's in the service of a greater good. It just happens to be a different, incompatible greater good each time.

As for the former worry, that the independent-minded are being oversensitive, and that free inquiry hasn't been shut down that much, you can't judge that unless you are yourself independent-minded. You can't know how much of the space of ideas is being lopped off unless you have them, and only the independent-minded have the ones at the edges. Precisely because of this, they tend to be very sensitive to changes in how freely one can explore ideas. They're the canaries in this coalmine.

The conventional-minded say, as they always do, that they don't want to shut down the discussion of all ideas, just the bad ones.

You'd think it would be obvious just from that sentence what a dangerous game they're playing. But I'll spell it out. There are two reasons why we need to be able to discuss even "bad" ideas.

The first is that any process for deciding which ideas to ban is bound to make mistakes. All the more so because no one intelligent wants to undertake that kind of work, so it ends up being done by the stupid. And when a process makes a lot of mistakes, you need to leave a margin for error. Which in this case means you need to ban fewer ideas than you'd like to. But that's hard for the aggressively conventional-minded to do, partly because they enjoy seeing people punished, as they have since they were children, and partly because they compete with one another. Enforcers of orthodoxy can't allow a borderline idea to exist, because that gives other enforcers an opportunity to one-up them in the moral purity department, and perhaps even to turn enforcer upon them. So instead of getting the margin for error we need, we get the opposite: a race to the bottom in which any idea that seems at all bannable ends up being banned. [4]

The second reason it's dangerous to ban the discussion of ideas is that ideas are more closely related than they look. Which means if you restrict the discussion of some topics, it doesn't only affect those topics. The restrictions propagate back into any topic that yields implications in the forbidden ones. And that is not an edge case. The best ideas do exactly that: they have consequences in fields far removed from their origins. Having ideas in a world where some ideas are banned is like playing soccer on a pitch that has a minefield in one corner. You don't just play the same game you would have, but on a different shaped pitch. You play a much more subdued game even on the ground that's safe.

In the past, the way the independent-minded protected themselves was to

congregate in a handful of places ♦ first in courts, and later in universities ♦ where they could to some extent make their own rules. Places where people work with ideas tend to have customs protecting free inquiry, for the same reason wafer fabs have powerful air filters, or recording studios good sound insulation. For the last couple centuries at least, when the aggressively conventional-minded were on the rampage for whatever reason, universities were the safest places to be.

That may not work this time though, due to the unfortunate fact that the latest wave of intolerance began in universities. It began in the mid 1980s, and by 2000 seemed to have died down, but it has recently flared up again with the arrival of social media. This seems, unfortunately, to have been an own goal by Silicon Valley. Though the people who run Silicon Valley are almost all independent-minded, they've handed the aggressively conventional-minded a tool such as they could only have dreamed of.

On the other hand, perhaps the decline in the spirit of free inquiry within universities is as much the symptom of the departure of the independent-minded as the cause. People who would have become professors 50 years ago have other options now. Now they can become quants or start startups. You have to be independent-minded to succeed at either of those. If these people had been professors, they'd have put up a stiffer resistance on behalf of academic freedom. So perhaps the picture of the independent-minded fleeing declining universities is too gloomy. Perhaps the universities are declining because so many have already left. [5]

Though I've spent a lot of time thinking about this situation, I can't predict how it plays out. Could some universities reverse the current trend and remain places where the independent-minded want to congregate? Or will the independent-minded gradually abandon them? I worry a lot about what we might lose if that happened.

But I'm hopeful long term. The independent-minded are good at protecting themselves. If existing institutions are compromised, they'll create new ones. That may require some imagination. But imagination is, after all, their specialty.

[1] I realize of course that if people's personalities vary in any two ways, you can use them as axes and call the resulting four quadrants personality types. So what I'm really claiming is that the axes are orthogonal and that there's significant variation in both.

[2] The aggressively conventional-minded aren't responsible for all the trouble in the world. Another big source of trouble is the sort of charismatic leader who gains power by appealing to them. They become much more dangerous when such leaders emerge.

[3] I never worried about writing things that offended the conventional-minded when I was running Y Combinator. If YC were a cookie company, I'd have faced a difficult moral choice. Conventional-minded people eat cookies too. But they don't start successful startups. So if I deterred them from applying to YC, the only effect was to save us work reading applications.

[4] There has been progress in one area: the punishments for talking about banned ideas are less severe than in the past. There's little danger of being killed, at least in richer countries. The aggressively conventional-minded are mostly satisfied with getting people fired.

[5] Many professors are independent-minded ♦ especially in math, the hard sciences, and engineering, where you have to be to succeed. But students are more representative of the general population, and thus mostly conventional-minded. So when professors and students are in conflict, it's not just a conflict between generations but also between different types of people.

**Thanks** to Sam Altman, Trevor Blackwell, Nicholas Christakis, Patrick Collison, Sam Gichuru, Jessica Livingston, Patrick McKenzie, Geoff Ralston, and Harj Taggar for reading drafts of this.

[German Translation](#)

[Korean Translation](#)

[Serbian Translation](#)

# Modeling a Wealth Tax

August 2020

Some politicians are proposing to introduce wealth taxes in addition to income and capital gains taxes. Let's try modeling the effects of various levels of wealth tax to see what they would mean in practice for a startup founder.

Suppose you start a successful startup in your twenties, and then live for another 60 years. How much of your stock will a wealth tax consume?

If the wealth tax applies to all your assets, it's easy to calculate its effect. A wealth tax of 1% means you get to keep 99% of your stock each year. After 60 years the proportion of stock you'll have left will be  $.99^{60}$ , or .547. So a straight 1% wealth tax means the government will over the course of your life take 45% of your stock.

(Losing shares does not, obviously, mean becoming *net* poorer unless the value per share is increasing by less than the wealth tax rate.)

Here's how much stock the government would take over 60 years at various levels of wealth tax:

wealth tax
government takes
0.1%
6%
0.5%
26%
1.0%
45%
2.0%
70%
3.0%
84%
4.0%
91%
5.0%
95%

A wealth tax will usually have a threshold at which it starts. How much difference would a high threshold make? To model that, we need to make some assumptions about the initial value of your stock and the growth rate.



Suppose your stock is initially worth \$2 million, and the company's trajectory is as follows: the value of your stock grows 3x for 2 years, then 2x for 2 years, then 50% for 2 years, after which you just get a typical public company growth rate, which we'll call 8%. [1] Suppose the wealth tax threshold is \$50 million. How much stock does the government take now?

wealth tax
government takes
0.1%
5%
0.5%
23%
1.0%
41%
2.0%
65%
3.0%
79%
4.0%
88%
5.0%
93%

It may at first seem surprising that such apparently small tax rates produce such dramatic effects. A 2% wealth tax with a \$50 million threshold takes about two thirds of a successful founder's stock.

The reason wealth taxes have such dramatic effects is that they're applied over and over to the same money. Income tax happens every year, but only to that year's income. Whereas if you live for 60 years after acquiring some asset, a wealth tax will tax that same asset 60 times. A wealth tax compounds.

**Note**

[1] In practice, eventually some of this 8% would come in the form of dividends, which are taxed as income at issue, so this model actually represents the most optimistic case for the founder.

# Early Work

October 2020

One of the biggest things holding people back from doing great work is the fear of making something lame. And this fear is not an irrational one. Many great projects go through a stage early on where they don't seem very impressive, even to their creators. You have to push through this stage to reach the great work that lies beyond. But many people don't. Most people don't even reach the stage of making something they're embarrassed by, let alone continue past it. They're too frightened even to start.

Imagine if we could turn off the fear of making something lame. Imagine how much more we'd do.

Is there any hope of turning it off? I think so. I think the habits at work here are not very deeply rooted.

Making new things is itself a new thing for us as a species. It has always happened, but till the last few centuries it happened so slowly as to be invisible to individual humans. And since we didn't need customs for dealing with new ideas, we didn't develop any.

We just don't have enough experience with early versions of ambitious projects to know how to respond to them. We judge them as we would judge more finished work, or less ambitious projects. We don't realize they're a special case.

Or at least, most of us don't. One reason I'm confident we can do better is that it's already starting to happen. There are already a few places that are living in the future in this respect. Silicon Valley is one of them: an unknown person working on a strange-sounding idea won't automatically be dismissed the way they would back home. In Silicon Valley, people have learned how dangerous that is.

The right way to deal with new ideas is to treat them as a challenge to your imagination ♦ not just to have lower standards, but to [switch polarity](#) entirely, from listing the reasons an idea won't work to trying to think of ways it could. That's what I do when I meet people with new ideas. I've become quite good at it, but I've had a lot of practice. Being a partner at Y Combinator means being practically immersed in strange-sounding ideas proposed by unknown people. Every six months you get thousands of new ones thrown at you and have to sort

through them, knowing that in a world with a power-law distribution of outcomes, it will be painfully obvious if you miss the needle in this haystack. Optimism becomes urgent.

But I'm hopeful that, with time, this kind of optimism can become widespread enough that it becomes a social custom, not just a trick used by a few specialists. It is after all an extremely lucrative trick, and those tend to spread quickly.

Of course, inexperience is not the only reason people are too harsh on early versions of ambitious projects. They also do it to seem clever. And in a field where the new ideas are risky, like startups, those who dismiss them are in fact more likely to be right. Just not when their predictions are [weighted by outcome](#).

But there is another more sinister reason people dismiss new ideas. If you try something ambitious, many of those around you will hope, consciously or unconsciously, that you'll fail. They worry that if you try something ambitious and succeed, it will put you above them. In some countries this is not just an individual failing but part of the national culture.

I wouldn't claim that people in Silicon Valley overcome these impulses because they're morally better. [\[1\]](#) The reason many hope you'll succeed is that they hope to rise with you. For investors this incentive is particularly explicit. They want you to succeed because they hope you'll make them rich in the process. But many other people you meet can hope to benefit in some way from your success. At the very least they'll be able to say, when you're famous, that they've known you since way back.

But even if Silicon Valley's encouraging attitude is rooted in self-interest, it has over time actually grown into a sort of benevolence. Encouraging startups has been practiced for so long that it has become a custom. Now it just seems that that's what one does with startups.

Maybe Silicon Valley is too optimistic. Maybe it's too easily fooled by impostors. Many less optimistic journalists want to believe that. But the lists of impostors they cite are suspiciously short, and plagued with asterisks. [\[2\]](#) If you use revenue as the test, Silicon Valley's optimism seems better tuned than the rest of the world's. And because it works, it will spread.

There's a lot more to new ideas than new startup ideas, of course. The fear of making something lame holds people back in every field. But Silicon Valley shows how quickly customs can evolve to support new ideas. And that in turn proves that dismissing new ideas is not so deeply rooted in human nature that it can't be unlearned.

---

Unfortunately, if you want to do new things, you'll face a force more powerful than

other people's skepticism: your own skepticism. You too will judge your early work too harshly. How do you avoid that?

This is a difficult problem, because you don't want to completely eliminate your horror of making something lame. That's what steers you toward doing good work. You just want to turn it off temporarily, the way a painkiller temporarily turns off pain.

People have already discovered several techniques that work. Hardy mentions two in *A Mathematician's Apology*:

Good work is not done by "humble" men. It is one of the first duties of a professor, for example, in any subject, to exaggerate a little both the importance of his subject and his importance in it.

If you overestimate the importance of what you're working on, that will compensate for your mistakenly harsh judgment of your initial results. If you look at something that's 20% of the way to a goal worth 100 and conclude that it's 10% of the way to a goal worth 200, your estimate of its expected value is correct even though both components are wrong.

It also helps, as Hardy suggests, to be slightly overconfident. I've noticed in many fields that the most successful people are slightly overconfident. On the face of it this seems implausible. Surely it would be optimal to have exactly the right estimate of one's abilities. How could it be an advantage to be mistaken? Because this error compensates for other sources of error in the opposite direction: being slightly overconfident armors you against both other people's skepticism and your own.

Ignorance has a similar effect. It's safe to make the mistake of judging early work as finished work if you're a sufficiently lax judge of finished work. I doubt it's possible to cultivate this kind of ignorance, but empirically it's a real advantage, especially for the young.

Another way to get through the lame phase of ambitious projects is to surround yourself with the right people ♦ to create an eddy in the social headwind. But it's not enough to collect people who are always encouraging. You'd learn to discount that. You need colleagues who can actually tell an ugly duckling from a baby swan. The people best able to do this are those working on similar projects of their own, which is why university departments and research labs work so well. You don't need institutions to collect colleagues. They naturally coalesce, given the chance. But it's very much worth accelerating this process by seeking out other people trying to do new things.

Teachers are in effect a special case of colleagues. It's a teacher's job both to see the promise of early work and to encourage you to continue. But teachers who are good at this are unfortunately quite rare, so if you have the opportunity to learn from one, take it. [3]

For some it might work to rely on sheer discipline: to tell yourself that you just have to press on through the initial crap phase and not get discouraged. But like a lot of "just tell yourself" advice, this is harder than it sounds. And it gets still harder as you get older, because your standards rise. The old do have one compensating advantage though: they've been through this before.

It can help if you focus less on where you are and more on the rate of change. You won't worry so much about doing bad work if you can see it improving. Obviously the faster it improves, the easier this is. So when you start something new, it's good if you can spend a lot of time on it. That's another advantage of being young: you tend to have bigger blocks of time.

Another common trick is to start by considering new work to be of a different, less exacting type. To start a painting saying that it's just a sketch, or a new piece of software saying that it's just a quick hack. Then you judge your initial results by a lower standard. Once the project is rolling you can sneakily convert it to something more. [\[4\]](#)

This will be easier if you use a medium that lets you work fast and doesn't require too much commitment up front. It's easier to convince yourself that something is just a sketch when you're drawing in a notebook than when you're carving stone. Plus you get initial results faster. [\[5\]](#) [\[6\]](#)

It will be easier to try out a risky project if you think of it as a way to learn and not just as a way to make something. Then even if the project truly is a failure, you'll still have gained by it. If the problem is sharply enough defined, failure itself is knowledge: if the theorem you're trying to prove turns out to be false, or you use a structural member of a certain size and it fails under stress, you've learned something, even if it isn't what you wanted to learn. [\[7\]](#)

One motivation that works particularly well for me is curiosity. I like to try new things just to see how they'll turn out. We started Y Combinator in this spirit, and it was one of main things that kept me going while I was working on [Bel](#). Having worked for so long with various dialects of Lisp, I was very curious to see what its inherent shape was: what you'd end up with if you followed the axiomatic approach all the way.

But it's a bit strange that you have to play mind games with yourself to avoid being discouraged by lame-looking early efforts. The thing you're trying to trick yourself into believing is in fact the truth. A lame-looking early version of an ambitious project truly is more valuable than it seems. So the ultimate solution may be to teach yourself that.

One way to do it is to study the histories of people who've done great work. What were they thinking early on? What was the very first thing they did? It can sometimes be hard to get an accurate answer to this question, because people are often embarrassed by their earliest work and make little effort to publish it. (They

too misjudge it.) But when you can get an accurate picture of the first steps someone made on the path to some great work, they're often pretty feeble. [8]

Perhaps if you study enough such cases, you can teach yourself to be a better judge of early work. Then you'll be immune both to other people's skepticism and your own fear of making something lame. You'll see early work for what it is.

Curiously enough, the solution to the problem of judging early work too harshly is to realize that our attitudes toward it are themselves early work. Holding everything to the same standard is a crude version 1. We're already evolving better customs, and we can already see signs of how big the payoff will be.

## Notes

[1] This assumption may be too conservative. There is some evidence that historically the Bay Area has attracted a [different sort of person](#) than, say, New York City.

[2] One of their great favorites is Theranos. But the most conspicuous feature of Theranos's cap table is the absence of Silicon Valley firms. Journalists were fooled by Theranos, but Silicon Valley investors weren't.

[3] I made two mistakes about teachers when I was younger. I cared more about professors' research than their reputations as teachers, and I was also wrong about what it meant to be a good teacher. I thought it simply meant to be good at explaining things.

[4] Patrick Collison points out that you can go past treating something as a hack in the sense of a prototype and onward to the sense of the word that means something closer to a practical joke:

I think there may be something related to being a hack that can be powerful ♦ the idea of making the tenuousness and implausibility a *feature*. "Yes, it's a bit ridiculous, right? I'm just trying to see how far such a naive approach can get." YC seemed to me to have this characteristic.

[5] Much of the advantage of switching from physical to digital media is not the software per se but that it lets you start something new with little upfront commitment.

[6] John Carmack adds:

The value of a medium without a vast gulf between the early work and the final work is exemplified in game mods. The original Quake game was a golden age for mods, because everything was very flexible, but so crude due to technical limitations, that quick hacks to try out a gameplay idea weren't all *that* far from the official game. Many careers were born from that, but as the commercial game quality improved over the years, it became almost a full time job to make a successful mod that would be appreciated by the community. This was dramatically reversed with Minecraft and later Roblox, where the entire esthetic of the experience was so explicitly crude that innovative gameplay concepts became the overriding value. These "crude" game mods by single authors are now often bigger deals than massive professional teams' work.

[7] Lisa Randall suggests that we

treat new things as experiments. That way there's no such thing as failing, since you learn something no matter what. You treat it like an experiment in the sense that if it really rules something out, you give up and move on, but if there's some way to vary it to make it work better, go ahead and do that

[8] Michael Nielsen points out that the internet has made this easier, because you can see programmers' first commits, musicians' first videos, and so on.


**Thanks** to Trevor Blackwell, John Carmack, Patrick Collison, Jessica Livingston, Michael Nielsen, and Lisa Randall for reading drafts of this.

# How to Think for Yourself



November 2020

There are some kinds of work that you can't do well without thinking differently from your peers. To be a successful scientist, for example, it's not enough just to be correct. Your ideas have to be both correct and novel. You can't publish papers saying things other people already know. You need to say things no one else has realized yet.

The same is true for investors. It's not enough for a public market investor to predict correctly how a company will do. If a lot of other people make the same prediction, the stock price will already reflect it, and there's no room to make money. The only valuable insights are the ones most other investors don't share.

You see this pattern with startup founders too. You don't want to start a startup to do something that everyone agrees is a good idea, or there will already be other companies doing it. You have to do something that sounds to most other people like a bad idea, but that you know isn't  like writing software for a tiny computer used by a few thousand hobbyists, or starting a site to let people rent airbeds on strangers' floors.

Ditto for essayists. An essay that told people things they already knew would be boring. You have to tell them something [new](#).

But this pattern isn't universal. In fact, it doesn't hold for most kinds of work. In most kinds of work  to be an administrator, for example  all you need is the first half. All you need is to be right. It's not essential that everyone else be wrong.

There's room for a little novelty in most kinds of work, but in practice there's a fairly sharp distinction between the kinds of work where it's essential to be independent-minded, and the kinds where it's not.

I wish someone had told me about this distinction when I was a kid, because it's one of the most important things to think about when you're deciding what kind of work you want to do. Do you want to do the kind of work where you can only win by thinking differently from everyone else? I suspect most people's unconscious mind will answer that question before their conscious mind has a chance to. I know mine does.



Independent-mindedness seems to be more a matter of nature than nurture. Which means if you pick the wrong type of work, you're going to be unhappy. If you're naturally independent-minded, you're going to find it frustrating to be a middle manager. And if you're naturally conventional-minded, you're going to be sailing into a headwind if you try to do original research.

One difficulty here, though, is that people are often mistaken about where they fall on the spectrum from conventional- to independent-minded. Conventional-minded people don't like to think of themselves as conventional-minded. And in any case, it genuinely feels to them as if they make up their own minds about everything. It's just a coincidence that their beliefs are identical to their peers'. And the independent-minded, meanwhile, are often unaware how different their ideas are from conventional ones, at least till they state them publicly. [1]

By the time they reach adulthood, most people know roughly how smart they are (in the narrow sense of ability to solve pre-set problems), because they're constantly being tested and ranked according to it. But schools generally ignore independent-mindedness, except to the extent they try to suppress it. So we don't get anything like the same kind of feedback about how independent-minded we are.

There may even be a phenomenon like Dunning-Kruger at work, where the most conventional-minded people are confident that they're independent-minded, while the genuinely independent-minded worry they might not be independent-minded enough.

---

Can you make yourself more independent-minded? I think so. This quality may be largely inborn, but there seem to be ways to magnify it, or at least not to suppress it.

One of the most effective techniques is one practiced unintentionally by most nerds: simply to be less aware what conventional beliefs are. It's hard to be a conformist if you don't know what you're supposed to conform to. Though again, it may be that such people already are independent-minded. A conventional-minded person would probably feel anxious not knowing what other people thought, and make more effort to find out.

It matters a lot who you surround yourself with. If you're surrounded by conventional-minded people, it will constrain which ideas you can express, and that in turn will constrain which ideas you have. But if you surround yourself with independent-minded people, you'll have the opposite experience: hearing other people say surprising things will encourage you to, and to think of more.

Because the independent-minded find it uncomfortable to be surrounded by conventional-minded people, they tend to self-segregate once they have a chance

to. The problem with high school is that they haven't yet had a chance to. Plus high school tends to be an inward-looking little world whose inhabitants lack confidence, both of which magnify the forces of conformism. So high school is often a [bad time](#) for the independent-minded. But there is some advantage even here: it teaches you what to avoid. If you later find yourself in a situation that makes you think "this is like high school," you know you should get out. [2]

Another place where the independent- and conventional-minded are thrown together is in successful startups. The founders and early employees are almost always independent-minded; otherwise the startup wouldn't be successful. But conventional-minded people greatly outnumber independent-minded ones, so as the company grows, the original spirit of independent-mindedness is inevitably diluted. This causes all kinds of problems besides the obvious one that the company starts to suck. One of the strangest is that the founders find themselves able to speak more freely with founders of other companies than with their own employees. [3]

Fortunately you don't have to spend all your time with independent-minded people. It's enough to have one or two you can talk to regularly. And once you find them, they're usually as eager to talk as you are; they need you too. Although universities no longer have the kind of monopoly they used to have on education, good universities are still an excellent way to meet independent-minded people. Most students will still be conventional-minded, but you'll at least find clumps of independent-minded ones, rather than the near zero you may have found in high school.

It also works to go in the other direction: as well as cultivating a small collection of independent-minded friends, to try to meet as many different types of people as you can. It will decrease the influence of your immediate peers if you have several other groups of peers. Plus if you're part of several different worlds, you can often import ideas from one to another.

But by different types of people, I don't mean demographically different. For this technique to work, they have to think differently. So while it's an excellent idea to go and visit other countries, you can probably find people who think differently right around the corner. When I meet someone who knows a lot about something unusual (which includes practically everyone, if you dig deep enough), I try to learn what they know that other people don't. There are almost always surprises here. It's a good way to make conversation when you meet strangers, but I don't do it to make conversation. I really want to know.

You can expand the source of influences in time as well as space, by reading history. When I read history I do it not just to learn what happened, but to try to get inside the heads of people who lived in the past. How did things look to them? This is hard to do, but worth the effort for the same reason it's worth travelling far to triangulate a point.

You can also take more explicit measures to prevent yourself from automatically

adopting conventional opinions. The most general is to cultivate an attitude of skepticism. When you hear someone say something, stop and ask yourself "Is that true?" Don't say it out loud. I'm not suggesting that you impose on everyone who talks to you the burden of proving what they say, but rather that you take upon yourself the burden of evaluating what they say.

Treat it as a puzzle. You know that some accepted ideas will later turn out to be wrong. See if you can guess which. The end goal is not to find flaws in the things you're told, but to find the new ideas that had been concealed by the broken ones. So this game should be an exciting quest for novelty, not a boring protocol for intellectual hygiene. And you'll be surprised, when you start asking "Is this true?", how often the answer is not an immediate yes. If you have any imagination, you're more likely to have too many leads to follow than too few.

More generally your goal should be not to let anything into your head unexamined, and things don't always enter your head in the form of statements. Some of the most powerful influences are implicit. How do you even notice these? By standing back and watching how other people get their ideas.

When you stand back at a sufficient distance, you can see ideas spreading through groups of people like waves. The most obvious are in fashion: you notice a few people wearing a certain kind of shirt, and then more and more, until half the people around you are wearing the same shirt. You may not care much what you wear, but there are intellectual fashions too, and you definitely don't want to participate in those. Not just because you want sovereignty over your own thoughts, but because [unfashionable](#) ideas are disproportionately likely to lead somewhere interesting. The best place to find undiscovered ideas is where no one else is looking. [\[4\]](#)

---

To go beyond this general advice, we need to look at the internal structure of independent-mindedness [◆](#) at the individual muscles we need to exercise, as it were. It seems to me that it has three components: fastidiousness about truth, resistance to being told what to think, and curiosity.

Fastidiousness about truth means more than just not believing things that are false. It means being careful about degree of belief. For most people, degree of belief rushes unexamined toward the extremes: the unlikely becomes impossible, and the probable becomes certain. [\[5\]](#) To the independent-minded, this seems unpardonably sloppy. They're willing to have anything in their heads, from highly speculative hypotheses to (apparent) tautologies, but on subjects they care about, everything has to be labelled with a carefully considered degree of belief. [\[6\]](#)

The independent-minded thus have a horror of ideologies, which require one to accept a whole collection of beliefs at once, and to treat them as articles of faith. To an independent-minded person that would seem revolting, just as it would seem

to someone fastidious about food to take a bite of a submarine sandwich filled with a large variety of ingredients of indeterminate age and provenance.

Without this fastidiousness about truth, you can't be truly independent-minded. It's not enough just to have resistance to being told what to think. Those kind of people reject conventional ideas only to replace them with the most random conspiracy theories. And since these conspiracy theories have often been manufactured to capture them, they end up being less independent-minded than ordinary people, because they're subject to a much more exacting master than mere convention. [7]

Can you increase your fastidiousness about truth? I would think so. In my experience, merely thinking about something you're fastidious about causes that fastidiousness to grow. If so, this is one of those rare virtues we can have more of merely by wanting it. And if it's like other forms of fastidiousness, it should also be possible to encourage in children. I certainly got a strong dose of it from my father. [8]

The second component of independent-mindedness, resistance to being told what to think, is the most visible of the three. But even this is often misunderstood. The big mistake people make about it is to think of it as a merely negative quality. The language we use reinforces that idea. You're *unconventional*. You *don't* care what other people think. But it's not just a kind of immunity. In the most independent-minded people, the desire not to be told what to think is a positive force. It's not mere skepticism, but an active [delight](#) in ideas that subvert the conventional wisdom, the more counterintuitive the better.

Some of the most novel ideas seemed at the time almost like practical jokes. Think how often your reaction to a novel idea is to laugh. I don't think it's because novel ideas are funny per se, but because novelty and humor share a certain kind of surprisingness. But while not identical, the two are close enough that there is a definite correlation between having a sense of humor and being independent-minded ♦ just as there is between being humorless and being conventional-minded. [9]

I don't think we can significantly increase our resistance to being told what to think. It seems the most innate of the three components of independent-mindedness; people who have this quality as adults usually showed all too visible signs of it as children. But if we can't increase our resistance to being told what to think, we can at least shore it up, by surrounding ourselves with other independent-minded people.

The third component of independent-mindedness, curiosity, may be the most interesting. To the extent that we can give a brief answer to the question of where novel ideas come from, it's curiosity. That's what people are usually feeling before having them.

In my experience, independent-mindedness and curiosity predict one another

perfectly. Everyone I know who's independent-minded is deeply curious, and everyone I know who's conventional-minded isn't. Except, curiously, children. All small children are curious. Perhaps the reason is that even the conventional-minded have to be curious in the beginning, in order to learn what the conventions are. Whereas the independent-minded are the gluttons of curiosity, who keep eating even after they're full. [\[10\]](#)

The three components of independent-mindedness work in concert: fastidiousness about truth and resistance to being told what to think leave space in your brain, and curiosity finds new ideas to fill it.

Interestingly, the three components can substitute for one another in much the same way muscles can. If you're sufficiently fastidious about truth, you don't need to be as resistant to being told what to think, because fastidiousness alone will create sufficient gaps in your knowledge. And either one can compensate for curiosity, because if you create enough space in your brain, your discomfort at the resulting vacuum will add force to your curiosity. Or curiosity can compensate for them: if you're sufficiently curious, you don't need to clear space in your brain, because the new ideas you discover will push out the conventional ones you acquired by default.

Because the components of independent-mindedness are so interchangeable, you can have them to varying degrees and still get the same result. So there is not just a single model of independent-mindedness. Some independent-minded people are openly subversive, and others are quietly curious. They all know the secret handshake though.

Is there a way to cultivate curiosity? To start with, you want to avoid situations that suppress it. How much does the work you're currently doing engage your curiosity? If the answer is "not much," maybe you should change something.

The most important active step you can take to cultivate your curiosity is probably to seek out the topics that engage it. Few adults are equally curious about everything, and it doesn't seem as if you can choose which topics interest you. So it's up to you to [find](#) them. Or invent them, if necessary.

Another way to increase your curiosity is to indulge it, by investigating things you're interested in. Curiosity is unlike most other appetites in this respect: indulging it tends to increase rather than to sate it. Questions lead to more questions.

Curiosity seems to be more individual than fastidiousness about truth or resistance to being told what to think. To the degree people have the latter two, they're usually pretty general, whereas different people can be curious about very different things. So perhaps curiosity is the compass here. Perhaps, if your goal is to discover novel ideas, your motto should not be "do what you love" so much as "do what you're curious about."

## Notes

[1] One convenient consequence of the fact that no one identifies as conventional-minded is that you can say what you like about conventional-minded people without getting in too much trouble. When I wrote ["The Four Quadrants of Conformism"](#) I expected a firestorm of rage from the aggressively conventional-minded, but in fact it was quite muted. They sensed that there was something about the essay that they disliked intensely, but they had a hard time finding a specific passage to pin it on.

[2] When I ask myself what in my life is like high school, the answer is Twitter. It's not just full of conventional-minded people, as anything its size will inevitably be, but subject to violent storms of conventional-mindedness that remind me of descriptions of Jupiter. But while it probably is a net loss to spend time there, it has at least made me think more about the distinction between independent- and conventional-mindedness, which I probably wouldn't have done otherwise.

[3] The decrease in independent-mindedness in growing startups is still an open problem, but there may be solutions.

Founders can delay the problem by making a conscious effort only to hire independent-minded people. Which of course also has the ancillary benefit that they have better ideas.

Another possible solution is to create policies that somehow disrupt the force of conformism, much as control rods slow chain reactions, so that the conventional-minded aren't as dangerous. The physical separation of Lockheed's Skunk Works may have had this as a side benefit. Recent examples suggest employee forums like Slack may not be an unmitigated good.

The most radical solution would be to grow revenues without growing the company. You think hiring that junior PR person will be cheap, compared to a programmer, but what will be the effect on the average level of independent-mindedness in your company? (The growth in staff relative to faculty seems to have had a similar effect on universities.) Perhaps the rule about outsourcing work that's not your "core competency" should be augmented by one about outsourcing work done by people who'd ruin your culture as employees.

Some investment firms already seem to be able to grow revenues without growing the number of employees. Automation plus the ever increasing articulation of the "tech stack" suggest this may one day be possible for product companies.

[4] There are intellectual fashions in every field, but their influence varies. One of the reasons politics, for example, tends to be boring is that it's so extremely subject to them. The threshold for having opinions about politics is much [lower](#) than the one for having opinions about set theory. So while there are some ideas in politics, in practice they tend to be swamped by waves of intellectual fashion.

[5] The conventional-minded are often fooled by the strength of their opinions into believing that they're independent-minded. But strong convictions are not a sign of independent-mindedness. Rather the opposite.

[6] Fastidiousness about truth doesn't imply that an independent-minded person won't be dishonest, but that he won't be deluded. It's sort of like the definition of a gentleman as someone who is never unintentionally rude.

[7] You see this especially among political extremists. They think themselves nonconformists, but actually they're niche conformists. Their opinions may be different from the average person's, but they are often more influenced by their peers' opinions than the average person's are.

[8] If we broaden the concept of fastidiousness about truth so that it excludes pandering, bogusness, and pomposity as well as falsehood in the strict sense, our model of independent-mindedness can expand further into the arts.

[9] This correlation is far from perfect, though. Gödel and Dirac don't seem to have been very strong in the humor department. But someone who is both "neurotypical" and humorless is very likely to be conventional-minded.

[10] Exception: gossip. Almost everyone is curious about gossip.

**Thanks** to Trevor Blackwell, Paul Buchheit, Patrick Collison, Jessica Livingston, Robert Morris, Harj Taggar, and Peter Thiel for reading drafts of this.

[Italian Translation](#)

# The Airbnbs

December 2020

To celebrate Airbnb's IPO and to help future founders, I thought it might be useful to explain what was special about Airbnb.

What was special about the Airbnbs was how earnest they were. They did nothing half-way, and we could sense this even in the interview. Sometimes after we interviewed a startup we'd be uncertain what to do, and have to talk it over. Other times we'd just look at one another and smile. The Airbnbs' interview was that kind. We didn't even like the idea that much. Nor did users, at that stage; they had no growth. But the founders seemed so full of energy that it was impossible not to like them.

That first impression was not misleading. During the batch our nickname for Brian Chesky was The Tasmanian Devil, because like the [cartoon character](#) he seemed a tornado of energy. All three of them were like that. No one ever worked harder during YC than the Airbnbs did. When you talked to the Airbnbs, they took notes. If you suggested an idea to them in office hours, the next time you talked to them they'd not only have implemented it, but also implemented two new ideas they had in the process. "They probably have the best attitude of any startup we've funded" I wrote to Mike Arrington during the batch.

They're still like that. Jessica and I had dinner with Brian in the summer of 2018, just the three of us. By this point the company is ten years old. He took a page of notes about ideas for new things Airbnb could do.

What we didn't realize when we first met Brian and Joe and Nate was that Airbnb was on its last legs. After working on the company for a year and getting no growth, they'd agreed to give it one last shot. They'd try this Y Combinator thing, and if the company still didn't take off, they'd give up.

Any normal person would have given up already. They'd been funding the company with credit cards. They had a *binder* full of credit cards they'd maxed out. Investors didn't think much of the idea. One investor they met in a cafe walked out in the middle of meeting with them. They thought he was going to the bathroom, but he never came back. "He didn't even finish his smoothie," Brian said. And now, in late 2008, it was the worst recession in decades. The stock market was in free fall and wouldn't hit bottom for another four months.



Why hadn't they given up? This is a useful question to ask. People, like matter, reveal their nature under extreme conditions. One thing that's clear is that they weren't doing this just for the money. As a money-making scheme, this was pretty lousy: a year's work and all they had to show for it was a binder full of maxed-out credit cards. So why were they still working on this startup? Because of the experience they'd had as the first hosts.

When they first tried renting out airbeds on their floor during a design convention, all they were hoping for was to make enough money to pay their rent that month. But something surprising happened: they enjoyed having those first three guests staying with them. And the guests enjoyed it too. Both they and the guests had done it because they were in a sense forced to, and yet they'd all had a great experience. Clearly there was something new here: for hosts, a new way to make money that had literally been right under their noses, and for guests, a new way to travel that was in many ways better than hotels.

That experience was why the Airbnbs didn't give up. They knew they'd discovered something. They'd seen a glimpse of the future, and they couldn't let it go.

They knew that once people tried staying in what is now called "an airbnb," they would also realize that this was the future. But only if they tried it, and they weren't. That was the problem during Y Combinator: to get growth started.

Airbnb's goal during YC was to reach what we call [ramen profitability](#), which means making enough money that the company can pay the founders' living expenses, if they live on ramen noodles. Ramen profitability is not, obviously, the end goal of any startup, but it's the most important threshold on the way, because this is the point where you're airborne. This is the point where you no longer need investors' permission to continue existing. For the Airbnbs, ramen profitability was \$4000 a month: \$3500 for rent, and \$500 for food. They taped this goal to the mirror in the bathroom of their apartment.

The way to get growth started in something like Airbnb is to focus on the hottest subset of the market. If you can get growth started there, it will spread to the rest. When I asked the Airbnbs where there was most demand, they knew from searches: New York City. So they focused on New York. They went there [in person](#) to visit their hosts and help them make their listings more attractive. A big part of that was better pictures. So Joe and Brian rented a professional camera and took pictures of the hosts' places themselves.

This didn't just make the listings better. It also taught them about their hosts. When they came back from their first trip to New York, I asked what they'd noticed about hosts that surprised them, and they said the biggest surprise was how many of the hosts were in the same position they'd been in: they needed this money to pay their rent. This was, remember, the worst recession in decades, and it had hit New York first. It definitely added to the Airbnbs' sense of mission to feel that people needed them.

In late January 2009, about three weeks into Y Combinator, their efforts started to show results, and their numbers crept upward. But it was hard to say for sure whether it was growth or just random fluctuation. By February it was clear that it was real growth. They made \$460 in fees in the first week of February, \$897 in the second, and \$1428 in the third. That was it: they were airborne. Brian sent me an email on February 22 announcing that they were ramen profitable and giving the last three weeks' numbers.

"I assume you know what you've now set yourself up for next week," I responded.

Brian's reply was seven words: "We are not going to slow down."

# Billionaires Build

December 2020

As I was deciding what to write about next, I was surprised to find that two separate essays I'd been planning to write were actually the same.

The first is about how to ace your Y Combinator interview. There has been so much nonsense written about this topic that I've been meaning for years to write something telling founders the truth.

The second is about something politicians sometimes say 💎 that the only way to become a billionaire is by exploiting people 💎 and why this is mistaken.

Keep reading, and you'll learn both simultaneously.

I know the politicians are mistaken because it was my job to predict which people will become billionaires. I think I can truthfully say that I know as much about how to do this as anyone. If the key to becoming a billionaire 💎 the defining feature of billionaires 💎 was to exploit people, then I, as a professional billionaire scout, would surely realize this and look for people who would be good at it, just as an NFL scout looks for speed in wide receivers.

But aptitude for exploiting people is not what Y Combinator looks for at all. In fact, it's the opposite of what they look for. I'll tell you what they do look for, by explaining how to convince Y Combinator to fund you, and you can see for yourself.

What YC looks for, above all, is founders who understand some group of users and can make what they want. This is so important that it's YC's motto: "Make something people want."

A big company can to some extent force unsuitable products on unwilling customers, but a startup doesn't have the power to do that. A startup must sing for its supper, by making things that genuinely delight its customers. Otherwise it will never get off the ground.

Here's where things get difficult, both for you as a founder and for the YC partners trying to decide whether to fund you. In a market economy, it's hard to make something people want that they don't already have. That's the great thing about

market economies. If other people both knew about this need and were able to satisfy it, they already would be, and there would be no room for your startup.

Which means the conversation during your YC interview will have to be about something new: either a new need, or a new way to satisfy one. And not just new, but uncertain. If it were certain that the need existed and that you could satisfy it, that certainty would be reflected in large and rapidly growing revenues, and you wouldn't be seeking seed funding.

So the YC partners have to guess both whether you've discovered a real need, and whether you'll be able to satisfy it. That's what they are, at least in this part of their job: professional guessers. They have 1001 heuristics for doing this, and I'm not going to tell you all of them, but I'm happy to tell you the most important ones, because these can't be faked; the only way to "hack" them would be to do what you should be doing anyway as a founder.

The first thing the partners will try to figure out, usually, is whether what you're making will ever be something a lot of people want. It doesn't have to be something a lot of people want now. The product and the market will both evolve, and will influence each other's evolution. But in the end there has to be something with a huge market. That's what the partners will be trying to figure out: is there a path to a huge market? [\[1\]](#)

Sometimes it's obvious there will be a huge market. If [Boom](#) manages to ship an airliner at all, international airlines will have to buy it. But usually it's not obvious. Usually the path to a huge market is by growing a small market. This idea is important enough that it's worth coining a phrase for, so let's call one of these small but growable markets a "larval market."

The perfect example of a larval market might be Apple's market when they were founded in 1976. In 1976, not many people wanted their own computer. But more and more started to want one, till now every 10 year old on the planet wants a computer (but calls it a "phone").

The ideal combination is the group of founders who are ["living in the future"](#) in the sense of being at the leading edge of some kind of change, and who are building something they themselves want. Most super-successful startups are of this type. Steve Wozniak wanted a computer. Mark Zuckerberg wanted to engage online with his college friends. Larry and Sergey wanted to find things on the web. All these founders were building things they and their peers wanted, and the fact that they were at the leading edge of change meant that more people would want these things in the future.

But although the ideal larval market is oneself and one's peers, that's not the only kind. A larval market might also be regional, for example. You build something to serve one location, and then expand to others.

The crucial feature of the initial market is that it exist. That may seem like an

obvious point, but the lack of it is the biggest flaw in most startup ideas. There have to be some people who want what you're building right now, and want it so urgently that they're willing to use it, bugs and all, even though you're a small company they've never heard of. There don't have to be many, but there have to be some. As long as you have some users, there are straightforward ways to get more: build new features they want, seek out more people like them, get them to refer you to their friends, and so on. But these techniques all require some initial seed group of users.

So this is one thing the YC partners will almost certainly dig into during your interview. Who are your first users going to be, and how do you know they want this? If I had to decide whether to fund startups based on a single question, it would be "How do you know people want this?"

The most convincing answer is "Because we and our friends want it." It's even better when this is followed by the news that you've already built a prototype, and even though it's very crude, your friends are using it, and it's spreading by word of mouth. If you can say that and you're not lying, the partners will switch from default no to default yes. Meaning you're in unless there's some other disqualifying flaw.

That is a hard standard to meet, though. Airbnb didn't meet it. They had the first part. They had made something they themselves wanted. But it wasn't spreading. So don't feel bad if you don't hit this gold standard of convincingness. If Airbnb didn't hit it, it must be too high.

In practice, the YC partners will be satisfied if they feel that you have a deep understanding of your users' needs. And the Airbnbs did have that. They were able to tell us all about what motivated hosts and guests. They knew from first-hand experience, because they'd been the first hosts. We couldn't ask them a question they didn't know the answer to. We ourselves were not very excited about the idea as users, but we knew this didn't prove anything, because there were lots of successful startups we hadn't been excited about as users. We were able to say to ourselves "They seem to know what they're talking about. Maybe they're onto something. It's not growing yet, but maybe they can figure out how to make it grow during YC." Which they did, about three weeks into the batch.

The best thing you can do in a YC interview is to teach the partners about your users. So if you want to prepare for your interview, one of the best ways to do it is to go talk to your users and find out exactly what they're thinking. Which is what you should be doing anyway.

This may sound strangely credulous, but the YC partners want to rely on the founders to tell them about the market. Think about how VCs typically judge the potential market for an idea. They're not ordinarily domain experts themselves, so they forward the idea to someone who is, and ask for their opinion. YC doesn't have time to do this, but if the YC partners can convince themselves that the founders both (a) know what they're talking about and (b) aren't lying, they don't

need outside domain experts. They can use the founders themselves as domain experts when evaluating their own idea.

This is why YC interviews aren't pitches. To give as many founders as possible a chance to get funded, we made interviews as short as we could: 10 minutes. That is not enough time for the partners to figure out, through the indirect evidence in a pitch, whether you know what you're talking about and aren't lying. They need to dig in and ask you questions. There's not enough time for sequential access. They need random access. [2]

The worst advice I ever heard about how to succeed in a YC interview is that you should take control of the interview and make sure to deliver the message you want to. In other words, turn the interview into a pitch. *elaborate expletive*. It is so annoying when people try to do that. You ask them a question, and instead of answering it, they deliver some obviously prefabricated blob of pitch. It eats up 10 minutes really fast.

There is no one who can give you accurate advice about what to do in a YC interview except a current or former YC partner. People who've merely been interviewed, even successfully, have no idea of this, but interviews take all sorts of different forms depending on what the partners want to know about most. Sometimes they're all about the founders, other times they're all about the idea. Sometimes some very narrow aspect of the idea. Founders sometimes walk away from interviews complaining that they didn't get to explain their idea completely. True, but they explained enough.

Since a YC interview consists of questions, the way to do it well is to answer them well. Part of that is answering them candidly. The partners don't expect you to know everything. But if you don't know the answer to a question, don't try to bullshit your way out of it. The partners, like most experienced investors, are professional bullshit detectors, and you are (hopefully) an amateur bullshitter. And if you try to bullshit them and fail, they may not even tell you that you failed. So it's better to be honest than to try to sell them. If you don't know the answer to a question, say you don't, and tell them how you'd go about finding it, or tell them the answer to some related question.

If you're asked, for example, what could go wrong, the worst possible answer is "nothing." Instead of convincing them that your idea is bullet-proof, this will convince them that you're a fool or a liar. Far better to go into gruesome detail. That's what experts do when you ask what could go wrong. The partners know that your idea is risky. That's what a good bet looks like at this stage: a tiny probability of a huge outcome.

Ditto if they ask about competitors. Competitors are rarely what kills startups. Poor execution does. But you should know who your competitors are, and tell the YC partners candidly what your relative strengths and weaknesses are. Because the YC partners know that competitors don't kill startups, they won't hold competitors against you too much. They will, however, hold it against you if you seem either to

be unaware of competitors, or to be minimizing the threat they pose. They may not be sure whether you're clueless or lying, but they don't need to be.

The partners don't expect your idea to be perfect. This is seed investing. At this stage, all they can expect are promising hypotheses. But they do expect you to be thoughtful and honest. So if trying to make your idea seem perfect causes you to come off as glib or clueless, you've sacrificed something you needed for something you didn't.

If the partners are sufficiently convinced that there's a path to a big market, the next question is whether you'll be able to find it. That in turn depends on three things: the general qualities of the founders, their specific expertise in this domain, and the relationship between them. How determined are the founders? Are they good at building things? Are they resilient enough to keep going when things go wrong? How strong is their friendship?

Though the Airbnbs only did ok in the idea department, they did spectacularly well in this department. The story of how they'd funded themselves by making Obama- and McCain-themed breakfast cereal was the single most important factor in our decision to fund them. They didn't realize it at the time, but what seemed to them an irrelevant story was in fact fabulously good evidence of their qualities as founders. It showed they were resourceful and determined, and could work together.

It wasn't just the cereal story that showed that, though. The whole interview showed that they cared. They weren't doing this just for the money, or because startups were cool. The reason they were working so hard on this company was because it was their project. They had discovered an interesting new idea, and they just couldn't let it go.

Mundane as it sounds, that's the most powerful motivator of all, not just in startups, but in most ambitious undertakings: to be [genuinely interested](#) in what you're building. This is what really drives billionaires, or at least the ones who become billionaires from starting companies. The company is their project.

One thing few people realize about billionaires is that all of them could have stopped sooner. They could have gotten acquired, or found someone else to run the company. Many founders do. The ones who become really rich are the ones who keep working. And what makes them keep working is not just money. What keeps them working is the same thing that keeps anyone else working when they could stop if they wanted to: that there's nothing else they'd rather do.

That, not exploiting people, is the defining quality of people who become billionaires from starting companies. So that's what YC looks for in founders: authenticity. People's motives for starting startups are usually mixed. They're usually doing it from some combination of the desire to make money, the desire to seem cool, genuine interest in the problem, and unwillingness to work for someone else. The last two are more powerful motivators than the first two. It's ok for

founders to want to make money or to seem cool. Most do. But if the founders seem like they're doing it *just* to make money or *just* to seem cool, they're not likely to succeed on a big scale. The founders who are doing it for the money will take the first sufficiently large acquisition offer, and the ones who are doing it to seem cool will rapidly discover that there are much less painful ways of seeming cool. [3]

Y Combinator certainly sees founders whose m.o. is to exploit people. YC is a magnet for them, because they want the YC brand. But when the YC partners detect someone like that, they reject them. If bad people made good founders, the YC partners would face a moral dilemma. Fortunately they don't, because bad people make bad founders. This exploitative type of founder is not going to succeed on a large scale, and in fact probably won't even succeed on a small one, because they're always going to be taking shortcuts. They see YC itself as a shortcut.

Their exploitation usually begins with their own cofounders, which is disastrous, since the cofounders' relationship is the foundation of the company. Then it moves on to the users, which is also disastrous, because the sort of early adopters a successful startup wants as its initial users are the hardest to fool. The best this kind of founder can hope for is to keep the edifice of deception tottering along until some acquirer can be tricked into buying it. But that kind of acquisition is never very big. [4]

If professional billionaire scouts know that exploiting people is not the skill to look for, why do some politicians think this is the defining quality of billionaires?

I think they start from the feeling that it's wrong that one person could have so much more money than another. It's understandable where that feeling comes from. It's in our DNA, and even in the DNA of other species.

If they limited themselves to saying that it made them feel bad when one person had so much more money than other people, who would disagree? It makes me feel bad too, and I think people who make a lot of money have a moral obligation to use it for the common good. The mistake they make is to jump from feeling bad that some people are much richer than others to the conclusion that there's no legitimate way to make a very large amount of money. Now we're getting into statements that are not only falsifiable, but false.

There are certainly some people who become rich by doing bad things. But there are also plenty of people who behave badly and don't make that much from it. There is no correlation  $\blacklozenge$  in fact, probably an inverse correlation  $\blacklozenge$  between how badly you behave and how much money you make.

The greatest danger of this nonsense may not even be that it sends policy astray, but that it misleads ambitious people. Can you imagine a better way to destroy social mobility than by telling poor kids that the way to get rich is by exploiting people, while the rich kids know, from having watched the preceding generation do



it, how it's really done?

I'll tell you how it's really done, so you can at least tell your own kids the truth. It's all about users. The most reliable way to become a billionaire is to start a company that [grows fast](#), and the way to grow fast is to make what users want. Newly started startups have no choice but to delight users, or they'll never even get rolling. But this never stops being the lodestar, and bigger companies take their eye off it at their peril. Stop delighting users, and eventually someone else will.

Users are what the partners want to know about in YC interviews, and what I want to know about when I talk to founders that we funded ten years ago and who are billionaires now. What do users want? What new things could you build for them? Founders who've become billionaires are always eager to talk about that topic. That's how they became billionaires.

## Notes

[1] The YC partners have so much practice doing this that they sometimes see paths that the founders themselves haven't seen yet. The partners don't try to seem skeptical, as buyers in transactions often do to increase their leverage. Although the founders feel their job is to convince the partners of the potential of their idea, these roles are not infrequently reversed, and the founders leave the interview feeling their idea has more potential than they realized.

[2] In practice, 7 minutes would be enough. You rarely change your mind at minute 8. But 10 minutes is socially convenient.

[3] I myself took the first sufficiently large acquisition offer in my first startup, so I don't blame founders for doing this. There's nothing wrong with starting a startup to make money. You need to make money somehow, and for some people startups are the most efficient way to do it. I'm just saying that these are not the startups that get really big.

[4] Not these days, anyway. There were some big ones during the Internet Bubble, and indeed some big IPOs.

**Thanks** to Trevor Blackwell, Jessica Livingston, Robert Morris, Geoff Ralston, and

Harj Taggar for reading drafts of this.

# Earnestness

December 2020

Jessica and I have certain words that have special significance when we're talking about startups. The highest compliment we can pay to founders is to describe them as "earnest." This is not by itself a guarantee of success. You could be earnest but incapable. But when founders are both formidable (another of our words) and earnest, they're as close to unstoppable as you get.

Earnestness sounds like a boring, even Victorian virtue. It seems a bit of an anachronism that people in Silicon Valley would care about it. Why does this matter so much?

When you call someone earnest, you're making a statement about their motives. It means both that they're doing something for the right reasons, and that they're trying as hard as they can. If we imagine motives as vectors, it means both the direction and the magnitude are right. Though these are of course related: when people are doing something for the right reasons, they try harder. [1]

The reason motives matter so much in Silicon Valley is that so many people there have the wrong ones. Starting a successful startup makes you rich and famous. So a lot of the people trying to start them are doing it for those reasons. Instead of what? Instead of interest in the problem for its own sake. That is the root of earnestness. [2]

It's also the hallmark of a nerd. Indeed, when people describe themselves as "x nerds," what they mean is that they're interested in x for its own sake, and not because it's cool to be interested in x, or because of what they can get from it. They're saying they care so much about x that they're willing to sacrifice seeming cool for its sake.

A [genuine interest](#) in something is a very powerful motivator ♦ for some people, the most powerful motivator of all. [3] Which is why it's what Jessica and I look for in founders. But as well as being a source of strength, it's also a source of vulnerability. Caring constrains you. The earnest can't easily reply in kind to mocking banter, or put on a cool facade of nihil admirari. They care too much. They are doomed to be the straight man. That's a real disadvantage in your [teenage years](#), when mocking banter and nihil admirari often have the upper hand. But it becomes an advantage later.

It's a commonplace now that the kids who were nerds in high school become the cool kids' bosses later on. But people misunderstand why this happens. It's not just because the nerds are smarter, but also because they're more earnest. When the problems get harder than the fake ones you're given in high school, caring about them starts to matter.

Does it always matter? Do the earnest always win? Not always. It probably doesn't matter much in politics, or in crime, or in certain types of business that are similar to crime, like gambling, personal injury law, patent trolling, and so on. Nor does it matter in academic fields at the more [bogus](#) end of the spectrum. And though I don't know enough to say for sure, it may not matter in some kinds of humor: it may be possible to be completely cynical and still be very funny. [4]

Looking at the list of fields I mentioned, there's an obvious pattern. Except possibly for humor, these are all types of work I'd avoid like the plague. So that could be a useful heuristic for deciding which fields to work in: how much does earnestness matter? Which can in turn presumably be inferred from the prevalence of nerds at the top.

Along with "nerd," another word that tends to be associated with earnestness is "naive." The earnest often seem naive. It's not just that they don't have the motives other people have. They often don't fully grasp that such motives exist. Or they may know intellectually that they do, but because they don't feel them, they forget about them. [5]

It works to be slightly naive not just about motives but also, believe it or not, about the problems you're working on. Naive optimism can compensate for the bit rot that [rapid change](#) causes in established beliefs. You plunge into some problem saying "How hard can it be?", and then after solving it you learn that it was till recently insoluble.

Naivete is an obstacle for anyone who wants to seem sophisticated, and this is one reason would-be intellectuals find it so difficult to understand Silicon Valley. It hasn't been safe for such people to use the word "earnest" outside scare quotes since Oscar Wilde wrote "The Importance of Being Earnest" in 1895. And yet when you zoom in on Silicon Valley, right into [Jessica Livingston's brain](#), that's what her x-ray vision is seeking out in founders. Earnestness! Who'd have guessed? Reporters literally can't believe it when founders making piles of money say that they started their companies to make the world better. The situation seems made for mockery. How can these founders be so naive as not to realize how implausible they sound?

Though those asking this question don't realize it, that's not a rhetorical question.

A lot of founders are faking it, of course, particularly the smaller fry, and the soon to be smaller fry. But not all of them. There are a significant number of founders who really are interested in the problem they're solving mainly for its own sake.

Why shouldn't there be? We have no difficulty believing that people would be interested in history or math or even old bus tickets for their own sake. Why can't there be people interested in self-driving cars or social networks for their own sake? When you look at the question from this side, it seems obvious there would be. And isn't it likely that having a deep interest in something would be a source of great energy and resilience? It is in every other field.

The question really is why we have a blind spot about business. And the answer to that is obvious if you know enough history. For most of history, making large amounts of money has not been very intellectually interesting. In preindustrial times it was never far from robbery, and some areas of business still retain that character, except using lawyers instead of soldiers.

But there are other areas of business where the work is genuinely interesting. Henry Ford got to spend much of his time working on interesting technical problems, and for the last several decades the trend in that direction has been accelerating. It's much easier now to make a lot of money by working on something you're interested in than it was [50 years ago](#). And that, rather than how fast they grow, may be the most important change that startups represent. Though indeed, the fact that the work is genuinely interesting is a big part of why it gets done so fast. [6]

Can you imagine a more important change than one in the relationship between intellectual curiosity and money? These are two of the most powerful forces in the world, and in my lifetime they've become significantly more aligned. How could you not be fascinated to watch something like this happening in real time?

I meant this essay to be about earnestness generally, and now I've gone and talked about startups again. But I suppose at least it serves as an example of an x nerd in the wild.

## Notes

[1] It's interesting how many different ways there are *not* to be earnest: to be cleverly cynical, to be superficially brilliant, to be conspicuously virtuous, to be cool, to be sophisticated, to be orthodox, to be a snob, to bully, to pander, to be on the make. This pattern suggests that earnestness is not one end of a continuum, but a target one can fall short of in multiple dimensions.

Another thing I notice about this list is that it sounds like a list of the ways people behave on Twitter. Whatever else social media is, it's a vivid catalogue of ways not to be earnest.

[2] People's motives are as mixed in Silicon Valley as anywhere else. Even the founders motivated mostly by money tend to be at least somewhat interested in the problem they're solving, and even the founders most interested in the problem they're solving also like the idea of getting rich. But there's great variation in the relative proportions of different founders' motivations.

And when I talk about "wrong" motives, I don't mean morally wrong. There's nothing morally wrong with starting a startup to make money. I just mean that those startups don't do as well.

[3] The most powerful motivator for most people is probably family. But there are some for whom intellectual curiosity comes first. In his (wonderful) autobiography, Paul Halmos says explicitly that for a mathematician, math must come before anything else, including family. Which at least implies that it did for him.

[4] Interestingly, just as the word "nerd" implies earnestness even when used as a metaphor, the word "politics" implies the opposite. It's not only in actual politics that earnestness seems to be a handicap, but also in office politics and academic politics.

[5] It's a bigger social error to seem naive in most European countries than it is in America, and this may be one of subtler reasons startups are less common there. Founder culture is completely at odds with sophisticated cynicism.

The most earnest part of Europe is Scandinavia, and not surprisingly this is also the region with the highest number of successful startups per capita.

[6] Much of business is schleps, and probably always will be. But even being a professor is largely schleps. It would be interesting to collect statistics about the schlep ratios of different jobs, but I suspect they'd rarely be less than 30%.

**Thanks** to Trevor Blackwell, Patrick Collison, Suhail Doshi, Jessica Livingston, Mattias Ljungman, Harj Taggar, and Kyle Vogt for reading drafts of this.

# What I Worked On

February 2021

Before college the two main things I worked on, outside of school, were writing and programming. I didn't write essays. I wrote what beginning writers were supposed to write then, and probably still are: short stories. My stories were awful. They had hardly any plot, just characters with strong feelings, which I imagined made them deep.

The first programs I tried writing were on the IBM 1401 that our school district used for what was then called "data processing." This was in 9th grade, so I was 13 or 14. The school district's 1401 happened to be in the basement of our junior high school, and my friend Rich Draves and I got permission to use it. It was like a mini Bond villain's lair down there, with all these alien-looking machines ♦ CPU, disk drives, printer, card reader ♦ sitting up on a raised floor under bright fluorescent lights.

The language we used was an early version of Fortran. You had to type programs on punch cards, then stack them in the card reader and press a button to load the program into memory and run it. The result would ordinarily be to print something on the spectacularly loud printer.

I was puzzled by the 1401. I couldn't figure out what to do with it. And in retrospect there's not much I could have done with it. The only form of input to programs was data stored on punched cards, and I didn't have any data stored on punched cards. The only other option was to do things that didn't rely on any input, like calculate approximations of pi, but I didn't know enough math to do anything interesting of that type. So I'm not surprised I can't remember any programs I wrote, because they can't have done much. My clearest memory is of the moment I learned it was possible for programs not to terminate, when one of mine didn't. On a machine without time-sharing, this was a social as well as a technical error, as the data center manager's expression made clear.

With microcomputers, everything changed. Now you could have a computer sitting right in front of you, on a desk, that could respond to your keystrokes as it was running instead of just churning through a stack of punch cards and then stopping.

[1]

The first of my friends to get a microcomputer built it himself. It was sold as a kit

by Heathkit. I remember vividly how impressed and envious I felt watching him sitting in front of it, typing programs right into the computer.

Computers were expensive in those days and it took me years of nagging before I convinced my father to buy one, a TRS-80, in about 1980. The gold standard then was the Apple II, but a TRS-80 was good enough. This was when I really started programming. I wrote simple games, a program to predict how high my model rockets would fly, and a word processor that my father used to write at least one book. There was only room in memory for about 2 pages of text, so he'd write 2 pages at a time and then print them out, but it was a lot better than a typewriter.

Though I liked programming, I didn't plan to study it in college. In college I was going to study philosophy, which sounded much more powerful. It seemed, to my naive high school self, to be the study of the ultimate truths, compared to which the things studied in other fields would be mere domain knowledge. What I discovered when I got to college was that the other fields took up so much of the space of ideas that there wasn't much left for these supposed ultimate truths. All that seemed left for philosophy were edge cases that people in other fields felt could safely be ignored.

I couldn't have put this into words when I was 18. All I knew at the time was that I kept taking philosophy courses and they kept being boring. So I decided to switch to AI.

AI was in the air in the mid 1980s, but there were two things especially that made me want to work on it: a novel by Heinlein called *The Moon is a Harsh Mistress*, which featured an intelligent computer called Mike, and a PBS documentary that showed Terry Winograd using SHRDLU. I haven't tried rereading *The Moon is a Harsh Mistress*, so I don't know how well it has aged, but when I read it I was drawn entirely into its world. It seemed only a matter of time before we'd have Mike, and when I saw Winograd using SHRDLU, it seemed like that time would be a few years at most. All you had to do was teach SHRDLU more words.

There weren't any classes in AI at Cornell then, not even graduate classes, so I started trying to teach myself. Which meant learning Lisp, since in those days Lisp was regarded as the language of AI. The commonly used programming languages then were pretty primitive, and programmers' ideas correspondingly so. The default language at Cornell was a Pascal-like language called PL/I, and the situation was similar elsewhere. Learning Lisp expanded my concept of a program so fast that it was years before I started to have a sense of where the new limits were. This was more like it; this was what I had expected college to do. It wasn't happening in a class, like it was supposed to, but that was ok. For the next couple years I was on a roll. I knew what I was going to do.

For my undergraduate thesis, I reverse-engineered SHRDLU. My God did I love working on that program. It was a pleasing bit of code, but what made it even more exciting was my belief 💎 hard to imagine now, but not unique in 1985 💎 that it was already climbing the lower slopes of intelligence.



I had gotten into a program at Cornell that didn't make you choose a major. You could take whatever classes you liked, and choose whatever you liked to put on your degree. I of course chose "Artificial Intelligence." When I got the actual physical diploma, I was dismayed to find that the quotes had been included, which made them read as scare-quotes. At the time this bothered me, but now it seems amusingly accurate, for reasons I was about to discover.

I applied to 3 grad schools: MIT and Yale, which were renowned for AI at the time, and Harvard, which I'd visited because Rich Draves went there, and was also home to Bill Woods, who'd invented the type of parser I used in my SHRDLU clone. Only Harvard accepted me, so that was where I went.

I don't remember the moment it happened, or if there even was a specific moment, but during the first year of grad school I realized that AI, as practiced at the time, was a hoax. By which I mean the sort of AI in which a program that's told "the dog is sitting on the chair" translates this into some formal representation and adds it to the list of things it knows.

What these programs really showed was that there's a subset of natural language that's a formal language. But a very proper subset. It was clear that there was an unbridgeable gap between what they could do and actually understanding natural language. It was not, in fact, simply a matter of teaching SHRDLU more words. That whole way of doing AI, with explicit data structures representing concepts, was not going to work. Its brokenness did, as so often happens, generate a lot of opportunities to write papers about various band-aids that could be applied to it, but it was never going to get us Mike.

So I looked around to see what I could salvage from the wreckage of my plans, and there was Lisp. I knew from experience that Lisp was interesting for its own sake and not just for its association with AI, even though that was the main reason people cared about it at the time. So I decided to focus on Lisp. In fact, I decided to write a book about Lisp hacking. It's scary to think how little I knew about Lisp hacking when I started writing that book. But there's nothing like writing a book about something to help you learn it. The book, *On Lisp*, wasn't published till 1993, but I wrote much of it in grad school.

Computer Science is an uneasy alliance between two halves, theory and systems. The theory people prove things, and the systems people build things. I wanted to build things. I had plenty of respect for theory ♦ indeed, a sneaking suspicion that it was the more admirable of the two halves ♦ but building things seemed so much more exciting.



The problem with systems work, though, was that it didn't last. Any program you wrote today, no matter how good, would be obsolete in a couple decades at best. People might mention your software in footnotes, but no one would actually use it. And indeed, it would seem very feeble work. Only people with a sense of the history of the field would even realize that, in its time, it had been good.

There were some surplus Xerox Dandelions floating around the computer lab at one point. Anyone who wanted one to play around with could have one. I was briefly tempted, but they were so slow by present standards; what was the point? No one else wanted one either, so off they went. That was what happened to systems work.

I wanted not just to build things, but to build things that would last.

In this dissatisfied state I went in 1988 to visit Rich Draves at CMU, where he was in grad school. One day I went to visit the Carnegie Institute, where I'd spent a lot of time as a kid. While looking at a painting there I realized something that might seem obvious, but was a big surprise to me. There, right on the wall, was something you could make that would last. Paintings didn't become obsolete. Some of the best ones were hundreds of years old.

And moreover this was something you could make a living doing. Not as easily as you could by writing software, of course, but I thought if you were really industrious and lived really cheaply, it had to be possible to make enough to survive. And as an artist you could be truly independent. You wouldn't have a boss, or even need to get research funding.

I had always liked looking at paintings. Could I make them? I had no idea. I'd never imagined it was even possible. I knew intellectually that people made art  that it didn't just appear spontaneously  but it was as if the people who made it were a different species. They either lived long ago or were mysterious geniuses doing strange things in profiles in *Life* magazine. The idea of actually being able to make art, to put that verb before that noun, seemed almost miraculous.

That fall I started taking art classes at Harvard. Grad students could take classes in any department, and my advisor, Tom Cheatham, was very easy going. If he even knew about the strange classes I was taking, he never said anything.

So now I was in a PhD program in computer science, yet planning to be an artist, yet also genuinely in love with Lisp hacking and working away at *On Lisp*. In other words, like many a grad student, I was working energetically on multiple projects that were not my thesis.

I didn't see a way out of this situation. I didn't want to drop out of grad school, but how else was I going to get out? I remember when my friend Robert Morris got kicked out of Cornell for writing the internet worm of 1988, I was envious that he'd found such a spectacular way to get out of grad school.

Then one day in April 1990 a crack appeared in the wall. I ran into professor Cheatham and he asked if I was far enough along to graduate that June. I didn't have a word of my dissertation written, but in what must have been the quickest bit of thinking in my life, I decided to take a shot at writing one in the 5 weeks or so that remained before the deadline, reusing parts of *On Lisp* where I could, and I

was able to respond, with no perceptible delay "Yes, I think so. I'll give you something to read in a few days."

I picked applications of continuations as the topic. In retrospect I should have written about macros and embedded languages. There's a whole world there that's barely been explored. But all I wanted was to get out of grad school, and my rapidly written dissertation sufficed, just barely.

Meanwhile I was applying to art schools. I applied to two: RISD in the US, and the Accademia di Belli Arti in Florence, which, because it was the oldest art school, I imagined would be good. RISD accepted me, and I never heard back from the Accademia, so off to Providence I went.

I'd applied for the BFA program at RISD, which meant in effect that I had to go to college again. This was not as strange as it sounds, because I was only 25, and art schools are full of people of different ages. RISD counted me as a transfer sophomore and said I had to do the foundation that summer. The foundation means the classes that everyone has to take in fundamental subjects like drawing, color, and design.

Toward the end of the summer I got a big surprise: a letter from the Accademia, which had been delayed because they'd sent it to Cambridge England instead of Cambridge Massachusetts, inviting me to take the entrance exam in Florence that fall. This was now only weeks away. My nice landlady let me leave my stuff in her attic. I had some money saved from consulting work I'd done in grad school; there was probably enough to last a year if I lived cheaply. Now all I had to do was learn Italian.

Only *stranieri* (foreigners) had to take this entrance exam. In retrospect it may well have been a way of excluding them, because there were so many *stranieri* attracted by the idea of studying art in Florence that the Italian students would otherwise have been outnumbered. I was in decent shape at painting and drawing from the RISD foundation that summer, but I still don't know how I managed to pass the written exam. I remember that I answered the essay question by writing about Cezanne, and that I cranked up the intellectual level as high as I could to make the most of my limited vocabulary. [2]

I'm only up to age 25 and already there are such conspicuous patterns. Here I was, yet again about to attend some august institution in the hopes of learning about some prestigious subject, and yet again about to be disappointed. The students and faculty in the painting department at the Accademia were the nicest people you could imagine, but they had long since arrived at an arrangement whereby the students wouldn't require the faculty to teach anything, and in return the faculty wouldn't require the students to learn anything. And at the same time all involved would adhere outwardly to the conventions of a 19th century atelier. We actually had one of those little stoves, fed with kindling, that you see in 19th century studio paintings, and a nude model sitting as close to it as possible without getting burned. Except hardly anyone else painted her besides me. The rest of the

students spent their time chatting or occasionally trying to imitate things they'd seen in American art magazines.

Our model turned out to live just down the street from me. She made a living from a combination of modelling and making fakes for a local antique dealer. She'd copy an obscure old painting out of a book, and then he'd take the copy and maltreat it to make it look old. [3]

While I was a student at the Accademia I started painting still lives in my bedroom at night. These paintings were tiny, because the room was, and because I painted them on leftover scraps of canvas, which was all I could afford at the time. Painting still lives is different from painting people, because the subject, as its name suggests, can't move. People can't sit for more than about 15 minutes at a time, and when they do they don't sit very still. So the traditional m.o. for painting people is to know how to paint a generic person, which you then modify to match the specific person you're painting. Whereas a still life you can, if you want, copy pixel by pixel from what you're seeing. You don't want to stop there, of course, or you get merely photographic accuracy, and what makes a still life interesting is that it's been through a head. You want to emphasize the visual cues that tell you, for example, that the reason the color changes suddenly at a certain point is that it's the edge of an object. By subtly emphasizing such things you can make paintings that are more realistic than photographs not just in some metaphorical sense, but in the strict information-theoretic sense. [4]

I liked painting still lives because I was curious about what I was seeing. In everyday life, we aren't consciously aware of much we're seeing. Most visual perception is handled by low-level processes that merely tell your brain "that's a water droplet" without telling you details like where the lightest and darkest points are, or "that's a bush" without telling you the shape and position of every leaf. This is a feature of brains, not a bug. In everyday life it would be distracting to notice every leaf on every bush. But when you have to paint something, you have to look more closely, and when you do there's a lot to see. You can still be noticing new things after days of trying to paint something people usually take for granted, just as you can after days of trying to write an essay about something people usually take for granted.

This is not the only way to paint. I'm not 100% sure it's even a good way to paint. But it seemed a good enough bet to be worth trying.

Our teacher, professor Ulivi, was a nice guy. He could see I worked hard, and gave me a good grade, which he wrote down in a sort of passport each student had. But the Accademia wasn't teaching me anything except Italian, and my money was running out, so at the end of the first year I went back to the US.

I wanted to go back to RISD, but I was now broke and RISD was very expensive, so I decided to get a job for a year and then return to RISD the next fall. I got one at a company called Interleaf, which made software for creating documents. You mean like Microsoft Word? Exactly. That was how I learned that low end software

tends to eat high end software. But Interleaf still had a few years to live yet. [5]

Interleaf had done something pretty bold. Inspired by Emacs, they'd added a scripting language, and even made the scripting language a dialect of Lisp. Now they wanted a Lisp hacker to write things in it. This was the closest thing I've had to a normal job, and I hereby apologize to my boss and coworkers, because I was a bad employee. Their Lisp was the thinnest icing on a giant C cake, and since I didn't know C and didn't want to learn it, I never understood most of the software. Plus I was terribly irresponsible. This was back when a programming job meant showing up every day during certain working hours. That seemed unnatural to me, and on this point the rest of the world is coming around to my way of thinking, but at the time it caused a lot of friction. Toward the end of the year I spent much of my time surreptitiously working on *On Lisp*, which I had by this time gotten a contract to publish.

The good part was that I got paid huge amounts of money, especially by art student standards. In Florence, after paying my part of the rent, my budget for everything else had been \$7 a day. Now I was getting paid more than 4 times that every hour, even when I was just sitting in a meeting. By living cheaply I not only managed to save enough to go back to RISD, but also paid off my college loans.

I learned some useful things at Interleaf, though they were mostly about what not to do. I learned that it's better for technology companies to be run by product people than sales people (though sales is a real skill and people who are good at it are really good at it), that it leads to bugs when code is edited by too many people, that cheap office space is no bargain if it's depressing, that planned meetings are inferior to corridor conversations, that big, bureaucratic customers are a dangerous source of money, and that there's not much overlap between conventional office hours and the optimal time for hacking, or conventional offices and the optimal place for it.

But the most important thing I learned, and which I used in both Viaweb and Y Combinator, is that the low end eats the high end: that it's good to be the "entry level" option, even though that will be less prestigious, because if you're not, someone else will be, and will squash you against the ceiling. Which in turn means that prestige is a danger sign.

When I left to go back to RISD the next fall, I arranged to do freelance work for the group that did projects for customers, and this was how I survived for the next several years. When I came back to visit for a project later on, someone told me about a new thing called HTML, which was, as he described it, a derivative of SGML. Markup language enthusiasts were an occupational hazard at Interleaf and I ignored him, but this HTML thing later became a big part of my life.

In the fall of 1992 I moved back to Providence to continue at RISD. The foundation had merely been intro stuff, and the Accademia had been a (very civilized) joke. Now I was going to see what real art school was like. But alas it was more like the Accademia than not. Better organized, certainly, and a lot more expensive, but it

was now becoming clear that art school did not bear the same relationship to art that medical school bore to medicine. At least not the painting department. The textile department, which my next door neighbor belonged to, seemed to be pretty rigorous. No doubt illustration and architecture were too. But painting was post-rigorous. Painting students were supposed to express themselves, which to the more worldly ones meant to try to cook up some sort of distinctive signature style.

A signature style is the visual equivalent of what in show business is known as a "schtick": something that immediately identifies the work as yours and no one else's. For example, when you see a painting that looks like a certain kind of cartoon, you know it's by Roy Lichtenstein. So if you see a big painting of this type hanging in the apartment of a hedge fund manager, you know he paid millions of dollars for it. That's not always why artists have a signature style, but it's usually why buyers pay a lot for such work. [6]

There were plenty of earnest students too: kids who "could draw" in high school, and now had come to what was supposed to be the best art school in the country, to learn to draw even better. They tended to be confused and demoralized by what they found at RISD, but they kept going, because painting was what they did. I was not one of the kids who could draw in high school, but at RISD I was definitely closer to their tribe than the tribe of signature style seekers.

I learned a lot in the color class I took at RISD, but otherwise I was basically teaching myself to paint, and I could do that for free. So in 1993 I dropped out. I hung around Providence for a bit, and then my college friend Nancy Parmet did me a big favor. A rent-controlled apartment in a building her mother owned in New York was becoming vacant. Did I want it? It wasn't much more than my current place, and New York was supposed to be where the artists were. So yes, I wanted it! [7]

Asterix comics begin by zooming in on a tiny corner of Roman Gaul that turns out not to be controlled by the Romans. You can do something similar on a map of New York City: if you zoom in on the Upper East Side, there's a tiny corner that's not rich, or at least wasn't in 1993. It's called Yorkville, and that was my new home. Now I was a New York artist ♦ in the strictly technical sense of making paintings and living in New York.

I was nervous about money, because I could sense that Interleaf was on the way down. Freelance Lisp hacking work was very rare, and I didn't want to have to program in another language, which in those days would have meant C++ if I was lucky. So with my unerring nose for financial opportunity, I decided to write another book on Lisp. This would be a popular book, the sort of book that could be used as a textbook. I imagined myself living frugally off the royalties and spending all my time painting. (The painting on the cover of this book, *ANSI Common Lisp*, is one that I painted around this time.)

The best thing about New York for me was the presence of Idelle and Julian Weber. Idelle Weber was a painter, one of the early photorealists, and I'd taken her


painting class at Harvard. I've never known a teacher more beloved by her students. Large numbers of former students kept in touch with her, including me. After I moved to New York I became her de facto studio assistant.

She liked to paint on big, square canvases, 4 to 5 feet on a side. One day in late 1994 as I was stretching one of these monsters there was something on the radio about a famous fund manager. He wasn't that much older than me, and was super rich. The thought suddenly occurred to me: why don't I become rich? Then I'll be able to work on whatever I want.

Meanwhile I'd been hearing more and more about this new thing called the World Wide Web. Robert Morris showed it to me when I visited him in Cambridge, where he was now in grad school at Harvard. It seemed to me that the web would be a big deal. I'd seen what graphical user interfaces had done for the popularity of microcomputers. It seemed like the web would do the same for the internet.

If I wanted to get rich, here was the next train leaving the station. I was right about that part. What I got wrong was the idea. I decided we should start a company to put art galleries online. I can't honestly say, after reading so many Y Combinator applications, that this was the worst startup idea ever, but it was up there. Art galleries didn't want to be online, and still don't, not the fancy ones. That's not how they sell. I wrote some software to generate web sites for galleries, and Robert wrote some to resize images and set up an http server to serve the pages. Then we tried to sign up galleries. To call this a difficult sale would be an understatement. It was difficult to give away. A few galleries let us make sites for them for free, but none paid us.

Then some online stores started to appear, and I realized that except for the order buttons they were identical to the sites we'd been generating for galleries. This impressive-sounding thing called an "internet storefront" was something we already knew how to build.

So in the summer of 1995, after I submitted the camera-ready copy of *ANSI Common Lisp* to the publishers, we started trying to write software to build online stores. At first this was going to be normal desktop software, which in those days meant Windows software. That was an alarming prospect, because neither of us knew how to write Windows software or wanted to learn. We lived in the Unix world. But we decided we'd at least try writing a prototype store builder on Unix. Robert wrote a shopping cart, and I wrote a new site generator for stores  in Lisp, of course.

We were working out of Robert's apartment in Cambridge. His roommate was away for big chunks of time, during which I got to sleep in his room. For some reason there was no bed frame or sheets, just a mattress on the floor. One morning as I was lying on this mattress I had an idea that made me sit up like a capital L. What if we ran the software on the server, and let users control it by clicking on links? Then we'd never have to write anything to run on users' computers. We could generate the sites on the same server we'd serve them from. Users wouldn't need

anything more than a browser.

This kind of software, known as a web app, is common now, but at the time it wasn't clear that it was even possible. To find out, we decided to try making a version of our store builder that you could control through the browser. A couple days later, on August 12, we had one that worked. The UI was horrible, but it proved you could build a whole store through the browser, without any client software or typing anything into the command line on the server.

Now we felt like we were really onto something. I had visions of a whole new generation of software working this way. You wouldn't need versions, or ports, or any of that crap. At Interleaf there had been a whole group called Release Engineering that seemed to be at least as big as the group that actually wrote the software. Now you could just update the software right on the server.

We started a new company we called Viaweb, after the fact that our software worked via the web, and we got \$10,000 in seed funding from Idelle's husband Julian. In return for that and doing the initial legal work and giving us business advice, we gave him 10% of the company. Ten years later this deal became the model for Y Combinator's. We knew founders needed something like this, because we'd needed it ourselves.

At this stage I had a negative net worth, because the thousand dollars or so I had in the bank was more than counterbalanced by what I owed the government in taxes. (Had I diligently set aside the proper proportion of the money I'd made consulting for Interleaf? No, I had not.) So although Robert had his graduate student stipend, I needed that seed funding to live on.

We originally hoped to launch in September, but we got more ambitious about the software as we worked on it. Eventually we managed to build a WYSIWYG site builder, in the sense that as you were creating pages, they looked exactly like the static ones that would be generated later, except that instead of leading to static pages, the links all referred to closures stored in a hash table on the server.

It helped to have studied art, because the main goal of an online store builder is to make users look legit, and the key to looking legit is high production values. If you get page layouts and fonts and colors right, you can make a guy running a store out of his bedroom look more legit than a big company.

(If you're curious why my site looks so old-fashioned, it's because it's still made with this software. It may look clunky today, but in 1996 it was the last word in slick.)

In September, Robert rebelled. "We've been working on this for a month," he said, "and it's still not done." This is funny in retrospect, because he would still be working on it almost 3 years later. But I decided it might be prudent to recruit more programmers, and I asked Robert who else in grad school with him was really good. He recommended Trevor Blackwell, which surprised me at first,



because at that point I knew Trevor mainly for his plan to reduce everything in his life to a stack of notecards, which he carried around with him. But Rtm was right, as usual. Trevor turned out to be a frighteningly effective hacker.

It was a lot of fun working with Robert and Trevor. They're the two most [independent-minded](#) people I know, and in completely different ways. If you could see inside Rtm's brain it would look like a colonial New England church, and if you could see inside Trevor's it would look like the worst excesses of Austrian Rococo.

We opened for business, with 6 stores, in January 1996. It was just as well we waited a few months, because although we worried we were late, we were actually almost fatally early. There was a lot of talk in the press then about ecommerce, but not many people actually wanted online stores. [8]

There were three main parts to the software: the editor, which people used to build sites and which I wrote, the shopping cart, which Robert wrote, and the manager, which kept track of orders and statistics, and which Trevor wrote. In its time, the editor was one of the best general-purpose site builders. I kept the code tight and didn't have to integrate with any other software except Robert's and Trevor's, so it was quite fun to work on. If all I'd had to do was work on this software, the next 3 years would have been the easiest of my life. Unfortunately I had to do a lot more, all of it stuff I was worse at than programming, and the next 3 years were instead the most stressful.

There were a lot of startups making ecommerce software in the second half of the 90s. We were determined to be the Microsoft Word, not the Interleaf. Which meant being easy to use and inexpensive. It was lucky for us that we were poor, because that caused us to make Viaweb even more inexpensive than we realized. We charged \$100 a month for a small store and \$300 a month for a big one. This low price was a big attraction, and a constant thorn in the sides of competitors, but it wasn't because of some clever insight that we set the price low. We had no idea what businesses paid for things. \$300 a month seemed like a lot of money to us.

We did a lot of things right by accident like that. For example, we did what's now called "doing things that [don't scale](#)," although at the time we would have described it as "being so lame that we're driven to the most desperate measures to get users." The most common of which was building stores for them. This seemed particularly humiliating, since the whole *raison d'être* of our software was that people could use it to make their own stores. But anything to get users.

We learned a lot more about retail than we wanted to know. For example, that if you could only have a small image of a man's shirt (and all images were small then by present standards), it was better to have a closeup of the collar than a picture of the whole shirt. The reason I remember learning this was that it meant I had to rescan about 30 images of men's shirts. My first set of scans were so beautiful too.

Though this felt wrong, it was exactly the right thing to be doing. Building stores for users taught us about retail, and about how it felt to use our software. I was

initially both mystified and repelled by "business" and thought we needed a "business person" to be in charge of it, but once we started to get users, I was converted, in much the same way I was converted to [fatherhood](#) once I had kids. Whatever users wanted, I was all theirs. Maybe one day we'd have so many users that I couldn't scan their images for them, but in the meantime there was nothing more important to do.

Another thing I didn't get at the time is that [growth rate](#) is the ultimate test of a startup. Our growth rate was fine. We had about 70 stores at the end of 1996 and about 500 at the end of 1997. I mistakenly thought the thing that mattered was the absolute number of users. And that is the thing that matters in the sense that that's how much money you're making, and if you're not making enough, you might go out of business. But in the long term the growth rate takes care of the absolute number. If we'd been a startup I was advising at Y Combinator, I would have said: Stop being so stressed out, because you're doing fine. You're growing 7x a year. Just don't hire too many more people and you'll soon be profitable, and then you'll control your own destiny.

Alas I hired lots more people, partly because our investors wanted me to, and partly because that's what startups did during the Internet Bubble. A company with just a handful of employees would have seemed amateurish. So we didn't reach breakeven until about when Yahoo bought us in the summer of 1998. Which in turn meant we were at the mercy of investors for the entire life of the company. And since both we and our investors were noobs at startups, the result was a mess even by startup standards.

It was a huge relief when Yahoo bought us. In principle our Viaweb stock was valuable. It was a share in a business that was profitable and growing rapidly. But it didn't feel very valuable to me; I had no idea how to value a business, but I was all too keenly aware of the near-death experiences we seemed to have every few months. Nor had I changed my grad student lifestyle significantly since we started. So when Yahoo bought us it felt like going from rags to riches. Since we were going to California, I bought a car, a yellow 1998 VW GTI. I remember thinking that its leather seats alone were by far the most luxurious thing I owned.

The next year, from the summer of 1998 to the summer of 1999, must have been the least productive of my life. I didn't realize it at the time, but I was worn out from the effort and stress of running Viaweb. For a while after I got to California I tried to continue my usual m.o. of programming till 3 in the morning, but fatigue combined with Yahoo's prematurely aged [culture](#) and grim cube farm in Santa Clara gradually dragged me down. After a few months it felt disconcertingly like working at Interleaf.

Yahoo had given us a lot of options when they bought us. At the time I thought Yahoo was so overvalued that they'd never be worth anything, but to my astonishment the stock went up 5x in the next year. I hung on till the first chunk of options vested, then in the summer of 1999 I left. It had been so long since I'd painted anything that I'd half forgotten why I was doing this. My brain had been

entirely full of software and men's shirts for 4 years. But I had done this to get rich so I could paint, I reminded myself, and now I was rich, so I should go paint.

When I said I was leaving, my boss at Yahoo had a long conversation with me about my plans. I told him all about the kinds of pictures I wanted to paint. At the time I was touched that he took such an interest in me. Now I realize it was because he thought I was lying. My options at that point were worth about \$2 million a month. If I was leaving that kind of money on the table, it could only be to go and start some new startup, and if I did, I might take people with me. This was the height of the Internet Bubble, and Yahoo was ground zero of it. My boss was at that moment a billionaire. Leaving then to start a new startup must have seemed to him an insanely, and yet also plausibly, ambitious plan.

But I really was quitting to paint, and I started immediately. There was no time to lose. I'd already burned 4 years getting rich. Now when I talk to founders who are leaving after selling their companies, my advice is always the same: take a vacation. That's what I should have done, just gone off somewhere and done nothing for a month or two, but the idea never occurred to me.

So I tried to paint, but I just didn't seem to have any energy or ambition. Part of the problem was that I didn't know many people in California. I'd compounded this problem by buying a house up in the Santa Cruz Mountains, with a beautiful view but miles from anywhere. I stuck it out for a few more months, then in desperation I went back to New York, where unless you understand about rent control you'll be surprised to hear I still had my apartment, sealed up like a tomb of my old life. Idelle was in New York at least, and there were other people trying to paint there, even though I didn't know any of them.

When I got back to New York I resumed my old life, except now I was rich. It was as weird as it sounds. I resumed all my old patterns, except now there were doors where there hadn't been. Now when I was tired of walking, all I had to do was raise my hand, and (unless it was raining) a taxi would stop to pick me up. Now when I walked past charming little restaurants I could go in and order lunch. It was exciting for a while. Painting started to go better. I experimented with a new kind of still life where I'd paint one painting in the old way, then photograph it and print it, blown up, on canvas, and then use that as the underpainting for a second still life, painted from the same objects (which hopefully hadn't rotted yet).

Meanwhile I looked for an apartment to buy. Now I could actually choose what neighborhood to live in. Where, I asked myself and various real estate agents, is the Cambridge of New York? Aided by occasional visits to actual Cambridge, I gradually realized there wasn't one. Huh.

Around this time, in the spring of 2000, I had an idea. It was clear from our experience with Viaweb that web apps were the future. Why not build a web app for making web apps? Why not let people edit code on our server through the browser, and then host the resulting applications for them? [\[9\]](#) You could run all sorts of services on the servers that these applications could use just by making an

API call: making and receiving phone calls, manipulating images, taking credit card payments, etc.

I got so excited about this idea that I couldn't think about anything else. It seemed obvious that this was the future. I didn't particularly want to start another company, but it was clear that this idea would have to be embodied as one, so I decided to move to Cambridge and start it. I hoped to lure Robert into working on it with me, but there I ran into a hitch. Robert was now a postdoc at MIT, and though he'd made a lot of money the last time I'd lured him into working on one of my schemes, it had also been a huge time sink. So while he agreed that it sounded like a plausible idea, he firmly refused to work on it.

Hmph. Well, I'd do it myself then. I recruited Dan Giffin, who had worked for Viaweb, and two undergrads who wanted summer jobs, and we got to work trying to build what it's now clear is about twenty companies and several open source projects worth of software. The language for defining applications would of course be a dialect of Lisp. But I wasn't so naive as to assume I could spring an overt Lisp on a general audience; we'd hide the parentheses, like Dylan did.

By then there was a name for the kind of company Viaweb was, an "application service provider," or ASP. This name didn't last long before it was replaced by "software as a service," but it was current for long enough that I named this new company after it: it was going to be called Aspra.

I started working on the application builder, Dan worked on network infrastructure, and the two undergrads worked on the first two services (images and phone calls). But about halfway through the summer I realized I really didn't want to run a company ♦ especially not a big one, which it was looking like this would have to be. I'd only started Viaweb because I needed the money. Now that I didn't need money anymore, why was I doing this? If this vision had to be realized as a company, then screw the vision. I'd build a subset that could be done as an open source project.

Much to my surprise, the time I spent working on this stuff was not wasted after all. After we started Y Combinator, I would often encounter startups working on parts of this new architecture, and it was very useful to have spent so much time thinking about it and even trying to write some of it.

The subset I would build as an open source project was the new Lisp, whose parentheses I now wouldn't even have to hide. A lot of Lisp hackers dream of building a new Lisp, partly because one of the distinctive features of the language is that it has dialects, and partly, I think, because we have in our minds a Platonic form of Lisp that all existing dialects fall short of. I certainly did. So at the end of the summer Dan and I switched to working on this new dialect of Lisp, which I called Arc, in a house I bought in Cambridge.

The following spring, lightning struck. I was invited to give a talk at a Lisp conference, so I gave one about how we'd used Lisp at Viaweb. Afterward I put a

postscript file of this talk online, on paulgraham.com, which I'd created years before using Viaweb but had never used for anything. In one day it got 30,000 page views. What on earth had happened? The referring urls showed that someone had posted it on Slashdot. [\[10\]](#)

Wow, I thought, there's an audience. If I write something and put it on the web, anyone can read it. That may seem obvious now, but it was surprising then. In the print era there was a narrow channel to readers, guarded by fierce monsters known as editors. The only way to get an audience for anything you wrote was to get it published as a book, or in a newspaper or magazine. Now anyone could publish anything.

This had been possible in principle since 1993, but not many people had realized it yet. I had been intimately involved with building the infrastructure of the web for most of that time, and a writer as well, and it had taken me 8 years to realize it. Even then it took me several years to understand the implications. It meant there would be a whole new generation of [essays](#). [\[11\]](#)

In the print era, the channel for publishing essays had been vanishingly small. Except for a few officially anointed thinkers who went to the right parties in New York, the only people allowed to publish essays were specialists writing about their specialties. There were so many essays that had never been written, because there had been no way to publish them. Now they could be, and I was going to write them. [\[12\]](#)

I've worked on several different things, but to the extent there was a turning point where I figured out what to work on, it was when I started publishing essays online. From then on I knew that whatever else I did, I'd always write essays too.

I knew that online essays would be a [marginal](#) medium at first. Socially they'd seem more like rants posted by nutjobs on their GeoCities sites than the genteel and beautifully typeset compositions published in *The New Yorker*. But by this point I knew enough to find that encouraging instead of discouraging.

One of the most conspicuous patterns I've noticed in my life is how well it has worked, for me at least, to work on things that weren't prestigious. Still life has always been the least prestigious form of painting. Viaweb and Y Combinator both seemed lame when we started them. I still get the glassy eye from strangers when they ask what I'm writing, and I explain that it's an essay I'm going to publish on my web site. Even Lisp, though prestigious intellectually in something like the way Latin is, also seems about as hip.

It's not that unprestigious types of work are good per se. But when you find yourself drawn to some kind of work despite its current lack of prestige, it's a sign both that there's something real to be discovered there, and that you have the right kind of motives. Impure motives are a big danger for the ambitious. If anything is going to lead you astray, it will be the desire to impress people. So while working on things that aren't prestigious doesn't guarantee you're on the

right track, it at least guarantees you're not on the most common type of wrong one.

Over the next several years I wrote lots of essays about all kinds of different topics. O'Reilly reprinted a collection of them as a book, called *Hackers & Painters* after one of the essays in it. I also worked on spam filters, and did some more painting. I used to have dinners for a group of friends every thursday night, which taught me how to cook for groups. And I bought another building in Cambridge, a former candy factory (and later, twas said, porn studio), to use as an office.

One night in October 2003 there was a big party at my house. It was a clever idea of my friend Maria Daniels, who was one of the thursday diners. Three separate hosts would all invite their friends to one party. So for every guest, two thirds of the other guests would be people they didn't know but would probably like. One of the guests was someone I didn't know but would turn out to like a lot: a woman called Jessica Livingston. A couple days later I asked her out.

Jessica was in charge of marketing at a Boston investment bank. This bank thought it understood startups, but over the next year, as she met friends of mine from the startup world, she was surprised how different reality was. And how colorful their stories were. So she decided to compile a book of [interviews](#) with startup founders.

When the bank had financial problems and she had to fire half her staff, she started looking for a new job. In early 2005 she interviewed for a marketing job at a Boston VC firm. It took them weeks to make up their minds, and during this time I started telling her about all the things that needed to be fixed about venture capital. They should make a larger number of smaller investments instead of a handful of giant ones, they should be funding younger, more technical founders instead of MBAs, they should let the founders remain as CEO, and so on.

One of my tricks for writing essays had always been to give talks. The prospect of having to stand up in front of a group of people and tell them something that won't waste their time is a great spur to the imagination. When the Harvard Computer Society, the undergrad computer club, asked me to give a talk, I decided I would tell them how to start a startup. Maybe they'd be able to avoid the worst of the mistakes we'd made.

So I gave this talk, in the course of which I told them that the best sources of seed funding were successful startup founders, because then they'd be sources of advice too. Whereupon it seemed they were all looking expectantly at me. Horrified at the prospect of having my inbox flooded by business plans (if I'd only known), I blurted out "But not me!" and went on with the talk. But afterward it occurred to me that I should really stop procrastinating about angel investing. I'd been meaning to since Yahoo bought us, and now it was 7 years later and I still hadn't done one angel investment.

Meanwhile I had been scheming with Robert and Trevor about projects we could

work on together. I missed working with them, and it seemed like there had to be something we could collaborate on.

As Jessica and I were walking home from dinner on March 11, at the corner of Garden and Walker streets, these three threads converged. Screw the VCs who were taking so long to make up their minds. We'd start our own investment firm and actually implement the ideas we'd been talking about. I'd fund it, and Jessica could quit her job and work for it, and we'd get Robert and Trevor as partners too.

[\[13\]](#)

Once again, ignorance worked in our favor. We had no idea how to be angel investors, and in Boston in 2005 there were no Ron Conways to learn from. So we just made what seemed like the obvious choices, and some of the things we did turned out to be novel.

There are multiple components to Y Combinator, and we didn't figure them all out at once. The part we got first was to be an angel firm. In those days, those two words didn't go together. There were VC firms, which were organized companies with people whose job it was to make investments, but they only did big, million dollar investments. And there were angels, who did smaller investments, but these were individuals who were usually focused on other things and made investments on the side. And neither of them helped founders enough in the beginning. We knew how helpless founders were in some respects, because we remembered how helpless we'd been. For example, one thing Julian had done for us that seemed to us like magic was to get us set up as a company. We were fine writing fairly difficult software, but actually getting incorporated, with bylaws and stock and all that stuff, how on earth did you do that? Our plan was not only to make seed investments, but to do for startups everything Julian had done for us.

YC was not organized as a fund. It was cheap enough to run that we funded it with our own money. That went right by 99% of readers, but professional investors are thinking "Wow, that means they got all the returns." But once again, this was not due to any particular insight on our part. We didn't know how VC firms were organized. It never occurred to us to try to raise a fund, and if it had, we wouldn't have known where to start. [\[14\]](#)

The most distinctive thing about YC is the batch model: to fund a bunch of startups all at once, twice a year, and then to spend three months focusing intensively on trying to help them. That part we discovered by accident, not merely implicitly but explicitly due to our ignorance about investing. We needed to get experience as investors. What better way, we thought, than to fund a whole bunch of startups at once? We knew undergrads got temporary jobs at tech companies during the summer. Why not organize a summer program where they'd start startups instead? We wouldn't feel guilty for being in a sense fake investors, because they would in a similar sense be fake founders. So while we probably wouldn't make much money out of it, we'd at least get to practice being investors on them, and they for their part would probably have a more interesting summer than they would working at Microsoft.

We'd use the building I owned in Cambridge as our headquarters. We'd all have dinner there once a week ♦ on tuesdays, since I was already cooking for the thursday diners on thursdays ♦ and after dinner we'd bring in experts on startups to give talks.

We knew undergrads were deciding then about summer jobs, so in a matter of days we cooked up something we called the Summer Founders Program, and I posted an [announcement](#) on my site, inviting undergrads to apply. I had never imagined that writing essays would be a way to get "deal flow," as investors call it, but it turned out to be the perfect source. [15] We got 225 applications for the Summer Founders Program, and we were surprised to find that a lot of them were from people who'd already graduated, or were about to that spring. Already this SFP thing was starting to feel more serious than we'd intended.

We invited about 20 of the 225 groups to interview in person, and from those we picked 8 to fund. They were an impressive group. That first batch included reddit, Justin Kan and Emmett Shear, who went on to found Twitch, Aaron Swartz, who had already helped write the RSS spec and would a few years later become a martyr for open access, and Sam Altman, who would later become the second president of YC. I don't think it was entirely luck that the first batch was so good. You had to be pretty bold to sign up for a weird thing like the Summer Founders Program instead of a summer job at a legit place like Microsoft or Goldman Sachs.

The deal for startups was based on a combination of the deal we did with Julian (\$10k for 10%) and what Robert said MIT grad students got for the summer (\$6k). We invested \$6k per founder, which in the typical two-founder case was \$12k, in return for 6%. That had to be fair, because it was twice as good as the deal we ourselves had taken. Plus that first summer, which was really hot, Jessica brought the founders free air conditioners. [16]

Fairly quickly I realized that we had stumbled upon the way to scale startup funding. Funding startups in batches was more convenient for us, because it meant we could do things for a lot of startups at once, but being part of a batch was better for the startups too. It solved one of the biggest problems faced by founders: the isolation. Now you not only had colleagues, but colleagues who understood the problems you were facing and could tell you how they were solving them.

As YC grew, we started to notice other advantages of scale. The alumni became a tight community, dedicated to helping one another, and especially the current batch, whose shoes they remembered being in. We also noticed that the startups were becoming one another's customers. We used to refer jokingly to the "YC GDP," but as YC grows this becomes less and less of a joke. Now lots of startups get their initial set of customers almost entirely from among their batchmates.

I had not originally intended YC to be a full-time job. I was going to do three things: hack, write essays, and work on YC. As YC grew, and I grew more excited



about it, it started to take up a lot more than a third of my attention. But for the first few years I was still able to work on other things.

In the summer of 2006, Robert and I started working on a new version of Arc. This one was reasonably fast, because it was compiled into Scheme. To test this new Arc, I wrote Hacker News in it. It was originally meant to be a news aggregator for startup founders and was called Startup News, but after a few months I got tired of reading about nothing but startups. Plus it wasn't startup founders we wanted to reach. It was future startup founders. So I changed the name to Hacker News and the topic to whatever engaged one's intellectual curiosity.

HN was no doubt good for YC, but it was also by far the biggest source of stress for me. If all I'd had to do was select and help founders, life would have been so easy. And that implies that HN was a mistake. Surely the biggest source of stress in one's work should at least be something close to the core of the work. Whereas I was like someone who was in pain while running a marathon not from the exertion of running, but because I had a blister from an ill-fitting shoe. When I was dealing with some urgent problem during YC, there was about a 60% chance it had to do with HN, and a 40% chance it had to do with everything else combined. [\[17\]](#)

As well as HN, I wrote all of YC's internal software in Arc. But while I continued to work a good deal *in* Arc, I gradually stopped working *on* Arc, partly because I didn't have time to, and partly because it was a lot less attractive to mess around with the language now that we had all this infrastructure depending on it. So now my three projects were reduced to two: writing essays and working on YC.

YC was different from other kinds of work I've done. Instead of deciding for myself what to work on, the problems came to me. Every 6 months there was a new batch of startups, and their problems, whatever they were, became our problems. It was very engaging work, because their problems were quite varied, and the good founders were very effective. If you were trying to learn the most you could about startups in the shortest possible time, you couldn't have picked a better way to do it.

There were parts of the job I didn't like. Disputes between cofounders, figuring out when people were lying to us, fighting with people who maltreated the startups, and so on. But I worked hard even at the parts I didn't like. I was haunted by something Kevin Hale once said about companies: "No one works harder than the boss." He meant it both descriptively and prescriptively, and it was the second part that scared me. I wanted YC to be good, so if how hard I worked set the upper bound on how hard everyone else worked, I'd better work very hard.

One day in 2010, when he was visiting California for interviews, Robert Morris did something astonishing: he offered me unsolicited advice. I can only remember him doing that once before. One day at Viaweb, when I was bent over double from a kidney stone, he suggested that it would be a good idea for him to take me to the hospital. That was what it took for Rtm to offer unsolicited advice. So I remember his exact words very clearly. "You know," he said, "you should make sure Y

Combinator isn't the last cool thing you do."

At the time I didn't understand what he meant, but gradually it dawned on me that he was saying I should quit. This seemed strange advice, because YC was doing great. But if there was one thing rarer than Rtm offering advice, it was Rtm being wrong. So this set me thinking. It was true that on my current trajectory, YC would be the last thing I did, because it was only taking up more of my attention. It had already eaten Arc, and was in the process of eating essays too. Either YC was my life's work or I'd have to leave eventually. And it wasn't, so I would.

In the summer of 2012 my mother had a stroke, and the cause turned out to be a blood clot caused by colon cancer. The stroke destroyed her balance, and she was put in a nursing home, but she really wanted to get out of it and back to her house, and my sister and I were determined to help her do it. I used to fly up to Oregon to visit her regularly, and I had a lot of time to think on those flights. On one of them I realized I was ready to hand YC over to someone else.

I asked Jessica if she wanted to be president, but she didn't, so we decided we'd try to recruit Sam Altman. We talked to Robert and Trevor and we agreed to make it a complete changing of the guard. Up till that point YC had been controlled by the original LLC we four had started. But we wanted YC to last for a long time, and to do that it couldn't be controlled by the founders. So if Sam said yes, we'd let him reorganize YC. Robert and I would retire, and Jessica and Trevor would become ordinary partners.

When we asked Sam if he wanted to be president of YC, initially he said no. He wanted to start a startup to make nuclear reactors. But I kept at it, and in October 2013 he finally agreed. We decided he'd take over starting with the winter 2014 batch. For the rest of 2013 I left running YC more and more to Sam, partly so he could learn the job, and partly because I was focused on my mother, whose cancer had returned.

She died on January 15, 2014. We knew this was coming, but it was still hard when it did.

I kept working on YC till March, to help get that batch of startups through Demo Day, then I checked out pretty completely. (I still talk to alumni and to new startups working on things I'm interested in, but that only takes a few hours a week.)

What should I do next? Rtm's advice hadn't included anything about that. I wanted to do something completely different, so I decided I'd paint. I wanted to see how good I could get if I really focused on it. So the day after I stopped working on YC, I started painting. I was rusty and it took a while to get back into shape, but it was at least completely engaging. [\[18\]](#)

I spent most of the rest of 2014 painting. I'd never been able to work so uninterruptedly before, and I got to be better than I had been. Not good enough,

but better. Then in November, right in the middle of a painting, I ran out of steam. Up till that point I'd always been curious to see how the painting I was working on would turn out, but suddenly finishing this one seemed like a chore. So I stopped working on it and cleaned my brushes and haven't painted since. So far anyway.

I realize that sounds rather wimpy. But attention is a zero sum game. If you can choose what to work on, and you choose a project that's not the best one (or at least a good one) for you, then it's getting in the way of another project that is. And at 50 there was some opportunity cost to screwing around.

I started writing essays again, and wrote a bunch of new ones over the next few months. I even wrote a couple that [weren't](#) about startups. Then in March 2015 I started working on Lisp again.

The distinctive thing about Lisp is that its core is a language defined by writing an interpreter in itself. It wasn't originally intended as a programming language in the ordinary sense. It was meant to be a formal model of computation, an alternative to the Turing machine. If you want to write an interpreter for a language in itself, what's the minimum set of predefined operators you need? The Lisp that John McCarthy invented, or more accurately discovered, is an answer to that question.

[19]

McCarthy didn't realize this Lisp could even be used to program computers till his grad student Steve Russell suggested it. Russell translated McCarthy's interpreter into IBM 704 machine language, and from that point Lisp started also to be a programming language in the ordinary sense. But its origins as a model of computation gave it a power and elegance that other languages couldn't match. It was this that attracted me in college, though I didn't understand why at the time.

McCarthy's 1960 Lisp did nothing more than interpret Lisp expressions. It was missing a lot of things you'd want in a programming language. So these had to be added, and when they were, they weren't defined using McCarthy's original axiomatic approach. That wouldn't have been feasible at the time. McCarthy tested his interpreter by hand-simulating the execution of programs. But it was already getting close to the limit of interpreters you could test that way ♦ indeed, there was a bug in it that McCarthy had overlooked. To test a more complicated interpreter, you'd have had to run it, and computers then weren't powerful enough.

Now they are, though. Now you could continue using McCarthy's axiomatic approach till you'd defined a complete programming language. And as long as every change you made to McCarthy's Lisp was a discoveredness-preserving transformation, you could, in principle, end up with a complete language that had this quality. Harder to do than to talk about, of course, but if it was possible in principle, why not try? So I decided to take a shot at it. It took 4 years, from March 26, 2015 to October 12, 2019. It was fortunate that I had a precisely defined goal, or it would have been hard to keep at it for so long.

I wrote this new Lisp, called [Bel](#), in itself in Arc. That may sound like a

contradiction, but it's an indication of the sort of trickery I had to engage in to make this work. By means of an egregious collection of hacks I managed to make something close enough to an interpreter written in itself that could actually run. Not fast, but fast enough to test.

I had to ban myself from writing essays during most of this time, or I'd never have finished. In late 2015 I spent 3 months writing essays, and when I went back to working on Bel I could barely understand the code. Not so much because it was badly written as because the problem is so convoluted. When you're working on an interpreter written in itself, it's hard to keep track of what's happening at what level, and errors can be practically encrypted by the time you get them.

So I said no more essays till Bel was done. But I told few people about Bel while I was working on it. So for years it must have seemed that I was doing nothing, when in fact I was working harder than I'd ever worked on anything. Occasionally after wrestling for hours with some gruesome bug I'd check Twitter or HN and see someone asking "Does Paul Graham still code?"

Working on Bel was hard but satisfying. I worked on it so intensively that at any given time I had a decent chunk of the code in my head and could write more there. I remember taking the boys to the coast on a sunny day in 2015 and figuring out how to deal with some problem involving continuations while I watched them play in the tide pools. It felt like I was doing life right. I remember that because I was slightly dismayed at how novel it felt. The good news is that I had more moments like this over the next few years.

In the summer of 2016 we moved to England. We wanted our kids to see what it was like living in another country, and since I was a British citizen by birth, that seemed the obvious choice. We only meant to stay for a year, but we liked it so much that we still live there. So most of Bel was written in England.

In the fall of 2019, Bel was finally finished. Like McCarthy's original Lisp, it's a spec rather than an implementation, although like McCarthy's Lisp it's a spec expressed as code.

Now that I could write essays again, I wrote a bunch about topics I'd had stacked up. I kept writing essays through 2020, but I also started to think about other things I could work on. How should I choose what to do? Well, how had I chosen what to work on in the past? I wrote an essay for myself to answer that question, and I was surprised how long and messy the answer turned out to be. If this surprised me, who'd lived it, then I thought perhaps it would be interesting to other people, and encouraging to those with similarly messy lives. So I wrote a more detailed version for others to read, and this is the last sentence of it.

## Notes

[1] My experience skipped a step in the evolution of computers: time-sharing machines with interactive OSes. I went straight from batch processing to microcomputers, which made microcomputers seem all the more exciting.

[2] Italian words for abstract concepts can nearly always be predicted from their English cognates (except for occasional traps like *polluzione*). It's the everyday words that differ. So if you string together a lot of abstract concepts with a few simple verbs, you can make a little Italian go a long way.

[3] I lived at Piazza San Felice 4, so my walk to the Accademia went straight down the spine of old Florence: past the Pitti, across the bridge, past Orsanmichele, between the Duomo and the Baptistery, and then up Via Ricasoli to Piazza San Marco. I saw Florence at street level in every possible condition, from empty dark winter evenings to sweltering summer days when the streets were packed with tourists.

[4] You can of course paint people like still lives if you want to, and they're willing. That sort of portrait is arguably the apex of still life painting, though the long sitting does tend to produce pained expressions in the sitters.

[5] Interleaf was one of many companies that had smart people and built impressive technology, and yet got crushed by Moore's Law. In the 1990s the exponential growth in the power of commodity (i.e. Intel) processors rolled up high-end, special-purpose hardware and software companies like a bulldozer.

[6] The signature style seekers at RISD weren't specifically mercenary. In the art world, money and coolness are tightly coupled. Anything expensive comes to be seen as cool, and anything seen as cool will soon become equally expensive.

[7] Technically the apartment wasn't rent-controlled but rent-stabilized, but this is a refinement only New Yorkers would know or care about. The point is that it was really cheap, less than half market price.

[8] Most software you can launch as soon as it's done. But when the software is an online store builder and you're hosting the stores, if you don't have any users yet, that fact will be painfully obvious. So before we could launch publicly we had to launch privately, in the sense of recruiting an initial set of users and making sure they had decent-looking stores.

[9] We'd had a code editor in Viaweb for users to define their own page styles. They didn't know it, but they were editing Lisp expressions underneath. But this wasn't an app editor, because the code ran when the merchants' sites were

generated, not when shoppers visited them.

[10] This was the first instance of what is now a familiar experience, and so was what happened next, when I read the comments and found they were full of angry people. How could I claim that Lisp was better than other languages? Weren't they all Turing complete? People who see the responses to essays I write sometimes tell me how sorry they feel for me, but I'm not exaggerating when I reply that it has always been like this, since the very beginning. It comes with the territory. An essay must tell readers things they [don't already know](#), and some people dislike being told such things.

[11] People put plenty of stuff on the internet in the 90s of course, but putting something online is not the same as publishing it online. Publishing online means you treat the online version as the (or at least a) primary version.

[12] There is a general lesson here that our experience with Y Combinator also teaches: Customs continue to constrain you long after the restrictions that caused them have disappeared. Customary VC practice had once, like the customs about publishing essays, been based on real constraints. Startups had once been much more expensive to start, and proportionally rare. Now they could be cheap and common, but the VCs' customs still reflected the old world, just as customs about writing essays still reflected the constraints of the print era.

Which in turn implies that people who are independent-minded (i.e. less influenced by custom) will have an advantage in fields affected by rapid change (where customs are more likely to be obsolete).

Here's an interesting point, though: you can't always predict which fields will be affected by rapid change. Obviously software and venture capital will be, but who would have predicted that essay writing would be?

[13] Y Combinator was not the original name. At first we were called Cambridge Seed. But we didn't want a regional name, in case someone copied us in Silicon Valley, so we renamed ourselves after one of the coolest tricks in the lambda calculus, the Y combinator.

I picked orange as our color partly because it's the warmest, and partly because no VC used it. In 2005 all the VCs used staid colors like maroon, navy blue, and forest green, because they were trying to appeal to LPs, not founders. The YC logo itself is an inside joke: the Viaweb logo had been a white V on a red circle, so I made the YC logo a white Y on an orange square.

[14] YC did become a fund for a couple years starting in 2009, because it was getting so big I could no longer afford to fund it personally. But after Heroku got bought we had enough money to go back to being self-funded.

[15] I've never liked the term "deal flow," because it implies that the number of new startups at any given time is fixed. This is not only false, but it's the purpose

of YC to falsify it, by causing startups to be founded that would not otherwise have existed.

[16] She reports that they were all different shapes and sizes, because there was a run on air conditioners and she had to get whatever she could, but that they were all heavier than she could carry now.

[17] Another problem with HN was a bizarre edge case that occurs when you both write essays and run a forum. When you run a forum, you're assumed to see if not every conversation, at least every conversation involving you. And when you write essays, people post highly imaginative misinterpretations of them on forums. Individually these two phenomena are tedious but bearable, but the combination is disastrous. You actually have to respond to the misinterpretations, because the assumption that you're present in the conversation means that not responding to any sufficiently upvoted misinterpretation reads as a tacit admission that it's correct. But that in turn encourages more; anyone who wants to pick a fight with you senses that now is their chance.

[18] The worst thing about leaving YC was not working with Jessica anymore. We'd been working on YC almost the whole time we'd known each other, and we'd neither tried nor wanted to separate it from our personal lives, so leaving was like pulling up a deeply rooted tree.

[19] One way to get more precise about the concept of invented vs discovered is to talk about space aliens. Any sufficiently advanced alien civilization would certainly know about the Pythagorean theorem, for example. I believe, though with less certainty, that they would also know about the Lisp in McCarthy's 1960 paper.

But if so there's no reason to suppose that this is the limit of the language that might be known to them. Presumably aliens need numbers and errors and I/O too. So it seems likely there exists at least one path out of McCarthy's Lisp along which discoveredness is preserved.

**Thanks** to Trevor Blackwell, John Collison, Patrick Collison, Daniel Gackle, Ralph Hazell, Jessica Livingston, Robert Morris, and Harj Taggar for reading drafts of this.

# Donate Unrestricted

March 2021

The secret curse of the nonprofit world is restricted donations. If you haven't been involved with nonprofits, you may never have heard this phrase before. But if you have been, it probably made you wince.

Restricted donations mean donations where the donor limits what can be done with the money. This is common with big donations, perhaps the default. And yet it's usually a bad idea. Usually the way the donor wants the money spent is not the way the nonprofit would have chosen. Otherwise there would have been no need to restrict the donation. But who has a better understanding of where money needs to be spent, the nonprofit or the donor?

If a nonprofit doesn't understand better than its donors where money needs to be spent, then it's incompetent and you shouldn't be donating to it at all.

Which means a restricted donation is inherently suboptimal. It's either a donation to a bad nonprofit, or a donation for the wrong things.

There are a couple exceptions to this principle. One is when the nonprofit is an umbrella organization. It's reasonable to make a restricted donation to a university, for example, because a university is only nominally a single nonprofit. Another exception is when the donor actually does know as much as the nonprofit about where money needs to be spent. The Gates Foundation, for example, has specific goals and often makes restricted donations to individual nonprofits to accomplish them. But unless you're a domain expert yourself or donating to an umbrella organization, your donation would do more good if it were unrestricted.

If restricted donations do less good than unrestricted ones, why do donors so often make them? Partly because doing good isn't donors' only motive. They often have other motives as well — to make a mark, or to generate good publicity [\[1\]](#), or to comply with regulations or corporate policies. Many donors may simply never have considered the distinction between restricted and unrestricted donations. They may believe that donating money for some specific purpose is just how donation works. And to be fair, nonprofits don't try very hard to discourage such illusions. They can't afford to. People running nonprofits are almost always anxious about money. They can't afford to talk back to big donors.



You can't expect candor in a relationship so asymmetric. So I'll tell you what nonprofits wish they could tell you. If you want to donate to a nonprofit, donate unrestricted. If you trust them to spend your money, trust them to decide how.

## **Note**

[1] Unfortunately restricted donations tend to generate more publicity than unrestricted ones. "X donates money to build a school in Africa" is not only more interesting than "X donates money to Y nonprofit to spend as Y chooses," but also focuses more attention on X.

**Thanks** to Chase Adam, Ingrid Bassett, Trevor Blackwell, and Edith Elliot for reading drafts of this.

# Write Simply

March 2021

I try to write using ordinary words and simple sentences.

That kind of writing is easier to read, and the easier something is to read, the more deeply readers will engage with it. The less energy they expend on your prose, the more they'll have left for your ideas.

And the further they'll read. Most readers' energy tends to flag part way through an article or essay. If the friction of reading is low enough, more keep going till the end.

There's an Italian dish called *saltimbocca*, which means "leap into the mouth." My goal when writing might be called *saltintesta*: the ideas leap into your head and you barely notice the words that got them there.

It's too much to hope that writing could ever be pure ideas. You might not even want it to be. But for most writers, most of the time, that's the goal to aim for. The gap between most writing and pure ideas is not filled with poetry.

Plus it's more considerate to write simply. When you write in a fancy way to impress people, you're making them do extra work just so you can seem cool. It's like trailing a long train behind you that readers have to carry.

And remember, if you're writing in English, that a lot of your readers won't be native English speakers. Their understanding of ideas may be way ahead of their understanding of English. So you can't assume that writing about a difficult topic means you can use difficult words.

Of course, fancy writing doesn't just conceal ideas. It can also conceal the lack of them. That's why some people write that way, to conceal the fact that they have nothing to say. Whereas writing simply keeps you honest. If you say nothing simply, it will be obvious to everyone, including you.

Simple writing also lasts better. People reading your stuff in the future will be in much the same position as people from other countries reading it today. The culture and the language will have changed. It's not vain to care about that, any more than it's vain for a woodworker to build a chair to last.

Indeed, lasting is not merely an accidental quality of chairs, or writing. It's a sign you did a good job.

But although these are all real advantages of writing simply, none of them are why I do it. The main reason I write simply is that it offends me not to. When I write a sentence that seems too complicated, or that uses unnecessarily intellectual words, it doesn't seem fancy to me. It seems clumsy.

There are of course times when you want to use a complicated sentence or fancy word for effect. But you should never do it by accident.

The other reason my writing ends up being simple is the way I do it. I write the first draft fast, then spend days editing it, trying to get everything just right. Much of this editing is cutting, and that makes simple writing even simpler.

# How People Get Rich Now

April 2021

Every year since 1982, *Forbes* magazine has published a list of the richest Americans. If we compare the 100 richest people in 1982 to the 100 richest in 2020, we notice some big differences.

In 1982 the most common source of wealth was inheritance. Of the 100 richest people, 60 inherited from an ancestor. There were 10 du Pont heirs alone. By 2020 the number of heirs had been cut in half, accounting for only 27 of the biggest 100 fortunes.

Why would the percentage of heirs decrease? Not because inheritance taxes increased. In fact, they decreased significantly during this period. The reason the percentage of heirs has decreased is not that fewer people are inheriting great fortunes, but that more people are making them.

How are people making these new fortunes? Roughly 3/4 by starting companies and 1/4 by investing. Of the 73 new fortunes in 2020, 56 derive from founders' or early employees' equity (52 founders, 2 early employees, and 2 wives of founders), and 17 from managing investment funds.

There were no fund managers among the 100 richest Americans in 1982. Hedge funds and private equity firms existed in 1982, but none of their founders were rich enough yet to make it into the top 100. Two things changed: fund managers discovered new ways to generate high returns, and more investors were willing to trust them with their money. [\[1\]](#)

But the main source of new fortunes now is starting companies, and when you look at the data, you see big changes there too. People get richer from starting companies now than they did in 1982, because the companies do different things.

In 1982, there were two dominant sources of new wealth: oil and real estate. Of the 40 new fortunes in 1982, at least 24 were due primarily to oil or real estate. Now only a small number are: of the 73 new fortunes in 2020, 4 were due to real estate and only 2 to oil.

By 2020 the biggest source of new wealth was what are sometimes called "tech" companies. Of the 73 new fortunes, about 30 derive from such companies. These

are particularly common among the richest of the rich: 8 of the top 10 fortunes in 2020 were new fortunes of this type.

Arguably it's slightly misleading to treat tech as a category. Isn't Amazon really a retailer, and Tesla a car maker? Yes and no. Maybe in 50 years, when what we call tech is taken for granted, it won't seem right to put these two businesses in the same category. But at the moment at least, there is definitely something they share in common that distinguishes them. What retailer starts AWS? What car maker is run by someone who also has a rocket company?

The tech companies behind the top 100 fortunes also form a well-differentiated group in the sense that they're all companies that venture capitalists would readily invest in, and the others mostly not. And there's a reason why: these are mostly companies that win by having better technology, rather than just a CEO who's really driven and good at making deals.

To that extent, the rise of the tech companies represents a qualitative change. The oil and real estate magnates of the 1982 Forbes 400 didn't win by making better technology. They won by being really driven and good at making deals. [2] And indeed, that way of getting rich is so old that it predates the Industrial Revolution. The courtiers who got rich in the (nominal) service of European royal houses in the 16th and 17th centuries were also, as a rule, really driven and good at making deals.

People who don't look any deeper than the Gini coefficient look back on the world of 1982 as the good old days, because those who got rich then didn't get as rich. But if you dig into *how* they got rich, the old days don't look so good. In 1982, 84% of the richest 100 people got rich by inheritance, extracting natural resources, or doing real estate deals. Is that really better than a world in which the richest people get rich by starting tech companies?

Why are people starting so many more new companies than they used to, and why are they getting so rich from it? The answer to the first question, curiously enough, is that it's misphrased. We shouldn't be asking why people are starting companies, but why they're starting companies *again*. [3]

In 1892, the *New York Herald Tribune* compiled a list of all the millionaires in America. They found 4047 of them. How many had inherited their wealth then? Only about 20%, which is less than the proportion of heirs today. And when you investigate the sources of the new fortunes, 1892 looks even more like today. Hugh Rockoff found that "many of the richest ... gained their initial edge from the new technology of mass production." [4]

So it's not 2020 that's the anomaly here, but 1982. The real question is why so few people had gotten rich from starting companies in 1982. And the answer is that even as the *Herald Tribune*'s list was being compiled, a wave of [consolidation](#) was sweeping through the American economy. In the late 19th and early 20th centuries, financiers like J. P. Morgan combined thousands of smaller companies

into a few hundred giant ones with commanding economies of scale. By the end of World War II, as Michael Lind writes, "the major sectors of the economy were either organized as government-backed cartels or dominated by a few oligopolistic corporations." [5]

In 1960, most of the people who start startups today would have gone to work for one of them. You could get rich from starting your own company in 1890 and in 2020, but in 1960 it was not really a viable option. You couldn't break through the oligopolies to get at the markets. So the prestigious route in 1960 was not to start your own company, but to work your way up the corporate ladder at an existing one. [6]

Making everyone a corporate employee decreased economic inequality (and every other kind of variation), but if your model of normal is the mid 20th century, you have a very misleading model in that respect. J. P. Morgan's economy turned out to be just a phase, and starting in the 1970s, it began to break up.

Why did it break up? Partly senescence. The big companies that seemed models of scale and efficiency in 1930 had by 1970 become slack and bloated. By 1970 the rigid structure of the economy was full of cosy nests that various groups had built to insulate themselves from market forces. During the Carter administration the federal government realized something was amiss and began, in a process they called "deregulation," to roll back the policies that propped up the oligopolies.

But it wasn't just decay from within that broke up J. P. Morgan's economy. There was also pressure from without, in the form of new technology, and particularly microelectronics. The best way to envision what happened is to imagine a pond with a crust of ice on top. Initially the only way from the bottom to the surface is around the edges. But as the ice crust weakens, you start to be able to punch right through the middle.

The edges of the pond were pure tech: companies that actually described themselves as being in the electronics or software business. When you used the word "startup" in 1990, that was what you meant. But now startups are punching right through the middle of the ice crust and displacing incumbents like retailers and TV networks and car companies. [7]

But though the breakup of J. P. Morgan's economy created a new world in the technological sense, it was a reversion to the norm in the social sense. If you only look back as far as the mid 20th century, it seems like people getting rich by starting their own companies is a recent phenomenon. But if you look back further, you realize it's actually the default. So what we should expect in the future is more of the same. Indeed, we should expect both the number and wealth of founders to grow, because every decade it gets easier to start a startup.

Part of the reason it's getting easier to start a startup is social. Society is (re)assimilating the concept. If you start one now, your parents won't freak out the way they would have a generation ago, and knowledge about how to do it is much

more widespread. But the main reason it's easier to start a startup now is that it's cheaper. Technology has driven down the cost of both building products and acquiring customers.

The decreasing cost of starting a startup has in turn changed the balance of power between founders and investors. Back when starting a startup meant building a factory, you needed investors' permission to do it at all. But now investors need founders more than founders need investors, and that, combined with the increasing amount of venture capital available, has driven up valuations. [8]

So the decreasing cost of starting a startup increases the number of rich people in two ways: it means that more people start them, and that those who do can raise money on better terms.

But there's also a third factor at work: the companies themselves are more valuable, because newly founded companies grow faster than they used to. Technology hasn't just made it cheaper to build and distribute things, but faster too.

This trend has been running for a long time. IBM, founded in 1896, took 45 years to reach a billion 2020 dollars in revenue. Hewlett-Packard, founded in 1939, took 25 years. Microsoft, founded in 1975, took 13 years. Now the norm for fast-growing companies is 7 or 8 years. [9]

Fast growth has a double effect on the value of founders' stock. The value of a company is a function of its revenue and its growth rate. So if a company grows faster, you not only get to a billion dollars in revenue sooner, but the company is more valuable when it reaches that point than it would be if it were growing slower.

That's why founders sometimes get so rich so young now. The low initial cost of starting a startup means founders can start young, and the fast growth of companies today means that if they succeed they could be surprisingly rich just a few years later.

It's easier now to start and grow a company than it has ever been. That means more people start them, that those who do get better terms from investors, and that the resulting companies become more valuable. Once you understand how these mechanisms work, and that startups were suppressed for most of the 20th century, you don't have to resort to some vague right turn the country took under Reagan to explain why America's Gini coefficient is increasing. Of course the Gini coefficient is increasing. With more people starting more valuable companies, how could it not be?

## Notes

[1] Investment firms grew rapidly after a regulatory change by the Labor Department in 1978 allowed pension funds to invest in them, but the effects of this growth were not yet visible in the top 100 fortunes in 1982.

[2] George Mitchell deserves mention as an exception. Though really driven and good at making deals, he was also the first to figure out how to use fracking to get natural gas out of shale.

[3] When I say people are starting more companies, I mean the type of company meant to [grow](#) very big. There has actually been a decrease in the last couple decades in the overall number of new companies. But the vast majority of companies are small retail and service businesses. So what the statistics about the decreasing number of new businesses mean is that people are starting fewer shoe stores and barber shops.

People sometimes get [confused](#) when they see a graph labelled "startups" that's going down, because there are two senses of the word "startup": (1) the founding of a company, and (2) a particular type of company designed to grow big fast. The statistics mean startup in sense (1), not sense (2).

[4] Rockoff, Hugh. "Great Fortunes of the Gilded Age." NBER Working Paper 14555, 2008.

[5] Lind, Michael. *Land of Promise*. HarperCollins, 2012.

It's also likely that the high tax rates in the mid 20th century deterred people from starting their own companies. Starting one's own company is risky, and when risk isn't rewarded, people opt for [safety](#) instead.

But it wasn't simply cause and effect. The oligopolies and high tax rates of the mid 20th century were all of a piece. Lower taxes are not just a cause of entrepreneurship, but an effect as well: the people getting rich in the mid 20th century from real estate and oil exploration lobbied for and got huge tax loopholes that made their effective tax rate much lower, and presumably if it had been more common to grow big companies by building new technology, the people doing that would have lobbied for their own loopholes as well.

[6] That's why the people who did get rich in the mid 20th century so often got rich from oil exploration or real estate. Those were the two big areas of the economy that weren't susceptible to consolidation.



[7] The pure tech companies used to be called "high technology" startups. But now that startups can punch through the middle of the ice crust, we don't need a separate name for the edges, and the term "high-tech" has a decidedly [retro](#) sound.

[8] Higher valuations mean you either sell less stock to get a given amount of money, or get more money for a given amount of stock. The typical startup does some of each. Obviously you end up richer if you keep more stock, but you should also end up richer if you raise more money, because (a) it should make the company more successful, and (b) you should be able to last longer before the next round, or not even need one. Notice all those shoulds though. In practice a lot of money slips through them.

It might seem that the huge rounds raised by startups nowadays contradict the claim that it has become cheaper to start one. But there's no contradiction here; the startups that raise the most are the ones doing it by choice, in order to grow faster, not the ones doing it because they need the money to survive. There's nothing like not needing money to make people offer it to you.

You would think, after having been on the side of labor in its fight with capital for almost two centuries, that the far left would be happy that labor has finally prevailed. But none of them seem to be. You can almost hear them saying "No, no, not *that* way."

[9] IBM was created in 1911 by merging three companies, the most important of which was Herman Hollerith's Tabulating Machine Company, founded in 1896. In 1941 its revenues were \$60 million.

Hewlett-Packard's revenues in 1964 were \$125 million.

Microsoft's revenues in 1988 were \$590 million.

**Thanks** to Trevor Blackwell, Jessica Livingston, Bob Lesko, Robert Morris, Russ Roberts, and Alex Tabarrok for reading drafts of this, and to Jon Erlichman for growth data.

# The Real Reason to End the Death Penalty

April 2021

When intellectuals talk about the death penalty, they talk about things like whether it's permissible for the state to take someone's life, whether the death penalty acts as a deterrent, and whether more death sentences are given to some groups than others. But in practice the debate about the death penalty is not about whether it's ok to kill murderers. It's about whether it's ok to kill innocent people, because at least 4% of people on death row are [innocent](#).

When I was a kid I imagined that it was unusual for people to be convicted of crimes they hadn't committed, and that in murder cases especially this must be very rare. Far from it. Now, thanks to organizations like the [Innocence Project](#), we see a constant stream of stories about murder convictions being overturned after new evidence emerges. Sometimes the police and prosecutors were just very sloppy. Sometimes they were crooked, and knew full well they were convicting an innocent person.

Kenneth Adams and three other men spent 18 years in prison on a murder conviction. They were exonerated after DNA testing implicated three different men, two of whom later confessed. The police had been told about the other men early in the investigation, but never followed up the lead.

Keith Harward spent 33 years in prison on a murder conviction. He was convicted because "experts" said his teeth matched photos of bite marks on one victim. He was exonerated after DNA testing showed the murder had been committed by another man, Jerry Crotty.

Ricky Jackson and two other men spent 39 years in prison after being convicted of murder on the testimony of a 12 year old boy, who later recanted and said he'd been coerced by police. Multiple people have confirmed the boy was elsewhere at the time. The three men were exonerated after the county prosecutor dropped the charges, saying "The state is conceding the obvious."

Alfred Brown spent 12 years in prison on a murder conviction, including 10 years on death row. He was exonerated after it was discovered that the assistant district attorney had concealed phone records proving he could not have committed the

crimes.

Glenn Ford spent 29 years on death row after having been convicted of murder. He was exonerated after new evidence proved he was not even at the scene when the murder occurred. The attorneys assigned to represent him had never tried a jury case before.

Cameron Willingham was actually executed in 2004 by lethal injection. The "expert" who testified that he deliberately set fire to his house has since been discredited. A re-examination of the case ordered by the state of Texas in 2009 concluded that "a finding of arson could not be sustained."

[Rich Glossip](#) has spent 20 years on death row after being convicted of murder on the testimony of the actual killer, who escaped with a life sentence in return for implicating him. In 2015 he came within minutes of execution before it emerged that Oklahoma had been planning to kill him with an illegal combination of drugs. They still plan to go ahead with the execution, perhaps as soon as this summer, despite [new evidence](#) exonerating him.

I could go on. There are hundreds of similar cases. In Florida alone, 29 death row prisoners have been exonerated so far.

Far from being rare, wrongful murder convictions are [very common](#). Police are under pressure to solve a crime that has gotten a lot of attention. When they find a suspect, they want to believe he's guilty, and ignore or even destroy evidence suggesting otherwise. District attorneys want to be seen as effective and tough on crime, and in order to win convictions are willing to manipulate witnesses and withhold evidence. Court-appointed defense attorneys are overworked and often incompetent. There's a ready supply of criminals willing to give false testimony in return for a lighter sentence, suggestible witnesses who can be made to say whatever police want, and bogus "experts" eager to claim that science proves the defendant is guilty. And juries want to believe them, since otherwise some terrible crime remains unsolved.

This circus of incompetence and dishonesty is the real issue with the death penalty. We don't even reach the point where theoretical questions about the moral justification or effectiveness of capital punishment start to matter, because so many of the people sentenced to death are actually innocent. Whatever it means in theory, in practice capital punishment means killing innocent people.

**Thanks** to Trevor Blackwell, Jessica Livingston, and Don Knight for reading drafts of this.

**Related:**

[Will Florida Kill an Innocent Man?](#)

[Was Kevin Cooper Framed for Murder?](#)

[Did Texas execute an innocent man?](#)

# An NFT That Saves Lives

May 2021

[Noora Health](#), a nonprofit I've supported for years, just launched a new NFT. It has a dramatic name, [Save Thousands of Lives](#), because that's what the proceeds will do.

Noora has been saving lives for 7 years. They run programs in hospitals in South Asia to teach new mothers how to take care of their babies once they get home. They're in 165 hospitals now. And because they know the numbers before and after they start at a new hospital, they can measure the impact they have. It is massive. For every 1000 live births, they save 9 babies.

This number comes from a [study](#) of 133,733 families at 28 different hospitals that Noora conducted in collaboration with the Better Birth team at Ariadne Labs, a joint center for health systems innovation at Brigham and Women's Hospital and Harvard T.H. Chan School of Public Health.

Noora is so effective that even if you measure their costs in the most conservative way, by dividing their entire budget by the number of lives saved, the cost of saving a life is the lowest I've seen. \$1,235.

For this NFT, they're going to issue a public report tracking how this specific tranche of money is spent, and estimating the number of lives saved as a result.

NFTs are a new territory, and this way of using them is especially new, but I'm excited about its potential. And I'm excited to see what happens with this particular auction, because unlike an NFT representing something that has already happened, this NFT gets better as the price gets higher.

The reserve price was about \$2.5 million, because that's what it takes for the name to be accurate: that's what it costs to save 2000 lives. But the higher the price of this NFT goes, the more lives will be saved. What a sentence to be able to write.

# Crazy New Ideas

May 2021

There's one kind of opinion I'd be very afraid to express publicly. If someone I knew to be both a domain expert and a reasonable person proposed an idea that sounded preposterous, I'd be very reluctant to say "That will never work."

Anyone who has studied the history of ideas, and especially the history of science, knows that's how big things start. Someone proposes an idea that sounds crazy, most people dismiss it, then it gradually takes over the world.

Most implausible-sounding ideas are in fact bad and could be safely dismissed. But not when they're proposed by reasonable domain experts. If the person proposing the idea is reasonable, then they know how implausible it sounds. And yet they're proposing it anyway. That suggests they know something you don't. And if they have deep domain expertise, that's probably the source of it. [\[1\]](#)

Such ideas are not merely unsafe to dismiss, but disproportionately likely to be interesting. When the average person proposes an implausible-sounding idea, its implausibility is evidence of their incompetence. But when a reasonable domain expert does it, the situation is reversed. There's something like an efficient market here: on average the ideas that seem craziest will, if correct, have the biggest effect. So if you can eliminate the theory that the person proposing an implausible-sounding idea is incompetent, its implausibility switches from evidence that it's boring to evidence that it's exciting. [\[2\]](#)

Such ideas are not guaranteed to work. But they don't have to be. They just have to be sufficiently good bets — to have sufficiently high expected value. And I think on average they do. I think if you bet on the entire set of implausible-sounding ideas proposed by reasonable domain experts, you'd end up net ahead.

The reason is that everyone is too conservative. The word "paradigm" is overused, but this is a case where it's warranted. Everyone is too much in the grip of the current paradigm. Even the people who have the new ideas undervalue them initially. Which means that before they reach the stage of proposing them publicly, they've already subjected them to an excessively strict filter. [\[3\]](#)

The wise response to such an idea is not to make statements, but to ask questions, because there's a real mystery here. Why has this smart and reasonable

person proposed an idea that seems so wrong? Are they mistaken, or are you? One of you has to be. If you're the one who's mistaken, that would be good to know, because it means there's a hole in your model of the world. But even if they're mistaken, it should be interesting to learn why. A trap that an expert falls into is one you have to worry about too.

This all seems pretty obvious. And yet there are clearly a lot of people who don't share my fear of dismissing new ideas. Why do they do it? Why risk looking like a jerk now and a fool later, instead of just reserving judgement?

One reason they do it is envy. If you propose a radical new idea and it succeeds, your reputation (and perhaps also your wealth) will increase proportionally. Some people would be envious if that happened, and this potential envy propagates back into a conviction that you must be wrong.

Another reason people dismiss new ideas is that it's an easy way to seem sophisticated. When a new idea first emerges, it usually seems pretty feeble. It's a mere hatchling. Received wisdom is a full-grown eagle by comparison. So it's easy to launch a devastating attack on a new idea, and anyone who does will seem clever to those who don't understand this asymmetry.

This phenomenon is exacerbated by the difference between how those working on new ideas and those attacking them are rewarded. The rewards for working on new ideas are weighted by the value of the outcome. So it's worth working on something that only has a 10% chance of succeeding if it would make things more than 10x better. Whereas the rewards for attacking new ideas are roughly constant; such attacks seem roughly equally clever regardless of the target.

People will also attack new ideas when they have a vested interest in the old ones. It's not surprising, for example, that some of Darwin's harshest critics were churchmen. People build whole careers on some ideas. When someone claims they're false or obsolete, they feel threatened.

The lowest form of dismissal is mere factionalism: to automatically dismiss any idea associated with the opposing faction. The lowest form of all is to dismiss an idea because of who proposed it.

But the main thing that leads reasonable people to dismiss new ideas is the same thing that holds people back from proposing them: the sheer pervasiveness of the current paradigm. It doesn't just affect the way we think; it is the Lego blocks we build thoughts out of. Popping out of the current paradigm is something only a few people can do. And even they usually have to suppress their intuitions at first, like a pilot flying through cloud who has to trust his instruments over his sense of balance. [\[4\]](#)

Paradigms don't just define our present thinking. They also vacuum up the trail of crumbs that led to them, making our standards for new ideas impossibly high. The current paradigm seems so perfect to us, its offspring, that we imagine it must

have been accepted completely as soon as it was discovered — that whatever the church thought of the heliocentric model, astronomers must have been convinced as soon as Copernicus proposed it. Far, in fact, from it. Copernicus published the heliocentric model in 1532, but it wasn't till the mid seventeenth century that the balance of scientific opinion shifted in its favor. [5]

Few understand how feeble new ideas look when they first appear. So if you want to have new ideas yourself, one of the most valuable things you can do is to learn what they look like when they're born. Read about how new ideas happened, and try to get yourself into the heads of people at the time. How did things look to them, when the new idea was only half-finished, and even the person who had it was only half-convinced it was right?

But you don't have to stop at history. You can observe big new ideas being born all around you right now. Just look for a reasonable domain expert proposing something that sounds wrong.

If you're nice, as well as wise, you won't merely resist attacking such people, but encourage them. Having new ideas is a lonely business. Only those who've tried it know how lonely. These people need your help. And if you help them, you'll probably learn something in the process.

## Notes

[1] This domain expertise could be in another field. Indeed, such crossovers tend to be particularly promising.

[2] I'm not claiming this principle extends much beyond math, engineering, and the hard sciences. In politics, for example, crazy-sounding ideas generally are as bad as they sound. Though arguably this is not an exception, because the people who propose them are not in fact domain experts; politicians are domain experts in political tactics, like how to get elected and how to get legislation passed, but not in the world that policy acts upon. Perhaps no one could be.

[3] This sense of "paradigm" was defined by Thomas Kuhn in his *Structure of Scientific Revolutions*, but I also recommend his *Copernican Revolution*, where you can see him at work developing the idea.

[4] This is one reason people with a touch of Asperger's may have an advantage in discovering new ideas. They're always flying on instruments.



[5] Hall, Rupert. *From Galileo to Newton*. Collins, 1963. This book is particularly good at getting into contemporaries' heads.

**Thanks** to Trevor Blackwell, Patrick Collison, Suhail Doshi, Daniel Gackle, Jessica Livingston, and Robert Morris for reading drafts of this.

# Fierce Nerds

May 2021

Most people think of nerds as quiet, diffident people. In ordinary social situations they are — as quiet and diffident as the star quarterback would be if he found himself in the middle of a physics symposium. And for the same reason: they are fish out of water. But the apparent diffidence of nerds is an illusion due to the fact that when non-nerds observe them, it's usually in ordinary social situations. In fact some nerds are quite fierce.

The fierce nerds are a small but interesting group. They are as a rule extremely competitive — more competitive, I'd say, than highly competitive non-nerds. Competition is more personal for them. Partly perhaps because they're not emotionally mature enough to distance themselves from it, but also because there's less randomness in the kinds of competition they engage in, and they are thus more justified in taking the results personally.

Fierce nerds also tend to be somewhat overconfident, especially when young. It might seem like it would be a disadvantage to be mistaken about one's abilities, but empirically it isn't. Up to a point, confidence is a self-fulfilling prophecy.

Another quality you find in most fierce nerds is intelligence. Not all nerds are smart, but the fierce ones are always at least moderately so. If they weren't, they wouldn't have the confidence to be fierce. [\[1\]](#)

There's also a natural connection between nerdiness and [independent-mindedness](#). It's hard to be independent-minded without being somewhat socially awkward, because conventional beliefs are so often mistaken, or at least arbitrary. No one who was both independent-minded and ambitious would want to waste the effort it takes to fit in. And the independent-mindedness of the fierce nerds will obviously be of the [aggressive](#) rather than the passive type: they'll be annoyed by rules, rather than dreamily unaware of them.

I'm less sure why fierce nerds are impatient, but most seem to be. You notice it first in conversation, where they tend to interrupt you. This is merely annoying, but in the more promising fierce nerds it's connected to a deeper impatience about solving problems. Perhaps the competitiveness and impatience of fierce nerds are not separate qualities, but two manifestations of a single underlying drivenness.

When you combine all these qualities in sufficient quantities, the result is quite formidable. The most vivid example of fierce nerds in action may be James Watson's *The Double Helix*. The first sentence of the book is "I have never seen Francis Crick in a modest mood," and the portrait he goes on to paint of Crick is the quintessential fierce nerd: brilliant, socially awkward, competitive, independent-minded, overconfident. But so is the implicit portrait he paints of himself. Indeed, his lack of social awareness makes both portraits that much more realistic, because he baldly states all sorts of opinions and motivations that a smoother person would conceal. And moreover it's clear from the story that Crick and Watson's fierce nerdiness was integral to their success. Their independent-mindedness caused them to consider approaches that most others ignored, their overconfidence allowed them to work on problems they only half understood (they were literally described as "clowns" by one eminent insider), and their impatience and competitiveness got them to the answer ahead of two other groups that would otherwise have found it within the next year, if not the next several months. [2]

The idea that there could be fierce nerds is an unfamiliar one not just to many normal people but even to some young nerds. Especially early on, nerds spend so much of their time in ordinary social situations and so little doing real work that they get a lot more evidence of their awkwardness than their power. So there will be some who read this description of the fierce nerd and realize "Hmm, that's me." And it is to you, young fierce nerd, that I now turn.

I have some good news, and some bad news. The good news is that your fierceness will be a great help in solving difficult problems. And not just the kind of scientific and technical problems that nerds have traditionally solved. As the world progresses, the number of things you can win at by getting the right answer increases. Recently [getting rich](#) became one of them: 7 of the 8 richest people in America are now fierce nerds.

Indeed, being a fierce nerd is probably even more helpful in business than in nerds' original territory of scholarship. Fierceness seems optional there. Darwin for example doesn't seem to have been especially fierce. Whereas it's impossible to be the CEO of a company over a certain size without being fierce, so now that nerds can win at business, fierce nerds will increasingly monopolize the really big successes.

The bad news is that if it's not exercised, your fierceness will turn to bitterness, and you will become an intellectual playground bully: the grumpy sysadmin, the forum troll, the [hater](#), the shooter down of [new ideas](#).

How do you avoid this fate? Work on ambitious projects. If you succeed, it will bring you a kind of satisfaction that neutralizes bitterness. But you don't need to have succeeded to feel this; merely working on hard projects gives most fierce nerds some feeling of satisfaction. And those it doesn't, it at least keeps busy. [3]

Another solution may be to somehow turn off your fierceness, by devoting yourself to meditation or psychotherapy or something like that. Maybe that's the right

answer for some people. I have no idea. But it doesn't seem the optimal solution to me. If you're given a sharp knife, it seems to me better to use it than to blunt its edge to avoid cutting yourself.

If you do choose the ambitious route, you'll have a tailwind behind you. There has never been a better time to be a nerd. In the past century we've seen a continuous transfer of power from dealmakers to technicians — from the charismatic to the competent — and I don't see anything on the horizon that will end it. At least not till the nerds end it themselves by bringing about the singularity.

## Notes

[1] To be a nerd is to be socially awkward, and there are two distinct ways to do that: to be playing the same game as everyone else, but badly, and to be playing a different game. The smart nerds are the latter type.

[2] The same qualities that make fierce nerds so effective can also make them very annoying. Fierce nerds would do well to remember this, and (a) try to keep a lid on it, and (b) seek out organizations and types of work where getting the right answer matters more than preserving social harmony. In practice that means small groups working on hard problems. Which fortunately is the most fun kind of environment anyway.

[3] If success neutralizes bitterness, why are there some people who are at least moderately successful and yet still quite bitter? Because people's potential bitterness varies depending on how naturally bitter their personality is, and how ambitious they are: someone who's naturally very bitter will still have a lot left after success neutralizes some of it, and someone who's very ambitious will need proportionally more success to satisfy that ambition.

So the worst-case scenario is someone who's both naturally bitter and extremely ambitious, and yet only moderately successful.

**Thanks** to Trevor Blackwell, Steve Blank, Patrick Collison, Jessica Livingston, Amjad Masad, and Robert Morris for reading drafts of this.



# A Project of One's Own

June 2021

A few days ago, on the way home from school, my nine year old son told me he couldn't wait to get home to write more of the story he was working on. This made me as happy as anything I've heard him say — not just because he was excited about his story, but because he'd discovered this way of working. Working on a project of your own is as different from ordinary work as skating is from walking. It's more fun, but also much more productive.

What proportion of great work has been done by people who were skating in this sense? If not all of it, certainly a lot.

There is something special about working on a project of your own. I wouldn't say exactly that you're happier. A better word would be excited, or engaged. You're happy when things are going well, but often they aren't. When I'm writing an essay, most of the time I'm worried and puzzled: worried that the essay will turn out badly, and puzzled because I'm groping for some idea that I can't see clearly enough. Will I be able to pin it down with words? In the end I usually can, if I take long enough, but I'm never sure; the first few attempts often fail.

You have moments of happiness when things work out, but they don't last long, because then you're on to the next problem. So why do it at all? Because to the kind of people who like working this way, nothing else feels as right. You feel as if you're an animal in its natural habitat, doing what you were meant to do — not always happy, maybe, but awake and alive.

Many kids experience the excitement of working on projects of their own. The hard part is making this converge with the work you do as an adult. And our customs make it harder. We treat "playing" and "hobbies" as qualitatively different from "work". It's not clear to a kid building a treehouse that there's a direct (though long) route from that to architecture or engineering. And instead of pointing out the route, we conceal it, by implicitly treating the stuff kids do as different from real work. [\[1\]](#)

Instead of telling kids that their treehouses could be on the path to the work they do as adults, we tell them the path goes through school. And unfortunately schoolwork tends to be very different from working on projects of one's own. It's usually neither a project, nor one's own. So as school gets more serious, working

on projects of one's own is something that survives, if at all, as a thin thread off to the side.

It's a bit sad to think of all the high school kids turning their backs on building treehouses and sitting in class dutifully learning about Darwin or Newton to pass some exam, when the work that made Darwin and Newton famous was actually closer in spirit to building treehouses than studying for exams.

If I had to choose between my kids getting good grades and working on ambitious projects of their own, I'd pick the projects. And not because I'm an indulgent parent, but because I've been on the other end and I know which has more predictive value. When I was picking startups for Y Combinator, I didn't care about applicants' grades. But if they'd worked on projects of their own, I wanted to hear all about those. [2]

It may be inevitable that school is the way it is. I'm not saying we have to redesign it (though I'm not saying we don't), just that we should understand what it does to our attitudes to work — that it steers us toward the dutiful plodding kind of work, often using competition as bait, and away from skating.

There are occasionally times when schoolwork becomes a project of one's own. Whenever I had to write a paper, that would become a project of my own — except in English classes, ironically, because the things one has to write in English classes are so [bogus](#). And when I got to college and started taking CS classes, the programs I had to write became projects of my own. Whenever I was writing or programming, I was usually skating, and that has been true ever since.

So where exactly is the edge of projects of one's own? That's an interesting question, partly because the answer is so complicated, and partly because there's so much at stake. There turn out to be two senses in which work can be one's own: 1) that you're doing it voluntarily, rather than merely because someone told you to, and 2) that you're doing it by yourself.

The edge of the former is quite sharp. People who care a lot about their work are usually very sensitive to the difference between pulling, and being pushed, and work tends to fall into one category or the other. But the test isn't simply whether you're told to do something. You can choose to do something you're told to do. Indeed, you can own it far more thoroughly than the person who told you to do it.

For example, math homework is for most people something they're told to do. But for my father, who was a mathematician, it wasn't. Most of us think of the problems in a math book as a way to test or develop our knowledge of the material explained in each section. But to my father the problems were the part that mattered, and the text was merely a sort of annotation. Whenever he got a new math book it was to him like being given a puzzle: here was a new set of problems to solve, and he'd immediately set about solving all of them.

The other sense of a project being one's own — working on it by oneself — has a

much softer edge. It shades gradually into collaboration. And interestingly, it shades into collaboration in two different ways. One way to collaborate is to share a single project. For example, when two mathematicians collaborate on a proof that takes shape in the course of a conversation between them. The other way is when multiple people work on separate projects of their own that fit together like a jigsaw puzzle. For example, when one person writes the text of a book and another does the graphic design. [3]

These two paths into collaboration can of course be combined. But under the right conditions, the excitement of working on a project of one's own can be preserved for quite a while before disintegrating into the turbulent flow of work in a large organization. Indeed, the history of successful organizations is partly the history of techniques for preserving that excitement. [4]

The team that made the original Macintosh were a great example of this phenomenon. People like Burrell Smith and Andy Hertzfeld and Bill Atkinson and Susan Kare were not just following orders. They were not tennis balls hit by Steve Jobs, but rockets let loose by Steve Jobs. There was a lot of collaboration between them, but they all seem to have individually felt the excitement of working on a project of one's own.

In Andy Hertzfeld's book on the Macintosh, he describes how they'd come back into the office after dinner and work late into the night. People who've never experienced the thrill of working on a project they're excited about can't distinguish this kind of working long hours from the kind that happens in sweatshops and boiler rooms, but they're at opposite ends of the spectrum. That's why it's a mistake to insist dogmatically on "work/life balance." Indeed, the mere expression "work/life" embodies a mistake: it assumes work and life are distinct. For those to whom the word "work" automatically implies the dutiful plodding kind, they are. But for the skaters, the relationship between work and life would be better represented by a dash than a slash. I wouldn't want to work on anything that I didn't want to take over my life.

Of course, it's easier to achieve this level of motivation when you're making something like the Macintosh. It's easy for something new to feel like a project of your own. That's one of the reasons for the tendency programmers have to rewrite things that don't need rewriting, and to write their own versions of things that already exist. This sometimes alarms managers, and measured by total number of characters typed, it's rarely the optimal solution. But it's not always driven simply by arrogance or cluelessness. Writing code from scratch is also much more rewarding — so much more rewarding that a good programmer can end up net ahead, despite the shocking waste of characters. Indeed, it may be one of the advantages of capitalism that it encourages such rewriting. A company that needs software to do something can't use the software already written to do it at another company, and thus has to write their own, which often turns out better. [5]

The natural alignment between skating and solving new problems is one of the reasons the payoffs from startups are so high. Not only is the market price of



unsolved problems higher, you also get a discount on productivity when you work on them. In fact, you get a double increase in productivity: when you're doing a clean-sheet design, it's easier to recruit skaters, and they get to spend all their time skating.

Steve Jobs knew a thing or two about skaters from having watched Steve Wozniak. If you can find the right people, you only have to tell them what to do at the highest level. They'll handle the details. Indeed, they insist on it. For a project to feel like your own, you must have sufficient autonomy. You can't be working to order, or [slowed down](#) by bureaucracy.

One way to ensure autonomy is not to have a boss at all. There are two ways to do that: to be the boss yourself, and to work on projects outside of work. Though they're at opposite ends of the scale financially, startups and open source projects have a lot in common, including the fact that they're often run by skaters. And indeed, there's a wormhole from one end of the scale to the other: one of the best ways to discover [startup ideas](#) is to work on a project just for fun.

If your projects are the kind that make money, it's easy to work on them. It's harder when they're not. And the hardest part, usually, is morale. That's where adults have it harder than kids. Kids just plunge in and build their treehouse without worrying about whether they're wasting their time, or how it compares to other treehouses. And frankly we could learn a lot from kids here. The high standards most grownups have for "real" work do not always serve us well.

The most important phase in a project of one's own is at the beginning: when you go from thinking it might be cool to do x to actually doing x. And at that point high standards are not merely useless but positively harmful. There are a few people who start too many new projects, but far more, I suspect, who are deterred by fear of failure from starting projects that would have succeeded if they had.

But if we couldn't benefit as kids from the knowledge that our treehouses were on the path to grownup projects, we can at least benefit as grownups from knowing that our projects are on a path that stretches back to treehouses. Remember that careless confidence you had as a kid when starting something new? That would be a powerful thing to recapture.

If it's harder as adults to retain that kind of confidence, we at least tend to be more aware of what we're doing. Kids bounce, or are herded, from one kind of work to the next, barely realizing what's happening to them. Whereas we know more about different types of work and have more control over which we do. Ideally we can have the best of both worlds: to be deliberate in choosing to work on projects of our own, and carelessly confident in starting new ones.

## Notes

[1] "Hobby" is a curious word. Now it means work that isn't *real* work — work that one is not to be judged by — but originally it just meant an obsession in a fairly general sense (even a political opinion, for example) that one metaphorically rode as a child rides a hobby-horse. It's hard to say if its recent, narrower meaning is a change for the better or the worse. For sure there are lots of false positives — lots of projects that end up being important but are dismissed initially as mere hobbies. But on the other hand, the concept provides valuable cover for projects in the early, ugly duckling phase.

[2] Tiger parents, as parents so often do, are fighting the last war. Grades mattered more in the old days when the route to success was to acquire [credentials](#) while ascending some predefined ladder. But it's just as well that their tactics are focused on grades. How awful it would be if they invaded the territory of projects, and thereby gave their kids a distaste for this kind of work by forcing them to do it. Grades are already a grim, fake world, and aren't harmed much by parental interference, but working on one's own projects is a more delicate, private thing that could be damaged very easily.

[3] The complicated, gradual edge between working on one's own projects and collaborating with others is one reason there is so much disagreement about the idea of the "lone genius." In practice people collaborate (or not) in all kinds of different ways, but the idea of the lone genius is definitely not a myth. There's a core of truth to it that goes with a certain way of working.

[4] Collaboration is powerful too. The optimal organization would combine collaboration and ownership in such a way as to do the least damage to each. Interestingly, companies and university departments approach this ideal from opposite directions: companies insist on collaboration, and occasionally also manage both to recruit skaters and allow them to skate, and university departments insist on the ability to do independent research (which is by custom treated as skating, whether it is or not), and the people they hire collaborate as much as they choose.

[5] If a company could design its software in such a way that the best newly arrived programmers always got a clean sheet, it could have a kind of eternal youth. That might not be impossible. If you had a software backbone defining a game with sufficiently clear rules, individual programmers could write their own players.

**Thanks** to Trevor Blackwell, Paul Buchheit, Andy Hertzfeld, Jessica Livingston, and Peter Norvig for reading drafts of this.

# How to Work Hard

June 2021

It might not seem there's much to learn about how to work hard. Anyone who's been to school knows what it entails, even if they chose not to do it. There are 12 year olds who work amazingly hard. And yet when I ask if I know more about working hard now than when I was in school, the answer is definitely yes.

One thing I know is that if you want to do great things, you'll have to work very hard. I wasn't sure of that as a kid. Schoolwork varied in difficulty; one didn't always have to work super hard to do well. And some of the things famous adults did, they seemed to do almost effortlessly. Was there, perhaps, some way to evade hard work through sheer brilliance? Now I know the answer to that question. There isn't.

The reason some subjects seemed easy was that my school had low standards. And the reason famous adults seemed to do things effortlessly was years of practice; they made it look easy.

Of course, those famous adults usually had a lot of natural ability too. There are three ingredients in great work: natural ability, practice, and effort. You can do pretty well with just two, but to do the best work you need all three: you need great natural ability *and* to have practiced a lot *and* to be trying very hard. [1]

Bill Gates, for example, was among the smartest people in business in his era, but he was also among the hardest working. "I never took a day off in my twenties," he said. "Not one." It was similar with Lionel Messi. He had great natural ability, but when his youth coaches talk about him, what they remember is not his talent but his dedication and his desire to win. P. G. Wodehouse would probably get my vote for best English writer of the 20th century, if I had to choose. Certainly no one ever made it look easier. But no one ever worked harder. At 74, he wrote

with each new book of mine I have, as I say, the feeling that this time I have picked a lemon in the garden of literature. A good thing, really, I suppose. Keeps one up on one's toes and makes one rewrite every sentence ten times. Or in many cases twenty times.

Sounds a bit extreme, you think. And yet Bill Gates sounds even more extreme. Not one day off in ten years? These two had about as much natural ability as anyone could have, and yet they also worked about as hard as anyone could work.

You need both.

That seems so obvious, and yet in practice we find it slightly hard to grasp. There's a faint xor between talent and hard work. It comes partly from popular culture, where it seems to run very deep, and partly from the fact that the outliers are so rare. If great talent and great drive are both rare, then people with both are rare squared. Most people you meet who have a lot of one will have less of the other. But you'll need both if you want to be an outlier yourself. And since you can't really change how much natural talent you have, in practice doing great work, insofar as you can, reduces to working very hard.

It's straightforward to work hard if you have clearly defined, externally imposed goals, as you do in school. There is some technique to it: you have to learn not to lie to yourself, not to procrastinate (which is a form of lying to yourself), not to get distracted, and not to give up when things go wrong. But this level of discipline seems to be within the reach of quite young children, if they want it.

What I've learned since I was a kid is how to work toward goals that are neither clearly defined nor externally imposed. You'll probably have to learn both if you want to do really great things.

The most basic level of which is simply to feel you should be working without anyone telling you to. Now, when I'm not working hard, alarm bells go off. I can't be sure I'm getting anywhere when I'm working hard, but I can be sure I'm getting nowhere when I'm not, and it feels awful. [2]

There wasn't a single point when I learned this. Like most little kids, I enjoyed the feeling of achievement when I learned or did something new. As I grew older, this morphed into a feeling of disgust when I wasn't achieving anything. The one precisely dateable landmark I have is when I stopped watching TV, at age 13.

Several people I've talked to remember getting serious about work around this age. When I asked Patrick Collison when he started to find idleness distasteful, he said

I think around age 13 or 14. I have a clear memory from around then of sitting in the sitting room, staring outside, and wondering why I was wasting my summer holiday.

Perhaps something changes at adolescence. That would make sense.

Strangely enough, the biggest obstacle to getting serious about work was probably school, which made work (what they called work) seem boring and pointless. I had to learn what real work was before I could wholeheartedly desire to do it. That took a while, because even in college a lot of the work is pointless; there are entire departments that are pointless. But as I learned the shape of real work, I found that my desire to do it slotted into it as if they'd been made for each other.

I suspect most people have to learn what work is before they can love it. Hardy wrote eloquently about this in *A Mathematician's Apology*:

I do not remember having felt, as a boy, any *passion* for mathematics, and such notions as I may have had of the career of a mathematician were far from noble. I thought of mathematics in terms of examinations and scholarships: I wanted to beat other boys, and this seemed to be the way in which I could do so most decisively.

He didn't learn what math was really about till part way through college, when he read Jordan's *Cours d'analyse*.

I shall never forget the astonishment with which I read that remarkable work, the first inspiration for so many mathematicians of my generation, and learnt for the first time as I read it what mathematics really meant.

There are two separate kinds of fakeness you need to learn to discount in order to understand what real work is. One is the kind Hardy encountered in school. Subjects get distorted when they're adapted to be taught to kids — often so distorted that they're nothing like the work done by actual practitioners. [3] The other kind of fakeness is intrinsic to certain types of work. Some types of work are inherently bogus, or at best mere busywork.

There's a kind of solidity to real work. It's not all writing the *Principia*, but it all feels necessary. That's a vague criterion, but it's deliberately vague, because it has to cover a lot of different types. [4]

Once you know the shape of real work, you have to learn how many hours a day to spend on it. You can't solve this problem by simply working every waking hour, because in many kinds of work there's a point beyond which the quality of the result will start to decline.

That limit varies depending on the type of work and the person. I've done several different kinds of work, and the limits were different for each. My limit for the harder types of writing or programming is about five hours a day. Whereas when I was running a startup, I could work all the time. At least for the three years I did it; if I'd kept going much longer, I'd probably have needed to take occasional vacations. [5]

The only way to find the limit is by crossing it. Cultivate a sensitivity to the quality of the work you're doing, and then you'll notice if it decreases because you're working too hard. Honesty is critical here, in both directions: you have to notice when you're being lazy, but also when you're working too hard. And if you think there's something admirable about working too hard, get that idea out of your head. You're not merely getting worse results, but getting them because you're showing off — if not to other people, then to yourself. [6]

Finding the limit of working hard is a constant, ongoing process, not something you do just once. Both the difficulty of the work and your ability to do it can vary

hour to hour, so you need to be constantly judging both how hard you're trying and how well you're doing.

Trying hard doesn't mean constantly pushing yourself to work, though. There may be some people who do, but I think my experience is fairly typical, and I only have to push myself occasionally when I'm starting a project or when I encounter some sort of check. That's when I'm in danger of procrastinating. But once I get rolling, I tend to keep going.

What keeps me going depends on the type of work. When I was working on Viaweb, I was driven by fear of failure. I barely procrastinated at all then, because there was always something that needed doing, and if I could put more distance between me and the pursuing beast by doing it, why wait? [Z] Whereas what drives me now, writing essays, is the flaws in them. Between essays I fuss for a few days, like a dog circling while it decides exactly where to lie down. But once I get started on one, I don't have to push myself to work, because there's always some error or omission already pushing me.

I do make some amount of effort to focus on important topics. Many problems have a hard core at the center, surrounded by easier stuff at the edges. Working hard means aiming toward the center to the extent you can. Some days you may not be able to; some days you'll only be able to work on the easier, peripheral stuff. But you should always be aiming as close to the center as you can without stalling.

The bigger question of what to do with your life is one of these problems with a hard core. There are important problems at the center, which tend to be hard, and less important, easier ones at the edges. So as well as the small, daily adjustments involved in working on a specific problem, you'll occasionally have to make big, lifetime-scale adjustments about which type of work to do. And the rule is the same: working hard means aiming toward the center — toward the most ambitious problems.

By center, though, I mean the actual center, not merely the current consensus about the center. The consensus about which problems are most important is often mistaken, both in general and within specific fields. If you disagree with it, and you're right, that could represent a valuable opportunity to do something new.

The more ambitious types of work will usually be harder, but although you should not be in denial about this, neither should you treat difficulty as an infallible guide in deciding what to do. If you discover some ambitious type of work that's a bargain in the sense of being easier for you than other people, either because of the abilities you happen to have, or because of some new way you've found to approach it, or simply because you're more excited about it, by all means work on that. Some of the best work is done by people who find an easy way to do something hard.

As well as learning the shape of real work, you need to figure out which kind

you're suited for. And that doesn't just mean figuring out which kind your natural abilities match the best; it doesn't mean that if you're 7 feet tall, you have to play basketball. What you're suited for depends not just on your talents but perhaps even more on your interests. A [deep interest](#) in a topic makes people work harder than any amount of discipline can.

It can be harder to discover your interests than your talents. There are fewer types of talent than interest, and they start to be judged early in childhood, whereas interest in a topic is a subtle thing that may not mature till your twenties, or even later. The topic may not even exist earlier. Plus there are some powerful sources of error you need to learn to discount. Are you really interested in x, or do you want to work on it because you'll make a lot of money, or because other people will be impressed with you, or because your parents want you to? [\[8\]](#)

The difficulty of figuring out what to work on varies enormously from one person to another. That's one of the most important things I've learned about work since I was a kid. As a kid, you get the impression that everyone has a calling, and all they have to do is figure out what it is. That's how it works in movies, and in the streamlined biographies fed to kids. Sometimes it works that way in real life. Some people figure out what to do as children and just do it, like Mozart. But others, like Newton, turn restlessly from one kind of work to another. Maybe in retrospect we can identify one as their calling — we can wish Newton spent more time on math and physics and less on alchemy and theology — but this is an [illusion](#) induced by hindsight bias. There was no voice calling to him that he could have heard.

So while some people's lives converge fast, there will be others whose lives never converge. And for these people, figuring out what to work on is not so much a prelude to working hard as an ongoing part of it, like one of a set of simultaneous equations. For these people, the process I described earlier has a third component: along with measuring both how hard you're working and how well you're doing, you have to think about whether you should keep working in this field or switch to another. If you're working hard but not getting good enough results, you should switch. It sounds simple expressed that way, but in practice it's very difficult. You shouldn't give up on the first day just because you work hard and don't get anywhere. You need to give yourself time to get going. But how much time? And what should you do if work that was going well stops going well? How much time do you give yourself then? [\[9\]](#)

What even counts as good results? That can be really hard to decide. If you're exploring an area few others have worked in, you may not even know what good results look like. History is full of examples of people who misjudged the importance of what they were working on.

The best test of whether it's worthwhile to work on something is whether you find it interesting. That may sound like a dangerously subjective measure, but it's probably the most accurate one you're going to get. You're the one working on the stuff. Who's in a better position than you to judge whether it's important, and what's a better predictor of its importance than whether it's interesting?



For this test to work, though, you have to be honest with yourself. Indeed, that's the most striking thing about the whole question of working hard: how at each point it depends on being honest with yourself.

Working hard is not just a dial you turn up to 11. It's a complicated, dynamic system that has to be tuned just right at each point. You have to understand the shape of real work, see clearly what kind you're best suited for, aim as close to the true core of it as you can, accurately judge at each moment both what you're capable of and how you're doing, and put in as many hours each day as you can without harming the quality of the result. This network is too complicated to trick. But if you're consistently honest and clear-sighted, it will automatically assume an optimal shape, and you'll be productive in a way few people are.

## Notes

[1] In "The Bus Ticket Theory of Genius" I said the three ingredients in great work were natural ability, determination, and interest. That's the formula in the preceding stage; determination and interest yield practice and effort.

[2] I mean this at a resolution of days, not hours. You'll often get somewhere while not working in the sense that the solution to a problem comes to you while taking a [shower](#), or even in your sleep, but only because you were working hard on it the day before.

It's good to go on vacation occasionally, but when I go on vacation, I like to learn new things. I wouldn't like just sitting on a beach.

[3] The thing kids do in school that's most like the real version is sports. Admittedly because many sports originated as games played in schools. But in this one area, at least, kids are doing exactly what adults do.

In the average American high school, you have a choice of pretending to do something serious, or seriously doing something pretend. Arguably the latter is no worse.

[4] Knowing what you want to work on doesn't mean you'll be able to. Most people have to spend a lot of their time working on things they don't want to, especially

early on. But if you know what you want to do, you at least know what direction to nudge your life in.

[5] The lower time limits for intense work suggest a solution to the problem of having less time to work after you have kids: switch to harder problems. In effect I did that, though not deliberately.

[6] Some cultures have a tradition of performative hard work. I don't love this idea, because (a) it makes a parody of something important and (b) it causes people to wear themselves out doing things that don't matter. I don't know enough to say for sure whether it's net good or bad, but my guess is bad.

[7] One of the reasons people work so hard on startups is that startups can fail, and when they do, that failure tends to be both decisive and conspicuous.

[8] It's ok to work on something to make a lot of money. You need to solve the money problem somehow, and there's nothing wrong with doing that efficiently by trying to make a lot at once. I suppose it would even be ok to be interested in money for its own sake; whatever floats your boat. Just so long as you're conscious of your motivations. The thing to avoid is *unconsciously* letting the need for money warp your ideas about what kind of work you find most interesting.

[9] Many people face this question on a smaller scale with individual projects. But it's easier both to recognize and to accept a dead end in a single project than to abandon some type of work entirely. The more determined you are, the harder it gets. Like a Spanish Flu victim, you're fighting your own immune system: Instead of giving up, you tell yourself, I should just try harder. And who can say you're not right?

**Thanks** to Trevor Blackwell, John Carmack, John Collison, Patrick Collison, Robert Morris, Geoff Ralston, and Harj Taggar for reading drafts of this.

[Arabic Translation](#)

# Weird Languages

August 2021

When people say that in their experience all programming languages are basically equivalent, they're making a statement not about languages but about the kind of programming they've done.

99.5% of programming consists of gluing together calls to library functions. All popular languages are equally good at this. So one can easily spend one's whole career operating in the intersection of popular programming languages.

But the other .5% of programming is disproportionately interesting. If you want to learn what it consists of, the weirdness of weird languages is a good clue to follow.

Weird languages aren't weird by accident. Not the good ones, at least. The weirdness of the good ones usually implies the existence of some form of programming that's not just the usual gluing together of library calls.

A concrete example: Lisp macros. Lisp macros seem weird even to many Lisp programmers. They're not only not in the intersection of popular languages, but by their nature would be hard to implement properly in a language without turning it into a dialect of Lisp. And macros are definitely evidence of techniques that go beyond glue programming. For example, solving problems by first writing a language for problems of that type, and then writing your specific application in it. Nor is this all you can do with macros; it's just one region in a space of program-manipulating techniques that even now is far from fully explored.

So if you want to expand your concept of what programming can be, one way to do it is by learning weird languages. Pick a language that most programmers consider weird but whose median user is smart, and then focus on the differences between this language and the intersection of popular languages. What can you say in this language that would be impossibly inconvenient to say in others? In the process of learning how to say things you couldn't previously say, you'll probably be learning how to think things you couldn't previously think.

**Thanks** to Trevor Blackwell, Patrick Collison, Daniel Gackle, Amjad Masad, and Robert Morris for reading drafts of this.

[Japanese Translation](#)

# Beyond Smart

October 2021

If you asked people what was special about Einstein, most would say that he was really smart. Even the ones who tried to give you a more sophisticated-sounding answer would probably think this first. Till a few years ago I would have given the same answer myself. But that wasn't what was special about Einstein. What was special about him was that he had important new ideas. Being very smart was a necessary precondition for having those ideas, but the two are not identical.

It may seem a hair-splitting distinction to point out that intelligence and its consequences are not identical, but it isn't. There's a big gap between them. Anyone who's spent time around universities and research labs knows how big. There are a lot of genuinely smart people who don't achieve very much.

I grew up thinking that being smart was the thing most to be desired. Perhaps you did too. But I bet it's not what you really want. Imagine you had a choice between being really smart but discovering nothing new, and being less smart but discovering lots of new ideas. Surely you'd take the latter. I would. The choice makes me uncomfortable, but when you see the two options laid out explicitly like that, it's obvious which is better.

The reason the choice makes me uncomfortable is that being smart still feels like the thing that matters, even though I know intellectually that it isn't. I spent so many years thinking it was. The circumstances of childhood are a perfect storm for fostering this illusion. Intelligence is much easier to measure than the value of new ideas, and you're constantly being judged by it. Whereas even the kids who will ultimately discover new things aren't usually discovering them yet. For kids that way inclined, intelligence is the only game in town.

There are more subtle reasons too, which persist long into adulthood. Intelligence wins in conversation, and thus becomes the basis of the dominance hierarchy. [\[1\]](#) Plus having new ideas is such a new thing historically, and even now done by so few people, that society hasn't yet assimilated the fact that this is the actual destination, and intelligence merely a means to an end. [\[2\]](#)

Why do so many smart people fail to discover anything new? Viewed from that direction, the question seems a rather depressing one. But there's another way to look at it that's not just more optimistic, but more interesting as well. Clearly

intelligence is not the only ingredient in having new ideas. What are the other ingredients? Are they things we could cultivate?

Because the trouble with intelligence, they say, is that it's mostly inborn. The evidence for this seems fairly convincing, especially considering that most of us don't want it to be true, and the evidence thus has to face a stiff headwind. But I'm not going to get into that question here, because it's the other ingredients in new ideas that I care about, and it's clear that many of them can be cultivated.

That means the truth is excitingly different from the story I got as a kid. If intelligence is what matters, and also mostly inborn, the natural consequence is a sort of *Brave New World* fatalism. The best you can do is figure out what sort of work you have an "aptitude" for, so that whatever intelligence you were born with will at least be put to the best use, and then work as hard as you can at it. Whereas if intelligence isn't what matters, but only one of several ingredients in what does, and many of those aren't inborn, things get more interesting. You have a lot more control, but the problem of how to arrange your life becomes that much more complicated.

So what are the other ingredients in having new ideas? The fact that I can even ask this question proves the point I raised earlier — that society hasn't assimilated the fact that it's this and not intelligence that matters. Otherwise we'd all know the answers to such a fundamental question. [3]

I'm not going to try to provide a complete catalogue of the other ingredients here. This is the first time I've posed the question to myself this way, and I think it may take a while to answer. But I wrote recently about one of the most important: an obsessive [interest](#) in a particular topic. And this can definitely be cultivated.

Another quality you need in order to discover new ideas is [independent-mindedness](#). I wouldn't want to claim that this is distinct from intelligence — I'd be reluctant to call someone smart who wasn't independent-minded — but though largely inborn, this quality seems to be something that can be cultivated to some extent.

There are general techniques for having new ideas — for example, for working on your own [projects](#) and for overcoming the obstacles you face with [early](#) work — and these can all be learned. Some of them can be learned by societies. And there are also collections of techniques for generating specific types of new ideas, like [startup ideas](#) and [essay topics](#).

And of course there are a lot of fairly mundane ingredients in discovering new ideas, like [working hard](#), getting enough sleep, avoiding certain kinds of stress, having the right colleagues, and finding tricks for working on what you want even when it's not what you're supposed to be working on. Anything that prevents people from doing great work has an inverse that helps them to. And this class of ingredients is not as boring as it might seem at first. For example, having new ideas is generally associated with youth. But perhaps it's not youth per se that

yields new ideas, but specific things that come with youth, like good health and lack of responsibilities. Investigating this might lead to strategies that will help people of any age to have better ideas.

One of the most surprising ingredients in having new ideas is writing ability. There's a class of new ideas that are best discovered by writing essays and books. And that "by" is deliberate: you don't think of the ideas first, and then merely write them down. There is a kind of thinking that one does by writing, and if you're clumsy at writing, or don't enjoy doing it, that will get in your way if you try to do this kind of thinking. [4]

I predict the gap between intelligence and new ideas will turn out to be an interesting place. If we think of this gap merely as a measure of unrealized potential, it becomes a sort of wasteland that we try to hurry through with our eyes averted. But if we flip the question, and start inquiring into the other ingredients in new ideas that it implies must exist, we can mine this gap for discoveries about discovery.

## Notes

[1] What wins in conversation depends on who with. It ranges from mere aggressiveness at the bottom, through quick-wittedness in the middle, to something closer to actual intelligence at the top, though probably always with some component of quick-wittedness.

[2] Just as intelligence isn't the only ingredient in having new ideas, having new ideas isn't the only thing intelligence is useful for. It's also useful, for example, in diagnosing problems and figuring out how to fix them. Both overlap with having new ideas, but both have an end that doesn't.

Those ways of using intelligence are much more common than having new ideas. And in such cases intelligence is even harder to distinguish from its consequences.

[3] Some would attribute the difference between intelligence and having new ideas to "creativity," but this doesn't seem a very useful term. As well as being pretty vague, it's shifted half a frame sideways from what we care about: it's neither separable from intelligence, nor responsible for all the difference between intelligence and having new ideas.

[4] Curiously enough, this essay is an example. It started out as an essay about

writing ability. But when I came to the distinction between intelligence and having new ideas, that seemed so much more important that I turned the original essay inside out, making that the topic and my original topic one of the points in it. As in many other fields, that level of reworking is easier to contemplate once you've had a lot of practice.

**Thanks** to Trevor Blackwell, Patrick Collison, Jessica Livingston, Robert Morris, Michael Nielsen, and Lisa Randall for reading drafts of this.



# Is There Such a Thing as Good Taste?

November 2021

*(This essay is derived from a talk at the Cambridge Union.)*

When I was a kid, I'd have said there wasn't. My father told me so. Some people like some things, and other people like other things, and who's to say who's right?

It seemed so obvious that there was no such thing as good taste that it was only through indirect evidence that I realized my father was wrong. And that's what I'm going to give you here: a proof by *reductio ad absurdum*. If we start from the premise that there's no such thing as good taste, we end up with conclusions that are obviously false, and therefore the premise must be wrong.

We'd better start by saying what good taste is. There's a narrow sense in which it refers to aesthetic judgements and a broader one in which it refers to preferences of any kind. The strongest proof would be to show that taste exists in the narrowest sense, so I'm going to talk about taste in art. You have better taste than me if the art you like is better than the art I like.

If there's no such thing as good taste, then there's no such thing as [good art](#). Because if there is such a thing as good art, it's easy to tell which of two people has better taste. Show them a lot of works by artists they've never seen before and ask them to choose the best, and whoever chooses the better art has better taste.

So if you want to discard the concept of good taste, you also have to discard the concept of good art. And that means you have to discard the possibility of people being good at making it. Which means there's no way for artists to be good at their jobs. And not just visual artists, but anyone who is in any sense an artist. You can't have good actors, or novelists, or composers, or dancers either. You can have popular novelists, but not good ones.

We don't realize how far we'd have to go if we discarded the concept of good taste, because we don't even debate the most obvious cases. But it doesn't just mean we can't say which of two famous painters is better. It means we can't say that any painter is better than a randomly chosen eight year old.

That was how I realized my father was wrong. I started studying painting. And it

was just like other kinds of work I'd done: you could do it well, or badly, and if you tried hard, you could get better at it. And it was obvious that Leonardo and Bellini were much better at it than me. That gap between us was not imaginary. They were so good. And if they could be good, then art could be good, and there was such a thing as good taste after all.

Now that I've explained how to show there is such a thing as good taste, I should also explain why people think there isn't. There are two reasons. One is that there's always so much disagreement about taste. Most people's response to art is a tangle of unexamined impulses. Is the artist famous? Is the subject attractive? Is this the sort of art they're supposed to like? Is it hanging in a famous museum, or reproduced in a big, expensive book? In practice most people's response to art is dominated by such extraneous factors.

And the people who do claim to have good taste are so often mistaken. The paintings admired by the so-called experts in one generation are often so different from those admired a few generations later. It's easy to conclude there's nothing real there at all. It's only when you isolate this force, for example by trying to paint and comparing your work to Bellini's, that you can see that it does in fact exist.

The other reason people doubt that art can be good is that there doesn't seem to be any room in the art for this goodness. The argument goes like this. Imagine several people looking at a work of art and judging how good it is. If being good art really is a property of objects, it should be in the object somehow. But it doesn't seem to be; it seems to be something happening in the heads of each of the observers. And if they disagree, how do you choose between them?

The solution to this puzzle is to realize that the purpose of art is to work on its human audience, and humans have a lot in common. And to the extent the things an object acts upon respond in the same way, that's arguably what it means for the object to have the corresponding property. If everything a particle interacts with behaves as if the particle had a mass of  $m$ , then it has a mass of  $m$ . So the distinction between "objective" and "subjective" is not binary, but a matter of degree, depending on how much the subjects have in common. Particles interacting with one another are at one pole, but people interacting with art are not all the way at the other; their reactions aren't *random*.

Because people's responses to art aren't random, art can be designed to operate on people, and be good or bad depending on how effectively it does so. Much as a vaccine can be. If someone were talking about the ability of a vaccine to confer immunity, it would seem very frivolous to object that conferring immunity wasn't really a property of vaccines, because acquiring immunity is something that happens in the immune system of each individual person. Sure, people's immune systems vary, and a vaccine that worked on one might not work on another, but that doesn't make it meaningless to talk about the effectiveness of a vaccine.

The situation with art is messier, of course. You can't measure effectiveness by

simply taking a vote, as you do with vaccines. You have to imagine the responses of subjects with a deep knowledge of art, and enough clarity of mind to be able to ignore extraneous influences like the fame of the artist. And even then you'd still see some disagreement. People do vary, and judging art is hard, especially recent art. There is definitely not a total order either of works or of people's ability to judge them. But there is equally definitely a partial order of both. So while it's not possible to have perfect taste, it is possible to have good taste.

**Thanks** to the Cambridge Union for inviting me, and to Trevor Blackwell, Jessica Livingston, and Robert Morris for reading drafts of this.

# Putting Ideas into Words

February 2022

Writing about something, even something you know well, usually shows you that you didn't know it as well as you thought. Putting ideas into words is a severe test. The first words you choose are usually wrong; you have to rewrite sentences over and over to get them exactly right. And your ideas won't just be imprecise, but incomplete too. Half the ideas that end up in an essay will be ones you thought of while you were writing it. Indeed, that's why I write them.

Once you publish something, the convention is that whatever you wrote was what you thought before you wrote it. These were your ideas, and now you've expressed them. But you know this isn't true. You know that putting your ideas into words changed them. And not just the ideas you published. Presumably there were others that turned out to be too broken to fix, and those you discarded instead.

It's not just having to commit your ideas to specific words that makes writing so exacting. The real test is reading what you've written. You have to pretend to be a neutral reader who knows nothing of what's in your head, only what you wrote. When he reads what you wrote, does it seem correct? Does it seem complete? If you make an effort, you can read your writing as if you were a complete stranger, and when you do the news is usually bad. It takes me many cycles before I can get an essay past the stranger. But the stranger is rational, so you always can, if you ask him what he needs. If he's not satisfied because you failed to mention x or didn't qualify some sentence sufficiently, then you mention x or add more qualifications. Happy now? It may cost you some nice sentences, but you have to resign yourself to that. You just have to make them as good as you can and still satisfy the stranger.

This much, I assume, won't be that controversial. I think it will accord with the experience of anyone who has tried to write about anything nontrivial. There may exist people whose thoughts are so perfectly formed that they just flow straight into words. But I've never known anyone who could do this, and if I met someone who said they could, it would seem evidence of their limitations rather than their ability. Indeed, this is a trope in movies: the guy who claims to have a plan for doing some difficult thing, and who when questioned further, taps his head and says "It's all up here." Everyone watching the movie knows what that means. At best the plan is vague and incomplete. Very likely there's some undiscovered flaw that invalidates it completely. At best it's a plan for a plan.

In precisely defined domains it's possible to form complete ideas in your head. People can play chess in their heads, for example. And mathematicians can do some amount of math in their heads, though they don't seem to feel sure of a proof over a certain length till they write it down. But this only seems possible with ideas you can express in a formal language. [1] Arguably what such people are doing is putting ideas into words in their heads. I can to some extent write essays in my head. I'll sometimes think of a paragraph while walking or lying in bed that survives nearly unchanged in the final version. But really I'm writing when I do this. I'm doing the mental part of writing; my fingers just aren't moving as I do it. [2]

You can know a great deal about something without writing about it. Can you ever know so much that you wouldn't learn more from trying to explain what you know? I don't think so. I've written about at least two subjects I know well — Lisp hacking and startups — and in both cases I learned a lot from writing about them. In both cases there were things I didn't consciously realize till I had to explain them. And I don't think my experience was anomalous. A great deal of knowledge is unconscious, and experts have if anything a higher proportion of unconscious knowledge than beginners.

I'm not saying that writing is the best way to explore all ideas. If you have ideas about architecture, presumably the best way to explore them is to build actual buildings. What I'm saying is that however much you learn from exploring ideas in other ways, you'll still learn new things from writing about them.

Putting ideas into words doesn't have to mean writing, of course. You can also do it the old way, by talking. But in my experience, writing is the stricter test. You have to commit to a single, optimal sequence of words. Less can go unsaid when you don't have tone of voice to carry meaning. And you can focus in a way that would seem excessive in conversation. I'll often spend 2 weeks on an essay and reread drafts 50 times. If you did that in conversation it would seem evidence of some kind of mental disorder. If you're lazy, of course, writing and talking are equally useless. But if you want to push yourself to get things right, writing is the steeper hill. [3]

The reason I've spent so long establishing this rather obvious point is that it leads to another that many people will find shocking. If writing down your ideas always makes them more precise and more complete, then no one who hasn't written about a topic has fully formed ideas about it. And someone who never writes has no fully formed ideas about anything nontrivial.

It feels to them as if they do, especially if they're not in the habit of critically examining their own thinking. Ideas can feel complete. It's only when you try to put them into words that you discover they're not. So if you never subject your ideas to that test, you'll not only never have fully formed ideas, but also never realize it.

Putting ideas into words is certainly no guarantee that they'll be right. Far from it. But though it's not a sufficient condition, it is a necessary one.

## Notes

[1] Machinery and circuits are formal languages.

[2] I thought of this sentence as I was walking down the street in Palo Alto.

[3] There are two senses of talking to someone: a strict sense in which the conversation is verbal, and a more general sense in which it can take any form, including writing. In the limit case (e.g. Seneca's letters), conversation in the latter sense becomes essay writing.

It can be very useful to talk (in either sense) with other people as you're writing something. But a verbal conversation will never be more exacting than when you're talking about something you're writing.

**Thanks** to Trevor Blackwell, Patrick Collison, and Robert Morris for reading drafts of this.

[French Translation](#)

# Heresy

April 2022

One of the most surprising things I've witnessed in my lifetime is the rebirth of the concept of heresy.

In his excellent biography of Newton, Richard Westfall writes about the moment when he was elected a fellow of Trinity College:

Supported comfortably, Newton was free to devote himself wholly to whatever he chose. To remain on, he had only to avoid the three unforgivable sins: crime, heresy, and marriage. [\[1\]](#)

The first time I read that, in the 1990s, it sounded amusingly medieval. How strange, to have to avoid committing heresy. But when I reread it 20 years later it sounded like a description of contemporary employment.

There are an ever-increasing number of opinions you can be fired for. Those doing the firing don't use the word "heresy" to describe them, but structurally they're equivalent. Structurally there are two distinctive things about heresy: (1) that it takes priority over the question of truth or falsity, and (2) that it outweighs everything else the speaker has done.

For example, when someone calls a statement "x-ist," they're also implicitly saying that this is the end of the discussion. They do not, having said this, go on to consider whether the statement is true or not. Using such labels is the conversational equivalent of signalling an exception. That's one of the reasons they're used: to end a discussion.

If you find yourself talking to someone who uses these labels a lot, it might be worthwhile to ask them explicitly if they believe any babies are being thrown out with the bathwater. Can a statement be x-ist, for whatever value of x, and also true? If the answer is yes, then they're admitting to banning the truth. That's obvious enough that I'd guess most would answer no. But if they answer no, it's easy to show that they're mistaken, and that in practice such labels are applied to statements regardless of their truth or falsity.

The clearest evidence of this is that whether a statement is considered x-ist often depends on who said it. Truth doesn't work that way. The same statement can't be true when one person says it, but x-ist, and therefore false, when another person

does. [2]

The other distinctive thing about heresies, compared to ordinary opinions, is that the public expression of them outweighs everything else the speaker has done. In ordinary matters, like knowledge of history, or taste in music, you're judged by the average of your opinions. A heresy is qualitatively different. It's like dropping a chunk of uranium onto the scale.

Back in the day (and still, in some places) the punishment for heresy was death. You could have led a life of exemplary goodness, but if you publicly doubted, say, the divinity of Christ, you were going to burn. Nowadays, in civilized countries, heretics only get fired in the metaphorical sense, by losing their jobs. But the structure of the situation is the same: the heresy outweighs everything else. You could have spent the last ten years saving children's lives, but if you express certain opinions, you're automatically fired.

It's much the same as if you committed a crime. No matter how virtuously you've lived, if you commit a crime, you must still suffer the penalty of the law. Having lived a previously blameless life might mitigate the punishment, but it doesn't affect whether you're guilty or not.

A heresy is an opinion whose expression is treated like a crime — one that makes some people feel not merely that you're mistaken, but that you should be punished. Indeed, their desire to see you punished is often stronger than it would be if you'd committed an actual crime. There are many on the far left who believe strongly in the reintegration of felons (as I do myself), and yet seem to feel that anyone guilty of certain heresies should never work again.

There are always some heresies — some opinions you'd be punished for expressing. But there are a lot more now than there were a few decades ago, and even those who are happy about this would have to agree that it's so.

Why? Why has this antiquated-sounding religious concept come back in a secular form? And why now?

You need two ingredients for a wave of intolerance: intolerant people, and an ideology to guide them. The intolerant people are always there. They exist in every sufficiently large society. That's why waves of intolerance can arise so suddenly; all they need is something to set them off.

I've already written an [essay](#) describing the aggressively conventional-minded. The short version is that people can be classified in two dimensions according to (1) how independent- or conventional-minded they are, and (2) how aggressive they are about it. The aggressively conventional-minded are the enforcers of orthodoxy.

Normally they're only locally visible. They're the grumpy, censorious people in a group — the ones who are always first to complain when something violates the current rules of propriety. But occasionally, like a vector field whose elements



become aligned, a large number of aggressively conventional-minded people unite behind some ideology all at once. Then they become much more of a problem, because a mob dynamic takes over, where the enthusiasm of each participant is increased by the enthusiasm of the others.

The most notorious 20th century case may have been the Cultural Revolution. Though initiated by Mao to undermine his rivals, the Cultural Revolution was otherwise mostly a grass-roots phenomenon. Mao said in essence: There are heretics among us. Seek them out and punish them. And that's all the aggressively conventional-minded ever need to hear. They went at it with the delight of dogs chasing squirrels.

To unite the conventional-minded, an ideology must have many of the features of a religion. In particular it must have strict and arbitrary rules that adherents can demonstrate their [purity](#) by obeying, and its adherents must believe that anyone who obeys these rules is ipso facto morally superior to anyone who doesn't. [\[3\]](#)

In the late 1980s a new ideology of this type appeared in US universities. It had a very strong component of moral purity, and the aggressively conventional-minded seized upon it with their usual eagerness — all the more because the relaxation of social norms in the preceding decades meant there had been less and less to forbid. The resulting wave of intolerance has been eerily similar in form to the Cultural Revolution, though fortunately much smaller in magnitude. [\[4\]](#)

I've deliberately avoided mentioning any specific heresies here. Partly because one of the universal tactics of heretic hunters, now as in the past, is to accuse those who disapprove of the way in which they suppress ideas of being heretics themselves. Indeed, this tactic is so consistent that you could use it as a way of detecting witch hunts in any era.

And that's the second reason I've avoided mentioning any specific heresies. I want this essay to work in the future, not just now. And unfortunately it probably will. The aggressively conventional-minded will always be among us, looking for things to forbid. All they need is an ideology to tell them what. And it's unlikely the current one will be the last.

There are aggressively conventional-minded people on both the right and the left. The reason the current wave of intolerance comes from the left is simply because the new unifying ideology happened to come from the left. The next one might come from the right. Imagine what that would be like.

Fortunately in western countries the suppression of heresies is nothing like as bad as it used to be. Though the window of opinions you can express publicly has narrowed in the last decade, it's still much wider than it was a few hundred years ago. The problem is the derivative. Up till about 1985 the window had been growing ever wider. Anyone looking into the future in 1985 would have expected freedom of expression to continue to increase. Instead it has decreased. [\[5\]](#)

The situation is similar to what's happened with infectious diseases like measles. Anyone looking into the future in 2010 would have expected the number of measles cases in the US to continue to decrease. Instead, thanks to anti-vaxxers, it has increased. The absolute number is still not that high. The problem is the derivative. [6]

In both cases it's hard to know how much to worry. Is it really dangerous to society as a whole if a handful of extremists refuse to get their kids vaccinated, or shout down speakers at universities? The point to start worrying is presumably when their efforts start to spill over into everyone else's lives. And in both cases that does seem to be happening.

So it's probably worth spending some amount of effort on pushing back to keep open the window of free expression. My hope is that this essay will help form social antibodies not just against current efforts to suppress ideas, but against the concept of heresy in general. That's the real prize. How do you disable the concept of heresy? Since the Enlightenment, western societies have discovered many techniques for doing that, but there are surely more to be discovered.

Overall I'm optimistic. Though the trend in freedom of expression has been bad over the last decade, it's been good over the longer term. And there are signs that the current wave of intolerance is peaking. Independent-minded people I talk to seem more confident than they did a few years ago. On the other side, even some of the [leaders](#) are starting to wonder if things have gone too far. And popular culture among the young has already moved on. All we have to do is keep pushing back, and the wave collapses. And then we'll be net ahead, because as well as having defeated this wave, we'll also have developed new tactics for resisting the next one.

## Notes

[1] Or more accurately, biographies of Newton, since Westfall wrote two: a long version called *Never at Rest*, and a shorter one called *The Life of Isaac Newton*. Both are great. The short version moves faster, but the long one is full of interesting and often very funny details. This passage is the same in both.

[2] Another more subtle but equally damning bit of evidence is that claims of x-ism are never qualified. You never hear anyone say that a statement is "probably x-ist" or "almost certainly y-ist." If claims of x-ism were actually claims about truth, you'd expect to see "probably" in front of "x-ist" as often as you see it in front of

"fallacious."

[3] The rules must be strict, but they need not be demanding. So the most effective type of rules are those about superficial matters, like doctrinal minutiae, or the precise words adherents must use. Such rules can be made extremely complicated, and yet don't repel potential converts by requiring significant sacrifice.

The superficial demands of orthodoxy make it an inexpensive substitute for virtue. And that in turn is one of the reasons orthodoxy is so attractive to bad people. You could be a horrible person, and yet as long as you're orthodox, you're better than everyone who isn't.

[4] Arguably there were two. The first had died down somewhat by 2000, but was followed by a second in the 2010s, probably caused by social media.

[5] Fortunately most of those trying to suppress ideas today still respect Enlightenment principles enough to pay lip service to them. They know they're not supposed to ban ideas per se, so they have to recast the ideas as causing "harm," which sounds like something that can be banned. The more extreme try to claim speech itself is violence, or even that silence is. But strange as it may sound, such gymnastics are a good sign. We'll know we're really in trouble when they stop bothering to invent pretenses for banning ideas — when, like the medieval church, they say "Damn right we're banning ideas, and in fact here's a list of them."

[6] People only have the luxury of ignoring the medical consensus about vaccines because vaccines have worked so well. If we didn't have any vaccines at all, the mortality rate would be so high that most current anti-vaxxers would be begging for them. And the situation with freedom of expression is similar. It's only because they live in a world created by the Enlightenment that kids from the suburbs can play at banning ideas.

**Thanks** to Marc Andreessen, Chris Best, Trevor Blackwell, Nicholas Christakis, Daniel Gackle, Jonathan Haidt, Claire Lehmann, Jessica Livingston, Greg Lukianoff, Robert Morris, and Garry Tan for reading drafts of this.

# What I've Learned from Users

September 2022

I recently told applicants to Y Combinator that the best advice I could give for getting in, per word, was

Explain what you've learned from users.

That tests a lot of things: whether you're paying attention to users, how well you understand them, and even how much they need what you're making.

Afterward I asked myself the same question. What have I learned from YC's users, the startups we've funded?

The first thing that came to mind was that most startups have the same problems. No two have exactly the same problems, but it's surprising how much the problems remain the same, regardless of what they're making. Once you've advised 100 startups all doing different things, you rarely encounter problems you haven't seen before.

This fact is one of the things that makes YC work. But I didn't know it when we started YC. I only had a few data points: our own startup, and those started by friends. It was a surprise to me how often the same problems recur in different forms. Many later stage investors might never realize this, because later stage investors might not advise 100 startups in their whole career, but a YC partner will get this much experience in the first year or two.

That's one advantage of funding large numbers of early stage companies rather than smaller numbers of later-stage ones. You get a lot of data. Not just because you're looking at more companies, but also because more goes wrong.

But knowing (nearly) all the problems startups can encounter doesn't mean that advising them can be automated, or reduced to a formula. There's no substitute for individual office hours with a YC partner. Each startup is unique, which means they have to be advised by specific partners who know them well. [\[1\]](#)

We learned that the hard way, in the notorious "batch that broke YC" in the summer of 2012. Up till that point we treated the partners as a pool. When a startup requested office hours, they got the next available slot posted by any partner. That meant every partner had to know every startup. This worked fine up

to 60 startups, but when the batch grew to 80, everything broke. The founders probably didn't realize anything was wrong, but the partners were confused and unhappy because halfway through the batch they still didn't know all the companies yet. [2]

At first I was puzzled. How could things be fine at 60 startups and broken at 80? It was only a third more. Then I realized what had happened. We were using an  $O(n^2)$  algorithm. So of course it blew up.

The solution we adopted was the classic one in these situations. We sharded the batch into smaller groups of startups, each overseen by a dedicated group of partners. That fixed the problem, and has worked fine ever since. But the batch that broke YC was a powerful demonstration of how individualized the process of advising startups has to be.

Another related surprise is how bad founders can be at realizing what their problems are. Founders will sometimes come in to talk about some problem, and we'll discover another much bigger one in the course of the conversation. For example (and this case is all too common), founders will come in to talk about the difficulties they're having raising money, and after digging into their situation, it turns out the reason is that the company is doing badly, and investors can tell. Or founders will come in worried that they still haven't cracked the problem of user acquisition, and the reason turns out to be that their product isn't good enough. There have been times when I've asked "Would you use this yourself, if you hadn't built it?" and the founders, on thinking about it, said "No." Well, there's the reason you're having trouble getting users.

Often founders know what their problems are, but not their relative importance. [3] They'll come in to talk about three problems they're worrying about. One is of moderate importance, one doesn't matter at all, and one will kill the company if it isn't addressed immediately. It's like watching one of those horror movies where the heroine is deeply upset that her boyfriend cheated on her, and only mildly curious about the door that's mysteriously ajar. You want to say: never mind about your boyfriend, think about that door! Fortunately in office hours you can. So while startups still die with some regularity, it's rarely because they wandered into a room containing a murderer. The YC partners can warn them where the murderers are.

Not that founders listen. That was another big surprise: how often founders don't listen to us. A couple weeks ago I talked to a partner who had been working for YC for a couple batches and was starting to see the pattern. "They come back a year later," she said, "and say 'We wish we'd listened to you.'"

It took me a long time to figure out why founders don't listen. At first I thought it was mere stubbornness. That's part of the reason, but another and probably more important reason is that so much about startups is [counterintuitive](#). And when you tell someone something counterintuitive, what it sounds to them is wrong. So the

reason founders don't listen to us is that they don't *believe* us. At least not till experience teaches them otherwise. [4]

The reason startups are so counterintuitive is that they're so different from most people's other experiences. No one knows what it's like except those who've done it. Which is why YC partners should usually have been founders themselves. But strangely enough, the counterintuitiveness of startups turns out to be another of the things that make YC work. If it weren't counterintuitive, founders wouldn't need our advice about how to do it.

Focus is doubly important for early stage startups, because not only do they have a hundred different problems, they don't have anyone to work on them except the founders. If the founders focus on things that don't matter, there's no one focusing on the things that do. So the essence of what happens at YC is to figure out which problems matter most, then cook up ideas for solving them — ideally at a resolution of a week or less — and then try those ideas and measure how well they worked. The focus is on action, with measurable, near-term results.

This doesn't imply that founders should rush forward regardless of the consequences. If you correct course at a high enough frequency, you can be simultaneously decisive at a micro scale and tentative at a macro scale. The result is a somewhat winding path, but executed very rapidly, like the path a running back takes downfield. And in practice there's less backtracking than you might expect. Founders usually guess right about which direction to run in, especially if they have someone experienced like a YC partner to bounce their hypotheses off. And when they guess wrong, they notice fast, because they'll talk about the results at office hours the next week. [5]

A small improvement in navigational ability can make you a lot faster, because it has a double effect: the path is shorter, and you can travel faster along it when you're more certain it's the right one. That's where a lot of YC's value lies, in helping founders get an extra increment of focus that lets them move faster. And since moving fast is the essence of a startup, YC in effect makes startups more startup-like.

Speed defines startups. Focus enables speed. YC improves focus.

Why are founders uncertain about what to do? Partly because startups almost by definition are doing something new, which means no one knows how to do it yet, or in most cases even what "it" is. Partly because startups are so counterintuitive generally. And partly because many founders, especially young and ambitious ones, have been trained to win the wrong way. That took me years to figure out. The educational system in most countries trains you to win by [hacking the test](#) instead of actually doing whatever it's supposed to measure. But that stops working when you start a startup. So part of what YC does is to retrain founders to stop trying to hack the test. (It takes a surprisingly long time. A year in, you still see them reverting to their old habits.)

YC is not simply more experienced founders passing on their knowledge. It's more like specialization than apprenticeship. The knowledge of the YC partners and the founders have different shapes: It wouldn't be worthwhile for a founder to acquire the encyclopedic knowledge of startup problems that a YC partner has, just as it wouldn't be worthwhile for a YC partner to acquire the depth of domain knowledge that a founder has. That's why it can still be valuable for an experienced founder to do YC, just as it can still be valuable for an experienced athlete to have a coach.

The other big thing YC gives founders is colleagues, and this may be even more important than the advice of partners. If you look at history, great work clusters around certain places and institutions: Florence in the late 15th century, the University of Göttingen in the late 19th, *The New Yorker* under Ross, Bell Labs, Xerox PARC. However good you are, good colleagues make you better. Indeed, very ambitious people probably need colleagues more than anyone else, because they're so starved for them in everyday life.

Whether or not YC manages one day to be listed alongside those famous clusters, it won't be for lack of trying. We were very aware of this historical phenomenon and deliberately designed YC to be one. By this point it's not bragging to say that it's the biggest cluster of great startup founders. Even people trying to attack YC concede that.

Colleagues and startup founders are two of the most powerful forces in the world, so you'd expect it to have a big effect to combine them. Before YC, to the extent people thought about the question at all, most assumed they couldn't be combined — that loneliness was the price of independence. That was how it felt to us when we started our own startup in Boston in the 1990s. We had a handful of older people we could go to for advice (of varying quality), but no peers. There was no one we could commiserate with about the misbehavior of investors, or speculate with about the future of technology. I often tell founders to make something they themselves want, and YC is certainly that: it was designed to be exactly what we wanted when we were starting a startup.

One thing we wanted was to be able to get seed funding without having to make the rounds of random rich people. That has become a commodity now, at least in the US. But great colleagues can never become a commodity, because the fact that they cluster in some places means they're proportionally absent from the rest.

Something magical happens where they do cluster though. The energy in the room at a YC dinner is like nothing else I've experienced. We would have been happy just to have one or two other startups to talk to. When you have a whole roomful it's another thing entirely.

YC founders aren't just inspired by one another. They also help one another. That's the happiest thing I've learned about startup founders: how generous they can be in helping one another. We noticed this in the first batch and consciously designed YC to magnify it. The result is something far more intense than, say, a university. Between the partners, the alumni, and their batchmates, founders are surrounded

by people who want to help them, and can.

## Notes

[1] This is why I've never liked it when people refer to YC as a "bootcamp." It's intense like a bootcamp, but the opposite in structure. Instead of everyone doing the same thing, they're each talking to YC partners to figure out what their specific startup needs.

[2] When I say the summer 2012 batch was broken, I mean it felt to the partners that something was wrong. Things weren't yet so broken that the startups had a worse experience. In fact that batch did unusually well.

[3] This situation reminds me of the research showing that people are much better at answering questions than they are at judging how accurate their answers are. The two phenomena feel very similar.

[4] The [Airbnbs](#) were particularly good at listening — partly because they were flexible and disciplined, but also because they'd had such a rough time during the preceding year. They were ready to listen.

[5] The optimal unit of decisiveness depends on how long it takes to get results, and that depends on the type of problem you're solving. When you're negotiating with investors, it could be a couple days, whereas if you're building hardware it could be months.

**Thanks** to Trevor Blackwell, Jessica Livingston, Harj Taggar, and Garry Tan for reading drafts of this.



# Alien Truth

October 2022

If there were intelligent beings elsewhere in the universe, they'd share certain truths in common with us. The truths of mathematics would be the same, because they're true by definition. Ditto for the truths of physics; the mass of a carbon atom would be the same on their planet. But I think we'd share other truths with aliens besides the truths of math and physics, and that it would be worthwhile to think about what these might be.

For example, I think we'd share the principle that a controlled experiment testing some hypothesis entitles us to have proportionally increased belief in it. It seems fairly likely, too, that it would be true for aliens that one can get better at something by practicing. We'd probably share Occam's razor. There doesn't seem anything specifically human about any of these ideas.

We can only guess, of course. We can't say for sure what forms intelligent life might take. Nor is it my goal here to explore that question, interesting though it is. The point of the idea of alien truth is not that it gives us a way to speculate about what forms intelligent life might take, but that it gives us a threshold, or more precisely a target, for truth. If you're trying to find the most general truths short of those of math or physics, then presumably they'll be those we'd share in common with other forms of intelligent life.

Alien truth will work best as a heuristic if we err on the side of generosity. If an idea might plausibly be relevant to aliens, that's enough. Justice, for example. I wouldn't want to bet that all intelligent beings would understand the concept of justice, but I wouldn't want to bet against it either.

The idea of alien truth is related to Erdos's idea of God's book. He used to describe a particularly good proof as being in God's book, the implication being (a) that a sufficiently good proof was more discovered than invented, and (b) that its goodness would be universally recognized. If there's such a thing as alien truth, then there's more in God's book than math.

What should we call the search for alien truth? The obvious choice is "philosophy." Whatever else philosophy includes, it should probably include this. I'm fairly sure Aristotle would have thought so. One could even make the case that the search for alien truth is, if not an accurate description of philosophy, a good definition *for* it.

I.e. that it's what people who call themselves philosophers should be doing, whether or not they currently are. But I'm not wedded to that; doing it is what matters, not what we call it.

We may one day have something like alien life among us in the form of AIs. And that may in turn allow us to be precise about what truths an intelligent being would have to share with us. We might find, for example, that it's impossible to create something we'd consider intelligent that doesn't use Occam's razor. We might one day even be able to prove that. But though this sort of research would be very interesting, it's not necessary for our purposes, or even the same field; the goal of philosophy, if we're going to call it that, would be to see what ideas we come up with using alien truth as a target, not to say precisely where the threshold of it is. Those two questions might one day converge, but they'll converge from quite different directions, and till they do, it would be too constraining to restrict ourselves to thinking only about things we're certain would be alien truths. Especially since this will probably be one of those areas where the best guesses turn out to be surprisingly close to optimal. (Let's see if that one does.)

Whatever we call it, the attempt to discover alien truths would be a worthwhile undertaking. And curiously enough, that is itself probably an alien truth.

**Thanks** to Trevor Blackwell, Greg Brockman, Patrick Collison, Robert Morris, and Michael Nielsen for reading drafts of this.

# What You (Want to)\* Want

November 2022

Since I was about 9 I've been puzzled by the apparent contradiction between being made of matter that behaves in a predictable way, and the feeling that I could choose to do whatever I wanted. At the time I had a self-interested motive for exploring the question. At that age (like most succeeding ages) I was always in trouble with the authorities, and it seemed to me that there might possibly be some way to get out of trouble by arguing that I wasn't responsible for my actions. I gradually lost hope of that, but the puzzle remained: How do you reconcile being a machine made of matter with the feeling that you're free to choose what you do?

[1]

The best way to explain the answer may be to start with a slightly wrong version, and then fix it. The wrong version is: You can do what you want, but you can't want what you want. Yes, you can control what you do, but you'll do what you want, and you can't control that.

The reason this is mistaken is that people do sometimes change what they want. People who don't want to want something — drug addicts, for example — can sometimes make themselves stop wanting it. And people who want to want something — who want to like classical music, or broccoli — sometimes succeed.

So we modify our initial statement: You can do what you want, but you can't want to want what you want.

That's still not quite true. It's possible to change what you want to want. I can imagine someone saying "I decided to stop wanting to like classical music." But we're getting closer to the truth. It's rare for people to change what they want to want, and the more "want to"s we add, the rarer it gets.

We can get arbitrarily close to a true statement by adding more "want to"s in much the same way we can get arbitrarily close to 1 by adding more 9s to a string of 9s following a decimal point. In practice three or four "want to"s must surely be enough. It's hard even to envision what it would mean to change what you want to want to want to want, let alone actually do it.

So one way to express the correct answer is to use a regular expression. You can do what you want, but there's some statement of the form "you can't (want to)\*

want what you want" that's true. Ultimately you get back to a want that you don't control. [\[2\]](#)

## Notes

[1] I didn't know when I was 9 that matter might behave randomly, but I don't think it affects the problem much. Randomness destroys the ghost in the machine as effectively as determinism.

[2] If you don't like using an expression, you can make the same point using higher-order desires: There is some  $n$  such that you don't control your  $n$ th-order desires.

**Thanks** to Trevor Blackwell, Jessica Livingston, Robert Morris, and Michael Nielsen for reading drafts of this.

[Irish Translation](#)

# The Need to Read

November 2022

In the science fiction books I read as a kid, reading had often been replaced by some more efficient way of acquiring knowledge. Mysterious "tapes" would load it into one's brain like a program being loaded into a computer.

That sort of thing is unlikely to happen anytime soon. Not just because it would be hard to build a replacement for reading, but because even if one existed, it would be insufficient. Reading about  $x$  doesn't just teach you about  $x$ ; it also teaches you how to write. [1]

Would that matter? If we replaced reading, would anyone need to be good at writing?

The reason it would matter is that writing is not just a way to convey ideas, but also a way to have them.

A good writer doesn't just think, and then write down what he thought, as a sort of transcript. A good writer will almost always discover new things in the process of writing. And there is, as far as I know, no substitute for this kind of discovery. Talking about your ideas with other people is a good way to develop them. But even after doing this, you'll find you still discover new things when you sit down to write. There is a kind of thinking that can only be done by [writing](#).

There are of course kinds of thinking that can be done without writing. If you don't need to go too deeply into a problem, you can solve it without writing. If you're thinking about how two pieces of machinery should fit together, writing about it probably won't help much. And when a problem can be described formally, you can sometimes solve it in your head. But if you need to solve a complicated, ill-defined problem, it will almost always help to write about it. Which in turn means that someone who's not good at writing will almost always be at a disadvantage in solving such problems.

You can't think well without writing well, and you can't write well without reading well. And I mean that last "well" in both senses. You have to be good at reading, and read good things. [2]

People who just want information may find other ways to get it. But people who

want to have ideas can't afford to.

## Notes

[1] Audiobooks can give you examples of good writing, but having them read to you doesn't teach you as much about writing as reading them yourself.

[2] By "good at reading" I don't mean good at the mechanics of reading. You don't have to be good at extracting words from the page so much as extracting meaning from the words.

[Japanese Translation](#)

[Chinese Translation](#)

[Italian Translation](#)

[French Translation](#)

# How to Get New Ideas

January 2023

*([Someone](#) fed my essays into GPT to make something that could answer questions based on them, then asked it where good ideas come from. The answer was ok, but not what I would have said. This is what I would have said.)*

The way to get new ideas is to notice anomalies: what seems strange, or missing, or broken? You can see anomalies in everyday life (much of standup comedy is based on this), but the best place to look for them is at the frontiers of knowledge.

Knowledge grows fractally. From a distance its edges look smooth, but when you learn enough to get close to one, you'll notice it's full of gaps. These gaps will seem obvious; it will seem inexplicable that no one has tried x or wondered about y. In the best case, exploring such gaps yields whole new fractal buds.

# How to Do Great Work

July 2023

If you collected lists of techniques for doing great work in a lot of different fields, what would the intersection look like? I decided to find out by making it.

Partly my goal was to create a guide that could be used by someone working in any field. But I was also curious about the shape of the intersection. And one thing this exercise shows is that it does have a definite shape; it's not just a point labelled "work hard."

The following recipe assumes you're very ambitious.

The first step is to decide what to work on. The work you choose needs to have three qualities: it has to be something you have a natural aptitude for, that you have a deep interest in, and that offers scope to do great work.

In practice you don't have to worry much about the third criterion. Ambitious people are if anything already too conservative about it. So all you need to do is find something you have an aptitude for and great interest in. [\[1\]](#)

That sounds straightforward, but it's often quite difficult. When you're young you don't know what you're good at or what different kinds of work are like. Some kinds of work you end up doing may not even exist yet. So while some people know what they want to do at 14, most have to figure it out.

The way to figure out what to work on is by working. If you're not sure what to work on, guess. But pick something and get going. You'll probably guess wrong some of the time, but that's fine. It's good to know about multiple things; some of the biggest discoveries come from noticing connections between different fields.

Develop a habit of working on your own projects. Don't let "work" mean something other people tell you to do. If you do manage to do great work one day, it will probably be on a project of your own. It may be within some bigger project, but you'll be driving your part of it.



What should your projects be? Whatever seems to you excitingly ambitious. As you grow older and your taste in projects evolves, exciting and important will converge. At 7 it may seem excitingly ambitious to build huge things out of Lego, then at 14 to teach yourself calculus, till at 21 you're starting to explore unanswered questions in physics. But always preserve excitingness.

There's a kind of excited curiosity that's both the engine and the rudder of great work. It will not only drive you, but if you let it have its way, will also show you what to work on.

What are you excessively curious about — curious to a degree that would bore most other people? That's what you're looking for.

Once you've found something you're excessively interested in, the next step is to learn enough about it to get you to one of the frontiers of knowledge. Knowledge expands fractally, and from a distance its edges look smooth, but once you learn enough to get close to one, they turn out to be full of gaps.

The next step is to notice them. This takes some skill, because your brain wants to ignore such gaps in order to make a simpler model of the world. Many discoveries have come from asking questions about things that everyone else took for granted.

[2]

If the answers seem strange, so much the better. Great work often has a tincture of strangeness. You see this from painting to math. It would be affected to try to manufacture it, but if it appears, embrace it.

Boldly chase outlier ideas, even if other people aren't interested in them — in fact, especially if they aren't. If you're excited about some possibility that everyone else ignores, and you have enough expertise to say precisely what they're all overlooking, that's as good a bet as you'll find. [3]

Four steps: choose a field, learn enough to get to the frontier, notice gaps, explore promising ones. This is how practically everyone who's done great work has done it, from painters to physicists.

Steps two and four will require hard work. It may not be possible to prove that you have to work hard to do great things, but the empirical evidence is on the scale of the evidence for mortality. That's why it's essential to work on something you're deeply interested in. Interest will drive you to work harder than mere diligence ever could.

The three most powerful motives are curiosity, delight, and the desire to do something impressive. Sometimes they converge, and that combination is the most powerful of all.

The big prize is to discover a new fractal bud. You notice a crack in the surface of

knowledge, pry it open, and there's a whole world inside.

Let's talk a little more about the complicated business of figuring out what to work on. The main reason it's hard is that you can't tell what most kinds of work are like except by doing them. Which means the four steps overlap: you may have to work at something for years before you know how much you like it or how good you are at it. And in the meantime you're not doing, and thus not learning about, most other kinds of work. So in the worst case you choose late based on very incomplete information. [\[4\]](#)

The nature of ambition exacerbates this problem. Ambition comes in two forms, one that precedes interest in the subject and one that grows out of it. Most people who do great work have a mix, and the more you have of the former, the harder it will be to decide what to do.

The educational systems in most countries pretend it's easy. They expect you to commit to a field long before you could know what it's really like. And as a result an ambitious person on an optimal trajectory will often read to the system as an instance of breakage.

It would be better if they at least admitted it — if they admitted that the system not only can't do much to help you figure out what to work on, but is designed on the assumption that you'll somehow magically guess as a teenager. They don't tell you, but I will: when it comes to figuring out what to work on, you're on your own. Some people get lucky and do guess correctly, but the rest will find themselves scrambling diagonally across tracks laid down on the assumption that everyone does.

What should you do if you're young and ambitious but don't know what to work on? What you should *not* do is drift along passively, assuming the problem will solve itself. You need to take action. But there is no systematic procedure you can follow. When you read biographies of people who've done great work, it's remarkable how much luck is involved. They discover what to work on as a result of a chance meeting, or by reading a book they happen to pick up. So you need to make yourself a big target for luck, and the way to do that is to be curious. Try lots of things, meet lots of people, read lots of books, ask lots of questions. [\[5\]](#)

When in doubt, optimize for interestingness. Fields change as you learn more about them. What mathematicians do, for example, is very different from what you do in high school math classes. So you need to give different types of work a chance to show you what they're like. But a field should become *increasingly* interesting as you learn more about it. If it doesn't, it's probably not for you.

Don't worry if you find you're interested in different things than other people. The

stranger your tastes in interestingness, the better. Strange tastes are often strong ones, and a strong taste for work means you'll be productive. And you're more likely to find new things if you're looking where few have looked before.

One sign that you're suited for some kind of work is when you like even the parts that other people find tedious or frightening.

But fields aren't people; you don't owe them any loyalty. If in the course of working on one thing you discover another that's more exciting, don't be afraid to switch.

If you're making something for people, make sure it's something they actually want. The best way to do this is to make something you yourself want. Write the story you want to read; build the tool you want to use. Since your friends probably have similar interests, this will also get you your initial audience.

This *should* follow from the excitingness rule. Obviously the most exciting story to write will be the one you want to read. The reason I mention this case explicitly is that so many people get it wrong. Instead of making what they want, they try to make what some imaginary, more sophisticated audience wants. And once you go down that route, you're lost. [\[6\]](#)

There are a lot of forces that will lead you astray when you're trying to figure out what to work on. Pretentiousness, fashion, fear, money, politics, other people's wishes, eminent frauds. But if you stick to what you find genuinely interesting, you'll be proof against all of them. If you're interested, you're not astray.

Following your interests may sound like a rather passive strategy, but in practice it usually means following them past all sorts of obstacles. You usually have to risk rejection and failure. So it does take a good deal of boldness.

But while you need boldness, you don't usually need much planning. In most cases the recipe for doing great work is simply: work hard on excitingly ambitious projects, and something good will come of it. Instead of making a plan and then executing it, you just try to preserve certain invariants.

The trouble with planning is that it only works for achievements you can describe in advance. You can win a gold medal or get rich by deciding to as a child and then tenaciously pursuing that goal, but you can't discover natural selection that way.

I think for most people who want to do great work, the right strategy is not to plan too much. At each stage do whatever seems most interesting and gives you the best options for the future. I call this approach "staying upwind." This is how most people who've done great work seem to have done it.

Even when you've found something exciting to work on, working on it is not always straightforward. There will be times when some new idea makes you leap out of bed in the morning and get straight to work. But there will also be plenty of times when things aren't like that.

You don't just put out your sail and get blown forward by inspiration. There are headwinds and currents and hidden shoals. So there's a technique to working, just as there is to sailing.

For example, while you must work hard, it's possible to work too hard, and if you do that you'll find you get diminishing returns: fatigue will make you stupid, and eventually even damage your health. The point at which work yields diminishing returns depends on the type. Some of the hardest types you might only be able to do for four or five hours a day.

Ideally those hours will be contiguous. To the extent you can, try to arrange your life so you have big blocks of time to work in. You'll shy away from hard tasks if you know you might be interrupted.

It will probably be harder to start working than to keep working. You'll often have to trick yourself to get over that initial threshold. Don't worry about this; it's the nature of work, not a flaw in your character. Work has a sort of activation energy, both per day and per project. And since this threshold is fake in the sense that it's higher than the energy required to keep going, it's ok to tell yourself a lie of corresponding magnitude to get over it.

It's usually a mistake to lie to yourself if you want to do great work, but this is one of the rare cases where it isn't. When I'm reluctant to start work in the morning, I often trick myself by saying "I'll just read over what I've got so far." Five minutes later I've found something that seems mistaken or incomplete, and I'm off.

Similar techniques work for starting new projects. It's ok to lie to yourself about how much work a project will entail, for example. Lots of great things began with someone saying "How hard could it be?"

This is one case where the young have an advantage. They're more optimistic, and even though one of the sources of their optimism is ignorance, in this case ignorance can sometimes beat knowledge.

Try to finish what you start, though, even if it turns out to be more work than you expected. Finishing things is not just an exercise in tidiness or self-discipline. In many projects a lot of the best work happens in what was meant to be the final stage.

Another permissible lie is to exaggerate the importance of what you're working on, at least in your own mind. If that helps you discover something new, it may turn out not to have been a lie after all. [\[7\]](#)

Since there are two senses of starting work — per day and per project — there are also two forms of procrastination. Per-project procrastination is far the more dangerous. You put off starting that ambitious project from year to year because the time isn't quite right. When you're procrastinating in units of years, you can get a lot not done. [\[8\]](#)

One reason per-project procrastination is so dangerous is that it usually camouflages itself as work. You're not just sitting around doing nothing; you're working industriously on something else. So per-project procrastination doesn't set off the alarms that per-day procrastination does. You're too busy to notice it.

The way to beat it is to stop occasionally and ask yourself: Am I working on what I most want to work on? When you're young it's ok if the answer is sometimes no, but this gets increasingly dangerous as you get older. [\[9\]](#)

Great work usually entails spending what would seem to most people an unreasonable amount of time on a problem. You can't think of this time as a cost, or it will seem too high. You have to find the work sufficiently engaging as it's happening.

There may be some jobs where you have to work diligently for years at things you hate before you get to the good part, but this is not how great work happens. Great work happens by focusing consistently on something you're genuinely interested in. When you pause to take stock, you're surprised how far you've come.

The reason we're surprised is that we underestimate the cumulative effect of work. Writing a page a day doesn't sound like much, but if you do it every day you'll write a book a year. That's the key: consistency. People who do great things don't get a lot done every day. They get something done, rather than nothing.

If you do work that compounds, you'll get exponential growth. Most people who do this do it unconsciously, but it's worth stopping to think about. Learning, for example, is an instance of this phenomenon: the more you learn about something, the easier it is to learn more. Growing an audience is another: the more fans you

have, the more new fans they'll bring you.

The trouble with exponential growth is that the curve feels flat in the beginning. It isn't; it's still a wonderful exponential curve. But we can't grasp that intuitively, so we underrate exponential growth in its early stages.

Something that grows exponentially can become so valuable that it's worth making an extraordinary effort to get it started. But since we underrate exponential growth early on, this too is mostly done unconsciously: people push through the initial, unrewarding phase of learning something new because they know from experience that learning new things always takes an initial push, or they grow their audience one fan at a time because they have nothing better to do. If people consciously realized they could invest in exponential growth, many more would do it.

Work doesn't just happen when you're trying to. There's a kind of undirected thinking you do when walking or taking a shower or lying in bed that can be very powerful. By letting your mind wander a little, you'll often solve problems you were unable to solve by frontal attack.

You have to be working hard in the normal way to benefit from this phenomenon, though. You can't just walk around daydreaming. The daydreaming has to be interleaved with deliberate work that feeds it questions. [\[10\]](#)

Everyone knows to avoid distractions at work, but it's also important to avoid them in the other half of the cycle. When you let your mind wander, it wanders to whatever you care about most at that moment. So avoid the kind of distraction that pushes your work out of the top spot, or you'll waste this valuable type of thinking on the distraction instead. (Exception: Don't avoid love.)

Consciously cultivate your taste in the work done in your field. Until you know which is the best and what makes it so, you don't know what you're aiming for.

And that *is* what you're aiming for, because if you don't try to be the best, you won't even be good. This observation has been made by so many people in so many different fields that it might be worth thinking about why it's true. It could be because ambition is a phenomenon where almost all the error is in one direction — where almost all the shells that miss the target miss by falling short. Or it could be because ambition to be the best is a qualitatively different thing from ambition to be good. Or maybe being good is simply too vague a standard. Probably all three are true. [\[11\]](#)

Fortunately there's a kind of economy of scale here. Though it might seem like you'd be taking on a heavy burden by trying to be the best, in practice you often end up net ahead. It's exciting, and also strangely liberating. It simplifies things. In some ways it's easier to try to be the best than to try merely to be good.

One way to aim high is to try to make something that people will care about in a hundred years. Not because their opinions matter more than your contemporaries', but because something that still seems good in a hundred years is more likely to be genuinely good.

Don't try to work in a distinctive style. Just try to do the best job you can; you won't be able to help doing it in a distinctive way.

Style is doing things in a distinctive way without trying to. Trying to is affectation.

Affectation is in effect to pretend that someone other than you is doing the work. You adopt an impressive but fake persona, and while you're pleased with the impressiveness, the fakeness is what shows in the work. [\[12\]](#)

The temptation to be someone else is greatest for the young. They often feel like nobodies. But you never need to worry about that problem, because it's self-solving if you work on sufficiently ambitious projects. If you succeed at an ambitious project, you're not a nobody; you're the person who did it. So just do the work and your identity will take care of itself.

"Avoid affectation" is a useful rule so far as it goes, but how would you express this idea positively? How would you say what to be, instead of what not to be? The best answer is earnest. If you're earnest you avoid not just affectation but a whole set of similar vices.

The core of being earnest is being intellectually honest. We're taught as children to be honest as an unselfish virtue — as a kind of sacrifice. But in fact it's a source of power too. To see new ideas, you need an exceptionally sharp eye for the truth. You're trying to see more truth than others have seen so far. And how can you have a sharp eye for the truth if you're intellectually dishonest?

One way to avoid intellectual dishonesty is to maintain a slight positive pressure in the opposite direction. Be aggressively willing to admit that you're mistaken. Once you've admitted you were mistaken about something, you're free. Till then you

have to carry it. [13]

Another more subtle component of earnestness is informality. Informality is much more important than its grammatically negative name implies. It's not merely the absence of something. It means focusing on what matters instead of what doesn't.

What formality and affectation have in common is that as well as doing the work, you're trying to seem a certain way as you're doing it. But any energy that goes into how you seem comes out of being good. That's one reason nerds have an advantage in doing great work: they expend little effort on seeming anything. In fact that's basically the definition of a nerd.

Nerds have a kind of innocent boldness that's exactly what you need in doing great work. It's not learned; it's preserved from childhood. So hold onto it. Be the one who puts things out there rather than the one who sits back and offers sophisticated-sounding criticisms of them. "It's easy to criticize" is true in the most literal sense, and the route to great work is never easy.

There may be some jobs where it's an advantage to be cynical and pessimistic, but if you want to do great work it's an advantage to be optimistic, even though that means you'll risk looking like a fool sometimes. There's an old tradition of doing the opposite. The Old Testament says it's better to keep quiet lest you look like a fool. But that's advice for *seeming* smart. If you actually want to discover new things, it's better to take the risk of telling people your ideas.

Some people are naturally earnest, and with others it takes a conscious effort. Either kind of earnestness will suffice. But I doubt it would be possible to do great work without being earnest. It's so hard to do even if you are. You don't have enough margin for error to accommodate the distortions introduced by being affected, intellectually dishonest, orthodox, fashionable, or cool. [14]

Great work is consistent not only with who did it, but with itself. It's usually all of a piece. So if you face a decision in the middle of working on something, ask which choice is more consistent.

You may have to throw things away and redo them. You won't necessarily have to, but you have to be willing to. And that can take some effort; when there's something you need to redo, status quo bias and laziness will combine to keep you in denial about it. To beat this ask: If I'd already made the change, would I want to revert to what I have now?

Have the confidence to cut. Don't keep something that doesn't fit just because you're proud of it, or because it cost you a lot of effort.



Indeed, in some kinds of work it's good to strip whatever you're doing to its essence. The result will be more concentrated; you'll understand it better; and you won't be able to lie to yourself about whether there's anything real there.

Mathematical elegance may sound like a mere metaphor, drawn from the arts. That's what I thought when I first heard the term "elegant" applied to a proof. But now I suspect it's conceptually prior — that the main ingredient in artistic elegance is mathematical elegance. At any rate it's a useful standard well beyond math.

Elegance can be a long-term bet, though. Laborious solutions will often have more prestige in the short term. They cost a lot of effort and they're hard to understand, both of which impress people, at least temporarily.

Whereas some of the very best work will seem like it took comparatively little effort, because it was in a sense already there. It didn't have to be built, just seen. It's a very good sign when it's hard to say whether you're creating something or discovering it.

When you're doing work that could be seen as either creation or discovery, err on the side of discovery. Try thinking of yourself as a mere conduit through which the ideas take their natural shape.

(Strangely enough, one exception is the problem of choosing a problem to work on. This is usually seen as search, but in the best case it's more like creating something. In the best case you create the field in the process of exploring it.)

Similarly, if you're trying to build a powerful tool, make it gratuitously unrestrictive. A powerful tool almost by definition will be used in ways you didn't expect, so err on the side of eliminating restrictions, even if you don't know what the benefit will be.

Great work will often be tool-like in the sense of being something others build on. So it's a good sign if you're creating ideas that others could use, or exposing questions that others could answer. The best ideas have implications in many different areas.

If you express your ideas in the most general form, they'll be truer than you intended.

True by itself is not enough, of course. Great ideas have to be true and new. And it takes a certain amount of ability to see new ideas even once you've learned enough to get to one of the frontiers of knowledge.

In English we give this ability names like originality, creativity, and imagination.

And it seems reasonable to give it a separate name, because it does seem to some extent a separate skill. It's possible to have a great deal of ability in other respects — to have a great deal of what's often called *technical* ability — and yet not have much of this.

I've never liked the term "creative process." It seems misleading. Originality isn't a process, but a habit of mind. Original thinkers throw off new ideas about whatever they focus on, like an angle grinder throwing off sparks. They can't help it.

If the thing they're focused on is something they don't understand very well, these new ideas might not be good. One of the most original thinkers I know decided to focus on dating after he got divorced. He knew roughly as much about dating as the average 15 year old, and the results were spectacularly colorful. But to see originality separated from expertise like that made its nature all the more clear.

I don't know if it's possible to cultivate originality, but there are definitely ways to make the most of however much you have. For example, you're much more likely to have original ideas when you're working on something. Original ideas don't come from trying to have original ideas. They come from trying to build or understand something slightly too difficult. [15]

Talking or writing about the things you're interested in is a good way to generate new ideas. When you try to put ideas into words, a missing idea creates a sort of vacuum that draws it out of you. Indeed, there's a kind of thinking that can only be done by writing.

Changing your context can help. If you visit a new place, you'll often find you have new ideas there. The journey itself often dislodges them. But you may not have to go far to get this benefit. Sometimes it's enough just to go for a walk. [16]

It also helps to travel in topic space. You'll have more new ideas if you explore lots of different topics, partly because it gives the angle grinder more surface area to work on, and partly because analogies are an especially fruitful source of new ideas.

Don't divide your attention *evenly* between many topics though, or you'll spread yourself too thin. You want to distribute it according to something more like a power law. [17] Be professionally curious about a few topics and idly curious about many more.

Curiosity and originality are closely related. Curiosity feeds originality by giving it new things to work on. But the relationship is closer than that. Curiosity is itself a kind of originality; it's roughly to questions what originality is to answers. And since questions at their best are a big component of answers, curiosity at its best is a creative force.

Having new ideas is a strange game, because it usually consists of seeing things that were right under your nose. Once you've seen a new idea, it tends to seem obvious. Why did no one think of this before?

When an idea seems simultaneously novel and obvious, it's probably a good one.

Seeing something obvious sounds easy. And yet empirically having new ideas is hard. What's the source of this apparent contradiction? It's that seeing the new idea usually requires you to change the way you look at the world. We see the world through models that both help and constrain us. When you fix a broken model, new ideas become obvious. But noticing and fixing a broken model is hard. That's how new ideas can be both obvious and yet hard to discover: they're easy to see after you do something hard.

One way to discover broken models is to be stricter than other people. Broken models of the world leave a trail of clues where they bash against reality. Most people don't want to see these clues. It would be an understatement to say that they're attached to their current model; it's what they think in; so they'll tend to ignore the trail of clues left by its breakage, however conspicuous it may seem in retrospect.

To find new ideas you have to seize on signs of breakage instead of looking away. That's what Einstein did. He was able to see the wild implications of Maxwell's equations not so much because he was looking for new ideas as because he was stricter.

The other thing you need is a willingness to break rules. Paradoxical as it sounds, if you want to fix your model of the world, it helps to be the sort of person who's comfortable breaking rules. From the point of view of the old model, which everyone including you initially shares, the new model usually breaks at least implicit rules.

Few understand the degree of rule-breaking required, because new ideas seem much more conservative once they succeed. They seem perfectly reasonable once you're using the new model of the world they brought with them. But they didn't at the time; it took the greater part of a century for the heliocentric model to be generally accepted, even among astronomers, because it felt so wrong.

Indeed, if you think about it, a good new idea has to seem bad to most people, or someone would have already explored it. So what you're looking for is ideas that seem crazy, but the right kind of crazy. How do you recognize these? You can't with certainty. Often ideas that seem bad are bad. But ideas that are the right kind of crazy tend to be exciting; they're rich in implications; whereas ideas that are merely bad tend to be depressing.

There are two ways to be comfortable breaking rules: to enjoy breaking them, and

to be indifferent to them. I call these two cases being aggressively and passively independent-minded.

The aggressively independent-minded are the naughty ones. Rules don't merely fail to stop them; breaking rules gives them additional energy. For this sort of person, delight at the sheer audacity of a project sometimes supplies enough activation energy to get it started.

The other way to break rules is not to care about them, or perhaps even to know they exist. This is why novices and outsiders often make new discoveries; their ignorance of a field's assumptions acts as a source of temporary passive independent-mindedness. Aspies also seem to have a kind of immunity to conventional beliefs. Several I know say that this helps them to have new ideas.

Strictness plus rule-breaking sounds like a strange combination. In popular culture they're opposed. But popular culture has a broken model in this respect. It implicitly assumes that issues are trivial ones, and in trivial matters strictness and rule-breaking *are* opposed. But in questions that really matter, only rule-breakers can be truly strict.

An overlooked idea often doesn't lose till the semifinals. You do see it, subconsciously, but then another part of your subconscious shoots it down because it would be too weird, too risky, too much work, too controversial. This suggests an exciting possibility: if you could turn off such filters, you could see more new ideas.

One way to do that is to ask what would be good ideas for *someone else* to explore. Then your subconscious won't shoot them down to protect you.

You could also discover overlooked ideas by working in the other direction: by starting from what's obscuring them. Every cherished but mistaken principle is surrounded by a dead zone of valuable ideas that are unexplored because they contradict it.

Religions are collections of cherished but mistaken principles. So anything that can be described either literally or metaphorically as a religion will have valuable unexplored ideas in its shadow. Copernicus and Darwin both made discoveries of this type. [\[18\]](#)

What are people in your field religious about, in the sense of being too attached to some principle that might not be as self-evident as they think? What becomes possible if you discard it?

People show much more originality in solving problems than in deciding which problems to solve. Even the smartest can be surprisingly conservative when deciding what to work on. People who'd never dream of being fashionable in any other way get sucked into working on fashionable problems.

One reason people are more conservative when choosing problems than solutions is that problems are bigger bets. A problem could occupy you for years, while exploring a solution might only take days. But even so I think most people are too conservative. They're not merely responding to risk, but to fashion as well. Unfashionable problems are undervalued.

One of the most interesting kinds of unfashionable problem is the problem that people think has been fully explored, but hasn't. Great work often takes something that already exists and shows its latent potential. Durer and Watt both did this. So if you're interested in a field that others think is tapped out, don't let their skepticism deter you. People are often wrong about this.

Working on an unfashionable problem can be very pleasing. There's no hype or hurry. Opportunists and critics are both occupied elsewhere. The existing work often has an old-school solidity. And there's a satisfying sense of economy in cultivating ideas that would otherwise be wasted.

But the most common type of overlooked problem is not explicitly unfashionable in the sense of being out of fashion. It just doesn't seem to matter as much as it actually does. How do you find these? By being self-indulgent — by letting your curiosity have its way, and tuning out, at least temporarily, the little voice in your head that says you should only be working on "important" problems.

You do need to work on important problems, but almost everyone is too conservative about what counts as one. And if there's an important but overlooked problem in your neighborhood, it's probably already on your subconscious radar screen. So try asking yourself: if you were going to take a break from "serious" work to work on something just because it would be really interesting, what would you do? The answer is probably more important than it seems.

Originality in choosing problems seems to matter even more than originality in solving them. That's what distinguishes the people who discover whole new fields. So what might seem to be merely the initial step — deciding what to work on — is in a sense the key to the whole game.

Few grasp this. One of the biggest misconceptions about new ideas is about the ratio of question to answer in their composition. People think big ideas are

answers, but often the real insight was in the question.

Part of the reason we underrate questions is the way they're used in schools. In schools they tend to exist only briefly before being answered, like unstable particles. But a really good question can be much more than that. A really good question is a partial discovery. How do new species arise? Is the force that makes objects fall to earth the same as the one that keeps planets in their orbits? By even asking such questions you were already in excitingly novel territory.

Unanswered questions can be uncomfortable things to carry around with you. But the more you're carrying, the greater the chance of noticing a solution — or perhaps even more excitingly, noticing that two unanswered questions are the same.

Sometimes you carry a question for a long time. Great work often comes from returning to a question you first noticed years before — in your childhood, even — and couldn't stop thinking about. People talk a lot about the importance of keeping your youthful dreams alive, but it's just as important to keep your youthful questions alive. [\[19\]](#)

This is one of the places where actual expertise differs most from the popular picture of it. In the popular picture, experts are certain. But actually the more puzzled you are, the better, so long as (a) the things you're puzzled about matter, and (b) no one else understands them either.

Think about what's happening at the moment just before a new idea is discovered. Often someone with sufficient expertise is puzzled about something. Which means that originality consists partly of puzzlement — of confusion! You have to be comfortable enough with the world being full of puzzles that you're willing to see them, but not so comfortable that you don't want to solve them. [\[20\]](#)

It's a great thing to be rich in unanswered questions. And this is one of those situations where the rich get richer, because the best way to acquire new questions is to try answering existing ones. Questions don't just lead to answers, but also to more questions.

The best questions grow in the answering. You notice a thread protruding from the current paradigm and try pulling on it, and it just gets longer and longer. So don't require a question to be obviously big before you try answering it. You can rarely predict that. It's hard enough even to notice the thread, let alone to predict how much will unravel if you pull on it.

It's better to be promiscuously curious — to pull a little bit on a lot of threads, and see what happens. Big things start small. The initial versions of big things were

often just experiments, or side projects, or talks, which then grew into something bigger. So start lots of small things.

Being prolific is underrated. The more different things you try, the greater the chance of discovering something new. Understand, though, that trying lots of things will mean trying lots of things that don't work. You can't have a lot of good ideas without also having a lot of bad ones. [21]

Though it sounds more responsible to begin by studying everything that's been done before, you'll learn faster and have more fun by trying stuff. And you'll understand previous work better when you do look at it. So err on the side of starting. Which is easier when starting means starting small; those two ideas fit together like two puzzle pieces.

How do you get from starting small to doing something great? By making successive versions. Great things are almost always made in successive versions. You start with something small and evolve it, and the final version is both cleverer and more ambitious than anything you could have planned.

It's particularly useful to make successive versions when you're making something for people — to get an initial version in front of them quickly, and then evolve it based on their response.

Begin by trying the simplest thing that could possibly work. Surprisingly often, it does. If it doesn't, this will at least get you started.

Don't try to cram too much new stuff into any one version. There are names for doing this with the first version (taking too long to ship) and the second (the second system effect), but these are both merely instances of a more general principle.

An early version of a new project will sometimes be dismissed as a toy. It's a good sign when people do this. That means it has everything a new idea needs except scale, and that tends to follow. [22]

The alternative to starting with something small and evolving it is to plan in advance what you're going to do. And planning does usually seem the more responsible choice. It sounds more organized to say "we're going to do x and then y and then z" than "we're going to try x and see what happens." And it is more *organized*; it just doesn't work as well.

Planning per se isn't good. It's sometimes necessary, but it's a necessary evil — a response to unforgiving conditions. It's something you have to do because you're working with inflexible media, or because you need to coordinate the efforts of a lot of people. If you keep projects small and use flexible media, you don't have to plan as much, and your designs can evolve instead.

Take as much risk as you can afford. In an efficient market, risk is proportionate to reward, so don't look for certainty, but for a bet with high expected value. If you're not failing occasionally, you're probably being too conservative.

Though conservatism is usually associated with the old, it's the young who tend to make this mistake. Inexperience makes them fear risk, but it's when you're young that you can afford the most.

Even a project that fails can be valuable. In the process of working on it, you'll have crossed territory few others have seen, and encountered questions few others have asked. And there's probably no better source of questions than the ones you encounter in trying to do something slightly too hard.

Use the advantages of youth when you have them, and the advantages of age once you have those. The advantages of youth are energy, time, optimism, and freedom. The advantages of age are knowledge, efficiency, money, and power. With effort you can acquire some of the latter when young and keep some of the former when old.

The old also have the advantage of knowing which advantages they have. The young often have them without realizing it. The biggest is probably time. The young have no idea how rich they are in time. The best way to turn this time to advantage is to use it in slightly frivolous ways: to learn about something you don't need to know about, just out of curiosity, or to try building something just because it would be cool, or to become freakishly good at something.

That "slightly" is an important qualification. Spend time lavishly when you're young, but don't simply waste it. There's a big difference between doing something you worry might be a waste of time and doing something you know for sure will be. The former is at least a bet, and possibly a better one than you think. [\[23\]](#)

The most subtle advantage of youth, or more precisely of inexperience, is that you're seeing everything with fresh eyes. When your brain embraces an idea for the first time, sometimes the two don't fit together perfectly. Usually the problem is with your brain, but occasionally it's with the idea. A piece of it sticks out awkwardly and jabs you when you think about it. People who are used to the idea have learned to ignore it, but you have the opportunity not to. [\[24\]](#)

So when you're learning about something for the first time, pay attention to things that seem wrong or missing. You'll be tempted to ignore them, since there's a 99% chance the problem is with you. And you may have to set aside your misgivings



temporarily to keep progressing. But don't forget about them. When you've gotten further into the subject, come back and check if they're still there. If they're still viable in the light of your present knowledge, they probably represent an undiscovered idea.

One of the most valuable kinds of knowledge you get from experience is to know what you *don't* have to worry about. The young know all the things that could matter, but not their relative importance. So they worry equally about everything, when they should worry much more about a few things and hardly at all about the rest.

But what you don't know is only half the problem with inexperience. The other half is what you do know that ain't so. You arrive at adulthood with your head full of nonsense — bad habits you've acquired and false things you've been taught — and you won't be able to do great work till you clear away at least the nonsense in the way of whatever type of work you want to do.

Much of the nonsense left in your head is left there by schools. We're so used to schools that we unconsciously treat going to school as identical with learning, but in fact schools have all sorts of strange qualities that warp our ideas about learning and thinking.

For example, schools induce passivity. Since you were a small child, there was an authority at the front of the class telling all of you what you had to learn and then measuring whether you did. But neither classes nor tests are intrinsic to learning; they're just artifacts of the way schools are usually designed.

The sooner you overcome this passivity, the better. If you're still in school, try thinking of your education as your project, and your teachers as working for you rather than vice versa. That may seem a stretch, but it's not merely some weird thought experiment. It's the truth economically, and in the best case it's the truth intellectually as well. The best teachers don't want to be your bosses. They'd prefer it if you pushed ahead, using them as a source of advice, rather than being pulled by them through the material.

Schools also give you a misleading impression of what work is like. In school they tell you what the problems are, and they're almost always soluble using no more than you've been taught so far. In real life you have to figure out what the problems are, and you often don't know if they're soluble at all.

But perhaps the worst thing schools do to you is train you to win by hacking the test. You can't do great work by doing that. You can't trick God. So stop looking for that kind of shortcut. The way to beat the system is to focus on problems and solutions that others have overlooked, not to skimp on the work itself.

Don't think of yourself as dependent on some gatekeeper giving you a "big break." Even if this were true, the best way to get it would be to focus on doing good work rather than chasing influential people.

And don't take rejection by committees to heart. The qualities that impress admissions officers and prize committees are quite different from those required to do great work. The decisions of selection committees are only meaningful to the extent that they're part of a feedback loop, and very few are.

People new to a field will often copy existing work. There's nothing inherently bad about that. There's no better way to learn how something works than by trying to reproduce it. Nor does copying necessarily make your work unoriginal. Originality is the presence of new ideas, not the absence of old ones.

There's a good way to copy and a bad way. If you're going to copy something, do it openly instead of furtively, or worse still, unconsciously. This is what's meant by the famously misattributed phrase "Great artists steal." The really dangerous kind of copying, the kind that gives copying a bad name, is the kind that's done without realizing it, because you're nothing more than a train running on tracks laid down by someone else. But at the other extreme, copying can be a sign of superiority rather than subordination. [\[25\]](#)

In many fields it's almost inevitable that your early work will be in some sense based on other people's. Projects rarely arise in a vacuum. They're usually a reaction to previous work. When you're first starting out, you don't have any previous work; if you're going to react to something, it has to be someone else's. Once you're established, you can react to your own. But while the former gets called derivative and the latter doesn't, structurally the two cases are more similar than they seem.

Oddly enough, the very novelty of the most novel ideas sometimes makes them seem at first to be more derivative than they are. New discoveries often have to be conceived initially as variations of existing things, *even by their discoverers*, because there isn't yet the conceptual vocabulary to express them.

There are definitely some dangers to copying, though. One is that you'll tend to copy old things — things that were in their day at the frontier of knowledge, but no longer are.

And when you do copy something, don't copy every feature of it. Some will make you ridiculous if you do. Don't copy the manner of an eminent 50 year old professor if you're 18, for example, or the idiom of a Renaissance poem hundreds of years later.

Some of the features of things you admire are flaws they succeeded despite. Indeed, the features that are easiest to imitate are the most likely to be the flaws.

This is particularly true for behavior. Some talented people are jerks, and this sometimes makes it seem to the inexperienced that being a jerk is part of being talented. It isn't; being talented is merely how they get away with it.

One of the most powerful kinds of copying is to copy something from one field into another. History is so full of chance discoveries of this type that it's probably worth giving chance a hand by deliberately learning about other kinds of work. You can take ideas from quite distant fields if you let them be metaphors.

Negative examples can be as inspiring as positive ones. In fact you can sometimes learn more from things done badly than from things done well; sometimes it only becomes clear what's needed when it's missing.

If a lot of the best people in your field are collected in one place, it's usually a good idea to visit for a while. It will increase your ambition, and also, by showing you that these people are human, increase your self-confidence. [\[26\]](#)

If you're earnest you'll probably get a warmer welcome than you might expect. Most people who are very good at something are happy to talk about it with anyone who's genuinely interested. If they're really good at their work, then they probably have a hobbyist's interest in it, and hobbyists always want to talk about their hobbies.

It may take some effort to find the people who are really good, though. Doing great work has such prestige that in some places, particularly universities, there's a polite fiction that everyone is engaged in it. And that is far from true. People within universities can't say so openly, but the quality of the work being done in different departments varies immensely. Some departments have people doing great work; others have in the past; others never have.

Seek out the best colleagues. There are a lot of projects that can't be done alone, and even if you're working on one that can be, it's good to have other people to

encourage you and to bounce ideas off.

Colleagues don't just affect your work, though; they also affect you. So work with people you want to become like, because you will.

Quality is more important than quantity in colleagues. It's better to have one or two great ones than a building full of pretty good ones. In fact it's not merely better, but necessary, judging from history: the degree to which great work happens in clusters suggests that one's colleagues often make the difference between doing great work and not.

How do you know when you have sufficiently good colleagues? In my experience, when you do, you know. Which means if you're unsure, you probably don't. But it may be possible to give a more concrete answer than that. Here's an attempt: sufficiently good colleagues offer *surprising* insights. They can see and do things that you can't. So if you have a handful of colleagues good enough to keep you on your toes in this sense, you're probably over the threshold.

Most of us can benefit from collaborating with colleagues, but some projects require people on a larger scale, and starting one of those is not for everyone. If you want to run a project like that, you'll have to become a manager, and managing well takes aptitude and interest like any other kind of work. If you don't have them, there is no middle path: you must either force yourself to learn management as a second language, or avoid such projects. [\[27\]](#)

Husband your morale. It's the basis of everything when you're working on ambitious projects. You have to nurture and protect it like a living organism.

Morale starts with your view of life. You're more likely to do great work if you're an optimist, and more likely to if you think of yourself as lucky than if you think of yourself as a victim.

Indeed, work can to some extent protect you from your problems. If you choose work that's pure, its very difficulties will serve as a refuge from the difficulties of everyday life. If this is escapism, it's a very productive form of it, and one that has been used by some of the greatest minds in history.

Morale compounds via work: high morale helps you do good work, which increases your morale and helps you do even better work. But this cycle also operates in the other direction: if you're not doing good work, that can demoralize you and make it even harder to. Since it matters so much for this cycle to be running in the right direction, it can be a good idea to switch to easier work when you're stuck, just so you start to get something done.

One of the biggest mistakes ambitious people make is to allow setbacks to destroy their morale all at once, like a balloon bursting. You can inoculate yourself against this by explicitly considering setbacks a part of your process. Solving hard problems always involves some backtracking.

Doing great work is a depth-first search whose root node is the desire to. So "If at first you don't succeed, try, try again" isn't quite right. It should be: If at first you don't succeed, either try again, or backtrack and then try again.

"Never give up" is also not quite right. Obviously there are times when it's the right choice to eject. A more precise version would be: Never let setbacks panic you into backtracking more than you need to. Corollary: Never abandon the root node.

It's not necessarily a bad sign if work is a struggle, any more than it's a bad sign to be out of breath while running. It depends how fast you're running. So learn to distinguish good pain from bad. Good pain is a sign of effort; bad pain is a sign of damage.

An audience is a critical component of morale. If you're a scholar, your audience may be your peers; in the arts, it may be an audience in the traditional sense. Either way it doesn't need to be big. The value of an audience doesn't grow anything like linearly with its size. Which is bad news if you're famous, but good news if you're just starting out, because it means a small but dedicated audience can be enough to sustain you. If a handful of people genuinely love what you're doing, that's enough.

To the extent you can, avoid letting intermediaries come between you and your audience. In some types of work this is inevitable, but it's so liberating to escape it that you might be better off switching to an adjacent type if that will let you go direct. [\[28\]](#)

The people you spend time with will also have a big effect on your morale. You'll find there are some who increase your energy and others who decrease it, and the effect someone has is not always what you'd expect. Seek out the people who increase your energy and avoid those who decrease it. Though of course if there's someone you need to take care of, that takes precedence.

Don't marry someone who doesn't understand that you need to work, or sees your work as competition for your attention. If you're ambitious, you need to work; it's almost like a medical condition; so someone who won't let you work either doesn't understand you, or does and doesn't care.

Ultimately morale is physical. You think with your body, so it's important to take

care of it. That means exercising regularly, eating and sleeping well, and avoiding the more dangerous kinds of drugs. Running and walking are particularly good forms of exercise because they're good for thinking. [29]

People who do great work are not necessarily happier than everyone else, but they're happier than they'd be if they didn't. In fact, if you're smart and ambitious, it's dangerous *not* to be productive. People who are smart and ambitious but don't achieve much tend to become bitter.

It's ok to want to impress other people, but choose the right people. The opinion of people you respect is signal. Fame, which is the opinion of a much larger group you might or might not respect, just adds noise.

The prestige of a type of work is at best a trailing indicator and sometimes completely mistaken. If you do anything well enough, you'll make it prestigious. So the question to ask about a type of work is not how much prestige it has, but how well it could be done.

Competition can be an effective motivator, but don't let it choose the problem for you; don't let yourself get drawn into chasing something just because others are. In fact, don't let competitors make you do anything much more specific than work harder.

Curiosity is the best guide. Your curiosity never lies, and it knows more than you do about what's worth paying attention to.

Notice how often that word has come up. If you asked an oracle the secret to doing great work and the oracle replied with a single word, my bet would be on "curiosity."

That doesn't translate directly to advice. It's not enough just to be curious, and you can't command curiosity anyway. But you can nurture it and let it drive you.

Curiosity is the key to all four steps in doing great work: it will choose the field for you, get you to the frontier, cause you to notice the gaps in it, and drive you to explore them. The whole process is a kind of dance with curiosity.

Believe it or not, I tried to make this essay as short as I could. But its length at least means it acts as a filter. If you made it this far, you must be interested in doing great work. And if so you're already further along than you might realize, because the set of people willing to want to is small.

The factors in doing great work are factors in the literal, mathematical sense, and they are: ability, interest, effort, and luck. Luck by definition you can't do anything about, so we can ignore that. And we can assume effort, if you do in fact want to do great work. So the problem boils down to ability and interest. Can you find a kind of work where your ability and interest will combine to yield an explosion of new ideas?

Here there are grounds for optimism. There are so many different ways to do great work, and even more that are still undiscovered. Out of all those different types of work, the one you're most suited for is probably a pretty close match. Probably a comically close match. It's just a question of finding it, and how far into it your ability and interest can take you. And you can only answer that by trying.

Many more people could try to do great work than do. What holds them back is a combination of modesty and fear. It seems presumptuous to try to be Newton or Shakespeare. It also seems hard; surely if you tried something like that, you'd fail. Presumably the calculation is rarely explicit. Few people consciously decide not to try to do great work. But that's what's going on subconsciously; they shy away from the question.

So I'm going to pull a sneaky trick on you. Do you want to do great work, or not? Now you have to decide consciously. Sorry about that. I wouldn't have done it to a general audience. But we already know you're interested.

Don't worry about being presumptuous. You don't have to tell anyone. And if it's too hard and you fail, so what? Lots of people have worse problems than that. In fact you'll be lucky if it's the worst problem you have.

Yes, you'll have to work hard. But again, lots of people have to work hard. And if you're working on something you find very interesting, which you necessarily will if you're on the right path, the work will probably feel less burdensome than a lot of your peers'.

The discoveries are out there, waiting to be made. Why not by you?

## Notes

[1] I don't think you could give a precise definition of what counts as great work. Doing great work means doing something important so well that you expand people's ideas of what's possible. But there's no threshold for importance. It's a matter of degree, and often hard to judge at the time anyway. So I'd rather people focused on developing their interests rather than worrying about whether they're important or not. Just try to do something amazing, and leave it to future generations to say if you succeeded.

[2] A lot of standup comedy is based on noticing anomalies in everyday life. "Did you ever notice...?" New ideas come from doing this about nontrivial things. Which may help explain why people's reaction to a new idea is often the first half of laughing: Ha!

[3] That second qualifier is critical. If you're excited about something most authorities discount, but you can't give a more precise explanation than "they don't get it," then you're starting to drift into the territory of cranks.

[4] Finding something to work on is not simply a matter of finding a match between the current version of you and a list of known problems. You'll often have to coevolve with the problem. That's why it can sometimes be so hard to figure out what to work on. The search space is huge. It's the cartesian product of all possible types of work, both known and yet to be discovered, and all possible future versions of you.

There's no way you could search this whole space, so you have to rely on heuristics to generate promising paths through it and hope the best matches will be clustered. Which they will not always be; different types of work have been collected together as much by accidents of history as by the intrinsic similarities between them.

[5] There are many reasons curious people are more likely to do great work, but one of the more subtle is that, by casting a wide net, they're more likely to find the right thing to work on in the first place.

[6] It can also be dangerous to make things for an audience you feel is less sophisticated than you, if that causes you to talk down to them. You can make a lot of money doing that, if you do it in a sufficiently cynical way, but it's not the route to great work. Not that anyone using this m.o. would care.

[7] This idea I learned from Hardy's *A Mathematician's Apology*, which I recommend to anyone ambitious to do great work, in any field.

[8] Just as we overestimate what we can do in a day and underestimate what we can do over several years, we overestimate the damage done by procrastinating for a day and underestimate the damage done by procrastinating for several years.



[9] You can't usually get paid for doing exactly what you want, especially early on. There are two options: get paid for doing work close to what you want and hope to push it closer, or get paid for doing something else entirely and do your own projects on the side. Both can work, but both have drawbacks: in the first approach your work is compromised by default, and in the second you have to fight to get time to do it.

[10] If you set your life up right, it will deliver the focus-relax cycle automatically. The perfect setup is an office you work in and that you walk to and from.

[11] There may be some very unworldly people who do great work without consciously trying to. If you want to expand this rule to cover that case, it becomes: Don't try to be anything except the best.

[12] This gets more complicated in work like acting, where the goal is to adopt a fake persona. But even here it's possible to be affected. Perhaps the rule in such fields should be to avoid *unintentional* affectation.

[13] It's safe to have beliefs that you treat as unquestionable if and only if they're also unfalsifiable. For example, it's safe to have the principle that everyone should be treated equally under the law, because a sentence with a "should" in it isn't really a statement about the world and is therefore hard to disprove. And if there's no evidence that could disprove one of your principles, there can't be any facts you'd need to ignore in order to preserve it.

[14] Affectation is easier to cure than intellectual dishonesty. Affectation is often a shortcoming of the young that burns off in time, while intellectual dishonesty is more of a character flaw.

[15] Obviously you don't have to be working at the exact moment you have the idea, but you'll probably have been working fairly recently.

[16] Some say psychoactive drugs have a similar effect. I'm skeptical, but also almost totally ignorant of their effects.

[17] For example you might give the  $n$ th most important topic  $(m-1)/m^n$  of your attention, for some  $m > 1$ . You couldn't allocate your attention so precisely, of course, but this at least gives an idea of a reasonable distribution.

[18] The principles defining a religion have to be mistaken. Otherwise anyone might adopt them, and there would be nothing to distinguish the adherents of the religion from everyone else.

[19] It might be a good exercise to try writing down a list of questions you wondered about in your youth. You might find you're now in a position to do something about some of them.

[20] The connection between originality and uncertainty causes a strange phenomenon: because the conventional-minded are more certain than the independent-minded, this tends to give them the upper hand in disputes, even though they're generally stupider.

The best lack all conviction, while the worst  
Are full of passionate intensity.

[21] Derived from Linus Pauling's "If you want to have good ideas, you must have many ideas."

[22] Attacking a project as a "toy" is similar to attacking a statement as "inappropriate." It means that no more substantial criticism can be made to stick.

[23] One way to tell whether you're wasting time is to ask if you're producing or consuming. Writing computer games is less likely to be a waste of time than playing them, and playing games where you create something is less likely to be a waste of time than playing games where you don't.

[24] Another related advantage is that if you haven't said anything publicly yet, you won't be biased toward evidence that supports your earlier conclusions. With sufficient integrity you could achieve eternal youth in this respect, but few manage to. For most people, having previously published opinions has an effect similar to ideology, just in quantity 1.

[25] In the early 1630s Daniel Mytens made a painting of Henrietta Maria handing a laurel wreath to Charles I. Van Dyck then painted his own version to show how much better he was.

[26] I'm being deliberately vague about what a place is. As of this writing, being in the same physical place has advantages that are hard to duplicate, but that could change.

[27] This is false when the work the other people have to do is very constrained, as with SETI@home or Bitcoin. It may be possible to expand the area in which it's false by defining similarly restricted protocols with more freedom of action in the nodes.

[28] Corollary: Building something that enables people to go around intermediaries and engage directly with their audience is probably a good idea.

[29] It may be helpful always to walk or run the same route, because that frees attention for thinking. It feels that way to me, and there is some historical evidence for it.

**Thanks** to Trevor Blackwell, Daniel Gackle, Pam Graham, Tom Howard, Patrick

Hsu, Steve Huffman, Jessica Livingston, Henry Lloyd-Baker, Bob Metcalfe, Ben Miller, Robert Morris, Michael Nielsen, Courtenay Pipkin, Joris Poort, Mieke Roos, Rajat Suri, Harj Taggar, Garry Tan, and my younger son for suggestions and for reading drafts.

# Superlinear Returns

October 2023

One of the most important things I didn't understand about the world when I was a child is the degree to which the returns for performance are superlinear.

Teachers and coaches implicitly told us the returns were linear. "You get out," I heard a thousand times, "what you put in." They meant well, but this is rarely true. If your product is only half as good as your competitor's, you don't get half as many customers. You get no customers, and you go out of business.

It's obviously true that the returns for performance are superlinear in business. Some think this is a flaw of capitalism, and that if we changed the rules it would stop being true. But superlinear returns for performance are a feature of the world, not an artifact of rules we've invented. We see the same pattern in fame, power, military victories, knowledge, and even benefit to humanity. In all of these, the rich get richer. [\[1\]](#)

You can't understand the world without understanding the concept of superlinear returns. And if you're ambitious you definitely should, because this will be the wave you surf on.

It may seem as if there are a lot of different situations with superlinear returns, but as far as I can tell they reduce to two fundamental causes: exponential growth and thresholds.

The most obvious case of superlinear returns is when you're working on something that grows exponentially. For example, growing bacterial cultures. When they grow at all, they grow exponentially. But they're tricky to grow. Which means the difference in outcome between someone who's adept at it and someone who's not is very great.

Startups can also grow exponentially, and we see the same pattern there. Some manage to achieve high growth rates. Most don't. And as a result you get qualitatively different outcomes: the companies with high growth rates tend to

become immensely valuable, while the ones with lower growth rates may not even survive.

Y Combinator encourages founders to focus on growth rate rather than absolute numbers. It prevents them from being discouraged early on, when the absolute numbers are still low. It also helps them decide what to focus on: you can use growth rate as a compass to tell you how to evolve the company. But the main advantage is that by focusing on growth rate you tend to get something that grows exponentially.

YC doesn't explicitly tell founders that with growth rate "you get out what you put in," but it's not far from the truth. And if growth rate were proportional to performance, then the reward for performance  $p$  over time  $t$  would be proportional to  $p^t$ .

Even after decades of thinking about this, I find that sentence startling.

Whenever how well you do depends on how well you've done, you'll get exponential growth. But neither our DNA nor our customs prepare us for it. No one finds exponential growth natural; every child is surprised, the first time they hear it, by the story of the man who asks the king for a single grain of rice the first day and double the amount each successive day.

What we don't understand naturally we develop customs to deal with, but we don't have many customs about exponential growth either, because there have been so few instances of it in human history. In principle herding should have been one: the more animals you had, the more offspring they'd have. But in practice grazing land was the limiting factor, and there was no plan for growing that exponentially.

Or more precisely, no generally applicable plan. There *was* a way to grow one's territory exponentially: by conquest. The more territory you control, the more powerful your army becomes, and the easier it is to conquer new territory. This is why history is full of empires. But so few people created or ran empires that their experiences didn't affect customs very much. The emperor was a remote and terrifying figure, not a source of lessons one could use in one's own life.

The most common case of exponential growth in preindustrial times was probably scholarship. The more you know, the easier it is to learn new things. The result, then as now, was that some people were startlingly more knowledgeable than the rest about certain topics. But this didn't affect customs much either. Although empires of ideas can overlap and there can thus be far more emperors, in preindustrial times this type of empire had little practical effect. [2]

That has changed in the last few centuries. Now the emperors of ideas can design bombs that defeat the emperors of territory. But this phenomenon is still so new that we haven't fully assimilated it. Few even of the participants realize they're benefitting from exponential growth or ask what they can learn from other

instances of it.

The other source of superlinear returns is embodied in the expression "winner take all." In a sports match the relationship between performance and return is a step function: the winning team gets one win whether they do much better or just slightly better. [3]

The source of the step function is not competition per se, however. It's that there are thresholds in the outcome. You don't need competition to get those. There can be thresholds in situations where you're the only participant, like proving a theorem or hitting a target.

It's remarkable how often a situation with one source of superlinear returns also has the other. Crossing thresholds leads to exponential growth: the winning side in a battle usually suffers less damage, which makes them more likely to win in the future. And exponential growth helps you cross thresholds: in a market with network effects, a company that grows fast enough can shut out potential competitors.

Fame is an interesting example of a phenomenon that combines both sources of superlinear returns. Fame grows exponentially because existing fans bring you new ones. But the fundamental reason it's so concentrated is thresholds: there's only so much room on the A-list in the average person's head.

The most important case combining both sources of superlinear returns may be learning. Knowledge grows exponentially, but there are also thresholds in it. Learning to ride a bicycle, for example. Some of these thresholds are akin to machine tools: once you learn to read, you're able to learn anything else much faster. But the most important thresholds of all are those representing new discoveries. Knowledge seems to be fractal in the sense that if you push hard at the boundary of one area of knowledge, you sometimes discover a whole new field. And if you do, you get first crack at all the new discoveries to be made in it. Newton did this, and so did Durer and Darwin.

Are there general rules for finding situations with superlinear returns? The most obvious one is to seek work that compounds.

There are two ways work can compound. It can compound directly, in the sense that doing well in one cycle causes you to do better in the next. That happens for example when you're building infrastructure, or growing an audience or brand. Or work can compound by teaching you, since learning compounds. This second case is an interesting one because you may feel you're doing badly as it's happening. You may be failing to achieve your immediate goal. But if you're learning a lot, then you're getting exponential growth nonetheless.

This is one reason Silicon Valley is so tolerant of failure. People in Silicon Valley aren't blindly tolerant of failure. They'll only continue to bet on you if you're learning from your failures. But if you are, you are in fact a good bet: maybe your company didn't grow the way you wanted, but you yourself have, and that should yield results eventually.

Indeed, the forms of exponential growth that don't consist of learning are so often intermixed with it that we should probably treat this as the rule rather than the exception. Which yields another heuristic: always be learning. If you're not learning, you're probably not on a path that leads to superlinear returns.

But don't overoptimize *what* you're learning. Don't limit yourself to learning things that are already known to be valuable. You're learning; you don't know for sure yet what's going to be valuable, and if you're too strict you'll lop off the outliers.

What about step functions? Are there also useful heuristics of the form "seek thresholds" or "seek competition?" Here the situation is trickier. The existence of a threshold doesn't guarantee the game will be worth playing. If you play a round of Russian roulette, you'll be in a situation with a threshold, certainly, but in the best case you're no better off. "Seek competition" is similarly useless; what if the prize isn't worth competing for? Sufficiently fast exponential growth guarantees both the shape and magnitude of the return curve — because something that grows fast enough will grow big even if it's trivially small at first — but thresholds only guarantee the shape. [4]

A principle for taking advantage of thresholds has to include a test to ensure the game is worth playing. Here's one that does: if you come across something that's mediocre yet still popular, it could be a good idea to replace it. For example, if a company makes a product that people dislike yet still buy, then presumably they'd buy a better alternative if you made one. [5]

It would be great if there were a way to find promising intellectual thresholds. Is there a way to tell which questions have whole new fields beyond them? I doubt we could ever predict this with certainty, but the prize is so valuable that it would be useful to have predictors that were even a little better than random, and there's hope of finding those. We can to some degree predict when a research problem *isn't* likely to lead to new discoveries: when it seems legit but boring. Whereas the kind that do lead to new discoveries tend to seem very mystifying, but perhaps unimportant. (If they were mystifying and obviously important, they'd be famous open questions with lots of people already working on them.) So one heuristic here is to be driven by curiosity rather than careerism — to give free rein to your curiosity instead of working on what you're supposed to.

The prospect of superlinear returns for performance is an exciting one for the ambitious. And there's good news in this department: this territory is expanding in both directions. There are more types of work in which you can get superlinear returns, and the returns themselves are growing.

There are two reasons for this, though they're so closely intertwined that they're more like one and a half: progress in technology, and the decreasing importance of organizations.

Fifty years ago it used to be much more necessary to be part of an organization to work on ambitious projects. It was the only way to get the resources you needed, the only way to have colleagues, and the only way to get distribution. So in 1970 your prestige was in most cases the prestige of the organization you belonged to. And prestige was an accurate predictor, because if you weren't part of an organization, you weren't likely to achieve much. There were a handful of exceptions, most notably artists and writers, who worked alone using inexpensive tools and had their own brands. But even they were at the mercy of organizations for reaching audiences. [6]

A world dominated by organizations damped variation in the returns for performance. But this world has eroded significantly just in my lifetime. Now a lot more people can have the freedom that artists and writers had in the 20th century. There are lots of ambitious projects that don't require much initial funding, and lots of new ways to learn, make money, find colleagues, and reach audiences.

There's still plenty of the old world left, but the rate of change has been dramatic by historical standards. Especially considering what's at stake. It's hard to imagine a more fundamental change than one in the returns for performance.

Without the damping effect of institutions, there will be more variation in outcomes. Which doesn't imply everyone will be better off: people who do well will do even better, but those who do badly will do worse. That's an important point to bear in mind. Exposing oneself to superlinear returns is not for everyone. Most people will be better off as part of the pool. So who should shoot for superlinear returns? Ambitious people of two types: those who know they're so good that they'll be net ahead in a world with higher variation, and those, particularly the young, who can afford to risk trying it to find out. [7]

The switch away from institutions won't simply be an exodus of their current inhabitants. Many of the new winners will be people they'd never have let in. So the resulting democratization of opportunity will be both greater and more authentic than any tame intramural version the institutions themselves might have cooked up.



Not everyone is happy about this great unlocking of ambition. It threatens some vested interests and contradicts some ideologies. [8] But if you're an ambitious individual it's good news for you. How should you take advantage of it?

The most obvious way to take advantage of superlinear returns for performance is by doing exceptionally good work. At the far end of the curve, incremental effort is a bargain. All the more so because there's less competition at the far end — and not just for the obvious reason that it's hard to do something exceptionally well, but also because people find the prospect so intimidating that few even try. Which means it's not just a bargain to do exceptional work, but a bargain even to try to.

There are many variables that affect how good your work is, and if you want to be an outlier you need to get nearly all of them right. For example, to do something exceptionally well, you have to be interested in it. Mere diligence is not enough. So in a world with superlinear returns, it's even more valuable to know what you're interested in, and to find ways to work on it. [9] It will also be important to choose work that suits your circumstances. For example, if there's a kind of work that inherently requires a huge expenditure of time and energy, it will be increasingly valuable to do it when you're young and don't yet have children.

There's a surprising amount of technique to doing great work. It's not just a matter of trying hard. I'm going to take a shot giving a recipe in one paragraph.

Choose work you have a natural aptitude for and a deep interest in. Develop a habit of working on your own projects; it doesn't matter what they are so long as you find them excitingly ambitious. Work as hard as you can without burning out, and this will eventually bring you to one of the frontiers of knowledge. These look smooth from a distance, but up close they're full of gaps. Notice and explore such gaps, and if you're lucky one will expand into a whole new field. Take as much risk as you can afford; if you're not failing occasionally you're probably being too conservative. Seek out the best colleagues. Develop good taste and learn from the best examples. Be honest, especially with yourself. Exercise and eat and sleep well and avoid the more dangerous drugs. When in doubt, follow your curiosity. It never lies, and it knows more than you do about what's worth paying attention to. [10]

And there is of course one other thing you need: to be lucky. Luck is always a factor, but it's even more of a factor when you're working on your own rather than as part of an organization. And though there are some valid aphorisms about luck being where preparedness meets opportunity and so on, there's also a component of true chance that you can't do anything about. The solution is to take multiple shots. Which is another reason to start taking risks early.

The best example of a field with superlinear returns is probably science. It has exponential growth, in the form of learning, combined with thresholds at the

extreme edge of performance — literally at the limits of knowledge.

The result has been a level of inequality in scientific discovery that makes the wealth inequality of even the most stratified societies seem mild by comparison. Newton's discoveries were arguably greater than all his contemporaries' combined.

[11]

This point may seem obvious, but it might be just as well to spell it out. Superlinear returns imply inequality. The steeper the return curve, the greater the variation in outcomes.

In fact, the correlation between superlinear returns and inequality is so strong that it yields another heuristic for finding work of this type: look for fields where a few big winners outperform everyone else. A kind of work where everyone does about the same is unlikely to be one with superlinear returns.

What are fields where a few big winners outperform everyone else? Here are some obvious ones: sports, politics, art, music, acting, directing, writing, math, science, starting companies, and investing. In sports the phenomenon is due to externally imposed thresholds; you only need to be a few percent faster to win every race. In politics, power grows much as it did in the days of emperors. And in some of the other fields (including politics) success is driven largely by fame, which has its own source of superlinear growth. But when we exclude sports and politics and the effects of fame, a remarkable pattern emerges: the remaining list is exactly the same as the list of fields where you have to be [independent-minded](#) to succeed — where your ideas have to be not just correct, but novel as well. [12]

This is obviously the case in science. You can't publish papers saying things that other people have already said. But it's just as true in investing, for example. It's only useful to believe that a company will do well if most other investors don't; if everyone else thinks the company will do well, then its stock price will already reflect that, and there's no room to make money.

What else can we learn from these fields? In all of them you have to put in the initial effort. Superlinear returns seem small at first. *At this rate*, you find yourself thinking, *I'll never get anywhere*. But because the reward curve rises so steeply at the far end, it's worth taking extraordinary measures to get there.

In the startup world, the name for this principle is "do things that don't scale." If you pay a ridiculous amount of attention to your tiny initial set of customers, ideally you'll kick off exponential growth by word of mouth. But this same principle applies to anything that grows exponentially. Learning, for example. When you first start learning something, you feel lost. But it's worth making the initial effort to get a toehold, because the more you learn, the easier it will get.

There's another more subtle lesson in the list of fields with superlinear returns: not to equate work with a job. For most of the 20th century the two were identical for nearly everyone, and as a result we've inherited a custom that equates

productivity with having a job. Even now to most people the phrase "your work" means their job. But to a writer or artist or scientist it means whatever they're currently studying or creating. For someone like that, their work is something they carry with them from job to job, if they have jobs at all. It may be done for an employer, but it's part of their portfolio.

It's an intimidating prospect to enter a field where a few big winners outperform everyone else. Some people do this deliberately, but you don't need to. If you have sufficient natural ability and you follow your curiosity sufficiently far, you'll end up in one. Your curiosity won't let you be interested in boring questions, and interesting questions tend to create fields with superlinear returns if they're not already part of one.

The territory of superlinear returns is by no means static. Indeed, the most extreme returns come from expanding it. So while both ambition and curiosity can get you into this territory, curiosity may be the more powerful of the two. Ambition tends to make you climb existing peaks, but if you stick close enough to an interesting enough question, it may grow into a mountain beneath you.

## Notes

There's a limit to how sharply you can distinguish between effort, performance, and return, because they're not sharply distinguished in fact. What counts as return to one person might be performance to another. But though the borders of these concepts are blurry, they're not meaningless. I've tried to write about them as precisely as I could without crossing into error.

[1] Evolution itself is probably the most pervasive example of superlinear returns for performance. But this is hard for us to empathize with because we're not the recipients; we're the returns.

[2] Knowledge did of course have a practical effect before the Industrial Revolution. The development of agriculture changed human life completely. But this kind of change was the result of broad, gradual improvements in technique, not the discoveries of a few exceptionally learned people.

[3] It's not mathematically correct to describe a step function as superlinear, but a step function starting from zero works like a superlinear function when it describes the reward curve for effort by a rational actor. If it starts at zero then the part before the step is below any linearly increasing return, and the part after the step must be above the necessary return at that point or no one would bother.

[4] Seeking competition could be a good heuristic in the sense that some people find it motivating. It's also somewhat of a guide to promising problems, because it's a sign that other people find them promising. But it's a very imperfect sign: often there's a clamoring crowd chasing some problem, and they all end up being trumped by someone quietly working on another one.

[5] Not always, though. You have to be careful with this rule. When something is popular despite being mediocre, there's often a hidden reason why. Perhaps monopoly or regulation make it hard to compete. Perhaps customers have bad taste or have broken procedures for deciding what to buy. There are huge swathes of mediocre things that exist for such reasons.

[6] In my twenties I wanted to be an [artist](#) and even went to art school to study painting. Mostly because I liked art, but a nontrivial part of my motivation came from the fact that artists seemed least at the mercy of organizations.

[7] In principle everyone is getting superlinear returns. Learning compounds, and everyone learns in the course of their life. But in practice few push this kind of everyday learning to the point where the return curve gets really steep.

[8] It's unclear exactly what advocates of "equity" mean by it. They seem to disagree among themselves. But whatever they mean is probably at odds with a world in which institutions have less power to control outcomes, and a handful of outliers do much better than everyone else.

It may seem like bad luck for this concept that it arose at just the moment when the world was shifting in the opposite direction, but I don't think this was a coincidence. I think one reason it arose now is because its adherents feel threatened by rapidly increasing variation in performance.

[9] Corollary: Parents who pressure their kids to work on something prestigious, like medicine, even though they have no interest in it, will be hosing them even more than they have in the past.

[10] The original version of this paragraph was the first draft of "[How to Do Great Work](#)." As soon as I wrote it I realized it was a more important topic than superlinear returns, so I paused the present essay to expand this paragraph into its own. Practically nothing remains of the original version, because after I finished "How to Do Great Work" I rewrote it based on that.

[11] Before the Industrial Revolution, people who got rich usually did it like emperors: capturing some resource made them more powerful and enabled them

to capture more. Now it can be done like a scientist, by discovering or building something uniquely valuable. Most people who get rich use a mix of the old and the new ways, but in the most advanced economies the ratio has [shifted dramatically](#) toward discovery just in the last half century.

[12] It's not surprising that conventional-minded people would dislike inequality if independent-mindedness is one of the biggest drivers of it. But it's not simply that they don't want anyone to have what they can't. The conventional-minded literally can't imagine what it's like to have novel ideas. So the whole phenomenon of great variation in performance seems unnatural to them, and when they encounter it they assume it must be due to cheating or to some malign external influence.

**Thanks** to Trevor Blackwell, Patrick Collison, Tyler Cowen, Jessica Livingston, Harj Taggar, and Garry Tan for reading drafts of this.

# The Best Essay

March 2024

Despite its title this isn't meant to be the best essay. My goal here is to figure out what the best essay would be like.

It would be well-written, but you can write well about any topic. What made it special would be what it was about.

Obviously some topics would be better than others. It probably wouldn't be about this year's lipstick colors. But it wouldn't be vaporous talk about elevated themes either. A good essay has to be surprising. It has to tell people something they don't already know.

The best essay would be on the most important topic you could tell people something surprising about.

That may sound obvious, but it has some unexpected consequences. One is that science enters the picture like an elephant stepping into a rowboat. For example, Darwin first described the idea of natural selection in an essay written in 1844. Talk about an important topic you could tell people something surprising about. If that's the test of a great essay, this was surely the best one written in 1844. And indeed, the best possible essay at any given time would usually be one describing the most important scientific or technological discovery it was possible to make. [\[1\]](#)

Another unexpected consequence: I imagined when I started writing this that the best essay would be fairly timeless — that the best essay you could write in 1844 would be much the same as the best one you could write now. But in fact the opposite seems to be true. It might be true that the best painting would be timeless in this sense. But it wouldn't be impressive to write an essay introducing natural selection now. The best essay *now* would be one describing a great discovery we didn't yet know about.

If the question of how to write the best possible essay reduces to the question of how to make great discoveries, then I started with the wrong question. Perhaps what this exercise shows is that we shouldn't waste our time writing essays but instead focus on making discoveries in some specific domain. But I'm interested in essays and what can be done with them, so I want to see if there's some other question I could have asked.

There is, and on the face of it, it seems almost identical to the one I started with. Instead of asking *what would the best essay be?* I should have asked *how do you write essays well?* Though these seem only phrasing apart, their answers diverge. The answer to the first question, as we've seen, isn't really about essay writing. The second question forces it to be.

Writing essays, at its best, is a way of discovering ideas. How do you do that well? How do you discover by writing?

An essay should ordinarily start with what I'm going to call a question, though I mean this in a very general sense: it doesn't have to be a question grammatically, just something that acts like one in the sense that it spurs some response.

How do you get this initial question? It probably won't work to choose some important-sounding topic at random and go at it. Professional traders won't even trade unless they have what they call an *edge* — a convincing story about why in some class of trades they'll win more than they lose. Similarly, you shouldn't attack a topic unless you have a way in — some new insight about it or way of approaching it.

You don't need to have a complete thesis; you just need some kind of gap you can explore. In fact, merely having questions about something other people take for granted can be edge enough.

If you come across a question that's sufficiently puzzling, it could be worth exploring even if it doesn't seem very momentous. Many an important discovery has been made by pulling on a thread that seemed insignificant at first. How can they *all* be finches? [\[2\]](#)

Once you've got a question, then what? You start thinking out loud about it. Not literally out loud, but you commit to a specific string of words in response, as you would if you were talking. This initial response is usually mistaken or incomplete. Writing converts your ideas from vague to bad. But that's a step forward, because once you can see the brokenness, you can fix it.

Perhaps beginning writers are alarmed at the thought of starting with something mistaken or incomplete, but you shouldn't be, because this is why essay writing works. Forcing yourself to commit to some specific string of words gives you a starting point, and if it's wrong, you'll see that when you reread it. At least half of essay writing is rereading what you've written and asking *is this correct and complete?* You have to be very strict when rereading, not just because you want to keep yourself honest, but because a gap between your response and the truth is often a sign of new ideas to be discovered.

The prize for being strict with what you've written is not just refinement. When you take a roughly correct answer and try to make it exactly right, sometimes you find that you can't, and that the reason is that you were depending on a false

assumption. And when you discard it, the answer turns out to be completely different. [3]

Ideally the response to a question is two things: the first step in a process that converges on the truth, and a source of additional questions (in my very general sense of the word). So the process continues recursively, as response spurs response. [4]

Usually there are several possible responses to a question, which means you're traversing a tree. But essays are linear, not tree-shaped, which means you have to choose one branch to follow at each point. How do you choose? Usually you should follow whichever offers the greatest combination of generality and novelty. I don't consciously rank branches this way; I just follow whichever seems most exciting; but generality and novelty are what make a branch exciting. [5]

If you're willing to do a lot of rewriting, you don't have to guess right. You can follow a branch and see how it turns out, and if it isn't good enough, cut it and backtrack. I do this all the time. In this essay I've already cut a 17-paragraph subtree, in addition to countless shorter ones. Maybe I'll reattach it at the end, or boil it down to a footnote, or spin it off as its own essay; we'll see. [6]

In general you want to be quick to cut. One of the most dangerous temptations in writing (and in software and painting) is to keep something that isn't right, just because it contains a few good bits or cost you a lot of effort.

The most surprising new question being thrown off at this point is *does it really matter what the initial question is?* If the space of ideas is highly connected, it shouldn't, because you should be able to get from any question to the most valuable ones in a few hops. And we see evidence that it's highly connected in the way, for example, that people who are obsessed with some topic can turn any conversation toward it. But that only works if you know where you want to go, and you don't in an essay. That's the whole point. You don't want to be the obsessive conversationalist, or all your essays will be about the same thing. [7]

The other reason the initial question matters is that you usually feel somewhat obliged to stick to it. I don't think about this when I decide which branch to follow. I just follow novelty and generality. Sticking to the question is enforced later, when I notice I've wandered too far and have to backtrack. But I think this is the optimal solution. You don't want the hunt for novelty and generality to be constrained in the moment. Go with it and see what you get. [8]

Since the initial question does constrain you, in the best case it sets an upper bound on the quality of essay you'll write. If you do as well as you possibly can on the chain of thoughts that follow from the initial question, the initial question itself is the only place where there's room for variation.

It would be a mistake to let this make you too conservative though, because you can't predict where a question will lead. Not if you're doing things right, because



doing things right means making discoveries, and by definition you can't predict those. So the way to respond to this situation is not to be cautious about which initial question you choose, but to write a lot of essays. Essays are for taking risks.

Almost any question can get you a good essay. Indeed, it took some effort to think of a sufficiently unpromising topic in the third paragraph, because any essayist's first impulse on hearing that the best essay couldn't be about x would be to try to write it. But if most questions yield good essays, only some yield great ones.

Can we predict which questions will yield great essays? Considering how long I've been writing essays, it's alarming how novel that question feels.

One thing I like in an initial question is outrageousness. I love questions that seem naughty in some way — for example, by seeming counterintuitive or overambitious or heterodox. Ideally all three. This essay is an example. Writing about the best essay implies there is such a thing, which pseudo-intellectuals will dismiss as reductive, though it follows necessarily from the possibility of one essay being better than another. And thinking about how to do something so ambitious is close enough to doing it that it holds your attention.

I like to start an essay with a gleam in my eye. This could be just a taste of mine, but there's one aspect of it that probably isn't: to write a really good essay on some topic, you have to be interested in it. A good writer can write well about anything, but to stretch for the novel insights that are the *raison d'être* of the essay, you have to care.

If caring about it is one of the criteria for a good initial question, then the optimal question varies from person to person. It also means you're more likely to write great essays if you care about a lot of different things. The more curious you are, the greater the probable overlap between the set of things you're curious about and the set of topics that yield great essays.

What other qualities would a great initial question have? It's probably good if it has implications in a lot of different areas. And I find it's a good sign if it's one that people think has already been thoroughly explored. But the truth is that I've barely thought about how to choose initial questions, because I rarely do it. I rarely *choose* what to write about; I just start thinking about something, and sometimes it turns into an essay.

Am I going to stop writing essays about whatever I happen to be thinking about and instead start working my way through some systematically generated list of topics? That doesn't sound like much fun. And yet I want to write good essays, and if the initial question matters, I should care about it.

Perhaps the answer is to go one step earlier: to write about whatever pops into your head, but try to ensure that what pops into your head is good. Indeed, now that I think about it, this has to be the answer, because a mere list of topics wouldn't be any use if you didn't have edge with any of them. To start writing an

essay, you need a topic plus some initial insight about it, and you can't generate those systematically. If only. [9]

You can probably cause yourself to have more of them, though. The quality of the ideas that come out of your head depends on what goes in, and you can improve that in two dimensions, breadth and depth.

You can't learn everything, so getting breadth implies learning about topics that are very different from one another. When I tell people about my book-buying trips to Hay and they ask what I buy books about, I usually feel a bit sheepish answering, because the topics seem like a laundry list of unrelated subjects. But perhaps that's actually optimal in this business.

You can also get ideas by talking to people, by doing and building things, and by going places and seeing things. I don't think it's important to talk to new people so much as the sort of people who make you have new ideas. I get more new ideas after talking for an afternoon with Robert Morris than from talking to 20 new smart people. I know because that's what a block of office hours at Y Combinator consists of.

While breadth comes from reading and talking and seeing, depth comes from doing. The way to really learn about some domain is to have to solve problems in it. Though this could take the form of writing, I suspect that to be a good essayist you also have to do, or have done, some other kind of work. That may not be true for most other fields, but essay writing is different. You could spend half your time working on something else and be net ahead, so long as it was hard.

I'm not proposing that as a recipe so much as an encouragement to those already doing it. If you've spent all your life so far working on other things, you're already halfway there. Though of course to be good at writing you have to like it, and if you like writing you'd probably have spent at least some time doing it.

Everything I've said about initial questions applies also to the questions you encounter in writing the essay. They're the same thing; every subtree of an essay is usually a shorter essay, just as every subtree of a Calder mobile is a smaller mobile. So any technique that gets you good initial questions also gets you good whole essays.

At some point the cycle of question and response reaches what feels like a natural end. Which is a little suspicious; shouldn't every answer suggest more questions? I think what happens is that you start to feel sated. Once you've covered enough interesting ground, you start to lose your appetite for new questions. Which is just as well, because the reader is probably feeling sated too. And it's not lazy to stop asking questions, because you could instead be asking the initial question of a new essay.

That's the ultimate source of drag on the connectedness of ideas: the discoveries you make along the way. If you discover enough starting from question A, you'll

never make it to question B. Though if you keep writing essays you'll gradually fix this problem by burning off such discoveries. So bizarrely enough, writing lots of essays makes it as if the space of ideas were more highly connected.

When a subtree comes to an end, you can do one of two things. You can either stop, or pull the Cubist trick of laying separate subtrees end to end by returning to a question you skipped earlier. Usually it requires some sleight of hand to make the essay flow continuously at this point, but not this time. This time I actually need an example of the phenomenon. For example, we discovered earlier that the best possible essay wouldn't usually be timeless in the way the best painting would. This seems surprising enough to be worth investigating further.

There are two senses in which an essay can be timeless: to be about a matter of permanent importance, and always to have the same effect on readers. With art these two senses blend together. Art that looked beautiful to the ancient Greeks still looks beautiful to us. But with essays the two senses diverge, because essays teach, and you can't teach people something they already know. Natural selection is certainly a matter of permanent importance, but an essay explaining it couldn't have the same effect on us that it would have had on Darwin's contemporaries, precisely because his ideas were so successful that everyone already knows about them. [\[10\]](#)

I imagined when I started writing this that the best possible essay would be timeless in the stricter, evergreen sense: that it would contain some deep, timeless wisdom that would appeal equally to Aristotle and Feynman. That doesn't seem to be true. But if the best possible essay wouldn't usually be timeless in this stricter sense, what would it take to write essays that were?

The answer to that turns out to be very strange: to be the evergreen kind of timeless, an essay has to be ineffective, in the sense that its discoveries aren't assimilated into our shared culture. Otherwise there will be nothing new in it for the second generation of readers. If you want to surprise readers not just now but in the future as well, you have to write essays that won't stick — essays that, no matter how good they are, won't become part of what people in the future learn before they read them. [\[11\]](#)

I can imagine several ways to do that. One would be to write about things people never learn. For example, it's a long-established pattern for ambitious people to chase after various types of prizes, and only later, perhaps too late, to realize that some of them weren't worth as much as they thought. If you write about that, you can be confident of a conveyor belt of future readers to be surprised by it.

Ditto if you write about the tendency of the inexperienced to overdo things — of young engineers to produce overcomplicated solutions, for example. There are some kinds of mistakes people never learn to avoid except by making them. Any of those should be a timeless topic.

Sometimes when we're slow to grasp things it's not just because we're obtuse or in

denial but because we've been deliberately lied to. There are a lot of things adults [lie](#) to kids about, and when you reach adulthood, they don't take you aside and hand you a list of them. They don't remember which lies they told you, and most were implicit anyway. So contradicting such lies will be a source of surprises for as long as adults keep telling them.

Sometimes it's systems that lie to you. For example, the educational systems in most countries train you to win by [hacking the test](#). But that's not how you win at the most important real-world tests, and after decades of training, this is hard for new arrivals in the real world to grasp. Helping them overcome such institutional lies will work as long as the institutions remain broken. [\[12\]](#)

Another recipe for timelessness is to write about things readers already know, but in much more detail than can be transmitted culturally. "Everyone knows," for example, that it can be rewarding to have [kids](#). But till you have them you don't know precisely what forms that takes, and even then much of what you know you may never have put into words.

I've written about all these kinds of topics. But I didn't do it in a deliberate attempt to write essays that were timeless in the stricter sense. And indeed, the fact that this depends on one's ideas not sticking suggests that it's not worth making a deliberate attempt to. You should write about topics of timeless importance, yes, but if you do such a good job that your conclusions stick and future generations find your essay obvious instead of novel, so much the better. You've crossed into Darwin territory.

Writing about topics of timeless importance is an instance of something even more general, though: breadth of applicability. And there are more kinds of breadth than chronological — applying to lots of different fields, for example. So breadth is the ultimate aim.

I already aim for it. Breadth and novelty are the two things I'm always chasing. But I'm glad I understand where timelessness fits.

I understand better where a lot of things fit now. This essay has been a kind of tour of essay writing. I started out hoping to get advice about topics; if you assume good writing, the only thing left to differentiate the best essay is its topic. And I did get advice about topics: discover natural selection. Yeah, that would be nice. But when you step back and ask what's the best you can do short of making some great discovery like that, the answer turns out to be about procedure. Ultimately the quality of an essay is a function of the ideas discovered in it, and the way you get them is by casting a wide net for questions and then being very exacting with the answers.

The most striking feature of this map of essay writing are the alternating stripes of inspiration and effort required. The questions depend on inspiration, but the answers can be got by sheer persistence. You don't have to get an answer right the first time, but there's no excuse for not getting it right eventually, because you

can keep rewriting till you do. And this is not just a theoretical possibility. It's a pretty accurate description of the way I work. I'm rewriting as we speak.

But although I wish I could say that writing great essays depends mostly on effort, in the limit case it's inspiration that makes the difference. In the limit case, the questions are the harder thing to get. That pool has no bottom.

How to get more questions? That is the most important question of all.

## Notes

[1] There might be some resistance to this conclusion on the grounds that some of these discoveries could only be understood by a small number of readers. But you get into all sorts of difficulties if you want to disqualify essays on this account. How do you decide where the cutoff should be? If a virus kills off everyone except a handful of people sequestered at Los Alamos, could an essay that had been disqualified now be eligible? Etc.

Darwin's 1844 essay was derived from an earlier version written in 1839. Extracts from it were published in 1858.

[2] When you find yourself very curious about an apparently minor question, that's an exciting sign. Evolution has designed you to pay attention to things that matter. So when you're very curious about something random, that could mean you've unconsciously noticed it's less random than it seems.

[3] Corollary: If you're not intellectually honest, your writing won't just be biased, but also boring, because you'll miss all the ideas you'd have discovered if you pushed for the truth.

[4] Sometimes this process begins before you start writing. Sometimes you've already figured out the first few things you want to say. Schoolchildren are often taught they should decide *everything* they want to say, and write this down as an outline before they start writing the essay itself. Maybe that's a good way to get them started — or not, I don't know — but it's antithetical to the spirit of essay writing. The more detailed your outline, the less your ideas can benefit from the sort of discovery that essays are for.

[5] The problem with this type of "greedy" algorithm is that you can end up on a local maximum. If the most valuable question is preceded by a boring one, you'll overlook it. But I can't imagine a better strategy. There's no lookahead except by writing. So use a greedy algorithm and a lot of time.

[6] I ended up reattaching the first 5 of the 17 paragraphs, and discarding the rest.

[7] Stephen Fry confessed to making use of this phenomenon when taking exams at Oxford. He had in his head a standard essay about some general literary topic, and he would find a way to turn the exam question toward it and then just reproduce it again.

Strictly speaking it's the graph of ideas that would be highly connected, not the space, but that usage would confuse people who don't know graph theory, whereas people who do know it will get what I mean if I say "space".

[8] Too far doesn't depend just on the distance from the original topic. It's more like that distance divided by the value of whatever I've discovered in the subtree.

[9] Or can you? I should try writing about this. Even if the chance of succeeding is small, the expected value is huge.

[10] There was a vogue in the 20th century for saying that the purpose of art was also to teach. Some artists tried to justify their work by explaining that their goal was not to produce something good, but to challenge our preconceptions about art. And to be fair, art can teach somewhat. The ancient Greeks' naturalistic sculptures represented a new idea, and must have been extra exciting to contemporaries on that account. But they still look good to us.

[11] Bertrand Russell caused huge controversy in the early 20th century with his ideas about "trial marriage." But they make boring reading now, because they prevailed. "Trial marriage" is what we call "dating."

[12] If you'd asked me 10 years ago, I'd have predicted that schools would continue to teach hacking the test for centuries. But now it seems plausible that students will soon be taught individually by AIs, and that exams will be replaced by ongoing, invisible micro-assessments.

**Thanks** to Sam Altman, Trevor Blackwell, Jessica Livingston, Robert Morris, Courtenay Pipkin, and Harj Taggar for reading drafts of this.

# How to Start Google

March 2024

*(This is a talk I gave to 14 and 15 year olds about what to do now if they might want to start a startup later. Lots of schools think they should tell students something about startups. This is what I think they should tell them.)*

Most of you probably think that when you're released into the so-called real world you'll eventually have to get some kind of job. That's not true, and today I'm going to talk about a trick you can use to avoid ever having to get a job.

The trick is to start your own company. So it's not a trick for avoiding *work*, because if you start your own company you'll work harder than you would if you had an ordinary job. But you will avoid many of the annoying things that come with a job, including a boss telling you what to do.

It's more exciting to work on your own project than someone else's. And you can also get a lot richer. In fact, this is the standard way to get [really rich](#). If you look at the lists of the richest people that occasionally get published in the press, nearly all of them did it by starting their own companies.

Starting your own company can mean anything from starting a barber shop to starting Google. I'm here to talk about one extreme end of that continuum. I'm going to tell you how to start Google.

The companies at the Google end of the continuum are called startups when they're young. The reason I know about them is that my wife Jessica and I started something called Y Combinator that is basically a startup factory. Since 2005, Y Combinator has funded over 4000 startups. So we know exactly what you need to start a startup, because we've helped people do it for the last 19 years.

You might have thought I was joking when I said I was going to tell you how to start Google. You might be thinking "How could we start Google?" But that's effectively what the people who did start Google were thinking before they started it. If you'd told Larry Page and Sergey Brin, the founders of Google, that the company they were about to start would one day be worth over a trillion dollars, their heads would have exploded.

All you can know when you start working on a startup is that it seems worth

pursuing. You can't know whether it will turn into a company worth billions or one that goes out of business. So when I say I'm going to tell you how to start Google, I mean I'm going to tell you how to get to the point where you can start a company that has as much chance of being Google as Google had of being Google.

[1]

How do you get from where you are now to the point where you can start a successful startup? You need three things. You need to be good at some kind of technology, you need an idea for what you're going to build, and you need cofounders to start the company with.

How do you get good at technology? And how do you choose which technology to get good at? Both of those questions turn out to have the same answer: work on your own projects. Don't try to guess whether gene editing or LLMs or rockets will turn out to be the most valuable technology to know about. No one can predict that. Just work on whatever interests you the most. You'll work much harder on something you're interested in than something you're doing because you think you're supposed to.

If you're not sure what technology to get good at, get good at programming. That has been the source of the median startup for the last 30 years, and this is probably not going to change in the next 10.

Those of you who are taking computer science classes in school may at this point be thinking, ok, we've got this sorted. We're already being taught all about programming. But sorry, this is not enough. You have to be working on your own projects, not just learning stuff in classes. You can do well in computer science classes without ever really learning to program. In fact you can graduate with a degree in computer science from a top university and still not be any good at programming. That's why tech companies all make you take a coding test before they'll hire you, regardless of where you went to university or how well you did there. They know grades and exam results prove nothing.

If you really want to learn to program, you have to work on your own projects. You learn so much faster that way. Imagine you're writing a game and there's something you want to do in it, and you don't know how. You're going to figure out how a lot faster than you'd learn anything in a class.

You don't have to learn programming, though. If you're wondering what counts as technology, it includes practically everything you could describe using the words "make" or "build." So welding would count, or making clothes, or making videos. Whatever you're most interested in. The critical distinction is whether you're producing or just consuming. Are you writing computer games, or just playing them? That's the cutoff.

Steve Jobs, the founder of Apple, spent time when he was a teenager studying calligraphy — the sort of beautiful writing that you see in medieval manuscripts. No one, including him, thought that this would help him in his career. He was just



doing it because he was interested in it. But it turned out to help him a lot. The computer that made Apple really big, the Macintosh, came out at just the moment when computers got powerful enough to make letters like the ones in printed books instead of the computery-looking letters you see in 8 bit games. Apple destroyed everyone else at this, and one reason was that Steve was one of the few people in the computer business who really got graphic design.

Don't feel like your projects have to be *serious*. They can be as frivolous as you like, so long as you're building things you're excited about. Probably 90% of programmers start out building games. They and their friends like to play games. So they build the kind of things they and their friends want. And that's exactly what you should be doing at 15 if you want to start a startup one day.

You don't have to do just one project. In fact it's good to learn about multiple things. Steve Jobs didn't just learn calligraphy. He also learned about electronics, which was even more valuable. Whatever you're interested in. (Do you notice a theme here?)

So that's the first of the three things you need, to get good at some kind or kinds of technology. You do it the same way you get good at the violin or football: practice. If you start a startup at 22, and you start writing your own programs now, then by the time you start the company you'll have spent at least 7 years practicing writing code, and you can get pretty good at anything after practicing it for 7 years.

Let's suppose you're 22 and you've succeeded: You're now really good at some technology. How do you get [startup ideas](#)? It might seem like that's the hard part. Even if you are a good programmer, how do you get the idea to start Google?

Actually it's easy to get startup ideas once you're good at technology. Once you're good at some technology, when you look at the world you see dotted outlines around the things that are missing. You start to be able to see both the things that are missing from the technology itself, and all the broken things that could be fixed using it, and each one of these is a potential startup.

In the town near our house there's a shop with a sign warning that the door is hard to close. The sign has been there for several years. To the people in the shop it must seem like this mysterious natural phenomenon that the door sticks, and all they can do is put up a sign warning customers about it. But any carpenter looking at this situation would think "why don't you just plane off the part that sticks?"

Once you're good at programming, all the missing software in the world starts to become as obvious as a sticking door to a carpenter. I'll give you a real world example. Back in the 20th century, American universities used to publish printed directories with all the students' names and contact info. When I tell you what these directories were called, you'll know which startup I'm talking about. They were called facebook, because they usually had a picture of each student next to their name.

So Mark Zuckerberg shows up at Harvard in 2002, and the university still hasn't gotten the facebook online. Each individual house has an online facebook, but there isn't one for the whole university. The university administration has been diligently having meetings about this, and will probably have solved the problem in another decade or so. Most of the students don't consciously notice that anything is wrong. But Mark is a programmer. He looks at this situation and thinks "Well, this is stupid. I could write a program to fix this in one night. Just let people upload their own photos and then combine the data into a new site for the whole university." So he does. And almost literally overnight he has thousands of users.

Of course Facebook was not a startup yet. It was just a... project. There's that word again. Projects aren't just the best way to learn about technology. They're also the best source of startup ideas.

Facebook was not unusual in this respect. Apple and Google also began as projects. Apple wasn't meant to be a company. Steve Wozniak just wanted to build his own computer. It only turned into a company when Steve Jobs said "Hey, I wonder if we could sell plans for this computer to other people." That's how Apple started. They weren't even selling computers, just plans for computers. Can you imagine how lame this company seemed?

Ditto for Google. Larry and Sergey weren't trying to start a company at first. They were just trying to make search better. Before Google, most search engines didn't try to sort the results they gave you in order of importance. If you searched for "rugby" they just gave you every web page that contained the word "rugby." And the web was so small in 1997 that this actually worked! Kind of. There might only be 20 or 30 pages with the word "rugby," but the web was growing exponentially, which meant this way of doing search was becoming exponentially more broken. Most users just thought, "Wow, I sure have to look through a lot of search results to find what I want." Door sticks. But like Mark, Larry and Sergey were programmers. Like Mark, they looked at this situation and thought "Well, this is stupid. Some pages about rugby matter more than others. Let's figure out which those are and show them first."

It's obvious in retrospect that this was a great idea for a startup. It wasn't obvious at the time. It's never obvious. If it was obviously a good idea to start Apple or Google or Facebook, someone else would have already done it. That's why the best startups grow out of projects that aren't meant to be startups. You're not trying to start a company. You're just following your instincts about what's interesting. And if you're young and good at technology, then your unconscious instincts about what's interesting are better than your conscious ideas about what would be a good company.

So it's critical, if you're a young founder, to build things for yourself and your friends to use. The biggest mistake young founders make is to build something for some mysterious group of other people. But if you can make something that you and your friends truly want to use — something your friends aren't just using out

of loyalty to you, but would be really sad to lose if you shut it down — then you almost certainly have the germ of a good startup idea. It may not seem like a startup to you. It may not be obvious how to make money from it. But trust me, there's a way.

What you need in a startup idea, and all you need, is something your friends actually want. And those ideas aren't hard to see once you're good at technology. There are sticking doors everywhere. [2]

Now for the third and final thing you need: a cofounder, or cofounders. The optimal startup has two or three founders, so you need one or two cofounders. How do you find them? Can you predict what I'm going to say next? It's the same thing: projects. You find cofounders by working on projects with them. What you need in a cofounder is someone who's good at what they do and that you work well with, and the only way to judge this is to work with them on things.

At this point I'm going to tell you something you might not want to hear. It really matters to do well in your classes, even the ones that are just memorization or blathering about literature, because you need to do well in your classes to get into a good university. And if you want to start a startup you should try to get into the best university you can, because that's where the best cofounders are. It's also where the best employees are. When Larry and Sergey started Google, they began by just hiring all the smartest people they knew out of Stanford, and this was a real advantage for them.

The empirical evidence is clear on this. If you look at where the largest numbers of successful startups come from, it's pretty much the same as the list of the most selective universities.

I don't think it's the prestigious names of these universities that cause more good startups to come out of them. Nor do I think it's because the quality of the teaching is better. What's driving this is simply the difficulty of getting in. You have to be pretty smart and determined to get into MIT or Cambridge, so if you do manage to get in, you'll find the other students include a lot of smart and determined people. [3]

You don't have to start a startup with someone you meet at university. The founders of Twitch met when they were seven. The founders of Stripe, Patrick and John Collison, met when John was born. But universities are the main source of cofounders. And because they're where the cofounders are, they're also where the ideas are, because the best ideas grow out of projects you do with the people who become your cofounders.

So the list of what you need to do to get from here to starting a startup is quite short. You need to get good at technology, and the way to do that is to work on your own projects. And you need to do as well in school as you can, so you can get into a good university, because that's where the cofounders and the ideas are.

That's it, just two things, build stuff and do well in school.

## Notes

[1] The rhetorical trick in this sentence is that the "Google"s refer to different things. What I mean is: a company that has as much chance of growing as big as Google ultimately did as Larry and Sergey could have reasonably expected Google itself would at the time they started it. But I think the original version is zippier.

[2] Making something for your friends isn't the only source of startup ideas. It's just the best source for young founders, who have the least knowledge of what other people want, and whose own wants are most predictive of future demand.

[3] Strangely enough this is particularly true in countries like the US where undergraduate admissions are done badly. US admissions departments make applicants jump through a lot of arbitrary hoops that have little to do with their intellectual ability. But the more arbitrary a test, the more it becomes a test of mere determination and resourcefulness. And those are the two most important qualities in startup founders. So US admissions departments are better at selecting founders than they would be if they were better at selecting students.

**Thanks** to Jared Friedman, Carolynn Levy, Jessica Livingston, Harj Taggar, and Garry Tan for reading drafts of this.

# The Reddits

March 2024

I met the Reddits before we even started Y Combinator. In fact they were one of the reasons we started it.

YC grew out of a talk I gave to the Harvard Computer Society (the undergrad computer club) about how to start a startup. Everyone else in the audience was probably local, but Steve and Alexis came up on the train from the University of Virginia, where they were seniors. Since they'd come so far I agreed to meet them for coffee. They told me about the startup idea we'd later fund them to drop: a way to order fast food on your cellphone.

This was before smartphones. They'd have had to make deals with cell carriers and fast food chains just to get it launched. So it was not going to happen. It still doesn't exist, 19 years later. But I was impressed with their brains and their energy. In fact I was so impressed with them and some of the other people I met at that talk that I decided to start something to fund them. A few days later I told Steve and Alexis that we were starting Y Combinator, and encouraged them to apply.

That first batch we didn't have any way to identify applicants, so we made up nicknames for them. The Reddits were the "Cell food muffins." "Muffin" is a term of endearment Jessica uses for things like small dogs and two year olds. So that gives you some idea what kind of impression Steve and Alexis made in those days. They had the look of slightly ruffled surprise that baby birds have.

Their idea was bad though. And since we thought then that we were funding ideas rather than founders, we rejected them. But we felt bad about it. Jessica was sad that we'd rejected the muffins. And it seemed wrong to me to turn down the people we'd been inspired to start YC to fund.

I don't think the startup sense of the word "pivot" had been invented yet, but we wanted to fund Steve and Alexis, so if their idea was bad, they'd have to work on something else. And I knew what else. In those days there was a site called Delicious where you could save links. It had a page called [del.icio.us/popular](http://del.icio.us/popular) that listed the most-saved links, and people were using this page as a de facto Reddit. I knew because a lot of the traffic to my site was coming from it. There needed to be something like [del.icio.us/popular](http://del.icio.us/popular), but designed for sharing links instead of being a

byproduct of saving them.

So I called Steve and Alexis and said that we liked them, just not their idea, so we'd fund them if they'd work on something else. They were on the train home to Virginia at that point. They got off at the next station and got on the next train north, and by the end of the day were committed to working on what's now called Reddit.

They would have liked to call it Snoo, as in "What snoo?" But snoo.com was too expensive, so they settled for calling the mascot Snoo and picked a name for the site that wasn't registered. Early on Reddit was just a provisional name, or so they told me at least, but it's probably too late to change it now.

As with all the really great startups, there's an uncannily close match between the company and the founders. Steve in particular. Reddit has a certain personality — curious, skeptical, ready to be amused — and that personality is Steve's.

Steve will roll his eyes at this, but he's an intellectual; he's interested in ideas for their own sake. That was how he came to be in that audience in Cambridge in the first place. He knew me because he was interested in a programming language I've written about called Lisp, and Lisp is one of those languages few people learn except out of intellectual curiosity. Steve's kind of vacuum-cleaner curiosity is exactly what you want when you're starting a site that's a list of links to literally anything interesting.

Steve was not a big fan of authority, so he also liked the idea of a site without editors. In those days the top forum for programmers was a site called Slashdot. It was a lot like Reddit, except the stories on the frontpage were chosen by human moderators. And though they did a good job, that one small difference turned out to be a big difference. Being driven by user submissions meant Reddit was fresher than Slashdot. News there was newer, and users will always go where the newest news is.

I pushed the Reddits to launch fast. A version one didn't need to be more than a couple hundred lines of code. How could that take more than a week or two to build? And they did launch comparatively fast, about three weeks into the first YC batch. The first users were Steve, Alexis, me, and some of their YC batchmates and college friends. It turns out you don't need that many users to collect a decent list of interesting links, especially if you have multiple accounts per user.

Reddit got two more people from their YC batch: Chris Slowe and Aaron Swartz, and they too were unusually smart. Chris was just finishing his PhD in physics at Harvard. Aaron was younger, a college freshman, and even more anti-authority than Steve. It's not exaggerating to describe him as a martyr for what authority later did to him.

Slowly but inexorably Reddit's traffic grew. At first the numbers were so small they were hard to distinguish from background noise. But within a few weeks it was

clear that there was a core of real users returning regularly to the site. And although all kinds of things have happened to Reddit the company in the years since, Reddit the *site* never looked back.

Reddit the site (and now app) is such a fundamentally useful thing that it's almost unkillable. Which is why, despite a long stretch after Steve left when the management strategy ranged from benign neglect to spectacular blunders, traffic just kept growing. You can't do that with most companies. Most companies you take your eye off the ball for six months and you're in deep trouble. But Reddit was special, and when Steve came back in 2015, I knew the world was in for a surprise.

People thought they had Reddit's number: one of the players in Silicon Valley, but not one of the big ones. But those who knew what had been going on behind the scenes knew there was more to the story than this. If Reddit could grow to the size it had with management that was harmless at best, what could it do if Steve came back? We now know the answer to that question. Or at least a lower bound on the answer. Steve is not out of ideas yet.

# The Right Kind of Stubborn

July 2024

Successful people tend to be persistent. New ideas often don't work at first, but they're not deterred. They keep trying and eventually find something that does.

Mere obstinacy, on the other hand, is a recipe for failure. Obstinate people are so annoying. They won't listen. They beat their heads against a wall and get nowhere.

But is there any real difference between these two cases? Are persistent and obstinate people actually behaving differently? Or are they doing the same thing, and we just label them later as persistent or obstinate depending on whether they turned out to be right or not?

If that's the only difference then there's nothing to be learned from the distinction. Telling someone to be persistent rather than obstinate would just be telling them to be right rather than wrong, and they already know that. Whereas if persistence and obstinacy are actually different kinds of behavior, it would be worthwhile to tease them apart. [\[1\]](#)

I've talked to a lot of determined people, and it seems to me that they're different kinds of behavior. I've often walked away from a conversation thinking either "Wow, that guy is determined" or "Damn, that guy is stubborn," and I don't think I'm just talking about whether they seemed right or not. That's part of it, but not all of it.

There's something annoying about the obstinate that's not simply due to being mistaken. They won't listen. And that's not true of all determined people. I can't think of anyone more determined than the Collison brothers, and when you point out a problem to them, they not only listen, but listen with an almost predatory intensity. Is there a hole in the bottom of their boat? Probably not, but if there is, they want to know about it.

It's the same with most successful people. They're never *more* engaged than when you disagree with them. Whereas the obstinate don't want to hear you. When you point out problems, their eyes glaze over, and their replies sound like ideologues talking about matters of doctrine. [\[2\]](#)

The reason the persistent and the obstinate seem similar is that they're both hard



to stop. But they're hard to stop in different senses. The persistent are like boats whose engines can't be throttled back. The obstinate are like boats whose rudders can't be turned. [3]

In the degenerate case they're indistinguishable: when there's only one way to solve a problem, your only choice is whether to give up or not, and persistence and obstinacy both say no. This is presumably why the two are so often conflated in popular culture. It assumes simple problems. But as problems get more complicated, we can see the difference between them. The persistent are much more attached to points high in the decision tree than to minor ones lower down, while the obstinate spray "don't give up" indiscriminately over the whole tree.

The persistent are attached to the goal. The obstinate are attached to their ideas about how to reach it.

Worse still, that means they'll tend to be attached to their *first* ideas about how to solve a problem, even though these are the least informed by the experience of working on it. So the obstinate aren't merely attached to details, but disproportionately likely to be attached to wrong ones.

Why are they like this? Why are the obstinate obstinate? One possibility is that they're overwhelmed. They're not very capable. They take on a hard problem. They're immediately in over their head. So they grab onto ideas the way someone on the deck of a rolling ship might grab onto the nearest handhold.

That was my initial theory, but on examination it doesn't hold up. If being obstinate were simply a consequence of being in over one's head, you could make persistent people become obstinate by making them solve harder problems. But that's not what happens. If you handed the Collisons an extremely hard problem to solve, they wouldn't become obstinate. If anything they'd become less obstinate. They'd know they had to be open to anything.

Similarly, if obstinacy were caused by the situation, the obstinate would stop being obstinate when solving easier problems. But they don't. And if obstinacy isn't caused by the situation, it must come from within. It must be a feature of one's personality.

Obstinacy is a reflexive resistance to changing one's ideas. This is not identical with stupidity, but they're closely related. A reflexive resistance to changing one's ideas becomes a sort of induced stupidity as contrary evidence mounts. And obstinacy is a form of not giving up that's easily practiced by the stupid. You don't have to consider complicated tradeoffs; you just dig in your heels. It even works, up to a point.

The fact that obstinacy works for simple problems is an important clue. Persistence and obstinacy aren't opposites. The relationship between them is more like the

relationship between the two kinds of respiration we can do: aerobic respiration, and the anaerobic respiration we inherited from our most distant ancestors. Anaerobic respiration is a more primitive process, but it has its uses. When you leap suddenly away from a threat, that's what you're using.

The optimal amount of obstinacy is not zero. It can be good if your initial reaction to a setback is an unthinking "I won't give up," because this helps prevent panic. But unthinking only gets you so far. The further someone is toward the obstinate end of the continuum, the less likely they are to succeed in solving hard problems.

[\[4\]](#)

Obstinacy is a simple thing. Animals have it. But persistence turns out to have a fairly complicated internal structure.

One thing that distinguishes the persistent is their energy. At the risk of putting too much weight on words, they persist rather than merely resisting. They keep trying things. Which means the persistent must also be imaginative. To keep trying things, you have to keep thinking of things to try.

Energy and imagination make a wonderful combination. Each gets the best out of the other. Energy creates demand for the ideas produced by imagination, which thus produces more, and imagination gives energy somewhere to go. [\[5\]](#)

Merely having energy and imagination is quite rare. But to solve hard problems you need three more qualities: resilience, good judgement, and a focus on some kind of goal.

Resilience means not having one's morale destroyed by setbacks. Setbacks are inevitable once problems reach a certain size, so if you can't bounce back from them, you can only do good work on a small scale. But resilience is not the same as obstinacy. Resilience means setbacks can't change your morale, not that they can't change your mind.

Indeed, persistence often requires that one change one's mind. That's where good judgement comes in. The persistent are quite rational. They focus on expected value. It's this, not recklessness, that lets them work on things that are unlikely to succeed.

There is one point at which the persistent are often irrational though: at the very top of the decision tree. When they choose between two problems of roughly equal expected value, the choice usually comes down to personal preference. Indeed, they'll often classify projects into deliberately wide bands of expected value in order to ensure that the one they want to work on still qualifies.

Empirically this doesn't seem to be a problem. It's ok to be irrational near the top of the decision tree. One reason is that we humans will work harder on a problem

we love. But there's another more subtle factor involved as well: our preferences among problems aren't random. When we love a problem that other people don't, it's often because we've unconsciously noticed that it's more important than they realize.

Which leads to our fifth quality: there needs to be some overall goal. If you're like me you began, as a kid, merely with the desire to do something great. In theory that should be the most powerful motivator of all, since it includes everything that could possibly be done. But in practice it's not much use, precisely because it includes too much. It doesn't tell you what to do at this moment.

So in practice your energy and imagination and resilience and good judgement have to be directed toward some fairly specific goal. Not too specific, or you might miss a great discovery adjacent to what you're searching for, but not too general, or it won't work to motivate you. [6]

When you look at the internal structure of persistence, it doesn't resemble obstinacy at all. It's so much more complex. Five distinct qualities — energy, imagination, resilience, good judgement, and focus on a goal — combine to produce a phenomenon that seems a bit like obstinacy in the sense that it causes you not to give up. But the way you don't give up is completely different. Instead of merely resisting change, you're driven toward a goal by energy and resilience, through paths discovered by imagination and optimized by judgement. You'll give way on any point low down in the decision tree, if its expected value drops sufficiently, but energy and resilience keep pushing you toward whatever you chose higher up.

Considering what it's made of, it's not surprising that the right kind of stubbornness is so much rarer than the wrong kind, or that it gets so much better results. Anyone can do obstinacy. Indeed, kids and drunks and fools are best at it. Whereas very few people have enough of all five of the qualities that produce the right kind of stubbornness, but when they do the results are magical.

## Notes

[1] I'm going to use "persistent" for the good kind of stubborn and "obstinate" for the bad kind, but I can't claim I'm simply following current usage. Conventional opinion barely distinguishes between good and bad kinds of stubbornness, and usage is correspondingly promiscuous. I could have invented a new word for the good kind, but it seemed better just to stretch "persistent."

[2] There are some domains where one can succeed by being obstinate. Some

political leaders have been notorious for it. But it won't work in situations where you have to pass external tests. And indeed the political leaders who are famous for being obstinate are famous for getting power, not for using it well.

[3] There will be some resistance to turning the rudder of a persistent person, because there's some cost to changing direction.

[4] The obstinate do sometimes succeed in solving hard problems. One way is through luck: like the stopped clock that's right twice a day, they seize onto some arbitrary idea, and it turns out to be right. Another is when their obstinacy cancels out some other form of error. For example, if a leader has overcautious subordinates, their estimates of the probability of success will always be off in the same direction. So if he mindlessly says "push ahead regardless" in every borderline case, he'll usually turn out to be right.

[5] If you stop there, at just energy and imagination, you get the conventional caricature of an artist or poet.

[6] Start by erring on the small side. If you're inexperienced you'll inevitably err on one side or the other, and if you err on the side of making the goal too broad, you won't get anywhere. Whereas if you err on the small side you'll at least be moving forward. Then, once you're moving, you expand the goal.

**Thanks** to Trevor Blackwell, Jessica Livingston, Jackie McDonough, Courtenay Pipkin, Harj Taggar, and Garry Tan for reading drafts of this.

# Founder Mode

September 2024

At a YC event last week Brian Chesky gave a talk that everyone who was there will remember. Most founders I talked to afterward said it was the best they'd ever heard. Ron Conway, for the first time in his life, forgot to take notes. I'm not going to try to reproduce it here. Instead I want to talk about a question it raised.

The theme of Brian's talk was that the conventional wisdom about how to run larger companies is mistaken. As Airbnb grew, well-meaning people advised him that he had to run the company in a certain way for it to scale. Their advice could be optimistically summarized as "hire good people and give them room to do their jobs." He followed this advice and the results were disastrous. So he had to figure out a better way on his own, which he did partly by studying how Steve Jobs ran Apple. So far it seems to be working. Airbnb's free cash flow margin is now among the best in Silicon Valley.

The audience at this event included a lot of the most successful founders we've funded, and one after another said that the same thing had happened to them. They'd been given the same advice about how to run their companies as they grew, but instead of helping their companies, it had damaged them.

Why was everyone telling these founders the wrong thing? That was the big mystery to me. And after mulling it over for a bit I figured out the answer: what they were being told was how to run a company you hadn't founded — how to run a company if you're merely a professional manager. But this m.o. is so much less effective that to founders it feels broken. There are things founders can do that managers can't, and not doing them feels wrong to founders, because it is.

In effect there are two different ways to run a company: founder mode and manager mode. Till now most people even in Silicon Valley have implicitly assumed that scaling a startup meant switching to manager mode. But we can infer the existence of another mode from the dismay of founders who've tried it, and the success of their attempts to escape from it.

There are as far as I know no books specifically about founder mode. Business schools don't know it exists. All we have so far are the experiments of individual founders who've been figuring it out for themselves. But now that we know what we're looking for, we can search for it. I hope in a few years founder mode will be

as well understood as manager mode. We can already guess at some of the ways it will differ.

The way managers are taught to run companies seems to be like modular design in the sense that you treat subtrees of the org chart as black boxes. You tell your direct reports what to do, and it's up to them to figure out how. But you don't get involved in the details of what they do. That would be micromanaging them, which is bad.

Hire good people and give them room to do their jobs. Sounds great when it's described that way, doesn't it? Except in practice, judging from the report of founder after founder, what this often turns out to mean is: hire professional fakers and let them drive the company into the ground.

One theme I noticed both in Brian's talk and when talking to founders afterward was the idea of being gaslit. Founders feel like they're being gaslit from both sides — by the people telling them they have to run their companies like managers, and by the people working for them when they do. Usually when everyone around you disagrees with you, your default assumption should be that you're mistaken. But this is one of the rare exceptions. VCs who haven't been founders themselves don't know how founders should run companies, and C-level execs, as a class, include some of the most skillful liars in the world. [\[1\]](#)

Whatever founder mode consists of, it's pretty clear that it's going to break the principle that the CEO should engage with the company only via his or her direct reports. "Skip-level" meetings will become the norm instead of a practice so unusual that there's a name for it. And once you abandon that constraint there are a huge number of permutations to choose from.

For example, Steve Jobs used to run an annual retreat for what he considered the 100 most important people at Apple, and these were not the 100 people highest on the org chart. Can you imagine the force of will it would take to do this at the average company? And yet imagine how useful such a thing could be. It could make a big company feel like a startup. Steve presumably wouldn't have kept having these retreats if they didn't work. But I've never heard of another company doing this. So is it a good idea, or a bad one? We still don't know. That's how little we know about founder mode. [\[2\]](#)

Obviously founders can't keep running a 2000 person company the way they ran it when it had 20. There's going to have to be some amount of delegation. Where the borders of autonomy end up, and how sharp they are, will probably vary from company to company. They'll even vary from time to time within the same company, as managers earn trust. So founder mode will be more complicated than manager mode. But it will also work better. We already know that from the examples of individual founders groping their way toward it.

Indeed, another prediction I'll make about founder mode is that once we figure out what it is, we'll find that a number of individual founders were already most of the

way there — except that in doing what they did they were regarded by many as eccentric or worse. [3]

Curiously enough it's an encouraging thought that we still know so little about founder mode. Look at what founders have achieved already, and yet they've achieved this against a headwind of bad advice. Imagine what they'll do once we can tell them how to run their companies like Steve Jobs instead of John Sculley.

## Notes

[1] The more diplomatic way of phrasing this statement would be to say that experienced C-level execs are often very skilled at managing up. And I don't think anyone with knowledge of this world would dispute that.

[2] If the practice of having such retreats became so widespread that even mature companies dominated by politics started to do it, we could quantify the senescence of companies by the average depth on the org chart of those invited.

[3] I also have another less optimistic prediction: as soon as the concept of founder mode becomes established, people will start misusing it. Founders who are unable to delegate even things they should will use founder mode as the excuse. Or managers who aren't founders will decide they should try to act like founders. That may even work, to some extent, but the results will be messy when it doesn't; the modular approach does at least limit the damage a bad CEO can do.

**Thanks** to Brian Chesky, Patrick Collison, Ron Conway, Jessica Livingston, Elon Musk, Ryan Petersen, Harj Taggar, and Garry Tan for reading drafts of this.

# When To Do What You Love

September 2024

There's some debate about whether it's a good idea to "follow your passion." In fact the question is impossible to answer with a simple yes or no. Sometimes you should and sometimes you shouldn't, but the border between should and shouldn't is very complicated. The only way to give a general answer is to trace it.

When people talk about this question, there's always an implicit "instead of." All other things being equal, why wouldn't you work on what interests you the most? So even raising the question implies that all other things aren't equal, and that you have to choose between working on what interests you the most and something else, like what pays the best.

And indeed if your main goal is to make money, you can't usually afford to work on what interests you the most. People pay you for doing what they want, not what you want. But there's an obvious exception: when you both want the same thing. For example, if you love football, and you're good enough at it, you can get paid a lot to play it.

Of course the odds are against you in a case like football, because so many other people like playing it too. This is not to say you shouldn't try though. It depends how much ability you have and how hard you're willing to work.

The odds are better when you have strange tastes: when you like something that pays well and that few other people like. For example, it's clear that Bill Gates truly loved running a software company. He didn't just love programming, which a lot of people do. He loved writing software for customers. That is a very strange taste indeed, but if you have it, you can make a lot by indulging it.

There are even some people who have a genuine intellectual interest in making money. This is distinct from mere greed. They just can't help noticing when something is mispriced, and can't help doing something about it. It's like a puzzle for them. [1]

In fact there's an edge case here so spectacular that it turns all the preceding advice on its head. If you want to make a really huge amount of money — hundreds of millions or even billions of dollars — it turns out to be very useful to work on what interests you the most. The reason is not the extra motivation you



get from doing this, but that the way to make a really large amount of money is to start a startup, and working on what interests you is an excellent way to discover [startup ideas](#).

Many if not most of the biggest startups began as projects the founders were doing for fun. Apple, Google, and Facebook all began that way. Why is this pattern so common? Because the best ideas tend to be such outliers that you'd overlook them if you were consciously looking for ways to make money. Whereas if you're young and good at technology, your unconscious instincts about what would be interesting to work on are very well aligned with what needs to be built.

So there's something like a midwit peak for making money. If you don't need to make much, you can work on whatever you're most interested in; if you want to become moderately rich, you can't usually afford to; but if you want to become super rich, and you're young and good at technology, working on what you're most interested in becomes a good idea again.

What if you're not sure what you want? What if you're attracted to the idea of making money and more attracted to some kinds of work than others, but neither attraction predominates? How do you break ties?

The key here is to understand that such ties are only apparent. When you have trouble choosing between following your interests and making money, it's never because you have complete knowledge of yourself and of the types of work you're choosing between, and the options are perfectly balanced. When you can't decide which path to take, it's almost always due to ignorance. In fact you're usually suffering from three kinds of ignorance simultaneously: you don't know what makes you happy, what the various kinds of work are really like, or how well you could do them. [2]

In a way this ignorance is excusable. It's often hard to predict these things, and no one even tells you that you need to. If you're ambitious you're told you should go to college, and this is good advice so far as it goes, but that's where it usually ends. No one tells you how to figure out what to work on, or how hard this can be.

What do you do in the face of uncertainty? Get more certainty. And probably the best way to do that is to try working on things you're interested in. That will get you more information about how interested you are in them, how good you are at them, and how much scope they offer for ambition.

Don't wait. Don't wait till the end of college to figure out what to work on. Don't even wait for internships during college. You don't necessarily need a job doing x in order to work on x; often you can just start doing it in some form yourself. And since figuring out what to work on is a problem that could take years to solve, the sooner you start, the better.

One useful trick for judging different kinds of work is to look at who your colleagues will be. You'll become like whoever you work with. Do you want to

become like these people?

Indeed, the difference in character between different kinds of work is magnified by the fact that everyone else is facing the same decisions as you. If you choose a kind of work mainly for how well it pays, you'll be surrounded by other people who chose it for the same reason, and that will make it even more soul-sucking than it seems from the outside. Whereas if you choose work you're genuinely interested in, you'll be surrounded mostly by other people who are genuinely interested in it, and that will make it extra inspiring. [3]

The other thing you do in the face of uncertainty is to make choices that are uncertainty-proof. The less sure you are about what to do, the more important it is to choose options that give you more options in the future. I call this "staying upwind." If you're unsure whether to major in math or economics, for example, choose math; math is upwind of economics in the sense that it will be easier to switch later from math to economics than from economics to math.

There's one case, though, where it's easy to say whether you should work on what interests you the most: if you want to do [great work](#). This is not a sufficient condition for doing great work, but it is a necessary one.

There's a lot of selection bias in advice about whether to "follow your passion," and this is the reason. Most such advice comes from people who are famously successful, and if you ask someone who's famously successful how to do what they did, most will tell you that you have to work on what you're most interested in. And this is in fact true.

That doesn't mean it's the right advice for everyone. Not everyone can do great work, or wants to. But if you do want to, the complicated question of whether or not to work on what interests you the most becomes simple. The answer is yes. The root of great work is a sort of ambitious curiosity, and you can't manufacture that.

## Notes

[1] These examples show why it's a mistake to assume that economic inequality must be evidence of some kind of brokenness or unfairness. It's obvious that different people have different interests, and that some interests yield far more money than others, so how can it not be obvious that some people will end up much richer than others? In a world where some people like to write enterprise

software and others like to make studio pottery, economic inequality is the natural outcome.

[2] Difficulty choosing between interests is a different matter. That's not always due to ignorance. It's often intrinsically difficult. I still have trouble doing it.

[3] You can't always take people at their word on this. Since it's more prestigious to work on things you're interested in than to be driven by money, people who are driven mainly by money will often claim to be more interested in their work than they actually are. One way to test such claims is by doing the following thought experiment: if their work didn't pay well, would they take day jobs doing something else in order to do it in their spare time? Lots of mathematicians and scientists and engineers would. Historically lots *have*. But I don't think as many investment bankers would.

This thought experiment is also useful for distinguishing between university departments.

**Thanks** to Trevor Blackwell, Paul Buchheit, Jessica Livingston, Robert Morris, Harj Taggar, and Garry Tan for reading drafts of this.

# Writes and Write-Notes

October 2024

I'm usually reluctant to make predictions about technology, but I feel fairly confident about this one: in a couple decades there won't be many people who can write.

One of the strangest things you learn if you're a writer is how many people have trouble writing. Doctors know how many people have a mole they're worried about; people who are good at setting up computers know how many people aren't; writers know how many people need help writing.

The reason so many people have trouble writing is that it's fundamentally difficult. To write well you have to think clearly, and thinking clearly is hard.

And yet writing pervades many jobs, and the more prestigious the job, the more writing it tends to require.

These two powerful opposing forces, the pervasive expectation of writing and the irreducible difficulty of doing it, create enormous pressure. This is why eminent professors often turn out to have resorted to plagiarism. The most striking thing to me about these cases is the pettiness of the thefts. The stuff they steal is usually the most mundane boilerplate — the sort of thing that anyone who was even halfway decent at writing could turn out with no effort at all. Which means they're not even halfway decent at writing.

Till recently there was no convenient escape valve for the pressure created by these opposing forces. You could pay someone to write for you, like JFK, or plagiarize, like MLK, but if you couldn't buy or steal words, you had to write them yourself. And as a result nearly everyone who was expected to write had to learn how.

Not anymore. AI has blown this world open. Almost all pressure to write has dissipated. You can have AI do it for you, both in school and at work.

The result will be a world divided into writes and write-nots. There will still be some people who can write. Some of us like it. But the middle ground between those who are good at writing and those who can't write at all will disappear. Instead of good writers, ok writers, and people who can't write, there will just be

good writers and people who can't write.

Is that so bad? Isn't it common for skills to disappear when technology makes them obsolete? There aren't many blacksmiths left, and it doesn't seem to be a problem.

Yes, it's bad. The reason is something I mentioned earlier: writing is thinking. In fact there's a kind of thinking that can only be done by writing. You can't make this point better than Leslie Lamport did:

If you're thinking without writing, you only think you're thinking.

So a world divided into writes and write-nots is more dangerous than it sounds. It will be a world of thinks and think-nots. I know which half I want to be in, and I bet you do too.

This situation is not unprecedented. In preindustrial times most people's jobs made them strong. Now if you want to be strong, you work out. So there are still strong people, but only those who choose to be.

It will be the same with writing. There will still be smart people, but only those who choose to be.

**Thanks** to Jessica Livingston, Ben Miller, and Robert Morris for reading drafts of this.

# The Origins of Wokeness

January 2025

The word "prig" isn't very common now, but if you look up the definition, it will sound familiar. Google's isn't bad:

A self-righteously moralistic person who behaves as if superior to others.

This sense of the word originated in the 18th century, and its age is an important clue: it shows that although wokeness is a comparatively recent phenomenon, it's an instance of a much older one.

There's a certain kind of person who's attracted to a shallow, exacting kind of moral purity, and who demonstrates his purity by attacking anyone who breaks the rules. Every society has these people. All that changes is the rules they enforce. In Victorian England it was Christian virtue. In Stalin's Russia it was orthodox Marxism-Leninism. For the woke, it's social justice.

So if you want to understand wokeness, the question to ask is not why people behave this way. Every society has prigs. The question to ask is why our prigs are priggish about these ideas, at this moment. And to answer that we have to ask when and where wokeness began.

The answer to the first question is the 1980s. Wokeness is a second, more aggressive wave of political correctness, which started in the late 1980s, died down in the late 1990s, and then returned with a vengeance in the early 2010s, finally peaking after the riots of 2020.

This was not the original meaning of "woke," but it's rarely used in the original sense now. Now the pejorative sense is the dominant one. What *does* it mean now? I've often been asked to define both wokeness and political correctness by people who think they're meaningless labels, so I will. They both have the same definition:

An aggressively performative focus on social justice.

In other words, it's people being prigs about social justice. And that's the real problem — the performativeness, not the social justice.

Racism, for example, is a genuine problem. Not a problem on the scale that the

woke believe it to be, but a genuine one. I don't think any reasonable person would deny that. The problem with political correctness was not that it focused on marginalized groups, but the shallow, aggressive way in which it did so. Instead of going out into the world and quietly helping members of marginalized groups, the politically correct focused on getting people in trouble for using the wrong words to talk about them.

As for where political correctness began, if you think about it, you probably already know the answer. Did it begin outside universities and spread to them from this external source? Obviously not; it has always been most extreme in universities. So where in universities did it begin? Did it begin in math, or the hard sciences, or engineering, and spread from there to the humanities and social sciences? Those are amusing images, but no, obviously it began in the humanities and social sciences.

Why there? And why then? What happened in the humanities and social sciences in the 1980s?

A successful theory of the origin of political correctness has to be able to explain why it didn't happen earlier. Why didn't it happen during the protest movements of the 1960s, for example? They were concerned with much the same issues. [1]

The reason the student protests of the 1960s didn't lead to political correctness was precisely that — they were student movements. They didn't have any real power. The students may have been talking a lot about women's liberation and black power, but it was not what they were being taught in their classes. Not yet.

But in the early 1970s the student protestors of the 1960s began to finish their dissertations and get hired as professors. At first they were neither powerful nor numerous. But as more of their peers joined them and the previous generation of professors started to retire, they gradually became both.

The reason political correctness began in the humanities and social sciences was that these fields offered more scope for the injection of politics. A 1960s radical who got a job as a physics professor could still attend protests, but his political beliefs wouldn't affect his work. Whereas research in sociology and modern literature can be made as political as you like. [2]

I saw political correctness arise. When I started college in 1982 it was not yet a thing. Female students might object if someone said something they considered sexist, but no one was getting *reported* for it. It was still not a thing when I started grad school in 1986. It was definitely a thing in 1988 though, and by the early 1990s it seemed to pervade campus life.

What happened? How did protest become punishment? Why were the late 1980s the point at which protests against male chauvinism (as it used to be called) morphed into formal complaints to university authorities about sexism? Basically, the 1960s radicals got tenure. They became the Establishment they'd protested

against two decades before. Now they were in a position not just to speak out about their ideas, but to enforce them.

A new set of moral rules to enforce was exciting news to a certain kind of student. What made it particularly exciting was that they were allowed to attack professors. I remember noticing that aspect of political correctness at the time. It wasn't simply a grass-roots student movement. It was faculty members encouraging students to attack other faculty members. In that respect it was like the Cultural Revolution. That wasn't a grass-roots movement either; that was Mao unleashing the younger generation on his political opponents. And in fact when Roderick MacFarquhar started teaching a class on the Cultural Revolution at Harvard in the late 1980s, many saw it as a comment on current events. I don't know if it actually was, but people thought it was, and that means the similarities were obvious. [3]

College students larp. It's their nature. It's usually harmless. But larping morality turned out to be a poisonous combination. The result was a kind of moral etiquette, superficial but very complicated. Imagine having to explain to a well-meaning visitor from another planet why using the phrase "people of color" is considered particularly enlightened, but saying "colored people" gets you fired. And why exactly one isn't supposed to use the word "negro" now, even though Martin Luther King used it constantly in his speeches. There are no underlying principles. You'd just have to give him a long list of rules to memorize. [4]

The danger of these rules was not just that they created land mines for the unwary, but that their elaborateness made them an effective substitute for virtue. Whenever a society has a concept of heresy and orthodoxy, orthodoxy becomes a substitute for virtue. You can be the worst person in the world, but as long as you're orthodox you're better than everyone who isn't. This makes orthodoxy very attractive to bad people.

But for it to work as a substitute for virtue, orthodoxy must be difficult. If all you have to do to be orthodox is wear some garment or avoid saying some word, everyone knows to do it, and the only way to seem more virtuous than other people is to actually be virtuous. The shallow, complicated, and frequently changing rules of political correctness made it the perfect substitute for actual virtue. And the result was a world in which good people who weren't up to date on current moral fashions were brought down by people whose characters would make you recoil in horror if you could see them.

One big contributing factor in the rise of political correctness was the lack of other things to be morally pure about. Previous generations of prigs had been prigs mostly about religion and sex. But among the cultural elite these were the deadest of dead letters by the 1980s; if you were religious, or a virgin, this was something you tended to conceal rather than advertise. So the sort of people who enjoy being moral enforcers had become starved of things to enforce. A new set of rules was just what they'd been waiting for.

Curiously enough, the tolerant side of the 1960s left helped create the conditions



in which the intolerant side prevailed. The relaxed social rules advocated by the old, easy-going hippy left became the dominant ones, at least among the elite, and this left nothing for the naturally intolerant to be intolerant about.

Another possibly contributing factor was the fall of the Soviet empire. Marxism had been a popular focus of moral purity on the left before political correctness emerged as a competitor, but the pro-democracy movements in Eastern Bloc countries took most of the shine off it. Especially the fall of the Berlin Wall in 1989. You couldn't be on the side of the Stasi. I remember looking at the moribund Soviet Studies section of a used bookshop in Cambridge in the late 1980s and thinking "what will those people go on about now?" As it turned out the answer was right under my nose.

One thing I noticed at the time about the first phase of political correctness was that it was more popular with women than men. As many writers (perhaps most eloquently George Orwell) have observed, women seem more attracted than men to the idea of being moral enforcers. But there was another more specific reason women tended to be the enforcers of political correctness. There was at this time a great backlash against sexual harassment; the mid 1980s were the point when the definition of sexual harassment was expanded from explicit sexual advances to creating a "hostile environment." Within universities the classic form of accusation was for a (female) student to say that a professor made her "feel uncomfortable." But the vagueness of this accusation allowed the radius of forbidden behavior to expand to include talking about heterodox ideas. Those make people uncomfortable too. [5]

Was it sexist to propose that Darwin's greater male variability hypothesis might explain some variation in human performance? Sexist enough to get Larry Summers pushed out as president of Harvard, apparently. One woman who heard the talk in which he mentioned this idea said it made her feel "physically ill" and that she had to leave halfway through. If the test of a hostile environment is how it makes people feel, this certainly sounds like one. And yet it does seem plausible that greater male variability explains some of the variation in human performance. So which should prevail, comfort or truth? Surely if truth should prevail anywhere, it should be in universities; that's supposed to be their specialty; but for decades starting in the late 1980s the politically correct tried to pretend this conflict didn't exist. [6]

Political correctness seemed to burn out in the second half of the 1990s. One reason, perhaps the main reason, was that it literally became a joke. It offered rich material for comedians, who performed their usual disinfectant action upon it. Humor is one of the most powerful weapons against priggishness of any sort, because prigs, being humorless, can't respond in kind. Humor was what defeated Victorian prudishness, and by 2000 it seemed to have done the same thing to political correctness.

Unfortunately this was an illusion. Within universities the embers of political correctness were still glowing brightly. After all, the forces that created it were still

there. The professors who started it were now becoming deans and department heads. And in addition to their departments there were now a bunch of new ones explicitly focused on social justice. Students were still hungry for things to be morally pure about. And there had been an explosion in the number of university administrators, many of whose jobs involved enforcing various forms of political correctness.

In the early 2010s the embers of political correctness burst into flame anew. There were several differences between this new phase and the original one. It was more virulent. It spread further into the real world, although it still burned hottest within universities. And it was concerned with a wider variety of sins. In the first phase of political correctness there were really only three things people got accused of: sexism, racism, and homophobia (which at the time was a neologism invented for the purpose). But between then and 2010 a lot of people had spent a lot of time trying to invent new kinds of -isms and -phobias and seeing which could be made to stick.

The second phase was, in multiple senses, political correctness metastasized. Why did it happen when it did? My guess is that it was due to the rise of social media, particularly Tumblr and Twitter, because one of the most distinctive features of the second wave of political correctness was the *cancel mob*: a mob of angry people uniting on social media to get someone ostracized or fired. Indeed this second wave of political correctness was originally called "cancel culture"; it didn't start to be called "wokeness" till the 2020s.

One aspect of social media that surprised almost everyone at first was the popularity of outrage. Users seemed to *like* being outraged. We're so used to this idea now that we take it for granted, but really it's pretty strange. Being outraged is not a pleasant feeling. You wouldn't expect people to seek it out. But they do. And above all, they want to share it. I happened to be running a forum from 2007 to 2014, so I can actually quantify how much they want to share it: our users were about three times more likely to upvote something if it outraged them.

This tilt toward outrage wasn't due to wokeness. It's an inherent feature of social media, or at least this generation of it. But it did make social media the perfect mechanism for fanning the flames of wokeness. [7]

It wasn't just public social networks that drove the rise of wokeness though. Group chat apps were also critical, especially in the final step, cancellation. Imagine if a group of employees trying to get someone fired had to do it using only email. It would be hard to organize a mob. But once you have group chat, mobs form naturally.

Another contributing factor in this second wave of political correctness was the dramatic increase in the polarization of the press. In the print era, newspapers were constrained to be, or at least seem, politically neutral. The department stores that ran ads in the New York Times wanted to reach everyone in the region, both liberal and conservative, so the Times had to serve both. But the Times didn't

regard this neutrality as something forced upon them. They embraced it as their duty as a *paper of record* — as one of the big newspapers that aimed to be chronicles of their times, reporting every sufficiently important story from a neutral point of view.

When I grew up the papers of record seemed timeless, almost sacred institutions. Papers like the New York Times and Washington Post had immense prestige, partly because other sources of news were limited, but also because they did make some effort to be neutral.

Unfortunately it turned out that the paper of record was mostly an artifact of the constraints imposed by print. [8] When your market was determined by geography, you had to be neutral. But publishing online enabled — in fact probably forced — newspapers to switch to serving markets defined by ideology instead of geography. Most that remained in business fell in the direction they'd already been leaning: left. On October 11, 2020 the New York Times announced that "The paper is in the midst of an evolution from the stodgy paper of record into a juicy collection of great narratives." [9] Meanwhile journalists, of a sort, had arisen to serve the right as well. And so journalism, which in the previous era had been one of the great centralizing forces, now became one of the great polarizing ones.

The rise of social media and the increasing polarization of journalism reinforced one another. In fact there arose a new variety of journalism involving a loop through social media. Someone would say something controversial on social media. Within hours it would become a news story. Outraged readers would then post links to the story on social media, driving further arguments online. It was the cheapest source of clicks imaginable. You didn't have to maintain overseas news bureaus or pay for month-long investigations. All you had to do was watch Twitter for controversial remarks and repost them on your site, with some additional comments to inflame readers further.

For the press there was money in wokeness. But they weren't the only ones. That was one of the biggest differences between the two waves of political correctness: the first was driven almost entirely by amateurs, but the second was often driven by professionals. For some it was their whole job. By 2010 a new class of administrators had arisen whose job was basically to enforce wokeness. They played a role similar to that of the political commissars who got attached to military and industrial organizations in the USSR: they weren't directly in the flow of the organization's work, but watched from the side to ensure that nothing improper happened in the doing of it. These new administrators could often be recognized by the word "inclusion" in their titles. Within institutions this was the preferred euphemism for wokeness; a new list of banned words, for example, would usually be called an "inclusive language guide." [10]

This new class of bureaucrats pursued a woke agenda as if their jobs depended on it, because they did. If you hire people to keep watch for a particular type of problem, they're going to find it, because otherwise there's no justification for their existence. [11] But these bureaucrats also represented a second and possibly even

greater danger. Many were involved in hiring, and when possible they tried to ensure their employers hired only people who shared their political beliefs. The most egregious cases were the new "DEI statements" that some universities started to require from faculty candidates, proving their commitment to wokeness. Some universities used these statements as the initial filter and only even considered candidates who scored high enough on them. You're not hiring Einstein that way; imagine what you get instead.

Another factor in the rise of wokeness was the Black Lives Matter movement, which started in 2013 when a white man was acquitted after killing a black teenager in Florida. But this didn't launch wokeness; it was well underway by 2013.

Similarly for the Me Too Movement, which took off in 2017 after the first news stories about Harvey Weinstein's history of raping women. It accelerated wokeness, but didn't play the same role in launching it that the 80s version did in launching political correctness.

The election of Donald Trump in 2016 also accelerated wokeness, particularly in the press, where outrage now meant traffic. Trump made the New York Times a lot of money: headlines during his first administration mentioned his name at about four times the rate of previous presidents.

In 2020 we saw the biggest accelerant of all, after a white police officer asphyxiated a black suspect on video. At this point the metaphorical fire became a literal one, as violent protests broke out across America. But in retrospect this turned out to be peak woke, or close to it. By every measure I've seen, wokeness peaked in 2020 or 2021.

Wokeness is sometimes described as a mind-virus. What makes it viral is that it defines new types of impropriety. Most people are afraid of impropriety; they're never exactly sure what the social rules are or which ones they might be breaking. Especially if the rules change rapidly. And since most people already worry that they might be breaking rules they don't know about, if you tell them they're breaking a rule, their default reaction is to believe you. Especially if multiple people tell them. Which in turn is a recipe for exponential growth. Zealots invent some new impropriety to avoid. The first people to adopt it are fellow zealots, eager for new ways to signal their virtue. If there are enough of these, the initial group of zealots is followed by a much larger group, motivated by fear. They're not trying to signal virtue; they're just trying to avoid getting in trouble. At this point the new impropriety is now firmly established. Plus its success has increased the rate of change in social rules, which, remember, is one of the reasons people are nervous about which rules they might be breaking. So the cycle accelerates. [\[12\]](#)

What's true of individuals is even more true of organizations. Especially organizations without a powerful leader. Such organizations do everything based on "best practices." There's no higher authority; if some new "best practice" achieves critical mass, they *must* adopt it. And in this case the organization can't

do what it usually does when it's uncertain: delay. It might be committing improprieties right now! So it's surprisingly easy for a small group of zealots to capture this type of organization by describing new improprieties it might be guilty of. [\[13\]](#)

How does this kind of cycle ever end? Eventually it leads to disaster, and people start to say enough is enough. The excesses of 2020 made a lot of people say that.

Since then wokeness has been in gradual but continual retreat. Corporate CEOs, starting with Brian Armstrong, have openly rejected it. Universities, led by the University of Chicago and MIT, have explicitly confirmed their commitment to free speech. Twitter, which was arguably the hub of wokeness, was bought by Elon Musk in order to neutralize it, and he seems to have succeeded — and not, incidentally, by censoring left-wing users the way Twitter used to censor right-wing ones, but without censoring either. [\[14\]](#) Consumers have emphatically rejected brands that ventured too far into wokeness. The Bud Light brand may have been permanently damaged by it. I'm not going to claim Trump's second victory in 2024 was a referendum on wokeness; I think he won, as presidential candidates always do, because he was more [charismatic](#); but voters' disgust with wokeness must have helped.

So what do we do now? Wokeness is already in retreat. Obviously we should help it along. What's the best way to do that? And more importantly, how do we avoid a third outbreak? After all, it seemed to be dead once, but came back worse than ever.

In fact there's an even more ambitious goal: is there a way to prevent any similar outbreak of aggressively performative moralism in the future — not just a third outbreak of political correctness, but the next thing like it? Because there will be a next thing. Prigs are prigs by nature. They need rules to obey and enforce, and now that Darwin has cut off their traditional supply of rules, they're constantly hungry for new ones. All they need is someone to meet them halfway by defining a new way to be morally pure, and we'll see the same phenomenon again.

Let's start with the easier problem. Is there a simple, principled way to deal with wokeness? I think there is: to use the customs we already have for dealing with religion. Wokeness is effectively a religion, just with God replaced by protected classes. It's not even the first religion of this kind; Marxism had a similar form, with God replaced by the masses. [\[15\]](#) And we already have well-established customs for dealing with religion within organizations. You can express your own religious identity and explain your beliefs, but you can't call your coworkers infidels if they disagree, or try to ban them from saying things that contradict its doctrines, or insist that the organization adopt yours as its official religion.

If we're not sure what to do about any particular manifestation of wokeness, imagine we were dealing with some other religion, like Christianity. Should we have people within organizations whose jobs are to enforce woke orthodoxy? No, because we wouldn't have people whose jobs were to enforce Christian orthodoxy.

Should we censor [writers](#) or [scientists](#) whose work contradicts woke doctrines? No, because we wouldn't do this to people whose work contradicted Christian teachings. Should job candidates be required to write DEI statements? Of course not; imagine an employer requiring proof of one's religious beliefs. Should students and employees have to participate in woke indoctrination sessions in which they're required to answer questions about their beliefs to ensure compliance? No, because we wouldn't dream of catechizing people in this way about their religion.

[16]

One shouldn't feel bad about not wanting to watch woke movies any more than one would feel bad about not wanting to listen to Christian rock. In my twenties I drove across America several times, listening to local radio stations. Occasionally I'd turn the dial and hear some new song. But the moment anyone mentioned Jesus I'd turn the dial again. Even the tiniest bit of being preached to was enough to make me lose interest.

But by the same token we should not automatically reject everything the woke believe. I'm not a Christian, but I can see that many Christian principles are good ones. It would be a mistake to discard them all just because one didn't share the religion that espoused them. It would be the sort of thing a religious zealot would do.

If we have genuine pluralism, I think we'll be safe from future outbreaks of woke intolerance. Wokeness itself won't go away. There will for the foreseeable future continue to be pockets of woke zealots inventing new moral fashions. The key is not to let them treat their fashions as normative. They can change what their coreligionists are allowed to say every few months if they like, but they mustn't be allowed to change what we're allowed to say. [17]

The more general problem — how to prevent similar outbreaks of aggressively performative moralism — is of course harder. Here we're up against human nature. There will always be prigs. And in particular there will always be the enforcers among them, the [aggressively conventional-minded](#). These people are born that way. Every society has them. So the best we can do is to keep them bottled up.

The aggressively conventional-minded aren't always on the rampage. Usually they just enforce whatever random rules are nearest to hand. They only become dangerous when some new ideology gets a lot of them pointed in the same direction at once. That's what happened during the Cultural Revolution, and to a lesser extent (thank God) in the two waves of political correctness we've experienced.

We can't get rid of the aggressively conventional-minded. [18] And we couldn't prevent people from creating new ideologies that appealed to them even if we wanted to. So if we want to keep them bottled up, we have to do it one step downstream. Fortunately when the aggressively conventional-minded go on the rampage they always do one thing that gives them away: they define new [heresies](#) to punish people for. So the best way to protect ourselves from future outbreaks of

things like wokeness is to have powerful antibodies against the concept of heresy.

We should have a conscious bias against defining new forms of heresy. Whenever anyone tries to ban saying something that we'd previously been able to say, our initial assumption should be that they're wrong. Only our initial assumption of course. If they can prove we should stop saying it, then we should. But the burden of proof is on them. In liberal democracies, people trying to prevent something from being said will usually claim they're not merely engaging in censorship, but trying to prevent some form of "harm". And maybe they're right. But once again, the burden of proof is on them. It's not enough to claim harm; they have to prove it.

As long as the aggressively conventional-minded continue to give themselves away by banning heresies, we'll always be able to notice when they become aligned behind some new ideology. And if we always fight back at that point, with any luck we can stop them in their tracks.

The number of true things we [can't say](#) should not increase. If it does, something is wrong.

## Notes

[1] Why did 1960s radicals focus on the causes they did? One of the people who reviewed drafts of this essay explained this so well that I asked if I could quote him:

The middle-class student protestors of the New Left rejected the socialist/Marxist left as unhip. They were interested in sexier forms of oppression uncovered by cultural analysis (Marcuse) and abstruse "Theory". Labor politics became stodgy and old-fashioned. This took a couple generations to work through. The woke ideology's conspicuous lack of interest in the working class is the tell-tale sign. Such fragments as are, er, left of the old left are anti-woke, and meanwhile the actual working class shifted to the populist right and gave us Trump. Trump and wokeness are cousins.

The middle-class origins of wokeness smoothed its way through the institutions because it had no interest in "seizing the means of production" (how quaint such phrases seem now), which would quickly have run up against hard state and corporate power. The fact that wokeness only expressed interest in other kinds of class (race, sex, etc) signalled compromise with existing power: give us power within



your system and we'll bestow the resource we control — moral rectitude — upon you. As an ideological stalking horse for gaining control over discourse and institutions, this succeeded where a more ambitious revolutionary program would not have.

[2] It helped that the humanities and social sciences also included some of the biggest and easiest undergrad majors. If a political movement had to start with physics students, it could never get off the ground; there would be too few of them, and they wouldn't have the time to spare.

At the top universities these majors are not as big as they used to be, though. A [2022 survey](#) found that only 7% of Harvard undergrads plan to major in the humanities, vs nearly 30% during the 1970s. I expect wokeness is at least part of the reason; when undergrads consider majoring in English, it's presumably because they love the written word and not because they want to listen to lectures about racism.

[3] The puppet-master-and-puppet character of political correctness became clearly visible when a bakery near Oberlin College was falsely accused of race discrimination in 2016. In the subsequent civil trial, lawyers for the bakery produced a text message from Oberlin Dean of Students Meredith Raimondo that read "I'd say unleash the students if I wasn't convinced this needs to be put behind us."

[4] The woke sometimes claim that wokeness is simply treating people with respect. But if it were, that would be the only rule you'd have to remember, and this is comically far from being the case. My younger son likes to imitate voices, and at one point when he was about seven I had to explain which accents it was currently safe to imitate publicly and which not. It took about ten minutes, and I still hadn't covered all the cases.

[5] In 1986 the Supreme Court ruled that creating a hostile work environment could constitute sex discrimination, which in turn affected universities via Title IX. The court specified that the test of a hostile environment was whether it would bother a reasonable person, but since for a professor merely being the subject of a sexual harassment complaint would be a disaster whether the complainant was reasonable or not, in practice any joke or remark remotely connected with sex was now effectively forbidden. Which meant we'd now come full circle to Victorian codes of behavior, when there was a large class of things that might not be said "with ladies present."

[6] Much as they tried to pretend there was no conflict between diversity and quality. But you can't simultaneously optimize for two things that aren't identical. What diversity actually means, judging from the way the term is used, is proportional representation, and unless you're selecting a group whose purpose is to be representative, like poll respondents, optimizing for proportional representation has to come at the expense of quality. This is not because of anything about representation; it's the nature of optimization; optimizing for x has to come at the expense of y unless x and y are identical.



[7] Maybe societies will eventually develop antibodies to viral outrage. Maybe we were just the first to be exposed to it, so it tore through us like an epidemic through a previously isolated population. I'm fairly confident that it would be possible to create new social media apps that were less driven by outrage, and an app of this type would have a good chance of stealing users from existing ones, because the smartest people would tend to migrate to it.

[8] I say "mostly" because I have hopes that journalistic neutrality will return in some form. There is some market for unbiased news, and while it may be small, it's valuable. The rich and powerful want to know what's really going on; that's how they became rich and powerful.

[9] The Times made this momentous announcement very informally, in passing in the middle of an [article](#) about a Times reporter who'd been criticized for inaccuracy. It's quite possible no senior editor even approved it. But it's somehow appropriate that this particular universe ended with a whimper rather than a bang.

[10] As the acronym DEI goes out of fashion, many of these bureaucrats will try to go underground by changing their titles. It looks like "belonging" will be a popular option.

[11] If you've ever wondered why our legal system includes protections like the separation of prosecutor, judge, and jury, the right to examine evidence and cross-examine witnesses, and the right to be represented by legal counsel, the de facto [parallel legal system](#) established by Title IX makes that all too clear.

[12] The invention of new improprieties is most visible in the rapid evolution of woke nomenclature. This is particularly annoying to me as a writer, because the new names are always worse. Any religious observance has to be inconvenient and slightly absurd; otherwise gentiles would do it too. So "slaves" becomes "enslaved individuals." But web search can show us the leading edge of moral growth in real time: if you search for "individuals experiencing slavery" you will as of this writing find five legit attempts to use the phrase, and you'll even find two for "individuals experiencing enslavement."

[13] Organizations that do dubious things are particularly concerned with propriety, which is how you end up with absurdities like tobacco and oil companies having higher ESG ratings than Tesla.

[14] Elon did something else that tilted Twitter rightward though: he gave more visibility to paying users. Paying users lean right on average, because people on the far left dislike Elon and don't want to give him money. Elon presumably knew this would happen. On the other hand, the people on the far left have only themselves to blame; they could tilt Twitter back to the left tomorrow if they wanted to.

[15] It even, as James Lindsay and Peter Boghossian pointed out, has a concept of

original sin: privilege. Which means unlike Christianity's egalitarian version, people have varying degrees of it. An able-bodied straight white American male is born with such a load of sin that only by the most abject repentance can he be saved.

Wokeness also shares something rather funny with many actual versions of Christianity: like God, the people for whose sake wokeness purports to act are often revolted by the things done in their name.

[16] There is one exception to most of these rules: actual religious organizations. It's reasonable for them to insist on orthodoxy. But they in turn should declare that they're religious organizations. It's rightly considered shady when something that appears to be an ordinary business or publication turns out to be a religious organization.

[17] I don't want to give the impression that it will be simple to roll back wokeness. There will be places where the fight inevitably gets messy — particularly within universities, which everyone has to share, yet which are currently the most pervaded by wokeness of any institutions.

[18] You can however get rid of aggressively conventional-minded people within an organization, and in many if not most organizations this would be an excellent idea. Even a handful of them can do a lot of damage. I bet you'd feel a noticeable improvement going from a handful to none.

**Thanks** to Sam Altman, Ben Miller, Daniel Gackle, Robin Hanson, Jessica Livingston, Greg Lukianoff, Harj Taggar, Garry Tan, and Tim Urban for reading drafts of this.

# What to Do

March 2025

What should one do? That may seem a strange question, but it's not meaningless or unanswerable. It's the sort of question kids ask before they learn not to ask big questions. I only came across it myself in the process of investigating something else. But once I did, I thought I should at least try to answer it.

So what *should* one do? One should help people, and take care of the world. Those two are obvious. But is there anything else? When I ask that, the answer that pops up is *Make good new things*.

I can't prove that one should do this, any more than I can prove that one should help people or take care of the world. We're talking about first principles here. But I can explain why this principle makes sense. The most impressive thing humans can do is to think. It may be the most impressive thing that can be done. And the best kind of thinking, or more precisely the best proof that one has thought well, is to make good new things.

I mean new things in a very general sense. Newton's physics was a good new thing. Indeed, the first version of this principle was to have good new ideas. But that didn't seem general enough: it didn't include making art or music, for example, except insofar as they embody new ideas. And while they may embody new ideas, that's not all they embody, unless you stretch the word "idea" so uselessly thin that it includes everything that goes through your nervous system.

Even for ideas that one has consciously, though, I prefer the phrasing "make good new things." There are other ways to describe the best kind of thinking. To make discoveries, for example, or to understand something more deeply than others have. But how well do you understand something if you can't make a model of it, or write about it? Indeed, trying to express what you understand is not just a way to prove that you understand it, but a way to understand it better.

Another reason I like this phrasing is that it biases us toward creation. It causes us to prefer the kind of ideas that are naturally seen as making things rather than, say, making critical observations about things other people have made. Those are ideas too, and sometimes valuable ones, but it's easy to trick oneself into believing they're more valuable than they are. Criticism seems sophisticated, and making new things often seems awkward, especially at first; and yet it's precisely those

first steps that are most rare and valuable.

Is newness essential? I think so. Obviously it's essential in science. If you copied a paper of someone else's and published it as your own, it would seem not merely unimpressive but dishonest. And it's similar in the arts. A copy of a good painting can be a pleasing thing, but it's not impressive in the way the original was. Which in turn implies it's not impressive to make the same thing over and over, however well; you're just copying yourself.

Note though that we're talking about a different kind of should with this principle. Taking care of people and the world are shoulds in the sense that they're one's duty, but making good new things is a should in the sense that this is how to live to one's full potential. Historically most rules about how to live have been a mix of both kinds of should, though usually with more of the former than the latter. [1]

For most of history the question "What should one do?" got much the same answer everywhere, whether you asked Cicero or Confucius. You should be wise, brave, honest, temperate, and just, uphold tradition, and serve the public interest. There was a long stretch where in some parts of the world the answer became "Serve God," but in practice it was still considered good to be wise, brave, honest, temperate, and just, uphold tradition, and serve the public interest. And indeed this recipe would have seemed right to most Victorians. But there's nothing in it about taking care of the world or making new things, and that's a bit worrying, because it seems like this question should be a timeless one. The answer shouldn't change much.

I'm not too worried that the traditional answers don't mention taking care of the world. Obviously people only started to care about that once it became clear we could ruin it. But how can making good new things be important if the traditional answers don't mention it?

The traditional answers were answers to a slightly different question. They were answers to the question of how to be, rather than what to do. The audience didn't have a lot of choice about what to do. The audience up till recent centuries was the landowning class, which was also the political class. They weren't choosing between doing physics and writing novels. Their work was foreordained: manage their estates, participate in politics, fight when necessary. It was ok to do certain other kinds of work in one's spare time, but ideally one didn't have any. Cicero's *De Officiis* is one of the great classical answers to the question of how to live, and in it he explicitly says that he wouldn't even be writing it if he hadn't been excluded from public life by recent political upheavals. [2]

There were of course people doing what we would now call "original work," and they were often admired for it, but they weren't seen as models. Archimedes knew that he was the first to prove that a sphere has  $\frac{2}{3}$  the volume of the smallest enclosing cylinder and was very pleased about it. But you don't find ancient writers urging their readers to emulate him. They regarded him more as a prodigy than a model.

Now many more of us can follow Archimedes's example and devote most of our attention to one kind of work. He turned out to be a model after all, along with a collection of other people that his contemporaries would have found it strange to treat as a distinct group, because the vein of people making new things ran at right angles to the social hierarchy.

What kinds of new things count? I'd rather leave that question to the makers of them. It would be a risky business to try to define any kind of threshold, because new kinds of work are often despised at first. Raymond Chandler was writing literal pulp fiction, and he's now recognized as one of the best writers of the twentieth century. Indeed this pattern is so common that you can use it as a recipe: if you're excited about some kind of work that's not considered prestigious and you can explain what everyone else is overlooking about it, then this is not merely a kind of work that's ok to do, but one to seek out.

The other reason I wouldn't want to define any thresholds is that we don't need them. The kind of people who make good new things don't need rules to keep them honest.

So there's my guess at a set of principles to live by: take care of people and the world, and make good new things. Different people will do these to varying degrees. There will presumably be lots who focus entirely on taking care of people. There will be a few who focus mostly on making new things. But even if you're one of those, you should at least make sure that the new things you make don't net *harm* people or the world. And if you go a step further and try to make things that help them, you may find you're ahead on the trade. You'll be more constrained in what you can make, but you'll make it with more energy.

On the other hand, if you make something amazing, you'll often be helping people or the world even if you didn't mean to. Newton was driven by curiosity and ambition, not by any practical effect his work might have, and yet the practical effect of his work has been enormous. And this seems the rule rather than the exception. So if you think you can make something amazing, you should probably just go ahead and do it.

[1] We could treat all three as the same kind of should by saying that it's one's duty to live well — for example by saying, as some Christians have, that it's one's duty to make the most of one's God-given gifts. But this seems one of those casuistries people invented to evade the stern requirements of religion: you could spend time studying math instead of praying or performing acts of charity because otherwise you were rejecting a gift God had given you. A useful casuistry no doubt, but we don't need it.

We could also combine the first two principles, since people are part of the world. Why should our species get special treatment? I won't try to justify this choice, but I'm skeptical that anyone who claims to think differently actually lives according to their principles.

[2] Confucius was also excluded from public life after ending up on the losing end of a power struggle, and presumably he too would not be so famous now if it hadn't been for this long stretch of enforced leisure.

**Thanks** to Trevor Blackwell, Jessica Livingston, and Robert Morris for reading drafts of this.

# THE END

